

AUTOMON/BATCH™

Technical Reference Guide

Release 3.1

UNICOM Systems, Inc.
15535 San Fernando Mission Blvd.
Mission Hills, CA 91345
818-838-0606

First Edition - September 2002

*The information in this document applies to Release 3.1
of the AUTOMON/BATCH product.*

*Published by UNICOM Systems, Inc
UNICOM Plaza Suite 310.
15535 San Fernando Mission Blvd.
Mission Hills, CA 91345
Phone 818-838-0606*

Copyright © 2002.

All rights reserved

AUTOMON/BATCH

Release 3.1

TABLE OF CONTENTS

SECTION

PREFACE

INTRODUCTION.....	1	01
FEATURES AND BENEFITS	2	
FCTD TRANSACTION OPERATION	3	
DESCRIPTION OF MENU OPTIONS	3	
AUTOMON/BATCH HELP	5	
THE FILE CONTROL FUNCTIONS OF AUTOMON/BATCH.....	7	02
GENERAL OPERATION	7	
ENTERING COMMANDS FROM A CLEAR SCREEN	7	
CREATING A NEW FILE DEFINITION.....	7	
DATA ENTRY FIELDS	8	
VALIDATION OF FIELDS	8	
DUPLICATING FILE DEFINITIONS	8	
DISPLAYING AND MODIFYING A FILE DEFINITION RECORD	9	
ENTERING COMMANDS IN THE AUTOMON/BATCH RDO INTERFACE	9	
MULTIPLE FILE DIRECTORY DISPLAY	11	
ENTERING COMMANDS AT THE DIRECTORY DISPLAY	11	
MULTIPLE FILE STATUS DISPLAY	12	
ENTERING COMMANDS AT THE STATUS DISPLAY	12	
GROUPING FILES, TRANSACTIONS AND PROGRAMS	15	03
THE FAMILY DIRECTORY DISPLAY	15	
CREATING A NEW FAMILY DEFINITION	16	
DISPLAYING AND MODIFYING A FAMILY DEFINITION	16	
DELETING A FAMILY DEFINITION	16	
FILE FAMILIES	17	
THE FILE FAMILY DEFINITION SCREEN	17	
ENTERING COMMANDS AT THE FILE FAMILY DEFINITION SCREEN	17	
ENTERING COMMANDS FOR FILE FAMILY DEFINITIONS.....	18	
TRANSACTION/PROGRAM FAMILIES.....	19	
THE TRANSACTION/PROGRAM FAMILY DEFINITION SCREEN.....	19	
ENTERING COMMANDS AT A TRANSACTION/PROGRAM FAMILY SCREEN.....	20	
ENABLING AND DISABLING A TRANSACTION/PROGRAM FAMILY	20	
MIGRATION CONSIDERATIONS AND RECOMMENDATIONS.....	21	04
CONVERTING FCT ENTRIES TO THE DFHCSD FILE	21	
AUTOMATIC OPEN/CLOSE	23	05
CONTROLLING AUTOMATIC-BACKOUT - THE FCTT TRANSACTION	23	
THE AUTO-BACKOUT SCREEN	24	
ENTERING COMMANDS AT THE AUTO-BACKOUT DISPLAY SCREEN	25	

TABLE OF CONTENTS

SECTION

THE BATCH INTERFACE TO AUTOMON/BATCH.....	27	06
COMMANDS SUPPORTED IN BATCH MODE.....	27	
ENABLING THE BATCH INTERFACE FEATURE.....	28	
RUNNING THE COMMAND PROCESSOR PROGRAMS.....	28	
ESTABLISHING A DEFAULT CICS APPLICATION ID.....	28	
CONTROL STATEMENT FORMAT.....	29	
VARYING THE SERVICE REQUEST VALUES OF A DATASET.....	30	
STARTING A USER TRANSACTION FROM THE BATCH INTERFACE.....	30	
CONSIDERATIONS FOR STARTING USER TRANSACTIONS.....	31	
CONTROL STATEMENT EXAMPLES.....	32	
CONTROLLING THE BATCH INTERFACE.....	33	
THE BATCH INTERFACE SCREEN.....	33	
BATCH INTERFACE SCREEN - FIELD DESCRIPTIONS:.....	34	
ENTERING COMMANDS AT THE BATCH INTERFACE STATISTICS SCREEN.....	35	
TECHNICAL CONSIDERATIONS.....	36	
THE PROGRAMMER INTERFACE TO AUTOMON/BATCH.....	37	07
COMMANDS THAT MAY BE ISSUED FROM A USER PROGRAM.....	37	
CONSIDERATIONS FOR USING THE PROGRAMMER INTERFACE.....	37	
PROGRAMMER INTERFACE EXAMPLES.....	38	
COMMAND LEVEL COBOL EXAMPLE.....	38	
THE FILE SERVICE FUNCTIONS OF AUTOMON/BATCH.....	39	08
GENERAL OPERATION.....	40	
INVOKING COMMANDS.....	40	
DESCRIPTION OF MENU SCREEN FIELDS.....	41	
THE GENERAL PURPOSE DATA DISPLAY SCREEN.....	44	
DESCRIPTION OF DISPLAY SCREEN FIELDS:.....	44	
DISPLAYING A RECORD.....	47	
PROGRAM OPERATION.....	47	
MODIFYING A RECORD.....	49	
PROGRAM OPERATION.....	49	
SPECIAL CONSIDERATIONS FOR LOWERCASE LETTERS.....	50	
ADDING A RECORD TO THE FILE.....	52	
PROGRAM OPERATION.....	52	
RECORD TYPE CONSIDERATIONS.....	54	
DELETING A DATA RECORD.....	55	
PROGRAM OPERATION.....	55	
PRINTING A RECORD ON AN ON-LINE PRINTER.....	56	
PROGRAM OPERATION.....	56	
SEARCHING AND CHANGING MULTIPLE RECORDS - LOCATE.....	57	09
PROGRAM OPERATION.....	57	
THE INPUT SELECTION SPECIFICATIONS SCREEN.....	58	
INPUT SELECTION FIELD DESCRIPTIONS.....	58	
NOTES ON INPUT SELECTION FIELDS:.....	61	
EXAMPLES OF INPUT SELECTION.....	61	
NOTES ON INPUT SELECTION EXAMPLES.....	65	
OUTPUT DISPOSITION OF SELECTED RECORDS.....	66	
THE OUTPUT DISPOSITION INSTRUCTIONS SCREEN.....	66	
SELECTING THE OUTPUT DISPOSITION.....	66	
DESCRIPTION OF OUTPUT DISPOSITION OPTIONS.....	67	
MODIFYING AND REFORMATTING SELECTED RECORDS.....	75	
THE DATA MODIFY/REFORMAT SPECIFICATIONS SCREEN.....	75	
ENTERING MODIFY/REFORMAT INSTRUCTIONS.....	76	
STARTING THE LOCATE OPERATION.....	78	
ERROR CONDITIONS.....	79	

TABLE OF CONTENTS

SECTION

POSSIBLE ACTION AT A LOCATE PAUSE.....	80	
CONSIDERATIONS FOR DIFFERENT FILE TYPES OR ATTRIBUTES.....	81	
TERMINATING AND RESTARTING THE LOCATE OPERATION.....	82	
THE SECURITY CONTROL SYSTEM.....	83	10
ASSIGNING CICS SECURITY KEYS.....	83	
ASSIGNING PASSWORDS.....	84	
WRITING A SECURITY USER EXIT	85	
EXAMPLES OF SECURITY USER EXITS	88	
COMMAND-LEVEL COBOL	88	
MACRO-LEVEL ASSEMBLER	90	
SECURITY CONTROL FILE MAINTENANCE	92	
ADDING SECURITY CONTROL RECORDS	93	
MODIFYING SECURITY CONTROL RECORDS	94	
DELETING SECURITY CONTROL RECORDS.....	94	
ADDITIONAL COMMANDS IN SECURITY MAINTENANCE.....	94	
INSTALLATION OF THE AUTOMON/BATCH PRODUCT.....	95	11
CICS RELEASE LEVEL REQUIREMENTS	95	
INSTALLATION STEPS	95	
THE INSTALLATION TAPE	96	
DOS/VSE INSTALLATION	98	
MVS INSTALLATION.....	100	
ADDITIONAL INSTALLATION NOTES, MVS AND DOS/VSE	102	
THE CONVERSION UTILITY PROGRAM	103	
THE CICS START-UP JCL	103	
SUMMARY	104	
KEYWORDS OF THE UNICOM INSTALLATION PROGRAM	105	
FORMAT OF THE STSINST KEYWORDS	105	
KEYWORD DESCRIPTIONS	105	
INSTALLING THE PRODUCT CONTROL PASSWORD	108	
DOS/VSE USERS.....	108	
MVS USERS 109		
DATA FIELDS OF THE PASSWORD CONTROL TABLE.....	110	
CUSTOMIZATION	111	12
CODING THE USER OPTION TABLE	112	
DEFAULT VALUES FOR THE USER OPTION TABLE	115	
INSTALLING THE USER OPTION TABLE	115	
THE USER OPTION DISPLAY SCREEN.....	117	
THE USER OPTIONS DISPLAY SCREEN FIELD DESCRIPTIONS.....	117	
SPECIAL CONSIDERATIONS.....	119	13
PARTITION STANDARD LABEL AREA.....	119	
PRODUCT LIMITATIONS	119	
NEWCOPY COMMANDS FOR PROGRAM MODULES	119	
SUFFIXING THE USER OPTION TABLE.....	120	
SPECIAL PURPOSE COMMANDS.....	121	14
THE 'Z' COMMAND 121		
DEACTIVATING THE AUTOMON/BATCH PRODUCT	121	
REACTIVATING THE AUTOMON/BATCH PRODUCT.....	121	
DEACTIVATING THE BATCH INTERFACE	122	
REACTIVATING THE BATCH INTERFACE.....	122	
CONTROLLING THE AUTOMATIC BACKOUT FEATURE.....	122	

TABLE OF CONTENTS

SECTION

MESSAGES	123	15
BATCH INTERFACE MESSAGES ('FCTDB' PREFIX).....	123	
ON-LINE OPERATIONAL MESSAGES ('F' PREFIX)	128	
AUTOMON/BATCH RDO INTERFACE	140	
AUTOMON/BATCH INITIALIZATION STAGE	142	
LOCATE FUNCTION MESSAGES ('L' PREFIX)	143	
FILE SERVICE UTILITY MESSAGES ('R' PREFIX)	146	
SECURITY MAINTENANCE MESSAGES ('S' PREFIX)	152	
AUTOMATIC-BACKOUT MESSAGES ('T' PREFIX).....	155	
DYNAMIC ALLOCATION MESSAGES: ('X' PREFIX)	157	
UN-NUMBERED MESSAGES	157	
APPENDIX A. SUMMARY OF AUTOMON/BATCH COMMANDS.....	161	
APPENDIX B. SYSTEM REQUIREMENTS	163	
APPENDIX C. FILE DEFINITION RECORD FORMAT	165	
APPENDIX D. INSTALLATION DOCUMENTATION	167	
APPENDIX E. REPORTING PROBLEMS	175	
APPENDIX F - THE PRODUCT LINE.....	177	
INDEX	181	

PREFACE

This preface is designed to provide a quick overview of the contents of the AUTOMON/BATCH reference guide. Each major section of the manual is briefly described, along with the starting page number for that section.

INTRODUCTION (SECTION 01) Page 1

This section describes, in very general terms, the capabilities and features of the AUTOMON/BATCH product. The various features and benefits are presented, along with a comparison of AUTOMON/BATCH and CICS features. General operation of the AUTOMON/BATCH transaction, the opening selection menu and a summary of the available menu options are discussed.

THE FILE CONTROL FUNCTIONS OF AUTOMON/BATCH (SECTION 02) Page 7

This section discusses the use and operation of the file control functions available in AUTOMON/BATCH. Included are instructions for creating a new file definition; procedures for locating, displaying, and modifying any existing file definition; and in addition to single file displays, multiple file and generic displays are discussed.

GROUPING FILES, TRANSACTIONS AND PROGRAMS (SECTION 03) Page 15

This section deals with the two types of family records that can be defined in AUTOMON/BATCH. File families, along with all of the commands that may be used with them and transaction/program families, with their commands, are presented. The methods of displaying, adding, updating and deleting family records are described.

MIGRATION CONSIDERATIONS / RECOMMENDATIONS (SECTION 04) Page 21

This section deals with the process of evaluating the AUTOMON/BATCH product, then the methods for converting your existing FCT entries into the AUTOMON/BATCH control file. Limitations of the product, along with some of the optional features and methods of operation are discussed.

AUTOMATIC OPEN/CLOSE (SECTION 03) Page 23

This section deals with automatic closing and deallocation of files that have been inactive for a user specified amount of time. This feature allows you to free storage tied up by open files that are not being accessed by the end-users. If the file is requested by an application at a later time, CICS will open it as if it were the first request for that particular dataset.

THE BATCH INTERFACE TO AUTOMON/BATCH (SECTION 06) Page 27

The description, operation, commands, control and technical considerations of the Batch Interface to AUTOMON/BATCH are presented in this section. Methods for using this feature, with job control examples, are presented.

THE PROGRAMMER INTERFACE TO AUTOMON/BATCH (SECTION 07) Page 37

AUTOMON/BATCH may be called from a user program to perform any of the file activation and deactivation commands, as well as enabling and disabling transactions and programs. This section discusses the calling sequence and protocol, then presents fully coded example programs, both in COBOL and ASSEMBLER language.

FILE SERVICE FUNCTIONS OF AUTOMON/BATCH (SECTION 08)**Page 39**

This section discusses the use and operation of the file service functions available in AUTOMON/BATCH. Instructions are given for displaying, modifying, adding and deleting records from any file in CICS.

SEARCHING AND CHANGING MULTIPLE RECORDS - LOCATE (SECTION 09)**Page 57**

This section describes the powerful record locate command of the file services functions. Procedures for building an input query, directing the selected records, modifying and reformatting records and the execution of the locate query are described.

THE SECURITY CONTROL SYSTEM (SECTION 10)**Page 83**

The various methods of enforcing security over some of the operational features of AUTOMON/BATCH are discussed in this section. The on-line security control file maintenance is described, detailing the procedures for adding, updating and deleting security control records. In addition, the process of writing a security control user exit program is presented, along with COBOL and ASSEMBLER examples.

INSTALLATION OF THE AUTOMON/BATCH PRODUCT (SECTION 9)**Page 95**

The process of installing the package is presented in this section. Step by step procedures are set forth, itemizing all actions that must be taken to make the product operational in your environment.

CUSTOMIZATION (SECTION 10)**Page 111**

This section deals with the methods of tailoring AUTOMON/BATCH to fit the needs of your specific environment. The process of coding and installing the user option table, along with an explanation of each of the customization features is presented.

SPECIAL CONSIDERATIONS (SECTION 11)**Page 119**

This section presents a number of conditions and considerations that the CICS system programmer should be aware of when using AUTOMON/BATCH.

SPECIAL PURPOSE COMMANDS (SECTION 12)**Page 121**

A series of commands designed for use by the system programmer are presented in this section. These are commands used to control the environment as well as monitor the activity of the product.

MESSAGES (SECTION 13)**Page 123**

The various messages that can be generated from AUTOMON/BATCH are described here. Messages are divided into nine categories, depending on their point of origin and message number prefix. Each message is presented in its entirety along with a suggested action to be taken.

APPENDICES

Page 161

APPENDIX A.

Summary of all of the available **AUTOMON/BATCH** commands.

APPENDIX B.
data.

Description of the system requirements, storage usage, and other technical data.

APPENDIX C.

File definition record format.

APPENDIX D.
installation process.

Shows the installation documentation that may be printed during the installation process.

APPENDIX E.
AUTOMON/BATCH product.

Procedures to use if a problem is encountered with the **AUTOMON/BATCH** product.

INDEX

Page 373

INTRODUCTION

AUTOMON/BATCH is an on-line menu-driven system utility, operating under CICS, that provides the means to dynamically create and maintain all file definitions required by CICS by using the AUTOMON/BATCH interface to RDO. The interface greatly simplifies the process for the user by doing the following:

Fits the most-used data for RDO file processing onto one screen rather than the many screens required by RDO.

Provides help in window boxes which appear beside the entry in question by simply pressing PF1 (the CUA standard help key). Help is provided on every field displayed in the **AUTOMON/BATCH** system. RDO does not provide help text.

AUTOMON/BATCH does not require that the user know what RDO group the file resides in. All files within the CICS region are connected to the group they were installed from by **AUTOMON/BATCH** and all inquiries on a file name will automatically display it from the correct group. Group name is only required when doing a delete or define function.

When a file is viewed using **AUTOMON/BATCH**, the current service requests that are in force are displayed along with the service requests that the file was initially installed with. The status of the file is displayed (open, closed, enabled, disabled).

One of the more powerful functions of the interface is the ability to page backward and forward through the DFHCSD file in file name sequence without changing group names. **AUTOMON/BATCH** automatically takes care of this for you.

AUTOMON/BATCH, in addition to the single file viewing capabilities, provides two multi-file displays. With the touch of a PF key, all files can be viewed in file name sequence along with their associated group name. To perform RDO functions on a file, simply tab to the desired file, press a PF key and the file is displayed by the interface system ready for processing. The second multi-file display shows files with their dsname, group, current status and a command area to open, close, activate, allocate or deallocate the file or files.

In addition to the RDO interface, **AUTOMON/BATCH** will automatically close and deactivate files which have not been accessed in a specified amount of time (specified by the option table) thereby releasing storage and resources tied up by the unused file.

A powerful batch interface system is provided which allows batch processing to run concurrently with CICS activity.

A file utility is provided which allows manipulation of CICS files: files can be searched for hex/character strings, data may be replaced, files may be merged or created, records can be added, deleted, or modified, record lengths can be modified, and records may be printed. This very abbreviated description of the file maintenance system will be expanded later in the manual.

Files, program names and transaction IDs can be placed in a **AUTOMON/BATCH** "family" file and operated on by family name. For example, up to ninety files can be opened by simply opening the family name; or all transactions that access a given file could be disabled by disabling the family name containing the transaction IDs.

FEATURES AND BENEFITS

Some of the major features and benefits **AUTOMON/BATCH** provides:

- Complete file status and file definition information is instantly available for inquiry or authorized modification.
- Allows on-line files to be opened or closed from a batch partition or region.
- May automatically close and deactivate files which have not been used for a specified interval of time, thereby freeing the resources that were tied up by an inactive file.
- Allows transactions and/or programs associated with a file to be disabled when the file is closed and enabled when the file is opened.
- File records of any file in CICS may be displayed, altered, added, deleted or browsed with the file service functions.
- Data records are displayed in both hex and character format, allowing changes to be made in either mode.
- Data records may be searched for the presence of a hex or character string.
- An entire file may be searched using the LOCATE function. Queries may be constructed using Boolean logic. Selected records can be displayed, modified, deleted, added to another file, or printed.
- Data records may be reformatted on-line using the LOCATE function.
- Duplicate key data can be isolated using the LOCATE function.
- Non-numeric data fields can be found with the LOCATE function.
- The length of variable-length records can be changed with the File Service functions.
- Service requests for files may be varied to UPDATE or NOUPDATE through the Batch Interface program.
- An extensive help system has been included, which provides detailed help displays for every field displayed by the **AUTOMON/BATCH** system.

