MUSIC/SP

Version 5

Release 1

Office Applications Guide

April 1996

First Edition (April 1996)

This edition applies to Release 1 of Multi-User System for Interactive Computing / System Product (MUSIC/SP) Version 5, and to all releases of this product until otherwise indicated in new editions or Technical Newsletters. MUSIC/SP Version 5 is published and licensed by McGill Systems Inc.

A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be addressed to: MUSIC Product Group, McGill Systems Inc., 550 Sherbrooke St. West, Suite 1650, Montreal, Quebec, Canada H3A 1B9. Fax: (514) 398-4488.

Preface

About this Guide

This publication describes how to use MUSIC/SP for office applications. Emphasis throughout this guide is placed on the TODO facility (Time, Office, and Documentation Organizer) of MUSIC/SP. TODO is intended for office professionals and other people who can benefit from the time management and word processing functions it provides. TODO operates under control of MUSIC/SP by means of menus, that simplify the use of the system.

When the term *MUSIC* is used in this publication, it refers to MUSIC/SP.

Trademarks

Chapter Outlines

Chapter 1 - Introduction

The introduction gives you a background to the MUSIC/SP (MUSIC) system in general, along with useful information for: signing on, interpreting screen messages, and getting help.

Chapter 2 - Menu Facilities

This chapter describes the menu facilities that are provided with MUSIC/SP. Full-Screen Interface (FSI) and Time, Office, and Documentation Organizer (TODO) is introduced. The optional IBM DisplayWrite/370 program is also described in this chapter.

Chapter 3 - MUSIC Editor

This chapter describes the Editor program for creating and revising files on MUSIC/SP. Full-screen operation of the Editor and Editor commands are described here.

Chapter 4 - Electronic Mail

The Electronic Mail Facility for sending and receiving mail is described in this chapter. Also, the Mail Profile and Mail Directory programs are described here.

Chapter 5 - MUSIC/SCRIPT

MUSIC/SCRIPT (SCRIPT), a word processing package on MUSIC, is described in detail in this chapter. All the necessary formatting instructions are included for designing high quality documents.

Chapter 6 - TODO Menu Items

This chapter describes TODO menu items (options). These functions of TODO include: The EXECUTE option for running documents through the SCRIPT program to produce formatted text; SUBMIT function for sending SCRIPT documents to remote printers; SCHEDULES for scheduling individual calendars, conference rooms or equipment; SCHEDULE MEETINGS for checking calendars and scheduling meetings; TELEPHONE LOG for recording telephone messages; CALCULATOR for solving routine calculations and complicated equations; SPELLING CHECK for pointing to and correcting spelling errors in full-screen mode; and LIBRARY for producing a listing of all the user's files on MUSIC.

Chapter 7 - Utilities

Additional features of TODO are included in a separate menu called "Utilities". The items on this menu include: NAMES facility for setting up nicknames to use with Electronic Mail and Schedule; PROFILE program for displaying and updating userid information; CONTENTS utility for creating a table of contents for SCRIPT documents; INDEX utility for creating indices.

Also included in this chapter, is the documentation for the REMIND and TMENU programs. REMIND is used for setting reminders with TODO and TMENU is used for creating and updating menus, like the TODO menu.

Appendixes

Appendixes A - C include miscellaneous items of interest to TODO users. These items are: Using SCRIPT without TODO, Sample editing session for non-full-screen, and common error messages. Appendixes D and E are of interest to systems administrators who wish to make changes to TODO. Appendix D discusses how to initialize conference rooms and equipment for the TODO menu items SCHEDULES and SCHEDULE MEETINGS. Appendix E shows the CREATE program in detail, highlighting the areas that could be changed.

MUSIC/SP Publications

The following is a list of all the current MUSIC/SP publications. These hardcopy publications can be ordered through the MUSIC Product Group. Online versions (softcopy) of the user publications can be accessed with the MUSIC/SP command called "MAN".

- *MUSIC/SP Administrator's Guide* (April 1996), describes how to install and operate MUSIC/SP.
- *MUSIC/SP Administrator's Reference* (April 1996), describes the internals of MUSIC/SP; utility programs and supervisory commands; gives detailed storage estimates; and documents console messages.

- *MUSIC/SP User's Reference Guide* (April 1996), describes how to use MUSIC/SP; its command language; terminal and batch set up; and job processing using the various language processors.

- *MUSIC/SP Guide for New Users* (April 1996), introduces new users to the use of MUSIC/SP via an IBM 3270-type workstation. It describes the FSI (Full Screen Interface) menu facility. New users learn how to use many programs on MUSIC/SP for such tasks as editing and running programs.

- *MUSIC/SP Office Applications Guide* (April 1996), describes the features of the TODO (Time, Office, and Documentation Organizer) facility. This includes the scheduling function, spell checking, and MUSIC/SCRIPT (text processing).

- *MUSIC/SP Mail and Conferencing Guide* (April 1996), describes electronic mail on MUSIC/SP. This includes Mail Profile, Mail Directory, using POP clients, and conferencing programs.

- *MUSIC/SP Internet Guide* (April 1996), describes the programs available on MUSIC/SP that provide communication between users through electronic conferencing and discussion lists. Emphasis is placed on access to the Internet with programs such as TELNET (logging on other computers), FTP (File Transfer Protocol), WEB (World-Wide Web), RN (Newsreader), and GOPHER (document search and retrieval protocol).

- *MUSIC/SP Campus-Wide Information Systems* (*CWIS*) *Guide* (April 1996), describes how to create and maintain a Campus-Wide Information System, Help facility, or Classified Ads facility; how to do full-text searching; and how to provide gopher access. MUSIC/SP's resources are used to provide online distribution of information to a wide audience.

- *MUSIC/SP Teacher's Guide* (April 1996), describes various MUSIC/SP facilities related to the academic environment. Emphasis is placed on communication between teacher and student and easy methods for learning how to use MUSIC applications.

- *MUSIC/SP Client/Server* (*MCS*) *Booklet* (April 1996) provides an overview of MCS. Full documentation is available on the MCS diskette.

- *MUSIC/SP Personal Computer Workstation User's Guide* (May 1994), describes the components of the Personal Computer Workstation (PCWS). It is intended for the novice or experienced user of a personal computer, who wishes to connect to MUSIC/SP or another host system. Note that documentation for *PCWS for Windows* is available on the PCWS diskette.

Chapter 1. Introduction

Introduction

What is MUSIC?

MUSIC is a time-sharing system that allows a number of users to access the computer concurrently. Typically, the user is connected to the computer by a typewriter-like device called a *workstation*. At the workstation, you can interact with the system directly, telling it what to do. This *interactive* system, in turn, responds to that request at your workstation.

The MUSIC system can be used for many different purposes. For example, word processing, programming, and electronic mail. The fact that MUSIC and its connected workstations can be used for different purposes by many people simultaneously, need not be a concern. Each person is treated as if they had the computer to themselves. The figure below shows a simple configuration of the MUSIC system.

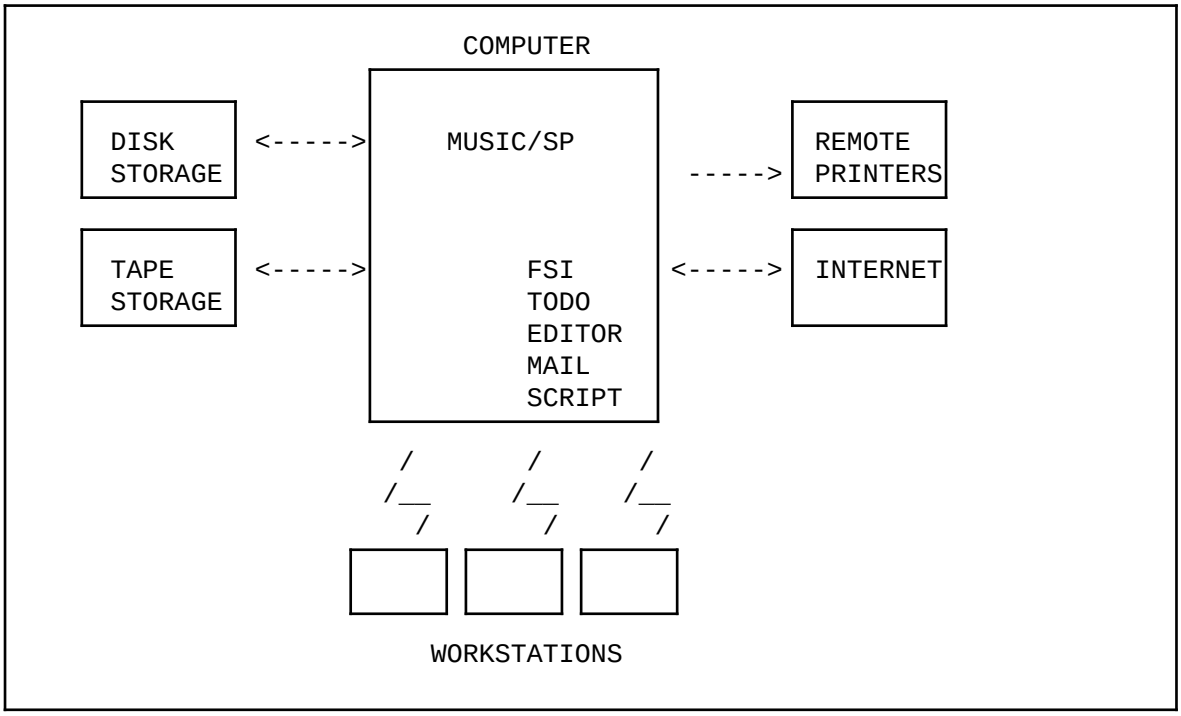


Figure 1.1 - Configuration of the MUSIC System.

MUSIC is an operating system that acts like a middleman between you and the computer's resources. It is stored internally in the computer's memory. MUSIC has many programs to help you use the computer productively. Menu-driven facilities on MUSIC provide an easy method of accessing the most common programs such as the Editor and MAIL. Full-Screen Interface (FSI) and Time, Office and Documentation Organizer (TODO) are two menu facilities available to you. FSI, TODO, the Editor, MAIL, and SCRIPT are all programs or *software* stored online on disk.

Some of the equipment (*hardware*) attached to the computer are: disk packs for online storage of your documents; tape drives for offline storage; batch printers at the central site for large volumes; and a variety of different types of workstations.

MUSIC supports many types of workstations. These include personal computers, 3270-type terminals, and ASCII terminals.

Commands typed in by the user are used to instruct MUSIC to do different things. Items on the TODO menu are examples of issuing commands to MUSIC. (Refer to Chapter 2 for information about the TODO menu.) MUSIC commands can also be typed in the SELECT OPTION area of the menu. For more information, read the MUSIC COMMANDS section.

Menu Facilities

MUSIC/SP includes menu facilities that are designed to provide easy access to MUSIC programs.

FSI (Full-Screen Interface) provides in interface for general use of the MUSIC system. It provides access to common functions such as electronic mail, submitting and printing files, and file management. This interface is described in *Chapter 2. Menu Facilities*.

TODO (Time, Office, and Documentation Organizer) is an office menu system that runs under MUSIC/SP (Multi-User System for Interactive Computing/System Product). TODO is designed for office professionals from management to secretarial staff. Menus can be tailored to suit individual needs. A knowledge of programming is not needed to use the TODO system. This interface is described in *Chapter 2. Menu Facilities* and in *Chapter 6. TODO Menu Items*.

Much of this guide is devoted to non-programming applications. The TODO facility is referred to in many of the chapters and is assumed to be the method used for accessing programs discussed in this manual. Programmers may wish to refer to the *MUSIC/SP User's Reference Guide* for details about programming on MUSIC.

What are Files?

Files are collections of typed lines or records. Word processing users often use the words *file* and *document* to mean the same thing.

Each user can store many files on MUSIC. Each file is given a file name that is used to identify it. Users can ask MUSIC to display a list of the names of all the files they own. This is called a library listing. For example, item "L" in the TODO menu performs this task.

How are Files Stored?

Your files are stored in the MUSIC Save Library on disk under your userid. You simply issue the command FILE after creating or revising a document to store it. To retrieve a document (file) use the name that you gave it.

If you wish, your files can be archived onto magnetic tape for backup or transport. Refer to *MUSIC/SP User's Reference Guide* for information about archiving files.

What is the Editor?

The Editor is the MUSIC program that is used to create and revise files. (These files can contain the text of your document or programs.) Often we just use the term edit to refer to the process of revising (or editing) the file using the Editor.

Word Processing

MUSIC/SP supports three word processing programs. MUSIC/SCRIPT, IBM DisplayWrite/370, and Waterloo SCRIPT (WATCOM Products Inc.). MUSIC/SCRIPT is included with the MUSIC/SP system and is described in detail in this manual.

DisplayWrite/370 is an optional IBM program that may be available at your installation. This program is a host-based text editor and formatter. Information about using this program can be found in *Chapter 2. Menu Facilities.*

Waterloo SCRIPT is a powerful and versatile text formatter, written and distributed by the University of Waterloo. Some of the features of Waterloo SCRIPT include proportional spacing, the ability to generate boxes around text, footnotes, producing multiple columns of text on a single page, creating indexes, and hyphenation. Waterloo SCRIPT supports many popular output devices ranging from simple printing terminals to sophisticated laser printers.

What is MUSIC/SCRIPT?

MUSIC/SCRIPT is the program used to format documents from files that are prepared using the Editor. MUSIC/SCRIPT documents consist of lines containing the text that make up the final document, and lines that contain formatting instructions. These formatting instructions are called control words. The control words are used only to inform MUSIC/SCRIPT on how to format the text and are never displayed in the final output document. (All control words are described in *Chapter 5 - MUSIC/SCRIPT.*)

Other programs are provided with MUSIC to aid the word processing user. These utility programs include: ones to perform spelling checks, form a table of contents, and create an index.

Throughout this guide we refer to MUSIC/SCRIPT whenever we use the term SCRIPT unless otherwise mentioned.)

SCRIPT and its associated programs allow secretarial, administrative and programming personnel to take advantage of the computer's resources to prepare, store and produce final documents ready for mailing or publication. Corrections and revisions are only made to the areas needing modification, thus saving the time needed to retype unchanged areas. (The modifications to the text are made with the MUSIC Editor.)

Establishing the Computer Connection

Naturally, before you can enter anything into the computer you must be connected to it first. Connection between the computer and the workstation is made through a regular telephone line or through a direct cable. Exactly how you establish the connection may be different from one user to another. The *MUSIC/SP User's Reference Guide* describes the various ways in detail.

Signing On

Once a connection has been made between the computer and your workstation, you sign on with a userid (identification code) and password. The userid is the means by which the computer identifies you, the user. Each userid can be from 1 to 16 characters long and can optionally have a subcode of up to 8 characters. Subcodes allow users with the same userids to share a common library of files. Typically, userids do not have subcodes and each person has their own private files. MUSIC does not restrict the number of users that

can be actively signed on with the same userid (unless the userid is restricted). The password is your authorization to use that userid. It is a good idea to change your password frequently to maintain security. Passwords may be composed of any characters, and may be from 1 to 8 characters long.

Figure 1.2 illustrates signing on to MUSIC. The first half of the figure shows the sign-on screen. The second half of the figure shows messages that MUSIC displays after you have signed on.

```
  *MUSIC/SP -- SIGN ON

MUSIC Userid:  _              <=== The cursor is on this line
                                   for you to type in your userid.
    Password:                      Use the TAB or NEW LINE key to
                                   skip to the password field, type
                                   your password, press ENTER.
  F1: Help       F3: Sign off
```

```
  *Userid last signed on 17:10 1994/04/29
  *Sign-on 1994/04/29 Time=09:10, Port=08E, TCB=104
  *Funds Remaining as of Last Accounting..$168.17
  *Go
```

Figure 1.2 - Signing on to MUSIC

Notes:

1. The computer (MUSIC) does not respond to text typed in until the ENTER key is pressed. Make sure you know what key or keys to press on your particular type of workstation to enter information to the computer.

2. The amount of money left under a MUSIC userid is displayed each time the user signs on if the funds for that userid are limited.

3. To have the TODO menu automatically display when you sign on, use the PROFILE program. Refer to *Chapter 7 - Utilities* under the section "Profile".

4. *Go
   This message appears for users who do not have TODO automatically invoked. *Go mode, or MUSIC Command mode, is the time to type MUSIC commands. The MUSIC command to invoke TODO is:

   ```
   exec todo <option>
   or
   todo <option>                  <-- EXEC is the default command
                                      on MUSIC and can be excluded.
   ```

   If only "todo" is entered, then the user is presented with the TODO menu. If "todo 1" is typed, then the user goes directly to item 1 of the menu. See the next section on the TODO menu for a list of items.

Workstation Messages

The following messages appear at the bottom of the screen when the workstation is signed on to the MUSIC system:

Reading	MUSIC is waiting for instructions.
Writing	MUSIC is currently writing information on the screen. (Since new screens of information appear quickly the user does not usually see this message.)
Working	The computer is processing your instructions. Wait until your request is finished (the message `More...` or `Reading` appears).
INPUT INHIBITED	When this message appears on the workstation then you cannot enter information. On some workstations, this message is indicated by a large X at the bottom of your screen. There are three reasons why this message can appear: 1. The system is working on your instructions. 2. The cursor is out of bounds. Press RESET and move the cursor back in bounds with an arrow key. 3. The system has gone *down* or is *hung up* due to hardware or software problems.
More...	The computer is waiting for you to press ENTER to proceed to the next screen.
Attn	When this message appears, you are in Break mode. The PA1 key brings you to this mode, then you can enter a blank line to continue or type /CANCEL to terminate whatever activity is in progress.

Important Keys on your Workstation

The figure below shows the keys at your workstation that are important in understanding how to use the workstation efficiently. Please read this section carefully before signing on to the computer for the first time.

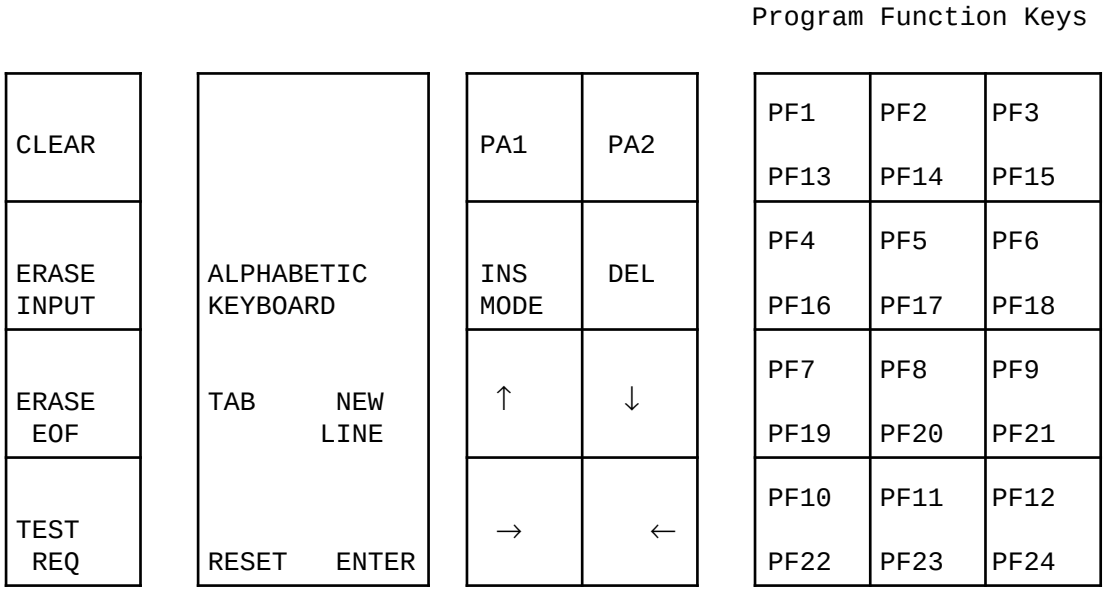


Figure 1.3 - Important Keys on your Workstation

Most workstations come with the keys shown above or equivalents. These display workstations are referred to as IBM 3270-type workstations.

Local Editing Keys

A 3270-type workstation stores in its own memory (not the computer's) the lines displayed on the screen. Changes can be made to the lines you have typed without interacting with the computer by using local editing keys. These keys and their functions are described in the following table.

Key	Function
DEL	Deletes one character at the current cursor position.
ERASE EOF	Erases remainder of line from the current cursor position.
INS MODE	Allows for insertion of characters at the current cursor position. Note that the RESET key terminates INS MODE and returns to replace mode. (You can then type over characters again.)
RESET	Cancels INS MODE and/or the INPUT INHIBITED condition. This key is on the bottom left-hand corner of the alphabetic keyboard.

Characters can be replaced by first positioning the cursor at their location and then typing over them (replace mode). The cursor position can be changed by using the various arrow keys on the keyboard.

The cursor indicates your current position on the screen. The cursor moves as you type. The "-->|" key is the TAB key. The NEW LINE key is an arrow which points down and to the left. These two keys are important for skipping from one field on the screen to another.

Action Keys

The keys which cause an interaction with the system are referred to as "action keys". They are: ENTER, the program function (PF) keys PF1 to PF24, CLEAR, PA1, PA2, and TEST REQ.

The ENTER key is the most common action key since the computer does not know that you have typed in information until you press an action key (most often the ENTER key). Also the ENTER key is used to flip to the next screen whenever the message `More...` appears in the bottom right corner.

Note: You can make corrections to your text before you press ENTER by using the local editing keys described above.

Depending on the features of the workstation, the number of function keys may be 0, 10, 12, or 24.

Getting Help

Online help provides interactive assistance on a wide variety of topics. Press F1 or type "HELP" to access the help facility.

Signing Off

I suppose all good things must come to an end. When you are ready to part company with the computer, type OFF in the command area.

MUSIC Commands

MUSIC commands are instructions to the MUSIC system and are identified with a slash (/) in front when issued from the TODO menu. For example, typing "/COPY file1 file2" asks MUSIC to make a copy of your file called *file1*. MUSIC commands can be typed in either upper or lower case letters (small or capital letters). MUSIC understands these commands when they are typed in the SELECT OPTION area (command area) of a menu screen.

Many commands are available on MUSIC. The most common commands are described below. Refer to the *MUSIC/SP User's Reference Guide* for a complete list of all MUSIC commands with full descriptions.

Most of these commands (except /CANCEL, /SKIP, and /TIME) are to be typed in when the system is in Command mode. You are in Command mode when the TODO menu or the Utility menu is displayed. Type MUSIC commands in the SELECT OPTION area (command area) and press ENTER. The following rules apply to MUSIC command descriptions:

Notes:

1. Below each command, where applicable, is the minimum abbreviation allowed. In some cases there may also be a variation of the form of the command.

2. Lower case portions of commands are variable items or character strings that you supply.

3. Upper case portions of commands are actual keywords and these must be entered as shown.

4. Commands need not be entered in upper case because the system translates commands to upper case when necessary.

5. Alternate options are found within brackets directly below initial choices.

6. All parameters within square brackets are optional and may be omitted. DO NOT TYPE [ ] as part of the command.

7. It is recommended that you enter MUSIC commands preceded by slash (/). This avoids confusion between menu items or TODO commands that can also be entered in the SELECT OPTION area. (In *Go mode the / is not necessary.)

```
/CANCEL
/CA
```

Terminates activity in progress. The /CANCEL command forces a return to the TODO menu or Utility menu. This command can be used in all modes of MUSIC, except Command mode.

This command is useful when you are in the middle of displaying a document and you wish to terminate and return to the menu. First, you must go to Break mode by pressing either PA1 or equivalent break key. Once you have done this, you have the system's "attention", now enter the /CANCEL command.

```
/CI
```

Invokes the Course Information facility for students.

```
/CM
```

Invokes the Course Management facility for teachers. For more information refer to the *MUSIC/SP Teachers Guide*.

```
/CONF confname
```

Invokes the Conferencing facility. For more information refer to the *MUSIC/SP Mail and Conferencing Guide*, or type "HELP CONF".

```
/COPY oldfile newfile
```

Creates a copy of the file *oldfile* into a new file as specified by *newfile*.

```
/DECRYPT infile [outfile]
```

Invokes the DECRYPT program for restoring files that have been encrypted. See ENCRYPT.

```
/DISPLAY name [,x     ] [,y     ]
/D            [,LAST  ] [,LAST  ]
              [,LAST-n] [,LAST-n]
```

Prints the file specified by the *name* starting from line *x* to line *y* and include line numbers specified in the

parameters. LAST means the last line of file. If *x* and *y* are omitted, the Editor displays the entire file.

```
/ENCRYPT infile [outfile]
```

Invokes the ENCRYPT program for coding files to provide extra security. This program randomly exchanges characters resulting in an unintelligible document. You are prompted for a password which will be needed later to restore your file. See DECRYPT.

```
/FINDTEXT 'text'[FILE(spec)][FROMLINE(n)][TOLINE(n)][FROMCOLUMN(n)]
/FT             [TOCOLUMN(n)][FIRST(YES|NO)][FINDS(YES|NO)][CASE(I|R)]
                [OUTPUT(filename)]
```

FINDTEXT is used to search through some or all of the files in your library for a text string. It produces a list of text lines, with line numbers, when the text is found. The file name is also reported.

FINDTEXT supports both full screen and line mode usage. If you want to enter all parameters on screen fields, enter FINDTEXT or FT without parameters. Otherwise a screen is only provide, to assist you in correcting parameters that are in error.

Parameters:

'text' text is a character string that is to be searched for in the list of files defined by FILE. If invoked from *Go, the quotes are required when any of the options below are also specified.

File(spec) *spec* can be one of:

a) a library pattern such as "*.s", "*", or "*work.*.?" etc. All file names in your library that match the specified pattern will be searched for "text".

b) a file that contains a list of files to be used in the search of "text", specified as "<file-name" where filename is an existing file. If this parameter is not specified then all your files are checked.

FROMLine(n) *n* is an integer greater than 0, that specifies the starting line within each file that the search is to begin at. The default is 1. (Abbreviations: FROML and FL.)

TOLine(n) *n* is an integer greater than 0, that specifies the last line within each file that the search is to stop at. Keywords "all", "max", and "end" are used to indicate the entire file. The default is end. (Abbreviations: TOL and TL.)

FROMColumn(n)
n is an integer greater than 0, that specifies the starting column within each line of the file that the search is to begin at. The default is 1. (Abbreviations: FROMC and FC.)

TOColumn(n) *n* is an integer greater than 0, that specifies the last column within each line of the file that the search is to stop at. Keywords "all", "max", and "end" are used to indicate the entire file. The default is end. (Abbreviations: TOC and TC.)

FIRst(YES|NO) When set to "yes", causes the search to stop at the very first match. The default is no.

FINds(n) specifies the number of times to search within each file. After *n* matches in a file, searching is halted in that file. The default is 1.

Case(I|R) When set to I (ignore) the matching is done as if all characters in "text" and the file were in the exact same case. So that "Case" will match with the string "case". When R (respect) is used "Case" will not match "case". The default is ignore.

Output(filename)
This option defines where the output of the search will be placed. You can enter here any file name. The default is "*terminal" to display output at your workstation.

Examples:

1. In this example the string "call tstime(" will be searched for in the library files that end in ".s" .

```
FT 'call tstime(' f(*.s)
```

2. This example searches for the string 'montreal' in the files that end in ".doc" and stores the output of the search in file LIST. Since we want to find only lower case "montreal", we will set case to respect.

```
ft 'montreal' f(*.doc) c(r) o(list)
```

3. In this example we will locate and display only those files where "/inc gork" occurs on line 5 of the file.

```
ft '/inc gork' froml(5) tol(5)
```

```
/FSI [item]
```

This command is used to invoke the FSI (Full Screen Interface) subsystem. This facility allows you access to various components of MUSIC system through a series of selection menus.

```
/HELP [topicname n]
```

This command invokes the MUSIC HELP facility for accessing information about a wide variety of topics. *topicname* is the name of the item on which you want information and *n* is the item selection code (usually a number) from a help menu. Most of the *MUSIC/SP User's Reference Guide* can be found online through this facility. For example, you can enquire about how to use a particular MUSIC command or a utility program. If the information about a particular item is not available, the item will be recorded in a system log file which will be reviewed by the MUSIC administrator.

Workstations with full-screen display enables you to easily browse the HELP facility through menus and text screens. You are able to page forward and backward. You can place your cursor on any highlighted topic name to jump from one topic to another.

If just HELP is entered without a topic, general information about MUSIC is given and a list of general topics is displayed.

Note: The HELP command used from *Go mode accesses MUSIC's general help facility. Other help facilities are provided with a variety of programs on MUSIC. For example, when you are using TODO, the command "HELP" (or F1) gives you the TODO help facility. If you want MUSIC's general help facility while you are within TODO, use the command "/HELP". The slash is necessary to distinguish MUSIC commands from TODO commands.

```
/ID userid;trmcls
```

Signs you on to the MUSIC system.

userid is your identification authorizing you to use MUSIC.

trmcls informs the system of terminal class. The following is a list of workstations currently supported.

trmcls	Manufacturer and Model
3101	IBM 3101 terminal or PC running 3101 emulator
3270A	IBM 3270 terminal with APL character set
3270B	IBM 3278 or 3279 terminal with APL character set
PCWS	PC running MUSIC's PC Workstation Software
IBMPC	PC running Async Communications Software
ASCII	Any ASCII printer or ASCII video display terminal

```
/IDP
```

Invokes the Information Display Program for creating help facilities and bulletin boards. Help is available once the program is invoked.

```
/LANGUAGE [language]
/LANG     [?       ]
```

The LANG command allows you to display or change the default language setting for messages, etc. Not all applications support all languages. If an application does not support the language you request, it uses English. National language names are: English, French, Kanji (Japanese), Portuguese, Spanish. Enter

"LANG ?" to get a list of the languages supported at your site.

```
/LIBRARY srchspc [FULL][TAG][VSAM][PACK][COM][NOSORT][SAVE(filenm)][X]
/LIB             [F   ][T  ][V   ][P   ][C  ]        [S(filenm)   ]

                 [FNAME][SPACE(n)][APPEND]
```

This command is used to obtain a list of file names saved in the Save Library under your userid. The list is produced in alphabetical order. This command is acceptable from batch by using its full form of LIBRARY with or without any parameters after it. The abbreviation of each parameter is shown under its full form.

The LIBRARY command can be entered without specifying *srchspc* or parameters. If you wish to add parameters to this command then you must specify *srchspc.*

Parameters:

srchspc — (search specification) specifies which file names, belonging to the user, are to be searched for in the Save Library index. It may be an actual file name, in which case only that file is listed. Or, the string may contain one or more *wild* characters ? and *, in which case all file names matching the pattern are listed. A ? matches any single character in the corresponding position of a file name. A * matches any group of 0 or more characters. If the srchspc parameter is not specified, all the file names belonging to the user are listed.

To list all your file names you can enter LIBRARY or if you wish to specify parameters then enter "LIBRARY * parameters". The * is your search specification indicating all files on your userid.

COM — lists only files in the common index.

FULL — indicates that for each file listed, its corresponding attributes are also given. See the discussion below on file attributes about the information provided.

TAG — is the same as specifying FULL except that it also displays the tag information for each file. TAG implies FULL.

VSAM — lists only VSAM (Virtual Storage Access Method) files. VSAM implies FULL.

PACK — Normally only one file name is displayed per line for the LIBRARY command. Specifying PACK indicates that several file names may be combined on one line. PACK cannot be specified if a FULL, TAG or VSAM parameter is used.

NOSORT — causes the file names to be put out in unsorted order. This causes output to appear immediately. When the library listing is sorted then there is a short delay.

SAVE(filenm) — indicates that the output of the LIBRARY command is to be saved in a file instead of displaying on the workstation (unit 6). If *(filenm)* is specified with the SAVE parameter, the output is written to the file called *filenm.* If *filenm* is omitted, the name @LIB is used. The original contents in the file is overwritten if the file already exists.

X — This option is similar to FULL, but the output is in a slightly different format and includes time of last open for write, userid of last writer, and number or records. An asterisk (*)

appears after the file size if the file has releasable unused space.

SPACE(n) specifies the initial space (in K) to be allocated for the new file specified by the SAVE parameter. The default is SPACE(32), meaning 32K.

FNAME Causes full file names (including userid and directory path) to be displayed.

APPEND specifies that the output should be written to the end of the file given by the SAVE parameter, after any existing data.

File Attributes

When FULL is specified on the LIBRARY command, extra information about each listed file is also displayed (See examples below). This information consists of:

FILENAME indicates the name of the file.

RSIZ indicates the logical record length of the file.

RFM indicates the record format of the file. The possible record formats are F (fixed length), FC (fixed compressed), V (variable length), VC (variable compressed), and U (undefined).

SIZE indicates the size of the file in number of K (1024) bytes incremented in 2K bytes. The smallest size for a file 2K.

USED indicates the percentage of the file space that is used.

EXT indicates the number of extents of disk space that are used by the file.

REF indicates the date the file was last opened for reading only. 0000000 means that the file has not been referenced since it was last written on.

WRITE indicates the date the file was last opened for writing.

T indicates the type of file. The file is in the common library (public) when the letter C appears in this column. If a V appears, the file is a VSAM file.

OWN indicates the access control of the file for the owner. R means read access is allowed. W means write access is allowed. X means only read access for *execute-only* is allowed. A means only write access for *append* is allowed.

OTHER indicates the access control of the file for non-owners. The meaning of various letters is the same as listed above.

```
/LIST [name] [,x     ] [,y     ]
/L           [,LAST  ] [,LAST  ]
             [,LAST-n] [,LAST-n]
```

Has same effect as the /DISPLAY command, except line numbers are not displayed.

```
/MAN
```

Invokes the word search facility for displaying MUSIC manuals online.

```
/NEWS
/N
```

Lists current news items of interest to MUSIC users.

```
/OFF  HOLD
```

Signs you off the MUSIC system. If the HOLD parameter is specified, a /ID command may be entered afterwards, without having to connect to MUSIC first.

```
/PRINT filename [R(location)] [CC]
```

Lists files at remote printers. The specified file prints in the same format as entered and does not execute. See "Printing Files" in *Chapter 6 - TODO Menu Items* under the SUBMIT section.

```
/PURGE name
/PUR
```

Removes a file from your library.

```
/RENAME oldname newname
/REN
```

Renames file from *oldname* to *newname*.

```
/SKIP [n  ]
/SK   [ALL]
```

Skips *n* (n=number) lines or ALL output lines, default setting is 1. See the /CANCEL command description above for information on Break mode.

```
/SUBMIT
/SUB
```

Submits a program for execution and printing at a remote printer. For details about this facility refer to "SUBMIT for Other Programs" in the SUBMIT section of *Chapter 6. TODO Menu Items*. SCRIPT users should use the SUBMIT SCRIPT item on the TODO menu to perform this operation.

```
/TIME
/T
```

Determines the time of day (during Break mode) and the amount of processing units used for the current job, running up to the time the command is issued. See the /CANCEL command description above for information on Break mode.

```
TODO [n]
```

This command invokes the Time, Office, and Documentation Organizer (TODO) facility. This facility allows you to access various components of the MUSIC system through a selection menu. This menu consolidates the most frequently used programs for an office environment. For example, access to SCRIPT (word processing program), MAIL (electronic mail), SPELL (spell checking), etc. Help is provided once the facility is invoked.

```
WEB [url address]
```

This command invokes the Web line-mode browser for displaying World-Wide Web documents on the Internet.

Chapter 2. Menu Facilities

Menu Facilities

Easy access to MUSIC programs can be done through menu (or panel) facilities. This menu approach reduces the need to remember commands and helps novice users to be productive immediately.

MUSIC/SP offers four menu facilities for users and one menu facility for the system administrator. The system administrator facility provides step-by-step menus for installing, monitoring, maintaining, and tailoring the system. It is documented in the *MUSIC/SP Administrator's Guide*.

Each of the four user facilities combine several functions of the MUSIC/SP system. These menu facilities fall under the categories of general users, student computing, and office applications. They are as follows:

- FSI (Full-Screen Interface) - General Users
- CM (Course Manager) - Teachers
- CI (Course Information) - Students
- PROG (Programmers Menu) - Students
- TODO (Time, Office, and Documentation Organizer) - Office Applications

MUSIC/SP provides the programming tools to support the creation and modification of menus. You can easily create your own environment on MUSIC/SP or create an environment tailored for a particular group of users. You can choose to have one of these facilities start automatically each time you sign on to MUSIC/SP. For information about creating and changing menus, refer to the topic "TMENU" in *Chapter 7 - Utilities*.

Notes:

1. Your installation may provide additional menu facilities other than the ones described here.

2. An optional word processing facility called IBM DisplayWrite/370 may be available at your installation. Information about using this program is given later in this chapter.

Full Screen Interface (FSI)

The Full Screen Interface (FSI) allows you to access various components of the MUSIC system through a series of selection menus. Enter the command FSI from command mode (*Go) to start the interface. You can specify FSI as the auto-program in your user profile if you wish to have it automatically started when you sign on to MUSIC.

Throughout the interface the following standard function key definitions are used.

F1	Provide detailed information on the function and usage of the screen currently being viewed.
F3	Return to the previous screen without performing any operation. This can be used to exit from the interface if pressed in the main selection menu.
ENTER	Perform the operations indicated on the screen.

Main Selection Screen

```
Help   End    Up    Down   Top   Bottom Main   Scan   Find   Topic  Quit
-----------------Full Screen Interface for MUSIC--------------- Page 1/1
 Command ===>
  * NO NEW MAIL *
  Place the cursor on an item and press ENTER or RETURN.

  MUSIC tools:
     Mail          Electronic mail facility
     Programming   Compilers, processors, tutorials, etc
     CI            Course Information
     Internet      Internet access, news reader, gopher, etc
     More          Other general MUSIC tools
  MUSIC files:
     FLIB *        Full Library Screen for all files
     FLIB          Filespec=> x.*                 < pattern
     FUTIL         Other file related utilities
  MUSIC environment:
     Help          General help and online documentation
     New Password  Change your password
     Defaults      FSI customization
     Profile       Profile utility and options
     Disconnect    Terminate your session and disconnect from MUSIC
     \Suggest      Make a suggestion or send a comment to support staff

 F1=Help       F2=Suggest       F3=End       F9=Find       F12=Retrieve
```

Figure 2.1 - Main Selection Screen of FSI

Menu for Student Computing (PROG)

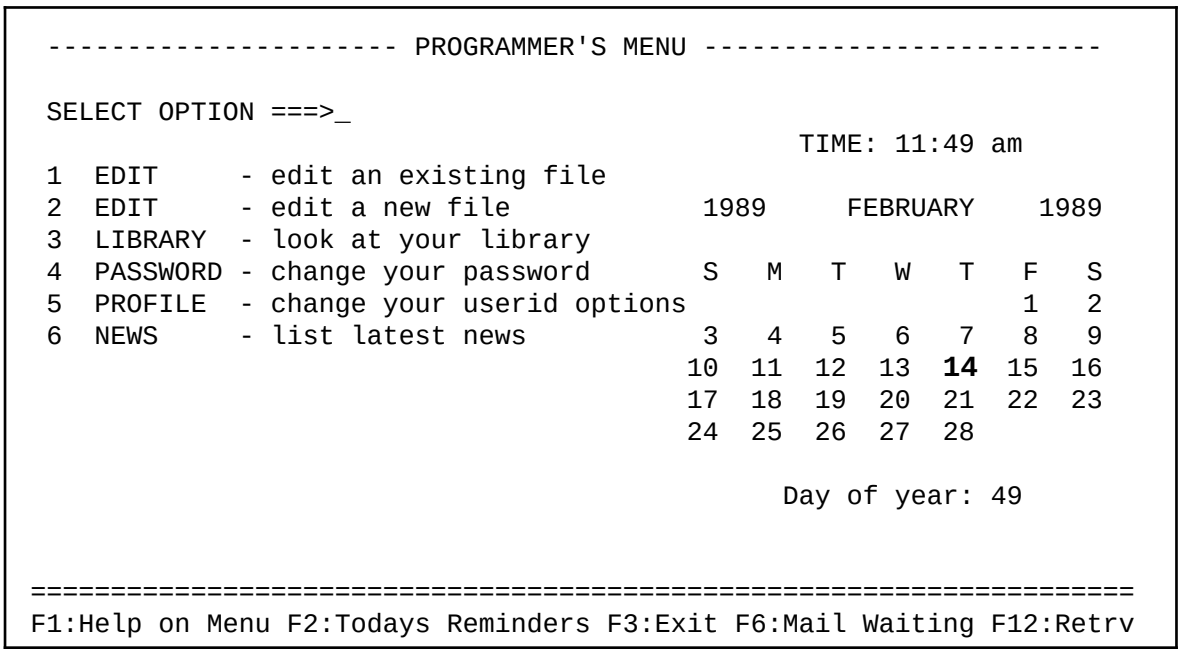


Figure 2.2 - PROG Menu Display

The above figure illustrates the menu facility called PROG. This is an example of how the working environment can be tailored for certain users. In this case, this is a typical configuration for students. This menu can be modified to include other programs as required. For information about creating and changing menus, refer to the topic "TMENU" in *Chapter 7 - Utilities*.

Menu Facility for Teachers (CM)

This system allows teachers to communicate with their class via the computer. The teacher uses the CM command (Course Manager) to manage the course material and the students use the CI command (Course Information) to access the information you have prepared.

For more information refer to the *MUSIC/SP Teacher's Guide*.

Time, Office, and Documentation Organizer (TODO)

To use the TODO facility, the first step is to make the connection from your workstation to the computer by *signing on* to the MUSIC system, with a MUSIC userid (ID) and password. MUSIC waits for instructions when the TODO menu is displayed. You can select an item from the menu or type in a MUSIC command. Some users may not go directly to the TODO menu but see the message `*Go` displayed on the screen. Type "TODO" to get to the menu. At this time, you need to select an item from the menu.

The following diagram illustrates the screen display for the TODO facility. The cursor is positioned in the SELECT OPTION area where you can type in your selection from the menu.

```
 ---------- TIME, OFFICE, AND DOCUMENTATION ORGANIZER ----------TODO

 SELECT OPTION ===> _
                                                TIME: 11:49 am
 1  Schedules
 2  Electronic Mail <option>                1989     FEBRUARY    1989
 3  Telephone Log
 4  Calculator <calc>                       S   M   T   W   T   F   S
 5  Spell Check document <option>                                   1   2
 C  Create new <filename>                    3   4   5   6   7   8   9
 R  Revise <filename>                       10  11  12  13  14  15  16
 X  Execute SCRIPT <filename>               17  18  19  20  21  22  23
 S  Submit SCRIPT <filename> <options>      24  25  26  27  28
 L  List File Names <options> <pattern>
 M  Schedule a Meeting <options>                 Day of year: 49
 U  Utilities <option>

================================================================================
F1:Help on Menu  F2:Todays Reminders F3:Exit F6:Mail Waiting F12:Retr
```

Figure 2.3 - TODO Menu Display

Function Keys on TODO menu

F1 Help on Menu - Describes how to use the menu.

F2 Todays Reminders - Show reminder(s) for today. Refer to the "Remind" section in *Chapter 7. Utilities*.

F3 End - Stop display of menu, terminate.

F6 Mail Waiting - Post message if mail is waiting or not waiting.

Selecting an Item

Consult the list of items (options) on the menu. Type the item code of your choice in the SELECT OPTION area. Along with the code you can type additional information for the selection, as indicated within angular brackets (<>) on the menu.

Example selecting "Revise" (code R) with the filename "myfile":

```
SELECT OPTION ===> R myfile
```

Notes:

1. There are three types of commands that you can enter in the SELECT OPTION area (command area):

   a. TODO selection items
   b. TODO commands
   c. MUSIC commands

   If the command is preceded by a slash (/), then the command is executed as a MUSIC command. If the command is entered without a slash, then TODO still processes the command as a MUSIC command, provided it is not a selection option or a TODO command. (Some installations may have restrictions on this feature.) For general help on the MUSIC system type "/HELP" in this area. (The F1 key and the TODO command HELP give you help on the TODO menu only.)

   Example: Type the MUSIC command to receive help with MUSIC.

   ```
   SELECT OPTION ===> /help
   ```

2. Online help is available with each menu item after it is selected.

3. The REMIND program is available with TODO although it is not a menu item. Use the F2 or the REMIND command to get or set reminders respectively. Refer to the "Remind" section in *Chapter 7. Utilities*.

TODO Commands

The following commands are available and some of these commands are in the form of function keys and have been explained previously.

CANCEL	GETREM (F2)	OFF
CLEAR	HELP (F1)	PFnn def
DELIM char	KEYS	REMIND
END	MAIL (F6)	*-n
EXEC	MENU	=x

CANCEL — Stop display of menu, terminate menu.

CLEAR — Clear the menu input areas.

DELIM char — Change the multiple command delimiter to *char*, where *char* is a character of length 1, and not one of the characters a to z, 0 to 9, *, =, /.

EXEC — Pass the SELECT OPTION area to MUSIC to be executed as a command

KEYS	Display a screen which allows you to change the function (PF) key definitions. These definitions are saved in a file, TMENU.KEYS, so that they can always be used.
MENU	Display the menu.
OFF	Exit from this program and sign off.
PFnn	Show the definition of PFnn, where *nn* is a number from 1 to 24.
PFnn def	Set the definition of PFnn to *def*, where *nn* is a number from 1 to 24. The definition can be from 1 to 50 characters long.
REMIND	Enter the reminder facility.
*	Display the last command entered in the SELECT OPTION area.
*-n	Display the previous nth command entered in the SELECT OPTION area, where *n* is a number from 0 to 4.
=x	Go to the first menu and process *x*.

Menu Items

The following is a brief description of the TODO menu items. The items in this menu select some of the more frequently used programs on MUSIC for an office environment. As well as menu items, TODO commands and MUSIC commands can be entered in the SELECT OPTION area. Your installation may have some restrictions on the commands allowed.

1. Schedule

This program is used for scheduling your time or the time of others (if authorized). Enter your agenda for each day then ask to see your schedule for a particular day or view a whole month at a time. Also you may view (and update if authorized) the schedule for a conference room or other items (i.e. equipment). Refer to "SCHEDULE" in *Chapter 6 - TODO Menu Items* for more information.

2. Electronic Mail <option>

This item invokes the MAIL program (Electronic Mail). Through this program you can receive or send mail to other MUSIC users and installations. Acknowledgements of your outgoing mail is listed. Formatting for the mail text is included with the program. See *Chapter 4 - Electronic Mail*.

3. Telephone Log

Keep track of your phone calls with this menu item. Current date and time is automatic when the program is invoked. Each month a new log file is created. For more information refer to the Telephone Log section of *Chapter 6 - TODO Menu Items*.

4. Calculator <calc>

This item gives you the POLYSOLVE program for calculations. See the section "Calculator" in *Chapter 6 - TODO Menu Items* for information.

5. Spell Check document <option>

This program is a full-screen Spelling Check facility. Misspelled words in your document are flagged and alternative spellings are offered. Many features are included with this program and are documented under "Spelling Check" in *Chapter 6 - TODO Menu ITEMS*.

C. Create new <filename>

This item of the menu invokes the Editor program and puts you in Input mode into an empty file for typing SCRIPT documents. An automatic Letter and Memo facility is also included with this item. Refer to the section "Creating New SCRIPT Files" in *Chapter 3 - MUSIC Editor* for entering files. Also refer to *Chapter 5 - MUSIC/SCRIPT* of this guide for formatting instructions.

R. Revise <filename>

Choosing this item brings you to Edit mode of the Editor. A copy of your *filename* is presented on the screen for you to revise (edit). Refer to *Chapter 3 - MUSIC Editor* for more information.

X. Execute SCRIPT <filename>

This item executes (runs) a SCRIPT document. To see the formatted version of your documents you need to "execute" them. SCRIPT Output Options can be entered at this time. Further details are found under the section "Executing SCRIPT Documents" in *Chapter 6 - TODO Menu Items*.

S. Submit SCRIPT <filename> <options>

Use this menu item for submitting your documents to print at remote printers. See the section "Submitting SCRIPT documents" in *Chapter 6 - TODO Menu Items* for details.

L. List File Names <options> <pattern>

Choose this item for a list of all your document names in alphabetical order. Also more information about your files is available: the size of the file, the date the file was last revised, the date the file was last referenced, and one line descriptions. Refer to the section "Library - List File Names" in *Chapter 6 - TODO Menu Items*.

M. Schedule a Meeting <options>

This item helps you to organize a meeting by: checking the schedules of attendees, conference rooms, and equipment items for free time; updating these schedules, and sending a memo to each participant. Refer to "Schedule a Meeting" in *Chapter 6 - TODO Menu Items* for information.

U. Utilities <option>

This item gives you another menu with the following items:

NAMES
PROFILE
CHANGE PASSWORD
TABLE OF CONTENTS
INDEX

Refer to *Chapter 7 - Utilities* for information.

Changing the Menu

The TODO menu can be changed by the user. New Items can be added or existing items deleted or changed. These items could call other MUSIC programs, your own programs, or other menus. Refer to the section "TMENU" in *Chapter 7 - Utilities*.

Using TODO for Word Processing

The following choices are available to perform word processing using the TODO facility.

- Create a new document file (item C)
- Revise an existing document file (item R)
- Execute the document using SCRIPT (item X)
- Submit a document for printing (item S)
- List the names of stored files (item L)

When creating a new document, you are brought to Input mode of the Editor. At this time, you can start typing in the text lines and SCRIPT control words. When you finish, you can then inform the Editor to file the new document in your library for future use.

When revising a document you are in Edit mode of the Editor. When in this mode, Editor commands (not MUSIC commands) are used to correct documents.

After a document has been created and/or revised, the Editor command "FILE" can be issued and you are brought back to the TODO menu.

To view the formatted version of a document, you need to execute it. The SCRIPT program comes into the picture at this point to interpret formatting instructions imbedded in your file. Even though the stored document (file) consists of control words and text, only the text, now formatted, is printed. Refer to "Executing a SCRIPT document" section in *Chapter 6. TODO Menu Items* for details.

DisplayWrite/370

IBM DisplayWrite/370 is a text editor that you can use to create and edit memos, letters, reports or any kind of office document. This program is an optional product of IBM and may not be available at your site.

DisplayWrite/370 shows you documents as they look when they print. While you are editing the document, you can usually see the effects of the changes you make to your document, even if they are format changes.

You can check spelling, find synonyms, get information about grammar, work with footnotes, compose your own dictionary, choose dictionaries, and work in other languages. You can also "undo" or "redo" what you have done.

To invoke DisplayWrite, issue the command:

```
DW370 filename options
```

filename is the name of the file to be edited or created and *options* are DW/370 options.

Figure 2.4 shows Displaywrite/370's basic screen presentation. This screen was displayed in *Go mode by issuing the following command:

```
dw370 dwsample
```

```
ABCD000.DWSAMPLE                                                    Page 1
                                                                    Line 0
===>
<---+_---1_---+_---2_---+_---3_--V+_---4_---+_---5_---+_---6_---+_>--7_---+----
------------------------------------------------------------------- Page 1 ----
To:  All Staff

From:  Planning Office

Subject:  Quarterly Business Meeting

Date:  September 23, 1985

This year the Corporate Division has scheduled the Autumn Quarterly
Meeting for Friday, October 23, in Wolverhampton, U.K.  So that we
might all prepare for this important meeting, the Planning Office
has compiled the following "mini-report" on one area considered
vital to the company's growth:  international development .

We hope the report will help prepare you for some of the major
discussions scheduled to take place at the Quarterly Meeting.  You
PF 1=HELP       2=Insert     3=END        4=Instr.     5=RFind          6=Aid
```

Figure 2.4 - DisplayWrite/370 Screen Display

Usage Notes

1. The RECOVER and CLEANUP options are NOT supported under Music.

2. Only V or VB files (that have been **created** by Displaywrite/370 can be edited. To 'edit' an existing MUSIC file, you will need to create a 'NEW' file and issue the 'GET' command. Most Music files can be 'VIEWED' with Displaywrite/370.

3. If a user wishes to create their own personal CLIST file, they must first get a copy of the master CLIST file. This must be done outside of Displaywrite/370. Issue the command: COPY $DW2:CLIST.DW370 CLIST.DW370

   To edit/change the personal CLIST, issue the command: DW370 CLIST.DW370.

4. Only 1403 type output is currently supported when using the PRINTDOC facility. The file is created for all other type devices, but currently there is no easy method of transferring the file to CMS and the Advance Printing Support Facilities that are needed to actually output the file on a real device.

References

Using Displaywrite/370 Release 2, (SH12-5172).

Displaywrite/370 Reference, (SH12-5176).

Chapter 3. MUSIC Editor

Using the MUSIC Editor

When creating or revising files you need to use the MUSIC Editor program. The Editor is a powerful interactive program that allows you to enter and change documents easily. The diagram below illustrates the order of steps taken when using the Editor.

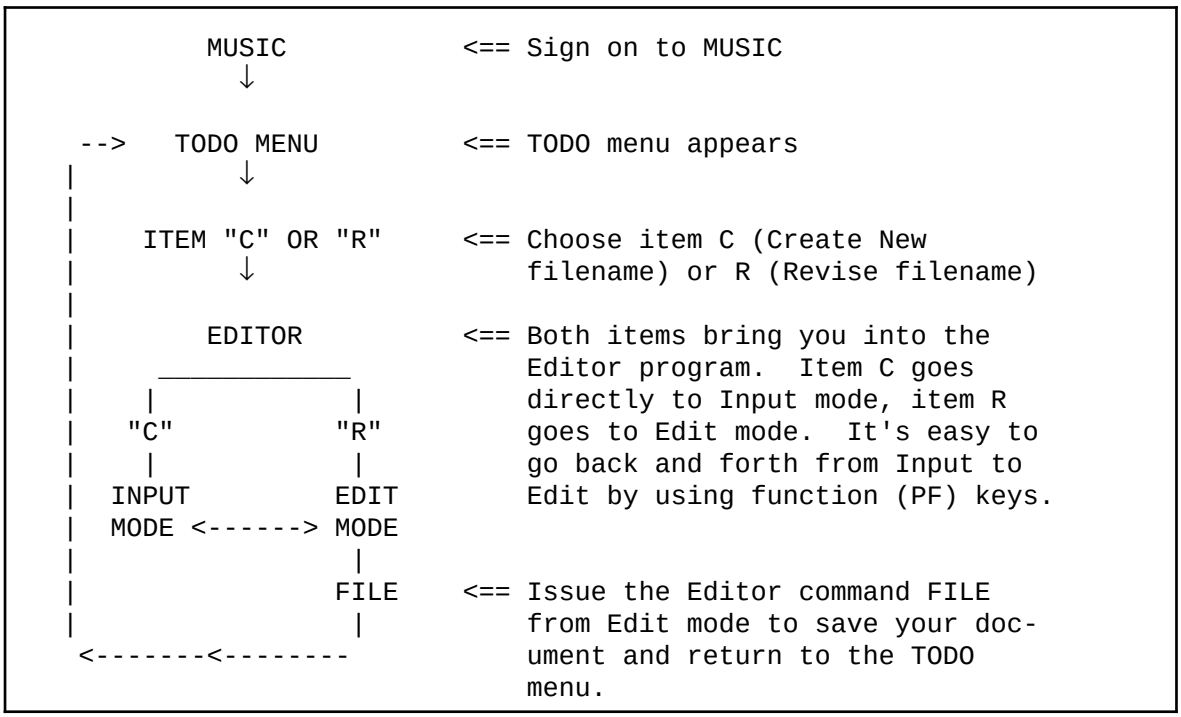


Figure 3.1 - Steps for Using the Editor

To use the Editor, choose item "C" (Create) on the TODO menu. This allows you to enter a new document. Choose item "R" (Revise) when you want to edit (change, add to, etc.) an existing document. Either item brings you into the Editor program. Once inside the Editor you can easily go back and forth between Input mode and Edit mode by using Program Function (PF) keys. If you are in Input mode then any function key will bring you into Edit mode. The F11 key will return you to Input mode. To return to the menu, issue one of the following Editor commands during Edit mode:

FILE Saves document and stores it in the MUSIC Save Library under your userid, and then returns you to the TODO menu.

QUIT Returns to the TODO menu without saving the document.

For more information about Editing, please see the section "Revising Files Using the Editor" later in this chapter.

Creating New SCRIPT Files - TODO Item C

This section outlines how you can enter information that is to be used by MUSIC/SCRIPT into the computer. The MUSIC/SCRIPT program takes the information you enter and formats it according to your instructions.

When you use the Editor to enter text into the MUSIC system, you are creating a file. SCRIPT files consist of text and control words. This file is then saved on a disk (online) or can be archived on a magnetic tape (offline) for future use.

Files on MUSIC are stored in the *Save Library*. A file can hold up to 2 million characters of information. However, your installation may have set a lower limit than this for your userid.

File Names

Each file name can be up to 17 characters in length. File names cannot contain blank characters. Each character of the name can be any letter (A-Z) or any number (0-9) or any of these special characters:

```
~  !  @  #  $  %  &  _  +  .
```

The exception to the above rule is that the first character of the name cannot be any of the following:

```
~ ! % & _ + .
```

and should not be the @ symbol as files starting with @ are reserved for program generated files. Blank spaces cannot be used. The name can be typed using either upper or lower case letters, as lower case letters are converted to upper case automatically by MUSIC for file names. Some examples of valid file names are:

```
ABC       letter2       $        memo.may.90
```

The "Create New" option on the TODO menu also uses predefined document shells or standard setups for writing letters and memos. These file names begin with the characters "**L.**" and "**M.**" You should use these characters at the start of a file name only when you wish to create a letter or a memo. Refer to "Automatic Letter and Memo Files" later in this section for more information.

Starting a New File

When the TODO menu is displayed, instruct MUSIC that you want to create a new file. Choose item "C" (Create) with a file name in the SELECT OPTION area. Figure 3.2 shows what to enter (underlined) in the command area if your new file was called "abc".

```
 ------------ TIME, OFFICE, AND DOCUMENTATION ORGANIZER ---------TODO

 SELECT OPTION ===> c abc
                                                 TIME: 11:49 am
 1  Schedules
 2  Electronic Mail <option>              1989     FEBRUARY    1989
 3  Telephone Log
 4  Calculator <calc>                      S   M   T   W   T   F   S
 5  Spell Check document <option>                                  1   2
 C  Create new <filename>                  3   4   5   6   7   8   9
 R  Revise <filename>                     10  11  12  13  14  15  16
 X  Execute SCRIPT <filename>             17  18  19  20  21  22  23
 S  Submit SCRIPT <filename> <options>    24  25  26  27  28
 L  List File Names <options> <pattern>
 M  Schedule a Meeting <options>               Day of year: 49
 U  Utilities <option>

=============================================================================
F1:Help on Menu F2:Todays Reminders F3:Exit  F6:Mail Waiting F12:Retrv
```

Figure 3.2 - Selecting a Menu Item

What you are doing here is creating an empty file at your workstation. The diagram below shows the screen that the Editor program presents after the create command is issued.

```
ABC                                          L 80   W 1 72     Rec 1/0
  _









  ---+-T---1----+---2----+---3----+---4----+---5----+---6----+----7--
                           * Input Mode *
                                                                Reading
Default PFs: 1:Help  2:Split  3:Quit    4:Mark    5:Center  6:Del line
  *EDIT*     7:Uppg  8:Dwnpg  9:Locate 10:Ins ln 11:Input  12:Command
```

Figure 3.3 - Screen Display for Input Mode

You are now in Input mode of the Editor and you can begin to type in your text. (If you wish to type only capital letters use the Editor Command "TEXT UC".) For a SCRIPT document this file will consist of text and SCRIPT control words for formatting instructions. Refer to *Chapter 5 - MUSIC/SCRIPT* for details on control words. The following messages that appear on the screen are described below.

File -- ABC The name of the file is shown in the upper left hand corner.

L 80	The Logical Record Length is the maximum number of characters allowed per line. The default for MUSIC files is 80 characters.
W 1 72	Columns 1 to 72 are shown on the screen. To change the window to show columns 9 to 80 then use the WINDOW FLIP command. This command and other Editor commands are described in the next section on revising files.
Rec 1/0	The current number of records (lines) is displayed in the upper right hand corner along with the current line number (1). This file is empty so the number of lines is 0.
1----+----2 etc.	This line of numbers and dashes is a scale line which shows the column numbers. (The diagram does not show the exact number of dashes.)
* Input Mode *	Indicates that you are in Input mode of the Editor. In this mode use the NEW LINE (down and to the left arrow) key to end the line. Press ENTER when the screen is filled or nearly filled. If you press an action key, such as a function key, then you leave input mode. Press F11 to return to input mode.
Reading	This message appears in the bottom right hand corner on MUSIC whenever it is waiting for instructions.
Default PFs	The Editor default definitions for function keys are displayed at the bottom of the screen. Descriptions for each function key can be found later in this chapter under the section "Action Keys".
_	This symbol is the cursor. In figure 2.3 above, it is positioned on the first line of your empty file.

You can now begin to type in your document. Use the NEW LINE key to end each line, or its equivalent, depending on the type of workstation you are using. Use the ENTER key when you have filled the screen to transmit your information to the computer. All your text moves up and there is more room for you to keep on typing in Input mode.

You can correct typing mistakes on the line you are currently typing by using the backspace key. Just backspace over the incorrect characters and retype the line from that point on. You are limited to entering 80 printed characters per line. It is a good idea not to make your input lines are not too long in case you might want to add on to them later. It is also a good idea to start each new sentence on a new line. This will make your editing easier later on. Don't worry if your document doesn't look lined up as you are typing it in. SCRIPT will do all the formatting later.

Also, remember to use the "1" (one) and "0" (zero) keys for the numeric one and zero, not the "L" (el) and "O" (oh), the alphabetic letters.

Note: You can use SCRIPT to format a SCRIPT file from the TODO menu by using the X item. This item executes a file coded with formatting instructions. Select the menu item X (Execute SCRIPT <file-name>) along with the name of the file you want to format. You will be prompted to enter SCRIPT output options. Refer to the section on "Executing SCRIPT Documents" in *Chapter 6 - TODO Menu Items* for details.

If you are not using the TODO menu system, you can execute SCRIPT by including the following instruction as the first line in your text file:

```
/include script
```

The second line in your text file must then include either SCRIPT output options or be blank. Refer

to Appendix B for an example.

Saving a File

You will, no doubt, want to FILE (save) your document after you finish entering the text and control words. This means the computer will keep it for you as long as you want. To do this, you must first go to Edit mode of the Editor, by pressing the Program Function (PF) key numbered 12. (function keys are described in the topic "Action Keys" later in this chapter.) The computer responds by moving your cursor to the COMMAND area and the `*Input Mode*` message at the bottom of the screen disappears indicating that the Editor is in Edit mode. This is the mode in which you can make changes to your file and, as in this case, save your file. Figure 3.4 below illustrates the screen format for Edit mode after the file ABC has been typed in and F12 has been pressed.

Complete information about the screen format is given with figure 2.6 under the topic "Screen Format" later in this chapter.

```
userid:ABC                                L 80   W 1 72     Rec 1/7
    *Top of File
         This is a short sample of a SCRIPT document
    without using any control words.
    Each input line should be short (don't go to the end
    of the line.
--> Start new sentences on a new line.
    *End of File




  ---+-T---1----+---2----+---3----+---4----+---5----+---6----+----7--
Command: _
                                                                Reading
Default PFs: 1:Help  2:Split  3:Quit    4:Mark    5:Center  6:Del line
  *EDIT*     7:Uppg  8:Dwnpg  9:Locate 10:Ins ln 11:Input  12:Command
```

Figure 3.4 - Screen Display for Edit Mode

The information displayed on the top of the screen remains the same for Input or Edit mode of the Editor. Other messages on the screen include:

*Top of File	This line appears when you are at the top of your file. The line pointer points to the first line at the beginning of an edit session.
*End of File	This line indicates that you are at the bottom of your file.
Command:	This is the command area of the Editor. You can type Editor commands in this field (not MUSIC commands). To get the cursor to this area use F12 use the NEW LINE key or TAB key. (Function keys are described in the topic "Action Keys" in the section Editing with Full-Screen Mode.

Once the cursor is in the command area, you can type the Editor command "FILE" in the command area to store your document. This document is now stored in the MUSIC Save Library under your userid. The Editor saves the document using the name specified on the top of the screen (Figure 3.4). After you type in the command, the system responds with a message assuring you that the document has been saved under the

name that you have provided.

```
ABC
NEW FILE
SAVED
```

Then you return to the TODO main menu. Sometimes when you issue the FILE command the following message appears:

```
FILE ALREADY EXISTS.  DO YOU WANT TO REPLACE IT?
```

This message is displayed if you already have a document with the same name as the file you are working with. Answer no and re-issue the FILE command with another name. Example:

```
COMMAND:  FILE newname
```

Changing an Existing File

You can easily make changes to an existing file. This is described in the next section on Revising files.

Purging a File

When you return to the TODO main menu, you can select items on the menu or you can type MUSIC commands. Besides choosing the menu items, you can do other common tasks such as purging a file. To delete a file from your save library enter:

```
SELECT OPTION ===> /PURGE filename
```

Refer to the previous section on "MUSIC Commands" in *Chapter 1 - Introduction* for more information about other commands.

Automatic Letter and Memo Files

The CREATE (item "C") function uses the file name to determine the type of file you are creating. This item can prepare a standard setup (shell) for either a letter or a memo. All you have to do is fill in the details.

If your file name begins with "L.", then CREATE automatically gives you the setup for a letter. If your file name begins with "M.", then you receive the setup for a memo. If your file name does not begin with L. or M. then the Editor presents an empty file. See the section "Automatic Letters and Memos" in *Chapter 5 - MUSIC/SCRIPT*, for information about letter and memo setups.

Revising Files Using the Editor - TODO Item R

The MUSIC Editor is used to make modifications to MUSIC files (as well as creating new files). These files may be SCRIPT documents or they may contain other information such as a program. (For details on SCRIPT see Chapter 5.)

A key feature of the Editor is its ability to search for a line based on its contents rather than its line number. Changes can be made to portions of the line without having to retype it completely. The same change can be done to multiple lines with a single command (global changes).

The Editor makes all requested changes to a **temporary** copy of your file while the original copy is left intact. This means that you see the effect of all your changes without disturbing the original copy. Once you are satisfied with the changes you can tell the Editor to apply them to the original by issuing the Editor Command FILE.

Besides changing lines, the Editor can be used to rearrange a file by moving or copying groups of lines. Completely new lines may be inserted at any location and the contents of another file can be merged into the file being edited.

The Editor is fully described in the *MUSIC/SP User's Reference Guide*. A summary of the basic concepts and the more common commands are explained in this manual.

Starting the Editor

To edit an existing file, choose the "R" (Revise) item from the TODO main menu, along with the file name. If you had a file with the name "sample" then you would type the following (which is underlined) and press ENTER.

```
SELECT OPTION ===> r sample
```

The following diagram illustrates what the screen would look like after entering the command above.

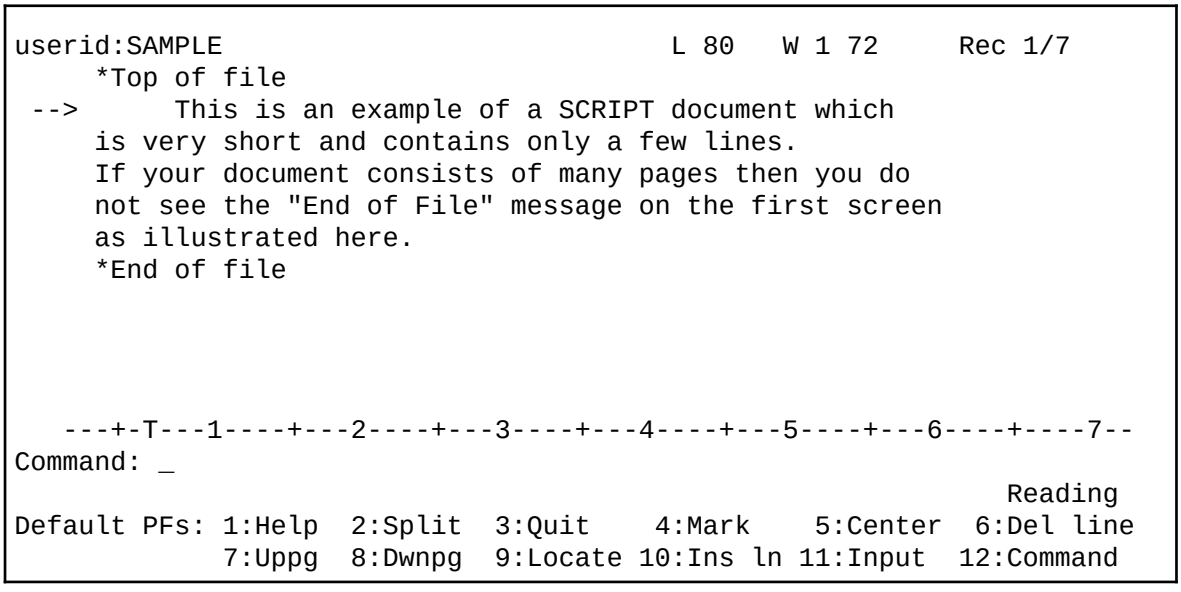


Figure 3.5 - Screen Display for Editing a File Called Sample

When you issue the Revise command the Editor presents a copy of your original file on the screen. You are now in Edit mode of the Editor and you are positioned at the beginning of the file. The line pointer points to the first line of your document and the cursor is in the Command Area. At this point you can make changes to your file by either:

- Moving the cursor to replace, delete, or insert text on a line.
- Typing in an Editor command in the command area.
- Pressing a function key.

The Editor has a current line pointer which points to one line in your file. At the beginning of an edit session the current line is the first line in the file. Several commands can move you to a new place in the file. Editor commands can be typed in the command area or they can be issued by pressing a function key.

Multiple Commands per Line

You can combine several Editor commands on a single line if you use a ";" character between them. You will find this very useful, particularly after you become familiar with the Editor. This *delimiter* character of ";" does, however, mean that you must never use it in any other way in the command area.

You can use a DELIM command to change the delimiter character for that edit session. For example, the command "delim #" changes the delimiter character to "#". Or you can use just the command "DELIM" by itself which informs the Editor that there is no delimiter defined.

Editor Full-Screen Mode

Introduction

Users of 3270-type workstations can take advantage of the special features of these workstations by using the MUSIC Editor in *full-screen* mode of operation.

In full-screen mode, you can modify data directly on the screen, using the local editing keys, as well as enter commands in an area near the bottom of the screen. Also, various editing operations can be done by pressing program function (PF) keys to enter Editor commands.

Screen Format

The screen display in full-screen mode is divided into the following areas:

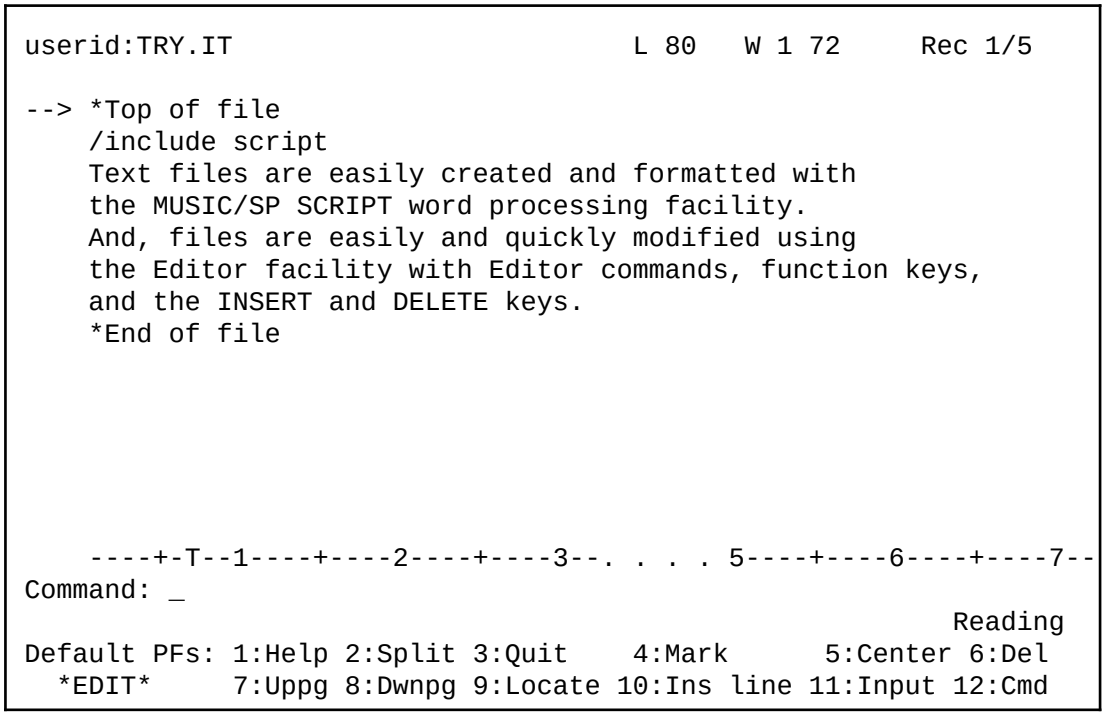


Figure 3.6 - Screen Format for the Editor

1. **Title line:** this is the first line on the screen, and contains the name of the file being edited, the line number of the current line, the file's logical record length (L), the starting and ending window columns (W), the current line, and the total number of lines in the file. This field is not modifiable on the screen, although Editor commands can be used to change the file name and window setting.

2. **Lines of the file:** this is the main body of the screen, immediately following the title line. It may be up to 20 lines on a 24-line screen, or 39 lines on a 43-line screen. Each screen line displays one record of the file, and the text is directly modifiable on the screen. The records displayed include the current line and the lines immediately preceding and following it. Only the window portion of each record (see the WINDOW command) is displayed, to a maximum of 72 characters. The current line is indicated by an

arrow pointer in the left margin and is also displayed in high intensity (red on a color terminal). If line numbering is in effect (the NUMBER command), line numbers are displayed in the left margin. The beginning of the file, if displayed, is indicated by `*Top of file.` Similarly the end of the file is indicated by `*End of file.` In Input mode, several empty lines are displayed following the current line, thus allowing you to add new lines to the file by typing them on the screen.

3. **Tab line:** this shows column numbers and input tab positions. The tab positions (displayed as T's) correspond to the column numbers specified on the TABIN command. The tab positions are relative to the starting window column, rather than to column 1 of the file's records.

4. **Command area:** this is used for entering Editor commands. All of the usual commands are valid here, including the NOFS command if you wish to terminate full-screen mode (but continue editing). Several commands may be entered by separating them by the command delimiter character, normally semicolon (;). If the Editor is in Input mode rather than command (Edit) mode, the command area is replaced by the centered message `* Input Mode *.`

5. **Message area:** this line of the screen is reserved for messages from the Editor. Up to 3 messages can be packed into this area. The messages are displayed in high intensity (white on a color terminal). If the messages do not fit, or if considerable output is generated by a command such as LIST or SCAN, the output is displayed on a new screen in normal MUSIC format, with the status message `More...` in the bottom right corner. Pressing the ENTER key allows output to continue on the next screen, or returns to the Editor full-screen display if no more output remains. Editor commands may not be entered until the `More...` condition has been cleared. While output is being displayed in normal MUSIC format, the PA1 key may be used to go to attention mode (**Attn**) and skip output (by the /SKIP command).

6. **Status indicator:** the current status of the workstation and the Editor is shown in the bottom right-hand corner of the screen. Possible status words are `Reading, Working, **Attn**,` and `More...`

7. **SHOW area:** if a "SHOW filename" command has been used, lines of the specified file are permanently displayed at the very bottom of the screen. This area can be from 0 to 12 lines. The message, command area, and tab lines described above are shifted up on the screen to make room for the SHOW area. For example, the area could be used to display the definitions of program function (PF) keys. Refer to the SHOW command for more information.

--> **Line Pointer:** The line pointer is indicated by an arrow in the left margin. It always points to the current line. Commands entered in the command area will take affect at this point in your file. Some users may have PREFIX ON as the default for their editor. The prefix area is in the left margin and the line pointer is smaller. The screen display is slightly different:

```
 ==== *Top of file
>==== /include script
 ==== Text files are easily...
```

The editor works the same except you are able to use prefix commands in the margin.

Important Keys

The diagram below shows the important keys on a workstation needed for editing files.

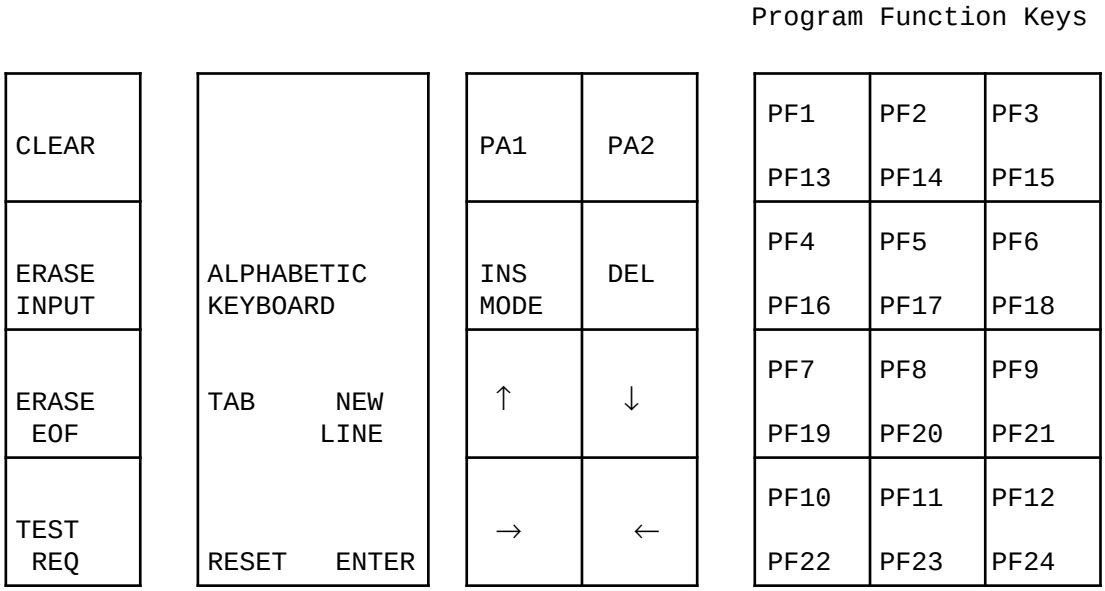


Figure 3.7 - Keys for Editing

Local Editing Keys

A 3270-type workstation stores in its own memory the lines displayed on the screen. Changes can be made to these lines without interacting with the computer by using local editing keys. These keys and their functions are described in the following table.

Key	Function
DEL	Deletes one character at the current cursor position
ERASE EOF	Erases remainder of line from current cursor position
INS MODE	Allows for insertion of characters at the current cursor position. Note that the RESET key terminates INS MODE.
RESET	Cancels INS MODE and/or the INPUT INHIBITED condition. This key is on the bottom left-hand corner of the alphabetic keyboard.

In addition, characters can be replaced by first positioning the cursor at their location and then typing over them. The cursor position can be changed by using the various arrow keys on the keyboard.

Changes made with local editing keys are only transferred to the Editor's temporary copy of the file when a Program Function key or the ENTER key is pressed. (Program Function Keys, are action keys and are described below.

If you inadvertently make unwanted changes when using local editing keys, the CLEAR key cancels the effect of all screen changes made since the last action key was pressed.

As mentioned above, changes made to the Editor's temporary copy of the file are only transferred to the permanent copy when a SAVE or FILE command is typed in.

Action Keys

The keys which cause an interaction with the system are referred to as *action keys*. They are ENTER, the program function keys F1 to F24, CLEAR, PA1, PA2, and SYS RQ (or TEST REQ).

The screen cursor may be used to move the current line pointer. Simply use the arrow keys to place the cursor at the desired new current line. That line becomes the new current line before any command area commands or function key operations are done. Of course, a command or function key operation may further move the pointer. Positioning the cursor at the `*Top of file` line is equivalent to a TOP command. Positioning it at the `*End of file` line is equivalent to a BOTTOM command.

The following describes the default definitions of the action keys. It is possible for your private EDITOR file or the system *COM:EDITOR file to change the definitions of the program function keys. Enter the command SHOW PF to see the actual function key definitions in effect for your edit session.

Key	Function
ENTER	This key transmits screen changes, input lines, and commands to the system. It can also be used to advance to the next screen when `More...` appears in the bottom right corner. If no other keys were pressed since the last action key, ENTER moves the cursor to the current line (if the Editor is in command mode) or terminates Input mode (if the Editor is in Input mode).
F1	(HELP) provides information about how to use the Editor. You are presented with a list of topics and are asked to enter the number(s) of the topic(s) you want more information about.
F2	(SPLIT) splits a line into two lines. The character at which the cursor is positioned becomes the first character of the second line.
F3	(QUIT) terminates the Editor session without performing any save operation. If you have made changes to the file but have not issued a SAVE command to make the changes permanent, you will be prompted for permission to end the edit session. Enter YES or Y to end the session, or NO or N to cancel the QUIT operation and continue editing.
F4	(MARK) is used to designate (mark) a line or group of lines. To mark a group of lines, mark the first and last line of the group. The marked group can be used by commands such as MOVE., COPY., DELETE., and STORE. Marked lines are displayed with a vertical bar character in the left margin of the screen.
F5	(CENTER) causes the screen to be redisplayed so that the current line (or the one containing the cursor when the PF key is pressed) appears in the middle of the screen display. The screen display is shifted the appropriate number of lines up or down in the file in order to center the current line. The UPWINDOW and DOWNWINDOW commands perform a similar function.
F6	(DELETE) removes the current line from the file. Note that if the cursor points to a line on the screen when the PF key is pressed, that line is the one which is deleted. This is because the current line pointer is moved to the cursor before the PF request is done. The same applies to the other PF keys. To delete a line, position the cursor to it and press PF6.
F7	(UPPAGE) displays the previous page (screen) in the file (towards the beginning of the file).
F8	(DOWNPAGE) displays the next page (screen) in the file (towards the end of the file).

F9 (LOCATE) locates the next occurrence of the character string used on the last LOCATE, SEARCH, FIND, HUNT, UPLOCATE, or UPFIND command. The search starts at the line following the current line. If the CURSOR LOCATE command is in effect, the cursor will be positioned at the start of the found string when the screen is displayed.

F10 (INSERT) inserts a blank (null) line after the current line. This is useful for adding a single line to the file. You can type the line over the inserted blanks.

F11 (INPUT FLIP) puts the Editor into Input mode. This is useful for entering several lines after the current line. In Input mode, the screen displays the current line and a few lines above it, followed by null lines in the remainder of the screen. The user types over these lines in order to enter the new lines. The NEW LINE local key is used to go to the next line. If more space is needed, press the ENTER key. If the Editor is already in input mode, PF11 terminates Input mode. In this way, PF11 *flips* back and forth between Input and command mode.

F12 (CMDPFK) moves the cursor to the command area. This is necessary for entering Editor commands that are not provided for by PF keys. The arrow keys may also be used to place the cursor in the command area prior to entering commands. PF12 and 24 also terminate Input mode. The Editor command corresponding to this PF operation is CMDPFK.

PA1 Not normally used. Refer to the description of the screen message area (above) and the S option on the SCAN and CHANGE commands (below) for more information.

PA2 This key is used for multi-session control. It adds or deletes a MUSIC/SP session, or switches to the previous or next session.

Note: Make sure you have pressed ENTER or one of the PF keys before pressing PA2. Otherwise any screen changes you have entered since the last action key are lost. If you press PA2 by mistake and do not wish to go to another session, press the ENTER key.

CLEAR Cancels the effect of any screen changes made since the last action key was pressed, redisplays the current screen, and places the cursor in the command area. Input mode is terminated if it was in effect.

SYS RQ
TEST REQ This key is normally not used. It has the same effect as the CLEAR key, except that the work-station is placed into MUSIC attention mode. You may then enter an attention mode command (such as /TIME) or a blank line. The Editor eventually redisplays the original screen (you may need to press the ENTER key). Use of the TEST REQ key may be prohibited or restricted by the system environment.

Other action keys should not be used.

The ERASE INPUT key should not be used either. It cancels all screen changes made since the last action key was pressed and erases the screen display. If you hit the ERASE INPUT key by mistake, press CLEAR to redisplay the screen.

After pressing an action key, the INPUT INHIBITED indicator is on briefly while the system processes the request. On 3277 terminals, INPUT INHIBITED is indicated by a bright square at the right of the screen. On newer models of workstations, it is indicated by a large X at the bottom of the screen. If INPUT INHIBITED comes on before an action key is pressed, it means that you have tried to modify a protected part of the screen; press the RESET key to continue.

Order of Operations

It is possible to combine several operations at the workstation before pressing an action key. Operations are done by the system in the following order:

1. Changes made to text on the screen.
2. Movement of the current line pointer to the line indicated by the cursor.
3. Command-area commands.
4. Program function key operation.

Note that the cursor must be placed into the command area before typing an Editor command. The cursor can be positioned by using any of the arrow keys. It can also be put into the command area by using the F12 key.

If the Editor is in Input mode when a function key is used (standard or user-defined), Input mode is terminated before the function key request is done.

Retrieving the Previous Command Line

To retrieve the previous command line entered in the command area, place the cursor at the tab line before pressing an action key. This redisplays the last non-blank text entered in the command area. You may then modify and re-issue the command, or the command may be re-issued (as is) by simply pressing the ENTER key.

Defining Function Keys and the X Command

Users editing in full-screen mode on 3270-type workstations may customize the operation of the program function (PF) keys. This is done by the DEFINE Editor command, which defines a function key as equivalent to any string of Editor commands. Any function keys defined in this way override the standard default function key definitions.

Note: It is possible that your installation has changed the standard default function key assignments, in which case the function definitions described above may not be accurate. You can check this by entering the command SHOW PF.

For workstations that have no function keys or equivalents use the DEFINE command to make the X command equivalent to a string of Editor commands. Then typing the command X causes the specified sequence of commands to be executed. Also, the Xn command (where *n* is a number from 1 to 24) can be used on any type of workstation to execute the command string defined for PFn.

Format:

```
DEFINE PFn  commands
       X
```

n is the program function key number (1 to 24). *commands* are a string of 1 or more Editor commands, separated by the command delimiter character (normally semicolon, ";"). The DEFINE command must be the only command on the input line. An X command string must not contain an X command.

If the command string is omitted, the function key or X command is made undefined.

The abbreviation DEF may be used for DEFINE.

Create Your Own EDITOR

If you wish you can create your own Editor file to define function keys in a more permanent fashion. The file must be called "Editor" and the first line in this file must be:

```
/INCLUDE *COM:EDITOR
```

The following is an example of a file which contains function key definitions that can be used for your own private Editor. These definitions are recommended for word processing applications.

```
/INC *COM:EDITOR
def pf18 unmark
def pf19 del.
def pf20 move.
def pf21 copy.
```

Figure 3.8 - Sample Editor File

The command SHOW MYPFK permanently displays the contents of file MYPFK at the bottom of the screen. This file could indicate your function key definitions, as a reminder during the edit session.

The SHOW command is used to display the current definition of function keys or the X command: SHOW PFn, SHOW PF, or SHOW X.

Using Full-Screen Mode Without Function Keys

If you wish to use full-screen mode with a 3270-type workstation which does not have any program function keys, use the command FS NOPFK. This causes the Editor to put the cursor in the command area whenever the screen is displayed, thus facilitating the entry of commands. To move the cursor to the current line, leave the command area null and press the ENTER key, or simply use the arrow keys.

In addition, the commands UPWINDOW, DOWNWINDOW, UPPAGE and DOWNPAGE may be used in place of function key operations.

Simulating Additional Function Keys

It is possible to make use of all 24 PF key operations even if your terminal only has keys for PF1 to 12. The extra keys for PF13 to 24 are simulated by placing the cursor into the left-hand margin of the screen, one column to the left of a screen line or the command area, and then pressing the corresponding PF key (1 to 12). The Editor detects the special cursor position, and simulates the appropriate PF key (it adds 12 to the number).

In this way, the LEFT ARROW (<--) key acts somewhat like a *shift* key for the PF keys. (If you are NET3270 then SHIFT-F1 to SHIFT-F12 correspond to F13 - F24.)

The special cursor position is screen column 4 (without line numbers) or column 7 (with line numbers) for a

line of the file, and is column 9 for the command area line.

If the cursor is in the margin column when a real PF13 to 24 is pressed, then the Editor will simulate the corresponding PF1 to 12 key (it subtracts 12 from the number).

Blank-Filled Screen Display

Normally, when the screen is displayed, blanks at the end of each field (i.e. each line of the screen) are replaced by null characters. This allows easy use of the INS MODE key for inserting characters in the middle of a line. When adding characters to the end of a line, be careful to use the space bar to enter blanks, rather than the arrow keys, since null characters are not transmitted to the Editor.

However, for some applications such as entering tables or diagrams, it is more convenient to have trailing blanks retained. This is requested by using the FILL command. If the INS MODE key must be used when FILL is in effect, first use the ERASE EOF key to remove trailing blanks from the field. The NOFILL command reverts to the normal method of display.

An alternate name for the FILL command is NONULLS; an alternate name for the NOFILL command is NULLS. The command NULLS FLIP reverses the NULLS setting.

FILL (or NONULLS) mode is indicated by the ")" character at the end of the tab line.

Output Cursor Positioning

When the Editor displays the screen, it normally puts the cursor at the first position of the current line or the command area. However, commands such as CURSOR, LOCATE and SPLIT can cause the cursor to be displayed at a different position in the current line.

The CURSOR command controls output cursor placement:

```
CURSOR LOCATE
CU     NOLOCATE
       n
       n TEMP
       END
```

n is a number from 1 to 72. It is a column position number relative to the start of the screen field for the current line. 1 refers to the first character of the field, 2 to the second character, etc.

CURSOR LOCATE places the cursor at the start of the found string after any of the commands: LOCATE, UPLOCATE, FIND, UPFIND, SEARCH, HUNT. CURSOR NOLOCATE cancels this option. You can put a CURSOR LOCATE command into your private EDITOR file to make this option the default for all your edits. Abbreviations are LOC, NOLOC.

"CURSOR n" places the cursor at the specified column position whenever the cursor is not put out in the command area, provided a command such as LOCATE or SPLIT or CURSOR END or "CURSOR n TEMP" does not result in a different placement. This stays in effect for the remainder of the edit. To cancel the effect of this command, use CURSOR 1.

"CURSOR n TEMP" places the cursor at position n in the screen field for the current line, but only for the next screen display. Abbreviation T may be used for TEMP, as in CU 15 T. This command is intended mainly for function key definitions. For example, DEFINE PF1 INSERT ABC---XYZ;CURSOR 4 TEMP.

CURSOR END is similar to "CURSOR n TEMP", except that the cursor is placed after the last nonblank character in the field.

RIGHT and LEFT Options on the WINDOW Command

The option RIGHT or LEFT may be used on the WINDOW command to shift the window towards the right or left hand side of the records being edited. The width of the window is not changed. Optionally, the number of columns for the shift may be specified. Also, the FLIP option can be used to shift the window to the extreme left or right, whichever is further from the existing window setting. These options may be used whether or not full-screen mode is in effect. The format of the commands is:

```
WINDOW RIGHT

WINDOW RIGHT n

WINDOW LEFT

WINDOW LEFT n

WINDOW FLIP
```

The parameter *n* is the number of columns by which the window is to be shifted. If *n* is omitted, the window is shifted by the maximum amount. RIGHT and LEFT may be abbreviated R and L. WINDOW FLIP shifts the window to the extreme opposite side.

When a WINDOW command is used in full-screen mode, the zone is automatically set to be from column 1 to the end of the window (refer to the ZONE command). So remember that if you have used a command such as ZONE 80, you must re-issue it after shifting the window left or right, otherwise the zone may be reset to 72. The ZONE setting affects which columns are changed by the CHANGE command.

Example: Assume the length of the records being edited is 80, and the current window is columns 1 to 72. Then "WINDOW RIGHT 2" results in a new window setting of 3, 74. WINDOW RIGHT or WINDOW FLIP results in a window setting of 9, 80.

Suggestions for Efficient use of Full-Screen Mode

- If your 3270-type workstation has more than 12 function keys, set up a private EDITOR file to define extra function key operations in addition to (or in place of) the standard ones. Using function keys is faster and easier than typing commands. Good candidates for function key operations are the commands: MOVE., COPY., DELETE, DELETE., WINDOW FLIP, JOIN, TOP, BOTTOM, DUP, UPLOCATE, MINSERT, MDELETE, UPWINDOW, DOWNWINDOW, FILE, ECHO, CURSOR END, CURSOR n TEMP, NULLS FLIP, NUMBER FLIP, SUBMIT.

- It is more efficient to use the NEW LINE key or TAB key after each line in input mode. The NEW LINE key has a *down and to the left* arrow on it, and is located on the right-hand side of the alphabetic keyboard. This gives much better response in Input mode, since no system interaction is required (NEW LINE and TAB are local keys). Once the screen is full or nearly full with input lines, press the ENTER key to get a fresh screen. Outside of Input mode, the NEW LINE or TAB key is often quicker than the command function key (F12) for positioning the cursor in the command area. For NET3270, the NEW LINE key is Ctrl-Enter, or the * on the right-hand keypad.

- When you know that you will be typing a command in the command area (rather than making a change

on the screen), end the previous action by using the command function key (F12) instead of the ENTER key. This ensures that the cursor will be positioned in the command area, and may eliminate one interaction with the system.

- Several Editor commands can be entered in the command area by separating them by a semicolon (;). This technique can really speed up editing, but use it carefully, since an error on one of the commands does not stop execution of the remaining commands.

Marking a Group of Lines

When editing a file, the need often arises to define a set of consecutive lines within the file, and then perform some operation on those lines. Examples of this are moving or copying a group of lines from one part of the file to another, restricting a global change to a section of the file, converting a set of lines to upper case, and deleting a group of lines.

The MARK command and the *dot* forms of commands such as MOVE, COPY and CHANGE, provide this capability in the MUSIC Editor. Use of marked groups greatly reduces the need for line numbers while editing.

To define a group of lines, the user moves the line pointer to the first line of the group and issues the MARK command, then moves the pointer to the last line of the group and issues the MARK command again. Once the group has been defined, it can be referred to in various commands by using a period (.) as a parameter. For example, the command CHANGE/ABC/DEF/.G changes every occurrence of ABC to DEF within the group. The commands MOVE. and COPY. move and copy the group. These and other "." commands are described below.

Although the MARK command and the various . commands can be used on any type of workstation, they are best suited to full-screen mode on a 3270-type workstation. In full-screen mode, each line of a marked group is identified by a vertical bar in the left margin on the screen. The user's private EDITOR file would define a program function key as MARK, and probably two other keys as MOVE. and COPY..

Some Notes on the MARK Command

Only one set of consecutive lines can be marked at a time. The marked group is maintained until a new group is defined, or until an UNMARK command is issued, or until all the lines of the group are deleted. As new lines are added to or deleted from the file, the marks are adjusted accordingly. The command MARK ? reports which lines are currently marked. The blank between MARK and ? may be omitted. Abbreviations for the MARK and UNMARK commands are MA and UNMA.

When defining a group, the first and last lines may be marked in either order. The group may consist of only one line, in which case the second MARK command is not needed. Once a multi-line group has been marked, or a 1-line group has been marked and used in a . command, a subsequent MARK command starts a new group.

Commands for Operating on a Marked Group

In these commands, the . character is a command parameter, and the blank between it and the command name may be omitted. Also, the command name may be abbreviated, as in MO., CO., and DEL..

= . Moves the current line pointer to the first line of the group. This is handy for returning to a particular point in the file. The command is usually entered as =. to save typing.

MOVE . Moves the marked group of lines to after the current line. The group remains marked. For example, if lines 51 through 75 are marked and the current line is 20, then MOVE. is equivalent to MOVE 20 51 75

COPY . Copies the marked group of lines to after the current line. The original group remains marked.

DELETE . Deletes the marked group. Note that the MDELETE command is another (sometimes faster) way of deleting a block of lines. The operation may be qualified by a logical expression as the second parameter, as in DEL.,(NOT/ABC/), which deletes all lines of the group which do not contain ABC.

CHANGE A period (.) can be used in place of the *n* (line count) option on a CHANGE or logical CHANGE (CHANGEL) command. This causes the change to apply to each line of the marked group (subject to the logical expression, if CHANGEL is used). The Editor automatically goes to the first line of the group before starting the change. Example: C/ABC/DEF/.F

REPEAT . Causes the next BLANK or OVERLAY command to apply to each line of the marked group. It also moves the current pointer to the first line of the group, and places the screen cursor into the command area.

TOUC .
TOLC . Converts the marked group to upper or lower case.

COPYCOL n1 n2 n3 .
When . is used as the 4th parameter, instead of a line count, the COPYCOL operation is applied to each line of the marked group.

STORE name [APPEND]
Writes the marked lines to the specified file.

Workstations unable to use Full-Screen mode

For ASCII-type (TTY) terminals, a question mark (?), automatically used as the prompt character, is displayed each time the Editor expects the user to type a line. This avoids the problem of the user starting to type before the workstation is ready to accept input. If desired, prompting may be removed by typing 'prompt' without a parameter. On an IBM 2741, the keyboard locks when the Editor is working.

The Editor has a special *screen mode* operation, intended mainly for display terminals. 3270-type terminals have screen mode but also have *full-screen mode* which has been described previously. In screen mode, the current line is displayed in the upper part of the screen, along with several preceding and following lines of the file. The current line is surrounded by lines of equal signs (=) and the number of the current line appears near the end of the first equals sign line.

This *window* is re-displayed whenever changes are made to the file. When the Editor starts, screen mode is on for 3270-type workstations. It may be turned off with the command NOSCREEN.

For some workstations, "screen mode" is off at the start of an editing session, but can be turned on by using the command SCREEN.

The number of lines to be displayed can be adjusted by specifying a number on the SCREEN command. The default is 15.

Editor Commands - Overview

This section discusses the most commonly used Editor commands. However, when a specific problem or need arises, you may consult the *MUSIC/SP User's Reference Guide* where a more complete discussion of the Editor and its commands can be found.

If you are working from a 3270-type workstation with full-screen mode, you can use the function keys and local editing keys instead of many of the Editor commands listed below. Full-screen mode was described previously in this chapter.

Obtaining Editor Help

The HELP command is used to obtain information about a particular Editor command, or about the Editor in general. Type HELP in the command area of the Editor.

HELP without a parameter gives general information about the Editor, such as a one-line description of common commands.

"HELP name", where *name* is the full name of an Editor command, display information about the command. If only selected information is desired, one or more topics may be specified after the command name, "HELP name list-of-topics".

Functional Summary of Commands

The following is a complete list of all the Editor commands broken down into groups by function. If an asterisk (*) appears beside the Editor command, it indicates that this command is not described in this guide. These commands are infrequently used and are described in the *MUSIC/SP User's Reference Guide*.

Moving the line pointer

TOP	Move the pointer to the first line of the file.
BOTTOM	Move the pointer to after the last line of the file.
LAST	Move the pointer to the last line of the file.
UP	Move the pointer towards the beginning of the file.
NEXT	Move the pointer towards the end of the file.
LOCATE	Locate the next line containing a specified character string.
ULOCATE	(Upward LOCATE) Move towards the top, looking for a string.
XL	Macro to locate a specified string without moving the current line pointer if the string is not found.
* LOCATEL	Locate the next line which tests true against a logical expression.
FIND	Same as LOCATE, except the string must start in column 1.
UFIND	(Upward FIND) Same as ULOCATE, but string in column 1.
SEARCH	Same as LOCATE, except start at the top of the file.
* SEARCHL	Same as LOCATEL, except start at the top of the file.
HUNT	Same as FIND, except start at the top of the file.
=n	Move the pointer to line number *n* (type "HELP =" for info).

Displaying lines

	PRINT	Display a number of lines, starting with the current line.
	LIST	Display the file being edited, or an external file.
*	SCAN	Display all lines containing a specified character string.
	NUMBER	Turn line numbering on.
	NONUMBER	Turn line numbering off.
	WINDOW	Define starting and ending columns to be displayed.