FCTD TRANSACTION OPERATION

To invoke the AUTOMON/BATCH transaction, enter the following transaction code:

FCTD

The FILE SERVICE MANAGER MENU will appear as follows:

```

              C I C S - F I L E S E R V
          RELEASE 3.1 PTF-LEVEL 07/01/02
              F I L E   S E R V I C E   M A N A G E R
              ----- SUB FUNCTIONS -----
(1)  HELP
(2)  FILE SERVICE UTILITY
(3)  EXIT
(4)  AUTO-BACKOUT ACTIVITY
(5)  BATCH INTERFACE ACTIVITY
(6)  USER OPTIONS DISPLAY
(7)  FAMILY PROCESSING
(8)  SECURITY MAINTENANCE
              ----- FILE FUNCTIONS -----
(AC)TIVATE A FILE      (CLO, ENA)
(C)LOSE A FILE         (CLO, DIS)
(O)PEN A FILE          (OPE, ENA)
(V)IEW/ALTER A FILE
(DEL)ETE A FILE
(DEF)INE A FILE
(DI)RECTORY OF FILES
(ST)ATUS OF FILES

```

ENTER PFKEY OR FUNCTION _____ FILE _____ GROUP _____

CICS-FILESERV RELEASE 3.1 SELECTION MENU
1=(H)ELP 3=(E)XIT

DESCRIPTION OF MENU OPTIONS

This is the primary AUTOMON/BATCH menu. All functions of this product can be accessed by passing through this menu. Sub-functions are accessed by pressing the corresponding PF key. File functions are accessed by entering the designated function command, file name (and, optionally, the group name).

HELP

This will display a help overview for this menu.

FILE SERVICE UTILITY

This is the menu used for the operation of the file service functions available in AUTOMON/BATCH. This may be used for displaying, modifying, adding records to, or deleting records from any file in CICS.

This is explained in *Section 6 -THE FILE SERVICE FUNCTIONS OF AUTOMON/BATCH* and *Section 7 - SEARCHING AND CHANGING MULTIPLE RECORDS - LOCATE*.

EXIT

This option will exit the FCTD transaction, and return to CICS.

AUTO-BACKOUT ACTIVITY This is used to display the AUTO-BACKOUT statistics screen. This screen is an accumulation of counters on various types of activity in AUTOMON/BATCH. The AUTO-BACKOUT program can be started or stopped from this screen.

This is explained in *Section 5 - Automatic Open/Close*.

BATCH INTERFACE FEATURE This is used to display the BATCH statistics screen. This screen is an accumulation of counters on commands issued from the batch program. This screen may also be used to activate or deactivate the batch program.

This is explained in *Section 6 - THE BATCH INTERFACE TO AUTOMON/BATCH*.

USER OPTIONS DISPLAY This is used to display the customization settings that are currently in effect for AUTOMON/BATCH. This display is for informational purposes only. Any customization changes will need to be enacted by recoding and assembling the customization table.

This is explained in *Section 12 - CUSTOMIZATION*.

FAMILY PROCESSING This function is used to display the dataset and transaction/program families, and optionally, the contents of any family. It may also be used to modify, create or delete a family.

This is explained in *Section 3 - GROUPING FILES, TRANSACTIONS AND PROGRAMS*.

SECURITY MAINTENANCE This is used to display, modify, create, or delete file security records.

This is explained in *Section 10 - THE SECURITY CONTROL SYSTEM*.

ACTIVATE A FILE Set a file to closed, enabled status.

BACKOUT A FILE Set a file to closed, enabled status. (This command is only used to preserve a function from AUTOMON/BATCH 1.5.)

CLOSE A FILE Set a file to closed, disabled status.

OPEN A FILE Set a file to open, enabled status.

ALLOCATE A FILE Set a file to closed status.

DEALLOCATE A FILE Set a file to closed status.

VIEW A FILE View a file using the AUTOMON/BATCH RDO interface.

DELETE A FILE Delete a file from a group using the AUTOMON/BATCH RDO interface.

DEFINE A FILE Define a file in a group using the AUTOMON/BATCH RDO interface.

DIRECTORY OF FILES List a directory of files in file name sequence with the associated group name of each file.

STATUS OF FILES List a multi-file display of files in file name sequence with the dsname, group name and current status.

AUTOMON/BATCH HELP

AUTOMON/BATCH provides an on-line HELP facility for quickly accessing operator instructions without the necessity of this Reference Guide.

Help is invoked by pressing PF1 or by changing the command code in the command line to 'H'.

There are three types of HELP that may be invoked (depending on the point in the AUTOMON/BATCH operation at which the HELP command is issued):

Field Level Help	This is specific help that applies to one field of the display. This type of help is accessed by tabbing the cursor to the field for which help is desired, then pressing PF1 (or ENTER if the H command is used.)
Screen Level Help	This is general help that applies to the entire screen from which the help was invoked. For this type of help, the position of the cursor is not important.
Nested Help	This help appears as a help menu while already in help mode. It is accessed by tabbing to the desired option, then pressing ENTER.

All types of help may be exited by pressing PF1, PF3 or the CLEAR key.

THE FILE CONTROL FUNCTIONS OF AUTOMON/BATCH

GENERAL OPERATION

To invoke the AUTOMON/BATCH menu, enter

FCTD

from a clear screen.

ENTERING COMMANDS FROM A CLEAR SCREEN

This same command sequence may be entered from a clear screen in the form

FCTD X DDDDDDDD

Where ...

FCTD	=	Transaction Code
X	=	Command Code
DDDDDDDD	=	Dataset Name

When entered from a clear screen, separate the transaction code, command and dataset name by a space.

CREATING A NEW FILE DEFINITION

From the AUTOMON/BATCH menu, enter DEFINE, the file name and group name.

Following is the FILE INTERFACE screen:

```

FILE          HWN$TXT          FUNCTION
GROUP        HELPGRP          CURRENTLY CLO,ENA - REA,UPD,ADD,DEL,BRO
DESCRIPT     HELP-WINDOWS TEXT FILE
VSAM PARAMETERS
DSNAME       CICS330.HELP30.TEXT.FILE
PASSWORD
LSRPOOLID   1                 BUFFERS                      OPERATIONS
DSNSHARING  ALLREQS          DATABUFF  00004             ADD                          YES
STRINGS     003              INDEXBUFF  00003             BROWSE                        YES
NSRGROUP    DATATABLE PARAMETERS  DELETE                       YES
REMOTE ATTRIBUTES  TABLE      NO              READ                          YES
REMOTESYS   MAXNUMRECS       UPDATE                       YES
REMOTENAM   DATA FORMAT     AUTO JOURNALLING
RECORDSIZ   RECORDFORM V     JOURNAL                      NO
KEYLENGTH   RECOVERY PARAMETERS  JNLREAD                      NONE
INITIAL STATUS  RECOVERY NONE          JNLSYNCREADJ                NO
STATUS      ENABLED          FWDRECOVLO NO              JNLUPDATE                    NO
OPENTIME    FIRSTREF        JNLADD                      NONE
DISP        SHARE          JNLSYNCWRITE                YES

CICS-FILESERV RELEASE 3.0      FILE INTERFACE
1=HELP      2=VIEW      3=EXIT      4=OPEN      5=ACTIV      6=CLOSE      7=PREV      8=NE
9=ALLO     10=DEAL     11=DELE     12=CEMT     13=ALTER     14=DEFIN     15=INST     16=DI
17=STAT

```

DATA ENTRY FIELDS

Each field description on the AUTOMON/BATCH RDO interface screen is identical to the corresponding field on the RDO screens, except a very long description may have been truncated in order to align the display for ease of viewer use. Each field that may be entered in the definition corresponds to the RDO input. Each field, unlike RDO, has a help display window associated with it. Simply press PF1 and the help display will appear in close proximity to the field giving the acceptable values for input.

VALIDATION OF FIELDS

This concludes the entry of fields for a new File Definition. Once all fields have been keyed, press PF14 or enter DEFINE in the function field and press enter.

Fields will be verified for valid codes or values. Any fields found invalid will trigger a redisplay of the screen with the corresponding message from RDO displayed at the bottom of the screen. The message may be from one to three lines depending on the response from RDO

If no errors are found, the following message will appear on line 21.

F5003. CEDA COMPLETED SUCCESSFULLY

DUPLICATING FILE DEFINITIONS

If possible, view an existing file definition, then change any fields that are unique to the new file such as file name, group name, description and dsname and press PF14 or enter DEFINE in the function field and press ENTER. If an existing definition that closely resembles the new definition does not exist, you may have to enter each new field for the new definition. However, this is seldom the case.

DISPLAYING AND MODIFYING A FILE DEFINITION RECORD

Use one of the following methods to display a single File Definition:

Method 1: Using the menus.

From the FILE SERVICE MANAGER menu enter VIEW followed by the file name and press ENTER.

The AUTOMON/BATCH RDO interface screen will display the file. Make the desired modifications and press PF13 or enter ALTER in the function field and press enter. The file definition will be updated in the RDO group.

Method 2: Bypassing the Menus.

From a clear screen enter the following:

```
FCTD V xxxxxxxx
```

Where 'xxxxxxx' is the DSNAME.

ENTERING COMMANDS IN THE AUTOMON/BATCH RDO INTERFACE.

When a File Definition is displayed on the screen, you may issue any of the commands that are listed on lines 22, 23 and 24.

The available commands at this point are:

PF1(H)	Display a HELP screen. Position the cursor on any field that help is needed and press PF1. To invoke the help overview for the display, position the cursor outside of any input field and press PF1.
PF2(VIEW)	View a file.
PF3(EXIT)	Return to the FILE SERVICE MANAGER menu.
PF4(OPEN)	Open this file to CICS. If the file is not yet activated in the FCT it will be activated automatically and then opened. If the file is already open, the command will be ignored.
PF5(ACTIVATE)	Activate this record in the FCT.
PF6(CLOSE)	Close this file to CICS. The file will not be deactivated. The file is marked as ineligible for automatic open.
PF7(PREVIOUS)	Browse backward to the previous File Definition and display it.
PF8(NEXT)	Browse forward to the next File Definition and display it.
PF9(ALLOCATE)	Allocate the file to the CICS region.
PF10(DEALLOCATE)	Deallocate this file from the CICS region.
PF11(DELETE)	Delete this file from the RDO group.

PF12(CEMT)	Link to CEMT and inquire on the file.
PF13(ALTER)	Alter the file with the changes just keyed on the screen.
PF14(DEFINE)	Define a new file definition to the RDO group designated in group field.
PF15(INSTALL)	Install the group designated in group field.
PF16(DIRECTORY)	List a multi-file directory in file name sequence.
PF17(STATUS)	List a multi-file directory in file name sequence.

Two other options are available at this point:

- **Multi-file Directory Display**

To request a directory display, enter the beginning file name in the file name field and press PF16.

- **Multi-file Status Display**

To request a status display, enter the beginning file name in the file name field and press PF17.

MULTIPLE FILE DIRECTORY DISPLAY

```
FILE      GROUP      FILE      GROUP      FILE      GROUP      FILE      GROUP
-----
HWN$TXT  -HELPGRP
JUGLFIL  -JUJLGRP
STS$FIL  -FCTDGRP
WNDOFIL  -WNOGRP
ZZZ$FIL  -FCTDGRP

FC530005. END OF FILES ENCOUNTERED
CICS-FILESERV RELEASE 3.0      FILE DIRECTORY
1=HELP   2=VIEW   3=EXIT   7=PREV   8=NEXT
```

If this were a generic file display, only those files that matched the first nn characters of the entered dataset (up to the asterisk) would be displayed.

At this point, you may invoke page forward, backward, invoke HELP, exit or place the cursor on a particular file and view the file definition. All of the above functions are invoked by use of the PF key listed on row 24 of the display.

Both of these options are described in the following topic, entitled *ENTERING COMMANDS AT THE MULTI-FILE STATUS DISPLAY*.

ENTERING COMMANDS AT THE DIRECTORY DISPLAY

The available commands at this point are:

- | | |
|---------------|---|
| PF1(HELP) | Display a HELP screen, which describes the available commands and the procedural options at this point. |
| PF2(VIEW) | Position the cursor on the desired file and press PF2. |
| PF3(EXIT) | Return to the FILE SERVICE MANAGER menu. |
| PF7(PREVIOUS) | Browse backward to display the previous screen of files. |
| PF8(NEXT) | Browse forward to display the next screen of files. |

MULTIPLE FILE STATUS DISPLAY

FILE	DSNAME	GROUP	STATUS
HWN\$TXT	CICS330.HELP30.TEXT.FILE	HELPGRP	CLO, ENA
JUGLFIL	CICS321.JUGL32.CONTROL.FILE	JUGLGRP	CLO, ENA
STS\$FIL	CICS330.FSRV30.FILE	FCTDGRP	OPE, ENA
WNOFIL	CICS321.WNO32.CONTROL.FILE	WNOGRP	CLO, ENA
ZZZ\$FIL	RADS330.FSRV30.CONTROL.FILE	FCTDGRP	CLO, ENA

CICS-FILESERV RELEASE 3.0 MULTI-FILE STATUS
1=HELP 2=VIEW 3=EXIT 7=PREV 8=NEXT

If this were a generic file display, only those files that matched the first nn characters of the entered dataset (up to the asterisk) would be displayed.

At this point, you may invoke one of the six listed PF keys, or their corresponding Command Code, by changing the 'I' in the command line to the desired function; or you may invoke one of the commands listed at the bottom of the screen, by typing the command code in the corresponding command field beside the file that you wish to effect.

Both of these options are described in the following topic, entitled *ENTERING COMMANDS AT THE MULTIPLE FILE DISPLAY*.

ENTERING COMMANDS AT THE STATUS DISPLAY

The available commands at this point are:

PF1(HELP)	Display a HELP screen, which describes the available commands and the procedural options at this point.
PF2(VIEW)	Position the cursor on the desired file and press PF2.
PF3(EXIT)	Return to the FILE SERVICE MANAGER menu.
PF7(PREVIOUS)	Browse backward to display the previous screen of files.
PF8(NEXT)	Browse forward to display the next screen of files.

You may also enter one of the listed commands to effect one of the file definitions on the display by tabbing to the command field at the end of the desired dataset, then keying the command code and pressing ENTER.

'O', 'C', 'ACT', 'ALL', and 'DEA' Commands
(OPEN, CLOSE, ACTIVATE, ALLOCATE AND DEALLOCATE)

For the above commands the action taken is as follows:

- 1) Each command is processed in the sequence that it appears on the screen.
- 2) If all commands are successfully performed, the screen will display the new file status of each file displayed in the STATUS field.
- 3) If the command is 'O' (open) and an open error occurs for a dataset, the dataset in error will be highlighted and the message 'ERRxx' will display in the STATUS FIELD. The 'xx' in the message is the open error code. For VSAM files, this code is documented in "VSAM MESSAGES". For non-VSAM files, this code is the return code from the dynamic open/close program (DFHOCP).
- 4) If an error condition other than an open error occurs while attempting to process a command, the dataset in error will be highlighted, a message describing the error will display on line 24, and any remaining commands on the screen will not be processed.

GROUPING FILES, TRANSACTIONS AND PROGRAMS

It is often the case that several datasets are related in some way, such as all participating in the same system, and the names of the datasets do not start with the the same common characters. This makes it difficult to issue Multiple File Commands to Open, Close, Backout, Activate, Allocate or Deallocate these files.

A feature is provided with AUTOMON/BATCH to define a FAMILY of datasets which may or may not be generically equivalent in their naming conventions. After defining a family, the entire family of datasets can participate in one file command.

Two types of families are provided: a FILE family, which is used to relate several datasets together, and a TRANSACTION/PROGRAM family, which is used to relate several transactions and/or programs together. Both types are explained in detail in the remainder of this section.

THE FAMILY DIRECTORY DISPLAY

The FAMILY DIRECTORY display is used to display up to 324 family names (both file and transaction/program families). This screen may be displayed by pressing PF7 from the FILE SERVICE MANAGER menu.

The FAMILY DIRECTORY screen will appear as follows:

```
FAMILY    FAMILY    FAMILY    FAMILY    FAMILY    FAMILY    FAMILY    FAMI
-----
ACCT
DOCUTEST
RADGRP
SMDFG1
SMDFG2
SMDFG5
SMDGROUP
SMDGRP
GRPTPRAD
RADGRPT
RADGRPTP
SMDTG1
SMDTG2
SMDTRAN

ENTER FUNCTION, FAMILY NAME, TYPE (F/T/P)
CICS-FILESERV RELEASE 3.0    FAMILY DIRECTORY
1=HELP    V=VIEW    3=EXIT    N=NEW GROUP    D=DELETE
```

CREATING A NEW FAMILY DEFINITION

To create a new family definition, from the FAMILY DIRECTORY SCREEN key an 'N' in the COMMAND field, the name of the family that you wish to create, and the type of family.

```
FCTD N xxxxxxxx GROUP
```

Where 'N' is the command code for creating a new definition and 'xxxxxxx' is the name of the family that you wish to create.

Upon pressing ENTER, the FILE FAMILY DISPLAY or the TRAN/PROG FAMILY DISPLAY screen will appear (depending on the type entry). At this point, the screen is ready to accept resource names that make up the family.

Enter dataset IDs (or transactions or programs) in as many fields as desired (up to 90). It does not matter if you skip a field, leaving it blank. All datasets defined on the screen will always be processed; blank fields will be ignored.

On file families, you cannot key anything in the return code (RC) fields. This is a 'display-only' field. When performing a function on the family, if an error occurs the error code will be displayed in the field corresponding to the file that caused the error.

When all desired fields have been keyed, press ENTER. This will cause the Family Record to be updated.

If any other message appears, consult *Section 15 - MESSAGES*, of this manual for an explanation and appropriate action.

DISPLAYING AND MODIFYING A FAMILY DEFINITION

To inquire on an already existing Family Definition, from the FAMILY DIRECTORY screen key the 'V' (View) command code in the COMMAND field and the family name in the NAME field.

```
FCTD I xxxxxxxx GROUP
```

Where 'xxxxxxx' is the name of the family.

Upon pressing ENTER, the Family Definition Record will be retrieved and displayed using the FILE FAMILY display screen.

To make changes to any of the fields in the family, tab to the desired fields, key over the data in the fields, then press ENTER.

To delete a field from the family (or to erase any data), simply erase the data with the SPACE bar.

DELETING A FAMILY DEFINITION

To delete a FAMILY Definition, from the FAMILY DIRECTORY screen, key the 'D' command code in the COMMAND field and the family name in the NAME field.

Upon pressing ENTER, the following message will appear on line 24:

```
PRESS ENTER TO DELETE FAMILY (XXXXXXXXX). ANY OTHER KEY TO IGNORE.
```

To delete this Family Definition Record, press ENTER. If deletion is not desired, press any other key. Note that deleting the family definition will only delete the definition. This will not delete the file definitions associated with that RDO group. If you wish to delete the file associated with an RDO Group, you will have to delete them through

the usual methods. (See *Section 2 - THE FILE CONTROL FUNCTIONS OF AUTOMON/BATCH* under the topics: *DESCRIPTION OF AUTOMON/BATCH COMMANDS.*)

FILE FAMILIES

THE FILE FAMILY DEFINITION SCREEN

The FILE FAMILY DEFINITION screen is used to display, create and modify FILE FAMILY definitions. To display this screen you may inquire on a FAMILY definition from the FAMILY DIRECTORY screen.

FCTD I xxxxxxxx GROUP

Where 'xxxxxxx' is the family name. 'GROUP' may also be abbreviated as 'GRP'.