Making changes to the file

	CHANGE	Make a change to a line or group of lines.
*	CHANGEL	Make a change to a line or group of lines.
	ADD	Add text to the end of a line.
	REPLACE	Replace an entire line.
	DELETE	Delete the current line from the file.
*	DELETEL	Delete a group of lines from the file.
*	MDELETE	Delete MINSERT unused lines or a group of lines.
*	UNDELETE	Inserts the line deleted by previous DELETE 1 command.
	INSERT	Add a new line following the current line.
*	MINSERT	Add a group of new lines following the current line.
	INPUT	Begin INPUT mode, to add lines to the file.
	DUP	Duplicate the current line a no. of times.
	MERGE	Bring a copy of an external file into the file being edited.
	MOVE	Move lines from one place to another within the file.
	COPY	Copy lines from one place to another within the file.
*	COPYCOL	Copy text from one part of a line (or lines) to another part.
*	REPEAT	Specify the no. of times to repeat the next BLANK or OVERLAY.
*	BLANK	Set specified characters of a line to blanks.
*	OVERLAY	Overlay part of a line with new characters.
	SPLIT	Break the current line into two lines.
	JOIN	Join the next line to the end of the current line.
	SORT	This macro uses the MUSIC SORT command to sort part or all of the file.
	UNSORT	This macro undoes the effect of the SORT macro.
	SPELL	This macro uses the SPELL program to spell check your file.
	FORMAT	This macro formats text using the SCRIPT program.
	UNFORMAT	This macro undoes the effect of the FORMAT macro.

Getting information

	HELP	Request information about the Editor or an Editor command.
	=	Get the line number of the current line.
*	SIZE	Get the total number of lines in the file.
	NAME	Display or set the file name for the edit session.
	TAG	Display or set the tag string of the file being edited.
*	TIME	Display time of day, date, cpu time, and number of users.
*	USERS	Same as TIME.
	SHOW	Display function key or X command definitions; display from a file.
	NOSHOW	Remove text displayed by SHOW command.
*	ATTRIB	Get information (attributes) about a file.
*	SPACE	Get Save Library space information for your userid.

Ending the edit

QUIT	Terminate the edit without saving the file.
* END	Same as QUIT.
* QQUIT	(Quick quit) terminates without any messages or prompting.
* OFF	Same as QUIT, but also terminates the MUSIC session.
FILE	Store the changed file in place of the original file.
EXECUTE	Store the file (as in FILE command) and then execute it.
* RUN	Same as EXECUTE, except use the MUSIC input file (/INPUT).
SAVE	Similar to FILE, but do not terminate the edit.

Full-Screen Mode

* FS	Start full-screen mode (for 3270-type workstations).
* NOFS	End full-screen mode.
* FILL	Put blanks at the end of each screen field.
* NOFILL	Put nulls at the end of each screen field (the default).
DEFINE	Define a program function (PF) key or the X command.
* ECHO	Cause information to be displayed in the command area.
UPWINDOW	Shift the screen display towards the beginning of the file.
DOWNWINDOW	Shift the screen display towards the end of the file.
UPPAGE	Shift the screen display 1 screen towards the beginning of the file.
DOWNPAGE	Shift the screen display 1 screen towards the end of the file.
CMDPFK	Cause the cursor to be put into the command area.
CENTER	Center the current line in the middle of the screen.
CURSOR	Control the placement of the cursor on the screen.

Other commands

* AIN	Terminate hexadecimal input mode.
* ALPHA	Terminate hexadecimal output mode.
* ARROW	Use arrow pointer in ASCII screen mode.
* NOARROW	Do not use arrow pointer in ASCII screen mode.
* BOTH	Obtain output in both hexadecimal and character form.
* BRIEF	Stop automatic verification (displaying) of file changes.
* CALC	Macro to evaluate a REXX expression, in particular, calculations.
CASE	Ignore or respect case differences during string searches.
* CMDS	Controls execution of MUSIC commands during an edit session.
* CREP	Enables or suppresses automatic command name replacement.
DEFINE	Define a program function (PF) key or the X command.
DELIM	Change or remove the command separator character (normally ;).
* FLIP	Define a *flip* character, used to control verification.
* HEX	Obtain output in hexadecimal form.
* LOG	Control frequency of writing to log file (restart feature).
MARK	Define a group of lines to be operated on.
UNMARK	Undefine a group of lines to be operated on.
* MSG	Display a message line on the workstation.
* MSGS	Suppress or enable all messages.
* NOCHANGE	Informs the Editor that there are no unsaved changes.
* PROMPT	Define or remove the prompting character.
PURGE	Remove (delete) a file permanently from the Save Library.
RENAME	Change the name of a file.
* REXX	Enable or disable the use of Rexx procedures as Editor commands

* SCREEN	Turn ASCII screen mode on.
* NOSCREEN	Turn ASCII screen mode off.
* SEQ	Put sequence numbers into each line of the file.
STORE	Store the marked group of lines to an external file.
* SUBSET	Define which subset of Editor commands is to be allowed.
* TABIN	Specify input tab positions.
* TABOUT	Specify output tab positions.
TEXT	Specify handling of input lower case and tab characters.
TOUC	Change characters to upper case.
TOLC	Change characters to lower case.
* TRAN	Turn output character translation on.
* NOTRAN	Turn output character translation off.
* VERIFY	Control verification (displaying) of changes to the file.
X	Execute a predefined string of commands (see DEFINE).
* XIN	Begin hexadecimal input mode.
ZONE	Define the ending column for some commands such as CHANGE.
* ASMFIX	Align assembler source statements to specified columns.
FLAG	Begin automatic flagging of changed lines.
NOFLAG	Stop automatic flagging of changed lines.
* DELCHAR	Define the delete character to be used by MDELETE & MINSERT command.
* PRINT	Print a file on a specified line printer.
* SUBMIT	Submit a job to MUSIC batch or other batch processors.
* *	Signify a comment line.

New Commands

The following commands or macros have been added to the Editor for MUSIC/SP V2.3. For full details about these commands and other new features of the Editor, type "HELP" in the Editor's command area.

* AUTOSKIP	cause automatic cursor skip at end of text fields.
* BEEP	beep the speaker the next time the screen is displayed.
* BR	(macro) uses show area to display a file.
* COLOR	define the color of various parts of the screen.
* GETV	(macro) displays the value of a SETV name.
* FF	(macro) format marked lines or entire file using SCRIPT.
* LEFT	shift the window display left.
* MARGINS	(macro) adjust margins before using FF.
* OKREPL	suppress FILE/SAVE verification prompt for a file name.
* POINT	assign a 1-8 character label to a line.
POWERINP	begin Power Input mode.
PREFIX	turn on or off the modifiable prefix area.
* RIGHT	shift the window display right.
* SET	(macro) compatible command with CMS editor.
* SETRC	set a job return code (exit code) for the edit.
* SETV	assign a character string yyy to a name xxx.
* UNFF	(macro) restore file to original form before FF was used.

Editor Commands - Descriptions

Notation

The notation *string* indicates a sequence of blanks, letters, etc. Leading blanks in character strings are considered part of the string starting with the second position after the command name. In the command descriptions, square brackets are used to indicate optional items.

Command Syntax

Command parameters may be separated by one or more commas or blanks. A blank is not required between the command name and the parameter unless the first character of the parameter is a letter (A to Z). A blank is not required following a flip character or column number suffix. The following commands require a blank in any case: SAVE, FILE, EXEC, RUN, LIST, MERGE, PURGE, INPUT, NAME, STORE. Commands may not be preceded by a slash (/).

In general, any abbreviation between the shortest abbreviation and the full command name may be used. For example, the following may be used for the CHANGE command: C, CH, CHA, CHAN, CHANG, CHANGE. An exception is the PRINT command, which allows P but not PR (PR is the PROMPT command). Another exception is the INSERT command, which allows I and INS but not IN, to avoid confusion with the INPUT command. In the following command descriptions, the shortest abbreviations are shown under the full form of the command.

The command delimiter character (normally semi-colon), the flip character, and all string delimiters must be special characters. They may not be letters, digits or blanks.

In logical expressions, the string modifiers F and Cn (where n is a column number) may be separated by blanks or commas, or the separator character may be omitted entirely. For example, (/ABC/(FC10)). The same applies to options on the CHANGE and SPLIT commands. Examples: C/ABC/DEF/*GV, SPLIT/XXX/C5IC10.

Upper and Lower Case

The user is reminded that commands and data may be entered in upper or lower case, since (unless translation is bypassed by using the MUSIC command TEDIT or the Editor command TEXT LC) lower case input is automatically translated to upper case. Even when TEXT LC is in effect, command names and keywords (such as the options V and G on the CHANGE command) may still be entered in upper or lower case. Although most of the examples given here use upper case, the user will normally find it more convenient to type in lower case.

Current Line Pointer

In the following descriptions the current line pointer is assumed to be unchanged after the command is issued. Exceptions to this are noted under those commands that do move it.

String Separator

Commands such as CHANGE require that strings be separated by a string separator character. The usual character used is a slash (/), but any non-alphanumeric character except the command delimiter character

may be used. The string separator character is self-defining. For example, the first nonblank character after the command CHANGE is automatically defined as the string separator for that command.

```
ADD  string
A
```

This command adds *string* after the last non-blank character of the current line.

Spacing: One blank character must occur between the keyword and the first character of *string*. Any additional blanks are considered part of *string*.

Example: Before: **SAMPLE LINE**
Command: **ADD 1234**
After: **SAMPLE LINE1234**

```
BOTTOM
B
```

The BOTTOM command moves the pointer to one line after the last line in the file.

Example: Before: **SAMPLE LINE <--Pointer**
ANOTHER LINE
THE LAST LINE
Command: **B**
After: **SAMPLE LINE**
ANOTHER LINE
THE LAST LINE
<--Pointer

```
CASE [IGNORE   ]
CA   [I        ]
     [RESPECT  ]
     [R        ]
     [MIXED    ]
     [M        ]
     [UPPERCASE]
     [U        ]
```

The CASE command specifies whether or not differences between upper and lower case characters should be ignored when the editor is searching for a string of characters (IGNORE/RESPECT), and/or whether text entered by the user should be left as is (MIXED) or converted to upper case (UPPERCASE).

The IGNORE/RESPECT setting affects all commands which involve string searches, such as LOCATE, LOCATEL, SCAN, CHANGE, etc. It does not affect changes typed over screen text in full-screen mode, or *string2* of the CHANGE command, or commands such as ADD or INSERT.

With CASE RESPECT, upper and lower case characters are considered different when searching.

With CASE IGNORE (the initial default), upper and lower case characters are considered the same, for the purposes of searching for a specified string of characters. This is as if all characters were in upper case. The file *COM:EDITOR normally contains the command CASE IGNORE, so that setting is the default for most edits.

When CASE is specified without a parameter, the current IGNORE/RESPECT setting is shown. To show the current UPPER/MIXED setting, use the TEXT command without any parameter.

The command CASE MIXED is equivalent to the command TEXT LC. CASE UPPERCASE is equivalent to the command TEXT UC.

More than one parameter can be used on a CASE command. For example, CASE UPPERCASE RESPECT. This can be abbreviated to CA U R.

Example:

```
CA I   The command "LOCATE abc" finds "ABC" and "aBc" as well as "abc".  The
       command "CHANGE/mcgill/McGill/*" changes "Mcgill", "MCGILL", "MCGill",
       etc. to "McGill" (assuming TEXT LC is in effect).
```

CENTER
CE

The CENTER command moves the full-screen display up or down in the file, so that the current line will be in the center of the screen display. By default, F5 is defined as CENTER. To shift the display up or down in the file, place the cursor at a line and press F5 to center that line.

CHANGE [/string1/string2/][n][G][D][F][Cn][V][P][S]
C

The CHANGE command replaces the first occurrence of a character string with another in a number of lines (normally 1), beginning with the current line. Within the range of lines, only the first occurrence of the string in each line is replaced. The number of lines searched depends on the parameters specified in the command. The last line searched becomes the current line. If the line pointer is at EOF (end of file), an automatic TOP command is done before the change.

When a CHANGE command is entered with no parameters, the editor's CHANGE Panel is displayed. The CHANGE Panel lets you do a change operation by filling in fields on the panel.

The search for the string is affected by whether or not CASE IGNORE is in effect. CASE IGNORE ignores

differences between upper and lower case characters. Refer to the CASE command.

The parameters n, G, D, F, Cn, V, P, S are not required. If used, they can be specified in any order. Commas are not required to separate the parameters.

Parameters:

string1 Character string which is to be replaced.
string2 Replacing character string. If omitted, *string1* is deleted. The final string delimiter (/) may be omitted if no parameters are used.
n Number of lines searched. If not specified, 1 is assumed. If * is specified, the search will be applied to the rest of the file, starting with the current line.
G If specified, the change will be applied to ALL occurrences of *string1* in as many lines as specified by the *n* parameter.
D If specified, the remainder of the line (or zone) following the changed string is replaced by blanks. ("D" stands for "delete".)
F If specified, the change is made only if "string1" begins in column 1 of the line (or in column n if Cn is used). ("F" stands for "first" or "FIND mode".)
Cn If specified, the change is made only if "string1" occurs starting in column n or later.
V If specified, all lines which are changed are displayed regardless of whether the editor is in the BRIEF status or the VERIFY status. If V is not specified and the range is more than one line, the changed lines are not displayed. ("V" stands for "verify".)
P If specified, the editor prompts for the user's permission to make a change. The user can respond Y (for yes), N (for no), S (to stop the execution of the CHANGE command), G (to continue command execution without further prompts), or = (to find the current line number).
S If specified, it causes the screen to be displayed for each line found (as in 3270 "screen mode"). On a 3270-type workstation, the PA2 key must be pressed between screens.

Examples:

`C/ABC/DEFG/`	Changes the first occurrence of ABC to DEFG in the current line. The line pointer is not changed.
`C/OLD/`	Removes the characters OLD from the current line.
`C/OLD//3,G`	Removes all occurrences of OLD from the next 3 lines, beginning with the current line. The line pointer is moved down 2 lines.
`CH /XXX/YY/ 20,F,C11`	In the next 20 lines, changes XXX to YY wherever XXX occurs starting in column 11.
`C/ /*/G`	Changes every blank to * in the current line.
`C//AAA/`	Adds AAA to the beginning of the line.
`CHANGE/E/ES/*GV`	This will change all occurrences of E to ES in the rest of the file, starting with the current line. Changed lines are displayed.
`CHANGE$/$.$`	This changes "/" to "." in the current line. The dollar sign ($) acts as the string delimiter in this case.

CMDPFK
CMDPF

This command is used only in 3270 full-screen mode, and is intended primarily for use in function key definitions. It causes the cursor to be put into the command area the next time the screen is displayed. The standard definition of F12 is equivalent to a CMDPFK command.

```
COPY  i,j,k
CO
```

This form of the COPY command allows copying lines by reference to the line numbers of the lines. The block of lines from line *j* to line *k* inclusive is copied to after line number *i*. (Line numbers may be determined by using the = command.) The original lines are not deleted. If *i* is specified as 0, the lines are copied to the beginning of the file. If *j=k*, only one line is copied.

Line numbers may also be specified in any of the following forms: *n, *+n, *-n, LAST, LAST-n,* where *n* is a number, * means the current line, and LAST means the last line of the file. Example: COPY LAST,*,*+3.

For the command "COPY.", used to copy a marked group of lines, refer to the topic "Marking a Group of Lines" later in "Advanced Features".

Pointer: Set to the last line of the inserted section.

Example:

```
Before:   A1
          A2
          A3
          A4
Command:  COPY 4,1,2
After:    A1
          A2
          A3
          A4
          A1
          A2
```

```
CURSOR LOCATE
CU     NOLOCATE
       n
       n TEMP
       END
```

The CURSOR command controls output cursor positioning during full-screen mode.

When the Editor displays the screen, it normally puts the cursor at the first position of the current line or the command area. However, commands such as CURSOR, LOCATE and SPLIT can cause the cursor to be displayed at a different position in the current line.

n is a number from 1 to 72. It is a column position number relative to the start of the screen field for the current line. 1 refers to the first character of the field, 2 to the second character, etc.

CURSOR LOCATE places the cursor at the start of the found string after any of the commands: LOCATE, UPLOCATE, FIND, UPFIND, SEARCH, HUNT. CURSOR NOLOCATE cancels this option. You can put a CURSOR LOCATE command into your private EDITOR file to make this option the default for all your edits. Abbreviations are LOC, NOLOC.

CURSOR *n* places the cursor at the specified column position whenever the cursor is not put out in the command area, provided a command such as LOCATE or SPLIT or CURSOR END or "CURSOR *n* TEMP" does not result in a different placement. This stays in effect for the remainder of the edit. To cancel the effect of this command, use CURSOR 1.

CURSOR *n* TEMP places the cursor at position *n* in the screen field for the current line, but only for the next screen display. Abbreviation T may be used for TEMP, as in CU 15 T. This command is intended mainly for function key definitions. For example, DEFINE PF1 INSERT ABC---XYZ;CURSOR 4 TEMP.

CURSOR END is similar to CURSOR n TEMP, except that the cursor is placed after the last nonblank character in the field.

Examples:

```
CURSOR 41
CU 15 T
CUR LOC
```

```
DEFINE  Fn      [commands]
DEF     PFn     <TAB>
        X       <RETRIEVE>

DEFINE  CMD xxx yyy
DEF     CMD 0
        PREFIX pppp tt mmmmmmmm
```

You can customize the operation of the program function keys by using the DEFINE command, which defines a function key as equivalent to any string of Editor commands. Any function keys defined in this way override the standard default function key definitions.

The DEFINE command can also change the definition of the X command, redefine command and macro names (by the CMD option), and define prefix area commands (by the PREFIX option). For more information type "help define" in the Editor's command area.

```
Form 1:

DELETE  [n][,(logical expression)]
DEL
```

This command deletes *n* lines from the file, beginning with the current line. If there are fewer than *n* lines left in the file from the current line to the end, those which exist are deleted and the message *EOF is displayed on the workstation. If * is specified instead of *n*, the delete will apply to the remainder of the file, starting with the current line.

If *(logical expression)* is specified, only the lines for which *(logical expression)* is true will be deleted. Parentheses are required around the logical expression. For example, the command "DEL*,(/TEST/(FC73))"

deletes all lines in the file with the characters TEST in columns 73 to 76, from the current line to the end of the file.

Following a delete, the new current line (that is, the line following the last line deleted) will be displayed if the Editor is in VERIFY mode. A flip character may be used to temporarily change to or from VERIFY mode (refer to the FLIP command).

For the command "DEL.", used to delete a marked group of lines, refer to the topic "Marking a Group of Lines" in the section "Advanced Features".

Spacing: The *n* may be anywhere on the command line after the keyword or abbreviation, or may be omitted, in which case 1 is assumed. If both parameters are specified they must be separated by a comma or blank.

Pointer: Set to the line after the last line deleted, or to after the end of the file.

Example:

```
Before:   SAMPLE LINE
          ANOTHER LINE        <--Pointer
          YET ANOTHER
Command:  DEL
After:    SAMPLE LINE
          YET ANOTHER         <--Pointer
```

```
Form 2:

DELETE /string/[,(logical expression)]
DEL
```

The current line and all lines after the current line down to, but not including, the line containing *string* are deleted. If no such line is found, an error message is displayed and the file is not changed.

If *(logical expression)* is specified, only those lines for which *(logical expression)* is true will be deleted. Parentheses are required around the logical expression.

Pointer: Set to the line containing *string.*

```
DELIM [x]
```

The DELIM command changes the command delimiter character to the character specified by *x.* The delimiter character *x* may be any non-alphanumeric character. It separates editor commands entered on the same line.

If the DELIM command is not used, the command delimiter character is assumed to be a semicolon (;) by default. If the command DELIM is entered with no parameter, then no character is the delimiter character, and only one command may be entered on a line.

The command DELIM OFF disables the delimiter character, and only one command can subsequently be entered on a line. However, the original delimiter character (if any) is remembered, and can be enabled later by the command DELIM ON. This makes it possible to temporarily undefine the command delimiter (for example, in order to use the INSERT command to insert text that may contain the delimiter), without knowing what the original delimiter character is. If DELIM ON is used after a DELIM command with no parameter, nothing is changed.

Example:

```
delim $
l xaz$c/a/y/$l abc
delim off
insert The $ in this line is not a delimiter
delim on
last$insert xyz
```

DOWN [n]
DN

The DOWN command is the same as the NEXT command. Refer to the description of the NEXT command.

DOWNPAGE
DOWNP
DNP

This command is used only in 3270 full-screen mode. It shifts the screen display to the next screen (page) in the file (towards the end of the file). The first line displayed on the new screen will be the line after the last line on the current screen. This command corresponds to F8 by default.

DOWNWINDOW [n]
DOWNW
DNW

The DOWNWINDOW command is used only in 3270 full-screen mode. It shifts the screen display (window) towards the end of the file. The parameter *n* is the number of lines by which the screen is to be shifted. If *n* is omitted, 6 lines is assumed.

```
DUP  [n]
```

This command causes the current line to be duplicated *n* times. If *n* is not specified, it is assumed to be 1.

Pointer: Set to the last of the new lines created.

Example:

```
Before:   SAMPLE LINE        <--Pointer
Command:  DUP 4
After:    SAMPLE LINE
          SAMPLE LINE
          SAMPLE LINE
          SAMPLE LINE
          SAMPLE LINE        <--Pointer
```

```
EXECUTE  [name]
EXEC
EX
```

This command saves the changed file in the Save Library under the name *name,* and then automatically requests MUSIC to execute the program contained in that file in a way equivalent to the user typing the MUSIC command "EXEC name". If *name* already exists in the user's Save Library, the Editor will prompt the user whether to replace the existing file or not. If YES (or Y) is replied, the edited file will replace the previously existing file, and the EXEC will be done. Otherwise, no operation will be done. The prompt is not done if replacing an existing file which was read by EDIT at the start of the edit. File attributes can also be specified after the file name, as in the FILE command. If no name is specified on the Editor EXEC command, the name on the original EDIT (or TEDIT) command or on the last NAME command will be used. If desired, column number limits (as on the FILE command) may be specified before *name.* The parameter *name* may be /INPUT.

If this command is used in the User Data Set version of the Editor, a file name must be specified with the command. The Editor will try to replace the named file by the temporary updated version of the UDS file being edited. If the replacement is successful, a request to execute the program in the named file will be made to MUSIC. The original UDS file is not updated, and the edit is terminated.

```
FILE  [m] [n] [name] [PRIV] [XO] [CNT]
              [*   ] [PUBL]
                    [SHR ]
```

The FILE command is the usual way of terminating an Editor session and saving the updated file. This command saves or replaces the updated file directly into your Save Library.

Parameters:

name	Specifies the name for saving (or replacing) the updated file in your Save Library. If *name* already exists in your library, the Editor will ask whether to replace the existing file or not. If the user replies with YES (or Y), the edited file will replace the previously existing file. No save operation will be done if the user's reply is not YES. Note that if *name* is not specified, the Editor will use the name which was specified on the EDIT command or on the last NAME command, or prompt you to enter a file name if no name is known to the Editor.
m n	These parameters specify the first and last column numbers, respectively, that will be saved for each line. Remaining columns are filled with blanks. If only one number is supplied it is assumed to be n and m will be assumed to be 1. The column numbers must appear before the "name" parameter. If omitted, each line will be saved in full. When you specify column numbers, the editor asks for verification before doing the command. Enter yes (or y) to allow the command to continue. Enter anything else to cancel the command. (If editor messages are currently suppressed by the MSGS OFF command, the verification is not done.)
PRIV	The file will be made private.
PUBL	The file will be publicly readable and in the common index.
SHR	The file will be publicly readable but not in the common index. Other users must specify the owner's userid in order to access the file, as in userid:filename.
XO	The file will be execute-only.
CNT	The system will maintain a usage count for the file (Save Library files only). The count is increased by 1 each time the file is opened. It is displayed by the "ATTRIB filename" command in *GO mode. Note that using the editor to modify the file resets the count to zero, since the editor always recreates the file.

Notes on File Attributes

If an existing file is edited and then saved back by using the SAVE, FILE or EXEC command without specifying a file name or attributes, all the original attributes of the file are preserved. This technique should be used when it is desired to change a file without changing its attributes. In other cases, PRIV is used as the default attribute.

Examples:

```
FILE 1,72,FILEAB
```
saves columns 1 to 72 inclusive for each line in the updated file, under the name "FILEAB".

```
FILE
```
replaces the original file with the updated version of the file.

```
FILE FILE1,PUBL,XO
```
saves the updated file as a public execute-only file in the Save Library under the name "FILE1".

FIND [string]
F

The FIND command searches the file, starting with the line after the current line, for a line beginning with

string. The search continues down the file until the first match, or until the pointer has been moved past the last line of the file (the message *EOF is displayed). If issued after the pointer is past the end of the file, an automatic TOP is performed before the search begins. If VERIFY status is in effect, the line found is displayed on the workstation.

If *string* is not specified, the same string as specified in the last used FIND, LOCATE, UFIND, ULOCATE, HUNT, or SEARCH command is used.

(The UFIND command is similar to the FIND command but searches upwards through the file.)

Spacing: One blank should be present between the keyword or abbreviation and *string*. It is ignored if present. All other blanks are considered to be part of *string*, which extends to and includes the rightmost non-blank in the command line.

Pointer: Set to the found line, or beyond the end of the file if no line is found which begins with *string*. (Message *EOF is displayed in this case).

Example:

```
Before:   SAMPLE LINE          <--Pointer
          ANOTHER LINE
          YET ANOTHER
Command:  F YE
After:    SAMPLE LINE
          ANOTHER LINE
          YET ANOTHER          <--Pointer
```

FLAG SCRIPT [COL=n]
FLA

This form of the FLAG command is particularly useful when using the Editor to make modifications to a file that is to be used by the MUSIC/SCRIPT program. (The MUSIC/SCRIPT program is described in Chapter 5.) This command will cause all modified lines to be date stamped in columns 73 through 77, with the current date in the form YYDDD. deletion text to be added to the front of all deleted lines when flag mode is in effect (see the general form of the FLAG command).

This command automatically issues the following Editor commands:

```
TOP
TEXT SCRIPT
ZONE 72
WINDOW 1,72
```

The specification of the COL=n parameter can be used to put the date stamp starting in column number *n* rather than 73. The generated ZONE and VERIFY commands would then be changed accordingly.

Pointer: The pointer is moved to the top of the file.

FORMAT

This macro is used to format all the text in a file while remaining in the edit session. The SCRIPT program is used to perform the formatting with default SCRIPT control words incorporated to fit the text in 72 columns. If you wish, you can include your own control words to override the default settings.

The original text (unformatted) is stored temporarily in a holding file at the time you issue the FORMAT macro. You can restore the file to its original form by using the UNFORMAT macro.

This macro is not recommended for large files as the process is time consuming. It is ideal for formatting memos when you are using the editor in the MAIL program.

HELP [command-name] [list-of-topics]
HE

This command is used to obtain information about a particular Editor command, or about the Editor in general.

If no parameter is specified, the command gives general information about the Editor, such as a one-line description of common commands, Editor concepts, etc.

If *name* is specified, where *name* is the name or abbreviation of an Editor command, information about the command is displayed.

To access MUSIC's general help facility from the Editor, use a slash (/) preceding the HELP command. Enter "/HELP" or "/HELP topicname" from the command area of the Editor. (Without the "/" you will receive help on the Editor and not MUSIC's general help facility.)

Examples:

`HELP`	obtains general information about the editor.
`HELP MOVE`	explains the use of the MOVE command.
`HELP TOPICS`	gives a list of all available topic names
`/HELP COPY`	places you in MUSIC's general help facility and gives information about MUSIC's COPY command (not the Editor's COPY command).

HUNT [string]
H

The HUNT command combines the effect of a TOP command followed by a FIND. It searches the entire file for the first line beginning with *string.* If in VERIFY status, the found line is displayed. If there is no line beginning with *string,* the message *EOF is displayed.

If *string* is not specified, the same string as specified in the last used FIND, LOCATE, UFIND, ULOCATE, SEARCH, or HUNT command is used.

Pointer: Set to the found line, or beyond the end of the file if no line is found which begins with *string.* (Message *EOF is displayed in this case.)

Spacing: One blank should be present between the keyword or abbreviation and the first character of *string.* All other blanks are part of *string,* which extends to the last non-blank in the command line.

```
INPUT [END=xx]
INP
```

To change the mode of operation from edit to input. All lines typed after the INPUT command are placed, sequentially, after the current line. When a blank line is entered during the input mode, the Editor switches to edit mode with the pointer at the last line entered (not counting the blank line which is not saved). If the command is given at the very beginning of the Editor session, or immediately after a TOP command, the new lines are placed before the first line of the file.

The parameter END=xx, if used, causes the Editor to recognize *xx* as the input mode terminator, rather than a blank line. This option cannot be used in 3270 full-screen mode. *xx* may be one or two characters long, and any nonblank characters may be used. Input mode is terminated by entering a line with *xx* starting in column 1, followed by blanks. This *xx* line is not saved. The *END=* parameter is useful when it is required to enter blank lines while in input mode.

See also POWERINP.

```
INSERT  [string]
I
```

INSERT places a new line, containing *string,* after the current line. Note: If the command is given immediately after TOP, the new line is inserted before the first line of the file. A blank line can be inserted in the file just by typing INSERT.

CAUTION: The use of tab characters with this command may not have the desired effect. For example, the sequence "I tx", where "t" is the tab character, will put "x" in column 8 if the first tab location was defined to be column 10.

Spacing: One blank must be present between the command name or abbreviation and *string.* Any other blanks are part of *string.*

Pointer: Set to the inserted line.

Example:

```
Before:   SAMPLE LINE          <--Pointer
          ANOTHER LINE
Command:  I YET ANOTHER
After:    SAMPLE LINE
```

```
YET ANOTHER        <--Pointer
ANOTHER LINE
```

```
JOIN [/string/]
JO
```

The JOIN command may be thought of as the opposite of the SPLIT command. It joins the next line to the end of the current line, and deletes the next line.

The second line is added after the last nonblank character in the "ZONE" part of the first line. A specified character string (or a single blank if a string is not specified) is placed between them. Only the "ZONE" parts of the lines participate (refer to the ZONE command). If the joined text is too long for one line, a warning message is displayed and the excess is left on the second line.

A single blank is used if */string/* is not specified.

Example:

Before:
```
line1
line2    <--- current line
line3
```
Command:
```
join /+++/
```
After:
```
line1
line2+++line3   <--- new current line
```

```
LAST
LA
```

This command positions the line pointer to the last line of the file. The last line is displayed if the Editor is in VERIFY status.

Pointer: Set to the last line of the file.

```
LIST  [filname[,m][,n]]
LI
```

This command is used to display the contents of a file, or part of a file, or the entire file being edited. LIST is often used in conjunction with the MERGE command.

Line numbers *m* through *n* of the file *filname* are listed. If the *m* or *n* options are omitted the entire file is listed. If the *n* option is omitted or is too big, then the listing stops at the end of the file. The LIST command without any parameters displays the entire file being edited. Line numbers will also be displayed if line numbering is in effect. Line numbering is requested by using the NUMBER command.

The file name may be /INPUT (the Input File) or /HOLD (the Holding File).

"(3)" may be specified instead of *filname.* This causes the UDS file on MUSIC unit number 3 to be listed. The file is rewound both before and after the listing operation.

```
LOCATE  [string]
L
```

LOCATE is used to find the first line after the current line which contains *string* anywhere in the line and, if in VERIFY mode, to display the line on the workstation. The search begins with the line following the current line and continues down the file. If *string* does not exist in the file between the line following the current line and the end of the file, the message *EOF is displayed. If the command is given after an end of file an automatic TOP is performed before the search begins. If the command is given immediately after *EOF (bottom of file) or a TOP command, the first line of the file will also be searched for *string.* The operation of LOCATE may be affected by the ZONE setting (refer to the description of the ZONE command).

If *string* is not specified, the same string as specified in the last used FIND, LOCATE, UFIND, ULOCATE, HUNT, or SEARCH command is used.

(The ULOCATE command is similar to the LOCATE command but searches upwards through the file.)

Pointer: Set to the line in which the *string* is found, or beyond the end of the file, if it is not found.

Spacing: One blank should be present between the keyword or abbreviation and *string.* It is ignored if present. All other blanks are part of *string,* which extends as far as the rightmost non-blank in the command line.

Example:

```
Before:  SAMPLE LINE          <--Pointer
         ANOTHER LINE
         YET MORE
Command: L ER
After:   SAMPLE LINE
         ANOTHER LINE         <--Pointer
         YET MORE
```

```
MARK  [?]
MA
```

The MARK command is used to identify the first and last lines of a group of lines. The group may then be operated on by the *dot* forms of the command =, MOVE, COPY, DELETE, CHANGE, CHANGEL, REPEAT, TOUC, TOLC, COPYCOL, SEQ, SORT, and STORE.

Refer to the topic "Marking a Group of Lines" in the section "Advanced Features".

```
MERGE  filnam[,n][,m]
ME
```

This command brings a copy of line numbers *n* through *m* of the file *filnam* into the file being edited. If the *n* and *m* options are omitted the entire file is copied. If the *m* option is omitted or is too big, the file is copied until the end of the file.

The new lines are inserted after the current line, and the last line inserted becomes the new current line. If the preceding command was a TOP, or the MERGE command is given at the very beginning of the Editor session, the lines are placed at the beginning of the file.

The records merged from the Save Library are truncated or filled out with trailing blanks, if necessary, to match the record length of the file being edited.

The file name may be /INPUT (the Input File) or /HOLD (the Holding File).

"(3)" may be specified instead of *filnam.* This causes the UDS file on MUSIC unit number 3 to be used as input for the merge. The file is rewound both before and after the merge. Examples: MERGE (3), MERGE (3),15,50.

The LIST command can be used to list this external file to verify its contents before the merge operation is done.

```
MOVE i,j,k
MO
```

This form of the MOVE command allows moving lines by reference to the line numbers of the lines. The block of lines from line *j* to line *k* inclusive is moved to after line number *i.* (Line numbers may be determined by using the "=" command.) The original lines *j* to *k* are deleted. If *i* is specified as 0, the lines are moved to the beginning of the file. If *j=k,* only one line is moved.

Line numbers may also be specified in any of the following forms: *n, *+n, *-n, LAST, LAST-n,* where *n* is a number, * means the current line, and LAST means the last line of the file. Example: MOVE LAST,*,*+3

For the command "MOVE.", used to move a marked group of lines, refer to the topic "Marking a Group of Lines" in the section "Advanced Features".

Pointer: Set to the last line of the inserted section.

Example:

```
Before:   A1
          A2
          A3
          A4
          A5
Command:  MOVE 4,1,2
After:    A3
          A4
```

```
A1
A2
A5
```

```
NAME [name]
NA   [0   ]
```

This command is used to display or change the file name associated with the edit session. *name* is the name you wish to use for this file.

If no parameter is used, the name of the file being edited is displayed. For a UDS file edit, the volume name is also given.

If 0 (zero) is used as the parameter, the file name is made undefined. Then if a SAVE or FILE command is done, the user will be prompted to enter the file name to be used.

The file name associated with the edit session may be defined or changed by using the new name as a parameter on the NAME command (Save Library Files only). A subsequent SAVE command, for example, will use the new name.

MUSIC also accepts FNAME, with abbreviation FN, to mean the same as the NAME command.

Spacing: The file name, if used, must be separated from the command by at least one blank.

```
NEXT  [n]
N
```

This command moves the pointer *n* lines down the file from the current line. If *n* is omitted, 1 is assumed. The new current line is displayed, unless a BRIEF command has been issued previously.

For compatibility with other systems, MUSIC also accepts DOWN (or DN) to mean the same as the NEXT command.

Pointer: Set *n* lines beyond the current line.

Example:

```
Before:  SAMPLE LINE          <--Pointer
         ANOTHER LINE
         YET MORE
Command: N2
After:   SAMPLE LINE
         ANOTHER LINE
         YET MORE             <--Pointer
```

```
NOFLAG
NOFL
```

This command terminates the effect of a previous FLAG command, except that the ZONE and WINDOW settings still apply. However, the flag options are remembered by the Editor, and the original flagging can later be continued by issuing a FLAG command without any parameters.

```
NONUMBER
NONUM
```

This command is used to suppress the line numbering caused by the NUMBER command. At the start of the Editor session, the line numbering is off.

```
NOSHOW
NOSH
```

The NOSHOW command removes the text displayed at the bottom of the screen by a previous "SHOW file-name" command, and enlarges the FS screen back to its original size.

```
NULLS  [FLIP]
       [F   ]
```

This command applies only to the 3270 full screen mode. It causes null characters to be displayed at the end of each field on the screen. An alternate name for NULLS is NOFILL.

The FLIP option reverses the current command; i.e., NULLS FLIP causes blanks to be displayed at the end of each field if null characters were previously displayed, or causes null characters to be displayed if blanks were previously displayed.

At the start of the Editor session, null characters are displayed. The opposite of this command is NONULLS.

```
NUMBER [FLIP]
NUM    [F   ]
```

This command is used to cause line numbers to appear whenever lines of the file are displayed. The FLIP option reverses the current setting for the command; i.e., NUMBER FLIP turns numbering on if it was off, and off if it was on.

At the start of the Editor session, the line numbering is off.

```
POWERINP
POW
```

The POWERINP command (actually a macro) is similar to the INPUT command, except that you can type continuously without having to press the NEWLINE key or TAB key at the end of each line of input. Power Input Mode is similar to Input Mode, except for these differences:

1. The cursor skips automatically to the beginning of the next line when the cursor reaches the end of a line. This allows you to type continuously without having to watch where the cursor is on the screen.

2. When you press ENTER or a function key while in Power Input Mode, the lines you have just typed are reformatted by the editor. This is called "word wrap". Words that were split at the end of a line are put back together, and text is formatted within the current WINDOW columns. A blank line causes a formatting break (paragraph separator). The formatting is similar to what the WW prefix command would do. See the WW command in the topic "Prefix Area" earlier in this chapter.

3. Pressing ENTER after typing one or more lines of input does not end Power Input. The Enter key causes a formatting break, and Power Input continues. This is useful for entering single lines (like Script control words) that should not be formatted with the rest of the text. To end Power Input, press ENTER twice, or F12 once.

Power Input can also be started by the POW prefix area command. Input starts after the line identified by POW.

If POWERINP is used while not in 3270 full-screen (FS) mode, normal Input Mode is used.

Note: To use Power Input with NET3270, document mode (the Alt-E toggle) must be OFF. NET3270 has its own word-wrap feature (Alt-W while in document mode), which makes the editor's Power Input unnecessary.

Internals:

Power Input is started by the macro POWERINP, which, among other things, issues the commands DEFINE INPUT $FINP and INPUT. The text typed during Power Input is processed by the macro $FINP. Automatic cursor skip is obtained by the command AUTOSKIP INPUT. For more information, see the DEFINE command and the comments in file $FINP.MAC.

```
PREFIX ON
PRE    OFF
       FLip
       CLear
```

The prefix area, turned on by the command PREFIX ON, is a 4-character modifiable field at the left of each displayed line of the file. It contains ====, or the last 4 digits of the line number if NUM ON is in effect. You can enter various special commands in the prefix area, to do editing operations such as inserting or deleting lines, moving and copying lines, etc.

For more information type "help prefix" in the Editor's command area.

```
PURGE name
PUR
```

This command is used to permanently remove (delete) a file from the Save Library, as in the MUSIC command /PURGE.

To delete the Input File, specify file name /INPUT. To delete the Holding File, specify file name /HOLD.

Spacing: The file name must be separated from the command by at least one blank.

```
QUIT
Q
```

The QUIT command terminates the Editor session, without performing any save operation.

If you have made changes to the file but have not issued a FILE command to make the changes permanent, you will be prompted for permission to end the edit session. Enter YES or Y to end the session, or NO or N to cancel the QUIT operation and continue editing.

The END, QUIT, and QQUIT commands terminate the edit session without writing the editor's copy of the file to disk. An optional job return code *n* (a number 0 or higher) can be specified on each of these commands. If the return code parameter is omitted, the code specified on the last SETRC command is used, or 0 if no SETRC was done. Please refer to the SETRC command for additional notes.

```
RENAME oldname newname
REN
```

The RENAME command changes the name of an existing file in the Save Library. It performs the same function as the MUSIC command RENAME.

Most attributes of the file, such as public/private, tag, and date last referenced, are unchanged by the command. The only exception is the date last written, which is set to the current date.

```
REPLACE  [string]
R
```

This command replaces the current line with *string.* You can replace the current line with a completely blank one by typing REPLACE without specifying a string.

Spacing: One blank must appear between the keyword and the first character of *string.* All succeeding characters are part of *string.*

Example:
Before: **SAMPLE LINE**
Command: **R NEW STUFF**
After: **NEW STUFF**

```
SAVE  [name]
SA    [(u) ]
SV
```

This command causes the current temporary copy of the file to replace the specified file, without terminating the Editor. This is similar to the FILE command except that the edit session will not be terminated. The line pointer is not changed. Therefore, the user may continue to issue additional edit commands on the same file.

Parameters:

name — If a file name is not specified on the command, the name on the original EDIT command or on the last NAME command will be used, or, in the case of editing a User Data Set (UDS) file, the UDS file being edited is replaced.

u — A unit number in parentheses may be specified in place of a file name, causing output to be done to that unit instead of to a file. For example, SAVE (3). The unit number must be 3, 7 or 10. The unit is not rewound before output is begun.

For a description of the various other parameters which can be used on the SAVE command, refer to the FILE command, which takes the same parameters. These include starting and ending column numbers and

file attributes (PRIV, PUBL, etc.)

Examples:

	`SAVE NEWFIL` It writes the updated version of file to a file named NEWFIL. The original file is not updated.
`SAVE 72,(3)`	It writes columns 1 to 72 of each line in the updated version of file to the UDS defined in unit 3. The original file is not updated.

See also FILE and STORE.

```
SEARCH  [string]
S
```

The SEARCH command combines the effect of a TOP command followed by a LOCATE. It searches the entire file for the first line containing *string.* If in VERIFY status, the found line is displayed on the work-station. If *string* does not exist in the file, the message *EOF is displayed.

If *string* is not specified, the same string as specified in the last used FIND, LOCATE, UFIND, ULOCATE, HUNT, or SEARCH command is used.

Pointer: Set to the line containing *string,* or past the end of the file.

Spacing: One blank should be present between the keyword or abbreviation and the first character of *string.* All other blanks are part of *string,* which extends to the last non-blank in the command line.

```
SHOW PFn
SH   PFPm
     PFAn
     PF
     X
     filename n1 n2
```

The SHOW command displays the current definition of Program Function (PF) keys or the X command, or displays the contents of a specified file at the bottom of the screen in FS mode.

SHOW PFPm or SHOW PFAm *(m* is a number normally from 1 to 12), or SHOW PFn *(n* is a number from 1 to 24) displays the command string currently defined for that function key. SHOW PF gives the definitions of all function keys 1 to 24. SHOW X displays the definition of the X command.

"SHOW filename" displays the first 12 lines of a file at the bottom of the screen. "SHOW filename n1 n2" *(n1* is the starting line number, *n2* is the ending line number) displays up to 12 lines of a file at the bottom of the screen. The default for *n1* is 1 and the default for *n2* is *n1+11.* The text remains on the screen until removed by a NOSHOW command or replaced by another SHOW command. The number of screen lines available for editing (the *n* in the command "FS n") is reduced by the number of lines displayed by SHOW. This form of the SHOW command could be used to permanently display function key definitions or other

helpful information.

```
SPELL
```

This macro invokes the SPELL program to spell the current file. After you are finished with the SPELL program, you are returned to the original edit session.

Note: Invoking SPELL from the Editor instead of *Go mode means that no exception list is kept for your dictionary.

```
SPLIT  /string/[Cn][ICm]
SP
```

This command causes the current line to be searched for *string* and if it is found, the current line is broken into two lines, the first consisting of columns 1 through the last column before *string* and the second extending from the beginning of *string* to the end of the line. The C*n* parameter specifies the column number in which the search for *string* is to begin. The second line will be adjusted to start in column 1 unless IC*m* is used to specify an indentation column number. The first line becomes the new current line.

In 3270 full-screen mode only, the screen cursor may be used to indicate the point at which the line is to be split. In this case, the SPLIT command is used without any parameters. The character at which the cursor is placed becomes the first character of the second line. The cursor is left at the end of the first line. F2 has the default definition of SPLIT.

Pointer: Set to the first of the two lines.

Spacing: One blank may optionally be present between the command name or abbreviation and the first slash (/). The second string delimiter (/) may be omitted if Cn and ICm are not used and the string does not end in a blank.

Example:

```
Before:   SAMPLE LINE        <--Pointer
Command:  SP /LI/
After:    SAMPLE             <--Pointer
          LINE
```

```
SORT * options
     . options
     * filename2 options
     . filename2 options
     filename1 filename2 options
```

The SORT editor macro uses the MUSIC SORT command to sort the entire file you are editing ("sort *") or part of the file that you have MARKed ("sort ."). The sorted records replace the original records, unless you specify a target file name as the second parameter.

Options can be specified, if needed. They are as on the MUSIC SORT command.

Possible options are:

m-n	Starting (m) and ending (n) column of the sort key.
-d	Sort in descending order.
-r	Replace the target file (filename2) without prompting.
-deldups	Delete output records which have the same key field as the previous output record (i.e. delete duplicates).

Default is to sort in ascending order, using the entire record (or the first 256 characters if the records are longer than 256) as the sort key.

When the first parameter is * or . , the records to be sorted are stored to file @SORT.TMP, which is then used as filename1 on a MUSIC SORT command. The resulting sorted file is merged back into the current file, replacing the original lines. The UNSORT macro uses the contents of file @SORT.TMP to undo the sort.

STORE [n] [m] filename [attributes] [APPEND]
STO

The STORE command writes the group of marked lines to an external file. The edit continues. The MARK command (or function key operation) must be used to mark one or more lines before the STORE command can be used.

If the optional APPEND keyword (abbreviation: APP or A) is used following the file name, the marked lines are added to the end of the file, if the file already exists. A new file is created if one does not exist.

Note: When lines are appended to an existing file, unused space is not released at the end of the file. This is intentional, in order to prevent an excessive number of extents being created when several stores are done to the same file. Unused space can be freed later by editing and filing (FILE command) the file, or by running the following job:

```
/FILE 1 NAME(filename) OLD
/LOAD IEFBR
```

The following parameters are available:

n	(optional) starting column number.
m	(optional) ending column number.
filename	the name of the target file.
attributes	(optional) file attributes: PUBL, PRIV, SHR, COM, XO, CNT. When appending to an existing file, the *attributes* are ignored.

Example:

`STORE MYFILE` Writes marked lines to file MYFILE. (If the file already exists

	you are asked if you wish to replace it.)
`STO 11 20 MYFILE PUBL`	Writes columns 11-20 of marked lines to file MYFILE, and creates the file as public.

```
TAG  [string]
```

The TAG command displays or sets the tag string associated with the file being edited. The tag is up to 64 characters long, and is stored with the file when a SAVE, FILE, or EXEC command is used. Often the tag is used as a one-line description of the file's contents.

If *string* is specified on the TAG command, it replaces the current tag. To remove the tag, use the command TAG ''. If no parameter is present, the current tag (if any) is displayed.

```
TEXT  [LC      ]
TEX   [UC      ]
      [SCRIPT  ]
      [NOSCRIPT]
```

The TEXT command is used to specify whether lower case input is to be translated to upper case or not. The command remains in effect until the end of the edit session, or until another TEXT command is used. When TEXT is entered without a parameter, the current setting is displayed.

Parameters:

UC	specifies that lower case input is to be translated into upper case automatically. This is the default when the EDIT command starts the editor with a file that contains all upper case characters.
LC	specifies that lower case input is to be left as is. This is the default for new files and when starting the Editor with files that already contain lower case characters.
SCRIPT	indicates that lower case input is to be left as is, and that input tab characters are not to be translated to the appropriate number of blanks. Tab characters in the file are displayed as "#" characters. TEXT SCRIPT is the default if /TEDIT was used to start the edit.
NOSCRIPT	removes the effect of a previous TEXT SCRIPT command, except that the UC/LC setting is not changed.

See also the CASE command.

```
TOLC  [n]
```

This command changes characters to lower case in a line or group of lines, starting with the current line. The last line converted becomes the new current line. Only alphabetic letters (A-Z) in the *zone* portion (as defined by the ZONE command) of each line are changed.

n is the number of lines (1 or more). If *n* is omitted, only the current line is changed. The dot form of this command (TOLC.) can be used for a marked group of lines.

```
TOP
T
```

The TOP command moves the pointer to the first line of the file.

Pointer: Set to the first line of the file. If the command INSERT, MERGE or INPUT is given immediately after the TOP command, the new lines are placed before the first line of the file. Also, a FIND, LOCATE or SEARCH command used immediately after a TOP command will include the first line of the file in the search. At the start of the Editor session, the pointer is at the top of the file.

```
TOUC  [n]
```

This command changes characters to upper case in a line or group of lines, starting with the current line. The last line converted becomes the new current line. Only alphabetic letters (a-z) in the *zone* portion (as defined by the ZONE command) of each line are changed.

n is the number of lines (1 or more). If *n* is omitted, only the current line is changed. The dot form of this command (TOUC.) can be used for a marked group of lines.

```
UFIND  [string]
UF
```

The UFIND command searches the file, starting with the line preceding the current line, for a line beginning with *string*. The search works towards the beginning (top) of the file until the first match, or until the pointer is at the top of the file. In the latter case, the message `TEXT NOT FOUND` will result which means that *string* cannot be located between the line preceding the current line and the first line of the file. If VERIFY status is in effect, the line found is displayed on the workstation.

If *string* is not specified, the same string as specified in the last used FIND, LOCATE, UFIND, ULOCATE, HUNT, or SEARCH command is used.

The command can also be specified as UPFIND or UPF.

Spacing: One blank should be present between the keyword or abbreviation and *string.* It is ignored if present. All other blanks are considered to be part of *string*, which extends to and includes the rightmost non-blank in the command line.

Pointer: Set to the found line, or to the top of the file if no line is found which begins with *string.*

Example:

```
Before:   SAMPLE LINE
          ANOTHER LINE
          YET ANOTHER          <--Pointer
Command:  UF SA
After:    SAMPLE LINE          <--Pointer
          ANOTHER LINE
          YET ANOTHER
```

```
ULOCATE  [string]
UL
```

ULOCATE is used to find the first line before the current line which contains *string* anywhere in the line and, if in VERIFY mode, to display the line on the workstation. The search begins with the line preceding the current line and works towards the beginning of the file. If *string* does not exist in the file between the line preceding the current line and the first line of the file, the message `TEXT NOT FOUND` is displayed. The operation of ULOCATE may be affected by the ZONE setting (refer to the description of the ZONE command).

If *string* is not specified, the same string as specified in the last used FIND, LOCATE, UFIND, ULOCATE, HUNT, or SEARCH command is used.

The command can also be specified as UPLOCATE or UPL.

Pointer: Set to the line in which the *string* is found, or to the top of the file, if it is not found.

Spacing: One blank should be present between the keyword or abbreviation and *string.* It is ignored if present. All other blanks are part of *string,* which extends as far as the rightmost non-blank in the command line.

Example:

```
Before:   SAMPLE LINE
          ANOTHER LINE
          YET MORE             <--Pointer
Command:  UL ER
After:    SAMPLE LINE
          ANOTHER LINE         <--Pointer
          YET MORE
```

```
UNFORMAT
```

This macro is used to undo the effect of the FORMAT macro. It restores the formatted file back to its unformatted form that was previous to using the FORMAT macro.

```
UNMARK
UNMA
```

This command unmarks any lines that were previously marked by the MARK command. Refer to the topic "Marking a Group of Lines" in the section "Advanced Features".

```
UNSORT *
       .
```

The UNSORT editor macro can be used to undo the effects of the immediately preceding SORT macro.

You should specify the same parameter (* or .) as you used on the sort command.

UNSORT uses the contents of file @SORT.TMP, which is created by the SORT command.

```
UP  [n]
U
```

This command moves the pointer *n* lines toward the beginning of the file. If in VERIFY status, the line at which the pointer is positioned after execution of the command is displayed on the workstation. If *n* is omitted, 1 is assumed.

If the number of lines specified is large enough to cause the line pointer to be moved beyond the first line of the file, the UP command has the same effect as a TOP command. For example, an immediately following INSERT command would insert a line before the first line.

Spacing: The number may appear anywhere in the line after the keyword or abbreviation.

Example:

```
Before:   SAMPLE LINE
          ANOTHER LINE        <--Pointer
Command:  U
After:    SAMPLE LINE         <--Pointer
          ANOTHER LINE
```

```
UPPAGE
UPP
```

This command is used only in 3270 full-screen mode. It shifts the screen display one screen (page) towards the beginning of the file. Refer to the section on full-screen mode for more details. UPPAGE corresponds to F7 by default.

```
UPWINDOW  [n]
UPW
```

This command applies only to 3270 full-screen mode. It shifts the screen display (window) towards the beginning of the file. The parameter *n* is the number of lines by which the window is to be shifted. If *n* is omitted, 6 lines is assumed.

```
WINDOW  [n      ]
W       [m n    ]
        [OFF    ]
        [RIGHT k]
        [LEFT k ]
        [FLIP   ]
```

This command specifies the starting and ending columns to be displayed by the Editor whenever a line of the file is displayed. This is particularly useful when long records are being edited. The listing of an external file, by the LIST command, is not affected.

m is the starting column number. *n* is the ending column number, and may be specified by an asterisk (*), which means the record length of the file being edited. If only one number is specified, the number is assumed to be *n,* and *m* is assumed to have a value of 1.

If OFF is specified instead of column numbers, the Editor resets the *window* to be the whole line, as it is when the Editor starts.

The option RIGHT or LEFT may be used on the WINDOW command to shift the window towards the right or left hand side of the records being edited. The width of the window is not changed. The parameter *k* is the number of columns by which the window is to be shifted. If *k* is omitted, the window is shifted by the

maximum amount. RIGHT and LEFT may be abbreviated R and L.

The FLIP option causes the window to be shifted to the extreme left or right, whichever is further from the existing window setting. FLIP can be abbreviated FL or F.

When a WINDOW command is used in 3270 full-screen mode, the zone is automatically set to be from column 1 to the end of the window (refer to the ZONE command).

If no parameters are specified, the current window setting is displayed.

The window setting can also be changed by specifying a number *n* on the VERIFY command. In this case, the window setting will become columns 1 to *n.*

When the WINDOW command is used, the portion of a line displayed by the PRINT command defaults to the window setting. The second parameter on the PRINT command may still be used to display more or fewer characters per line starting from column 1.

Spacing: The parameters must be separated from each other by commas or blanks. If OFF, RIGHT or LEFT is specified, it must be separated from the command by at least one blank.

```
X  [n]
```

The X command causes a predefined set of Editor commands to be executed. The commands are defined by the DEFINE command. For example, if "DEFINE X LOCATE ABC;INSERT **" is used, then entering the command X causes the commands "LOCATE ABC" and "INSERT **" to be performed. An X command string may not contain an X command. The X command may be used on all types of workstations.

If parameter *n* is used (a number from 1 to 24), the command string defined for program function key *n* (PFn) is executed. In this way, function key operations can be defined and used on any type of workstation.

```
XL string
XUL string
XF string
XUF string
```

The XL macro locates a character string that you specify. It issues a LOCATE command for the specified character string. However, if the search is unsuccessful, XL does not alter the current line pointer. This has the effect of not altering the current screen when the specified string is not found. If the specified character string is not located, then the current line number is not changed, and a message is displayed indicating that the search failed.

All of the considerations applicable to the LOCATE command apply to the XL macro.

XUL This is the same as the XL macro except that ULOCATE command is used.

XF This is the same as the XL macro except that FIND command is used.

XUF This is the same as the XF macro except that UFIND command is used.

```
ZONE  [n]
Z     [FIXED]
```

The ZONE command is used to display the current zone setting, or to change the zone setting. The zone setting defines the ending column for CHANGE and ADD commands, and for commands which may scan for character strings (such as DELETE, FIND, HUNT, UFIND, ULOCATE, LOCATE, SCAN, SPLIT, and SEARCH), including logical expressions. Columns following the ZONE specification are not changed by CHANGE or ADD commands and are not examined when looking for a character string. ZONE does not affect BLANK, OVERLAY and PRINT commands.

When the SPLIT command is used to break a line into two parts, the second part extends only as far as the zone setting. Characters following the zone remain on the first line.

n must be a number from 1 to the record length. * is equivalent to specifying the record length, which is the value assumed when the Editor begins execution. If no parameter is specified, the current zone specification is displayed.

The option FIXED (abbr. F) keeps the zone setting unchanged by later WINDOW, LEFT, RIGHT or FS commands (e.g. ZONE * F). The command ZONE F sets the "fixed zone" option without changing the current zone setting.

```
=
=n
=n/string1/string2/[options]
=.
```

This command, without any parameters, causes the Editor to type the line number of the current line. Lines of the file are numbered consecutively starting at 1. When you see the message *EOF (pointer beyond the end of the file), the number of lines in the file, plus 1, will be displayed.

If a line number *n* is specified with the command, the line pointer moves to the line numbered *n* and the line becomes the new current line. The line is also displayed. For example, =30 makes the line numbered 30 the new current line.

If "=." is specified the Editor moves to the first line in a marked group.

A change request, with or without options, as in the CHANGE command, can be specified following the line number. Refer to the description of the CHANGE command. For example, =30/ABC/XYZ/ causes the line pointer to move to line 30 and the string ABC is changed to XYZ. The changed line is displayed.

Chapter 4. Electronic Mail

Overview of the Mail Facility

The Mail program on MUSIC is used to send electronic messages to your colleagues. These messages can be sent internally to another person at your institution, or externally to another country. Full documentation on the Mail facility can be found in the *MUSIC/SP Mail and Conferencing Guide*, or by typing "MAN" and viewing the guide online. Also, help is available once the MAIL program is invoked by pressing F1. This chapter describes:

- How to Use Mail
- Sending Messages (exercise)
- Mail Directory
- Mail Profile
- External Mail
- Other Forms of Communication

How to Use the Mail Facility

The Mail Facility is invoked by choosing the topic "Mail" on the FSI main menu. The following diagram illustrates the main menu of Mail.

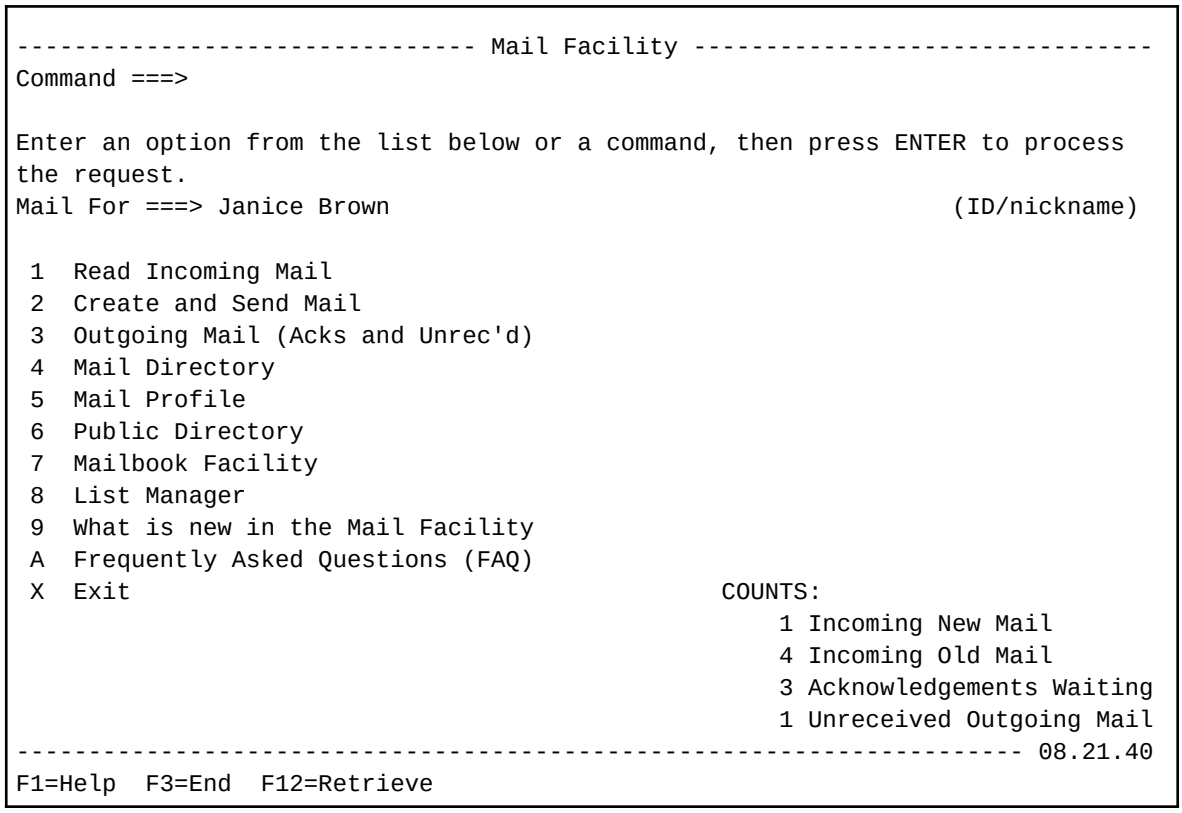


Figure 4.1 - Main Menu of Mail

Description of Main Menu of Mail

The following describes the fields on the screen in figure 4.1.

Command => This is the command area. Enter an option code from the list on the screen (1 - 9, A and X).

Mail For => Your name or userid appears here. The procedure for adding your name to your mail profile is described in the section "Mail Profile". It is important to add your name so it will be included with your return address for all your outgoing mail.

Selection Codes The following selection codes are available from the main menu. Each selection is discussed in more detail later in this chapter.

1 - Read Incoming Mail:
view list of all incoming mail items.

2 - Create and Send Mail:
for typing and sending your messages.

3 - Outgoing Mail (acks and unrec'd):
view list of your outgoing mail items. You can check to see if someone has read your message.

4 - Mail Directory:
You can assign names to an individual or a group of userids to make it easier to remember mail addresses.

5 - Mail Profile:
This facility allows you to tailor the mail facility to your needs. For example, you have a choice of how the incoming and outgoing mail is sorted in the list.

6 - Public Directory:
This facility allows you to view a list of public nicknames maintained at your site.

7 - Mailbook Facility
This facility allows you to view an existing mailbook file. You must include a file name with the selection number (7 filename).

8 - List Manager
This selection invokes the LM (List Manager) program for maintaining BITNET or Internet discussion lists.

9 - What is new in the Mail Facility
This selection describes the new features of the Mail facility since the last version.

A - Frequently Asked Questions (FAQ)
This selection provides answers to frequently asked questions about electronic mail.

X - Exit:
Type "x" in the command area or press F3 to exit mail.

Counts — The number of incoming and outgoing mail items is displayed in the bottom right-hand corner of the screen.

Incoming New Mail:
is the number of incoming mail items that you have not read (no action has been taken).

Incoming Old Mail:
is the number of incoming mail items that you have looked at but have not deleted.

Acknowledgements Waiting:
is the number of your outgoing mail items that have been received (the receiver has read the mail you sent or taken some other action).

Unreceived Outgoing Mail:
is a combination of the number of your outgoing mail items that have NOT been received (no action has been taken by the receiver on the mail you sent) and suspended mail items.

Function Keys for Mail

Throughout this facility, the following keys are valid:

F1=Help — provides help for the current screen display.

F3=End — returns to the previous function. If you are viewing the main menu screen, then you exit from the Mail program.

F12=Retrieve — displays the previous command in the command area. Up to 5 commands can be recalled.

PA1=Exit Mail cancels the current request and exits the Mail program.

Entering Commands

Mail commands and MUSIC commands are allowed in the command area. However, most tasks for Mail are done with function keys and select codes. You can type "HELP COMMANDS" if you wish information on Mail commands. MUSIC commands can be entered with a "/" in front to distinguish them from Mail commands. Unless you need to leave the Mail environment for another task, MUSIC commands are not necessary for Mail.

The "=" command is very useful to skip from one screen to another. Enter "=n" where *n* is the selection code from the main menu. For example, from the VIEW INCOMING MAIL screen, enter "=2" to bypass the main menu and go directly to the CREATE AND SEND MAIL screen.

Incoming Mail

Selection 1 on the main menu of mail is for reading incoming mail. Figure 4.2 below shows a sample of the VIEW INCOMING MAIL screen.

```
----------------------------- View Incoming Mail ---------------------- Rec 1/5
Command ===>

Mail For ==> Janice Brown                                        (ID/nickname)

Select Codes => A-Answer B-ReplyFrm C-Copy D-Delete F-Forward G-Get L-Listd
                P-Print R-Refuse T-Transfer U-Undel V-View X-Xlogdel Z-Zoom

   From               Date Sent Type  Subject
_  CCDO@MUSICA.MCGI   23AUG90   Old   final call
_  MUG@MARIST         23AUG90   Old   Too much mail.
_  CCRMMUS@MCGILLM    24AUG90   Old   change to user guide
_  CCMS000@MCGILLM    25AUG90   Old   New Pcws ... Again
_  CCMS000@MCGILLM    27AUG90   New   New Pcws




------------------------------------------------------------------------ 08.24.31
F1=Help       F3=End        F7=Prev       F8=Next       F9=Loc
F10=Refresh   F11=Outgoing  F12=Retrieve  PA1=Exit Mail
```

Figure 4.2 - Incoming Mail

The VIEW INCOMING MAIL screen shows a list of all your incoming mail items. By default the list is displayed in ascending order by date received.

The "Command ===>" field at the top of the screen is for entering Mail commands. Most of the time this screen is manipulated through the use of select codes typed beside mail items and the use of function keys. Type "HELP COMMANDS" is you wish to learn about Mail commands.

Incoming Mail List

Beside each mail item is a space for you to type in a select code (as described below) to take some action on each piece of mail. The first column under the heading "From" is the userid or nickname of the sender(s). The "Date Sent" column lists the date the mail was sent to you. One of the following appears in the "Type" column:

New	- mail item has not been read yet
Old	- some kind of action has been taken on this mail item
Reply	- answer to one of your outgoing mail items

The last column of the incoming mail list displays 40 characters of the subject.

Select Codes

One of the following codes can be entered beside each incoming mail item. Several codes can be typed in at the same time. The select code used most often is V (View) for reading a new item.

A - Answer — indicates that you wish to answer that particular mail item. You will go to the ANSWER MAIL screen where the "Reply-To" mail address and subject of the mail is displayed.

Note: The answer function of Mail is also available when you are viewing a mail item. The F2 (answer) key is used for this purpose.

B - ReplyFrm — indicates that you wish to answer using the "From" address of that particular mail item.

Note: Sometimes a mail item has a "Reply-To" address and a "From" address. For example, with discussion lists, you can answer everyone on the list (default) or answer just to the individual. Select code A answers to the "Reply-To" address (discussion list). Select code B sends the answer to the "From" address (individual).

C - Copy — Type "C" beside the mail that you want to copy and press ENTER to go to the COPY MAIL screen. Help is available when the COPY screen appears.

D - Delete — Type "D" beside the mail that you want deleted from your list of incoming mail. These entries disappear when F10 (Refresh) is pressed, or when you exit Mail.

Note: The delete function of Mail is also available when you are viewing a mail item. The F6 (delete) key is used for this purpose.

F - Forward — Type "F" beside the mail item to forward mail to another person. Help is available when the FORWARD screen appears.

G - Get — Type "G" beside the mail item(s) and press ENTER to get (receive) a mail item that was sent to you via SENDFILE. You can tell the mail item was sent via sendfile when the mail subject starts with "Sendfile:". For more information type "HELP SENDFILE" in the command area.

L - Listd — Shows the distribution list for mail if sent to more than one person.

P - Print — Presents a screen that allows you to send the incoming mail to a printer. Help is available when the PRINT screen appears.

R - Refuse — Allows you to refuse the mail item. The sender is notified that you refused to accept the

mail. You cannot view (or perform any other action) with the mail before refusing it. These entries disappear when F10 (Refresh) is pressed or the next time you view incoming mail. If you wish to automatically refuse mail from someone, see the Mail Profile Facility (select option 5 from the main menu).

T - Transfer — Type "T" beside the mail item to transfer mail to another person. Help is available when the TRANSFER screen appears.

U - Undel — Type "U" beside a mail item to retrieve it after you have accidentally deleted it. This function is only available for mail items that have been deleted, refused, or logged and deleted.

V - View — The "V" select code allows you to read the mail item. A tailored version of the VIEW program is used for this function of Mail.

X - Xlogdel — Copies the mail item to a file (specified in your Mail Profile) and then deletes the entry from your incoming mail list. This select code is similar to using "C" (Copy) and then "D" (Delete).

Z - Zoom — Type "Z" beside the mail to see what action has been taken on the mail item by you and the sender. The VIEW CONTROL INFORMATION screen is displayed. For example, you can tell if you have replied, copied, or printed the mail item. Help is available when the VIEW CONTROL INFORMATION screen appears.
Select code "L" (LISTDIST) should be used if you want to see who on the list received the mail.

Example of Viewing Mail

After selecting "1" from the main menu, the list of incoming mail items is presented. In figure 4.3 three mail items have been chosen for viewing.

```
----------------------------- View Incoming Mail ---------------------- Rec 1/5
Command ===>

Mail For ==> Janice Brown                                        (ID/nickname)

Select Codes => A-Answer B-ReplyFrm C-Copy D-Delete F-Forward G-Get L-Listd
                P-Print R-Refuse T-Transfer U-Undel V-View X-Xlogdel Z-Zoom

   From               Date Sent Type  Subject
v  CCDO@MUSICA.MCGI   23AUG90   Old   final call
v  MUG@MARIST         23AUG90   Old   Too much mail.
v  CCRMMUS@MCGILLM    24AUG90   Old   change to user guide
_  CCMS000@MCGILLM    25AUG90   Old   New Pcws ... Again
_  CCMS000@MCGILLM    27AUG90   Old   New Pcws




------------------------------------------------------------------------ 08.24.32
F1=Help       F3=End        F7=Prev       F8=Next       F9=Loc
F10=Refresh   F11=Outgoing  F12=Retrieve  PA1=Exit Mail
```

Figure 4.3 - Selecting Mail Items for Viewing

When the ENTER key is pressed, the first mail item is displayed for viewing. After each item, you can press F3 to go to the next.

Figure 4.4 below shows the third file from the list above.

```
 ------- VIEW MAIL FILE -------         L 80    C 1-80         Rec 1/16
 COMMAND ===>                                                 SCROLL ===> PAGE
 *** TOP OF FILE *** ----------------------------------------VC/TEXTLC/IGNORE
Message-Id:  <23AUG90.12875572.0021.MUSIC@MCGILLM>
Date:        Thu, 23 Aug 90 11:55:18 EDT
From:        Roy Miller <CCRMMUS@MCGILLM>
To:          Janice Brown <CCJB@MCGILLM>
Subject: change to user guide

I have a change to the users ref guide.
It is in the profile section.
Add the following text under AUTOPROG:

         The autoprog file can be a REXX file that can run
         several programs when you sign on.

 *** BOTTOM OF FILE ***

F1=Help   F2=Answer  F3=End      F4=Forward    F5=Center   F6=Delete/Undelete
F7=Up     F8=Down    F9=Locate   F10=Previous  F11=Next    F12=Retrieve
```

Figure 4.4 - Reading a Mail Item

In figure 4.4 the header information at the top of the screen includes the date and time the mail was sent, who it is from, who it was sent to, and the subject. The text of the mail item follows. At the bottom of the screen the following keys are listed:

F2 (Answer) allows you to answer the mail item immediately.
F4 (Forward) allows you to forward the mail.
F6 (Delete/Undelete) allows you to delete and undelete the mail and its entry from your incoming mail list.
F10 (Prev) allows you to view the previous mail item from the incoming Mail list.
F11 (Next) allows you to view the next mail item from the incoming Mail list.
F3 (End) returns to the VIEW INCOMING MAIL screen.

Outgoing Mail

After selecting "3" from the main menu, the list of outgoing mail items is presented. Figure 4.5 shows an example of the VIEW OUTGOING MAIL screen.

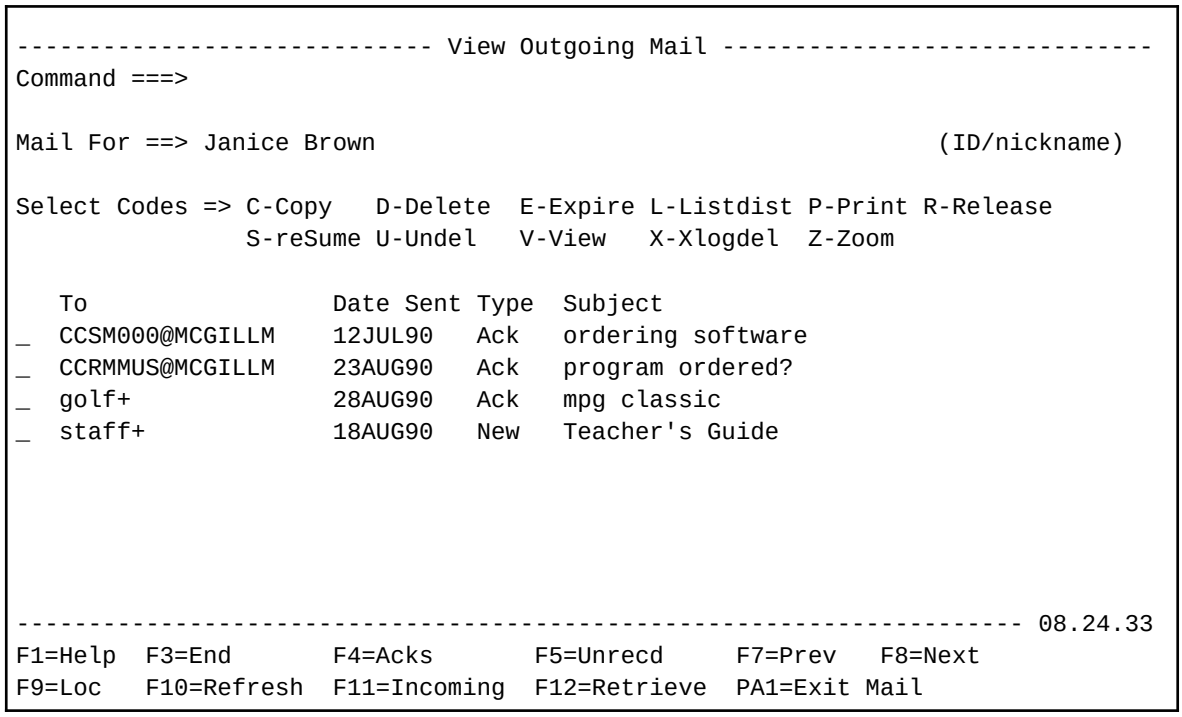


Figure 4.5 - Viewing Outgoing Mail

This screen displays a list of all outgoing mail including those that were read (acknowledged) and those that were not received. This list can be separated by using F4 (Acks) to show only acknowledgements or F5 (Unrecd) to show only unreceived.

The "Command ===>" field at the top of the screen is for entering Mail commands. Most of the time this screen is manipulated through the use of select codes typed beside mail items and the use of function keys.

Outgoing Mail List

Beside each mail item is a space for you to type in a select code (as described below) to take some action on each piece of mail. The first column under the heading "To" is the userid or nickname of the receiver(s). If a plus sign (+) appears at the end of the name, it indicates that this is a nickname for a distribution list (see the topic "Mail Directory" later for information about creating nicknames). The "Date Sent" column lists the dates the mail were sent by you. The "Type" in column 33 can be one of the following:

Ack	- has been acknowledged
New	- no one has received it yet
Suspd	- suspended mail item

The last column displays 40 characters of the subject of the mail.

Select Codes

The following codes can be entered beside the list of outgoing mail. Several codes can be typed in at the same time, one for each item. Many of the codes are the same as the ones for the VIEW INCOMING MAIL screen.

C - Copy Copies the mail item. Help is available when the COPY screen appears.

D - Delete	Deletes the mail item. These entries disappear when F10 (Refresh) is pressed, or when you exit the Mail program.
E - Expire	Presents the EXPIRE MAIL screen that allows you to change the expiry date and time of the mail item.
L - Listdist	Shows the distribution list for mail if it has been sent to more than one person.
P - Print	Displays the PRINT MAIL screen for sending the mail item to a printer.
R - Release	Presents the RELEASE MAIL screen that allows you to change the release date and time of the mail item.
S - reSume	Presents the RESUME MAIL screen that resumes the sending of a suspended mail item.
U - Undelete	Restores the last mail item that was deleted.
V - View	The "V" selection code allows you to read the mail.
X - Xlogdel	Copies the mail item to a file (specified in your Mail Profile) and then deletes the entry from your outgoing mail list. This select code is similar to using "C" (Copy) and then "D" (Delete).
Z - Zoom	Type "Z" beside the mail to see what action has been taken on the mail item by you and the receiver. The VIEW CONTROL INFORMATION screen is displayed. Select code "L" (LISTDIST) should be used if more than one person received the mail.

Viewing outgoing mail is similar to viewing incoming mail. Refer to figure 4.3 to see an example of viewing mail items.

Mail Directory

The DIRECT program allows you to create, change, and remove entries from the mail directory file. This file is used by the Mail facility to make it easier for you to refer to other users.

A directory file is a list of users (local or remote) with whom you communicate. Each "entry" in the file defines a user or group of users by a *nickname*. When you are sending mail, you can use the nickname in place of the person's mail address.

When you choose item "4" on the Mail's main menu, the DIRECT program is invoked. Figure 4.6 shows the MAIL DIRECTORY screen of this program.

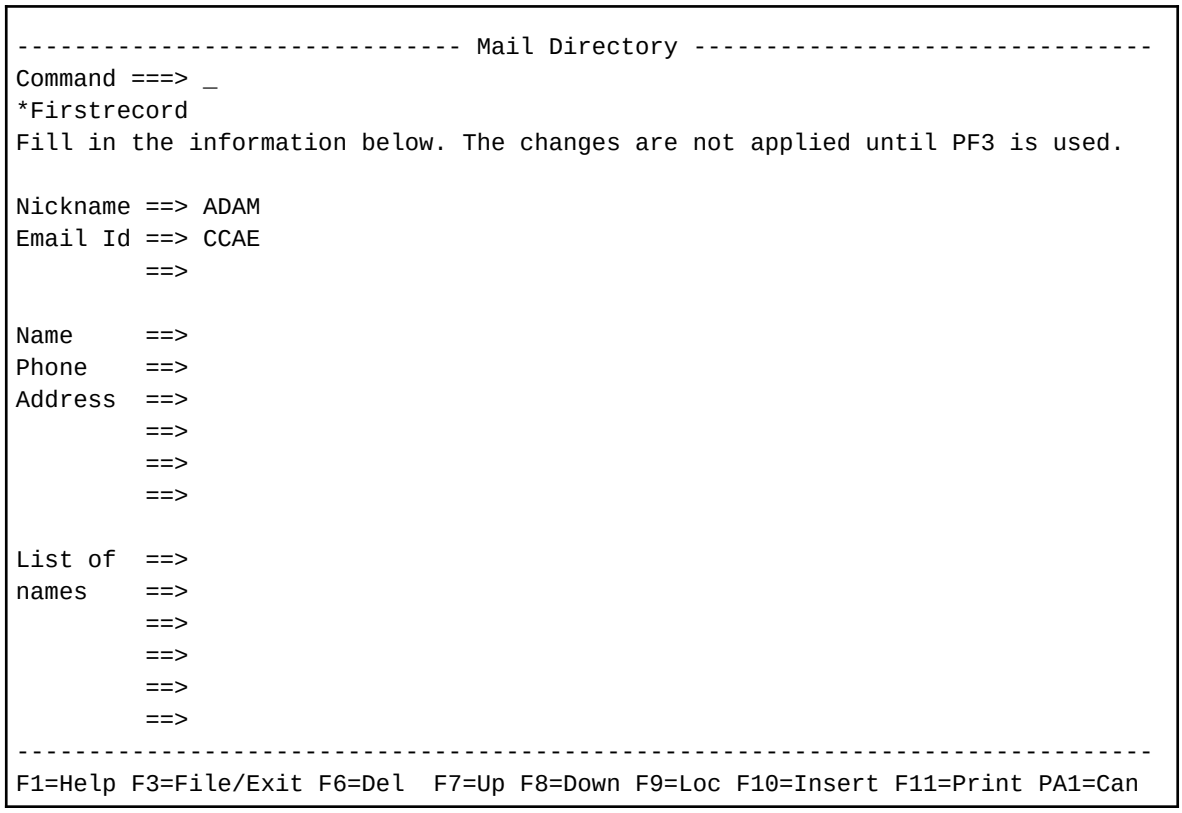


Figure 4.6 - Mail Directory

When the Mail Directory program is started, the first record of the directory file is displayed. The fields on the screen are as follows:

Nickname — is any name you choose to represent a single user or list of users. In figure 4.6 a sample entry is shown, where "ADAM" is the nickname that will represent the userid "CCAE".

Email Id — is the full email address of the person whose nickname you specified. If this user is located on the same system as you, you can leave out the system name. You must leave this field blank if the nickname represents a list of users. The LIST OF NAMES field is filled in instead.

Name, Phone, Address — The name, phone, and address fields are not needed by the directory program. You can put any information you like in these fields, or leave them blank.

List of Names — Specifies all the names of the people for a distribution list. The names in the list can be specified in the following ways:

1. The userid. Each userid is assumed to exist at the location specified by SYSTEM.
2. A nickname that is already defined in your mail directory.
3. A userid and a system name for people on other systems. For example:

```
userid@other
```

Where @ is used to separate the userid from the other system name.

Example of a Distribution List

Figure 4.7 below shows an example of a nickname representing several people.

```
------------------------------ Mail Directory --------------------------------
Command===> _

Fill in the information below. The changes are not applied until PF3 is used.

Nickname==> GOLF
Email Id==>
        ==>

Name    ==>
Phone   ==>
Address ==>
        ==>
        ==>
        ==>

List of ==> CCSM CCKM CCMS CCEL CCFP CCGM CCDT
names   ==> ADAM JOYCE STEVE
        ==> GARY@SMU1.LAN.MCGILL.CA
        ==> ROSIE@SERVICE.LAN.MCGILL.CA
        ==>

------------------------------------------------------------------------------
F1=Help F3=File/Exit F6=Del  F7=Up F8=Down F9=Loc F10=Insert F11=Print PA1=Can
```

Figure 4.7 - Nickname for Several People

To send mail to all the people listed above, you need to use the nickname "GOLF" for the mail address. Notice that each person can be entered in the list of names in three ways. Here are some examples:

CCSM is the userid of someone on your system.

ADAM is a nickname that you have previously added to the directory file.

GARY@SMU1.LAN.MCGILL.CA
is the full address of someone on another system.

See the section "External Mail" later in this chapter for information on sending mail outside your system.

Mail Profile

Item "5" on the main menu of Mail invokes the Mail Profile program (MPROF). It allows you to change the default settings for certain fields on the screens of the Mail program. After you have changed your mail profile, your values will appear on the screens instead of the pre-set values. For example, you can add your full name to display in the MAIL FOR field instead of your userid.

Help is available by pressing F1 (Help) once this program is invoked.

Figure 4.8 shows the first screen (main menu) of the Mail Profile program.

```
--------------------------- Mail Profile Facility ---------------------------
COMMAND ===>

Enter a SELECT OPTION, then press ENTER to process the request.

 1  General Mail Options
 2  Create and Send Mail Options
 3  Print Mail Options
 4  Incoming/Outgoing Mail Options
 5  Refuse Mail Options
 6  Surrogate Mail Options
 X  Exit the Mail Profile Facility




------------------------------------------------------------------------ 08.53.27
F1=Help  F3=End/Save  F12=Retrieve  PA1=Exit/Nosave
```

Figure 4.8 - Mail Profile Screen

When you start the Mail Profile program, there are several choices on the menu to choose from (1 - 6 and X). Each selection allows you to alter default values for different functions of the Mail program. Once you have made a selection, you can page forward to other selections on the main menu by using F8 (Next Item). This way you do not have to return to the main menu. Help is available with each screen by pressing F1.

Selection "1" on the main menu is for general mail options. Figure 4.9 shows a sample of this screen.

```
----------------------- Mail Profile General Options -----------------------
COMMAND ===>

Name      => Janice Brown
Email Id  => CCJB@MUSICM.MCGILL.CA


Forward   =>
mail to

Default Dates and Times
Release Date => ________ (dd/mm/yy)     Release Time => ____ (hhmm)
 Expiry Date => ________ (dd/mm/yy)      Expiry Time => ____ (hhmm)

Initial Command ===> _____________________________________________


------------------------------------------------------------------------ 08.54.16
F1=Help  F2=Alt System  F3=End  F8=Next Item  F12=Retrieve  PA1=Exit/Nosave
```

Figure 4.9 - Mail Profile General Options

The GENERAL OPTIONS screen has the following fields:

Name Your name should be added to this field.

Email Id — is your full email address including userid and system name.

Forward mail to — Enter a email address or nickname in this field if you wish all your mail to be received by another person.

Default Dates and Times

Whenever you send mail, the Mail program includes default dates and times for release and expiry of each mail item. The release of mail is done at the current date and time and the expiry is in one year from the current date. It is recommended that you leave these fields as is. Release and expiry times can be changed for an individual mail item at the time of sending it.

Initial Command — Enter a selection code from the main menu of Mail in this field. For example if you enter "1", then whenever you invoke the mail program you will go directly to the "View Incoming Mail" screen, bypassing the main menu of Mail. The following are the list of selection codes on the main menu:

1 Read Incoming Mail
2 Create and Send Mail
3 Outgoing Mail (Acks and Unrec'd)
4 Mail Directory
5 Mail Profile
6 Public Directory
7 Mailbook Facility
8 List Manager
9 What is new in the Mail Facility
A Frequently Asked Questions (FAQ)
X Exit

Function Keys for the Mail Profile Facility

The following function keys are valid:

F1 Help
provides help for each screen.

F2 Alt System
selects an alternate system for your profile (General Options screen only). If your MUSIC system is known by a number of aliases, you can choose which one of these alternates to use for your system name. This is helpful when you are sending mail to a system which does not understand domain names e.g. music.univofx.edu. In this case, you would want to use your BITNET (RSCS) nodename e.g. univxmus. See the section "External Mail" for information about sending mail outside your system.

F3 End
returns to the main menu of the Mail Profile. If you are viewing the main menu of the Mail Profile, then all changes made to the profile screens are saved and you exit the Mail Profile Facility.

F7 Previous Item
goes to the previous screen (for selections 2 - 6).

F8 Next Item
goes to the next screen (for selections 1 - 5).

F12 Retrieve
redisplays the last command entered in the command area. Up to 5 previous commands can be recalled.

PA1 Exit/Nosave
exits the Mail PROFILE program and ignores changes made to any screens.

> **IMPORTANT:** No changes to your mail profile are recorded until you press F3 to exit the facility. If you use PA1, then changes to any of the profile screens are ignored.

External Mail

It is possible to send mail outside of your MUSIC system to another system on your mainframe computer or to another computer. Check with your installation to see what is supported.

Many educational institutions are members of BITNET or the Internet. These networks provide access to other colleges and universities around the world. Many institutions host discussion lists that provide a forum for people to discuss topics of interest.

To join a discussion list, use the "List Manager" selection on the MAIL main menu. (Your institution must have access to BITNET or the Internet.) Help is provided once this selection is made.

If your institution has access to BITNET or the Internet, then you can send messages to other institutions. The only difference in sending mail outside your system is the longer mail address needed. (Nicknames come in handy for storing long mail addresses.) Be careful when taking note of someone's electronic mail address, not to miss any characters. Once the mail has left your system, there is no way to keep track of where it goes.

If you wish, try sending a message to the the MUSIC Product Group of McGill University. We would appreciate hearing from you about your experience with this guide. Were you able to learn MUSIC easily? Do you have any comments for improving future editions?

Our electronic mail address is:

NEWUSER@MUSICM.MCGILL.CA

Other Forms of Communication

Conferencing

The CONF program offers a forum for MUSIC users to discuss topics of mutual interest. An electronic conference can contain several topics that you can add your comments to. More information can be found in the help topic "CONF".

Discussion Lists

Through e-mail you can join discussion lists. The LISTSERV software on BITNET provides a forum for you to discuss topics of interest with other computer users at different sites.

MCS (MUSIC/SP Client/Server)

The client/server software is available for users to communicate easily between their own computer and a computer running MUSIC. The software runs on a PC platform and uses the familiar graphical user (GUI) interface. E-mail can be done offline and later sent through MUSIC. MCS is described in the help topic "MCS" (complete documentation is included on the MCS diskette).

Internet Access

The Internet is a large network encompassing heterogeneous sites world-wide. The TELNET and FTP commands provide access to other computer sites. TELNET lets you sign on other computers (provided you have a userid and password), and FTP lets you transfer files to and from other sites. Many sites allow anonymous access.

The *MUSIC/SP Internet Guide* describes various ways to communicate using different MUSIC programs. The following is a brief description of some of these methods:

News Reader	The RN program on MUSIC provides access to Usenet. Through this interface you can: read incoming news from a variety of news groups, post your own news items, and follow-up on existing items.
Gopher	GOPHER is a command that accesses the Internet Gopher Protocol for document search and retrieval. Users can view documents residing on many different computers connected to the Internet.
Web	MUSIC includes a Web line-mode browser and a Web server. The Web server allows users to store WWW (World-Wide Web) documents ready for access by a Web browser on your computer. These documents include HTML (HyperText Markup Language) tags. More information about creating these files on MUSIC can be found in the *MUSIC/SP Internet Guide*.

Chapter 5. MUSIC/SCRIPT

Sample SCRIPT Session

The following is a sample session. Computer messages are shown in capital letters. The user in this sample selected item "C" (Create) from the TODO menu, and called the new file *sample*. You might want to try this sample out yourself at your workstation.

```
SELECT OPTION ===> c sample          (puts you into Input mode
                                     of the Editor; begin
                                     typing your document)
```

```
    This sample text shows what SCRIPT can do when given
no special options.
Notice how a
sentence can be split
into
multiple
segments
when you are entering text.
    It is easier to add or remove
complete sentences later on if you start each new
sentence on a new line as shown in this sample.
The SCRIPT program will take care of combining sentences
to form a full print line.
                                    (Press F12 to go to the
                                     command area; type the
                                     Editor command FILE)
file
SAMPLE
NEW FILE
SAVED
PRESS ENTER TO CONTINUE...
```

Figure 5.1 - Selecting Item C and Input of a New Document

```
    This sample text shows what SCRIPT  can do when given no
special options.   Notice how a  sentence can be  split into
multiple segments when you are entering text.
    It is  easier to add or remove  complete sentences later
on if you start each  new sentence on a new line as shown in
this sample.  The SCRIPT program will take care of combining
sentences to form a full print line.
```

Figure 5.2 - Output Document

Now you have seen what SCRIPT does by default, when no specific instructions are given. Lines are

combined and additional spaces put in between words to expand these lines where necessary to give an even right margin. To conform to accepted typing standards, SCRIPT automatically inserts two blank spaces between the character ending the first sentence and the first letter or symbol which begins the next sentence. The end of a sentence is recognized by the occurrence of three characters: a period (.) a question mark (?), or an exclamation point (!), optionally followed by either the quotation mark (") or close parenthesis ()) characters.

You may want SCRIPT to center input lines or perhaps double space the output when it is printed. The next section describes SCRIPT *control words* which will allow you to format your document.

Note: In the examples above, the input file is completely different from the output document. You receive the formatted version only after the input file is executed. Documents are stored in the Save Library exactly the way they have been typed in. Each time you wish to print the formatted version, select the TODO menu. Refer to *Chapter 6 - TODO Menu Items* and read the first section about Executing SCRIPT documents for more information.

Input Control Words

You can control the format of the output document by using *control words* in your file. Control words are 2-letter codes preceded by a period (.) which dictate what a document page will look like. Control words define parameters like top and bottom margins, line spacing, line length, and so forth.

A control word may have additional information on it which is separated from the control word name by **one** blank space. For example, the control word to center two text lines is ".CE 2".

Control words are printed in this guide in upper case. However, they can be typed in either upper or lower case. It is usually easier to use lower case.

Lines that start with a control word are called "control lines". Lines that are not control lines in your file are called "text lines".

You might see control words starting with two periods. These are special "user-defined" control words, which are explained later in this section.

General Formatting Rules

Filling lines (giving even right and left margins) is automatic (default) with SCRIPT, but there are some rules to follow if you want this type of formatting. These same rules apply if you want concatenation (.CO), which is how this guide was produced. With .CO there are no blank spaces inserted between words to give an even right margin. SCRIPT will arrange the text to print as much as possible on each line without exceeding the line length.

- Input lines must be typed starting on the left, in column 1 and must be singled spaced. SCRIPT then knows that each line is a continuation of the line before. It can then rearrange the words to produce formatted text.

- If you move away from the left margin then you are asking SCRIPT to break the formatting at this point. If you are indenting the first line of a new paragraph then this is exactly what you want. However, if you accidentally space a line over, even by just one space, your formatting will break at this place in your document. You might not notice a small mistake like this as you are typing, but this mistake is obvious when you later see the formatted version.

- Leaving blank lines between lines of text also implies a break. For separating paragraphs this is ideal, but don't try to type your document double spaced (leaving blank lines between text) if you want filling. Instead, use the .DS control word in front of your text and type the text single spaced. Later when you execute your SCRIPT document, you will see the double spacing.

- Sometimes you may want to break the formatting without indenting or leaving blank lines. See the BREAK (.BR) control word and the NO FILL (.NF) control word for information. Most control words imply a break and this will be mentioned in their descriptions.

Many defaults are already included in the SCRIPT program to format documents. For example, a page length of 66 (.PL 66) lines is automatic. See the next section "MUSIC/SCRIPT Defaults" for a complete list of format defaults.

Note: As a rule, do not type in any text that starts with a period, unless it is a SCRIPT control word. However, if you must enter text that starts with a period at the beginning of an input line, you can use the control word .LI to tell the system not to read control words for the number of lines you

specify. See "Miscellaneous Control Words", later in this section for more information about .LI.

MUSIC/SCRIPT Defaults

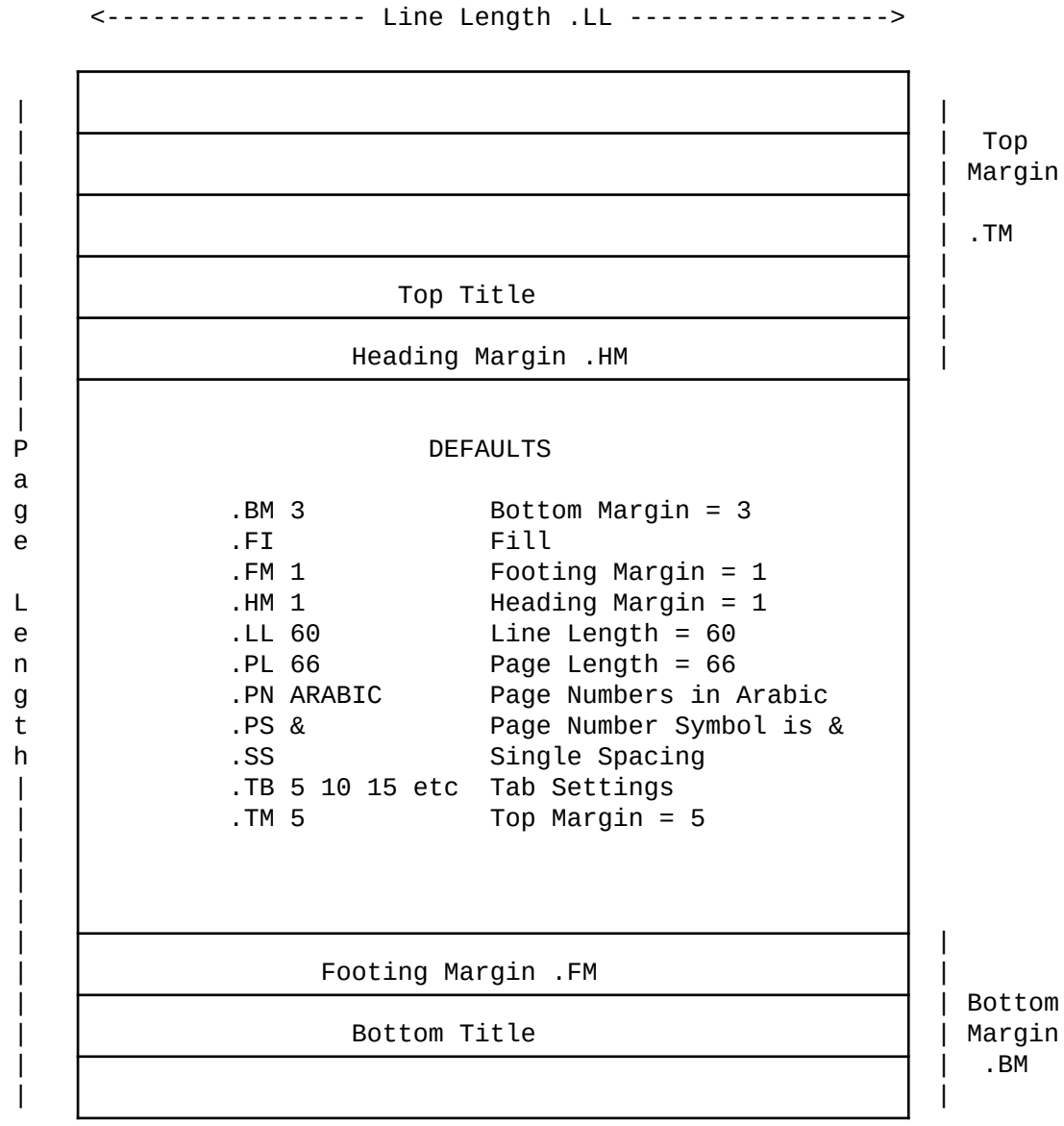


Figure 5.3 - SCRIPT Default Control Words

Note: The TOP (.TM) and BOTTOM (.BM) margins indicate the amount of blank lines to leave at the top or bottom of each output page. This space is included within the PAGE LENGTH (.PL). The default page length is 66 lines per page. Most printers print at six lines per inch (LPI), so 66 completely fills up 11-inch long paper. The top margin is five lines by default, so there is 5 blank lines at the top of each page (almost one inch). The bottom margin is three lines by default, so there will be three blank lines (1/2 inch) at the bottom of each page.

```
 66      (page length)
- 8      (top and bottom margin)
 58      (= lines of text per page)
```

Space is reserved for top and bottom titles as shown in the diagram. SCRIPT does not include titles by default. (Read "Single Line Titles" later in this chapter for instructions.) Often these titles are used for page numbering. See "Page Numbering" later in this chapter.

Control Word Descriptions

This section contains descriptions of all MUSIC/SCRIPT control words, divided into the following categories:

Page and Line Length Control

.PL	Page Length	.PA	Page Eject
.LL	Line Length	.EP	Even Page Eject
.TM	Top Margin	.OP	Odd Page Eject
.BM	Bottom Margin	.CP	Conditional Page Eject

Basic Formatting

.FI	Fill	.LS	Line Spacing
.NF	No Fill	.CE	Center
.CO	Concatenate	.RI	Right Adjust
.SP	Space	.UP	Upper Case
.SK	Skip.	.UL	Underline
.BR	Break	.TB	Tab
.SS	Single Spacing	.TS	Tab Substitute
.DS	Double Spacing		

Accented Characters

.BS	Backspace	.DK	Dead Key

Paragraph Styles

.IN	Indent	.UN	Un-Indent

Text Groups

.(	Group Start	.)(	Group End/Start
.)	Group End		

Title Groups

.TT(	Top Title Start	.BT(	Bottom Title Start
.ET(	Even Top Title Start	.EB(	Even Bottom Title Start
.OT(	Odd Top Title Start	.OB(	Odd Bottom Title Start

Single Line Titles

.TT ////	Top Title	.EB ////	Even Bottom Title
.ET ////	Even Top Title	.OB ////	Odd Bottom Title

.OT //// Odd Top Title
.BT //// Bottom Title
.HM Heading Margin
.FM Footing Margin

Page Numbering

.PS Page Number Symbol
.PN Page Numbering

Miscellaneous Control Words

.CM Comment
.LI Literal
.BS Backspace
.DK Dead Key
.ZN Zone
.TR Translate

User-Defined Control Words

.DF Define

Modification Flagging

.MF Modification Flagging

Conditional Sectioning

.CS Conditional Section

Creating a Table of Contents

.OX Output Auxilliary

Incorporating Other Files

.IM Imbed
.AP Append
.EF End File

Workstation Interaction Control Words

.RD Read
.TE Terminal Input
.TY Terminal Type

Equivalent Functions

.AD Adjust
.FO Format
.ST Stop
.OF Offset
.HE Heading

For a list of control words in alphabetical order, refer to the index under "Control Words".

Page and Line Length Control

Page Length .PL

```
.PL n
```

Specifies the length of the output page in units of lines. By default the page length is set up as 66 lines (the actual length for 11-inch paper). This page length includes lines reserved for titles, page numbers, etc. (top and bottom margins which are discussed later), so the actual number of lines in the main body of the page is usually less than this number (approximately 58 lines of text in a page length of 66 lines). If the page length is changed within the document, the new setting becomes effective on the next page. It may take effect immediately if the new length will fit on the current page. The page length may be in the range of 10 to 250.