Upon pressing ENTER, the FILE FAMILY DEFINITION screen will appear as follows:

family;

```

                                     FAMILY ==> RADGRP
DATASET  RC  DATASET  RC  DATASET  RC  DATASET  RC  DAASET  RC  DATASET  F
-----  -  -
HWN$TXT
HWN$ROG

(PPOSITION CURSOR TO PERFORM FUNCTION ON SINGLE DATASET)          ==> _____
CICS-FILESERV RELEASE 3.0      FILE FAMILY DISPLAY
1=(H)ELP  2=(V)IEW  3=EXIT  4=(O)PEN  5=(A)CTV  6=(C)LOS  7=(P)REV  8=(F)NE
```

The screen is divided into six columns, each column containing 15 sets of two fields. The two fields are:

DATASET This is the DDNAME of the dataset to be included in this FAMILY. There must be a file defined to CICS with this dataset name.

RC This field is used to display the return code for each dataset whenever an OPEN command is issued for the family.

ENTERING COMMANDS AT THE FILE FAMILY DEFINITION SCREEN

Any of the commands listed at the bottom of the Family Definition screen may be issued at this point by either entering the alpha command code or by pressing the designated PF key. The available commands are:

PF1(HELP) Display a Help window for Family Definition screen.

PF2(VIEW)	View the file that the cursor is in when ENTER (or PF2) is pressed.
PF3(EXIT)	Return to the FAMILY DIRECTORY screen.
PF4(OPEN)	Open all datasets in the family.
PF5(ACTIVATE)	Activate all datasets in the family.
PF6(CLOSE)	Close all datasets in the family.
PF7(PREVIOUS)	Page to the previous screen.
PF8(NEXT)	Page to the next screen.

When viewing a family definition screen, you can perform any valid command for an individual file in the family in either of the following two ways:

- 1) Enter the command to be performed in the command field, then tab the cursor to the desired file name and press ENTER.
- 2) Tab to the desired file, then press the PF key corresponding to the command to be performed.

For example if you tab to the desired entry and press ENTER while the OPEN command is present, the cursored file will be opened.

ENTERING COMMANDS FOR FILE FAMILY DEFINITIONS

The following commands may be issued for a File Family in order to perform the requested command for every dataset defined in the family. These commands may be issued from a terminal or from the Batch Interface.

ACT - Activate (Set file closed, enabled)
B - Backout (Set file closed, enabled)
C - Close (Set file closed, disabled)
O - Open (Set file open, enabled)
ALL - Allocate the dataset (MVS only)
DEA - Deallocate the dataset (MVS only)

When any of these commands are issued, the following actions are performed:

- 1) The dataset names in the Family Definition Records are scanned left-to-right, top-to-bottom.
- 2) The requested command is performed for that file.

If a file open or close error occurs for any field in the family, the operation continues to the end of the family, then the FILE FAMILY DEFINITION screen is displayed with the following message on line 24 of the screen:

F0131. OPEN/CLOSE ERROR HAS OCCURRED

The file(s) that had the error will display the error in the RC field for that file. The return code is the "URESPI" code set by CICS/ESA.

TRANSACTION PROGRAM FAMILIES

You may also define families of transactions, or families of program names, or a mixed family of transactions and programs. This type of family is used to define a set of resources to be enabled or disabled using the ENABLE or DISABLE commands.

The Transaction/Program Family Definition Screen can be used to define from 1 to 90 transactions and/or programs that belong to one family. A unique 8-position name is given to the family, and the result is a Transaction/Program Family Definition Record that resides in the AUTOMON/BATCH control file.

Only two commands can be performed on a transaction/program family record. They are:

ENABLE (E) Enable all transactions and/or programs in the family.

DISABLE(DIS) Disable all transactions and/or programs in the family.

THE TRANSACTION/PROGRAM FAMILY DEFINITION SCREEN

The TRANSACTION/PROGRAM FAMILY DEFINITION screen is used to display, create and modify family definitions. To display this screen you may inquire on a family definition from the FAMILY DIRECTORY screen. Upon pressing ENTER, the TRANSACTION/PROGRAM FAMILY DEFINITION screen will appear as follows:

```

                                FAMILY ==> RADGRPT

TRN/PRG ?   TRN/PRG ?   TRN/PRG ?   TRN/PRG ?   TRN/PRG ?   TRN/PRG
----- -   ----- -   ----- -   ----- -   ----- -   -----
HELPPWDO P
HWIN       T
HELPPAUXL P
HAUX      T
HELPEXIT P
HWIX      T

ENTER TRANID OR PROGRAM FOLLOWED BY (T)RAN OR (P)ROG UNDER ?
CICS-FILESERV RELEASE 3.0      TRAN/PROG FAMILY DISPLAY
1= (H) ELP      3=EXIT      4= (E) NAB      6= (D) ISAB      7= (P) REV      8= (F)

```

The screen is divided into six columns, each column containing 15 sets of two fields. The two fields are:

? This field is used to distinguish between a transaction and a program. A 'T' designates a transaction, 'P' designates a program.

TRAN/PROG This is the transaction ID or program name to be included in this family. If the ? field contains a 'T', this must be a 1 to 4 character transaction ID. If the ? field contains a 'P', this must be the 1 to 8 character program ID.

ENTERING COMMANDS AT A TRANSACTION/PROGRAM FAMILY SCREEN

Any of the commands listed at the bottom of the Tran/Prog Family Display screen may be issued at this point by either entering the alpha command code or by pressing the designated PF key.

The available commands are:

PF1(HELP)	Display a Help window for Family Definition screen.
PF3(EXIT)	Return to the FAMILY DIRECTORY screen.
PF4(ENAB)	Enable the transaction or program.
PF6(DISABLE)	Disable the transaction or program.
PF7(PREVIOUS)	Page to the previous screen.
PF8(NEXT))	Page to the next screen.

When viewing a TRAN/PROG Family display screen, you can perform any valid command for an individual transaction or program in the family in either of the following two ways:

- 1) Enter the command to be performed in the command field, then tab the cursor to the desired file name and press ENTER.
- 2) Tab to the desired file, then press the PF key corresponding to the command to be performed.

For example if you tab to the desired entry and press ENTER while the ENABLE command is present, the censored transaction or program will be enabled.

Any of the commands listed at the bottom of the TRAN/PROG Family Display Screen may be issued at this point by either entering the alpha command code or by pressing the designated PF key.

ENABLING AND DISABLING A TRANSACTION/PROGRAM FAMILY

The following commands may be issued for a Transaction/Program Family Definition in order to perform the requested command for every entry defined in the family. The commands can be issued from a terminal or from the Batch Interface.

E	Enable all entries in the family
DIS	Disable all entries in the family

When either of these commands are issued, the following actions are performed:

- 1) The entry names on the TRAN/PROG Family Display are scanned left-to-right, top-to-bottom.
- 2) The requested command is performed for each entry.

If an error occurs for any entry in the family, such as a transaction or program not being defined to CICS, the operation continues to the end of the family. The transactions or programs for which the error occurred will be highlighted, and an error message will display at the bottom of the screen.

MIGRATION CONSIDERATIONS AND RECOMMENDATIONS

If you have been using AUTOMON/BATCH 2.1 and are migrating to CICS/ESA 3.2.1 or greater, you may use the FCTDUTIL program to convert existing file definitions in the AUTOMON/BATCH control file to the DFHCSD file for CICS/ESA.

COVERING FCT ENTRIES TO THE DFHCSD FILE

For users that have FCT definitions currently defined in a AUTOMON/BATCH control file, a migration feature is provided with AUTOMON/BATCH that eliminates the work of re-coding FCT entries in order to define them to CICS/ESA. (See THE CONVERSION UTILITY PROGRAM in *Section 11 - INSTALLATION OF THE AUTOMON/BATCH PRODUCT*).

AUTOMATIC OPEN CLOSE

CONTROLLING AUTOMATIC BACKOUT - THE FCTT TRANSACTION

The FCTT transaction can be used to control the automatic-backout feature and review statistics for automatic-backout activity.

There are four commands that can be entered with the FCTT transaction. They are:

1) `FCTT,START`

This command will activate the automatic-backout feature of AUTOMON/BATCH which 'wakes up' at periodic intervals to see if any files need to be backed out. The START command is not necessary if AUTOBINT and AUTOBOUT were specified in the FCTDTBL macro options.

2) `FCTT,START,INT=MMSS`

This command can be used to start the automatic-backout feature and temporarily change the 'wake-up' interval coded in the user option table. The wake-up interval is coded in the user option table by means of the AUTOBINT keyword, which specifies the interval of time that FCTD1000 will be inactive before checking for anything to process.

To temporarily (until changed back or until CICS comes down) change the time interval, specify a new number of minutes and seconds following the INT= keyword in the command. Enter the number of minutes as two digits (00 to 60) and seconds as two digits (00 to 59).

Note that if the automatic-backout feature is already active, the START command used in this option has no effect; however, the wake-up time interval will still be changed.

3) `FCTT,STOP`

This command is used to deactivate the automatic-backout feature altogether. When the STOP command is issued, no more automatic backouts will occur until the START command is issued again.

The automatic-backout feature may also be started and stopped from the AUTO-BACKOUT screen, which is described in the following topic.

THE AUTO BACKOUT SCREEN

This screen is an accumulation of counters on various types of activity in AUTOMON/BATCH. This screen may also be used to activate and deactivate the automatic-backout feature through the use of the PF10 and PF11 keys, respectively.

To display the AUTO-BACKOUT screen, press PF4 from the FILE SERVICE MANAGER menu.

The screen will appear generally as follows:

C I C S - F I L E S E R V		DATE	: 10/19/90
A U T O C L O S E / B A C K O U T		TIME	: 12:11
		TERMINID	: V001

CURRENT STATUS	INACTIVE	INTERVAL	0000 (HHMM)
TOTAL FILES	000000	AUTO BACKOUTS	000000
TOTAL OPEN	000000	AUTO CLOSES	000000
TOTAL CLOSED	000000	AUTO OPENS	000000
TOTAL VSAM	000000	PENDING BACKOUTS	000000
TOTAL BDAM	000000	PENDING CLOSES	000000
TOTAL REMOTE	000000		

F1=HELP	F3=RETURN	F10=START	F11=STOP

(C) SOFTOUCH SYSTEMS, INC. 1986			

The following fields are displayed on the FCTT statistics screen:

DATE This is the current date in MM/DD/YY format.

TIME This is the current time in HH:MM format.

TERMINID This is the terminal ID of this terminal.

AUTOMATIC-BACKOUT FEATURE FIELDS:

STATUS	The current status of the automatic-backout feature. One of the following will be displayed: ACTIVE The automatic-backout feature has been started with an FCTT,START command or at PLTPI time. INACTIVE The automatic-backout feature is currently inactive. It has been stopped with an FCTT,STOP command or was never started.
INTERVAL	This is the wake-up interval for the automatic-backout feature. It is displayed as MMSS, where MM is the total number of minutes and SS is the number of seconds.
VSAM FILES	This is the total number of VSAM files currently in the CICS region along with the number open and the number of files closed.
BDAM FILES	This is the total number of BDAM files currently in the CICS region along with the number open and the of files number closed.
REMOTE FILES	This is the total number of REMOTE files currently in the CICS region along with the number open and the number of files closed.

ENTERING COMMANDS AT THE AUTO-BACKOUT DISPLAY SCREEN

The PF key list at the bottom of the screen displays the available options that can be performed at this point.

PF1=HELP	Pressing PF1 will display HELP for the display.
PF3=EXIT	Pressing PF3 will exit the automatic-backout display screen and return to the FILE SERVICE MANAGER menu.
PF10=START	Pressing PF10 will activate the automatic-backout feature. This has the same effect as issuing the FCTT,START command, from a clear screen.
PF11=STOP	Pressing PF11 will deactivate the automatic-backout feature. This has the same effect as issuing the FCTT,STOP command, from a clear screen.

THE BATCH INTERFACE TO AUTOMON/BATCH

A feature of AUTOMON/BATCH allows AUTOMON/BATCH file commands for CICS to be issued from another region. This allows on-line files running in CICS to be closed, opened, activated, deactivated, allocated or deallocated from within a batch jobstream, eliminating the requirement of operator intervention to close a file in order to run a batch update job. In addition, you may disable or enable one or more transactions and/or programs that are associated with a file.

To use the Batch Interface, FCTDBCHT can be executed at any point in the batch jobstream, as many times as needed. Control statements input to the command processor program are coded in exactly the same format as on-line AUTOMON/BATCH commands, that is, TRANCODE, COMMAND CODE, DATASET ID. All command options supported by the on-line FCTD transaction are supported in batch mode. This includes Generic commands, ALL commands and FAMILY commands.

Multiple CICS systems are supported by the FCTDBCHT program. The CICS system to which a command is directed is designated by means of a special operand, CICS=XXXXXXXX on the control statement. XXXXXXXX is the CICS Application ID (APPLID) coded in the AUTOMON/BATCH User Option Table for that CICS system.

Any number of commands may be input in a single execution of the command processor program. Processing consists of the following three steps for each command:

- 1) The command is displayed on the system console.
- 2) The command is routed to the requested CICS system.
- 3) Upon completion, the command-result message is displayed on the system console.

COMMANDS SUPPORTED IN BATCH MODE

The following commands may be issued from a batch partition by using the FCTDBCHT program:

ACTIVATE (ACT)	Activate one or more files.
BACKOUT (B)	Backout one or more files.
CLOSE (C)	Close one or more files.
OPEN (O)	Open one or more files.
ALLOCATE (ALL)	Allocate one or more files (MVS only).
DEALLOCATE (DEA)	Deallocate one or more files (MVS only).
VARY (V)	Vary the service request value of a file.
ENABLE(E)	Enable a transaction, program or family of transactions and/or programs.
DISABLE(DIS)	Disable a transaction, program or family of transactions and/or programs.
Z STOPBCH (Z)	Deactivate the on-line Batch Interface.
START (S)	Start a transaction and optionally pass it a data stream.

ENABLING THE BATCH INTERFACE FEATURE

In order to enable the Batch Interface feature of AUTOMON/BATCH, the following steps must be performed:

- 1) The AUTOMON/BATCH Control File (FCT\$FIL) must be defined using SHAREOPTIONS (4).
- 2) The User Option Table (FCTDTBL) must be installed with the BCHINTF operand coded YES and the APPLID coded with the identifying name for this CICS system.
- 3) If the preceding two steps are performed while CICS is up with AUTOMON/BATCH already installed, you may either bring CICS down and back up, or issue a 'Z STRTBCH' command to activate the Batch Interface (see *Section 14 - SPECIAL PURPOSE COMMANDS*).

RUNNING THE COMMAND PROCESSOR PROGRAMS

Following is the MVS JCL required to run the FCTDBCHT program:

```
//BCHJOB          JOB1,'ACCOUNT-ID',MSGCLASS=X
//STEP1          EXEC      PGM=FCTDBCHT
//FCT$FIL        DD DSN='CICS.ABATCH.CONTROL.FILE',
//              DISP=SHR
//RDRFIL         DD *
CICS=XXXXXXXXX,INACTIVE=[BYPASS | CANCEL | IGNORE]
(Control Statement(s) Input)
/*
//
```

ESTABLISHING A DEFAULT CICS APPLICATION ID

Each CICS region is uniquely identified to AUTOMON/BATCH by the FCTDTBL parm APPLID=xxxxxxx. The CICS=xxxxxxx statement must correspond to the FCTDTBL APPLID in order for FCTDBCHT to process files.

The following statement must be the first statement encountered by FCTDBCHT:

```
CICS=xxxxxxx,INACTIVE=[BYPASS | CANCEL | IGNORE]
```

where xxxxxxxx is the job name for the region. The INACTIVE parameter is used to specify the action to be taken by FCTDBCHT if the CICS is not active or if the Batch Interface Control Record is not found. The operands are described as follows:

BYPASS	This will cause FCTDBCHT to bypass all commands for the inactive CICS. Processing will continue, however all commands for the inactive CICS will be ignored.
CANCEL	This will immediately terminate the FCTDBCHT process if the CICS is not found.

IGNORE (VSE users only) This will cause FCTDBCHT to ignore that CICS is inactive. The following commands will be performed for the inactive CICS. This would be used for sites running VM with multiple VSE guests that are sharing datasets. Note that due to the differences between MVS and VSE, this is action is not valid for MVS.

INACTIVE is a new keyword. In past releases, the action taken for MVS was to CANCEL, and for VSE to BYPASS. These actions have not changed and are the defaults if the INACTIVE keyword is omitted.

The default Application ID will apply to all control statements until another Default Application ID control statement is encountered, or the end of control statements is reached.

CONTROL STATEMENT FORMAT

Control statements, whether input into the data stream or from the system console, are entered in the following format:

TRANSACTION CODE, COMMAND CODE, DATASET ID

Where ...

Transaction code = FCTD
(followed by one or more spaces)

Command code = 'ACT', 'B', 'C', 'E', 'O', 'DIS', 'S', 'V', 'ALL', 'DEA', 'Z'
(followed by one or more spaces)

Dataset ID = This may be any of the following:

- 1) A single file name
- 2) A generic name (XX*)
- 3) The word 'ALL'
- 4) A File Family Definition name followed by the word 'GROUP', 'GRP' or 'GRPFL'
- 5) A Transaction/Program family name followed by the word 'GRPTP'.
- 6) The word 'STOPBCH' ('Z' command).

NOTE: In order to allow past users of AUTOMON/BATCH to utilize the old command formats of previous releases of AUTOMON/BATCH, the old command formats have been retained along with the new command formats. The user is urged to use the newer formats as they are more readily recognized as to the function they perform. The following list shows the old command along with the new command:

COMMAND	OLD	NEW
ACTIVATE		A (ACT)IVATE
BACKOUT		B (B)ACKOUT
CLOSE		C (C)LOSE
OPEN		O (OPEN)
ALLOCATE		A (ALL)OCATE
DEALLOCATE		Y (DEA)LLOCATE
VARY		V (V)ARY
ENABLE		E (E)NABLE
QUIESCE		Q (DIS)ABLE

VARYING THE SERVICE REQUEST VALUES OF A DATASET

The VARY command may be issued from the batch interface to vary the service request values of a file to UPDATE or NOUPDATE as required. This will allow the batch program to perform its processing on the file and still allow other partitions or regions to access the file (for read only).

This command may be issued from the batch interface in the form:

```
FCTD V xxxxxxxx [UPD | NOUPD]
```

where xxxxxxxx is the file name to be varied.

The NOUPD operand specifies that the services request values are to be set to BROWSE, READ ONLY. This is the default if omitted.

UPD specifies that the service request values are to be reset to the original state when the file was opened.

STARTING A USER TRANSACTION FROM THE BATCH INTERFACE

The Batch Interface can be used to initiate user transactions as well as issuing AUTOMON/BATCH commands. A special command is provided that is only valid when used in the Batch Interface. This is the "S" command, meaning START a transaction. The format of the "S" command in a Batch Interface control statement is as follows:

```
FCTD S TTTT[DATA='xxxxxxxxx'][(TERM=yyyy)]
```

where ...

S	=	the command to start a user transaction.
TTTT	=	the CICS transaction code to be initiated.
DATA='xxxxxxxxx'	=	an optional keyword which allows data to be passed to the invoked transaction via normal Interval Control procedures. The data to be passed must be enclosed in single quotes.
TERM=yyyy	=	an optional keyword that specifies a terminal to which the user transaction is to be attached. yyyy is the CICS terminal ID.

You may optionally code the CICS= operand on this statement if desired.