Line Length .LL

```
.LL n
```

Defines the length of each output line. The default is 60 characters long. The length may be in the range of 10 to 130. The output device must be able to handle the length you give here. The length specified here is the visual length, that is, it is not affected by backspace characters. The control word .BR is implied.

Example:

The following is an example showing the use of the .LL control word. Also you can see that the input file looks completely different from the output. The SCRIPT program formats the document when you choose the TODO EXECUTE (X) menu item. Refer to the section EXECUTE in Chapter 6 for details.

```
.ll 30
This control word will cause SCRIPT to use a line
length of 30 characters per line.
As you can see, when this document is executed,
it will have even right and left margins.
.ll 50
Now the line length has been changed and a break
(.BR) in formatting has occurred.
```

Figure 5.4 - Sample Input Using .LL

```
This control  word will  cause
SCRIPT to use a line length of
30  characters per  line.   As
you   can   see,   when   this
document is executed,  it will
have   even  right   and  left
margins.
Now the line  length has been changed  and a break
(.BR) in formatting has occurred.
```

Figure 5.5 - Output Document

Top Margin .TM

```
.TM n
```

Reserves *n* lines at the top of each page to contain titling information. At the start of each SCRIPT run, SCRIPT leaves 5 spaces to contain the single line titles. The definition of a title group (discussed later) overrides any specification given with this control word. The *n* parameter may be zero. Specifying .TM 0 is a convenient way to tell SCRIPT not to give any blank lines at the top of a page (an option you may want to use if you are typing letters. A .BR control word is implied.

Bottom Margin .BM

```
.BM n
```

Similar to .TM except it defines the number of lines to be reserved at the bottom of each page for bottom titles. At the start, SCRIPT reserves three lines for this function. A .BR control word is implied by .BM.

Page Eject .PA

```
.PA n
    +n
    -n
```

Causes text to start at the top of a new page. Usually, SCRIPT tries to print the entire current output page defined by the .PL and .LL control words, before it jumps to a new page and prints the subsequent lines. A new page is not started if the .PA control word follows a .PA or similar new page control word.

If the option *n* is used, then the new page is called page number *n*. For example, specifying ".pa 8" causes the new page to be numbered page 8 if a page number control (see "TITLE GROUPS" later in this chapter) is in effect. This option of the .PA control is useful if you want to print only certain numbered pages of a document. A .BR is implied by this control word.

Also, if page numbering controls are in effect, new pages subdivided by decimals can be inserted into the document by specifying ".pa 15.5", which causes the next new page to be numbered 15.5. If the page number contains a decimal portion, then the subsequent pages will be in the decimal format (i.e., 15.6., 15.7, etc). See the .PN control word later in this chapter for more information.

The specification of +n or -n is used to increment or decrement the page number by the specified amount. The page number is never allowed to go below 1.

Even Page Eject .EP

```
.EP text
```

Similar to .PA except that the new page is always even numbered. Thus a blank page may be generated by this control word. If you wish to put a message on the otherwise blank page, you may follow the .EP control word with text. For example, ".ep This page is intentionally left blank" causes a page with that message to appear if the current page number is even. The message is centered on the page. (Note that the page number 15.1 would be considered even since it would follow the odd page number 15, while 16.1 is considered odd. A .BR control word is implied by .EP.

Odd Page Eject .OP

```
.OP text
```

Similar to .EP except that the new page is always odd numbered. This control word is particularly useful when the output document is to be printed on both sides of the paper and you want each new chapter or section to begin on a right-hand page. A .BR is implied by this control word.

Conditional Page .CP

```
.CP n
```

Causes an immediate skip to a new page only if *n* more text lines cannot be printed on the current page. For example, if you have a portion of text that has 10 lines that you want to keep together, specifying ".cp 10" causes SCRIPT to check if there are 10 lines remaining on the current page and to move that portion of text to the next page if there aren't. (The specification of n as zero is meaningless.) The .CP control word can ensure that a paragraph heading and the start of a paragraph occur on the same page.

A .BR control word is NOT implied by this control word. Since the .CP control word does not perform a break operation, it is possible for the last partially filled line to be forced to print on the new page. This does not happen when the .CP control word is preceded by a .BR, .SP or other control word that forces a break. Text groups, (to be discussed later) can accomplish a similar function to this control word.

Example:

The following shows the use of .PL, .TM, .BM, .PA, and .CP control words.

```
.pl 10
.tm 1
.bm 1
If a page length of 10 lines per page is used then only
2 lines of text shows on each page.
This happens because SCRIPT will leave a top
margin of 5 blank lines at the top of every page
and a bottom margin of 3 by default.
By decreasing the top and the bottom margins to 1
(.TM 1 & .BM 1) then another 6 lines of text is added.
Each page now consists of 8 lines of text with 1 blank
line at the top and bottom.
.pa
Using .PA will cause SCRIPT to start a new page even if
the previous page is not full yet.
.sp
.cp 7
By including the control word .CP 7, SCRIPT will make sure
there are at least 6 empty lines left on the page.
If there isn't room, the text will start at the top of
the next page.
```

Figure 5.6 - Input Document with .PL .TM .BM .PA .CP

If a page length of 10 lines per page is used then only 2 lines of text shows on each page. This happens because SCRIPT will leave a top margin of 5 blank lines at the top of every page and a bottom margin of 3 by default. By decreasing the top and the bottom margins to 1 (.TM 1 & .BM 1) then another 6 lines of text is added. Each page now consists of 8 lines of text with 1 blank line at the top and bottom.

Using .PA will cause SCRIPT to start a new page even if the previous page is not full yet.

By including the control word .CP 7, SCRIPT will make sure there are at least 7 empty lines left on the page. If there isn't room, the text will start at the top of the next page.

Figure 5.7 - Output Document

This section explains how to control the format of the printed document through the use of SCRIPT control words.

Fill .FI

```
.FI
```

Instructs SCRIPT to format output lines by shifting words to or from the next line and filling out each line with extra interword blanks so that the left and right margins will be even or "justified". This is the default setting at the beginning of each SCRIPT run, so you don't have to specify this control word at the beginning of your document. The effect of the .FI control word is continual. You must enter an .NF to deactivate it. A .FI implies a .BR.

No Fill .NF

```
.NF
```

Causes the output lines to be printed in the same format as you entered them as input text lines.

You may have a requirement to type tabular information into your document. In this case, you may not want SCRIPT to fill each line. In order to stop the .FI use the control word .NF, which stands for "no fill". The .NF control word remains in effect until SCRIPT encounters a .FI control word. In .NF mode you can still use the .IN control word (described later) if you wish to have indented lines in the output. A .BR is implied by .NF.

Concatenate .CO

```
.CO
```

Forms output lines by shifting words to or from the next input line without adding blanks between the words. The output is as close as possible to the specified line length without exceeding it, giving you a fairly even (but ragged) right margin. A .BR control word is implied by .CO.

Note: For concatenation you need both .NF and .CO.

Space .SP

```
.SP n
```

Inserts *n* blank lines between sections of output text. If *n* is omitted, 1 is assumed, therefore, .SP is the same as .SP 1. If the output document is being double spaced by the .DS or .LS control words, then the number of spaces generated will be double that given as n.

If the end of page is reached during the space operation, no more blank lines are inserted. If the current page is filled up at the time the .SP control word is encountered, then no spaces are added. If you must leave a specified number of lines you can do this by the combination of the .CP and .SP control words To insert blank lines at the top of a page use .PA and then .SP. The .SP control word implies a .BR.

Notes:

1. When using full-screen mode, simply use the the NEW LINE (down and to the left arrow) key to insert blank lines between paragraphs. Blank lines entered with this key will be appear in the output document even if they occur at the top of a page.

2. If you are not using full-screen mode, you must use .SP (or .SK) to insert blank lines because the NEW LINE key does not function. Pressing the ENTER key twice puts you into Edit mode with the message "EDIT" appearing. To continue inputting text, just type in the Editor command INPUT and you return to Input mode.

Skip Lines .SK

```
.SK n
```

Similar to .SP except for the following two cases. The .SK command does not cause blank lines to be left at the top of a page. In addition, when the .SK command occurs between lines of text it causes at **least** *n* blank text lines. (This distinction allows for a possible future change to SCRIPT whereby blank filler lines can be expanded to best meet the page formatting style.)

The .SK command is particularly useful when used in user-defined control words, which are explained later in this chapter.

Break .BR

```
.BR
```

Informs SCRIPT that the next text line is not to be joined to the end of the last one, thereby breaking formatting. .BR is normally used to indicate the start of a new paragraph. A blank in the first position of an input text line also implies a .BR. This control word is only meaningful when either .FI or .CO is in effect. (Several other control words have the effect of the .BR control word in addition to their other function.)

Example:

The following shows the use of .FI, .NF and .CO controls words and their effect on formatting text. Note that these control words imply breaks (.BR).

```
.tm 0
.bm 0
Without specifying any control words
SCRIPT will start off in fill mode (even right margin).
To separate paragraphs with a blank line use .SP n.
.sp 1
If you need more blank lines between the text, then
change the number beside this control word.
Another way to separate paragraphs, thereby stopping
the filling, is by using a break control word.
.br
If you want SCRIPT to print out text in the
same format as it was
entered, use the no fill control.
.sp
.nf
Now the document will print exactly
as it was typed in.
A break (.BR) in the formatting is implied
when this control word is used.
.sp
.co
By using the concatenate control, SCRIPT will put
as much on a line as possible without adding spaces
between words.
Notice that the right margin is now ragged.
.sk
.fi
To have SCRIPT change back to fill mode,
a fill control word is necessary.
After inserting this control word, the output document
will have an even margin again.
```

Figure 5.8 - Input Document with .SP .BR .NF .CO .FI

```
Without specifying any  control words SCRIPT will  start off
in fill  mode (even right  margin).  To  separate paragraphs
with a blank line use .SP n.

If you  need more blank lines  between the text, then change
the  number  beside  this  control  word.   Another  way  to
separate  paragraphs,  thereby  stopping  the filling, is by
using a break control word.
If you want SCRIPT  to print out text in the  same format as
it was entered, use the no fill control.

Now the document will print exactly
as it was typed in.
A break (.BR) in the formatting is implied
when this control word is used.

By using the concatenate control, SCRIPT will put as much on
a line as possible without adding spaces between words.
Notice that the right margin is now ragged.

To have SCRIPT change back to fill mode, a fill control word
is necessary.  After inserting this control word, the output
document will have an even margin again.
```

Figure 5.9 - Output Document

Single Space .SS

```
.SS
```

Cancels the effect of a previous .DS (Double Spacing) or .LS (Line Spacing) control word and returns the output to single spacing. .SS is a default SCRIPT control word. A .BR is implied by this control word.

Double Space .DS

```
.DS
```

Causes subsequent output lines to be double spaced. This is not to be confused with the .SP and .SK control words, which apply to the spacing only between the previous text line and the next text line. A .BR control word is implied.

Line Spacing .LS

```
.LS n
```

Specifies the number of blank lines to be inserted between the lines of text in the output document. Thus .LS 1 is identical to .DS. This line spacing feature is particularly useful when printing rough draft copies. A .BR is implied by .LS.

Example:

In this example .DS and .SS control words are emphasized. Notice that these control words do not add spaces immediately, but affect the spacing between lines.

```
By default SCRIPT will automatically start
with single spacing (.SS).
If you want to change this formatting
style use .DS or .LS n.
.sp 1
.ds
This portion of the document is now double spaced.
Notice that .DS does not give a blank line
immediately; .SP 1
was needed to put a blank line between each paragraph.
However, .DS will add a blank line after every text
line from now on.
.ss
Notice that a .SP was not used this
time to separate each paragraph, because a blank
line was automatically added to the last
text line because of the .DS control word.
```

Figure 5.10 - Input Document with .DS .SS

```
By  default  SCRIPT  will automatically  start  with  single
spacing (.SS).  If you want  to change this formatting style
use .DS or .LS n.

This portion of  the document is now  double spaced.  Notice

that .DS does  not give a blank line immediately;  .SP 1 was

needed to put a blank line between each paragraph.  However,

.DS will add a blank line after every text line from now on.

Notice that  a .SP was not  used this time to  separate each
paragraph, because a  blank line was automatically  added to
the last text line because of the .DS control word.
```

Figure 5.11 - Output Document

Center .CE

```
.CE n
    ON
    OFF
```

Centers the next *n* lines. If n is not given, the default is one line. If a .CE ON control is specified, then all subsequent text lines are centered until SCRIPT reads a .CE OFF control word.

Note that each input text line will form one output line. It is therefore important that the input line not exceed the current line length or else the line will be truncated. A .CE control word implies .BR.

Right Adjust .RI

```
.RI n
    ON
    OFF
```

Similar to .CE except that the output lines flushed right rather than centered. creating even right and ragged left margin.

Upper Case .UP

```
.UP n
    ON
    OFF
```

Translates all lower case alphabetic letters into upper case, but leaves numbers and symbols intact. This control word is useful when certain parts of the document are to be printed only in upper case. If the whole document is to be printed in upper case, then you can use the ALLUP control word when SCRIPT is being run. The .UP control word is particularly convenient when typing in lines that are a mixture of upper case letters and numbers since you would normally have to hit the SHIFT key to type the letters and unshift to type the numbers. For example,

```
.up
May 13, 1989
```

would look like:

```
MAY 13, 1989
```

Entering .UP *n* tells SCRIPT how many input text lines are to be printed in upper case (where *n* equals the number of lines). If *n* is unspecified, the default is one. The ON option will translate all the text lines to upper case until SCRIPT encounters a .UP OFF control word.

Example:

The following shows the use of .CE, .RI and .UP. The first figure shows a section of a file and the second figure displays the output document.

```
.fi
The .FI control word is in effect now.
Notice that I am going to use the .SP control word
to insert a blank line between this line and
the next one.
.sp
.ce 2
Now I want to center this line,
and also this one.
.sp 1
.ri 2
.up 1
This line is right adjusted and capitalized,
this line is right adjusted only.
```

Figure 5.12 - Input Document with .FI .CE .RI .UP

```
The .FI  control word is  in effect  now.  Notice that  I am
going to  use the .SP  control word  to insert a  blank line
between this line and the next one.

              Now I want to center this line,
                     and also this one.

                THIS LINE IS RIGHT ADJUSTED AND CAPITALIZED,
                           this line is right adjusted only.
```

Figure 5.13 - Output Document

Underline .UL

```
.UL ALL
    MASK
    OFF
```

Controls underlining of your text. A .UL underlines all alphanumeric characters, plus three more special characters / ' - (slash, single quote and dash). It does not underline the blank spaces between words. .UL ALL underlines the entire line including blank spaces between words. Some examples are given below:

```
Input:      This shows how you can
            .ul
            underline
            only the word "underline".

Output:     This shows how you can underline only the word "underline".


Input:      .ul
            Each individual word will be underlined.

Output:     Each individual word will be underlined.


Input:      .ul all
            The entire line will be underlined in this case.

Output:     The entire line will be underlined in this case.
```

Specifying MASK sets a mask option on until SCRIPT reads .UL OFF When this option is active, you can tell SCRIPT to underline specific characters in a line by first typing in the line and then following that line with a line that contains only underline characters located immediately below those you wish to underline. Look at the following example:

```
Input:      .us mask
            This shows how you can use the MASK option to underline
                  _
            certain characters, in this case the o of "shows".

Output:     This shows how you can use the MASK option to underline
            certain characters, in this case the o of "shows".
```

Example:

The following show an actual document that was typed in using all of the underlining control words.

```
.sp
.ce
.ul
How to Underline Text
.sk
     This sample document will show the use of
underlining control words.
If only
.ul
one
word should be underlined just stop at that word and
use the control word .UL just before it.
.ul
By using this control word only the words will be
underlined.
If you need the spaces and punctuation underlined
as well,
.ul all
then add the parameter "ALL".
.ul mask
By specifying "MASK" then from now on
Whenever SCRIPT sees underscores it will
_       _       _     _
use them to underline the line above.
.ul off
Type in OFF with the control word to stop "MASK".
Now _______________ can be used as text.
Notice that underlining does not break (.BR) the
formatting.
```

Figure 5.14 - Input Document with Underlining

```
                    How to Underline Text

     This sample document  will show the use  of underlining
 control words.  If  only one word should  be underlined just
 stop at that  word and use the control word  .UL just before
 it.   By using  this control  word  only the  words will  be
 underlined.   If   you  need  the  spaces   and  punctuation
 underlined   as  well,   then add the parameter "ALL".    By
 specifying  "MASK" then  from now  on  Whenever SCRIPT  sees
 underscores it  will use them  to underline the  line above.
 Type  in OFF  with the  control  word to  stop "MASK".   Now
 _______________ can be used as text.  Notice that underlining
 does not break (.BR) the formatting.
```

Figure 5.15 - Output Document

Tab .TB

```
.TB n1 n2 n3 n4 (etc)
```

Allows you to define and change the tab settings within a document. Tabs are normally used only when typing in tabular information. The default tab settings are .TB 5 10 15 ... 80.

The tab locations are specified by *n1, n2*, etc. These numbers must be listed in ascending order. Note that SCRIPT defines tab stops in a slightly different way than the normal convention. For instance, a setting of .TB 10 15 20 will mean that tab stops will be at columns 11, 16, and 21.

The Editor will show the position of any tab characters in a line by printing a # symbol (this does not apply to tab substitute characters defined by .TS, see below). For example, a tabbed line would appear as "#15#3.2#76".

Note: The .NF control word should be in effect when using tabs to form tables.

Tab Substitute .TS

```
.TS x
```

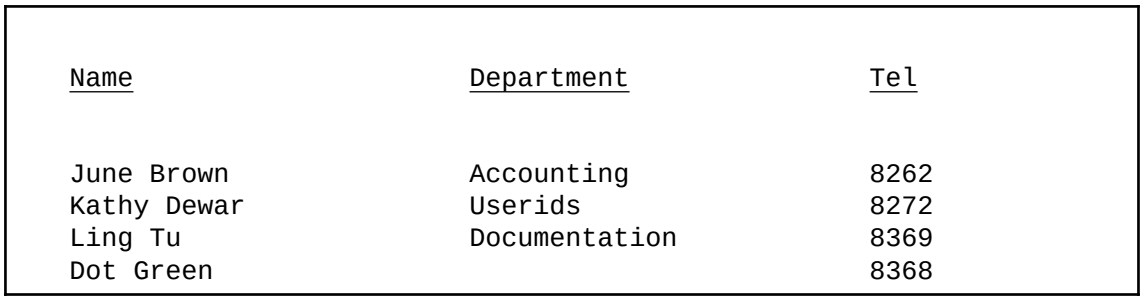
Defines a logical tab character Certain workstations do not have a tab key so it is necessary to specify another key to be used as the tab key. The command .TS > will make the character > the tab character.

Example:

The following shows how to enter a document with tabs. (If the space between columns needs to be increased or decreased, then only the .TB control word needs to be changed.)

```
.sp
.nf
.ts >
.tb 25 50
.ul
Name>Department>Tel
.sk 2
June Brown>Accounting>8262
Kathy Dewar>Userids>8272
Ling Tu>Documentation>8369
Dot Green>>8368
```

Figure 5.16 - Input Document with Tabs



Name	Department	Tel
June Brown	Accounting	8262
Kathy Dewar	Userids	8272
Ling Tu	Documentation	8369
Dot Green		8368

Figure 5.17 - Output Document

Accented Characters

Accented characters are formed by striking two separate keys. For example, an è is formed by the two keys "`" and "e", and ø is formed by the two keys "/" and "o". Since one character must be superimposed on the other, it is necessary either to use a backspace key or to treat one character as a dead key. SCRIPT handles accents in both these ways.

Substitute Backspace Character .BS

```
.BS
```

Normally on SCRIPT, a backspace erases the character backspaced over. In order to have a true backspace, you must define a substitute backspace character. The SCRIPT control word .BS < will define the character < as a backspace. Thus, ∅ would be entered as /<0.

Dead Key .DK

```
.DK x
```

This method requires a keying action that is identical to that of an office typewriter with a dead key facility. Specifying .DK ` will make the grave key (`) a dead key, so the character è would be entered "`e". This technique can be used to make any character into a dead key, making it possible for you to enter other languages that require accents.

Text Entry from Standard Keyboards for Accented Languages

For example, French text can be entered from keyboards not equipped with French characters through the use of substitute characters, although this method is not as convenient as using a French keyboard.

The substitute characters are defined using .TR (described later in this chapter). The internal hexadecimal representations of the French characters are given below. (These representations are for a standard IBM printer; your installation may use a different printer.)

Hex	Character	
EF	é	e with acute accent
46	ç	c with cedilla accent
79	`	grave accent
71	^	circumflex
74	´	acute accent
75	¸	cedilla accent

Note: These two techniques will display the French correctly only when printed with SCRIPT. During the

input and editing phases, the typist will see the character sequences that were used to create the compound characters. When entering French text, the accent must be entered before the letter.

The following figure illustrates the three most widely used paragraph styles:

```
This  illustrates  the   "straight"  paragraph
style.  Notice  that  all three  lines of  the
paragraph are flushed left.

     This  illustrates  the  "indented" style.
Notice  that   the  first  line   is  indented
relative to the second and third line.

This illustrates the "hanging" style since its
       first line  starts  to the  left of the
       second and third line.
```

Figure 5.18 - Paragraph Styles

In SCRIPT, only the overhanging paragraph style needs special control words. The indented style is easiest to accomplish by starting off the first line with the required number of blanks as you would if you were typing this type of paragraph without the help of SCRIPT. The other lines of a paragraph are always typed beginning at the first input position (flushed left).

For the other paragraph types, all the lines of the paragraph are entered starting at the first position. If you wish to indent the entire paragraph a certain number of spaces to the right, then you can use the .IN control word.

Indent .IN

```
.IN n
```

Defines the starting location of the left margin in the output text. The *n* indicates the number of spaces to indent the left margin. Thus if n is 10, then the output will start at position 11. To reset the margin to 0, specify .IN 0. This control word implies a .BR.

```
This sample shows how to use the .IN control word.
If you want a paragraph to be indented by 5
spaces use .IN 5.
.in 5
Notice that a break (.BR) is implied with an indent
control word.
From now on every line will start 5 spaces from
the left.
.in 0
To bring text back to the left margin you must enter .IN 0.
```

Figure 5.19 - Input Document with Indenting

```
This sample shows  how to use the .IN control  word.  If you
want a paragraph to be indented by 5 spaces use .IN 5.
     Notice that  a break  (.BR) is  implied with  an indent
     control  word.  From  now on  every line  will start  5
     spaces from the left.
To bring text back to the left margin you must enter .IN 0.
```

Figure 5.20 - Output Document

Undent .UN

```
.UN n
```

Forms overhanging paragraphs. First enter a .IN to give the position of the left margin of the paragraph. This .IN specification remains in control until reset by another .IN. Second, enter a .UN control word. The number given on the .UN control line is the number of characters that are to overhang on the first line of the paragraph. The .UN is effective only on the one text line that immediately follows. The *n* parameter must not be larger than the number for .IN. The following diagram illustrates how to visualize the combination of .IN and .UN to form an overhanging paragraph.

```
             <-- UN -->
             xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                       xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
<------- IN ---------->xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
                       xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

Figure 5.21 - Indenting and Un-indenting

For following sample input shows how to make the first line of the paragraph to start flush left and the remaining lines to start at position number 10.

```
.IN 9
.UN 9
This paragraph will start at column 1 and continue
at column 10 since the indent
will leave 9 blank spaces.
```

If you want several overhanging paragraphs to appear in a row all you have to do is enter .UN before the first line of each paragraph.

To allow for some special effects, .UN has the following special rule: SCRIPT will never attempt to insert blank characters between the words which will occur in the overhanging portion. Furthermore, if the text line following the .UN control word contains fewer characters than the overhanging portion, then the text from the following line will appear in the output starting at the same column as the remaining lines in the paragraph.

You may use a control word such as .UN -5 to cause a indented paragraph style. This negative form is mainly used in connection with styling standards as described in a later in this chapter. A .BR is implied by

the .UN control word.

The following figures give sample uses of the .UN control word and how it can be used to form overhanging paragraph styles. You should study each example carefully and try them out. (The examples assume a line length of 30).

```
   Sample input to SCRIPT   |    Sample output from SCRIPT
----------------------------+-------------------------------
.in 10                      |
.un 10                      | OPTION    This sample shows
OPTION    This sample shows |           one way to use the
one way to use the .UN      |           .UN control word.
control word.               |           There are easier
There are easier ways.      |           ways.
```

Figure 5.22 - Overhanging Paragraph, Style 1

```
.in 10                      |
.un 10                      |
OPTION                      | OPTION    This is easier to
This is easier to type in   |           type in since you
since you don't have        |           don't have to put
to put extra spaces in      |           extra spaces in your
your first line.            |           first line.
.un 10                      | OPTION2   Note how you can put
OPTION2                     |           in several items
Note how you can put in     |           that use the same
several items that          |           style without having
use the same style          |           to type a .IN
without having              |           again between each
to type a .IN               |           one.
again between each one.     |
```

Figure 5.23 - Overhanging Paragraph, Style 2

```
.in 10                      |
.un 10                      |
PRINT                       | PRINT
.un 10                      | NOPRINT   This example shows
NOPRINT                     |           that .UN can be used
This example shows that     |           to form a variation
.UN can be used to form a   |           of the overhanging
variation of the overhanging|           style.
style.                      |
```

Figure 5.24 - Overhanging Paragraph, Style 3

```
.in 3                        |
.un 3                        |
1. You may want to list some| 1. You may want  to list  some
points with an item number   |    points with an item  number
in front of them.            |    in front of them.
.un 3                        | 2. This is  easy to do, as you
2. This is easy to do,       |    can see.
as you can see.              | 3. Of course  you can use this
.un 3                        |    style with  the first  word
3.                           |    being  separate   from  the
Of course you can use this   |    rest of the first line.
style with the first word    |
being separate from the      |
rest of the first line.      |
```

Figure 5.25 - Overhanging Paragraphs

Text Groups

SCRIPT can assure that a collection of text lines will be kept together on one page in your printout. These groups can be used for various purposes such as charts, tables or title creation. This section deals only with text groups.

Text groups are used to ensure that all the lines in the group appear on the same output page. The .CP control word accomplishes part of this function, although the text group handles the more general case when the number of output lines generated by a sequence of input text is unknown or is variable.

Text Group Start .(

```
. (
```

The start of a group is indicated by the .(control word A .BR is implied by this control word.

Text Group End .)

```
. )
```

The end of a group is indicated by the .) control word. The text group end control word does **not** imply a .BR. (Other types of groups use the .) to end them but in these other cases a .BR is implied.)

A group end can also be implied by either another group start control word including those of other types of groups (such as title groups, for instance), or the occurrence of a .PA. (A .CP control word is ignored if it is found within a text group.)

TEXT GROUP START AND END .)(

```
. ) (
```

If you wish, you may use the .)(to end one group and start another.

Title Groups

Title groups are similar to text groups except that the collection is used to form a title which will be repeated at the top or bottom of every page. A .NF is implied by the start of a title group. When the title group is closed, the original format (ie. either .NF or .FI) returns. (See the next section on page numbering for an example that uses a title group.)

Top Title .TT(

```
.TT(
```

The following example show how .tt(is used to define a title at the top of all subsequent pages:

```
.TT(
This is a title
.SP 2
.)
```

subsequent pages. Notice that a space control word is generally given to separate the title from the main body of the text. This top title remains in effect until it is replaced by another top title. To remove a top title, enter the sequence

```
.TT(
.)
```

Even and Odd Top Titles .ET(.OT(

```
.ET(
text and/or
control words
.)
```

If the output document is to be printed on both sides of the page, you may be required to specify a different title for even and odd numbered pages. This is done by the .ET(and the .OT(control words for the even and odd titles respectively.

Notes:

1. A new title becomes effective immediately and replaces any previous definition.

2. Using .TT(will override any .TM specification.

Bottom Titles .BT(.EB(.OB(

```
.BT(
text and /or
control words
.)
```

The following example shows how to define a bottom title with the .BT(control word:

```
.BT(
.SP 2
bottom title
.)
```

The even and odd bottom title control words are .EB(and .OB(. The same rules apply for bottom titles as for top titles.

Notes:

1. A new bottom title becomes effective immediately and it replaces any previous definition.

2. No bottom title will be printed if a new bottom title being defined requires more lines than are remaining on the current page. (This condition would occur only if the new bottom title generated more output lines than the previous one did.)

3. Using .BT(will override any .BM specifications.

Single line titles can be abbreviated which might make them easier for you to use. The text portion of the title is given right on the control word line as in the following example.

```
.TT /Chapter 1/Introduction/Page &/
```

The slash (/) characters split the title into three parts.

- The first part will start at output position 1.
- The second part will be centered.
- The third part will end in the last output column (see next section about the page number symbol "&").

Any special character may be used instead of the slash but the same one must be used in all four places. Some other examples are given below.

```
.TT ///This title is on the right/
.TT /This title is on the left///
.TT //This title is centered//
.TT //This has a page number on the right/&/
.TT /  This title is indented///
.TT $$This title is centered$$
```

The control words .OB, .EB, .ET, .OT, .BT can also be used in either single line title or group form. You may use either form of titling technique you wish (even a mixture of the two). Single line titles are not allowed to contain backspace characters or be underlined.

Spacing Single Line Titles

You will no doubt want to leave spaces between the title and the start of text on a page. You may also want to leave space at the top of the page before the title line. The following control words control the spacing for single line titles.

Note: Refer to the section "MUSIC/SCRIPT Defaults", earlier in this chapter, for default settings of each output page.

Heading Margin .HM

```
.HM n
```

Specifies the number of blank lines that follow the single line title. If *n* is not given, the default is one. The title group

```
.TT(
.CE
Introduction
.SP 2
```

```
.)
```

will have the same effect as the following sequence using a single line title

```
.TM 3
.TT //Introduction//
.HM 2
.)
```

As mentioned earlier, SCRIPT allows you to reserve space at the top and bottom of your page using .TM and .BM. It is within this space that your single line title would go, as in the previous example which specified .TM 3. Remember that the value of the top or bottom title must be equal to or greater than the line of the title plus the value allowed for the heading and footing margin.

Footing Margin .FM

```
.FM n
```

Similar to .HM except it applies to the bottom title or footer. .FM *n* indicates the number of blank lines to precede the single line title. SCRIPT assumes a value of 1 unless you specify otherwise.

Page Numbering Specification

The occurrence of a special page number symbol ("&" by default) in a title group or single line title (see previous section) causes the current page number to be substituted for that character whenever the title is printed.

For example, either of these sequences

```
.BT //&//

    or

.BT(
.CE
&
.SP
.)
```

will cause page numbering to appear centered on the bottom of every page.

Page Number Symbol .PS

```
.PS x
```

Causes the page number symbol to be defined as the character *x*. The page number symbol of "&" is defined at the start of a SCRIPT run and will be used unless redefined by .PS. If no symbol is specified on .PS, then no page number symbol is defined. The *x* parameter may also be defined by its two character hexadecimal equivalent. (These hexadecimal equivalents are explained later in this chapter.)

Page Number Control .PN

```
.PN ROMAN
    ARABIC
    DEC
    OFF
    OFFNO
    ON
```

Specifies the type of page numbering required. Specifying ROMAN causes the current page number to be printed as lower case Roman numerals when called for by the page number symbol in a title. The maximum Roman numeral that SCRIPT can generate is for the number 3999. Entering ARABIC turns off the ROMAN

option.

The DEC option specifies that the page number is to be incremented by .1 when a skip to a new page is done. Thus if the current page number is 102, then the following page numbers will be 102.1, 102.2 etc. This option is reset by the ON option or by a .PA with a number specified.

.PA can imply the DEC option. It may also be used to force the next page number to a specific value which may include a decimal part.

If the OFF parameter is specified, the page numbers are discontinued on the output (become blanks), although the internal page numbering is still maintained.

The OFFNO parameter discontinues the internal page number incrementing and suppresses the output page numbering as does the OFF parameter.

The ON option resets the OFF and the OFFNO conditions as well as turning off the decimal page numbering option, if it was in effect.

Example:

The following example illustrates the use of title groups and single line titles.

```
.pl 14
.tt(
.ri
May 1, 1989
.sp
.ul
Item                          Cost       Quantity      Total
.sp
.)
.bt /Stationery//Page &/
.cm the text starts here...
.nf
Pencils (HB)                   .15         2 doz        3.80
Pads 8 1/2 x 11               1.50         10 pkgs     15.00
Office tape                    .75         15 rolls    11.25
.sk
.fi
*This sample shows the use of top and bottom titles.
These titles will continue to print on every page unless
they are changed.
```

Figure 5.26 - Input Document with Top and Bottom Titles

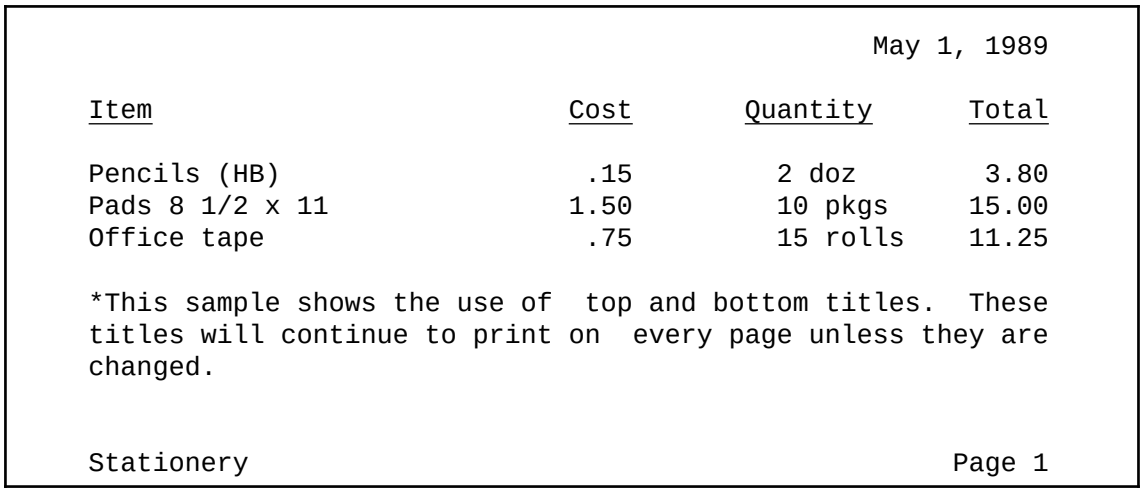


Figure 5.27 - Output Document

Comment .CM

```
.CM text
```

Allows you to insert comments or notes to yourself in the input text which will not show up in the output. by one or more blanks. Thus, the following examples are valid:

```
.CM this is a sample comment
.CM       and so is this
```

Literal .LI

```
.LI n
```

Informs SCRIPT that the next line does not contain a control word. It is only required if the line starts with a period in the first position. The *n* field is used to specify the number of lines that are not to be interpreted as control lines. For example ".LI 3" means that the next three input lines are not control lines. If the *n* parameter is not specified, one is assumed.

Backspace .BS

```
.BS x
```

Defines a logical backspace character which can be used for special overprinting effects. Use .BS x, where the *x* in the control word is the character you wish to use for the backspace. If the backspace character was defined as "<", then the sequence "o</" will cause the compound character "ø" to print. Only certain workstations can print such overstrike characters.

The character x can be defined by a 2-character hexadecimal equivalent. If the character *x* is not given, there is no logical backspace defined. (The handling of the true backspace character is not affected by this control word.)

Dead Key .DK

```
.DK x
```

This method involves an identical keying action to that of an office typewriter with the dead key facility. Specifying ".DK /" will make the slash (/) a dead key, so the character ø would be entered "/o". This technique can be used to make any character into a dead key, enabling you to enter accented characters in various languages. To change the dead key character back to a standard character use .DK x OFF.

Zone .ZN

```
.ZN n
```

Informs SCRIPT how many character positions it should process from each input line. For example, if the input lines contain sequence numbers or modification date information in positions 73 to 80, you will want SCRIPT just to process up to position 72. This is done by specifying .ZN 72. If the ZN control word is not used, or if just .ZN is typed, then SCRIPT will process the entire input line. The .ZN control word is different from all the other control words in that any characters after the number will be ignored. This allows the .ZN control word itself to have a sequence number.

Translate .TR

```
.TR i o
```

Instructs SCRIPT to print any occurrence of character *i* as if it were the character *o*. This is of particular assistance when you need to print characters which you do not have on your keyboard. The *i* and/or *o* parameter may be entered as their 2-character hexadecimal equivalent given in the table below. (See the section "Accented Characters" earlier in this chapter for additional information about .TR.)

The translation remains in effect until specifically modified by another .TR. The specification of just .TR will reset all the translations in effect. The specification of the same parameter in the i and o field has the effect of resetting (turning off) any translation in effect for that character. Note that the translation only affects the output and the translation process is done only at the time each print line has been completely formed.

For example, suppose your keyboard does not have the special character ¬ and you wish to insert this character into the document. You could do this by telling SCRIPT that whenever it sees the character # it is to print it as ¬. (5F is the hexadecimal equivalent for the ¬ character.) The SCRIPT control word for this would be as follows:

```
.TR # 5F
```

You can also use this control word to bypass restrictions about the characters that can be typed in the first position of each input line. For example, if you must type text that starts with a /, you can use a ".TR % /" specification and type in the / characters using the % symbol instead.

Another use for .TR is if you wish to enter a character that will be printed as a blank in the output. This is useful when you wish to leave room in the output text that will be filled in later by hand. The hexadecimal equivalent for a blank is 40.

```
.tr 1 b1
.tr 2 b2
.tr # 8e
.tr % ae
When printing documents on a batch Printer
extra characters that are
not normally found at the workstation are available.
For example:
.sk
        X2 % Y1#2
.sk
.tr 1 1
.tr 2 2
To restore the characters back to the original keys,
use .TR again.
.sk
        1 2
```

Figure 5.28 - Input Document with Translates (*.TR*)

When printing documents on a batch Printer extra characters that are not normally found at the workstation are available. For example:

$X_2 \geq Y_{1+2}$

To restore the characters back to the original keys, use .TR again.

1 2

Figure 5.29 - Output Document with Translates

Note: When restoring or changing characters using .TR, type .TR at a break in formatting. SCRIPT reads ahead to format input lines and it might see the change (.TR) before it has printed the output. In the example above, if ".tr 2 2" were inserted before the .SK, then the "X_2" would print as a "X2" in the output.

The hexadecimal equivalents for special characters varies according to the font chosen at the time of printing.

User-Defined Control Words

User-Defined control words are used for styling and repetitive sequences. When preparing SCRIPT input, it is common to periodically use a specific sequence of SCRIPT control words. For example, before each section heading the following sequence might be specified:

```
.SK 2
.CP 5
.IN 0
.UL
```

SCRIPT allows the user or a group of users to define their own SCRIPT control words. These user-defined control words are made up of a combination of SCRIPT control words and text lines. They can be redefined throughout a document as often as necessary.

Defining your own control words allows you and other users to set your own styling standards. The typist need only know that a certain control word is given at the start of each type of text. The number of spaces between paragraphs, underlining and indenting rules can thus be automatically established and changed without retyping the input text.

Define Control Word .DF

```
.DF ..name
```

Specifies that the lines that follow are to be substituted for the occurrence of the control word *..name.* The name field may be up to six characters in length and may consist of alphabetic as we;; as numeric characters. The letters are taken as upper case letters though your may specify lower case letters, so, ..hd would not be different from ..HD. Thus, ..l, ..newchp are valid specifications on a .DF. Note that a user-defined control word always starts with two periods, while regular SCRIPT control words start with only one.

All the lines following a .DF control word (up to but not including the next .DF control word) become the substitute lines. A .DF with no parameter is used to end the definition. The sequence of lines can even include a .. control word referring to other user defined control words. Such "nested" user control words should not be allowed to occur to a depth of greater than five.

Define Control Word: Short Form .DF

```
.DF ..name=xxxxxxxxxxxx
```

Allows a definition of a user control word in just one line. It is taken to mean exactly the same as the sequence:

```
.DF ..name
xxxxxxxxxxxx
.DF
```

This short form of the define control word can be used to cause a user control word to have no effect such as in the following example:

```
.DF ..name=.CM
```

The information after the equal (=) sign is considered to be all the characters up to the last non-blank one. Thus the following:

```
.DF ..name=this is a sample
```

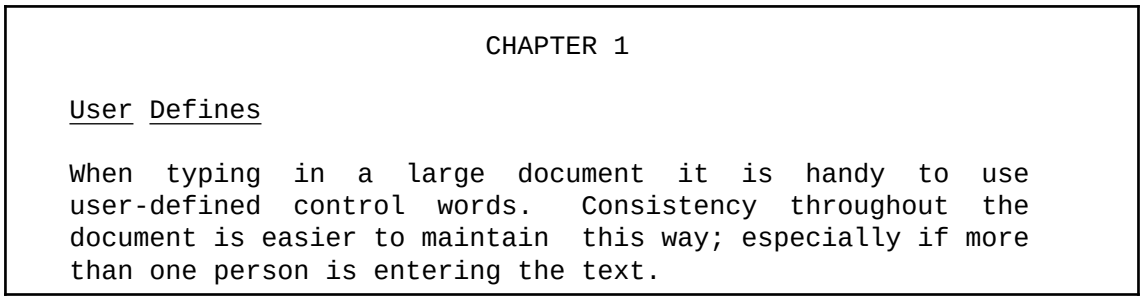
causes the text "this is a sample" to be substituted whenever the control word ..name is given.

Example:

The following example illustrates the use of user-defined control words. Several SCRIPT control lines can be incorporated into one user-defined control word. Comments inserted into the input file will help other users understand your document.

```
.df ..ud=user-defined
.df ..h1
.cm             heading 1
.pa
.ce
.up
.df ..h2
.cm             heading 2
.sk 2
.ul
.cp 4
.df
.cm             the text starts here...
..h1
Chapter 1
..h2
User Defines
.sk
When typing in a large document it is handy to use
..ud
control words.
Consistency throughout the document is easier to
maintain this way; especially if more than one person
is entering the text.
```

Figure 5.30 - Input Document with User Defines



CHAPTER 1

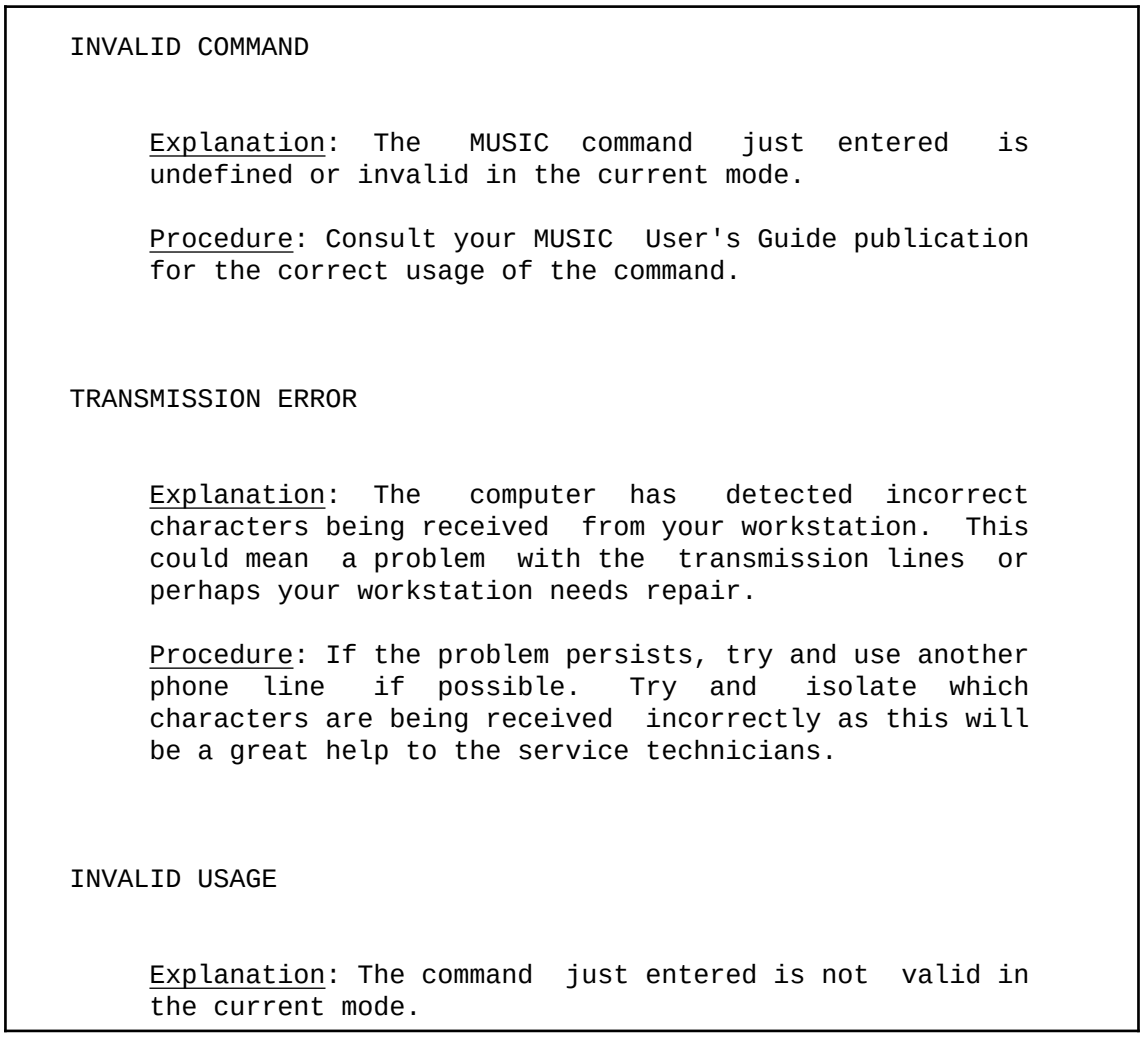
User Defines

When typing in a large document it is handy to use user-defined control words. Consistency throughout the document is easier to maintain this way; especially if more than one person is entering the text.

Figure 5.31 - Output Document

Styling Example

The following example shows how you can set up control words to help reduce the number of input characters required to produce complex output forms. In this case SCRIPT produces a list of error messages along with an explanation and a procedure for each one:



INVALID COMMAND

Explanation: The MUSIC command just entered is undefined or invalid in the current mode.

Procedure: Consult your MUSIC User's Guide publication for the correct usage of the command.

TRANSMISSION ERROR

Explanation: The computer has detected incorrect characters being received from your workstation. This could mean a problem with the transmission lines or perhaps your workstation needs repair.

Procedure: If the problem persists, try and use another phone line if possible. Try and isolate which characters are being received incorrectly as this will be a great help to the service technicians.

INVALID USAGE

Explanation: The command just entered is not valid in the current mode.

Figure 5.32 - Styling Example Output for User Defines

To produce the above, you can defines three special control words ..EM, ..EX and ..PR. These names do not have to mean anything but for convenience they can mean Error Message, EXplanation, and PRocedure.

```
.df ..em
.sk 3
.cp 4
.in 0
.nf
.up
.df ..ex
.sk 2
.in 5
.fi
.ul
Explanation:
.df ..pr
.sk 1
.ul
Procedure:
.df
```

Figure 5.33 - User Defines for Styling Example

Now that the definitions are setup, they can be used for typing the error messages. (These definitions can be stored in a separate file that other users could have access to.)

The following shows how the special control words are used. The ..em is typed before the error message, the ..ex is typed before the explanation part and the ..pr is typed before the procedure part.

```
..em
invalid command
..ex
The MUSIC command just entered is undefined or invalid
in the current mode.
..pr
Consult your MUSIC User's Guide publication for the correct
usage of the command.
..em
transmission error
..ex
The computer has detected incorrect characters being received
from your workstation.
This could mean a problem with the transmission lines
or perhaps
your workstation needs repair.
..pr
If the problem persists, try and use another phone line if
possible.
Try and isolate which characters are being received
incorrectly
as this will be a great help to the service technicians.
..em
invalid usage
..ex
The command just entered is not valid in the current mode.
```

Figure 5.34 - Input Document for Styling Example

Modification Flagging

In some publications, textual changes since the last edition are annotated with a vertical bar in the left margin. SCRIPT allows you to produce similar flagged changes. Modification flagging is initiated by entering FLAG SCRIPT when you create or begin editing your document with the Editor (see the example below). The Editor keeps track of the date on which modifications were made and will process this information. The flagging can be further controlled by the .MF control word as specified below.

This automatic modification flagging is one of the more attractive features of the SCRIPT facility.

Modification Flagging .MF

```
.MF ON
    OFF
    AUTO
```

The OFF parameter forces the modification flags off for the following lines until reset by another .MF. This would be used if there is a portion of your document which you do not want flagged. The ON option forces the modification flag to be set on for all future lines until reset by another .MF. AUTO sets the modification flag according to the date rules starting with the **next** input line. This option is in effect at the start of a document if the MOD output option is given to the SCRIPT program. If no option is given on the .MF control word, then the ON option is assumed.

SCRIPT Output Options for Modification Flagging

No flagging is done in the output document unless the MOD output option is specified when the SCRIPT program is run. The MODONLY output option can be specified to print only those pages that have been modified since a given date. (The ZONE option (.ZN) should be given in your input text to separate the modification date field from your text.) Refer to "SCRIPT Output Options" in the next chapter for additional information.

Example:

The following examples show how modification flagging actually works. Lines you enter are printed in bold and underlined. *Sample* is the name of the file used in the following figures. The FLAG SCRIPT Editor command is issued at the beginning of the edit session. In this case this file was created on May 22, 1989 (the 112th day of 1989). On May 23 the file was edited and a line added.

```
   SELECT OPTION ===> c sample;flag script

   *IN PROGRESS
   FLAG  89112   COL 73   DEL  .CM/

                             (press F11 and type document)
```

```
                                                  (cols 73-77)

.zn 72                                                  89112
.tm 0                                                   89112
This example will show how modification                 89112
flagging works.                                         89112
Each time this document is edited the date is           89112
included by the Editor program with every line          89112
that is changed or added.                               89112
The proofreader doesn't have to read the whole          89112
document every time there are changes.                  89112
```

Figure 5.35 - Creating a New File for Flagging

```
   SELECT OPTION ===> r sample;flag script

   FLAG  89113   COL 73   DEL  .CM/
```

```
.zn 72                                                  89112
.tm 0                                                   89112
This example will show how modification                 89112
flagging works.                                         89112
Each time this document is edited the date is           89112
included by the Editor program with every line          89112
that is changed or added.                               89112
The symbol "|" will appear beside each output line      89113
The proofreader doesn't have to read the whole          89112
document every time there are changes.                  89112
```

Figure 5.36 - Revising a Flagged Document

```
  SELECT OPTION ===> x sample

  MUSIC/SCRIPT...ENTER OPTIONS OR 'HELP'
  mod=89113
  LOAD PAPER, HIT RETURN
```

```
  This  example will  show  how  modification flagging  works.
  Each time  this document is edited  the date is  included by
  the Editor program with every line that is changed or added.
| The symbol  "|" will  appear beside  each output  line.  The
  proofreader doesn't  have  to read the  whole document every
  time there are changes.
```

Figure 5.37 - Executing a File with MOD SCRIPT Option

Conditional Section

The conditional section ability of SCRIPT allows you to include sections of text, in your output document, based upon a version number specified at the time the SCRIPT program is run.

Conditional Section .CS

```
.CS n1 n2 n3 n4     (etc)
```

Instructs SCRIPT to label a specific sections for different versions. These lines will then be processed if the version number is listed on the VERSION= option of SCRIPT. If the current version number is not one of those given, then the text will be ignored until another .CS control word is found. These skipped lines will not appear in the output document. The end of the section can be specified by a .CS with another version number or just .CS.

```
.CS NOT n1 n2 n3 n4  (etc)
```

The use of "NOT" with .CS above does the reverse of the .CS described earlier. This form of .CS causes the lines that follow to be skipped if the current version number is included. .CS does not imply a .BR.

Range Specification on the .CS Command

A range of version numbers can be given as in the following example:

```
.CS 1 3 5-10 2
```

The numbers and ranges may be specified in any order but the second number of the range must be larger than the first.

Example:

The following examples show what the input file looks like when using conditional sectioning, and how to execute the document to get each version.

```
.tm 0
.cs 1
This section will print if "VERSION=1" is specified
when the document is executed.
.cs 2
.sp
If "VERSION=2" is specified then this paragraph will
print.
.cs
.sp
To end a section use .CS by itself or .CS n to start
another one.
This last paragraph will print when either version is
specified.
```

Figure 5.38 - Input Document with Conditional Sections

```
     SELECT OPTION ===> x sample

      MUSIC/SCRIPT...ENTER OPTIONS OR 'HELP'

      version=1
```

Figure 5.39 - Executing the File Called Sample

```
This section will print if "VERSION=1" is specified when the
document is executed.

To end a section use .CS by itself or .CS n to start another
one.  This last paragraph will  print when either version is
specified.
```

Figure 5.40 - Output when VERSION=1 is Specified

```
If "VERSION=2" is specified then this paragraph will print.

To end a section use .CS by itself or .CS n to start another
one.  This last paragraph will  print when either version is
specified.
```

Figure 5.41 - Output when VERSION=2 is Specified

```
This section will print if "VERSION=1" is specified when the
document is executed.

If "VERSION=2" is specified then this paragraph will print.

To end a section use .CS by itself or .CS n to start another
one.  This last paragraph will  print when either version is
specified.
```

Figure 5.42 - Output when No Version is Specified

Output Auxiliary .OX

```
.OX n
```

Allows you to write certain records to an auxiliary file. The record contains the current page number and the single digit n (0-9) together with a copy of the first 68 characters from the next text line. This auxiliary file can be processed with the contents utility program to produce a Table of Contents. This command will have no effect if AUXOUT=0 option is used when the SCRIPT program is run. Read the section "Table of Contents" in *Chapter 6 - TODO Menu Items* for more information.

You will find it most convenient to use the following convention for the digit n. Use .OX 0 for major divisional headings such as a new section of a guide. Use .OX 1 for chapter titles. Use .OX 2 through 9 for progressively lower heading levels.

.OX is often used as part of a user-defined heading control word. This usage can result in the automatic generation of a table of contents item corresponding to the heading. The following illustrates the usage of .OX control words in heading definitions.

```
.cm the following defines heading 1
.df ..h1
.pa
.in 0
.fi
.up
.ul all
.ox 1
.cm the following defines heading 2
.df ..h2
.in 0
.fi
.sk 3
.cp 5
.up
.ul all
.ox 2
.df
```

Incorporating Other Files

You may want to incorporate other files or parts of other files into your current document. This would be helpful if you wanted to form a letter by selecting among several standard paragraphs previously entered, or if you wanted to include a standard set of user-defined control words in your document.

Imbed .IM

```
.IM name
```

Causes SCRIPT to temporarily stop reading the current file and start reading from a separate file called *name*. After this new file has been read, SCRIPT will return to reading the old file from where it left off. This control word is handy when doing form letters (refer to the section "Form Letters" later in this chapter.) The .IM control word does **not** imply a .BR.

Append .AP

```
.AP name
```

Similar to .IM except that SCRIPT will not return to reading the old file after the named file is finished. This control word can be used to repeat the processing of the current file. In this case, you would have to limit the number of pages SCRIPT is to produce so that the program wouldn't run indefinitely.

End File .EF

```
.EF
```

Indicates the end of a file. This control word can be inserted within a file if you want SCRIPT to stop reading the current file before the end. This is useful if you only want to process part of a file.

Workstation Interaction Control Words

Normally the SCRIPT output proceeds without any interaction from the workstation. In some cases, however, you may wish to interact with the SCRIPT program while it is running. You can do this with the following control words.

Read .RD

```
.RD n
```

Causes the output to be suspended until the user enters *n* lines. The contents of the lines entered have no effect on SCRIPT. This facility is useful if you want to enter names or addresses manually in a letter which is stored in SCRIPT format. .RD is ignored if the output is not directed to a workstation. It is also ignored if the workstation is a 3270-type. A .BR control word is implied by .RD.

Terminal Input .TE

```
.TE n
    ON
    OFF
```

Reads *n* lines from the workstation when the SCRIPT program is running. The lines entered are taken as if they were part of the SCRIPT input text, so you may type in any valid SCRIPT control word or text. .TE ON reads from the workstation until a .TE OFF is entered. When .UL MASK is in effect SCRIPT will always read one line ahead.

Any .TE command (other than .TE OFF) entered during the input for the first .TE, will change the number of lines read by the first .TE. This means that the first .TE, terminating the first .TE and starting a new one.

Caution: SCRIPT may print output at any time and therefore .TE is mainly used when output is not being directed to the workstation.

Terminal Type .TY

```
.TY text
```

Causes the text portion associated with it to print on the workstation when the control word is being processed. The main use of .TY is to prompt you about what to enter when input is requested by a .TE. The following illustrates the use of the .TE and .TY.

```
This copy printed for
.ty enter employee name
.te
and is for your own personal use only.
```

Equivalent Functions

This section describes some SCRIPT control words that perform the same functions as others do. These control words are supported by SCRIPT so that users of other SCRIPT systems, including the older MUSIC versions can run their old SCRIPT files. New users of SCRIPT can ignore this section.

Adjust .AD

```
.AD
```

Equivalent to .RI except that the number of lines affected is not specified on the control word. The effect simply continues until it is turned off by one of the following SCRIPT control words:

.CE, .CO, .FI, .FO, .NF

Format .FO

```
.FO
```

Identical to .FI.

Stop .ST

```
.ST
```

This control word is identical to .RD.

Offset .OF

```
.OF n
```

Specifies that the output is to indented by *n* columns. Once the next output line has been entered .OF adds the value n on to the current indent. .OF is similar in effect to .IN and .UN but you will probably find it more complicated to use. A .BR is implied by this control word.

Heading .HE

```
.HE text
```

Equivalent to the control specification of

```
.TT //text/PAGE %/
```

SCRIPT will not read "PAGE &" if .PN OFF is in effect at the time .HE is processed.

Automatic Letters and Memos

The CREATE item on the TODO menu includes a facility for an automatic generation of shell documents for letters and memos. The default shells can be altered by the systems administrator at your installation or by yourself. Refer to Appendix F for information about modifying the CREATE item.

The figures below illustrate the default setup for letters and memos included with CREATE.

The file names that you choose indicate whether you want to make use of this facility or not. By starting your file name with ".L", CREATE will invoke the Editor and present you with the shell for a letter. If your file name starts with "M.", then you will receive the memo shell.

To begin a letter or memo, choose the TODO item "C" in the SELECT OPTION area. (Your cursor is positioned in this area when the TODO menu is displayed.)

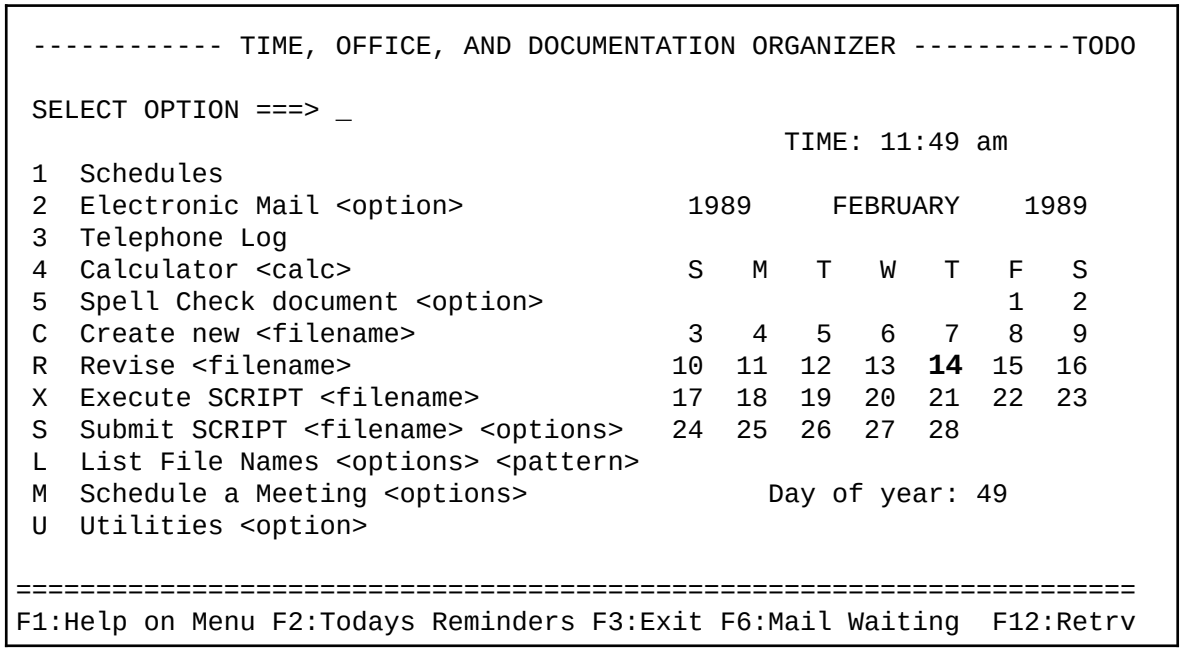


Figure 5.43 - TODO Menu Screen

In the following figures a shell has been generated for a letter and a memo going to a person called "Kathy".

Letter

When item "C" on the TODO menu is chosen with the file name "l.kathy",

```
SELECT OPTION ===> c l.kathy
```

then the following file is presented on the screen:

```
.tm 0
.ll 70
.sp 15
.bm 6
.tt(
.sp 5
Page &
.br
February 21, 1989
.sp 4
.)
.in 0
February 21, 1989
.sp 4
.nf
.cm ===> Type in name and address following this <===
.sp
Dear
.nf
.co
.df ..par
.sp
.cp 2
.df
.cm ===> Type ..par on a separate line before each  <===
.cm ===> paragraph, after the first paragraph,      <===
.cm ===> to skip one line and to make sure that the <===
.cm ===> page can hold at least 2 lines of paragraph<===
..par
.cm ===> Type in main body of letter following this <===
.sp 2
.(
.in 0
.nf
Yours truly,
.sp 4
.cm ===> Type in sender's name and title after this <===
.sp 3
/kw
Encl:
```

Figure 5.44 - Automatic Setup for a Letter

Fill in the text that is missing: name and address, body of the letter, and sender's name and title. Issue the Editor command FILE when finished.

The current date is automatically included in this letter and the initials at the bottom of the letter are inserted according to your userid.

Any changes to the shell of this letter can be made on the screen or the CREATE program can be adjusted to create a letter that suits your needs better. Also included with this program is an automatic generation for return address's and sender's information. The name of your file indicates to the program that you want this information included.

For example, a letter file (L) going to a company with the initials INC from a person with the initials ME would have a file name that looks like this:

```
L.INC.ME
```

Similarly, to create a memo (M) file to INC from ME you would use a file name that looks like this M.INC.ME. You can instruct CREATE to recognize the initials and fill in the full names when it sets up the file for you.

Appendix E at the back of this manual lists the CREATE program and indicates in bold the areas that can be changed by system support personnel at your installation.

Memo

The following illustrates the automatic setup for memos. CREATE generates the current date automatically along with your initials. Please read the description for LETTER described above for more information.

```
.tm 0
.ll 70
.sp 5
.bm 6
.tt(
.sp 5
Page &
.sp
February 21, 1989
.sp 5
.)
.in 0
.nf
_____________________________________________________________________________


                M E M O R A N D U M        Date: February 21, 1989

_____________________________________________________________________________
.sp 4
.cm ===> Fill in the following lines as required <===
To:                             From:


.sp
Subject:
.sp 3
.nf
.co
.cm ===> Type in main body of memo following this <===
.sp 4
/kw
```

Figure 5.45 - Automatic Setup for a Memo

Form Letters

The following examples show how to enter files that will be used for form letters. In the first example there is one file for the addresses and one file for the letter. It will be the address file that is executed. The letter will be imbedded by SCRIPT for each person. In order to have the letters continuous, a .PA and the start of the next letter is included at the end of the letter file.

As little as possible should be typed in the address file. Make sure that all SCRIPT control words and text are included in the letter file. The beginning of the letter must be typed in the address file only once. After that, all you need to change is the names and addresses.

```
.ri
May 1, 1981
.sk
.nf
Mrs. B. Black
876 Crescent Ave
Westmount, Que.
.im letter
Mr. G. Green
999 Green Ave
N.D.G., Que.
.im letter
Mr. T. Thomas
877 Mountain St.
Montreal, Que.
.im letter
```

Figure 5.46 - Input Document, File Called ADDRS

```
.sk
Dear Customer,
.sk
.fi
We would like to inform you that your name has been
removed from our mailing list.
If you still wish to receive our publication each month
we are asking that send a money order for $12.95 to cover
expenses for the year.
.sk
We hope to be hearing from you soon.
.nf
.sk 2
.in 40
Sincerely,
.sk 2
Mr. W. Morgan
ABCD Magazine
.in 0
.pa
.ri
May 1, 1981
.sk
```

Figure 5.47 - Input, File Called LETTER

```
                                                  May 1, 1981

Mrs. B. Black
876 Crescent Ave
Westmount, Que.

Dear Customer,

We would like to inform you  that your name has been removed
from our  mailing list.   If you still  wish to  receive our
publication each month we are asking that send a money order
for $12.95 to cover expenses for the year.

We hope to be hearing from you soon.


                                        Sincerely,


                                        Mr. W. Morgan
                                        ABCD Magazine
```

Figure 5.48 - Output Form Letter

Chapter 6. TODO Menu Items

Executing SCRIPT Documents - TODO Item X

When you want to print a file ask TODO to execute a document for formatting. To do this, type an X and the filename on the SELECT OPTION line of the TODO menu, then press the ENTER key. (Filename is the name of your document.)

```
------------ TIME, OFFICE, AND DOCUMENTATION ORGANIZER ---------TODO

SELECT OPTION ===> X filename
                                                TIME: 11:49 am
1  Schedules
2  Electronic Mail <option>                1989     FEBRUARY    1989
3  Telephone Log
4  Calculator <calc>                        S   M   T   W   T   F   S
5  Spell Check document <option>                                   1   2
C  Create new <filename>                    3   4   5   6   7   8   9
R  Revise <filename>                       10  11  12  13  14  15  16
X  Execute SCRIPT <filename>               17  18  19  20  21  22  23
S  Submit SCRIPT <filename> <options>      24  25  26  27  28
L  List File Names <options> <pattern>
M  Schedule a Meeting <options>                  Day of year: 49
U  Utilities <option>

=====================================================================
F1:Help on Menu  F2:Todays Reminders F3:Exit  F6:Mail Waiting F12:Retr
```

Figure 6.1 - TODO Menu with X Item Selected

After this item is selected then the SCRIPT program will take control and ask you to enter options.

Example:

In this example the X item is chosen with the file name *sample*.

```
SELECT OPTION ===> x sample

MUSIC/SCRIPT...ENTER OPTIONS OR 'HELP'
?
```

Figure 6.2 - SCRIPT Prompts for Options

OPTIONS refer to the special output options which SCRIPT uses to format your document. (These options for formatting are in addition to SCRIPT control words that are imbedded in your file.) These options are explained in the next section "SCRIPT Output Options".

Note: Because a display workstation cannot show a full page of text on one screen, you should only check the executed document for general formatting and SCRIPT errors. A printed copy on paper is better for detailed proofreading.

Prior to printing the document, you may want to select item 5 SPELL CHECK from the TODO menu.

SCRIPT Output on a Hardcopy Workstation

If you are printing documents on a hardcopy workstation, you need to sign on to MUSIC first. Instead of printing the whole TODO menu on paper, MUSIC prints the following message:

```
ENTER SELECTION, "HELP TOPIC", "MENU", or ("END, "OFF)
```

ENTER SELECTION	Type an item from the TODO menu. To execute a SCRIPT document choose item X.
HELP TOPIC	By typing HELP you will receive general help on MUSIC.
MENU	If you want to see the TODO menu, then enter type MENU.
END	Terminates the TODO facility and gives you the *GO message of MUSIC.
OFF	Terminates your MUSIC session (signs off).

Example:

The following figure illustrates the steps necessary to execute a file called SAMPLE on a hardcopy workstation.

```
ENTER SELECTION, "HELP TOPIC", "MENU", or ("END", "OFF")
x sample

*IN PROGRESS

MUSIC/SCRIPT...ENTER OPTIONS OR 'HELP'
?

LOAD PAPER; HIT RETURN




    This sample document illustrates what a formatted SCRIPT
document  looks like  when it is executed.   Notice that the
text  is justified and  both the left and right  margins are
even.  This is because .FI is the default for SCRIPT.
    If you wish to have it unformatted, as in the  following
address, then use .NF for No Fill.

Mrs. John Smith
1004 42nd Ave.
New York, NY
10095
```

Figure 6.3 - Executing a SCRIPT Document on Paper

After typing your SCRIPT options and pressing the RETURN key, you will see a message from SCRIPT as shown in the figure below, asking you to load the paper it needs to print the hardcopy. You can use your choice of paper: bond, letterhead, etc.

Press the RETURN key to inform SCRIPT to print the file.

After each page is printed, SCRIPT will skip to the bottom of the page and stop. This allows you to insert another blank sheet of paper. When the last page is printed, press the RETURN key, then you will return to the TODO menu.

Incorporating Several Files for Execution

If you have a sequence of several files that you want to print together with one SCRIPT run, use the .IM control word to join them. See the section "Incorporating Other Files" in the previous chapter on SCRIPT for more information.

The figure below shows the file *guide.print* which consists of only .IM control lines.

```
.im guide.intro
.im guide.chap1
.im guide.chap2
.im guide.chap3
.im guide.concl
```

Figure 6.4 - Imbedding Several Files

There are two other ways to incorporate files using .IM:

1. Include the .IM control words at the end of the first file.
2. Use the .IM control word at the end of each file.

SCRIPT Output Options

When you execute or submit a SCRIPT document, you will get the message:

```
MUSIC/SCRIPT...ENTER OPTIONS OR HELP
?
```

SCRIPT will do different things depending on which option you choose when program is run. You can respond in one of three ways:

1. Type HELP to get a list of the options available.

2. Enter a blank line if no options are required.

3. Enter the options you desire, separated by commas. If a line ends with a comma, options are assumed to continue on the next line.

Output Options

ALLUP	Prints all lower case letters printed as upper case letters.
NOPAUSE	Informs SCRIPT not to pause at the bottom of each page to allow a new sheet of paper to be inserted. Choose this option if you are using continuous forms at your hardcopy workstation. This option is ignored if the output is directed to a remote printer or a 3270-type workstation.
FIRSTPG=n	Specifies the page number printed on the first page. Successive page numbers are automatically incremented by 1. This option can be overridden by the .PA control word. The start page number cannot be specified by this option as anything other than a whole number in the sequence 1 2 3 4 (etc). The default is n=1.
STARTPG=n	Starts printing when SCRIPT encounters the specified page number. If n=0, this option is ignored. Negative numbers can be used to start the document at a specified Roman numeral. For example, STARTPG=-5 starts printing at page number "v". Use this option even if no page numbering are in the document.
LINELEN=nn	Specifies the length of each output line in characters. The default setting is LINELEN=60. This option has no effect if .LL in included in your document.
PAGELEN=nn	Specifies the length of each output page in lines. The default setting is PAGELEN=66. If .PL is included in your document then this output option has no affect.
DOUBLESP	Works the same as the .DS control word and causes the output to be double spaced at this time.
MARK	Underlines the first non-blank character found on each input text line. This may be useful as an aid to locate specific lines when when using the Editor to make document corrections.
OKERR	Causes SCRIPT to continue printing after an error condition. An error message will

occur at that point in your output and a message may also be printed referring to the line number. This option is very useful for printing rough drafts, because it allows you to see your whole document even if there are control word errors. Without this option SCRIPT stops printing at the first error.

MOD=nnnnn — Activates the SCRIPT modification flagging feature. All input lines having a change date of *nnnnn* or higher are flagged. The nnnnn field is the five digit date indicating the year and the number of days. For example, 89111 indicates the 111th day of the year 89. The .ZN control word must be used to identify the location of the date flag in the input text. If there are not at least 5 characters beyond the ZONE column, the line is considered modified. If the characters in the date field are not numeric, the line is also considered modified. The change flag " prints as the first character on the output line if the remainder of the line contains modified text. When this option is in effect all the output document shifts to the right by two spaces to allow for the change flag. If a changed line is a control word or a blank line, the change flag prints on the next output line.

The change flags are ignored on lines that define user-defined control words.

MODONLY — Prints only pages that contain modified text. If a title is modified, then the first occurrence prints an unmodified page. This option works with the MOD= option. The STARTPG= option may be used in conjunction with this option to print modified pages occurring after a specific page number.

VERSION=n — Specifies the version number used with the .CS control word. No specification or VERSION=0 prints all conditional sections.

AUXOUT=n — Defines an auxiliary output unit number for storing the output of the .OX control word. The specification of 0 will ignore all such output. The default is AUXOUT=0.

Note: This option is not necessary when using the Table of Contents item from the Utility menu. Read the section "Table of Contents" in *Chapter 7 - Utilities*.

EVENLM=n
ODDLM=n — Specifies a left margin of *n* spaces if the output is directed to a remote printer. If the EVENLM and ODDLM are the same, the output document is uniformly shifted to the right when it is printing. When creating masters for a manual printed on both sides of a page, specify the odd margin greater than the even. These options are ignored if the output is being printed on a workstation. The default value for both options is 0.

REALPG
NOREALPG — Specifies that a '1' carriage control to skip to the top of each new page. This is the default when running SCRIPT from batch or a 3270-type workstation. This is ignored when the output is directed to any other type of workstation. NOREALPG uses the PAGELEN= option to insert the required number of spaces to get to the next page.

COMPRS
NOCOMPRS — COMPRS uses a blank, zero (0) or a minus (-). carriage control for single, double or triple spacing. NOCOMPRS uses only blank and skip to new page carriage controls. This is not useful if output is directed to a remote workstation.

INPUT=n — Unit number from which the input text is to be read; *n* must not be 3 or 4. The default is n=5.

OUTPUT=n — Unit number where the output document is written. If *n* is other than 6 or if the job is run from batch, each output line will has a printer carriage control in the first position. If n=0 is specified, no output document is produced although the input text is processed and any

error messages are printed. The default is n=6; *n* must not be 3 or 4.

Unit Numbers

Unit numbers define what devices are to be used for input or output. For example, unit number 6 is how output is directed to the workstation. MUSIC/SCRIPT and its utility programs can read and write from many different units just by specifying which number to use. The unit numbers are detailed below.

6 Unit 6 is the printer. Specifying this number directs the output to your workstation. From batch, it will cause the output to be directed to the high speed printer.

7 Unit 7 is the card punch. This unit number is rarely used.

5 Unit 5 is always used to read files.

10 Unit 10 is a temporary hold area for approximately 400 output lines. You can save the output from this unit with the MUSIC command "/SAVE name,SV" from the TODO menu.

1-4 Units 1,2,3 and 4 refer to files containing large amounts of information. These types of files are *User Data Set Files* (UDS). You always need a /FILE control statement before you can use any of these numbers. Refer to the *MUSIC/SP User's Reference Guide* for information about /FILE statements.

Interrupting the SCRIPT Program

Once SCRIPT has started processing your data, you may interrupt the program at any time by "breaking" the session. Use the PA1 key if your workstation is a 3270-type and the BREAK key if it is a TTY-type machine.

Type in any of the commands listed below:

/CANCEL (or /CAN) This command immediately stops your program, and returns to the TODO menu.

/SKIP n Skips past output lines you do not want to see displayed at your workstation. The *n* indicates the number of lines you wish to skip.

/SKIP ALL Skips to the start of the next page when the SCRIPT program is run using the PAUSE option, otherwise it skips to the end of the file.

Submitting SCRIPT Documents - TODO Item S

For submitting SCRIPT documents to remote printers use the TODO item:

```
S  Submit SCRIPT <filename> <option>
```

S Submit SCRIPT — "S" (Submit SCRIPT) is the TODO selection for sending SCRIPT documents to remote (batch) printers. This item invokes both the SCRIPT program (for formatting) and the SUBMIT program (for submission). The SUBMIT program is described later in this chapter.

<filename> — Name of your MUSIC file containing your SCRIPT document. If you leave out the file name then you will be prompted for it.

<options> — The options that are available are described in the section "Submission to MUSIC Batch" in this chapter. For SCRIPT documents the following keyword options are useful:

```
R(location)    Route
FOR(xxxx)      FORms
COP(xxx) COPies
```

ROUTE specifies the destination (location of the batch printer). If you do not specify "route", your document will be sent to the default destination at your installation. The defaults for FORMS depends on your installation. If COPIES is not specified then 1 copy of your document will print.

Example:

The following figure illustrates the screen display for the TODO facility with the S item selected for a file called SAMPLE and the copies option.

```
------------ TIME, OFFICE, AND DOCUMENTATION ORGANIZER ---------TODO

SELECT OPTION ===> s sample copies(3)
                                                TIME: 11:49 am
1  Schedules
2  Electronic Mail <option>              1989      FEBRUARY     1989
3  Telephone Log
4  Calculator <calc>                      S   M   T   W   T   F   S
5  Spell Check document <option>                                  1   2
C  Create new <filename>                  3   4   5   6   7   8   9
R  Revise <filename>                     10  11  12  13  14  15  16
X  Execute SCRIPT <filename>             17  18  19  20  21  22  23
S  Submit SCRIPT <filename> <options>    24  25  26  27  28
L  List File Names <options> <pattern>
M  Schedule a Meeting <options>                Day of year: 49
U  Utilities <option>

=====================================================================
F1:Help on Menu  F2:Todays Reminders F3:Exit  F6:Mail Waiting F12:Retr
```

Figure 6.5 - Selecting the Submit Item on the TODO Menu

Example:

The following shows the messages that appear when instructing the system to send a SCRIPT document for batch printing. In this sample the file name is "myfile" and the route is "room1".

```
SELECT OPTION ===> s myfile r(room1)

MUSIC/SCRIPT...ENTER OPTIONS OR 'HELP'
?

SUBMITTED...

PRESS ENTER TO CONTINUE....
```

Figure 6.6 - Submitting a SCRIPT Document

For information about submitting jobs other than SCRIPT documents, see "SUBMIT" program in the *MUSIC/SP User's Reference Guide*.

Printing Files

The PRINT command is used to send documents to remote printers. Files submitted using this command will not be executed (formatted). SCRIPT documents will print out the way they were typed in, with control lines and text lines. Files in your Save Library print on any of the line printers defined in the system by using this PRINT command. The PRINT command is issued on the SELECT OPTION line of the TODO menu as follows:

```
/PRINT filename R[location] CC
```

Use this command from the Editor too, (remember Editor commands do not use slashes). The information within square brackets is optional. They are described as follows:

filename	The name of the file to print. Under the Editor, use the special name "*cur" to indicate the current contents of the file being edited.
location	The routing location name of the printer where the file prints. If not specified, the file prints at a default location designated by your installation. Consult your installation for default and other valid locations.
CC	Indicates the file to print already contains a carriage control character as the first character of each line in the file.

The PRINT program copies the file to the print queue for subsequent printing when the specified printer is available.

Library - List File Names - TODO Item L

This command is used to produce a list of all the file names that you own. It can also just produce a list of those that start with certain letters. The editor can be called to allow you to further examine the list. If no options are given, then a list of all the file names will be produced.

Typing "L SCAN L.ABC" will just show the files that start with the characters "L.ABC". You can use any other characters you want instead of the "L.ABC" shown in this example. For example "L SCAN M." is valid. (Do not enter the " marks though!)

Typing "L EDIT L.ABC" will do the same as the above SCAN example except that the editor will be called to allow you to further examine the list.'

Typing "L SCAN L.ABC FULL" or "L EDIT L.ABC FULL" will show the date each of the selected files was referenced and last changed. The information listed will be the same as that shown with the MUSIC /LIBRARY command. Refer to the /LIBRARY command in the MUSIC COMMAND section of *Chapter 1 - Introduction* for details.

Example:

Select from the TODO menu item "L". A list of all your documents will be displayed in alphabetical order.

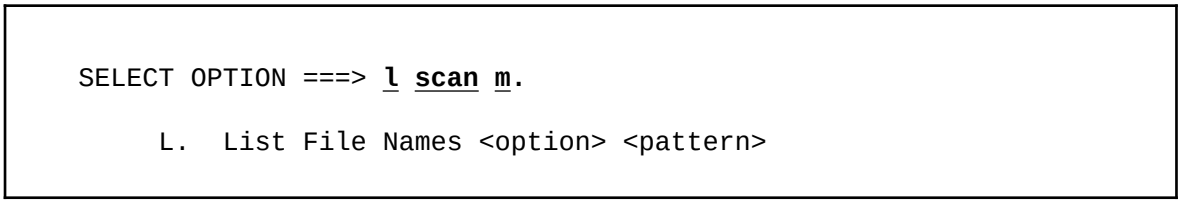


Figure 6.7 - Selecting List File Names from TODO Menu

<option>

Options must be either SCAN (S), EDIT (E), or HELP (H).

SCAN	Display a alphabetical list of all files under the user's userid. If a "pattern" is supplied, then only files with this pattern are listed.
EDIT	Same as above but store the list in the file called @LIB and start an edit session for that file.
HELP	Help Function
(none)	Implies SCAN.

The words SCAN, EDIT and HELP can be abbreviated S, E or H. Use option "H" for help information. Example:

```
SELECT OPTION ===> l h
```

Schedules - TODO Item 1

The SCHEDULE program offers you a way to organize your agenda on a daily or monthly basis. Also, use it to schedule conference rooms or to keep track of equipment.

To use SCHEDULE, type a 1 (one) on the SELECT OPTION line of the TODO menu and press ENTER. From *Go mode, type SCHED.

With this program, you can give other MUSIC users authorization to look at your calendar and update it too. For example, a person whom you have authorized could check and then schedule a meeting during a time when your calendar says you are free.

Many levels of authorization are available. You can authorize one person to view all item descriptions on your calendar, while another person can view all descriptions except personal or confidential ones. For more information about this, refer AUTHORIZE/DE-AUTHORIZE ACCESS later in this section.

How to use the Process Calendar

The following figure illustrates what the screen looks like after selecting SCHEDULE from the TODO menu. In this example the MUSIC user's name is Fred and the current date is February 19, 1989.

```
------------------------- Process Calendar ---------------------------

 Calendar for: Fred
        Date: 02 / 19 / 1989   (mm/dd/year)
                                                  TIME: 4:50 pm
  Select one of the following options:
                                        1989    FEBRUARY   1989
  PF2  Look at or change Calendar
  PF4  Select Equipment/Conference Room     S   M   T   W   T   F   S
  PF5  Change to the next day                                   1   2
  PF6  Change to the previous day           3   4   5   6   7   8   9
  PF7  Look at the whole month             10  11  12  13  14  15  16
  PF8  Display / Print Daily Calendar      17  18  19  20  21  22  23
                                           24  25  26  27  28
 PF10  Change to the next month
 PF11  Change to the previous month              Day of year:  49
 PF12  Authorize/De-authorize Access

 F1:Help                                                 F12:Return
```

Figure 6.8 - Process Calendar Display

The name following **Calendar for:** is the person, conference room or equipment item whose calendar this is. The date following **Date:** indicates which daily calendar for that person, conference room or equipment item. When the screen is first shown, these two fields will display your name and the current date.

Note: Your userid is shown in the "Calendar for:" field if your name was not added in the Mail Profile. See the "Mail Profile" topic in *Chapter 4. Electronic Mail* for information about adding your name.

To select a different calendar (for another person, conference room or equipment item), type over the name in the "Calendar for:" field. Remember, to get the cursor from one field to the next use the NEW LINE key or the TAB key. The name can be:

a. The userid of the person whose calendar you wish to view.
b. A nickname which represents the user. This must be set up previously with the Mail Directory facility. (Refer to the DIRECT program.)
c. The name of a Conference room or Equipment item. This needs to be set up by the person authorized to initialize Conference rooms and Equipment items. See Appendix D for information.

If this name field is set to all blanks, your name will appear again.

To change the date, type the new date over the date shown. Another way to change the date is to use program function keys 5, 6, 10 or 11. If any of the date fields are blank, the current date appears in the the date field.

When the "Calendar for:" and the "Date:" choices are correct, press an appropriate function key to continue using SCHEDULE. For example, F2 will display the "Look at or change Calendar" screen.

Function Key Definitions for the Process Calendar:

Descriptions for function keys 2, 4, 7, 8, and 12 are described in more detail on separate pages.

F1 HELP
Provides help information on SCHEDULE.

F2 LOOK AT OR CHANGE CALENDAR
If you chose your own calendar then you can view and update the items displayed on your agenda. If the calendar is for another person, you can view and/or update the agenda, depending on your authorization level. Refer to LOOK AT OR CHANGE THE CALENDAR later in this section.

F4 SELECT EQUIPMENT/CONFERENCE ROOM
Displays a list of Equipment items and Conference rooms from which you can make a selection. Any of those items for which you are authorized will be shown.

F5 CHANGE TO THE NEXT DAY
Changes the date to the next day. The date highlighted on the monthly calendar shown on the right of the screen changes to match the new date selected.

F6 CHANGE TO THE PREVIOUS DAY
Performs the same function as PF5, but the previous day is chosen.

F7 LOOK AT THE WHOLE MONTH
Displays the selected Calendar for the entire month, not just for a single day. The month and year displayed disappears from the date field.

F8 DISPLAY / PRINT DAILY CALENDAR
Displays a Print Calendar screen, where you can choose the destination name for printing the calendar and how many copies you want. The calendar may also be displayed on your terminal.

F10 CHANGE TO THE NEXT MONTH
Changes the month in date field to the next one. The monthly calendar on the right of the screen will be changed to match the new date.

F11 CHANGE TO THE PREVIOUS MONTH

Performs the same function as PF10 but the previous month is chosen.

F12 AUTHORIZE/DE-AUTHORIZE ACCESS
A screen will be presented on which you can specify who is authorized to process your Calendar.

How to use Look at/Change the Calendar (F2)

In the following figure displays the LOOK AT/CHANGE THE CALENDAR screen that appears when F2 is pressed from the PROCESS CALENDAR screen. Two sample items for "Fred's" calendar are illustrated.

```
                    Look at/change the Calendar

  Calendar for: Fred                              Date: 02 / 18 / 1989

   Begin       End          Description
   9 am        11 am        Meeting with the boss
   12 pm       1 pm         Lunch









F1:Help   F2:Add Line   F3:End/Save  F4:Print  F5:Next Day  F6:Prv day
F7:Previous  F8:Forward F10:View Creator F11:View Mth F12/A1:End/No Sv
```

Figure 6.9 - Look at/Change Calendar Display

This screen shows you the day's schedule for a particular person, Conference room or Equipment item. Based on your authorization level, you can VIEW, ADD, or UPDATE the calendar.

If your authority level is VIEW, then you can not add or change anything. If your authority is ADD, you can add new items to the calendar and also change/delete any item you previously added. Entries you added are highlighted, unless it is your daily calendar.

With UPDATE authority, you c change/delete any item on the calendar. You can also add new items and notes.

The fields **Begin, End** are used to specify the starting and ending times for the day's events. **Description** describes the actual event.

Entering Times

Times can be entered in either 12 or 24 hour format. If you use the 12 hour format you must observe the following:

Unless you specify, the system assumes all times are in the morning. For example, if BEGIN time is 8 and END time is 10 the system interprets this as being 8:00 am and 10:00 am, respectively. Likewise, a beginning time of 3 and an ending time of 4 would be 3:00 am and 4:00 am, not 3 and 4 in the afternoon.

Specify noon in any of the following ways:

1200
12 pm
noon

Specify midnight in any of the following ways:

0
12 am
2400
midnite

You can add entries to the calendar in any order. When you press Enter, the system will automatically sort them into time sequence.

You can create a "to do" List by specifying no Begin and End times, and have only entries. in the Description field. This list must be the first set of entries on your calendar.

Entering Descriptions

Descriptions show the purpose of this event. To enter multiple lines of description, leave the Begin and End times blank, and enter additional comments on the following lines.

To make items personal, type **pers:** as the first part of the description entry. When you make an item personal, only individuals with the proper authority can see the description.

To make an item confidential type **conf:** as the first part of the description entry. When you make an item confidential, only individuals with the proper authority can see it.

You can add **notes** to the bottom of your calendar by typing the word **notes** in the Begin time field, then typing the information in the description fields. You can enter multiple lines. These notes show at the bottom of the calendar. The first two lines of the notes also appear on the monthly calendar. You cannot add any notes to another person's calendar.

Deleting Items

To erase an item type in a D (for delete) in the first position of the Begin time for the item you wish to remove. If D is entered on the first line of an entry (with a Begin / End time), the entire entry is removed. If this item has a multiple line description, all lines are deleted.

If D is placed on the description line of a multiple line entry, only that line is deleted.

Items in the "to do" list are treated as separate entries, so the D removes only a line at a time.

Moving Items

To move an item to another date, type "M" in the first position of the Begin time for the item you wish to move. Change the date entry to reflect the date this entry should be moved to. Press ENTER and the entry is moved to the date specified.

Items in the "to do" list are treated as separate entries, so the "M" moves only that specific entry to the "to do" list of the new date.

During the MOVE function, the selected entries are removed from the current date and "moved" to the new date. If you CANCEL the save operation on the new date (press PF12), then the entry is lost.

Copying Items

To copy an item to another date, type "C" in the first position of the Begin time for the item you wish to copy. Change the date entry to reflect the date this entry should be copied to. Press ENTER and the entry is copied to the date specified.

Items in the "to do" list are treated as separate entries, so the "C" copies only that specific entry to the "to do" list of the new date.

Recurring Items

To set a recurring event, type "R" in the first position of the Begin time for the item that will be scheduled on a recurring basis. A date entry screen will be displayed to allow entry of when the event will be scheduled.

For example, you could schedule an appointment for the first Monday of each month, between today and the end of the year.

Changing the Date

To select a calendar for a different day either type over the date or use function keys 5 or 6.

If any of the fields of the date are blank, then today's daily calendar appears.

Function Keys for Look at or Change Calendar:

F1 HELP
Provides help information for looking and changing the calendar.

F2 ADD LINE
Position the cursor on the screen where you wish to add a blank line. A blank line is placed after the line where the cursor is positioned. To add a blank line as the first line, place the cursor on Begin and press F2.

F3 END/SAVE
Accepts and files any changes you made on this calendar. If you selected another date, by typing over the date field, the new daily calendar shows. Otherwise you return to the initial PROCESS CALENDAR screen.

F4 PRINT
Allow for printing of the detail calendar information to a printer.

F5 NEXT DAY
Displays the next day's daily calendar. If any changes were made to the currently displayed Calendar, they are filed before the next day's calendar is displayed.

F6 PREVIOUS DAY
Same function as F5, except the previous day's calendar is displayed.

F7 PREVIOUS SCREEN

Goes to the previous screen of the daily calendar. If at the first screen, no function is performed.

F8 NEXT SCREEN
Goes to the next screen of the daily calendar. If there are no additional screens, a message is displayed.

F10 VIEW CREATOR
Displays the userid and date of the item that placed on your calendar. Position the cursor on the item in question and press F10 to see who it is. If you have authority, you can view the creator of a calendar that is not yours (see Authorize/De-authorize Access later in this section).

F11 VIEW MONTH
Shows the schedule for the Whole month. This is based on the date selected in the date field at the top of the screen. Any changes are filed before displaying this schedule.

F12 END/NO SAVE
No changes you made to this daily calendar are filed. If a new date was entered the daily calendar for that date is displayed. If no new date was entered, the initial PROCESS CALENDAR screen is displayed.

PA1 END/NO SAVE
No changes you made to this daily calendar are filed.

How to use Select Equipment/Conference Room (F4)

The following figure illustrates the screen display that appears when F4 is pressed from the PROCESS CALENDAR screen. This shows two examples of item entries.

```
-------------------- Conference Room/Equipment List -----------------

You are authorized to view, add or update the following items.  Move
the cursor to the item you wish to select and press ENTER.

  Item Name       Description                              Access type
  Room95          Seminar Room                             Add
  T1245           Laptop Computer                          Update









 F1:Help        F3:Return        F7:Previous Screen     F8:Next Screen
```

Figure 6.10 - Conference Room/Equipment List Screen

This screen displays all the names of items available for scheduling. These items are chosen by using the Conference and Equipment List program described in Appendix D. This program is restricted to MUSIC users who have the proper authorization.

The "Access Type" for each item indicates the your level of authorization. The following codes will appear

in this column.

View means you can view the item's calendar.
Add means you can add to the item's calendar.
Update means you can add or change the item's calendar.

Note: If you have the ADD authorization level, you can change any additions you made to the calendar.

How to Use Look at the Whole Month (F7)

The following illustrates the screen display showing a whole month. The user's name is Fred and his schedule is displayed here.

Note: In order to fit this figure on the page, only nine positions (dots and other characters) are shown for each day. On your workstation there are ten positions representing each hour.

```
                          Month at a Glance
Schedule for: Fred                                   Date: 02 / 19 / 1989
  Sunday    Monday    Tuesday  Wednesday  Thursday  Friday   Saturday
+---------+---------+---------+---------+---------+---------+---------+
|         |         |         |         |         |  1      |  2      |
|         |         |         |         |         |.........|.........|
+---------+---------+---------+---------+---------+---------+---------+
|  3      |  4      |  5      |  6      |  7   <  |  8      |  9      |
|.........|.s**.....|.f.......|.1.......|....**...|....**...|.........|
+---------+---------+---------+---------+---------+---------+---------+
| 10      | 11      | 12      | 13      | 14      | 15      | 16      |
|.........|.........|.**...**.|.......4.|....**...|......**.|.........|
+---------+---------+---------+---------+---------+---------+---------+
| 17      | 18      | 19      | 20      | 21   >  | 22      | 23      |
|.........|.****....|.........|*********|.......**|.***.....|.........|
+---------+---------+---------+---------+---------+---------+---------+
| 24      | 25      | 26      | 27      | 28      |         |         |
|.........|.....*...|.........|.........|.........|         |         |
+---------+---------+---------+---------+---------+---------+---------+
F1:Help       F2:Look at Day Detail     F3:Return         F5:Next Day
F6:Previous Day     F7:Print     F10:Next Month     F11:Previous Month
```

Figure 6.11 - Month at a Glance Screen

This screen provides a quick look at what you scheduled in your calendar for the month. The display format is:

```
xx     < >

1234567890
```

xx is the day of the month.

< displays something scheduled prior to 8:00 am.

> displays something scheduled at or after 6:00 pm.

1234567890 Each position represents an hour, starting at 8:00 am and going to 5:00 pm. The character displayed uses one of those shown below. Each hour is divided into 15 minute intervals. This allows you to quickly see when you have something scheduled and what your free time is.

.	Hour is free
*	Entire hour scheduled
1	1st quarter hour
2	2nd quarter hour
3	3rd quarter hour
4	4th quarter hour
f	1st half hour
s	2nd half hour
k	1st & 3rd quarter hour
l	1st & 4th quarter hour
o	1st, 2nd & 3rd quarter hour
n	1st, 2nd & 4th quarter hour
m	1st, 3rd & 4th quarter hour
h	2nd & 3rd quarter hour
i	2nd & 4th quarter hour
j	2nd, 3rd & 4th quarter hour

The day of the month that is highlighted is the current date or the date of the calendar you last displayed. If there are any notes for this date, that you are authorized to see, the first two lines show at the bottom of the monthly calendar.

To select a new date, either use the function keys or enter a date in the field at the top of the screen.

Function Keys for Look at the Whole Month

F1 HELP
Provides help for looking at the whole month.

F2 LOOK AT DAY DETAIL
Requests that the system display the detail calendar for the date specified by the date field.

F3 RETURN
Leaves the monthly display and returns to the PROCESS SCHEDULE screen. The date last specified is the current date used on this screen.

F5 NEXT DAY
Specifies that the next day is selected for the monthly display. Any notes for this day show at the bottom of the calendar.

F6 PREVIOUS DAY
Specifies that the preceding day is displayed on the monthly display. Any notes for this date show at the bottom of the calendar.

F7 PRINT
Requests that the system print the monthly calendar. You are shown a screen on which to select the destination for the printout.

F10 NEXT MONTH

Specifies that the next month is displayed. The first day of the month is selected as the current day. Any notes for this day are shown at the bottom of the calendar.

F11 PREVIOUS MONTH
Specifies that the preceding month is displayed. The first day of the month is selected as the current day. Any notes for this day show at the bottom of the calendar.

How to use Display/Print Daily Calendar Screen (F8)

To print the daily calendar press F8 from the PROCESS CALENDAR screen. The following illustrates what the screen looks like:

```
-------------------- Calendar Display / Print -----------------------

Specify the location where the calendar will be displayed/printed. To
display the calendar on your terminal, specify *Display. Specify the
Printer location where you want the calendar printed. Your default
Printer will initially be displayed.

        Location: IMAGEN      (use *DISPLAY to show calendar on scrn)


Number of Copies: 01


Specify the starting date (if different than shown) and the # of days.

Starting Date: 04 / 30 / 1993            Number of Days: 05
               (mm / dd / year)


F1:Help               Enter:Process request                  F3:Return
```

Figure 6.12 - Calendar Printing Screen

When requesting the system to print selected days from your calendar you must specify where you wish the printout sent and how many copies you want.

What to Put in the Fields

Print Location	Specifies the name of the location where the printout is to be sent. See your systems administrator for a list of valid names used at your installation. Use the location "*DISPLAY" to have the information displayed on your terminal.
Copies	The default number of copies is one; you can specify up to 99.
Starting Date	When printing your daily calendar, you can specify the print date of the first one.
Number of Days	Beginning with the above date, daily calendars are printed for the specific number of days. The default is five days; the maximum number of days is 31.

How to use Authorize/De-authorize Access (F12)

The following illustrates the screen display for authorizing and de-authorizing access. In this example "cckw" and "fred" are two users with the Access Type of XA and TA respectively.