CONSIDERATIONS FOR STARTING USER TRANSACTIONS

1) AUTOMON/BATCH will initiate the requested transaction in the requested CICS region using a Command-Level START command). If the TERM operand is omitted, the resultant transaction is not attached to a terminal, therefore no terminal activity can be performed.

2) If the invoked program needs to receive data, it must be done using standard Interval Control Procedures. AUTOMON/BATCH will issue the following command to write the data between the quotes of the DATA= operand on the control statement to the Interval Control queue file:

```
EXEC CICS START TRANSID(XXXX)
      FROM(data between quotes)
      LENGTH(length of data).
```

The invoked program must issue an EXEC CICS RECEIVE to retrieve the data. The data is in standard variable-length format as required by CICS Interval Control. That is, the actual data will be preceded by a four-byte record descriptor word (RDW) containing the hexadecimal length (including the RDW length) in the first two bytes and hex zeros in the last two bytes.

3) The invoked transaction can perform any Macro-Level or Command-Level CICS functions, such as enabling or disabling transactions, altering the FCT to read-only, etc.

4) If the TERM operand is specified, the invoked transaction will be attached to the requested terminal and any terminal commands are valid. The invoked transaction will not receive an input terminal I/O area and therefore should not do an initial terminal RECEIVE command or attempt to address the TIOA. You can, however, output a terminal screen and initiate a conversation on the terminal if desired.

5) Since Interval Control is used to initiate the transaction, there are two considerations when the TERM operand is specified:

a) The transaction will not be invoked if the terminal is busy.

b) The requested transaction must run locally in the designated CICS region. DFHIC will not initiate a remote transaction in an MRO environment.

CONTROL STATEMENT EXAMPLES

The following examples illustrate several possible types of control statement input to the command processor programs:

CICS=TESTCICS,INACTIVE=BYPASS

Set TESTCICS as the default CICS, with the bypass action.

CICS=PRODCICS

Set PRODCICS as the default CICS, with the default action for your system.

FCTD O A* CICS=CICSB

Open all files starting with A in CICSB.

FCTD C PROSPCT CICS=PCICS

Close file PROSPCT in PCICS.

FCTD B ALL

Backout all files in TESTCICS.

FCTD A ACCTNG GROUP

Activate files in the ACCTNG family.

FCTD C ALL

Close all files in PRODCICS.

FCTD C MASTER CICS=TCICS

Close MASTER file in TCICS.

FCTD VARY MASTER CICS=TCICS

Vary the service request value of MASTER file in TCICS.

FCTD DIS MAST TRAN

Disable transaction MAST in PRODCICS.

FCTD B MASTER

Backout MASTER file in PRODCICS.

FCTD ENAB MASTUPD PROG

Enable program MASTUPD in PRODCICS.

FCTD S DISA

Start transaction DISA in PRODCICS.

FCTD S DISA CICS=TCICS

Start transaction DISA in TCICS.

FCTD S FLCT DATA='MASTER,PROSPCT'

Start transaction FLCT in PRODCICS passing two file names.

FCTD S FLCT,TERM=L77B

Start transaction FLCT, connect to terminal L77B in local CICS.

[Note]: If commands to more than one CICS system are input in one jobstream, all referenced CICS systems must be sharing the same AUTOMON/BATCH control file (FCT\$FIL). Otherwise, separate job steps must be run with the DD statement for the correct VSAM file.

CONTROLLING THE BATCH INTERFACE

Two methods are provided to turn the Batch Interface on or off as desired. The first method uses the 'Z' command code, followed by a command word. The commands are:

- 1) FCTD Z STOPBCH Deactivate the Batch Interface
- 2) FCTD Z STRTBCH Activate the Batch Interface

The STOPBCH command may be entered as an on-line command or through the batch command processor program. When executed, it causes the on-line version of AUTOMON/BATCH to terminate the Batch Interface program from any further activity.

The STRTBCH command causes the on-line version of AUTOMON/BATCH to restart the Batch Interface program, which then waits for a command from another region or partition.

The second method of starting and stopping the batch interface is explained in the next topic, *THE BATCH INTERFACE SCREEN*.

THE BATCH INTERFACE SCREEN

This screen is an accumulation of counters on commands issued from the batch program. This screen may also be used to activate or deactivate the Batch Interface.

To display the BATCH INTERFACE screen, from the FILE SERVICE MANAGER menu, press PF5.

The screen will appear as follows:

```

                                C I C S - F I L E S E R V
                                DATE      : 11/11
                                BATCH I N T E R F A C E D I S P L A Y    TIME      : 16:00
                                STATUS INACTIVE                          TERMID   : 2002
                                INTERVAL 0010

                                (A)CTIVATE    0000
                                (B)ACKOUT     0000
                                (C)LOSE       0000
                                (E)NABLE      0000
                                (O)PEN        0000
                                (Q)DISABLE    0000
                                (S)TART       0000
                                (V)ARY        0000
                                (X)ALLOCATE   0000
                                (Y)DEALLOCATE 0000

ENTER PFKEY OR FUNCTION CODE ==>
CICS-FILESERV RELEASE 3.0      BATCH INTERFACE DISPLAY
1=(H)ELP                      3=(E)XIT          10=(S)TART          11=(T)OP
```

In addition to the STRTBCH and STOPBCH commands, the Batch Interface may be controlled from this screen using the PF10 and PF11 keys, respectively.

BATCH INTERFACE SCREEN - FIELD DESCRIPTIONS:

DATE	This is the current date in MM/DD/YY format.
TIME	This is the current time in HH:MM format.
TERMINID	This is the terminal ID of this terminal.
STATUS	The current status of the or FCTDBCHT program. One of the following will be displayed: ACTIVE The Batch Interface is currently activated. It has been started with an FCTD Z STRTBCH command or a PLTPI entry when CICS was started. INACTIVE The Batch Interface is currently deactivated. It has been stopped with an FCTD Z STOPBCH command or a PLTPI entry was not present when CICS was started and it has not yet been started with an FCTD Z STRTBCH command.
INTERVAL	This is the wake-up interval for the FCTDBCHT program. It is displayed in MMSS format, where: MM= minutes (0-59) SS = seconds (0-59)
ACTIVATE	This is the total number of ACTIVATE commands that have been received from a batch region or partition via the Batch Interface program.
BACKOUT	This is the total number of BACKOUT commands that have been received from a batch region or partition via the Batch Interface program.
CLOSE	This is the total number of CLOSE commands that have been received from a batch region or partition via the Batch Interface program.
ENABLE	This is the total number of ENABLE commands that have been received from a batch region or partition via the Batch Interface program.
OPEN	This is the total number of OPEN commands that have been received from a batch region or partition via the Batch Interface program.
DISABLE	This is the total number of DISABLE commands that have been received from a batch region or partition via the Batch Interface program.
START	This is the total number of START commands that have been received from a batch region or partition via the Batch Interface program.
VARY	This is the total number of VARY commands that have been received from a batch region or partition via the Batch Interface program.
ALLOCATE	This is the total number of ALLOCATE commands that have been received from a batch region or partition via the Batch Interface program.
DEALLOCATE	This is the total number of DEALLOCATE commands that have been received from a batch region or partition via the Batch Interface program.

ENTERING COMMANDS AT THE BATCH INTERFACE STATISTICS SCREEN

The PF key list at the bottom of the screen displays the available options that can be performed at this point.

PF1=HELP	Pressing PF1 will display help for the display.
PF3=EXIT	Pressing PF3 will exit the BATCH INTERFACE screen and return to the FILE SERVICE MANAGER menu.
PF10=START	Pressing PF10 will activate the Batch Interface. This has the same effect as issuing the FCTD Z STRTBCH command from a clear screen.
PF11=STOP	Pressing PF11 will deactivate the Batch Interface. This has the same effect as issuing the FCTD Z STOPBCH command from a clear screen.

TECHNICAL CONSIDERATIONS

The Batch Interface to AUTOMON/BATCH works as follows:

- 1) When AUTOMON/BATCH is initialized at CICS Start-up, program module FCTD1000 starts the FCTD transaction with the specified interval (BCHINTV=mmss). The started transaction is not attached to a terminal..
- 2) The FCTD transaction does a WAIT on the interval timer for the number of minutes and seconds specified by BCHINTV in the User Option Table.
- 3) When the FCTDBCHT program is run, it writes all commands to control records in FCT\$FIL, which has a key equal to the APPLID value in the User Option Table.
- 4) When the timer expires in CICS, the control record in FCT\$FIL is read to see if a command has been set by a batch partition or region.
- 5) If any commands have been set, the on-line AUTOMON/BATCH transaction processes all commands before going back into a timer wait.
- 6) The on-line FCTD transaction writes the response from each command back to the file, where the batch program picks it up and displays both the command and the response.

The System Programmer and Operations personnel should be aware of the following considerations and operating procedures when using the Batch Interface:

- 1) The Batch Interface will not show an FCTD transaction on a CEMT TASK display unless the CEMT command is done while FCTDBCHT is actually running.
- 2) If the designated CICS system is not active or if the Batch Interface has been stopped when FCTDBCHT tries to execute a command, a message to that effect will display on the system console and the command will be ignored by the command processor program. It will try each command, then go to end-of-job.
- 3) Multiple-file OPEN commands execute in the Batch Interface just like they do on-line. That is, if a VSAM open error occurs on one of the files being opened, the error is ignored and OPEN is attempted for the remaining files in the command. The IBM VSAM open error message will display on the system console.

4) Various error conditions can occur during operation of the Batch Interface because of some of the special commands that are used, or because of conditions in the on-line CICS system. The BATCH INTERFACE MESSAGES in the MESSAGE section provide an explanation of the errors that are unique to the Batch Interface. Any other message number that occurs is the same message that would be received if the command were issued on-line.

5) When multiple partitions or regions are running the Batch Interface command processor program simultaneously, the batch interface will interleave the commands, processing them according to system dispatch priority

6) AUTOMON/BATCH commands may also be issued through the operator interrupt routine of CICS on the system console (if the console is defined in the TCT). When this option is used, the response on row 24 is all that is displayed back to the console, just as when using the Batch Interface.

THE PROGRAMMER INTERFACE TO AUTOMON/BATCH

Included is a Programmer Interface to AUTOMON/BATCH that allows certain AUTOMON/BATCH commands to be issued from a user program. The program may be written in either Macro Level or Command Level CICS. In general the Programmer Interface to AUTOMON/BATCH works as follows:

- 1) The application program must format any AUTOMON/BATCH command in the CICS Transaction Work Area (TWA) or in a COMMAREA in the same format that it would be entered from a terminal.
- 2) The application performs a LINK to program FCTD1000.
- 3) AUTOMON/BATCH will attempt to perform the requested command.
- 4) The normal message that AUTOMON/BATCH would produce in response to the command if it were entered from a terminal is returned in the second position of the TWA or the COMMAREA, depending upon which method was used to pass the command.
- 5) The first byte of the TWA or COMMAREA is used as a return code indicating the success or failure of the command. This byte will contain a X'00' (low values) if the command was successful and a X'FF' (high values) if the command failed.

COMANDS THAT MAY BE ISSUED FROM A USER PROGRAM

The following AUTOMON/BATCH commands may be issued from a user program using the Programmer Interface to AUTOMON/BATCH:

ACTIVATE	Activate one or more files.
BACKOUT	Backout one or more files.
CLOSE	Close one or more files.
OPEN	Open one or more files.
ENABLE	Enable a transaction or program or a transaction/program family.
DISABLE	Disable a transaction or program or a transaction/program family.
ALLOCATE	Allocate one or more files (MVS only).
DEALLOCATE	Deallocate one or more files (MVS only).

The command must be formatted in the TWA or COMMAREA exactly as it would be if entered from a clear screen. That is, the first four bytes must contain FCTD, then a space (or comma), then the 1-byte command, then a space, then the dataset ID (up to eight bytes).

CONSIDERATIONS FOR USING THE PROGRAMMER INTERFACE

The following rules apply when issuing AUTOMON/BATCH commands from a user program:

- 1) The program can be written in Assembler or COBOL using Command Level.

- 2) AUTOMON/BATCH will not produce any terminal output displays when it is called from a user program. The completion message, which normally is returned on row 24 of the terminal, is returned to the calling program in positions 2 thru 71 of the COMMAREA.

- 3) The completion response code is returned in position 1 of the COMMAREA. Low values (X'00') indicate successful completion. High values (X'FF') indicate unsuccessful completion.

PROGRAMMER INTERFACE EXAMPLES

Following are two examples of programs using the AUTOMON/BATCH Programmer Interface. The first example is using Command Level COBOL, the second uses Macro Level Assembler. Both programs open one file, then, if successful, open another file and display the AUTOMON/BATCH message on the terminal.

COMMAND LEVEL COBOL EXAMPLE

IDENTIFICATION DIVISION.
PROGRAM-ID. FCTDCALL.

DATA DIVISION.
WORKING-STORAGE SECTION.

77 COMMLENG PIC S9(4) COMP.

LINKAGE SECTION.

01 DFHBL LDS.
02 FILLER PIC S9(8) COMP.
02 COMM-POINTER PIC S9(8) COMP.

01 COMM-AREA.
05 ABATCH-COMMAND-AREA.
10 FCTD-TRANCODE PIC X(4).
10 FILLER PIC X.
10 ABATCH-COMMAND PIC X.
10 FILLER PIC X.
10 ABATCH-FILE-ID PIC X(8).
10 FILLER X(56).
05 ABATCH-RESPONSE-AREA REDEFINES ABATCH-COMMAND-AREA.
10 RESPONSE-CODE PIC X.
10 RESPONSE-MESSAGE PIC X(70).

PROCEDURE DIVISION.

*

**** BUILD ABATCH OPEN COMMANDS AND LINK TO ABATCH

*

MOVE 'FCTD' TO FCTD-TRANCODE.
MOVE 'O' TO ABATCH-COMMAND.
MOVE 'MASTER' TO ABATCH-FILE-ID.
EXEC CICS LINK PROGRAM('FCTD1000') COMMAREA(COMM-AREA)
LENGTH(COMMLENG) END-EXEC.
IF RESPONSE-CODE = HIGH-VALUES
GO TO END-OF-COMMANDS.

MOVE 'FCTD' TO FCTD-TRANCODE.
MOVE 'O' TO ABATCH-COMMAND.
MOVE 'PROSPCT' TO ABATCH-FILE-ID.
EXEC CICS LINK PROGRAM('FCTD1000') COMMAREA(COMM-AREA)
LENGTH(COMMLENG) END-EXEC
EXEC CICS SEND FROM(RESPONSE-MESSAGE) END-EXEC.
EXEC CICS RETURN END-EXEC.

THE FILE SERVICE FUNCTIONS OF AUTOMON/BATCH

File service to any type of file record is fast and easy when using this on-line utility. The user has the ability to create a new record or quickly fetch any existing record for display, modification or deletion.

Record data is conveniently displayed in both character and hex formats. The relative location of the data within the record is indicated on the screen. Scanning, scrolling, paging and search commands make it extremely easy to locate any desired information. Entire files can be browsed both forward and backward for record selection and display.

Records can be retrieved by key, record location or character string searches. VSAM, BDAM or even sequential files may be readily accessed.

Modification of the record data, by authorized utility users, can be performed in either hex or character format. New records may be created on-line and added to files. Undesired existing records can be deleted from files.

The powerful LOCATE functions allows an entire file to be searched for records qualifying for up to ten lines of selection "query" criteria. Selected records may be displayed, printed, deleted, added to another file, modified and rewritten, or modified and added to another file.

Security is maintained at three levels within AUTOMON/BATCH, in addition to the system's normal security facilities. Record display and update functions may be effectively limited to authorized users only.

Help windows give interactive assistance to the utility user at just the press of a key. Windows are used to give a simultaneous view of both the function in use and the operating instructions.

GENERAL OPERATION

To invoke the file service functions of AUTOMON/BATCH, press PF2 from the FILE SERVICE MANAGER menu or enter the following transaction code from a clear screen:

FSRV

The following information screen will be displayed:

```

                                C I C S - F I L E S E R V
                                RELEASE 3.0 , PTF LEVEL 91xxxx
                                DATE      : 11/19/90
                                TIME      : 10:08
                                TERMDID   : VOO1

                                F I L E   S E R V I C E   U T I L I T Y

ALLOWS DISPLAY, UPDATE, DELETE, ADDITION TO VSAM, ISAM AND BDAM FILES.
FILES MUST EXIST IN THE FCT OR BE DYNAMICALLY ADDED BY FCTD.

THE FOLLOWING COMMANDS ARE AVAILABLE ...
(H) HELP      - REQUEST OPERATOR INSTRUCTIONS (N) NEW      - ADD NEW DATA RECORD
(I) INQUIRE   - DISPLAY A DATA RECORD      (D) DELETE     - DELETE DATA RECORD
(C) CHANGE    - CHANGE (MODIFY) DATA RECORD (U) UP       - SCROLL UP ONE LINE
(P) PAGE      - PAGE THROUGH SINGLE RECORD (W) DOWN     - SCROLL DOWN ONE LINE
(F) FORWARD   - BROWSE FORWARD              (B) BACK      - BROWSE BACKWARD
(L) LOCATE    - SEARCH FOR QUALIFIED RECORD  (Q) QUEUE    - PRINT RECORD
(T) TOGGLE    - ALTERNATE HEX/CHAR MODE     (R) RETURN   - RETURN TO MENU

COMMANDS MAY BE ENTERED FROM A CLEAR SCREEN OR FROM THIS SCREEN.
YOU MAY FOLLOW THE DATASET ID WITH THE RECORD KEY IN HEX OR CHAR.
HEX KEYS MUST BE ENTERED AS X'...' . FOR PARTIAL KEY, END WITH * .

ENTER COMMAND => _ _____
KEY => _____
```

INVOKING COMMANDS

Enter the COMMAND, DATASET NAME and RECORD KEY in the fields provided on the above screen, or ...

From a clear screen, use the following as a format for any entry to the FSRV Transaction:

FSRV C DDDDDDDD NNNN...NNNN +PPPP

Where ...

FSRV	=	Transaction Code
C	=	Command Code
DDDDDDDD	=	Dataset Name
NNNN...NNNN	=	Record Key
+PPPP	=	Optional starting position

Each field above must be separated by at least one space when entered. When a record is displayed, the Transaction Code, Command Code, Dataset Name and Record Key are re-displayed in the upper left-hand corner of the screen.

If you omit the Dataset Name or the Record Key, you will be prompted for these values.

To invoke a different Command, Dataset or Record Key at any time, key the change in the command line and press ENTER.

PF3 is used throughout AUTOMON/BATCH as the RETURN function. PF3 (or the 'R' command) will always return to the next higher level from whatever function you are currently performing. When a RETURN command is issued from the opening menu, above, the FSRV utility is terminated and the FILE SERVICE MANAGER menu is displayed.

Press the CLEAR key at any time to completely exit the FSRV utility.

DESCRIPTION OF MENU SCREEN FIELDS

COMMAND CODES

The following table lists the Command Code along with its corresponding PF Key and the Function Description.

<u>CODE</u>	<u>KEY</u>	<u>FUNCTION DESCRIPTION</u>
B	PF7	Browse backward through the file
C	PF9	Change (modify) a data record
D	PF11	Delete a data record
F	PF8	Browse forward through the file
H	PF1	Request a help screen for the current function
I	PF17	Inquire and optionally Modify a data record
L	PF13	Locate and optionally modify multiple records
N	PF10	Add a new data record to the file
P	PF4	Page forward through multiple pages of one record
Q	PF15	Print the record on an on-line printer
R	PF3	Return to the previous logical level
S	PF12	Scan a data record for hex or character string
T	PF5	Toggle between hex and character input mode
U	PF2	Scroll the record up one line on the screen
W	PF6	Scroll the record down one line on the screen
X	PF14	Toggle uppercase translation on and off

Command codes may be entered as an 'alpha' code or by pressing the 'PF' key associated with that command.