```
------------------------ Authorized User List ------------------------

User IDs entered on this screen are authorized to view and/or update
your calendar. The Userid can be either a unique 1-16 character code
or 1-16 char code which uses a * or ? for a wildcard operator. You
may also use  a 'nickname' to refer to a list of Userids.  Refer to
the Help screen for additional information.
Type defines the authorization status for a User ID.  The type field
is a two character field.  The first character defines the amount of
information the  user can see, while the second sets the update status
(level of access). Access types are: T TA   L LA LU   C CA CU  X XA XU

ID/Nickname    T  ID/Nickname    T  ID/Nickname    T  ID/Nickname    T
cckw          xa  Fred           ta




F1:Help    Enter-Verify data    F3:End/Save       F12/A1:Exit - Cancel
```

Figure 6.13 - Authorize/De-authorize Access Screen

This facility allows you to specify who has access to your calendar and what the access level is.

ID/Nickname Field

Defines the MUSIC userid of the person having access to your calendar. You can specify the ID in three different ways.

1. As a "nickname". If you have set up a nickname via the Mail Directory program, the ID can be specified as a nickname. When you use a nickname as the ID, the authorization facility verifies that it actually exists in the nickname file. An error message is displayed if an ID entered cannot be located in the nickname file.

2. As a 1-16 character userid. Examples: CCH1 or CFX1 or MYNAMEIS

3. As a 1-16 character userid, which uses wildcard operators The wildcard operators are defined as a "?" and "*".

   The ? and * characters are the wildcard operators. You can use these characters to allow greater flexibility in authorizing access to you schedules.

   When the wildcard character "?" is used, it means that any character will match in that position. The wildcard character "*" will match any or no characters in that position. For example:

   CX* authorize any userid starting with CX.
   ??00 authorize any four character userid which ended with two zeros.
   G*VV allow any userid starting with G, ending with VV.

* authorize any userid.

Note: The valid characters for userids are: (A-Z, $#@_?*) in the first position and (A-Z, $#@_?*, 0-9) for the second through to the sixteenth positions.

Access Type

This field defines the type of access for that userid. It is a two character code. The first character defines the amount of information the authorized user is allowed to view. Valid options are:

T Individual(s) authorized to look at your calendar but will only be shown the Times you have something scheduled, not the descriptions.

L Individual(s) authorized to look at your calendar and view both the Times and Descriptions for all items except confidential and personal.

C Individual(s) authorized to look at your calendar and view both the Times and Descriptions for all items except personal.

X Individual(s) authorized to look at your calendar and view both the Times and Descriptions for all items.

The second character specifies if the user can view, add or update your Calendar. Only fields you have authorized him to look at (via the first character) can be changed.

" " Individual(s) allowed to view but not add, update or schedule meetings.

A Individual(s) allowed to view, add and schedule meetings. Any item that he added can be changed or deleted.

U Individual(s) allowed to view, add, update and schedule meetings.

How to use Month at a Glance Print

When requesting the system to print your Month at a Glance Calendar you must specify where you wish the printout to be sent. You may also request that the system produce multiple copies of the output.

Fields

Print Location	Specifies the name of the location where the printout is to be sent. See your systems administrator for a list of valid names used at your installation.
Copies	The default number of copies is one. Up to 9 may be selected.

How to use Recurring Calendar Events

Specify the starting date and ending date for the recurring event. You also specify how the recurring event will be scheduled. Only one of the following options may be specified.

Daily	Only the Everyday or Everywork day option may be specified
Weekly	Only the day(s) of the week may be specified
Bi-weekly	Only the day(s) of the week may be specified
Monthly	- Specify a specific Day of month (date). Only that date will be scheduled. - Specify a specific day (Monday - Sunday). Every selected day is scheduled. - add (First, Second...Last) option to limit to first Monday of a month. - Specify (First, Second...Last) and either Day or Weekday to schedule a specific day in a month. For example the second weekday in each month. - Specify the specific Months that will be scheduled.

Electronic Mail - TODO Item 2

This program offers you the ability to send electronic mail, to read incoming mail, or use the "HELP" feature to obtain further information. To use the MAIL system select item 2 from the TODO menu:

```
SELECT OPTION ===> 2

2.  Electronic Mail <option>
```

Figure 6.14 - Selecting Item 2 from the TODO Menu

See *Chapter 4. Electronic Mail* for full details about this program.

Telephone Log - TODO Item 3

Selecting item 3 from the TODO menu displays the following screen:

```
Date:                      Telephone Log

Time:

Caller's Name and Phone #:

Comments:









PF1-Help   ENTER-Process    PF2-Process + set new Caller   PF12-Return
```

Figure 6.15 - Telephone Log Screen

This program allows you to maintain a log of your telephone conversations.

The program automatically creates a new log for each month. All your phone conversations for a month can be found in a single file and remain on the system for one year. The program automatically deletes this file after a year.

The system can print a hardcopy version of the telephone log if you use the PRINT command. The phone log is maintained in a file with the name: PLOG.mmmyy where *mmm* is the month and *yy* is the last two digits of the year. Thus the telephone log for March 1986 would have the file name: PLOG.MAR86.

The program can also display the current date and time. This information cannot be changed by you. The time is updated whenever a new conversation is begun or by pressing F2 or changing the Caller's field.

Filling in the Required fields.

You must make an entry for the Caller's name and phone number. If you enter nothing, an error message is displayed.

Comments:
This area is used to contain any information about the call.

There is no limit to line length. Start typing a word on one line of the screen and finish it on the next. When this information is placed into the LOG file, any split words are correctly put together.

If you want to start on a new line, either use the NEW Line key on your keyboard or type a # character. If there are two #s in a row, a blank line is inserted into the log file. This is the only way blank lines can be placed into the Log.

If you have filled the comments area and need more room, press the ENTER key. The information in the

comments area is formatted and placed into the Log. The Comments area is cleared for additional information about a call. The time and caller information remains the same.

When you complete this call, press F2 instead of ENTER. Your comments are placed into the Log. The screen is cleared and you are ready to type in the information for the next call to be logged.

To exit, press F12. This causes an immediate return to the TODO menu. If you entered any information on the screen and then pressed F12, that information will NOT be written to the Log. You must have either pressed ENTER or F2 prior to pressing F12.

Summary of action keys.

ENTER	After you have entered in your comments about a call and you discover notes to make, use this key. The caller information remains the same and a blank comment area is displayed for you.
F1	Provides help information on the program.
F2	After you have entered in your comments about this call and you find you don't have any additional notes, use this key. The time field is updated and the Caller's and Comments area is cleared. You are ready for a new call.
F12	To exit from the Phone Log facility. Nothing that is presently in the comments area is saved. If you have entered notes for a call, use ENTER or F2 first.

Calculator - TODO Item 4

This program (called POLYSOLVE) can be used as a "desk calculator" and can be used to solve equations. The following operators are recognized when you use the POLYSOLVE program:

+	addition
-	subtraction
*	multiplication
/	division
**	exponentiation (Raising to a power. **2 means squared)
'	exponentiation (equivalent to **)

Sample uses as a desk calculator:

100 + 99 + 29	(addition)
10.72 * 97	(multiply 10.72 x 97)
100 / 3	(divide 100 by 3)
sqrt(97)	(the square root of 97)
10*(10.72/2 +17) + 29/3	(more complicated usage)

Use Polysolve to solve polynomial equations such as:

x=15/40	
x + 64 = 0	
17x2 = 16	(Note how we write x to the power 2)
16= x3 + 5x	
x= sqrt(15)	
ax3 + bx2 +cx +d = 0	

You will notice that you cannot solve the last equation immediately because of the unknown coefficients of a,b,c and d. Polysolve realizes this and asks you to define them.

Refer to the end of this section for more examples.

To use the calculator from the TODO menu, select item 4.

```
SELECT OPTION ===> 4

4.  Calculator <calc>
```

Figure 6.16 - Selecting Item 4 from the TODO Menu

The following screen appears with the message:

```
Enter Calculation or type HELP or /CAN
?
```

Or, If you wish you can type your calculation directly on the TODO menu:

```
SELECT OPTION ===> 4 8.95+67.89
```

In the above example, the 4 is the menu code for the calculator and 8.95+67.89 is the calculation.

Additional Polynomial Information

The program solves any polynomial in X with a maximum of 25 real coefficients. All computations are carried out in double precision. First order polynomials are solved algebraically.

The program solves polynomials for the unknown x entered according to the following conventions:

1. An expression not involving X is assumed to mean X=expression, and the value of X is evaluated and printed. For example, if the expression 2+2 is entered, the result ANS 4 is displayed.

2. An expression in X without an equal sign is assumed to mean expression=0 and the value or values of X are evaluated and typed.

3. An expression in X written with an equal sign is taken as is.

All coefficients must be real numbers. For example, SQRT(-2) is invalid.

Constants

- Stored in double precision form.
- Maximum constant value is approximately 10**50.

Variables

- The independent variable must be X.
- Coefficients may be represented by a single letter from A to Z except X.
- Avoid using E because of possible confusion with exponential notation for constants.
- Variables are stored as double precision numbers.
- 20 is the highest power of X which may be entered.

Implied Operations

- Parentheses imply multiplication if no explicit operator is given. For example 3(A+B) is taken as 3 times the sum of A and B.
- A constant immediately followed by a variable or a function name is assumed to have a multiplication operator in between the two parameters. For example, 3X is taken as 3 times X, and 3SQRT(B) is taken as 3 times the square root of B. However, 3XSQRT(B) is invalid since it would be taken as 3 times XSQRT(B) and there is no function called XSQRT.
- X followed by a constant is assumed to have an exponentiation symbol in between. For example, X2 is taken as X to the power 2.
- Blanks are always ignored.

Supported Functions

- The following functions are supported:

```
ALOG  log to base e
EXP   e to a power
SQRT  square root
```

```
SIN   sine of an angle expressed in radians
COS   cosine of an angle expressed in radians
```

- The double precision forms of these functions are used automatically

Continuing an Expression on the Next Line If:

- It ends in a comma.
- Is completely blank.
- Ends with the right parentheses count lower than the left parentheses count.
- The last character is an operator or an equal sign (=).
- A function name must not be split between lines, nor can a constant be split between lines.

Entering Coefficient Values

- The values may be entered explicitly in the form variable = constants.
- Expressions in the order in which they are requested.
- Implicit and explicit entries may be intermixed on one line; expressions must be preceded by an equal sign.
- Separate more than one value entered on a line with commas.
- Continuation rules shown above under "Continuing an Expression..." also apply for entering coefficients.

Solution

- The value or values of X are printed by the program.
- Complex roots are shown as real part and imaginary part.
- The letter "i" follows the imaginary part in the answer.

Examples

```
SELECT OPTION ===> 4

Enter calculation or type HELP or /CAN
?
3(x+1)=3
Incorrect character sequence found

Enter calculation or type HELP or /CAN
?
3x+3=3
Ans
        .0

Enter calculation or type HELP or /CAN
?
sqrt(-1)
IHN261I DSQRT NEGATIVE ARGUMENT=-0.9999999999999997D+00
Solution terminated

Enter calculation or type HELP or /CAN
?
```

```
1403+75-155/158+3*2501+
Continue statement
?
145.5+4
Ans
        9129.52

Enter calculation or type HELP or /CAN
?
6sqrt(b)=3x

Enter B.
?
=p+3f

Enter P,F.
?
f=c-d,25.5

Enter C,D.
?
d=1,10
Ans
        14.4914

Solve old equation again?
Type yes or no
?
no

Enter calculation or type HELP or /CAN
?
20.5x3 + 10x2 +32.3x=15/2.5 + sqrt(2.3)
Ans
         .212616
       -.350210      -1.26566   i
       -.350210       1.26566   i

Enter calculation or type HELP or /CAN
?
/cancel
PRESS ENTER TO CONTINUE....
```

Spelling Check - TODO Item 5

This full-screen spell checking program (called SPELL) offers you alternative spellings so you can correct the word on the screen. To use this program, type 5 on the SELECT OPTION line of the TODO menu, then press ENTER. The DOCUMENT SPELL CHECKING screen appears as shown in the figure below.

```
---------------------- DOCUMENT SPELL CHECKING -----------------------

File name  ==> _

Dictionary ==>

* For automatic generation and maintenance of an individual exception
  list for this document leave the exception list items blank.

Exception Lists:
Current   ==>
New       ==>

* To spell check only part of the document fill in the items below.

RANGE:     From line ===>
             To line ===>

ZONE:      To column ===>        (72 or 79 only, default is 72)

================================================================================
ENTER: Spell Check the Document Specified   PF1/13: Help  PF3/15: Exit
```

Figure 6.17 - Spell Check Screen

If you want to use the spell checking program for your documents, follow these simple instructions:

Type your information into the appropriate fields on the screen. (Use the next line key to skip to each field.) The file name field must be filled in. The other fields are optional.

File name	Type in the name of the document that you want spell checked.
Dictionary	Name of your dictionary. These words are checked along with the words in the system dictionary.
Current Exception List	Name of the file containing words which are exceptions. It is recommended that you leave this option blank and let SPELL take care of it automatically. "Filename@" is the default for the Exception List File (this file is created or appended to each time you spell check "filename"). See "New Exception List" below for more information.
New Exception List	Name of output file for new exceptions. As you are spell checking a file, you have the option to store words in this file. It is recommended that you leave this option blank. By default SPELL will append these words to "filename@" or create "filename@" if it doesn't exist.

RANGE: From line — The first line number of the section in the file that you want spell checked. The default is line 1.

To line — The last line number of the section in the file that you want spell checked. The default is the last line in your document.

ZONE: To column — The number of columns that you want checked by SPELL. The default is columns 1 to 72. Your choices are 72 or 79 only.

SPELL Options

The following SPELL options are in the form of function keys or commands that can be typed in the command area of the SPELL screen. When an option is both a function key and a command, it is usually faster to use the function key.

Program Function Key Pad

PA1 CANCEL/ NO SAVE	PA2

F1 HELP WORD	F2 CHECK FILE	F3 END/
F4 EXCEPT/ KEEP	F5 CORRECT WORD	F6 IGNORE/ NO KEEP
F7 UP LINE	F8 DOWN LINE	F9 REPEAT LOCATE
F10 TOP	F11 BOTTOM	F12 AUTO/ SPELL

* — This command echoes back the last command issued in the command area.

ANALYZE
AN — This command produces statistics about the document file being processed. The information includes the total number of words in the file, the the number of SCRIPT lines and the average number of words per line.

F12
AUTOSP
AU — Initiates or continues automatic spell checking. Automatic spell checking is performed on each word of every line from the current line onward. When a word is detected that is not in the master dictionary, not in the user dictionary, not in the exception list, nor in the change pair list, it halts and flags the target word.

YOU CAN THEN...
- enter the word in the exception list with F4 or F6, or the EXCEPT command.
- use the F2/F5 keys to enter the word in the change list.
- use the SET command to enter the word in the change list.
- correct the word by typing over it.

Note: In auto spell, attempts are made to make changes as words are encountered. If the new word is larger than the incorrect one and will not fit on the line, a message appears to tell you this. You can accommodate the change by modifying the line and restart auto spell check, or use either of F4 or F6 to ignore the underscored word.

F11
BOTTOM
B

This command sets the last line or range of the file as the current line.

PA1
CANCEL
CAN

Terminates the current spell checking, closes the file without saving the changes or output exception list, and drops back to the entry screen.

F2
CHECK word
C

Spell checks the word pointed to by the cursor. If the word is not found or recognized, it is flagged and a few suggestions or alternatives are offered.

The word is flagged as not found if
- It is not in the dictionary.
- It is not in the user's alternate dictionary.
- It was not flagged as an exception.
- It was not corrected using F5.
- It was not corrected using the SET command.

The suggested alternative spellings of a word are from the system dictionary only.

If you are typing the command rather than using F2 then this command is used along with a "word". "Word" is any word of length 1 to 24.

F5
CORRECT

The correct command must be used as a function key and cannot be typed in the command area. This key is used to automatically correct a flagged misspelling. It can only be used after F2. When a word is considered misspelled it is flagged and a few possible correct spellings are offered. By placing the cursor on one of the correction choices provided and pressing F5, the flagged word is corrected. In addition, the pair of words, the word flagged and the correct word, are stored as a change pair. During auto spell checking these words are automatically changed from the incorrect spelling to the corrected spelling as they are encountered. The SET command can be used to set a change pair without the use of the F2 combination of keys.

Note: After the use of F5 to correct a misspelling then pressing F5 a second time, automatically returns to auto spell checking. F5 will only reinitiate auto spell checking if had been just used for correction. You can, of course, reinitiate auto spell checking by pressing F12.

If the new word is larger than the incorrect one, and will not fit on the line,

you are told of the condition. The word pair are not placed in the change list, you must resolve the problem of fitting the new word on the line first. You can accommodate the change by modifying the line and restarting auto spell check, or use either F4 or F6 to ignore the underscored word.

DUPS
DUP

Initiates or continues the successive duplicate word check. Whenever a word follows itself it is flagged. You can then choose to correct it or reinitiate DUP. If you do not correct it, or the successive occurrence of the word is correct, use F8 to go to the next line before re-issuing the DUP command.

F3
END

Terminates spell checking and saves the file with changes and stores the accumulated EXCEPT words into the output exception list from the spell checking session. These words can be appended to your own master dictionary.

F4
EXCEPT word
E

Flags a word as an exception. When the cursor is positioned on a word, press F4 to make this word an exception. If EXCEPT is typed in the command area then *word* has to be included with the command.

When a word is flagged as misspelled, (not found in the master dictionary or your dictionary), you can tell the program to place the word in the exception list. For the duration of the session, words in this exception list are considered correctly spelled even though they may not be. The exception list is written to the output exception list file that you specified (or the default) when the session started. This list can then be appended to your own master dictionary, so the words will not be flagged as misspelled in subsequent spell check sessions.

Note: After the use of F4, you automatically return to auto spell checking. If the slow command was used to turn on the slow mode, restart auto spell by pressing F12.

FILE
FI

Performs the identical function as END.

FIND string
F

This command finds the next occurrence of *string.* FIND locates only string if it starts in column one (flush left).

F1
HELP

Use the function key only. Do not type HELP in the command area. This key provides help information on the SPELL program.

F6
IGNORE word
I

This is a companion key for F4. Like F4, F6 also flags a word as an exception. The word (if repeated in the document) is ignored for the rest of the spell check session. The word is not placed in the output exception list.

Note: After using F6, you automatically return to auto spell checking. If the slow command was used to turn on the slow mode, restart auto spell by pressing F12.

LINE n

LI	Places the current line pointer at the line specified by *n.* If *n* is outside the line range of the file or outside of the range specified on the entry panel, *n* will be assumed to be the top or bottom line respectively.
F9 LOCATE string L	Locates the next occurrence of *string.* LOCATE locates any string in any column. F9 will repeat the locate for the last string specified.
F8 NEXT n N	Moves the current line pointer *n* lines down. F8 means NEXT 1.
RULE R	Places a rule or tab line just below the current line. The rule is removed after each interaction.
SENCAPS SE	Initiates or continues the "sentence capitalization" checking. Whenever a word follows a period/blank (". ") combination and the word begins with a lower case letter, the word is flagged. You can then choose to capitalize it or re-initiate "SENCAPS". If you do not correct it, or the period is indeed correctly placed as in the case of an abbreviation, use F8 to go to the next line before re-issuing the SENCAPS command.
SLOW ON or OFF	Used to stop the automatic restart of auto spell checking after the use of F4, F5, or F6. It is preferable to run with SLOW OFF (default) since it saves time and interactions. However if you want to monitor the each step you can do so by issuing the command SLOW ON.
STATUS c ST	Queries the program about the following. - CP# reports the number of change pairs. - OE# reports the number of new exceptions and ignore words. - IE# reports the number of exceptions from the input exception file. - UD# reports the number of words in your master dictionary. - SD# reports the number of words in the system dictionary. - OEN reports the name of output exception file. - IEN reports the name of input exception file. - UDN reports the name of your master dictionary. - SDN reports the name of the system dictionary. - ALL reports all of the above.
SAVE SA	Saves the changes into the file being checked. The session is not terminated by this command.
SET word1 word2	Sets a change pair. This command is analogous to the "word checking" key (F2) followed by the "correction" key (F5). Whenever *word1* is encountered it is changed to *word2. Word1* must be a misspelled word, otherwise the pair is ignored. This is not a change command.
F10 TOP T	Moves the current line pointer to the top of the document file or range.

F7
UP n
U — Moves the current line pointer up *n* lines. F7 means UP 1.

CLEAR key — Clear error messages, the command area, and any change made to the current line since the last interaction with the program.

ENTER key — Processes the command area and performs any valid command entered here.

SPELL Checking in QUIET Mode (batch)

SPELL can spell check a document or list of words without your interaction. This can be done by setting up a file as follows:

```
/PARM *QUIET
/INC SPELL
options....     (as required)
/INC filename
```

Where filename is a file containing your document or words. Several files can be included here, or the text to be spell checked can follow in place of the "/inc filename".

Your master dictionary and an exception list can be specified on the option line. For example if your master dictionary is stored in the file "MYMASTER" and the exception list in file "DOC@" then the option line would be as follows:

```
master='mymaster',except='doc@'
```

Note the single quotes, these are required as shown. The specifications must be separated by a coma. If you do not want to specify either file name leave the option line blank. The option line must be present whether or not it is blank.

SPELL produces a listing of any word not in the system or your master dictionary, or in the exception list (if specified). The line number of any flagged word is listed along with the word. You can then use the editor to make the required changes to your document.

If you want to store this list of "misspelled" words you can change the previous file as follows:

```
/FILE 6 N(NEWWORDS) NEW(REPL)
/PARM *QUIET
/INC SPELL
 options.....
/INC filename
```

Where newwords is a file which contains the list of misspelled words.

Schedule a Meeting - TODO Item M

The MEET program is an extension of the SCHEDULE program (TODO item 1). It allows you to schedule a meeting or find a free time for a meeting. Also, you can see which calendars you are authorized to access for conference rooms, equipment items, and MUSIC users. (Users authorize access to their calendars by using the AUTHORIZE/DE-AUTHORIZE ACCESS function of the SCHEDULE program.) Finally, the MEET program updates the calendars of attendees and rooms/equipment items and sends mail to each participant.

To use MEET, type an M on the SELECT OPTION line of the TODO menu and press ENTER. The following panel (referred to as the MEETING screen) appears:

```
---------------------- SCHEDULE A MEETING ---------------------------

 Command ===>

                                                  TIME: 10:15 am
 Conference Rooms &
 Equipment Items   ===>                     1989   FEBRUARY   1989
                   ===>
                   ===>                    S   M   T   W   T   F   S
                                                                1   2
 Attendees ===>CCKW                        3   4   5   6   7   8   9
           ===>                           10  11  12  13  14  15  16
 Meeting Information                      17  18  19  20  21  22  23
  Begin date    ===> 14FEB89 (ddmmmyy)    24  25  26  27  28
  End date      ===> 14FEB89 (ddmmmyy)
  Begin time    ===> 1200    (hhmm)             Day of year:  49
  End time      ===> 1700    (hhmm)
  Time required ===>    (hours)    (minutes)

F1:Help ================================================== PA1:End
F2:Potential Attendees   F4:List Rooms/Equip   F5:Find a Time   F3:End
F6:Schedule a Meeting  F9:Send a Notice  F10:Prev Month  F11:Nxt Month
```

Figure 6.18 - Schedule a Meeting Screen

How to Enter Information on the Meeting Screen:

The fields on the MEETING screen that are filled in by default include: your userid in the attendee field, dates, and times. Before you fill in or change information on the screen, you may want to use either F2 - POTENTIAL ATTENDEES or F4 - LIST ROOMS/EQUIP to help you select items.

Once the screen has been filled in, press F5 - FIND A TIME. The information on the MEETING screen is used to check the calendars of attendees and rooms/equipment items to look for free time. After updating the screen with the correct rooms and time, press F6 - SCHEDULE A MEETING. The information on the MEETING screen is used to schedule the meeting by updating the calendars for attendees and rooms/equipment items. (See FUNCTION KEY DESCRIPTIONS later for more information.) The fields on the MEETING screen are described below:

Command ===>

This area is reserved for entering commands. Most commands for this program are assigned to function keys.

Conference Rooms &
Equipment Items ===>

Conference rooms and equipment items can include any rooms or items whose calendars you are allowed to access. To select more than one room or item, leave a comma or a space between the multiple selections. A room or item prefixed with an asterisk (e.g. *TOKYO) will not have its schedule checked for any function selected. This allows you to use unknown rooms or items. An example of this usage would be a meeting that is to be scheduled to take place at an unknown location in TOKYO. By using *TOKYO in the CONFERENCE ROOMS & EQUIPMENT ITEMS field, the system will not look for the schedules for TOKYO. Rooms or item names can be from 1 to 16 characters in length, excluding an asterisk prefix. See "Appendix D" for further information on conference rooms and equipment items.

Attendees ===>

The ATTENDEES field is a list of MUSIC userids or nicknames (see Mail Directory in *Chapter 4 - Electronic Mail* for information on nicknames) who have allowed you to access their calendars. Multiple ATTENDEES can be selected by leaving a comma or a space between the selections. Your userid is displayed by default.

A total of up to 40 conference rooms, equipment items and attendees can be selected.

Meeting Information (Dates and Times)

Meeting information is used by the FIND A TIME (F5), the SCHEDULE A MEETING (F6), and the SEND A NOTICE (F9) commands. The dates and times serve two purposes: they provide the range required to find a time, and they indicate the actual date and time for the meeting.

Begin Date ===>
End Date ===>

Todays date appears by default. To change this date, type over it in the form ddmmmyy (e.g. 01apr87). If it is after 3:00 p.m. (this may be different at your installation depending on the default BEGIN and END times) when you are using this program, then the dates for the next day are displayed. The BEGIN DATE indicates the starting range for finding a time or the start of the meeting. The END DATE indicates the last day of the meeting and should be changed if you wish to check several days for free time, or if you wish to schedule your meeting for more than one day.

Begin Time ===>
End Time ===>

Default times appear in these fields (see examples below for more information about default times). To change the start of the range for finding a time (F5) or the start of the meeting (F6 or F9), type a new BEGIN TIME in the form hhmm (i.e. 1015). Times can be given in either 2400 hour format or 12 hour format using AM or PM suffix (i.e. 1015 or 10:15am). The END TIME indicates the end of the range for finding a time, or the end of the meeting. See the topic Examples for more information about default times.

Time required ===>

Indicates how long the meeting will take and is important when you are finding a time. For example, if the times for BEGIN and END are 0900 and 1700 respectively, and you want to schedule a 1 hour meeting, specify "1" in the TIME REQUIRED (HOURS) field. If you do

not specify a time length in this field, the MEET program searches everyone's schedule to find 8 hours of free time instead of looking for just one hour. For scheduling the meeting (F6) or sending notice of a meeting (F9), you can set the length of the meeting by using this field or the END TIME field.

Function Key Definitions

PA1 END
Exits from this program.

F1 HELP
Provides help for this program.

F2 POTENTIAL ATTENDEES
Displays a selection list indicating all the schedules which you are allowed to access.

F3 END
Exits from this program.

F4 AVAILABLE ROOMS/EQUIP
Displays a selection list indicating which schedules can be accessed for rooms or equipment.

F5 FIND A TIME
Use this key to find available times for meetings (i.e. no conflicts with schedules). The range used by the program for finding times is specified on the screen under dates and times. The rooms/equipment items and attendees on the screen indicate which schedules to check. When you press F5 a screen appears informing you of the free times available (i.e. when schedules indicate free time periods at the same time). It is recommended that you use this key before F6 or F9.

F6 SCHEDULE A MEETING
Use this key to choose a specific meeting time, inform attendees, and update the schedules of those concerned. Before pressing F6, make sure that the MEETING screen now reflects the actual room, attendees, dates and times for your meeting. The following steps occur:

1. If you are not authorized to access a schedule of a room/equipment or an attendee, you are given a chance to revise the room/equipment and attendees list. If this is not the case, then you go directly to step 2.
2. Next you are informed if there are time conflicts.
3. You are then presented with a screen to enter your mail text.
4. Mail is sent and the schedules for the rooms/equipment and the attendees are updated.

F9 SEND A NOTICE
Use this key to send mail for intent to schedule a meeting. This function is similar to SCHEDULE A MEETING except that schedules are not updated. Mail is sent to rooms and attendees, and unknowns if desired.

F10 PREVIOUS MONTH
Change the calendar display to show the previous month.

F11 NEXT MONTH
Change the calendar display to show the next month.

Potential Attendees (F2)

This selection displays a list of all the MUSIC userids whose schedules you are allowed to access. This list contains a USERID field, a NAME field and an ACCESS field. The NAME field gives the name for the MUSIC userid stored in the user's MAIL profile (MAIL option 5). The possible types of access are VIEW, ADD, and UPDATE (see the help facility in the SCHEDULE program under AUTHORIZE/DE-AUTHORIZE ACCESS for definitions). Select a displayed item if you want the item put into the Attendees field on the MEETING screen and made available for subsequent use. You can select as many items as can fit into the field on the MEETING screen. To select a displayed item, enter an S in the position on the left side of the item.

Press F3 to cancel the displayed items list and return to the MEETING screen. The Attendees field on the MEETING screen is not changed.

Press the ENTER key to return to the MEETING screen. If some displayed item are selected, these are placed in the Attendees field on the MEETING screen with commas separating them. If no items are selected, then the field on the MEETING screen is not changed.

The following is an example of the information on this screen:

```
 Userid            Name            Access

_CCES              EARL SMITH      UPDATE
_CCJS              JILL SMITH      ADD
_IN05              TOM BROWN       VIEW
```

Available Rooms/Equipment (F4)

This selection displays a list indicating which schedules can be accessed for rooms or equipment items. The list contains a NAME field which gives you the name of the conference room or equipment item. The DESCRIPTION field describes each room or item. The type of access you have to a particular room or item is displayed in the ACCESS field. The possible types of access are VIEW, ADD, and UPDATE. Your access to a particular conference room or equipment item is assigned by the administrator of each item.

Select a displayed item if you want the item put into the Conference Rooms and Equipment Items field on the MEETING screen and made available for subsequent use. You can select as many items as can fit into the field on the MEETING screen. To select a displayed item, enter an S in the position on the left side of the item.

Press F3 to cancel the displayed items list and return to the MEETING screen. The Conference Rooms and Equipment Items field on the MEETING screen is not changed.

Press the ENTER key to return to the MEETING screen. If some displayed items are selected, these are placed in the Conference Rooms and Equipment Items field on the MEETING screen with commas separating them. If no items are selected, then the field on the MEETING screen is not changed.

The following is an example of the information that displays on the screen:

```
 Name              Description        Access

_ROOM104           Room for 25        UPDATE
_ROOM105           Room for 15        ADD
_LOUNGE1           Lounge 1st fl      UPDATE
```

Find a Time (F5)

Use this key to find available times for meetings (i.e. no conflicts with schedules). The range used by the program for finding times is specified on the screen under dates and times. The rooms/equipment items and attendees on the screen indicate which schedules to check. When you press F5 a screen appears informing you of the free times available (i.e. when schedules indicate free time periods at the same time). It is recommended that you use this key before F6 or F9.

The following illustrates the type of information that displays on the screen:

```
Date                   Begin Time    End Time

_21APR86 TO 22APR86     0900          1700
_23APR86                0900          1630
_24APR86                0900          1245
_24APR86                1415          1700
```

The DATE field indicates the date (in format DDMMMYY) for the free time. If this field shows two dates, then it signifies that the begin and end times apply to all dates between and including the two dates specified.

You can select one of the displayed free times by entering an S in the position on the left side of the free time.

Press F3 to cancel the displayed items list and return to the MEETING screen. The dates and times fields on the MEETING screen are not changed.

Press the ENTER key to return to the MEETING screen. If a displayed free time is selected, then the dates and times are placed in the meeting information fields on the MEETING screen. If a free time is not selected, then the dates and times on the MEETING screen are not changed.

Schedule a Meeting (F6)

Use this key to choose a specific meeting time, inform attendees, and update the schedules of those concerned. Before pressing F6, make sure that the MEETING screen now reflects the actual rooms/equipment, attendees, dates and times for your meeting. The following steps occur:

1. If you are not authorized to access a schedule of a room, an equipment item, or an attendee, you are presented with a list of all rooms, items, and attendees listing your access to their schedules. See the topic Rooms/Equipment and Attendees Selected for further details.

   If access is allowed in all cases, then this screen does not appear and you go directly to step 2.

2. If there are time conflicts with the schedules of attendees and/or rooms/equipment items, then you are informed. See the topic Time Conflicts for details.

   If there are no time conflicts, then this screen does not appear and you go directly to step 3.

3. You are presented with a screen to enter your mail text. See Entering Mail Information for details.

4. Mail is sent and schedules for rooms/equipment items and attendees are updated.

Sending a Notice (F9)

When you are sending a notice using F9 - NOTICE, you are presented with the following screen for entering you mail text:

```
---------------------- Meeting Message Information --------------------

Type in the information requested below.


Mail text for the meeting ===> ____________________________________________
                              ____________________________________________
                              ____________________________________________
                              ____________________________________________
                              ____________________________________________
                              ____________________________________________

Do you wish to have the mail sent to unknown attendees? _ (Y or N)


================================================================================
F1:Help               ENTER: End/Continue                F3:End/Cancel
```

Figure 6.19 - Sending a Notice Screen

Mail is sent to all attendees (who have allowed you to access their schedules) to inform them of the meeting. If a conference room or equipment item has been selected, the mail is sent to the administrator of the room or item.

The Mail Text for the Meeting information is used as the mail text. The first line of the mail text includes the date and time of the meeting. This line is added for you by the system. The subject line of the mail item includes the date and time of the meeting.

You can have the mail sent to all attendees who have not allowed you to access their schedules. To have this done, enter a Y to the question "Do you wish to have the mail sent to unknown attendees?" The default is N.

Press F3 to cancel the selected function and return to the MEETING screen. Press the ENTER key to continue the selected function.

Rooms/Equipment and Attendees Selected

If you are not authorized to access a schedule of a room, an equipment item, or an attendee, you are presented with a list of all selected rooms, items, and attendees and your access to their schedules.

The list contains a USERID OR NAME field which gives you the userids or the names of conference rooms or equipment items.

The R/A field indicates whether the listed item is a conference room or an equipment item, represented by an R, or an attendee, represented by an A.

The NAME/DESCRIPTION field gives the name of the MUSIC userid if known, or the description of the conference room or equipment item if available.

Under the STATUS heading, ACCESS tells you that you are allowed access to the schedules of this item. UNKNOWN status implies that either the selected item is not known to the system, or the item is known to the system and you are not allowed to access the schedules of this item.

Press F3 to cancel the selected function and return to the MEETING screen. None of the fields on the MEETING screen are changed.

Press the ENTER key to continue the selected function. The MEET program still allows you to schedule the meeting even if there is an UNKNOWN status. The schedules for those with UNKNOWN status will not be updated and mail may not be sent.

You can revise this displayed items list by selecting from this list the items desired. To select an item, enter an S in the position on the left side of the item. If you select some items, only the items selected are used. However, if no items are selected, all of the displayed items are used.

The following shows an example of the screen which appears:

```
 Userid or Name    R/A   Name/Description   Status

_RM104              R    SPACE FOR 10       ACCESS
_CCTS               A    TOM SMITH          UNKNOWN
_CCJB               A    JOE BROWN          ACCESS
 ================================================
 F1:Help      ENTER:End/Continue      F3:End/Cancel
```

Time Conflicts

There may be time conflicts with the schedules of attendees and/or equipment items when you Schedule a Meeting (F6). You are shown a list of the time conflicts if they occur.

The Userid or Name field displays either the userid of the attendee or the name of a conference room or equipment item.

The R/A field indicates whether the listed item is a conference room or equipment item, represented by a R, or an attendee, represented by an A.

The Date field gives the date found for the time conflict in DDMMMYY format.

The Begin Time and End Time fields give the begin and end time for the date(s) of the time conflict respectively.

Press F3 to cancel the selected function and return to the MEETING screen. None of the fields on the MEETING screen are changed.

Press the ENTER key to continue the selected function.

The following is an example of the screen that appears when time conflicts occur:

```
Userid or Name   R/A     Date    Begin Time   End Time

  CCKW            A    23APR89      1330        1630
  RM104           R    23APR89      1600        1700
=====================================================
F1:Help         ENTER:End/Continue        F3:End/Cancel
```

This screen displays all the time conflicts that are found for each selected attendee or room/equipment item. In the above example, the meeting was to be scheduled at 4:00 p.m. on April 23rd. The calendar for CCKW already has something scheduled between 1:30 and 4:30 p.m. and this is indicated on the screen. At this point you may press F3 to cancel and return to the MEETING screen or you may continue scheduling the meeting by pressing the ENTER key. The MEET program allows you to schedule the meeting even if there are time conflicts. These schedules are updated and include overlapping times.

Entering MAIL Information

When you schedule a meeting using F6 - SCHEDULE A MEETING, you are presented with the following screen for entering your mail text:

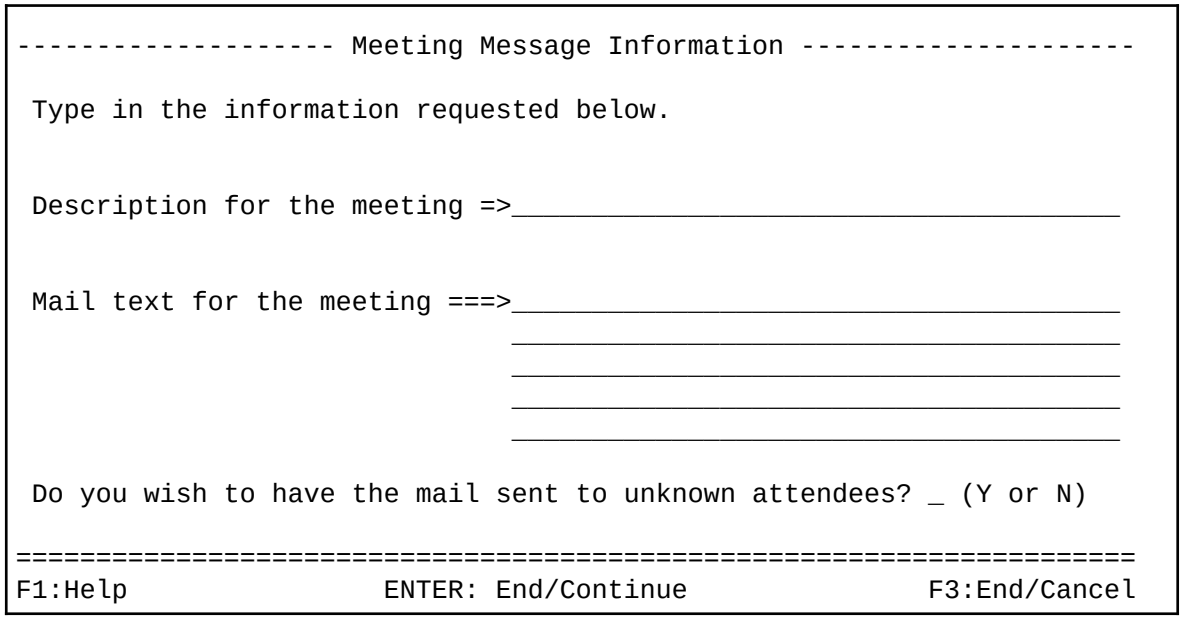


Figure 6.20 - Meeting Message Information (*Mail*)

The DESCRIPTION FOR THE MEETING information is placed in the description field associated with the schedule entry. The schedule entry is added to the schedule as a confidential item. The times used for the schedule entry are the times you specified for scheduling the meeting.

Mail is sent to all attendees (who have allowed you access to their schedules) to inform them of the meeting and that their schedules have been updated. If a conference room or equipment has been selected, the mail is sent to the administrator of the room or item.

The Mail Text for the Meeting information is used for the mail text. The first line of the mail text includes the date and time of the meeting. This line is added for you by the system. The subject line of the mail item includes the date and time of the meeting.

You can have the mail sent to all attendees who have not allowed you to access their schedules. To have this done, enter a Y to the question "Do you wish to have the mail sent to unknown attendees?" The default is N.

Press F3 to cancel the selected function and return to the MEETING screen. Press the ENTER key to continue the selected function.

Examples for Meeting Information

The default times which appear on the MEETING screen, when the program is first invoked, are determined by your installation. In the following examples, the default times are assumed to be 0900 for BEGIN TIME and 1700 for END TIME. When you request the MEET program, the END time is set at 1700, but the BEGIN time is set at least 1 full hour away from the current time. (BEGIN time will not be more than 2 hours away from the current time unless it is before 7:00 am, then the BEGIN time is 0900.) The default times are affected by the actual time of day it is when you are using this program. If the current time is after 3:00 pm, then the date changes to the next day.

You can always override the dates and times displayed on the MEETING screen by typing over them. Make sure that the BEGIN time does not come after your END time.

Examples:

Current		Screen Display		
Date	Time	DATE	BEGIN Time	END Time
14FEB89	0600	14FEB89	0900	1700
14FEB89	0930	14FEB89	1100	1700
14FEB89	1600	15FEB89	0900	1700

The next example indicates how the schedules will be affected if your meeting is scheduled for several days. The following shows how the MEETING screen was filled in and the results entered in to the schedules for the conference room and attendees.

```
      MEETING Screen                          Schedules

Begin date    ===> 10MAY89              10MAY89     1100   1700
End date      ===> 13MAY89              11MAY89     0900   1700
Begin time    ===> 1100___              12MAY89     0900   1700
End time      ===> 1600___              13MAY89     0900   1600
Time required ===> __ (hrs) __ (min)
```

The above example illustrates one long continuous meeting. If you wish to have different times for each day, then you need to schedule separate meetings.

MEET Commands

The following commands are available and some of these commands are in the form of function keys and have been explained previously.

CANCEL (PA1)	HELP (F1)	NOTICE (F9)
CLEAR	KEYS	OFF
DELIM char	LISTCODE (F2)	PFnn def
END (F3)	LISTROOM (F4)	PREVMON (F10)
EXEC	MEETING (F6)	*-n
FINDTIME (F5)	NEXTMON (F11)	

CLEAR Clear the screen input areas.

DELIM char	Change the multiple command delimiter to *char*, where *char* is a character of length 1, and not one of the characters a to z, 0 to 9, *, =, /.
EXEC	Pass the command to MUSIC to be executed.
KEYS	Display a screen which allows you to change the function key definitions. These changed definitions are saved in a file, MEET.KEYS, so that they can always be used.
OFF	Exit from this program.
PFnn	Show the definition of PFnn, where *nn* is a number from 1 to 24.
PFnn def	Set the definition of PFnn to *def*, where *nn* is a number from 1 to 24. The definition can be from 1 to 50 characters long.
*	Display the last command entered in the command area.
*-n	Display the previous *nth* command entered in the command area, where *n* is a number from 0 to 4.

Chapter 7. Utilities

Utility Menu

The Utility menu is displayed when you select item "U" from the TODO menu, as follows:

```
SELECT OPTION ===> u
```

The following figure illustrates the Utility menu:

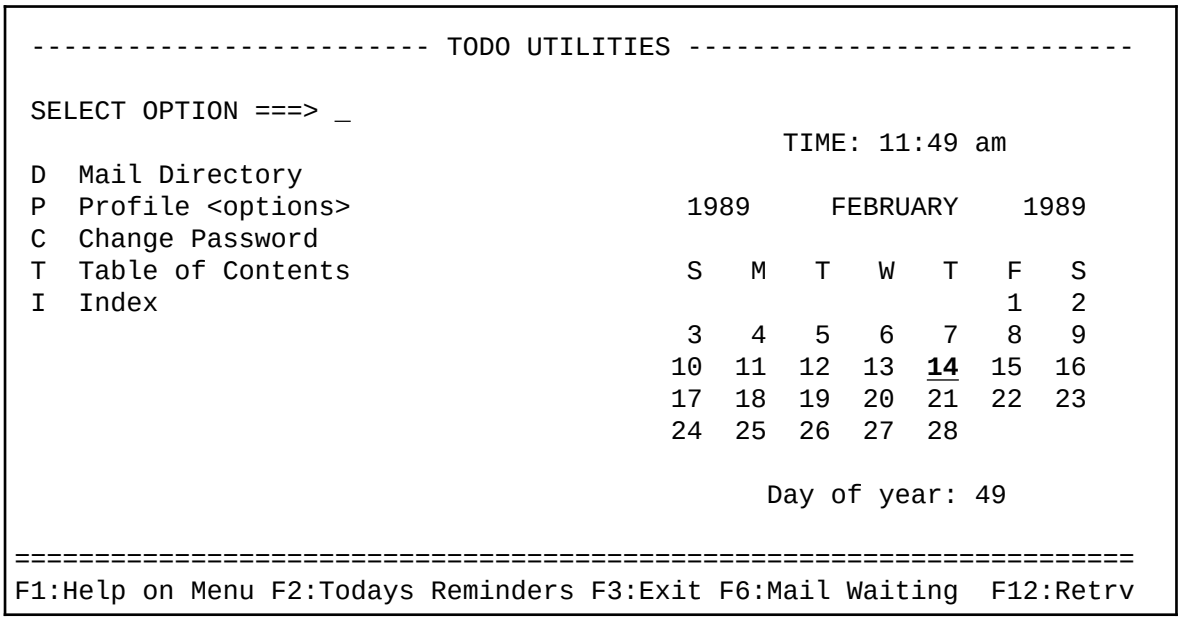


Figure 7.1 - Utility Menu Screen

Note: You can go directly to an Utility menu item from the TODO menu if you include the code for that item.

```
SELECT OPTION ===> u p
```

"u p" brings you directly into the Profile facility from the TODO menu.

Read the following brief descriptions of the selections available from the Utility menu. For more information, go to the detailed description for each selection.

Mail Directory — This item invokes the Mail Directory program for entering nicknames for other user-ids. Instead of referring to a person by their userid, you can use his or her name. It's much easier to remember a name than it is to remember another person's userid. These nicknames are used for sending mail (see ELECTRONIC MAIL) and scheduling calendars (see SCHEDULE). Refer to the topic "Mail Directory" in *Chapter 4. Electronic Mail* for more information about nicknames.

Profile <options> — Used to access the PROFILE program for information about your userid or if you need to change a PROFILE option. Some of the options include changing your userid sign-on password, changing the batch password, or printing the userid profile. See the section PROFILE in this chapter for details.

Change Password — This invokes the PROFILE program, but only the sign-on password can be changed.

	This is the most frequently used PROFILE option. See the section PROFILE for details.
Table of Contents	This item invokes the TABCON program for creating a Table of Contents for a SCRIPT document. See this section for instructions.
Index	This item invokes the MAKIDX program for generating an index. Please see this section for information.

Mail Directory - Utility Item D

This item on the Utility menu invokes the Mail Directory for entering nicknames. To use this facility select item D from the Utility menu:

```
SELECT OPTION ===> d

D.  Mail Directory
```

Figure 7.2 - Selecting Item D from the Utility Menu

See *Chapter 4. Electronic Mail* under the topic "Mail Directory" for full details about this program.

Profile - Utility Item P

User Profile Program

The PROFILE program can be used to list information about your userid "profile". It can also be used to alter some of the options.

To use this program, type P on the SELECT OPTION line of the Utility menu, then press ENTER. The following figure illustrates the messages that appears after selecting this item:

```
SELECT OPTION ===> p

USER PROFILE - ENTER COMMAND OR HELP
?
```

Figure 7.3 - Selecting Item P from the Utility Menu

You then specify changes (options) to the userid profile in the form:

option1(value1) option2(value2) etc.

A PROFILE command is the same as an option. Options are separated by blanks or commas.

If only one option is needed for the PROFILE program, it can be entered directly from the Utility menu. For example, "p print" or "p bpw(xxx)". The most common options are described below. For all options refer to the *MUSIC/SP User's Reference Guide*.

Note: Some changes to the user profile take effect only after the you sign off, then sign on again.

AUTOPROG(name)
AUTO(name)
: Sets up the name of an autoprogram which automatically runs when the user signs on with an /ID command. *name* is the name of a file. To remove an existing autoprog specify AUTO(). You will be unable to change the name of the autoprog if your userid was authorized with the noncancellable autoprog feature.

  To have the TODO menu automatically display when you sign on to MUSIC, enter the following PROFILE option:

```
AUTOPROG(todo)
```

BPW(xxx)
: This parameter specifies a new batch password (from 1 to 8 characters). Use this option if your userid is authorized for batch processing.

END
: Stops the PROFILE program.

HELP
: Displays information on PROFILE commands.

PW(xxxxxx)
: Assigns a new password that will then be required the next time you sign on. The password can contain one to eight characters. When this option is used, the system

prompts you for the old password. Your userid may not allow you to change your own password.

PRINT — Displays some information about your userid profile, such as defaults for tabs, job times, etc.

PRINT$ — Displays the accumulated connect and processing unit charges for your userid. This field is normally updated each evening so it does not reflect the charges accumulated for the current day. A maximum charge allocation is also displayed and you are not allowed to sign on, once you reach your limit. Contact the MUSIC Systems Administrator to have this allocated limit increased. Some installations may choose not to maintain these fields.

PRINTSP — This parameter causes Save Library space information to be displayed. The values displayed are total file space for your userid (in units of 1K = 1024 bytes), the maximum total space allowed, and the maximum size for an individual file.

ROUTE(destination) — Assigns a default route destination for your userid.

TERM(xxxxxxxx) — At sign-on time, MUSIC checks the general type of workstation you are using against the one defined for this option. Only if they match, and TRMCLS is not given on the /ID line, will the backspace and tabs be used as defined in your profile. This prevents, for example, tab settings for one type of workstation being used for another. The workstation types that MUSIC automatically distinguishes between are: 3270, 2741 (including 3767 and CMCST terminals), 1050 and TTY. These names can be used in the TERM option. For example, "TERM(2741)". For other names which can be used (such as 3101), refer to the description of the /ID command in the MUSIC COMMANDS section of Chapter 1.

This TERM parameter also defines a terminal subclass. For example, the computer cannot distinguish between a 72-character TTY terminal and a 132-character wide TTY Model 38 terminal. If your profile has TERM(TTY38) specified, it assumes that the TTY-type terminal you are using is the longer line Model 38 type. See the discussion of "trmcls" under the description of the /ID command in the MUSIC COMMANDS section of Chapter 1 for other valid names.

To remove the workstation specification use TERM().

Example

```
 SELECT OPTION ===> p

*IN PROGRESS
User Profile - Enter command or HELP
?
password(trees) term(3101)
Enter your current MUSIC sign-on password
?

________
CHANGED
?
print
  USERID=ABCD              FILE OWNERSHIP ID=ABCD
  ID=                  NAME=
  TIME LIMITS (IN SERVICE UNITS):
```

```
    PRIME=NOLIMIT  NONPRIME=NOLIMIT  BATCH=NOLIMIT  DEFAULT=32
  MAX NUMBER OF EXTRA SESSIONS PER TERMINAL:  3
  PASSWORD CAN BE CHANGED BY USER
  AUTOPROG:  TODO                      (CANCELLABLE)
  TERMINAL TYPE:  3101     (0C)
  INPUT TABS ARE    7  73
  NO OUTPUT TABS
  FUNDS ($):      420.14 USED,  NO LIMIT
  SAVE LIBRARY:  TOTAL =  6594K   LIMIT = 10000K   MAX/FILE = 4000K
  MAX TRACKS PER DATA SET (UDS) AT ALLOCATION:       50
  CREATED  1985/12/15  (YEAR/MONTH/DAY)
  LAST SIGN-ON: 1986/03/28  9:21  LAST BATCH JOB: 1986/03/27 10:21
  LAST PASSWORD CHANGE: TERMINAL PW 1986/01/31  BATCH PW 1986/03/12

?
end
PRESS ENTER TO CONTINUE....
```

Table of Contents - Utility Item T

This item invokes the MAKCON program which is used to create a table of contents and other similar lists such as table of diagrams. The program's input is normally generated by .OX control words imbedded in your SCRIPT document. (Often, these control words are included in user defines, so the typist may be unaware of their existence.) The following figure shows the messages that appear on the screen after entering the "T" Utility item on the SELECT OPTION line. The text that is underlined indicates answers to the messages (your information would replace these answers). The answers in the figure correspond with the sample files in figures 7.5 and 7.6.

```
SELECT OPTION ===> t                        (item T on the utility menu)

*** MUSIC/SCRIPT TABLE OF CONTENTS ***
PLEASE REFER TO THE TODO MANUAL FOR INFORMATION

ENTER THE NAME(S) OF YOUR SCRIPT FILE(S), HELP, OR /CANCEL

sample                                      (type the name of your file)

DO YOU WISH INFORMATION FOR THE SCRIPT TABLE OF CONTENTS?

no

ENTER SCRIPT OPTIONS, HELP, OR /CANCEL

_____________                          (type script options or blank line)

ENTER HERE THE NAME OF A FILE WHERE YOU WANT TO STORE YOUR
YOUR TABLE OF CONTENTS, HELP, OR /CANCEL

sample.table                           (give a name for table of contents)

ENTER THE VALUE FOR ".LL" (LINE LENGTH), HELP OR /CANCEL

65                                                (choose a line length)

ENTER THE VALUES FOR INDENT, HELP, OR BLANK FOR DEFAULT

0,3,6,9,12                                 (indent values or blank line)

ENTER VALUES FOR SPACE, HELP, OR BLANK FOR DEFAULT

1,0,0,0,0                                   (space values or blank line)

ENTER T OR F VALUES FOR EACH UPPER CASE LEVEL, HELP, OR BLANK

t,f,f,f,f                                (upper case levels or blank line)

PRESS ENTER TO CONTINUE.....
```

Figure 7.4 - Selecting and Running the Table of Contents Program

Table of Contents Options

INPUT=n Specifies the unit number that contains the file generated by the AUXOUT option of SCRIPT. The default unit number is 5.

OUTPUT=n Specifies the unit number where the program is to write the Table of Contents. The output of this program must be processed by SCRIPT to do the required formatting. You can save the output of this program in a file and use the SCRIPT control word of .IM to incorporate it into your manual at the appropriate place. The last line generated by this program is the .FI control word. The default for this option is OUTPUT=10.

LL=n Specifies the maximum width of the output table of contents. The default is LL=60 meaning 60 characters wide.

INDEN=n0,n1,n2,n3..n9
Specifies up to 10 indents corresponding to the digits 0 through 9 given on the .OX control word in your document. If specific indents are negative, then the corresponding items are not included in the table of contents. The default values are INDEN=0,0,4,6,8,10,12,14,16,18.

SP=n0,n1,n2,n3...n9
Specifies how many spaces are to be generated before items with the corresponding identification numbers. If you specify a number as -1 then a page eject is generated before that item (unless the item is the first one found). The default settings are SP=-1,1,0,0,0,0,0,0,0,0.

UP=TF0,TF1,TF2,TF3...TF9
Specifies if the contents item is to display in upper case (even if the input text in your document is lower case). The option can be given as either T (true) or F (false). If the input text line was already in upper case, this option has no effect. The default settings are UP=T,T,F,F,F,F,F,F,F,F.

The following figures show a sample SCRIPT document with .OX control words and the sample Table of Contents generated from this file.

```
.ox 0
.ce
Section 1
.pa
.ce
.ox 1
Chapter 1
.sk 2
.ox 2
Introduction
.br
The following is an introduction etc...
.sk 2
.ox 2
Contents Utility
```

Figure 7.5 - File Called Sample with .OX Control Words

```
.nf
.up
Section 1  . . . . . .   1
.sk  1
.up
Chapter 1  . . . . . .   2
    Introduction . . . . . .   2
    Contents Utility . . . . . .   2
.fi
```

Figure 7.6 - File Called Sample.Table

Index - Utility Item I

This item invokes the MAKIDX program which produces an index suitable for inclusion at the end of a manual or book. The index can be easily modified at a later time without major effort.

Normally an index is prepared when the main body of the document is in final form.

The following figure shows the messages that appear on the screen after entering the "I" Utility item on the SELECT OPTION line. The text that is underlined indicates answers to the messages (your information would replace these answers). The answers in the figure correspond with the sample files in figures 7.8, 7.9 and 7.10.

```
SELECT OPTION ===> i                        (item I on the utility menu)
PLEASE REFER TO THE TODO MANUAL FOR INFORMATION

ENTER THE NAME(S) OF YOUR INDEX DATA FILE(S), HELP, OR /CANCEL


index.data                                  (type the name of your file)

ENTER THE NAME OF THE OUTPUT FILE INTO WHICH YOU WANT TO STORE THE
NEW INDEX FILE.


index.new                                (type the name of the new file)
PRESS ENTER TO CONTINUE.....               (you will return to the menu)
```

Figure 7.7 - Selecting and Running the Index Program

The user creates a file consisting of a sequence of page numbers each followed by the list of keywords which are to refer to that page. An example is shown below.

```
#23   <----------------------(page number 23)
Computer Aided Instruction
CAI
Instruction, Computer Aided
#26
CAI, Examples
Examples, CAI
#3
Computer, Charge Rates
```

Sometimes you need to consider several words as one item. This is to prevent related sequences from being listed under the same main heading. A defined "concatenation" character handles such occurrences. The index program uses the underscore character (_) for the concatenation character. Thus "Computer Aided Instruction" can be entered as "Computer_Aided_Instruction".

You can specify the page number as "##". This means the last page number plus one. The use of this feature is helpful if the page numbers are subject to modification at a later time. A page number symbol of ## following a page number of #23.1 is interpreted as #23.2.

The page numbers need not be specified in numeric order. Furthermore, the same page number can be specified more than once. It will prove easier to proofread the list and make modifications if the page numbers are in order. Note that if the same item is found on several pages, the index will list the page numbers in the order they were found in the input. This might be useful for some applications.

A page number specification of "# " is treated as a null page. The effect of this will allow the inclusion of cross-reference notes such as "CAI (see Computer Aided Instruction)" without a page number present.

The page number may be up to 20 characters in length and the keyword line is limited to 50 characters in length.

Parameter Line

The first line of the index list contains a control line of the following form:

```
#_
```

where the first character is the page number symbol and the second is the concatenation symbol. If, for example, a symbol "#" is to appear as the first character of a keyword, then it cannot be used as the page number symbol. The concatenation character will always display as a blank character in the output index.

Index Options

INPUT=n	Specifies the unit number which contains the index list. The default value is 5.
OUTPUT=n	Specifies the unit number to contain the output document. n=6 is the default. When n is specified as anything other than 6, then the output of the index program is suitable for input to the SCRIPT program.
LSTINP	Will list the input index list and show the page numbers including those substituted for the ## page number symbols.

The following figures show the sample files that correspond to figure 7.7 above.

The first box shows what your input file should look like. This program does not decide what information should go into your index. This you have to type in yourself. The second box shows the file that the program creates for you. Finally the third shows the final output.

```
#_
#1
Preface
Introduction
#2
SCRIPT
Editor_Commands
Commands, Editor
#3
SCRIPT
Input_Commands
Commands, Input
SCRIPT Control_Words
Control_Words
#4
Control Functions
LOCATE Editor Command
```

Figure 7.8 - File Called Index.data

```
      .nf
      .sk 2
      .cp 5
      C
      .sk 1
      Commands, Editor  2
          Input  3
      Control Functions  4
      Control Words  3
      .sk 2
      .cp 5
      E
      .sk 1
      Editor Commands  2
      etc....
```

Figure 7.9 - Input SCRIPT File Called Index.new

```
C

Commands, Editor  2
    Input  3
Control Functions  4
Control Words  3

E

Editor Commands  2

I

Input Commands  3
Introduction  1

L

LOCATE Editor Command  4

P

Preface  1

S

SCRIPT  2,3
    Control Words  3
```

Figure 7.10 - Output SCRIPT File Called Index.new

Remind

The REMIND program allows you to set a reminder of a future event. If you have reminders for the day, they appear on the screen when you request the TODO facility.

To access this program press either F2 (Todays Reminders) or enter the REMIND command from the TODO main menu. When F2 is pressed, reminders for that day are listed and then you are returned to the TODO menu. After entering the REMIND command, the following is displayed on the workstation:

```
------------------------- Remind Facility -------------------------

To have the system remind you of an event,
enter in the date of the reminder and the
message you wish the system to display.            TIME: 11:49 am

Date: ____________________________________        1989  FEBRUARY 1989
 (e.g.*, *+2, Mon, 19Aug87, *Tue Thu, *1 16)
Reminder text:                                    S  M  T  W  T  F  S
____________________________________________                 1  2
____________________________________________      3  4  5  6  7  8  9
____________________________________________     10 11 12 13 14 15 16
____________________________________________     17 18 19 20 21 22 23
____________________________________________     24 25 26 27 28
____________________________________________
                                                    Day of year: 49
   Expiration Date:_____________________
                        (e.g. 22Mar89)
PF3:End/Save ======================================== PA1:End/No Save
PF1:Help    PF2:New   PF4:Add    PF5:Delete   PF6:Change    PF9:Locate
PF7:Previous     PF8:Next      PF10:Previous Month     PF11:Next Month
```

Figure 7.11 - Display for Setting Reminders

When creating a reminder, you must specify the date of the reminder and the text. If you have specified a repeating reminder (see below), you must also specify an expiration date. Use the NEW LINE key to end each line (skip to next field) when entering information on the screen.

Date — Specifies the date you wish the reminder set for. It takes one of the following forms:

*	The reminder is for the current day.
*+2	The reminder is for the day after tomorrow.
22Aug87	The reminder is for the August 22, 1987.
Thu	The reminder is for the up coming Thursday.
*Tue	The reminder is for every Tuesday.
*1 16	The reminder is for the 1st and 16th day of every month

The last two items above are known as repeating reminders. That is, the system continues to present you with a reminder until the expiration date you have specified is reached. If an asterisk (*) appears directly before a day or a number then this indicates a repeating reminder. If an * appears in front of a blank or a + then the asterisk indicates the current day or current day plus a number (the number can be from 1 to 366). You may specify up to 16 different dates or 7 different days of the week for a repeat reminder.

Remind Text Enter the message that you wish the system to display on the selected date. The system does not perform any type of re-formatting of the information that you enter. It will be displayed exactly as you have entered it.

Expiration Date

If you had specified the DATE as a repeating date, then you must specify an expiration date. If you wish the reminder never to expire, then specify 99 as the expiration date. To specify that a repeat reminder is to expire on March 22, 1989 you would specify: 22mar89

Note: The expiration date must be greater than the current date.

Program Function Key Definitions

PA1 END/NO SAVE
Terminates the reminder facility without saving the changes and additions that have been made to the reminder file.

F1 HELP
Provides help for Remind.

F2 NEW
Displays a blank reminder entry screen with the fields filled in with underscores.

F3 END/SAVE
Terminates the reminder facility and saves the reminder file with the changes and additions that have been made.

F4 ADD
Add the displayed reminder to the reminder file.

F5 DELETE
Delete the displayed reminder. Display the reminder you want to delete and then press F6.

F6 CHANGE
Change the fields in the displayed reminder. First display the reminder, then make the changes and then press F6. Now the old reminder is changed.

F7 PREVIOUS
Display the previous reminder in the remind file. (Note: the users remind file is in the order as reminders are entered.)

F8 NEXT
Display the next reminder in the remind file. (See note for F7.)

F9 LOCATE
Locate the text in a reminder that was specified in a previous LOCATE command.

F10 PREVIOUS MONTH
Display the calendar for the previous month.

F11 NEXT MONTH
Display the calendar for the next month.

Reminder Commands

The following commands can be entered in the COMMAND area at the top of the screen. Many of these commands are in the form of function keys and have been explained previously.

ADD (F4)	EXEC	NEXTMON (F11)
AFTER x	EXPIRES x	NEW (F2)
BEFORE x	GET x	OFF
BOTTOM	HELP (F1)	PFnn
CANCEL (PA1)	KEYS	PFnn def
CHANGE (F6)	LDATE x	PREVIOUS (F7)
CLEAR	LIST	PREVMON (F10)
DELETE (F5)	LOCATE string	TOP
DELIM char	MAIL	*-n
END (F3)	NEXT (F8)	

X represents a date that can be specified in a number of formats. See a later help screen for these formats.

AFTER x — Locate and display the next reminder dated after the date specified by *x.*

BEFORE x — Locate and display the previous reminder dated before the date specified by *x.*

BOTTOM — Go to the bottom of the remind file and display the reminder.

CLEAR — Clear the screen input areas.

DELIM char — Change the multiple command delimiter to *char*, where *char* is a character of length 1, and not one of the characters a to z, 0 to 9, *, =, /.

EXEC — Pass the command to MUSIC to be executed.

EXPIRES x — Display the next reminder with the expiry date given by *x.* Note that only repeat reminders have expiry dates.

GET x — Lists all of the reminders for the date given by *x.*

KEYS — Display a screen which allows you to change the function key definitions. These changed definitions are saved in a file, REMIND.KEYS, so that they can always be used.

LDATE x — Display the next reminder with the specified date.

LIST — Display a list of existing reminder dates.

LOCATE string — Display the next reminder with the specified *string* found in the reminder text.

MAIL — Post a message if mail are waiting or not waiting.

OFF — Save the changes and additions made to the remind file and sign off.

PFnn — Show the definition of PFnn, where *nn* is a number from 1 to 24.

PFnn def — Set the definition of PFnn to *def,* where *nn* is a number from 1 to 24. The definition can be from 1 to 50 characters long.

TOP — Go to the top of the remind file and display the reminder.

*	Display the last command entered in the command area.
*-n	Display the previous *nth* command entered in the command area, where *n* is a number from 0 to 4.

Possible Formats used for a Date Given by x

blank	if *x* is not specified, *x* is set to today's date.
* or *+n	*x* is today's date plus *n* if *n* is given, where *n* is a number from 0 to 366.
DDD	DDD is the 3 letter day of the week, and *x* is the next DDD. (Example if *x* is TUE and today is monday, tuesday's date is used.)
DDMMMYY	*x* is the date given by DDMMMYY (example 11FEB89).
YYYYMMDD	*x* is the date given by YYYYMMDD (example 19890211).
99999999	*x* is a non-expiring repeat reminder (used for EXPIRES command).

TMENU - Tailoring the User View

TMENU is the name of the program on MUSIC for creating tailored user views of MUSIC. This facility is available for making your own menus, or tailoring existing menus to better suit your needs.

This section is of a technical nature and is intended for experienced MUSIC users. If you do not wish to change the TODO menu or create menus then skip this section.

Some of the available features include filtering the commands to which users of this view have access, signing off from MUSIC when the menu is terminated, and executing your own programs. Programs that are used often can be added to the menu or items on this menu can be exchanged for other programs offered at your installation. For example, the sample menu TODO.MENU can be added to or altered so that when the sample program TODO is executed, these facilities you defined are available to the users of this menu.

Making a Menu

The menu is used to provide four functions:

1. It provides TMENU with the menu items and descriptive text.

2. It provides TMENU with the names of program files to be executed as required by the user.

3. It provides labels for the menu items for menu selection.

4. It provides default parameters for the respective programs.

Example:

The following figure shows the MUSIC file $TDO:TODO.MENU. This file is used each time the TODO menu is displayed on your screen.

```
)+ - TO DO
)% - TIME, OFFICE, AND DOCUMENTATION ORGANIZER -
1. Schedules
2. Electronic Mail <option>
3. Telephone Log
4. Calculator <calc>
5. Spell Check Document <option>
C. Create New <filename>
R. Revise <filename>
X. Execute SCRIPT <filename>
S. Submit SCRIPT <filename> <options>
L. List File Names <options> <pattern>
M. Schedule a Meeting <options>
U. Utilities <option>
)1 $TDO:SCHED
)2 $EML:MAIL
)3 $TDO:PHONE
)4 $PGM:POLYSO
)5 $TDO:SPELL
)C $TDO:CREATE
)R EDITOR+ ;TEXT SCRIPT
)X $TDO:SCR
)S $TDO:SCRSUB
)L $TDO:TODO.LIB|S
)M $TDO:MEET|
)U $TDO:UTIL.MENU?
```

Figure 7.12 - MUSIC File for the TODO Menu

General Format of the Menu

The menu is made up of four types of specification lines and is divided into two logical components. The first component describes the menu items (selection codes) and the descriptive text. The second component describes the actual program names and their parameters if any. The program names and the menu items are connected by the *option code*. For example, item 'H' in Figure 7.13 has both a descriptive entry in component one as well as a program name and parameters definition in component two.

```
)+                            <--------  Menu start flag
)% -  TITLE OF MENU  -        <--------  Menu title line

1. Item One                   <--------
2. Item Two                             |  Option codes
P. Profile, print $ remaining           |  and descriptions
H. Help                       <--------

)1 PROG1                      <--------  Corresponding
)2 PROG2                                |  file and
)P PROFILE/PRINT$                       |  parameter
)H HELP                       <--------  specifications
```

Figure 7.13 - Sample Menu for User View Tailoring

)+ is the marker that indicates that the file being processed is a menu file. It must be the first line of a menu file and appear only once. An optional 1-16 character menu name can be placed on this line. This text will be placed on the extreme right of the title line of the menu.

)% is the title marker. The title marker provides a title for the menu posted on the screen. A blank must follow the "%" character. If more than one title line is present, the last one will be used.

The title itself will be centered by the menu formatter of TMENU, therefore do not attempt to center it. The standard that should be followed is to place a dash (-) before and after the title text separated by one or two blanks. Example:

```
)% - title -
```

The title will be centered and padded by dashes.

```
---------- title ----------
```

xx. this is the one or two character combination that the user will know the item by. This becomes the item option code or name.

The option code character(s) must be directly followed by a period and a blank (.).

General Format of the Option Specification Line.

Type 1

```
xx. OPTION  -  Description
```

xx.	is the 1 or 2 character option code followed by a period and a blank.
OPTION	is the option name (usually in upper case).
-	is an optional delimiter.
Description	is the option description.

Type 2

```
xx. Description
```

xx.	is the 1 or 2 character option code followed by a period and a blank.
Description	is the option description.

Guidelines for Specifying Option Lines

Option Code

- The option code can be numbers or characters except period and close parenthesis.

- Do not use roman numerals, arabic numerals are more familiar and more easily distinguishable. When a short number of items are used, letters indicative of the function performed by the item is appropriate, for example, 'P' for Profile and 'H' for Help.
- If letters and numerals are to be mixed, it is best to place all the numeric option lines in sequential order followed by the letter option codes.
- The option lines should be in the order of greatest anticipated usage (not withstanding the previous point). This will facilitate usage.

Option Name

- The option name is not always used or appropriate, however when this format is used it should be in upper case. (Type 1)

Option Description

- The option description should begin with a capital letter.
- The description should be a statement with correct syntax. Statements that seem to be questions should be avoided.

```
        1. Submit a File
        11. Submit a File
        H. Help on File
        1. HELP  -  Get help on file usage
```

Figure 7.14 - Sample Option Specification Menu

Program Specification Lines

Besides the program name and the default parameters (program options), two other types of information can be included on the program specification line. The first is whether the user's parameters are to override, not override, or be appended to the default parameters. The second is whether the program is to be an 'always' program, a noncancellable program, or just an ordinary program (that is neither of the first two characteristics).

```
  )xx programYparametersZ
```

)xx — The required close parenthesis indicates a program specification line. *xx* is a 1- 2-character item name or option code. There should be as many of these lines as there were item specification lines. These lines define the files to be executed and the default parameters if any.

program — is the file name to be scheduled for execution.

Y — is one of 4 optional delimiters (| + - >).

parameters — is the optional parameter list.

Z is one of 7 optional program types (blank % ! " ? / <).

Defining Parameter Processing and Passing

The parameters are separated from the program name by one of four special characters. These characters are shown in Figure 7.15.

```
|  vertical bar   - user if specified, otherwise
                    the default parm
+  plus sign      - user parm prefixed to defaults
-  minus sign     - user parm appended to default
>  greater than   - user parm ignored
```

Figure 7.15 - Parameter Processing Flags for User View Tailoring

Each of these serves to delimit the program name from the parameters as well as to indicate the parameter handling option.

The following defines and describes these four delimiter characters.

| The vertical bar indicates that any user specified parameters are to override the default parameters. That is either the user's parameters or the default parameters are used and not both.

+ The plus sign indicates that any user specified parameters are not to override the default parameters. Rather the user parameters are to be prefixed to the default parameters. That is both specifications are passed. The two parameter lists will be separated by a blank.

- The minus sign indicates that any user specified parameters are not to override the default parameters. Rather the user parameters are to be appended to the default parameters. That is both specifications are passed. The two parameter lists will be separated by a blank.

> The greater than sign indicates that any user specified parameters are to be ignored. Only the default parameters are passed.

Defining Program Types

One of five characters can be used to describe the program type. These characters and their functions are given in Figure 7.16.

```
" " no type flag     -  "nonalways"/cancellable
 %  percent sign     -  "always"/noncancellable
 !  exclamation mark -  "nonalways"/noncancellable
 "  double quote     -  "always"/cancellable
 ?  question mark    -  menu file
 /  slash           -  MUSIC command
 <  less than sign  -  TMENU command
```

Figure 7.16 - Program Type Flags for User View Tailoring

Any one of these can appear as the last character of the parameter list to indicate the program type. These

program characters, when specified, must be appended to the parameter list on the program specification line. It will be converted to a blank and is not considered as part of the parameter list. If no parameter list or *parameter option* character (delimiter) is required or used the *program type* character can be appended to the program name.

The scan for the program type is done from right to left. Therefore the *program type* characters can appear as part of the parameter list provided that a program type character is used at the end of the parameter list.

The following defines and describes these five characters.

" " A blank or no program specification indicates that a program is to be a cancellable nonalways program. When no program type character is specified " " is the default.

% The percent sign indicates that a program is to be a noncancellable always program.

! The exclamation mark indicates that a program is to be noncancellable and nonalways.

" The double quote mark indicates that a program is to have the *always* attribute and is to be cancellable. In brief an *always* program is one that is automatically reinvoked after it has completed running. As well if it scheduled a program via a NXTPGM call, at the completion of the scheduled program the always program is reinvoked.

? The question mark indicates that the program name is really a menu file.

/ The slash indicates that this is a MUSIC command string. Any commands listed with filter are not allowed here. See the TMENU FILTER option.

< The less than sign indicates that this is a TMENU command.

```
)3 GORK|PARMS! <-- default parms are overridable, the program
                   is noncancellable and nonalways.

)3 GORK!       <-- default parms are overridable, the program
                   GORK is noncancellable and nonalways.

)3 GORK+PARMS% <-- user parms will precede default parms, the
                   program is cancellable and always.

)3 GORK-PARMS  <-- user parms will follow default parms, the
                   program is cancellable and nonalways.

)3 GORK>PARMS" <-- no user parms, default parms only and gork
                   is an always and noncancellable program.

)3 GORK>PARMS  <-- no user parms, default parms only, gork is
                   cancellable and nonalways.

)3 GORK?       <-- the file name is a menu to be displayed,
                   and not a program.

)3 NEWS/       <-- indicates that news is a MUSIC command and
                   will be executed as such.

)3 END<        <-- indicates that the user will be returned to
                   *Go or the calling program.
```

Figure 7.17 - Sample Program Specification Menu

Invoking TMENU

After a menu has been created you must set up an exec file for it. For example, the following is the EXEC file required for TODO:

```
/INC TMENU
MYNAME='TODO'
```

TMENU Options

ACCESS='filename' Where *filename* is the name of the file containing a list of userids and access levels. Userids in the file can include wild characters. The access level can be any number from 0 to 9999999 but access is granted only for the value 1. If a level is not included with a userid entry then no access is assumed.

When a user requests a menu facility, his userid is matched against the ACCESS file (if it exists). The first userid in the file that matches his userid determines whether he will have access or not.

Note: If the ACCESS parameter is not specified, then all userids have access to that menu facility.

Example:

```
* This is a sample access file
ccfp 0       Keep this guy off this facility
????xxx 0    His friends cannot use it either
??????? 1    Everyone else can
```

In this example the userid *ccfp* (ccfp 0) and all userids of the length 7 and ending with *xxx* (????xxx 0) are not allowed access. All other userids have access (??????? 1). Notice that comments can be included by using an asterisk (*) in column one or they can be appended after the userid and access level.

CAL=t or f — Specify either t or f to indicate true or false respectively. If CAL=t is used a calendar is posted on the right of the menu. If CAL=f is used no calendar is displayed. When CAL is not specified the default is CAL=t.

CANCEL=t or f — When CANCEL=t is specified the user can return to *Go by PA1, /CANCEL ALL, or successive END commands or F3's. However when CANCEL=f, /CANCEL is rejected and normal termination of TMENU results in a sign-off. In other words CANCEL=f prevents users from returning to *Go. The default is CANCEL=t.

ENDCMD='filename' — Where filename is the name of the file to be executed when the user exits the facility. ENDCMD is honoured only when CANCEL=t. filename could be an accounting program that records usage of the facility.

ERRFIL='filename' — Where *filename* is the name of a file that contains the error messages for the program. The default file name is $TDO:TMENU.MSGS.

FILTER=t or f — When FILTER=f, any command string which is not an option code or a command used by TMENU is passed to the MUSIC command processor. When FILTER=t, the option codes, the commands used by TMENU, and only the commands and/or programs listed below the parameter line are allowed. The default is FILTER=f.

A number of commands processed by the MUSIC terminal scanner can not be issued from TMENU. These commands are: /COMPRESS, /CTL, /DISCON, /NS, /PAUSE, /PROMPT, /REQUEST, /RUN, /STATUS, /TIME, /TEXT, /TABIN, /TABOUT, /USERS, and /WINDOW.

The commands can be entered with any abbreviation up to the minimum abbreviation as specified by an integer to its right. If no minimum abbreviation is specified the command will have to be entered in its entirety as specified. Before the command string is passed the abbreviated name is expanded to its full length. For example, the following abbreviations would be allowed for this list of commands:

Command	Abbreviations
PURGE 2	purg, pur, pu
OUTPUT 3	outpu, outp, out
SUBMIT 2	submi, subm, sub, su
PRINT 3	prin, pri

If no commands and/or program file names are specified, then the "/" command is not allowed at all. This is the method for barring the use of the "/" command. For example, in the file ABC:

```
/INC TMENU
```

```
MYNAME='ABC',FILTER=T
```

ABC will not allow the use of the "/" command and will use the default menu of TODO.MENU as a first menu. The calendar will be posted on each menu screen. The stack file will be @TMENU.STACK.tcb where *tcb* is the current TCB number.

FMENU=menu — Where *menu* is the name of the first menu file to be displayed. When no menu file is specified the default menu is used.

FS=t or f — When FS=t and TMENU is invoked on a 3270-type workstation, automatic full screen support is provided. On a non full screen workstation such support is never allowed.

When FS=f, no matter what workstation type you are using, only non full screen support will be available. The default is FS=t.

KEYFIL='filename' — Where *filename* is the name of a file that contains the site definitions for the function keys used in the program. Lines in this file have the following format: "PFnn x" where *nn* is a number from 1 to 24, and *x*, the definition, can be from 1 to 50 characters.

Users can store their own function key definitions for later use by entering a definition for any function key. These definitions are stored in the file *USR:TMENU.KEYS at the program termination for use when the program is invoked again. If this file is available when the program begins, then these definitions are used and the site definitions are not. Lines in the file *USR:TMENU.KEYS have the same format as the lines in the KEYFIL filename.

LOG=t or f — The LOG=t parameter can be used if you wish to keep a usage log. LOG=f is the default.

LOGFIL='filename' — Where *filename* is the name of a file that is used to log information from the program. The default file name is *USR:@TMENULOG.tcb, where *tcb* is the current TCB number.

MDELAY=n — Where *n* is a value indicating the delay between attempts at finding out if mail is waiting. When mail is detected, a mail waiting message is posted. When *n* is a positive integer it is taken as representing minutes. When *n* is negative it is taken as seconds. To stop all mail checking, use a value for *n* that is greater than 24 hours such as -86400 secs or 1440 mins. This will prevent the posting of even the initial mail waiting message at the start of the program. The default is MDELAY=-300 or MDELAY=5 (five minutes).

MYNAME='filename' — Where *filename* is the name of the file. In our example it would be TODO. If this parameter does not specify the file name in which it is stored TMENU will not make the file an *always* program.

REMS=t or f — When REMS=t, TMENU will automatically display any reminders for the current day. You can set reminders and query for the day's reminders with the REMIND command or the use of F2. The default is REMS=t.

When REMS=f the REMIND program will not be available, and the REMIND command and F2 will be ignored. The default is REMS=t.

STACK='somefile' — Where *somefile* is a 1-12 character name of a file to be used as the stack file. The tcb number is appended to it as follows: *somefile.tcb*. The stack file is used to store

a trace of menus and programs that have been called. In this way the program can return to the correct menu as it drops back through the stack. When no stack file is specified the default "@TMENU.STACK.tcb" is used, where *tcb* is the TCB number.

Builtin Functions

Command	Description
CANCEL	exit the current menu.
END	exit the current menu.
EXEC	pass the string to MUSIC as a command (abbr. /). See the FILTER option.
GETREM	post message reminders for today or no reminders.
HELP	provide help on how to manipulate the menu screen.
KEYS	post the KEYS screen to enter definitions for the function keys.
MEMOS	post message mail waiting or not waiting.
OFF	terminate the program and sign off.
PFnn	show the definition of PFnn, where *nn* is a number from 1 to 24.
PFnn x	set the definition of PFnn to *x*, where *nn* is a number from 1 to 24. The definition can be from 1 to 50 characters long.
=x	return to the first screen and process *x*.
REMIND	view reminder file (abbr. REM).
*	display the last command entered in the SELECT OPTION area.
*-n	display the previous *nth* command entered in the SELECT OPTION area, where *n* is a number from 0 to 4.

Program Function Key Definitions

Key	Definition
F1	provide help on how to manipulate the menu screen.
F2	check today's reminders.
F3	terminate the current menu.
F6	post message mail waiting or not waiting.
PA1	terminate the current menu.
ENTER	process the command area.

IDP (Information Display Program)

The IDP (Information Display Program) facility is used to create and run bulletin boards, help facilities, and ads facilities. This facility can also be used to build menus, as topics can invoke text or programs.

Refer to the *MUSIC/SP Campus-Wide Information Systems* (*CWIS*) *Guide* for complete information.

Appendixes

Appendix A - Using SCRIPT without TODO

The following example shows the steps necessary to use SCRIPT without the TODO facility.

```
1.      *Go
2. -->  tedit sample new

3. -->  /include script
4. -->  OKERR
5. -->  This is a SCRIPT document called "sample", which
   -->  will be formatted at "execution" time.
   -->  Type "exec sample" or simply "sample" when you
   -->  are in *Go mode to execute.
   -->  A shortcut for executing files from the Editor is
   -->  to enter the Editor command EXEC.
   -->  This is equivalent to giving the Editor command FILE
   -->  and then issuing the MUSIC command "EXEC sample".

6.      (press F12 to go to the command area)

7. -->  file
8.      SAMPLE
        NEW FILE
        SAVED
        *End
        *Go

9. -->  exec sample

10.-->  MUSIC/SCRIPT...ENTER OPTIONS OR HELP
        ?
11.-->  x
```

1. *Go
   This message appears whenever you are in command mode of MUSIC (*Go mode). Enter MUSIC Immediate commands at this time.

2. tedit sample new
   "tedit" is the MUSIC command to invoke the Editor. TEDIT stands for text edit. "Sample" is the name of the file in this example. "New" tells the Editor that this is a new file, and the Editor will go directly to INPUT mode.

3. /include script
   This must be the first line in the document. It tells MUSIC to use the SCRIPT program when the file is executed.

4. OKERR
   The second line in your file is for SCRIPT output options or it can be blank. Options need to be in uppercase.

5. This is a ...
   The text and control lines begin on the third line in your file.

6. (press ...
   F12 is the function key to bring your cursor to the command area.

7. file
   FILE is the Editor command to store your document.

8. SAMPLE...
   These messages inform you that your document has been saved. MUSIC returns to *Go mode.

9. exec sample
   This MUSIC command executes the file "sample". In other words, you wish to see the formatted document.

10. MUSIC/SCRIPT...ENTER...
    The SCRIPT program has now taken control and prompts for additional options.

11. x
    Type "x" to inform SCRIPT to read options from document. Additional options can be added to this line and separated with commas.

Appendix B - Sample Edit Session, Non-Full-Screen

The sample Editor session below shows the usage of some commands for the Line Editor. It is strongly recommended that you become familiar with all the Editor commands we described. Pretty soon you will be using it like an expert!

In the example below the lines typed by the user are identified by a -> symbol immediately to the left of them. This symbol is just used here to point these lines out; it will NOT appear when you use the Editor.

```
      Edit Session                    Remarks

  *Go
->tedit sample              <----Edit the file
  *In Progress                   named "sample".
  EDIT

->p *                       <----Print the file
  This is a sample file that we
  will use to demonstrate the
  functions of the Editor
  The Editor is a
  good thing
  *EOF                      <----End of File
->top                       <----Go to the top of the file
->l func                    <----Locate string "func"
  functions of the Editor   <----The Editor prints out the
                                 line it found.
->c/Ed/MUSIC Ed/            <----change the string "Ed" to
  functions of the MUSIC Editor    say "MUSIC Ed"
->u                         <----Go up one line
  will use to demonstrate the
->a  various                <----Add a word to the end
                                 of the line
  will use to demonstrate the various

->last;input                <----Go to the last line and
  good thing                     enter input mode.  Note we
  INPUT                          combined 2 commands in one.
->We will not show all the  <----Now we can proceed to type
->uses of the change           in new lines
->command here.
->We suggest that you
->experiment with these
->commands yourself.
->                          <----Blank line entered.
  EDIT                      <----We are now out of input mode
```

```
->t;f The Edi                <----Go to the top and find the
  The Editor is a                 sentence starting with
                                  "The Edi".  (This will only
                                  work if you started that
                                  sentence on a new line like
                                  we recommend.)
->split/is/                  <----Split current line at "is"
  The Editor                 <----We are now pointing to 1st
                                  line formed by the split.
->n                          <----Go to the next line.
  is a                       <----Current line is printed.
->r on MUSIC can be used     <----Replace the current line
                                  with the new one given.
->i to great advantage.      <----Insert a new line after the
                                  current one.
->n                          <----Go to next line.
  good thing
->del                        <----Delete it
  We will not show all the   <----The Editor prints the new
                                  current line.
->t;p *                      <----Now let's print the file
  This is a sample file that we      before we save it.
  will use to demonstrate the various
  functions of the MUSIC Editor.
  The Editor
  on MUSIC can be used
  to great advantage.
  We will not show all the
  uses of the change
  command here.
  We suggest that you
  experiment with these
  commands yourself.
  *EOF
->file                       <----Tell the Editor to save the
  SAMPLE                          updated file in place of the
  REPLACED                        old one.
  *End
  *Go
```

Appendix C - Common Error Messages

MUSIC Command Error Messages

*** FILE NAME IS INVALID

The file name has been misspelled. A file name was used that did not follow naming conventions.

*** FILE NOT FOUND

You have asked to edit a file that does not exist in your Save Library or the Common Public Library.

name ERR11 FILE NOT ACCESSIBLE

You have tried to execute a file that does not exist in your Save Library or the Common Public Library.

Editor Error Messages

INVALID COMMAND

The command you entered is misspelled or invalid.

MISSING OPERAND

There was no option with your editor command. Examples: 'LOCATE' should be 'LOCATE string'; 'MERGE' should be 'MERGE name'.

OPERANDS ARE NOT ALLOWED ON THIS COMMAND

No options are allowed with this editor command. Example: 'TOP 6' should be 'TOP'.

INVALID OPERAND

The option you typed with this editor command is invalid. Example: 'FILE 88' will be wrong because 88 does not follow the rules for naming files.

MISSING STRING DELIMITER

A delimiter is used to separate options in your edit commands. In the following examples the slash is used. Example: 'SPLIT/string' should be 'SPLIT/string/'. Also check that the command delimiter (;) was not used within the string.

SCRIPT Output Error Messages

UNKNOWN CONTROL WORD ENCOUNTERED...

A control word in your document is invalid or misspelled. Example: '.FN' instead of '.NF'.

INVALID PARAMETER IN FOLLOWING CARD...

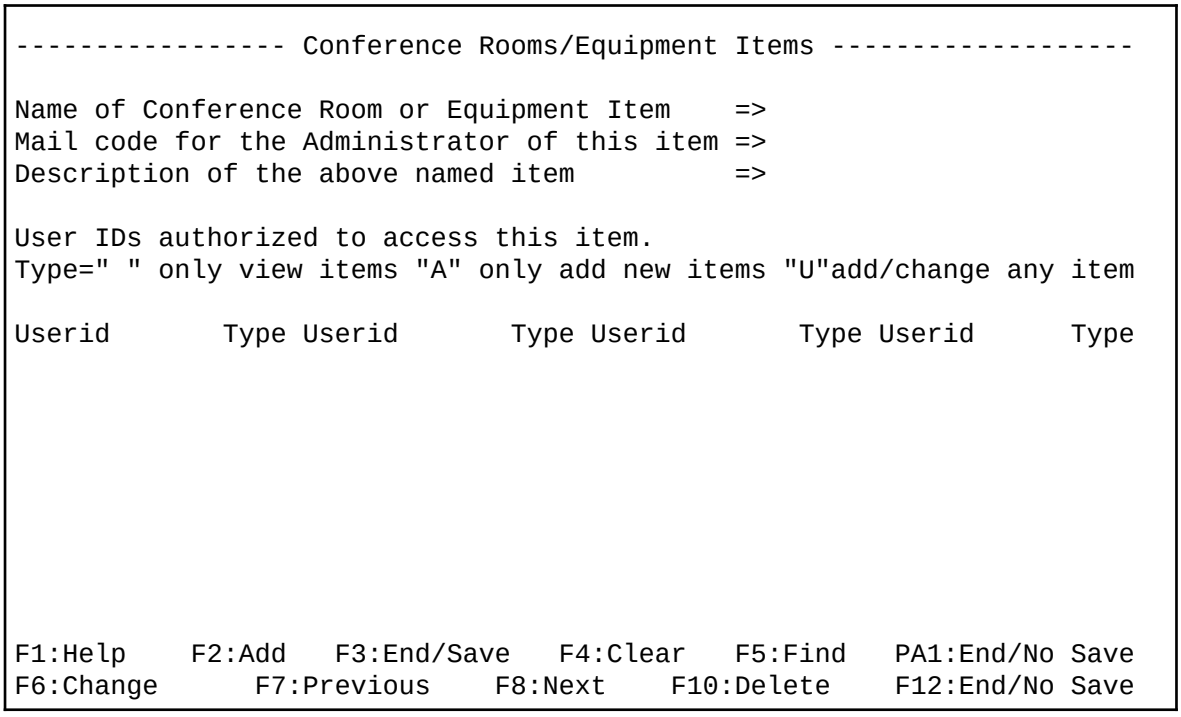
Incorrect option has been found in a SCRIPT control line. Examples: '.SP2' should be '.SP 2'; '.LL 140' should be '.LL 130' or less.

FILE NOT ACCESSIBLE...

SCRIPT cannot get the file name you specified on the .IM control word because it is not in your Save Library or the Common Public Library.

Appendix D - Conference Rooms/Equipment

The program CONFLIST is available to initialize conference room and equipment items. Only privileged codes are authorized to add items to the Conference Room and Equipment List. This program is also used to give other users authorization to: view, add, or update the calendar for these items. See the section on SCHEDULE in *Chapter 5 - TODO Menu Items* for viewing and updating calendars.



This program is used to Create/Update the list of valid Conference Rooms and or Equipment items for your location. In addition, the User IDs which are authorized to view or schedule these items, is also specified.

ITEM Name — Sixteen character name used to identify this entry. Each entry in the file must be unique. A user will request a Conference Room or Equipment item via this name.

Mail Code — Userid of the person who is acting as the Administrator of this item.

Description — Thirty character description of the item. When the user selects an entry from this file, this field will be displayed for the user at the workstation.

Authorized Userids IDs — This is a two part field which describes the Userids which are authorized to access this item and also the type of access allowed.

Userid — The Userids can take one of two forms. The standard MUSIC userid, or a Userid which uses the 'wildcard' operator.

The "?" and "*" characters are the wildcard operators. You can use these characters to allow greater flexibility in authorizing access to Conference rooms and Equipment items.

When the wildcard character or "?" is used, it means that any character will match in that position. The wildcard character "*" will match any or no characters in that position. For

example:
CX* would authorize any userid starting with "CX".
??00 would authorize any four character userid which ended with two zeros.
G*VV would allow any userid starting with G, ending with VV.
* would authorize any userid.

Note: The valid characters for Userids are: (A-Z, $#@_?*) in the first position and (A-Z, $#@_?*, 0-9) for the second through to the sixteenth positions.

Type field " " or "A" or "U"

This field defines the type of access for that Userid. If it's blank, the user is allowed view authority to the item. If the type is A, the user can add new items or change those he has added. If the type is specified as U, the user will be allowed to view and update any item.

Program Function Keys for Conference and Equipment List

F1 HELP
Provide help.

F2 Add
Add this entry to the file. This named item must NOT presently exist in the file.

F3 End/Save
Exit from the facility saving any changes made to the file.

F4 Clear
Clear of all of the input fields.

F5 Find
Locate in the file the specified ITEM name.

F6 Change
Update this entry with the changed fields.

F7 Previous
Display the previous entry in the file.

F8 Next
Display the next entry in the file.

F10 Delete
Erase this entry from the file.

F12 Quit
Exit from the facility without saving any changes made.

PA1 End/No Save
Exit from the facility without saving any changes made.

Appendix E - Create Program

This appendix includes a listing of the program CREATE. This program invokes the Editor for typing new documents, and has features for an automatic generation of SCRIPT documents for letters and memos. Return addresses and sender information can be changed or added to this program.

The file (CREATE) is public and can be tailored to individual needs. Sections that are in italics indicate areas that can be changed by the user or by the system administrator. Comments are included in the program to explain each of the steps. These comments begin with "/*" and end with "*/".

The user can make changes to this program and save it on their own code and name it "CREATE". If another name is given then this program will not be accessible by the TODO menu unless the menu is changed. (See the section on TMENU in *Chapter 6 - Utilities* for changing the menu.)

The system administrator can change the public program CREATE and it would affect all TODO users.

Create Program Listing

```
/LOAD REXX
/* NOTE TO READER: Use the information contained in the comment
   lines of this file to help you understand the logic of the
   REXX commands in this program.
   You can copy this file and modify the command statements
   if you wish, to tailor the file to your own needs.
*/

/*Now get the filename the user passed to the program.
  This is done by the following REXX "parse arg" statement.
  Note that in this procedure, the filename is seen to be in
  capital letters.

*/

parse arg filename .

if filename=' ' then do
    say '0No file name given on CREATE command.'
    say ' Type in HELP for information.'
    say ' Enter a blank line if you do not want help.'
    say ' '

    pull filename .

    if filename = '' then exit

   end

/* Change all dots (.) in the filename to blanks, and
   check to make sure that only valid filename characters are used.
*/
```

```
    if verify(filename,'ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890$#@.') ¬=0
        then do
           say '0Filename contains invalid characters'
           say '0Valid characters are: alphabetic characters a-z'
           say '                       numeric characters 0-9'
           say '                       the 4 special characters $ # @ .'
           say ' Note: The first character should be alphabetic'
           exit
         end

    if verify(left(filename,1),' ABCDEFGHIJKLMNOPQRSTUVWXYZ$#@') ¬=0
        then do
           say '0Filename cannot start with a number or '
           say ' a period (.)'
           exit
         end

/* Now the filename will be considered as consisting of several
   parts or "words", for the internal use of the program.
   Each word is separated by a dot. Replace each dot with a blank.
   For example, the filename "L.ABC.DE.F" becomes the words
   "L ABC D F". The first three words determine the file type
   (a letter in this case), the recipient, and the sender.
*/

      x = translate(filename,' ','.')  /* change periods to blanks */
      xtype = word(x,1)   /* set xtype to 1st word of string x*/
      xto   = word(x,2)
      xfrom = word(x,3)

/* Now get the date in a format suitable for letters and memos.
   This is the desired format  "January 10, 1985".
   Because MUSIC uses a different date format, some REXX commands
   are necessary to modify the date to produce the desired format.
*/
       x = date()   /* date in format "10 Jan 1985"  */
       mydate =date(m)' 'subword(x,1,1)', 'subword(x,3,1)

/* The MUSIC sign-on code of the person running this file
   is obtained from the REXX function "userid".
*/


select

/* help wanted */
  when xtype='HELP' then do

     say '0The CREATE function uses the file name to determine'
     say ' the type of file being created.'
     say ' It prepares a standard setup for each type of file.'
     say ' All you have to do is fill in the details.'
     say ' '
     say ' For example, a letter file (L) going to a company with the'
     say ' initials INC from a person with the initials ME would have'
```

```
     say ' a file name that looks like this  L.INC.ME'
     say ' Similarly, to create a memo (M) file to INC from ME'
     say ' you would use a file name that looks like this M.INC.ME'
     say '0You can instruct CREATE to recognize the initials and'
     say ' fill in the full names when it sets up the file for you.'
     end

/* Letter file to be created */
  when xtype='L' then do
     queue '/PARM 'filename' NEW'
     queue '/inc editor'
     queue 'text script'
     queue 'i /inc script'
     queue 'i .tm 0'
     queue 'i .ll 70'
/*
   The following .sp 15 allows for 15 spaces at the top of the letter.
   This allows for the use of pre-printed letterhead.
   If you are not using pre-printed letterhead, include your company's
   name here, and reduce the .sp 15 control word to accommodate it.
*/
     queue 'i .sp 15'
     queue 'i .bm 6'
/* Allow for page number and date to be printed on additional pages*/
     queue 'i .tt('
     queue 'i .sp 5'
     queue 'i Page &'
     queue 'i .br '
     queue 'i 'mydate
     queue 'i .sp 4'
     queue 'i .)'
     queue 'i .in 0'
/* end of information for multiple page letters */
     queue 'i ' mydate
     queue 'i .sp 4'
     queue 'i .nf'
/*
  If you always send a letter or memo to the same company or person,
  fill in their full name and address here.
*/

 select
    when xto='IBM' then do
/*     Sample one filled in here.  Change it as required  */
     queue 'i Mr. IBM Salesman'
     queue 'i IBM Corporation'
     queue 'i 1234 Main Street'
     queue 'i Anytown, NY USA 12345'
    end

    when xto='INC' then do
/*     Sample one filled in here.  Change it as required  */
     queue 'i Ms. Smith'
     queue 'i INC Company'
     queue 'i 4321 Main Street'
     queue 'i Anyville, CA USA 54321'
```

```
    end

    otherwise do
/*  do this if not one of the above names */
     queue 'i .cm ===> Type in name and address following this <===='
    end
   end

     queue 'i .sp'
     queue 'i Dear'
     queue 'i .nf'
     queue 'i .co'
/*
   The following sequence is useful for paragraphs. To use it, type
   in a line that contains ..par between each paragraph.
*/
     queue 'i .df ..par'
     queue 'i .sp'      /* put in a blank line between paragraphs */
     queue 'i .cp 2'    /* make sure page holds at least 2 lines */
     queue 'i .df'      /* end of definition */
/* If you want to indent 5 spaces at the start of each paragraph,
   insert the command .un -5 right after the .cp 2 command. */
     queue 'i .cm ===> Type ..par on a separate line before each  <===='
     queue 'i .cm ===> paragraph, after the first paragraph,      <===='
     queue 'i .cm ===> to skip one line and to make sure that the <===='
     queue 'i .cm ===> page can hold at least 2 lines of paragraph<===='
     queue 'i ..par'
     queue 'i .cm ===> Type in main body of letter following this <===='
     queue 'i .sp 2'
     queue 'i .('
     queue 'i .in 0'
     queue 'i .nf'
     queue 'i Yours truly,'
     queue 'i .sp 4'
/*
  If you always send letters from the same person,
  then fill in their full name here.
*/

 select
    when xfrom='YZ' then do
/*     Sample one filled in here.  Change it as required  */
     queue 'i Mrs. Yolande Zed'
     queue 'i President'
     queue 'i XYZ Company Inc.'
    end

    when xfrom='ZY' then do
/*     Sample one filled in here.  Change as required  */
     queue 'i Mr. Zed Yoland'
     queue 'i Manager'
     queue 'i XYZ Company Inc.'
    end

    otherwise do
/*  do this if not one of the above names */
```

```
     queue "i .cm ===> Type in sender's name and title after this <==="
    end
   end

/* Fill in typist's initials at the end of the letter.

   The 3rd and 4th characters of the MUSIC sign-on code are taken
   as the typist's initials.
   The initials are translated into lower case letters.
   If this is not your convention, it can be easily changed.
   (The MUSIC sign-on code of the person running this file
    is obtained from the REXX function "userid".)

*/
     x=translate(substr(userid(),3,2),'abcdefghijklmnopqrstuvwxyz' ,
                                      ,'ABCDEFGHIJKLMNOPQRSTUVWXYZ')
     queue 'i .sp 3'
     queue 'i 'xfrom'/'x
     queue 'i Encl:'
           'EXEC'

     end

/* memo file to be created */
  when xtype='M' then do
     queue '/PARM 'filename' NEW'
     queue '/inc editor'
     queue 'text script'
     queue 'i /inc script'
     queue 'i .tm 0'
     queue 'i .ll 70'
     queue 'i .sp 5'
     queue 'i .bm 6'
/* Allow for page number and date to be printed on additional pages*/
     queue 'i .tt('
     queue 'i .sp 5'
     queue 'i Page &'
     queue 'i .br '
     queue 'i 'mydate
     queue 'i .sp 5'
     queue 'i .)'
     queue 'i .in 0'
     queue 'i .nf'
/* end of information for multiple-page memos */
     queue ,
 'i ____________________________________________________________________
     queue 'i'
     queue 'i'
     queue ,
 'i                 M E M O R A N D U M        Date: 'mydate
     queue 'i'
     queue ,
 'i ____________________________________________________________________
     queue 'i .sp 4'
/* if memos always from the same person, fill information in here */
     queue 'i .cm ===> Fill in the following lines as required <==='
```

```
      queue 'i To:                                  From:'
      queue 'i'
      queue 'i'
      queue 'i .sp'
      queue 'i Subject:'
      queue 'i .sp 3'
      queue 'i .nf'
      queue 'i .co'
      queue 'i .cm ===> Type in main body of memo following this <==='
      x=translate(substr(userid(),3,2),'abcdefghijklmnopqrstuvwxyz' ,
                                          ,'ABCDEFGHIJKLMNOPQRSTUVWXYZ')
      queue 'i .sp 4'
      queue 'i 'xfrom'/'x
            'EXEC'
   end

/* not any of the above */
   otherwise do
      /* If filename does not start with L. or M. then create a new
         file without any SCRIPT control words in it */
      queue '/PARM 'filename' NEW'
      queue '/inc editor'
      queue 'text script'
            'EXEC'
    end
end
```

Index

Index

B

C

D

E

F

G

H

I

J

K

N

O

P

Q

R

S

T

U

V

Table of Contents

Figures

MUSIC/SP
Office Applications Guide
April 1996

READER'S COMMENT FORM

You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that the MUSIC Product Group may use or distribute whatever information you supply in any way it believes appropriate without inccurring any obligation to you.

Possible topics for comments are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name, institution, mailing address and date:

_____

_____

_____

_____

What is your occupation?_____

Number of latest Newsletter associated with this publication: _____

Thank you for your cooperation.

MUSIC/SP Office Applications Guide (April 1996)

Reader's Comment Form

fold and tape please do not staple fold and tape

Place stamp here

MUSIC Product Group
McGill Systems Inc.
550 Sherbrooke St. West
Suite 1650, West Tower
Montreal, Quebec H3A 1B9
CANADA

fold and tape please do not staple fold and tape