DATASET NAME

The Dataset Name is a required field. This is the ID of any dataset that has been previously defined to CICS, either through the standard CICS File Control Table (FCT) or dynamically through the use of AUTOMON/BATCH.

For **DOS**, the value assigned should not exceed 7 characters. This is the same value used as the file ID in the DLBL statement.

For **OS**, the value assigned should not exceed 8 characters. This value is the same value used as the name field in the DD statement.

RECORD KEY

The Record Key is a required field and must be entered for all commands except 'N' (New Record) and 'H' (Help).

The Record Key format is dependent upon the type of file being accessed. The key may be entered in Character or Hex mode.

[Note]: For variable block files, this will be the block and relative record specifications.

CHARACTER MODE

Enter the key in character format using no enclosing delimiters.

Example: A5019X

Enter the key in character format using enclosing quotes if the key contains embedded spaces, single quotes, commas, underscores, etc. In other words, any character that could be construed by AUTOMON/BATCH to be a key input delimiter must be enclosed in quotes in the following manner or entered in hex format.

Example: C'ABC,_ 'DEF' (contains delimiters)

HEX MODE

Precede the key with X' and terminate with a single quotation mark (').

Example: X'C14040F0F1F20001'

MIXED CHARACTER AND HEX MODE

A record key consisting of mixed character and hexadecimal data can be entered by separating each hex and character portion of the key with a command and enclosing the hex portions with the hex delimiters (X'.....'). There must be no imbedded spaces.

Example: PN50,X'0000',999,X'4001'
X'404040',002345

GENERIC KEY

The Generic Key or partial key is specified by terminating the key field with an asterisk (*). Records with the exact or closest key will be retrieved and displayed on the screen. For a hexadecimal generic key, replace the ending quote with the asterisk.

Example: 002*
A*
X'4040*
P80,X'00*
C'678'*

RECORD NUMBER

For BDAM, SAM and VSAM RRDS files, the record key is a 1 to 8 byte relative record number. It may be entered without leading zeros.

Example: 290
001005
1

RELATIVE BYTE ADDRESS

For VSAM ESDS files, the record key is the RBA of the requested record. It must be entered as hexadecimal data, with or without leading zeros

Example: X'00A0'
X'0'
X'CAF4'

If a required Record Key is omitted, you will be prompted for the key in the format that is correct for the record type.

The following table describes the Key Format for each file type:

VSAM KSDS	1 to 64 byte alpha-numeric data
VSAM RRDS or BDAM FILES	1 to 8 byte relative record number
VSAM ESDS FILES	1 to 8 byte relative byte address

STARTING RECORD POSITION

If the record to be displayed exceeds 340 positions in length, it will not all display on one page of the screen. You can optionally follow the record key with a plus sign (+) followed by a 1 to 4 digit number, which represents the position in the record where you want the display to start. The record position is considered as relative to 1. That is, the number 1 represents position 1 of the record, 500 = position 500, etc.

THE GENERAL PURPOSE DATA DISPLAY SCREEN

The FSRV transaction has a general purpose record display screen that is used to:

- 1) Display and modify an existing record
- 2) View a record before deletion
- 3) Enter data during the creation of a new record

The command line in the upper left corner of the screen allows the user to specify the function, file name and record key. The PF keys associated with the various functions are also active when using this screen.

The following screen is used for all functions of data record access:

```

FSRV C DDDDDDDD NNN....NNN
(1) (2) (3) (4) FSRV RECORD DISPLAY FOR FILE _____ LENGTH=00000
(5) (6)
HEX .0.1 .2.3 .4.5 .6.7 .8.9 .A.B.C.D.E.F .0.1 .2.3 CHAR 1...5...10...15...20
000 _____ (7) _____ 001 _____ (8) _____
014 _____ _____ _____ _____ 021 _____
028 _____ _____ _____ _____ 041 _____
03C _____ _____ _____ _____ 061 _____
050 _____ _____ _____ _____ 081 _____
064 _____ _____ _____ _____ 101 _____
078 _____ _____ _____ _____ 121 _____
08C _____ _____ _____ _____ 141 (11) (12) _____
0A0 (9) _____ (10) _____ 161 _____
0B4 _____ _____ _____ _____ 181 _____
0C8 _____ _____ _____ _____ 201 _____
0DC _____ _____ _____ _____ 221 _____
0F0 _____ _____ _____ _____ 241 _____
104 _____ _____ _____ _____ 261 _____
118 _____ _____ _____ _____ 281 _____
12C _____ _____ _____ _____ 301 _____
140 _____ _____ _____ _____ 321 _____

(13)
F1=(H)ELP F3=(R)ETN F5=(T)OGL F7=(B)ACK F9=(C)HNG F11=(D)ELET
F2=(U)P F4=(O)RG F6=(W)DOWN F8=(F)WD F10=(N)EW F12=(S)CAN
    
```

DESCRIPTION OF DISPLAY SCREEN FIELDS:

The small bold numbers in parentheses do not appear on the display screen. They are used here to designate the display fields. The following gives a description of each field designated:

- (1) Transaction code (FSRV).
- (2) Command Code

This is the 'current' command code. The cursor will automatically be positioned in this field for all functions except 'N' (NEW Record).

The commands available are displayed at the bottom of the screen. A different command code may be entered in this field to request a different function OR the corresponding PF Key may be used.

(3) Dataset Name

This is the Dataset Name of the current record displayed. A different Dataset Name may be entered in this field to request a different Dataset to be used.

(4) Record Key

This is the Record Key of the current record displayed. A different Record Key may be entered in this field to request a display of the record containing the entered key.

[Note]: For variable blocked BDAM files, this will be the BLOCK and RECORD numbers. To the right of these fields will be two additional fields labeled BLKSIZ and RPB, which is the blocksize and records per block.

(5) Current Dataset Name

Displays the current Dataset Name - for informational purposes only.

(6) Record Length

This is the length of any variable length record being displayed. The length of fixed - length records will not be displayed. When creating new records, enter the desired record length or use the default of 340. The record length of any variable length record may be changed by modifying this field.

(7) Hexadecimal Horizontal Scale

This series of two (2) digit fields designates the position of a hex byte relative to the start of each line. The period (.) marks the first digit of the byte; the number indicates the byte's relative position. The Hexadecimal Horizontal Scale starts with zero.

(8) Character Horizontal Scale

This series of one (1) digit fields designates the position of a character byte relative to the beginning of each line. The Character Horizontal Scale starts with one.

(9) Hexadecimal Vertical Scale

This scale designates the position of the first byte of data relative to the start of the record. This scale will change when scrolling or paging is performed. The Hexadecimal Vertical Scale starts with zero.

(10) Hexadecimal Data Lines

The screen allows up to 17 lines of hex data. Each field contains 20 bytes (40 digits) of data.

(11) Character Vertical Scale

This scale designates the position of the first byte of data on the line relative to the start of the record. This scale will change when scrolling or paging is performed. The Character Vertical Scale starts with one.

(12)

Character Data Line

The same twenty (20) bytes of data displayed on the left side of the screen in 'hex' form are displayed on the right side in 'character' form. Up to 17 lines of data are displayed.

Non-displayable characters are displayed as an underscore (_).

When re-entering data using the character form, the use of an underscore will prevent the byte from being changed in the record. This prevents destroying non-displayable hex codes.

If the underscore is blanked out with the space bar, the corresponding byte in the record will be replaced with a space (X'40').

Actual underscores in a data record **MUST** be entered in hex mode as a 6D.

(13)

Command Prompts

Displays the available commands that can be issued at this point.

Commands can be invoked by changing the command code in the command line at the top of the screen and pressing ENTER, or by pressing the appropriate PF key.

DISPLAYING A RECORD

Records from various file types can be displayed, one record at a time, using the INQUIRY (I) function of FSRV.

The INQUIRY function will display the record using the General Purpose Record Display screen.

PROGRAM OPERATION

To display a record in the file, enter the transaction as follows:

FSRV I DDDDDDDD NNNN....NNNN +PPPP

Where ...

FSRV	=	Transaction Code
I	=	Command Code for Inquiry
DDDDDDDD	=	Requested Dataset Name
NNNN.....NNNN	=	Record Key for this file type
+PPPP	=	Optional starting record position

The Transaction Code, Command Code, and Dataset Name must be separated by at least one space.

[Note]: For variable block BDAM files, key only the transaction code, command code and dataset name. You will then be prompted for the block and record number.

Refer to *INVOKING COMMANDS* in the topic entitled *GENERAL OPERATION* in this section, for a detailed description of transaction field entries.

Press ENTER after keying the TRANCODE, FUNCTION, DATASET and KEY.

When in display mode, the following options are available:

PAGE: Page function is used to advance the record information displayed on the screen. When the record exceeds 340 positions in length, you may use paging to display the next 340 positions.

P (PF4) Displays the next 340 positions of record or remainder thereof.

R (PF3) If you have paged forward one or more pages, PF3 may be used to return to page 1. If no paging has been performed, PF3 will return to the FILE SERVICE UTILITY MENU.

SCROLL: Scrolling moves the record information up or down on the screen one line at a time.

U (PF2) Scrolling up moves the screen up one line so that the next 20 bytes of data within the record appear on the screen. When the end of the record is reached, continued 'UP' commands will have no effect.

W (PF6) Scrolling down moves the screen down one line so that the previous 20 bytes of data within the record appear on the screen. When the beginning of the record is reached, continued 'DOWN' commands will have no effect.

BROWSE: The Browse command allows the user to sequentially browse an entire file one record at a time.

F (PF8) Forward browse to display the next sequential record on file.

B (PF7) Backward browse to display the previous sequential record on file.

RETURN: The Return command returns the utility to the main menu or next higher level command within a multiple level command.

R (PF3) Return to the next higher level.

SCAN: This command allows a single record to be scanned for the presence of a particular data string.

S (PF12) Scan the record for a data string.

Entry of the command will cause the following message to be displayed:

F0029 ENTER SCAN DATA

The scan data can then be entered in CHARACTER or HEX Mode.

Hexadecimal data is entered as X'...'.

Up to 64 bytes of data may be entered.

Press ENTER after entry of the scan data and the record will be searched. When the scan data is found, the display will appear with the scan data as the first line of data displayed. The record position counter will reflect the position of the data in the record.

If the scan is not found, the message R0030 will display on row 24.

To perform another scan for the same data, press PF6 (or enter command 'W') to scroll UP one line, then enter the SCAN command and data again.

To start or restart a scan from the top of the record, you may press PF3 or enter command 'R' in the command line. This will re-position the record to the beginning and allow another scan to be issued.

MODIFYING A RECORD

Records may be modified by using the CHANGE command of AUTOMON/BATCH. Data may be changed in either hexadecimal or character format. Any position of the data record may be changed with the exception of the key field of a VSAM KSDS file. AUTOMON/BATCH will replace the key field with the original data in the field to insure that a record key is not changed.

In addition, the record length of variable length records may be changed to any valid length.

PROGRAM OPERATION

To modify a record in the file, enter the transaction as follows:

```
FSRV C DDDDDDDD NNNN ....NNNN +PPPP
```

Where ...

FSRV	=	Transaction Code
C	=	Command Code for CHANGE
DDDDDDDD	=	Requested Dataset Name
NNNN.....NNNN	=	Record Key for this file type
+PPPP	=	Optional starting record position

The Transaction Code, Command Code, and Dataset Name must be separated by at least one space.

Refer to *INVOKING COMMANDS* in the topic entitled *GENERAL OPERATION* in this section, for a detailed description of transaction field entries.

Press ENTER after keying the TRANCODE, FUNCTION, DATASET and KEY. This will display the record, making the character side of the display unprotected, thereby allowing data to be altered. The following message will appear on row 24 of the screen:

```
R0002. TO MODIFY, MAKE CHANGES AND PRESS ENTER.
```

When in CHANGE mode, the following options are available:

MODIFY: The data anywhere on the currently displayed page may be modified and the record will then be rewritten into the file.

In order to modify data, move the cursor to the position within the displayed data and key over the existing data.

When keying data on the CHARACTER side of the display, do not erase the underscore characters unless you intend to insert spaces (X'40') into the record in those positions.

[Note]: The erase-EOF Key will not erase. You must use the space bar.

When all modifications have been keyed, press ENTER to rewrite the record in the file.

SPECIAL CONSIDERATIONS FOR LOWERCASE LETTERS

When keying data on the CHARACTER side of the display, you may notice that occasionally one or more lines will be protected, preventing you from changing anything on the line. When this happens, it is because there are one or more lower-case characters somewhere on the line. It is often the case that a packed field on the left side of the screen contains hex digits that represent lowercase letters. This would be the hex characters 81 - 89, 91 - 99 and A1 - A9.

If the line were unprotected and you changed some other character on the line, the normal uppercase translation that CICS does for input data would cause the lowercase character to translate to a different hex byte, thereby changing the value of the packed field inadvertently. AUTOMON/BATCH prevents this situation by protecting the character side on any line that contains lowercase letters. If you need to change something on this line, use the MODE SWITCH command to switch to the hexadecimal side and make the change there.

If you have legitimate lowercase letters in the data record and you wish to make changes on the CHARACTER side, you may use the TRANSLATE (X) command, described below.

MODE SWITCH:

T (PF5) Toggle between HEXADECIMAL and CHARACTER mode input.

The CHARACTER mode is the initial mode for input on the CHANGE screen. Thus, the first time the TOGGLE command is given, the character side of the display screen becomes protected and the HEX side is then unprotected.

The next issue of the TOGGLE command causes the character side to be unprotected and the HEX side protected.

You may switch mode between HEX and CHARACTER as often as desired when entering data.

PAGE: The PAGE function is used to advance the record information displayed on the screen. When the record exceeds 340 positions in length, you may use paging to display the next 340 positions.

P (PF4) Displays the next 340 positions of record or remainder thereof.

R (PF3) If the PAGE key has been pressed, pressing PF3 once will return you to the first page. However if PF3 is pressed again, or if the PAGE key has not been used, PF3 will return you to the AUTOMON/BATCH menu.

SCROLL: Scrolling moves the record information up or down on the screen one line at a time.

U (PF2) Scrolling up moves the screen up one line so that the next 20 bytes of data within the record appear on the screen. When the end of the record is reached, continued 'UP' commands will have no effect.

W (PF6) Scrolling down moves the screen down one line so that the previous 20 bytes of data within the record appear on the screen. When the beginning of the record is reached, continued 'DOWN' commands will have no effect.

TRANSLATE:

X (PF14) This command is provided as alternative to making changes on the HEXADECIMAL side of the display when you have lowercase letters in the data record. The X command will turn OFF the uppercase translation feature of CICS for this terminal and allow the changes to be made on the CHARACTER side of the display.

To use the TRANSLATE command, do the following:

1) Key the 'X' command and press ENTER or press PF14. The following message will display on row 24:

R0046. UPPERCASE TRANSLATION IS OFF

2) Key the 'C' command and press ENTER or press PF9. This puts you back in CHANGE mode and unprotects all lines on the CHARACTER side. (if you're currently on the HEX side, press PF5 or enter 'T' to get back to the character side).

3) Make all desired changes. If you need for a character to be in uppercase, you must hold down the SHIFT key as you enter it.

4) Press ENTER. The data record will be rewritten with the data exactly as it was keyed.

When you want to turn uppercase translation back on, issue the TRANSLATE command again. It works like a toggle switch; the first occurrence turns translation off, the second turns it back on, etc.

If you exit AUTOMON/BATCH (via CLEAR or PF3) while uppercase translation is turned off, the following message will display:

R0047. RESET UPPERCASE TRANSLATION ? (Y/N) Y

If you want to turn it back on, press ENTER. Otherwise, enter 'N' and press ENTER to leave uppercase translation off.

ADDING A RECORD TO THE FILE

New records may be added to existing files through the use of this command. Records may be added to VSAM files only.

PROGRAM OPERATION

To add a new record to the file, enter the transaction as follows:

FSRV N DDDDDDDD

Where ...

FSRV	=	Transaction Code
N	=	Command Code for new record
DDDDDDDD	=	Requested Dataset Name

(Operands Must be separated by at least one space)

Upon pressing ENTER, the GENERAL PURPOSE DISPLAY will appear with the following message on line 24 of the screen.

R0001. KEY DATA FIELDS AND PRESS ENTER TO ADD RECORD

When in Record Addition mode, the following options are available:

NEW DATA: Key data anywhere on the currently displayed page to create a new record for this file.

When keying data on the CHARACTER side of the display, any omitted data positions will insert spaces (X'40') into the data record in those positions.

When all data has been keyed, press ENTER to add the record to the file.

If lowercase letters need to be entered in the data record, these can be entered on the HEXADECIMAL side by using the MODE SWITCH command or by turning off uppercase translation by using the TRANSLATE command.

MODE SWITCH:

T (PF5) Toggle between HEX and CHARACTER Mode input.

The CHARACTER mode is the initial mode for input on the NEW RECORD screen. Thus, the first time the TOGGLE command is given, the character side of the display screen becomes protected and the HEX side is then unprotected.

The next issue of the TOGGLE command causes the character side to be unprotected and the HEX side protected.

You may switch mode between HEX and CHARACTER as often as desired when entering data.

TRANSLATE:

X (PF14) This command is provided as alternative to entering data on the HEXADECIMAL side of the display when you have lowercase letters in the data record. The X command will turn OFF the uppercase translation feature of CICS for this terminal and allow the data to be entered on the CHARACTER side of the display.

To use the TRANSLATE command in record addition mode, do the following:

- 1) Key the 'N' command and press ENTER or press PF10. This puts you in NEW RECORD mode and places the cursor at the first CHARACTER field on the screen.
- 2) Prior to keying any other data, key the X command and press ENTER or press PF14. The following message will display on row 24:

R0046. UPPERCASE TRANSLATION IS OFF

- 3) Enter all desired data. If you need for a character to be in uppercase, you must hold down the SHIFT key as you enter it.
- 4) Press ENTER. The data record will be written with the data exactly as it was keyed.

Note that when the X command is entered, any data that has been keyed in the data record will be cleared if the record has not yet been added. This is why you should issue the TRANSLATE command first, before keying any data. If you have keyed some data and then realize that you need to issue the TRANSLATE command, you should go ahead and press ENTER to add the record, switch to CHANGE mode (enter the 'C' command), then issue the TRANSLATE command and continue entering data.

When you want to turn uppercase translation back on, issue the TRANSLATE command again. It works like a toggle switch: the first occurrence turns translation off, the second turns it back on, etc.

If you exit AUTOMON/BATCH (via CLEAR or PF3) while uppercase translation is turned off, the following message will display:

R0047. RESET UPPERCASE TRANSLATION ? (Y/N) Y

If you want to turn it back on, press ENTER. Otherwise, enter 'N' and press ENTER to leave uppercase translation off.

RECORD TYPE CONSIDERATIONS

When adding a new record to the file, the following considerations apply:

- MULTIPLE-PAGE RECORDS:** When the record is longer than 340 positions in length, enter all required data on the first page and press ENTER to ADD the record.
- Then scroll (PF6 or 'U') or page (PF4 or 'P') to the next available position, change the command to 'C' and press ENTER (or press PF9), key the remaining data and press ENTER.
- VARIABLE-LENGTH RECORDS:** When the file contains variable-length records, the record length will be requested in the upper left-hand corner of the screen.
- The default will be set to 340 positions. Enter the desired record length in this field if other than default.
- CHARACTER/HEX MODE:** The screen is initially set to character input mode. Use the change mode command (PF5 or 'T') for hex mode entry.
- KEYED FILES:** Data must be entered in the actual position(s) of the record that will be used as the key. Failure to do this will result in a key field of all spaces.
- RRDS FILES:** The record number must be specified in the key position of the command line following the file name when the file is a VSAM relative record file.
- There must be at least one space between the file name and the Relative Record Number.
- ESDS FILES:** When the file is a VSAM ESDS file, the record will be added at the next available RBA in the file. The RBA will display when the add is completed.
- BDAM FILES:** The 'N' function can not be used to ADD a record to a BDAM file.
- An empty record must first be retrieved at the desired record number location using the 'I' (INQUIRY) command. The empty record is then filled with data and then updated on the file using the CHANGE command.

DELETING A DATA RECORD

Existing records on files may be deleted through the use of the DELETE command. Deletion is valid for VSAM files only.

PROGRAM OPERATION

To delete a data record, enter the transaction as follows:

FSRV D XXXXXXXX NNNN...NNNN

Where ...

FSRV	= Transaction Code
X	= Requested Dataset Name
NNNN...NNNN	= Record Key of record to be deleted.

When invoked, the record display screen will appear with the requested record and the following message will display on row 24:

R0003. PRESS ENTER TO DELETE THIS RECORD

If deletion is not desired, change an operand in the command line or press PF3 or CLEAR.

PRINTING A RECORD ON AN ON-LINE PRINTER

Any record in a file can be printed on an on-line printer through the use of the QUEUE command. The printer must be a standard terminal printer that is known to CICS by a 4-character ID. It may reside anywhere in the CICS network (it does not have to be on the same control unit with the originating terminal) and may be defined as local or remote.

PROGRAM OPERATION

To print a data record, enter the transaction as follows:

FSRV Q DDDDDDDD NNNN...NNNN

Where ...

FSRV	= Transaction Code.
Q	= Command Code for printing a record.
DDDDDDDD	= Requested Dataset Name.
NNNN...NNNN	= Record Key of record to be printed.

When invoked the following message will display:

R0037. ENTER ON-LINE PRINTER ID _ _ _ _

Enter the four-character ID of the terminal printer. If the ID is not known to CICS, you will receive the message

R0040. INVALID TERMINAL ID REQUESTED.

You must then re-issue the QUEUE command and enter the correct ID.

If the printer is known to CICS, you will receive the message

R0039. RECORD PRINT SUCCESSFULLY STARTED.

This indicates that a print task has been started for the printer. If the printer is currently busy with another task, this record will print when it is finished. Otherwise, the print should commence immediately.

The terminal is now free for any CICS activity.

SEARCHING AND CHANGING MULTIPLE RECORDS - LOCATE

The LOCATE command of AUTOMON/BATCH provides the ability to perform powerful file manipulation functions without the necessity of writing a program.

LOCATE allows you to enter a 'Query' for any VSAM or BDAM file which consists of up to ten fields of selection criteria. Selection information can be connected with AND or OR statements to effectively 'program' the logic to be used in searching the file.

After entering the query and selecting all output options (described below), AUTOMON/BATCH will proceed sequentially through the file, starting with the beginning key or record number that you specify, searching for records which match the selection criteria.

The OUTPUT DISPOSITION OPTIONS which you choose instruct AUTOMON/BATCH what to do with the records that match the selection criteria. You may display them on the terminal, print them on an on-line printer, add them to another file, delete them, or perform simple or complex data modification to them.

To effectively use the LOCATE function, you need to be familiar with the record format of the records in the file to be queried. Fields are identified according to their relative position and length in the record. You should know the data format of the record fields so that you will know when to enter selection data in character or hexadecimal. You may need to know if the file contains variable length records or multiple format records.

PROGRAM OPERATION

To initiate the LOCATE function, enter the transaction as follows:

```
FSRV L DDDDDDDD NNN....NNN
```

Where ...

FSRV	=	Transaction Code
L	=	Command Code Locate
DDDDDDDD	=	Requested Dataset Name
NNN....NNN	=	The starting key or record number.

The starting key or record number will position the file to a certain point prior to beginning the search. All of the rules for key entry which are used in displaying records apply here as well. If you want to start the search at the beginning of the file, enter the key of the first record in the file (this could be a generic key, if desired).

When invoked, the INPUT SELECTION SPECIFICATIONS screen will appear with the following message on line 24 of the screen.

SUPPLY LOCATE SPECIFICATIONS AND PRESS ENTER

The cursor is automatically positioned at the first POSITION field, ready for entry.

THE INPUT SELECTION SPECIFICATIONS SCREEN

The following screen is used to construct the Query, which controls the selection of records from the input file:

```

FSRV L CPXCSL    00000001

                CICS-FILESERV LOCATE FUNCTION
                INPUT SELECTION SPECIFICATIONS

POSITION LENGTH OPERATOR COMPARISON DATA          CONNECTOR

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

____  _____  _____  _____  _____

POSITION          RECORD POSITION, RELATIVE TO (1).
LENGTH           REQUIRED FOR NUMERIC COMPARISON DATA ONLY.
OPERATOR         = (EQUAL),      < (LESS THAN),      > (GREATER THAN),
                ~= (NOT EQUAL),  ~= (NOT LESS THAN),  ~> (NOT GREATER THAN)
COMPARISON DATA L (RANGE) SEPARATE LOW-HIGH LIMITS WITH (-)
CONNECTOR        'AND' OR 'OR'
R0035. SUPPLY LOCATE SPECIFICATIONS AND PRESS ENTER
    
```

INPUT SELECTION FIELD DESCRIPTIONS

You may enter up to ten field selection entries. If more than one line is entered, all lines except the last one must contain a CONNECTOR field (AND or OR). Following is a description of each of the fields and the acceptable entries:

POSITION Enter a number which represents the position of the first byte of the data field to be compared. The number can be entered without leading zeros. The field position must be entered relative to '1'. That is, byte 1 of the record is position 1, byte 2 is position 2, etc.

If an asterisk (*) is entered in the POSITION field (with no other numbers) it signifies a 'record scan'. In this case, each entire record will be scanned for the COMPARISON DATA.

LENGTH Length is not required (but may be entered) if the COMPARISON DATA for this field is alpha-numeric. In this case, the length of the COMPARISON DATA is used as the field length.

If the data to be compared is all numbers, or if alpha-numeric data has trailing spaces, the length must be entered. One or two characters may be entered. The maximum field length for one compare is 35 bytes.

OPERATOR The OPERATOR is the type of compare to be done for this field. OPERATOR codes are :

- = (equals sign) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is equal to the COMPARISON DATA.
- ≠ (not sign, equals sign) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is not equal to the COMPARISON DATA.
- < (left arrow) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is less than the COMPARISON DATA.
- > (right arrow) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is greater than the COMPARISON DATA.
- ↯< (not sign, left arrow) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is not less than the COMPARISON DATA.
- ↯> (not sign, right arrow) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is not greater than the COMPARISON DATA.
- L (low-high limits) Indicates that the condition will be TRUE for this field if the data at the INPUT POSITION for the specified LENGTH is within the range of the first and second comparison data fields.

The first and second comparison data fields are specified by entering the LOW value, followed by a slash (/) and the high value in the COMPARISON DATA field. Both comparison values must be the same length unless @FIELD specification is used (see COMPARISON DATA).

The condition will be TRUE if the data at the INPUT POSITION for the specified LENGTH is equal to or greater than the first comparison value AND less than or equal to the second comparison value.

COMPARISON DATA

The comparison data is the constant value to be compared to the data in the record field at the specified POSITION for the specified LENGTH. Up to 35 bytes of comparison data may be entered. The COMPARISON DATA may be entered in any of the following five ways:

- 1) CHARACTER CONSTANT - A character constant is entered as 1 to 35 bytes of alpha-numeric data or 1 to 16 bytes of numeric data. No quotes or other delimiters need be keyed. Negative numeric values are entered by preceding the number with a minus sign (-).

Example: ABCD
 24ZXY

105

-3300

2) **DELIMITED CHARACTER CONSTANT** - If spaces are to be included in the comparison value, or numeric data is to be compared as if it were alpha-numeric, enclose the CHARACTER CONSTANT (as described above) in single quotes ('). In this case, up to 33 bytes of data (either numeric or alpha-numeric) may be entered.

Example: 'ABCD '
'24 ZXY'
'000000111112222244444'

3) **HEXADECIMAL DATA** - Hexadecimal comparison data can be entered by preceding the value with X' and following the last digit with a single quotation mark ('). Up to 32 digits may be keyed, representing up to 16 bytes of hexadecimal data.

Example: X'0001052C'
X'F0F140400000'

4) **@FIELD SPECIFICATION** - If it is desired that the comparison data be another field in the record rather than a data constant, this can be accomplished by entering an 'at' sign (@) followed by the POSITION specification for the comparison field. The position is entered as 1 to 5 digits, the value of which represents the starting byte of the field to be compared.

Example: @120 @00350

5) **LOW-HIGH LIMITS** - If the OPERATOR code for this field is 'L', the comparison data must be entered as two numeric or alpha-numeric values, separated by a slash (/). The two values must both be the same type (hex, character, etc.) and may be entered in any of the forms discussed above. In all cases except @FIELD specification, the two values must be the same length.

Example: 100/500 X'00100C'/X'00500C'
LLLL/HHHHH @103/@218
' /' ZZ'

CONNECTOR The connector field must be entered if more than one line of selection data is specified. All selection lines except the last line must contain a connector. Connector values are:

AND A connector of AND means that both the current selection line and the next selection line must be TRUE in order for the record to be selected. The word 'AND' or the first letter 'A' can be entered.

OR A connector of OR means that either the current selection line or the next selection line must be TRUE in order for the record to be selected. The word 'OR' or the first letter 'O' can be entered.

If all connectors are AND, every specified condition must be TRUE for the record to be selected.

If all connectors are OR, the record will be selected if any of the specified conditions are TRUE.

When both AND and OR connectors are present, selection logic proceeds from the first condition to the last condition evaluating the condition as follows:

- 1) If the condition is TRUE and the connector is OR, the record is selected.

- 2) If the condition is TRUE and the connector is AND, evaluation continues with the next condition.

- 3) If the condition is FALSE and the connector is OR, evaluation continues with the next condition.

- 4) If the condition is FALSE and the connector is AND, the record is not selected.

NOTES ON INPUT SELECTION FIELDS:

- 1) Alpha-numeric data is compared left-to-right for the specified LENGTH, or the length of the comparison data, if LENGTH is omitted.
- 2) Numeric data is compared algebraically. Both the field data and the comparison data are converted to binary integers and compared. Negative fields will always compare lower than positive fields. A negative value of 200 will compare lower than a negative value of 100.
- 3) When the COMPARISON DATA is numeric, you may specify a negative value by preceding the value with a minus sign (-).
- 4) When the COMPARISON DATA is numeric, AUTOMON/BATCH validates that the field POSITION contains a numeric value before attempting the compare. The numeric value can be Packed or Zoned-Decimal in format. If a valid numeric value is not present in the field, the selection will stop and an error to that effect displayed. You may bypass the record and continue from that point if desired.
- 5) To compare to binary numbers in a data record, use hexadecimal specification for the COMPARISON DATA.
- 6) The characters ' (quote), @ (at-sign) and / (slash) can not be used for COMPARISON DATA since they will be interpreted as special delimiters by AUTOMON/BATCH. If you need to compare to these characters you must specify them in hexadecimal.
- 7) When the Record Scan option is used (* in POSITION), it takes AUTOMON/BATCH longer to evaluate each record to see if it should be selected, since every byte of the input record must be compared to every selection entry. You can expect a slower response using this function, particularly if the records in the file are large.

EXAMPLES OF INPUT SELECTION

The following examples illustrate the coding for various types of selection queries. An explanation follows each example.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00100	---	=	PARTNUM	
---	---	---	_____	---
---	---	---	_____	---
---	---	---	_____	---
---	---	---	_____	---

This is a simple query. Position 100 of the data record is compared for the value 'PARTNUM'. The length of the compare will be six bytes. All records which are EQUAL will be selected.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00100	_5	>	300	
_____	---	---	_____	---
_____	---	---	_____	---
_____	---	---	_____	---
_____	---	---	_____	---

This is a simple numeric query. The 5-byte field at position 100 of the data record is compared for a numeric value of '300'. All records which contain a positive number of '301' or greater will be selected. The data field in the record may be packed or unpacked.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00085	_3	<	1000	AND
00085	_3	>	5000	
_____	---	---	_____	---
_____	---	---	_____	---
_____	---	---	_____	---

In this example, the 3-byte field at position 85 of the data record is compared for a numeric value which is less-than 10,000 and greater-than 5,000. All records which contain a positive number of 5,001 through 9,999 or will be selected. The data field in the record must be packed since a 3-byte unpacked field could not contain these values.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00085	_3	L	5000/10000	OR
00085	_3	=	0	
_____	---	---	_____	---
_____	---	---	_____	---
_____	---	---	_____	---

In this example, the 3-byte field at position 85 of the data record is compared for a numeric value which is equal-to or greater-than 5,000 and equal-to or less-than 10,000, or is equal-to zero. All records which contain zeros or a positive number of 5,000 through 10,000 or will be selected. The data field in the record must be packed since a 3-byte unpacked field could not contain these values.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00210	_8	¬=	X'0000000000000000'	A
00062	10	¬=	'ALPHA BETA'	O
_130	_4	L_	120/320	
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

In order for a record to be selected using this query, position 210 for 8 bytes must NOT consist entirely of binary zeros AND position 62 for 10 bytes must NOT contain 'ALPHA BETA'. If this condition is TRUE, the record will be selected. If this condition is FALSE (210 for 8 bytes contains only binary zeros or 62 contains 'ALPHA BETA'), the record can still be selected if position 130 contains a 4-byte numeric value in the range or 120 through 320.

Remember that multiple NOT EQUAL conditions (of the same position) must always be connected by AND. NOT EQUAL conditions connected by OR will always be TRUE, resulting in record selection of unwanted records (see *NOTES ON EXAMPLES*, below).

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
*	___	=	X'507D'	O
*	___	=	'POLICY'	
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

In this example, the entire record will be scanned for a hexadecimal value of 507D occurring in any two consecutive bytes OR a character value of 'POLICY' occurring in any six consecutive bytes. The record will be selected if either value is found.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00540	_6	<	@235	AND
00540	_6	¬=	0	
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

In this query, the 6-byte field at position 540 of the data record is compared to position 235 of the data record. If position 540 contains an alpha-numeric value which is less-than the 6-byte value at position 235 AND if it contains a non-zero numeric value it will be selected.

Note that when using @FIELD specification, it is not possible to specify a numeric compare.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00085	_3	>	00 _____	O
00085	_3	=	-100 _____	
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

In this example, the 3-byte field at position 85 of the data record is compared for a positive numeric value (greater-than zero) OR a negative value of -100. The record will be selected if either condition is met.

POSITION	LENGTH	OPERATOR	COMPARISON DATA	CONNECTOR
00018	5	L	@25/30	
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

This is an example of @FIELD specification and LOW/HIGH limit checking. The 5-byte field at position 18 must be equal-to or greater-than the 5-byte value at position 25 AND equal-to or less-than the value at position 30 in order to be selected.

NOTES ON INPUT SELECTION EXAMPLES

1) When using NOT-EQUAL (\neq) operators for multiple conditions for the same position, always use AND as the connector. Using OR will cause unwanted records to be selected. The following decision logic shows why this is so:

QUERY - POSITION 10 NOT EQUAL 'AAA' OR
POSITION 10 NOT EQUAL 'BBB'

Assume: Position 10 contains 'BBB'

Logic: Does position 10 contain 'AAA' ?

Answer: No

First condition result: TRUE

Logic: Is connector OR ?

Answer: Yes

Result: Select the record since a TRUE condition was found with an OR connector. In fact, the field contains 'BBB' and was not intended to be selected.

If the connector were AND, the second condition would be tested and found FALSE, therefore the record would not be selected.

2) When using @FIELD specification, you cannot specify that a numeric compare be done.

3) When using the Record Scan option (* in POSITION), it is illogical to use the @FIELD specification since every byte of the record is being compared. The @FIELD position will ultimately be compared to itself.

4) When comparing to 1-byte numeric fields, it is advisable to use the DELIMITED CONSTANT specification (enclose the comparison data byte in quotes). This is because AUTOMON/BATCH allows the last byte of a numeric field to contain a 'C' or 'D' in the zone portion of the byte. This is interpreted as the sign (positive or negative) of a zoned-decimal numeric field.

A 1-byte numeric field, therefore, could validly contain the hexadecimal characters 'F0' through 'F9', 'C0' through 'C9' and 'D0' through 'D9'. Thus the alpha characters 'A' through 'I' and 'J' through 'R' will be interpreted as numeric characters. 'A' through 'I' would compare equal-to a positive '1' through '9' and 'J' through 'R' would compare equal-to a negative '1' through '9'.

If you know for sure that the data field in the record contains only numeric data there is no problem. If there is any doubt, however, enclosing the comparison data byte in quotes will force an alpha-numeric compare which would give the correct results.

5) The following method is used to determine if a field in the data record contains all numeric data:

- If every byte in the field except the last byte contains X'F0' through X'F9' and the last byte contains X'F0' - X'F9', X'C0' - X'C9' or X'D0' - X'D9' it is a valid zoned-decimal numeric field.

- If every byte except the last byte contains a zone portion of 0 to 9 and a numeric portion of 0 to 9 and the last byte contains a zone portion of 0 to 9 and a numeric portion of C, D or F it is a valid packed-decimal numeric field.

- AUTOMON/BATCH can not determine if a field is a valid binary number. You must use hexadecimal representation for these fields.

OUTPUT DISPOSITION OF SELECTED RECORDS

Following completion of the INPUT SELECTION SPECIFICATION SCREEN, the next screen to appear is the OUTPUT DISPOSITION INSTRUCTIONS SCREEN. On this screen, you instruct AUTOMON/BATCH as to what you want to do with the records which match the input query. Depending on your selection, this may be the last input required for the LOCATE function, or you may need to provide the MODIFY/REFORMAT SPECIFICATIONS.

THE OUTPUT DISPOSITION INSTRUCTIONS SCREEN

The following screen is used to direct the disposition of all records that are selected from the file:

```
FSRV L CPXCSL      X'4040404040404040404040404040404040'
                    CICS-FILESERV LOCATE FUNCTION
                    OUTPUT DISPOSITION INSTRUCTIONS

CHOOSE ONE OF THE FOLLOWING OPTIONS FOR THE SELECTED RECORDS:
_  DISPLAY THE RECORD AND OPTIONALLY CONTINUE SEARCH
_  ADD THE UNALTERED RECORD TO FILE _____ AND CONTINUE SEARCH
_  DISPLAY THE RECORD AND OPTIONALLY ADD IT TO FILE _____
_  MODIFY OR REFORMAT THE RECORD AND REWRITE IT INTO SAME FILE
_  MODIFY OR REFORMAT, DISPLAY AND OPTIONALLY REWRITE IT
_  MODIFY OR REFORMAT THE RECORD AND ADD IT TO FILE _____
_  MODIFY OR REFORMAT, DISPLAY AND OPTIONALLY ADD IT TO FILE _____
_  DELETE THE SELECTED RECORD FROM THE FILE
_  DISPLAY THE RECORD AND OPTIONALLY DELETE IT
_  PRINT THE RECORD USING ONLINE PRINTER ____ AND CONTINUE SEARCH

ENTER A "Y" FOR THE DESIRED OPTION AND SUPPLY FILENAME
OR PRINTER IDENTIFICATION AS NEEDED.

ENTER MAXIMUM RECORDS TO SEARCH WITHOUT PAUSING IF NO MATCHES FOUND
DUPLICATE RECORD OPTION (B=BYPASS, R=REPLACE, P=PROMPT) P
ENTER STARTING RELATIVE RECORD NUMBER FOR RRDS ADDITION____

R0035. SUPPLY LOCATE SPECIFICATIONS AND PRESS ENTER
```

SELECTING THE OUTPUT DISPOSITION

Choose the output disposition desired, then TAB to the selection field to the left of the option and enter a 'Y'. If your selection contains a field for the OUTPUT FILENAME or PRINTER ID, enter the CICS name of the receiving file or printer in that field.

If desired, change the MAXIMUM RECORDS TO SEARCH WITHOUT PAUSING. This provides a limit to the number of records which will be searched without finding a record that matches the INPUT SELECTION SPECIFICATIONS. When the limit is reached, AUTOMON/BATCH will issue a message to that effect then ask if you wish to continue the search.

If you chose an option that will add records to another file, enter the desired DUPLICATE RECORD OPTION. This option determines the action to be taken if a duplicate record is found in the output file. The choices are:

- B - Bypass the duplicate record without adding it to the file.
- R - Replace the duplicate record with the selected record.
- P - Prompt the operator for the decision, displaying the record

If you chose an option that will add records to another file and that file is a VSAM RRDS file, you may want to supply the **STARTING RELATIVE RECORD NUMBER FOR RRDS ADDITION**. With this option, the first selected record is added using the specified starting number, then each subsequent selected record is added at the previous record number +1. If you omit this option, you will be prompted for the record number when the record is selected.

If you selected one of the **MODIFY/ REFORMAT** options, you will next receive the **DATA MODIFY/REFORMAT SPECIFICATIONS SCREEN**. Otherwise, the **LOCATE** function will begin upon pressing **ENTER**.

DESCRIPTION OF OUTPUT DISPOSITION OPTIONS

The following discussion details the results of each of the Output Disposition Options.

1) **DISPLAY THE RECORD AND OPTIONALLY CONTINUE SEARCH**

With this option, each record that qualifies for selection based on the input query will be displayed on the terminal. The record is displayed using the General Purpose Data Display Screen. When displayed, the data which matched the query is highlighted on both the hex and character side of the screen. The following message is displayed on row 24 of the screen:

PRESS ENTER TO CONTINUE LOCATE, PF3 TO END LOCATE

At this point you may respond with any of the following:

- Press **ENTER**. This causes the **LOCATE** function to continue through the file, searching for the next record that matches the input query.
- Perform any valid **AUTOMON/BATCH** function on this data record and then continue or terminate the **LOCATE** function. To perform another function, change the command code (which is currently 'L') or press the PF key which invokes the desired function. For instance, to update this record, change the command code to 'I' and press **ENTER** or press **PF9**. Make the desired change and press **ENTER** to update the record. You may then change the command code back to 'L' and press **ENTER** to continue the **LOCATE** function.
- Press **PF3**. This terminates the **LOCATE** function, producing the following message:

LOCATE FUNCTION TERMINATED

- Press **CLEAR**. This terminates the **LOCATE** function and exits from **AUTOMON/BATCH**. You will be prompted with an option to save or purge the **LOCATE** specifications. (see *EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

2) **ADD THE UNALTERED RECORD TO FILE _____ AND CONTINUE SEARCH**

With this option, each record that qualifies for selection based on the input query will be added to another file. The output file name must be entered in the space provided.

The output file must be a VSAM KSDS, RRDS or ESDS file. Records can not be added to a BDAM file. When the record is added to the output file, AUTOMON/BATCH will determine the file type of the output file and add the record according to the following logic:

- 1) For KSDS files, the key location is determined from the CICS File Control Table and the portion of the record that corresponds to the key location and length is used as the record key for addition.
- 2) For ESDS files, the record is added at the next available RBA.
- 3) For RRDS file, you will be prompted for a relative record number to use to add the record unless you provide the starting relative record number in the space provided on this screen.

If a duplicate record is found in the output file (KSDS or RRDS files), it is handled according to the DUPLICATE RECORD OPTION selected on this screen.

If you select 'B' (Bypass), the selected record is bypassed and the LOCATE function continues uninterrupted.

If you select 'R' (Replace), the duplicate record is over-written with the selected record and the LOCATE function continues uninterrupted.

If you select 'P' (Prompt), the following message

DUPLICATE FOUND, ENTER TO REPLACE IT, PF3 TO BYPASS

will be displayed on row 24 along with the selected record, which is displayed using the General Purpose Data Display Screen. At this time, you may respond as follows:

- 1) Press ENTER. This causes the duplicate record on file to be over-written with the selected record. The LOCATE function then continues through the file, searching for the next record which matches the input query.
- 2) Press PF3. This causes the LOCATE function to continue after bypassing the selected record without adding it to the file.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

3)

DISPLAY THE RECORD AND OPTIONALLY ADD IT TO FILE

With this option, each record that qualifies for selection based on the input query will be displayed on the terminal, then, based on the operator response, either bypassed or added to another file. The output file name must be entered in the space provided.

When a record is selected, it is displayed using the General Purpose Data Display Screen. When displayed, the data that matched the query is highlighted on both the hex and character side of the screen. The message

PRESS ENTER TO ADD THIS RECORD, PF3 TO BYPASS

is displayed on row 24 of the screen. At this point you may respond with any of the following:

- 1) Press ENTER. This causes the record to be added to the output file using the logic described below, then the LOCATE continues.
- 2) Press PF3. This causes the LOCATE to continue after bypassing this record without adding it to the file.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

The output file must be a VSAM KSDS, RRDS or ESDS file. Records can not be added to a BDAM file. When the record is added to the output file, AUTOMON/BATCH will determine the file type of the output file and add the record according to the following logic:

- For KSDS files, the key location is determined from the CICS FCT and the portion of the record that corresponds to the key location and length is used as the record key for addition.
- For ESDS files, the record is added at the next available RBA for the file.
- For RRDS file, you will be prompted for a relative record number to use to add the record unless you provide the starting relative record number in the space provided on this screen.

If a duplicate record is found in the output file (KSDS or RRDS files), it is handled according to the DUPLICATE RECORD OPTION selected on this screen.

If you select 'B' (Bypass), the selected record is bypassed and the LOCATE function continues.

If you select 'R' (Replace), the duplicate record is over-written with the selected record and the LOCATE function continues.

If you select 'P' (Prompt), the message

DUPLICATE FOUND, ENTER TO REPLACE IT, PF3 TO BYPASS

will be displayed on row 24. You may respond as follows:

- 1) Press ENTER. This causes the duplicate record on file to be over-written with the selected record. The LOCATE function then continues through the file, searching for the next record which matches the input query.
- 2) Press PF3. This causes the LOCATE function to continue after bypassing the selected record without adding it to the file.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *@EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)
- 4) **MODIFY OR REFORMAT THE RECORD AND REWRITE IT INTO THE SAME FILE**

With this option, each record that qualifies for selection is altered according to the DATA MODIFY/REFORMAT SPECIFICATIONS, then re-written in the input file.

The entire record or only selected fields can be changed using MODIFY/REFORMAT. With this option, the only data that can not be changed is the key field in a KSDS file. AUTOMON/BATCH will replace the key field with the original data prior to re-writing it to insure that the key is not changed.

[Note]: You should use this option with discretion. Since the records are to be modified in-place, you should be very sure that the selection criteria and modify/reformat specifications have been entered correctly. If a mistake is made it could corrupt the input file. You will receive a warning message to this effect whenever you choose this option.

- 5) **MODIFY OR REFORMAT, DISPLAY AND OPTIONALLY REWRITE IT**

With this option, each record that qualifies for selection is altered according to the DATA MODIFY/REFORMAT SPECIFICATIONS, displayed on the terminal, then, depending on the operator response, bypassed or re-written in the input file.

When a record is selected, it is displayed using the General Purpose Data Display Screen. When displayed, the changes specified by the DATA MODIFY/REFORMAT SPECIFICATIONS have been made in memory only. The file record has not been changed. The message

PRESS ENTER TO REWRITE THIS RECORD, PF3 TO BYPASS

is displayed on row 24 of the screen. At this point you may respond with any of the following:

- 1) Press ENTER. This causes the record to be re-written into the input file, then the LOCATE continues.
- 2) Press PF3. This causes the LOCATE to continue after bypassing this record without rewriting it.

- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

The entire record or only selected fields can be changed using MODIFY/REFORMAT. With this option, AUTOMON/BATCH will replace the key field with the original data prior to re-writing it to insure that the key is not changed.

- 6) _____
FILE _____ MODIFY OR REFORMAT THE RECORD AND ADD IT TO

With this option, each record that qualifies for selection is altered according to the DATA MODIFY/REFORMAT SPECIFICATIONS, then added to another file. The entire record or only selected fields may be altered. The output file name must be entered in the space provided.

The output file must be a VSAM KSDS, RRDS or ESDS file. Records can not be added to a BDAM file. When the record is added to the output file, AUTOMON/BATCH will determine the file type of the output file and add the record according to the following logic:

- 1) For KSDS files, the key location is determined from the CICS File Control Table and the portion of the record which corresponds to the key location and length is used as the record key for addition.
- 2) For ESDS files, the record is added at the next available RBA for the file.
- 3) For RRDS file, you will be prompted for a relative record number to use to add the record unless you provide the starting relative record number in the space provided on this screen.

If a duplicate record is found in the output file (KSDS or RRDS files), it is handled according to the DUPLICATE RECORD OPTION selected on this screen.

If you select 'B' (Bypass), the selected record is bypassed and the LOCATE function continues un-interrupted.

If you select 'R' (Replace), the duplicate record is over-written with the selected record and the LOCATE function continues un-interrupted.

If you select 'P' (Prompt), the message

DUPLICATE FOUND, ENTER TO REPLACE IT, PF3 TO BYPASS

will be displayed on row 24 along with the selected record, which is displayed using the General Purpose Data Display Screen. At this time, you may respond as follows:

- 1) Press ENTER. This causes the duplicate record on file to be over-written with the selected record. The LOCATE function then continues through the file, searching for the next record which matches the input query.
- 2) Press PF3. This causes the LOCATE to continue after bypassing the selected record without adding it to the file.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

7)
ADD IT TO

MODIFY OR REFORMAT, DISPLAY AND OPTIONALLY

FILE _____

With this option, each record that qualifies for selection is altered in memory according to the DATA MODIFY/REFORMAT SPECIFICATIONS, displayed on the terminal, then, according to the operator response, bypassed or added to another file. The entire record or only selected fields may be altered. The output file name must be entered in the space provided.

When a record is selected, it is displayed using the General Purpose Data Display Screen. When displayed, the changes specified by the DATA MODIFY/REFORMAT SPECIFICATIONS have been made in memory only. The message

PRESS ENTER TO ADD THIS RECORD, PF3 TO BYPASS

is displayed on row 24 of the screen. At this point you may respond with any of the following:

- 1) Press ENTER. This causes the record to be added to the output file using the logic described below, then the LOCATE continues.
- 2) Press PF3. This causes the LOCATE to continue after bypassing this record without adding it to the file.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *@EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

The output file must be a VSAM KSDS, RRDS or ESDS file. Records can not be added to a BDAM file. When the record is added to the output file, AUTOMON/BATCH will determine the file type of the output file and add the record according to the following logic:

- 1) For KSDS files, the key location is determined from the CICS File Control Table and the portion of the record that corresponds to the key location and length is used as the record key for addition.
- 2) For ESDS files, the record is added at the next available RBA for the file.
- 3) For RRDS file, you will be prompted for a relative record number to use to add the record unless you provide the starting relative record number in the space provided on this screen.

If a duplicate record is found in the output file (KSDS or RRDS files), it is handled according to the DUPLICATE RECORD OPTION selected on this screen.

If you select 'B' (Bypass), the selected record is bypassed and the LOCATE function continues un-interrupted.

If you select 'R' (Replace), the duplicate record is over-written with the selected record and the LOCATE function continues uninterrupted.

If you select 'P' (Prompt), the message

DUPLICATE FOUND, ENTER TO REPLACE IT, PF3 TO BYPASS

will be displayed on row 24. At this time, you may respond as follows:

- 1) Press ENTER. This causes the duplicate record on file to be over-written with the selected record. The LOCATE function then continues through the file, searching for the next record which matches the input query.
- 2) Press PF3. This causes the LOCATE to continue after bypassing the selected record without adding it to the file.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *@EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

8) **DELETE THE SELECTED RECORD FROM THE FILE**

With this option, each record that qualifies for selection based on the input query will be deleted from the input file. This option applies to VSAM files only, records can not be deleted from a BDAM file.

[Note]: You should use this option with discretion. Since the records are to be deleted from the input file, you should be very sure that the selection criteria has been entered correctly. If a mistake is made it could corrupt the input file. You will receive a warning message to this effect whenever you choose this option.

9) **DISPLAY THE RECORD AND OPTIONALLY DELETE IT**

With this option, each record that qualifies for selection based on the input query will be displayed on the terminal, then, depending on the operator response, bypassed or deleted from the input file.

The record is displayed using the General Purpose Data Display Screen. When displayed, the data that matched the query is highlighted on both the hex and character side of the screen. The message

PRESS ENTER TO DELETE THIS RECORD, PF3 TO BYPASS

is displayed on row 24 of the screen. At this point you may respond with any of the following:

- 1) Press ENTER. This causes the record to be deleted from the file, then the LOCATE function will continue through the file, searching for the next record which matches the input query.
- 2) Press PF3. This causes the LOCATE function to continue after bypassing this record without deleting it.
- 3) Press CLEAR. This terminates the LOCATE function and exits from AUTOMON/BATCH. You will be prompted with an option to save or purge the LOCATE specifications. (see *EXITING AUTOMON/BATCH WHEN LOCATE IS ACTIVE*)

10)
CONTINUE SEARCH

PRINT THE RECORD USING ON-LINE PRINTER _____ AND

With this option, each record that qualifies for selection based on the input query will be printed on an on-line terminal printer, the ID of which is entered in the space provided. Record print is continuous; all selected records are sent to the printer without interruption.

As each record is selected, it is displayed on the terminal with a running count of the number of records selected and sent to the printer. You do not need to respond to the display, it is for informational purposes only.

Records are printed in character and hex format, similar to the General Purpose Data Display screen.

Upon completion of the OUTPUT DISPOSITION INSTRUCTIONS screen input, the LOCATE function will begin, unless you selected one of the options which involve modifying or reformatting the record. In this case, the DATA MODIFY/REFORMAT SPECIFICATIONS screen will be presented. The input to this screen is described on the following pages.

ENTERING MODIFY/REFORMAT INSTRUCTIONS

You may enter up to ten lines of modify/reformat instructions on each screen. Each line must contain an OUTPUT POSITION, a FIELD LENGTH, and either an INPUT POSITION or a CONSTANT DATA field.

Following is a description of each of these fields:

OUTPUT POSITION

This is the starting position in the record (relative to 1) that will contain either the CONSTANT DATA field or the data from the INPUT POSITION. Up to five digits may be entered. Leading zeros are not necessary.

FIELD LENGTH

This is the length of the data to be moved into the OUTPUT POSITION field. Up to four digits may be entered. Leading zeros are not necessary.

INPUT POSITION

Either this field or the CONSTANT DATA field, but not both, must be entered. If this field is entered you are specifying the starting position in the record for the data that is to be moved to the OUTPUT POSITION field. The data will be moved from left to right for the specified FIELD LENGTH.

CONSTANT DATA

Either this field or the INPUT POSITION field, but not both, must be entered. If this field is entered it is the actual data which is to be moved to the OUTPUT POSITION field. The data may be entered as a character constant or as hexadecimal data.

Character constants are entered as consecutive bytes of alpha-numeric data. Do not enclose the constant in quotes. If the constant contains imbedded spaces, use the SPACE bar to enter these. Up to 40 bytes of data may be keyed.

Hexadecimal data must be preceded by X' and followed by a closing quote ('). Up to 32 digits, representing up to 16 bytes of hex data, may be keyed.

The data will be moved from left to right for the specified FIELD LENGTH.

Three of the YES/NO fields at the bottom of the screen provide additional instructions for the MODIFY/REFORMAT operation. These are:

CLEAR UNMODIFIED INPUT FIELDS

If you are doing a record reformat which involves moving several selected fields from the input record to the output record, entering a 'Y' for this option will cause AUTOMON/BATCH to clear the entire output record prior to moving the specified fields. The record will be cleared with spaces (X'40').

INCREASE LENGTH TO OUTPUT RESULT

If the output record is for a variable length file, entering a 'Y' in this field provides that for any records in the file whose length is shorter than the highest OUTPUT POSITION plus FIELD LENGTH of all of the specified fields, those records will have their length set to that result value. Records whose length exceeds the highest OUTPUT POSITION plus FIELD LENGTH of all of the specified fields will not have their length changed.

This option insures that the specified modification will be made to all qualifying records, even if a length change is involved.

Specifying 'N' for this options means that you do not want to change any record lengths in a variable length file. If a record is too short to contain all of the modification, any modification beyond the end of the record will not be made. Modifications up to the end of the record will be made.

SET LENGTH TO OUTPUT RESULT

If the output record is for a variable length file, entering a 'Y' in this field provides that the resultant record length will be set to the highest OUTPUT POSITION plus FIELD LENGTH of all of the specified fields.

This option provides for both lengthening and truncation of records. If a record is shorter than the computed result length, it will be increased to that length. If it is longer, it will be decreased to that length.

This option insures that the specified modification will be made to all qualifying records, even if a length change is involved.

Specifying 'N' for this options means that you do not want to change any record lengths in a variable length file. If a record is too short to contain all of the modification, any modification beyond the end of the record will not be made. Modifications up to the end of the record will be made.

MORE REFORMAT SPECIFICATIONS

Enter a 'Y' in this field if you need to enter more modify/reformat specifications. This will provide another screen, on which up to ten more fields can be entered. You may enter a total of forty fields (four full screens).

STARTING THE LOCATE OPERATION

Upon completion of all MODIFY/REFORMAT specifications, or if an output option was selected that does not involve modify/reformat, the LOCATE operation will commence.

Control will proceed sequentially through the file, beginning with the starting record key or record number that you provided. Each input record is evaluated according to the selection criteria and a decision is made as to whether the record qualifies for selection or not.

If a record does not qualify, the next input record is read. Consecutive non-qualifying records are counted and when the count reaches the value specified for MAXIMUM RECORDS TO SEARCH WITH NO QUALIFIERS, the current record is displayed, with the message

L1006. XXXX RECORDS REACHED WITH NO QUALIFIERS, ENTER TO CONTINUE, PF3 TO END LOCATE

on row 24. If you wish to continue the search, press ENTER. To terminate the Locate function at this point, press PF3 or CLEAR.

If a record qualifies, the output disposition option is taken. You may receive one of the following messages, depending on the output option selected:

L1001. PRESS ENTER TO CONTINUE LOCATE, PF3 TO END LOCATE

L1002. PRESS ENTER TO ADD THIS RECORD, PF3 TO BYPASS

L1003. PRESS ENTER TO REWRITE THIS RECORD, PF3 TO BYPASS

L1004. PRESS ENTER TO DELETE THIS RECORD, PF3 TO BYPASS

In all cases, pressing ENTER causes the output disposition option to be taken, while pressing PF3 (or changing the command to 'R') causes the output disposition option to not be taken for this record.

ERROR CONDITIONS

There are two other conditions that can interrupt the Locate operation with a message on row 24. The messages involved are:

DUPLICATE FOUND, ENTER TO REPLACE IT, PF3 TO BYPASS

This message occurs if a record with a duplicate key or record number is found in the output file for one of the output options that add records to another file. The message will only appear if you chose 'P' (Prompt) for the DUPLICATE RECORD OPTION on the Output Dispositions screen.

The record displayed at this point is the duplicate record from the output file. It is not the input record. If you choose to replace the duplicate record with the input record, press ENTER. Pressing PF3 will leave the output record intact and continue with the record selection process.

L1008. NON-NUMERIC DATA AT POSITION xxxx, LENGTH=yy, ENTER=CONTINUE, PF3=END

This message can occur if one or more of the COMPARISON DATA fields from the Selection Specifications screen was a numeric constant. AUTOMON/BATCH has determined that the data in the record at the corresponding position and length is not a valid numeric field (either zoned-decimal or packed-decimal). xxxx in the message indicates the record position, yy is the field length.

Pressing ENTER at this point will cause the Locate operation to continue, but this record will not be selected. Non-numeric data present for a numeric compare will make the specified condition untrue, since an algebraic compare can not be done. You may want to review the process that AUTOMON/BATCH goes through to recognize valid numeric data, which is documented under *NOTES ON INPUT SELECTION FIELDS* in the subject entitled *INPUT SELECTION FIELD DESCRIPTIONS* earlier in this section.

Pressing PF3 at this point will terminate the Locate operation.

You can correct the non-numeric field at this point, if desired, by entering the CHANGE command ('C' or PF9) and then altering the data at the location. If you do this and want the record to be re-evaluated for selection, back-up one record in the file using the 'B' (or PF7) command before changing the command back to 'L' to continue. This will cause the record to be read again and re-evaluated for selection.

POSSIBLE ACTION AT A LOCATE PAUSE

When the Locate operation pauses for any of the reasons described above, there are several actions that can be taken. These are:

- 1) Respond to the pause with ENTER or PF3 to take the indicated action.
- 2) Perform any valid AUTOMON/BATCH command on the currently displayed record, then continue the Locate operation.

You might want to CHANGE the displayed record, PAGE forward in the record, SCROLL up or down, DELETE the record or even add a NEW record at this point. You can do any of these things, then change the command code back to 'L' and press ENTER to continue the Locate operation. The selection process will proceed from that point; you do not have to re-specify the selection criteria.

[Note]: When you continue the Locate operation, the currently displayed record will not be evaluated again for selection. It will be skipped and the operation continued with the next sequential record. If you want it to be re-evaluated, back-up one record in the file ('B' or PF7) before changing the command to 'L' and pressing ENTER.

- 3) Skip forward or backward in the file. You can change the currently displayed record key or record number at any Locate pause, then continue the Locate operation. You can move forward or backward in the file. To do this, simply change the currently displayed key and press ENTER.

If the Locate pause that you are at is one that expects a response concerning the action to take for the currently displayed record (messages L102, L103, L104, L105), AUTOMON/BATCH will recognize the fact that you changed the record key and treat it as if you had pressed PF3. That is, it will not take the action that would normally have resulted from pressing ENTER (adding, rewriting, deleting the record). It will continue the Locate operation with the new key or record number.

- 4) Switch to another input file. You can effectively search more than one file at a time by changing the currently displayed dataset name at a Locate Pause. If you have two files that have similar record formats (so that the selection criteria would apply to both) you could search one file until a pause occurs, change the dataset name and optionally the record key, press ENTER (command code 'L' must still be there) and continue the search in the other file.

When the Dataset name is changed, AUTOMON/BATCH takes the same action as described above for a key change.

CONSIDERATIONS FOR DIFFERENT FILE TYPES OR ATTRIBUTES

When the OUTPUT DISPOSITION OPTION selected is one that adds records to another file, the output file can be a different file type or have different attributes than the input file.

For instance, you can select from a BDAM file and add to a VSAM file. Or select from a VSAM RRDS file and add to a VSAM KSDS file. Any combination is acceptable.

Likewise, you can select from a fixed-length file and add to a variable-length file or vice-versa. Or the key location and length of an output KSDS file could be different from the input file.

All of these situations are valid, but you must be aware of the logic used by AUTOMON/BATCH to handle each case.

The following chart describes the technique used for each situation:

<u>INPUT FILE</u>	<u>OUTPUT FILE</u>	<u>ACTION TAKEN</u>
Non-keyed (BDAM, RRDS, ESDS)	Keyed (KSDS)	The data in the output record (after any modifications) at the key location for key length of the output file is used as the record key.
Keyed (KSDS)	Non-keyed (RRDS, ESDS)	For RRDS, the record number must be supplied. For ESDS, the high-RBA is used to write the record.
Keyed (KSDS)	Keyed with different key location or length.	The data in the output record (after any modifications) at the key location for key length of the output file is used as the record key.
Fixed-length	Fixed-length with larger record size.	The excess data beyond the end of the input (or modified) record is padded with spaces.
Fixed-length	Fixed-length with smaller record size.	The excess data beyond the end of the output record is truncated to the output record size.
Fixed-length	Variable Length	The size of the input record or the computed result size after modifications (if chosen) is used as the output record length. If this exceeds the maximum for the output file, the maximum is used and the excess is truncated.
Variable Length	Fixed length	If the input (or modified) record length is shorter than the output record length, the excess is padded with spaces. If it is longer, the excess is truncated.
Variable Length	Variable Length	The size of the input record or the computed result size after modifications (if chosen) is used as the output record length. If this

exceeds the maximum for the output file, the maximum is used and the excess is truncated.

TERMINATING AND RESTARTING THE LOCATE OPERATION

At any Locate pause, you may terminate the Locate operation by pressing the CLEAR key. When this is done, the message

R0036. PURGE LOCATE SPECIFICATIONS ? (Y/N) Y

will display. If you press ENTER (leaving the 'Y' as the response), the Selection, Output Disposition and Modify/Reformat specifications are purged.

If you change the response to 'N' and press ENTER, all specifications are saved. This means that the next time you enter a LOCATE command, the previously entered Locate specifications are restored and the various specification screens are displayed with the data filled in.

This feature allows you to re-run a particular Locate several times without re-keying the specifications each time.

[Note]: The Locate specifications are only saved for the duration that CICS is up.

THE SECURITY CONTROL SYSTEM

Since AUTOMON/BATCH provides instant access to any file known to CICS, dataset security from unauthorized persons may be an important consideration in your installation.

This need is met by the Security Control System, which is part of the AUTOMON/BATCH package.

With the Security Control System, security can be implemented in three ways. You may optionally choose which method or combination of methods best suits your environment. These three methods are:

- 1) Assigning CICS security keys to the various I/O functions (Inquire, Modification, Addition, Deletion) which may be performed on a File Definition or record.
- 2) Assigning passwords to File Definitions or CICS files.
- 3) Writing a Security User Exit.

[Note]: You may use any combination or all three of these methods concurrently.

ASSIGNING CICS SECURITY KEYS

If your installation uses normal RACF security, the TCT will have security key values stored in it by the system. These keys are still accessible by AUTOMON/BATCH and can be used to allow or limit access to files when using the FCTD and FSRV transactions.

RACF allows you to assign a security key to each transaction. The operator must have that key in their list of security keys to use that transaction.

This type of security works at the transaction level. Use this technique to limit access of AUTOMON/BATCH to only those operators who have a need for this transaction.

An operator could be allowed to INQUIRE into a particular file definition, but may not be allowed to CREATE, UPDATE, or DELETE it.

This is accomplished using the Security Control System. To define a security control record in the SECURITY CONTROL FILE:

- 1) Key from one to forty-four positions of the dataset label of the file to be controlled.
- 2) Assign a security key value for the INQUIRE, UPDATE, NEW RECORD and DELETE functions. A key value of zero means security is not checked for that function.

Once assigned, the operator issues a command to AUTOMON/BATCH for a particular function. AUTOMON/BATCH will ensure that the security key assigned to that function (for any file definition with this label) is contained in the operator's list of security keys before allowing the command to proceed.

The following example illustrates this method for two datasets and three operations:

FCTD or FSRV transaction is assigned security key 12:

Dataset PAYROLL.FILE is assigned the keys:

INQUIRE	=	06
UPDATE	=	09
NEW RECORD	=	07
DELETE	=	09

Dataset VENDOR.FILE is assigned the keys:

INQUIRE	=	00
UPDATE	=	08
NEW RECORD	=	07
DELETE	=	07

Operator A has RACF security keys of 01, 06, 12, and 14:

Operator B has RACF security keys 06, 07, 08 and 12.

Operator C has RACF security keys 06, 07 and 09.

RESULTS:

Operator A can inquire and perform Open/Close commands on PAYROLL.FILE and VENDOR.FILE but can not update, add or delete on either file definition.

Operator B can inquire and perform Open/Close commands on both files, update, add and delete on VENDOR.FILE, but only add on PAYROLL.FILE.

Operator C can not do anything because he does not have security key 12 which allows access to the FCTD transaction.

The procedure for assigning security keys is described under SECURITY CONTROL FILE MAINTENANCE later in this section.

ASSIGNING PASSWORDS

When setting up the dataset records in the SECURITY CONTROL FILE, you may optionally assign a 1 to 8 character PASSWORD for access to a particular file-definition or family of file definitions.

You may further designate:

- 1) If the password is to be required for ANY access to the file.
- 2) If the password is to be required for ONLY update-type functions (Update, Add, Delete).

If a password is assigned to the file, any time a file definition is accessed containing the corresponding label, AUTOMON/BATCH will request the password with the message:

S1924. PASSWORD REQUIRED FOR xxxxxxxxxxxxxxxxxxxx

where xxxxxxxx is the dataset label in question.

If the operator enters the correct password (password is a non-display field), the operation is allowed to continue; and, as long as the operator keeps working with the same file definition, the password will not be requested again.

If an incorrect password is entered, the following message will display:

S1932. SECURITY VIOLATION, TRANSACTION TERMINATED

and the operation will terminate.

The use of passwords on highly sensitive files, in combination with CICS security keys by function, ensures a high degree of dataset security.

WAITING A SECURITY USER EXIT

A Macro-Level or Command-Level user exit program can be developed to communicate with AUTOMON/BATCH for installations that have special security packages.

Most security packages provide a program subroutine that can be called to interrogate the security parameters for a particular resource. If you have this situation, or perhaps you have a user-developed security system, the AUTOMON/BATCH SECURITY USER EXIT will allow you to access your security system in whatever manner required.

A SECURITY USER EXIT can be implemented globally to access all AUTOMON/BATCH files or locally by dataset label. In either case, AUTOMON/BATCH will link to the user exit program prior to each type of access: INQUIRE, MODIFY, ADD, and DELETE. The user-exit program will inform AUTOMON/BATCH whether to allow or prevent the requested function.

The SECURITY USER EXIT is implemented as a CICS program defined in the DFHCSD file. It may be written in any CICS-supported language, Macro-Level or Command Level.

Also, there may be more than one security exit if desired when implemented by dataset-label.

When AUTOMON/BATCH links to the exit program, FILERSERV checks to see if a TWA area is available. The following information is passed in a COMMAREA, unless a TWA is available, in which case the address of the information is passed in the TWA.

Address of a communication area in AUTOMON/BATCH that contains the following data fields:

- Pos 1** The function byte - this 1-byte character code designates the type of function that
AUTOMON/BATCH is about to issue for this file definition. Values are:
- 0 - Inquiry: a definition has been retrieved from the file.
 - 1 - First call of the NEW command (see Note).
 - 2 - Second call of the NEW command, ready to write the definition (see Note).
 - 3 - Update: the operator has made changes to the definition.
 - 4 - Delete: the operator has requested deletion of the definition in memory.
 - 5 - ENABLE command.
 - 6 - DISABLE command.
 - 7 - START command (from Batch Interface).

[Note]: For NEW RECORD creation, the security exit is called twice: once when the NEW command is issued (no record is built yet), and again after the record is built, prior to writing it in the file.

- Pos 2** The operator command code. This is the 1 byte command code entered by the operator to invoke this function: The command codes are:
- I - Inquire
 - C - Change this record (FSRV only)
 - N - New File Definition creation (record add for FSRV)
 - D - File Definition deletion (record deletion for FSRV)
 - F - Browse forward
 - P - Browse backward
 - O - Open the file
 - C - Close the file
 - A - Activate the file
 - B - Backout the file
 - E - Enable a transaction, program or family
 - Q - Disable a transaction, program or family
 - S - Start a user transaction
- Pos 3** The 8-byte dataset, family, transaction or program name.
- Pos 11** The 3-byte operator ID (USERID) from the sign-on table (DFHSNT).
- Pos 14** The 20-byte operator name (NAME) from the sign-on table (DFHSNT).
- Pos 34** The 8-byte password (PASSWORD) from the sign-on table (DFHSNT).
- Pos 42** The 4-byte terminal identification field from the Terminal Control Table.
- Pos 46** A 75-byte message area where the exit program may place an error message to be displayed rather than message F132.
- Pos 121** A 1-byte field used to identify FAMILY commands. This field will contain a 'D' if the 8-byte field at position 3 is a single entity (file, transaction or program) and will contain a 'G' if the 8-byte field at position 3 is a family name (either a file or a transaction/program family, depending upon the command).
- Pos 122** The 44-byte file label. This enables security processing to allow or reject users from modifying the file label in FCTD description displays.
- Pos 166** For VSE, this is the 8-byte APPLID in the FCTDTBL macro for this CICS. For MVS, this is the 8-byte CICSNAM in the FCTDTBL macro.
- Pos 174** 4-byte Transaction ID. This will be 'FCTD' or 'FSRV'.

The user exit program must address these parameter fields by moving them to BLL-CELLS (COBOL) or loading them into registers (Assembler), as demonstrated in the examples below.

The exit program performs any desired logic to determine the authorization of the requested function, such as call a security package subroutine, access external security tables or files, etc.

The function byte (pos. 1 of the passed parameter) is used to communicate back to AUTOMON/BATCH. The exit program must place one of the following values in the function byte before returning to AUTOMON/BATCH. These codes are character values.

- 0 - Access permitted, continue with the requested function.
- 1 - Access denied, issue message F132.
- 2 - Access denied, issue the 75-byte user message that has been placed in the Message Area.

To install the User Exit, do the following:

- 1) Compile the program and link-edit it into the CICS Core-Image or Load library.
- 2) Add an entry to the CICS Processing Program Table (PPT) for the program name.
- 3) If the Exit is to be Global, that is, pertaining to every access to a File Definition by AUTOMON/BATCH, add the program name to the User Option Table by coding the SECGBL operand of the FCTDTBL macro. (See Section 12 - *CUSTOMIZATION*).
- 4) If the Exit is to be Local, pertaining only to certain datasets, add a Security Control record with the program name in the USER EXIT field for each dataset to which it applies. (See *SECURITY CONTROL FILE MAINTENANCE* later in this section).

EXAMPLES OF SECURITY USER EXITS

The following example is a Command-Level Cobol program that consults a table of terminal IDs to authorize or refuse the requested function. Following that is a Macro-Level Assembler Program that does the same thing.

COMMAND-LEVEL COBOL

IDENTIFICATION DIVISION.

PROGRAM-ID. FCTDSECX.

ENVIRONMENT DIVISION.

DATA DIVISION.

WORKING-STORAGE SECTION.

```
77 SUB                                PIC S9(4) COMP.
01 SECURITY-TABLE.
   02 FILLER    PIC X(4) VALUE 'T001'.
   02 FILLER    PIC X(4) VALUE 'T002'.
   02 FILLER    PIC X(4) VALUE 'T003'.
   02 FILLER    PIC X(4) VALUE 'T004'.
01 SEC-TAB REDEFINES SECURITY-TABLE.
   02 SEC-TERM                                PIC X(4) OCCURS 4 TIMES.

01 DFHBL LDS.
   02 FILLER    PIC S9(8) COMP.
   02 TWA-POINTER                                PIC S9(8) COMP.
   02 SEC-WORK-POINTER                            PIC S9(8) COMP.

01 TWA-AREA.
   02 TWA-BLL-CELLS                            PIC S9(8) COMP.

01 SEC-WORKAREA.
   02 FUNCTION-CODE                            PIC X(1).
   02 OPER-COMMAND                            PIC X(1).
   02 DATASET-NAME                            PIC X(8).
   02 OPERATOR-ID                             PIC X(3).
   02 OPERATOR-NAME                           PIC X(20).
   02 PASSWORD                                PIC X(8).
   02 TERMINAL-ID                             PIC X(4).
   02 MESSAGE-AREA                            PIC X(75).
   02 FAMILY-TYPE                             PIC X.
   02 FILE-LABEL                              PIC X(44).
   02 APPLID    PIC X(8).
   02 TRAN-CODE                                PIC X(4).
   02 FILLER    PIC X(5).
```

PROCEDURE DIVISION.

*

**** ADDRESS TRANSACTION WORKAREA

*

EXEC CICS ADDRESS TWA (TWA-POINTER) END-EXEC.

```

*
**** ADDRESS PASSED PARAMETER AREA
*

MOVE TWA-BLL-CELLS TO SEC-WORK-POINTER.
MOVE 0 TO SUB.

FIND-TERM.
ADD 1 TO SUB.
IF SUB IS GREATER THAN 4 GO TO ERR-TERM.
IF SEC-TERM (SUB) NOT = TERMINAL-ID
GO TO FIND-TERM.

*
**** TERMINAL FOUND IN TABLE
*

MOVE 0 TO FUNCTION-CODE.
GO TO RETURN-GOBACK.

*
**** INVALID TERMINAL USER
*

ERR-TERM.
MOVE '2' TO FUNCTION-CODE.
MOVE 'THIS TERMINAL IS NOT AUTHORIZED TO USE CICS-FCTD'
TO MESSAGE-AREA.

RETURN-GOBACK.
EXEC CICS RETURN END-EXEC.

```


MACRO-LEVEL ASSEMBLER

```

        TITLE      'AUTOMON/BATCH SECURITY EXIT'
R0     EQU        0
R1     EQU        1
R2     EQU        2
R3     EQU        3
R4     EQU        4
R5     EQU        5
R6     EQU        6
R7     EQU        7
R8     EQU        8
R9     EQU        9
R10    EQU        10
R11    EQU        11
R12    EQU        12
R13    EQU        13
R14    EQU        14
R15    EQU        15
        SPACE 2
        COPY      DFHTCADSCOPY TCA
        SPACE 2
*
*****  PARAMETER WORKAREA PASSED FROM AUTOMON/BATCH
*
PARMAREA      DSECT
FUNCTION      DS      CL1 FUNCTION CODE
OPERCMD      DS      CL1 OPERATOR COMMAND CODE
DDNAME       DS      CL8 DATASET NAME
OPERID       DS      CL3 OPERATOR ID
OPERNAME     DS      CL20          OPERATOR NAME
PASSWORD     DS      CL8 SIGN-ON PASSWORD
TERMINID    DS      CL4 TERMINAL ID
MSGAREA     DS      CL75          MESSAGE AREA
FAMILYTYP   DS      CL1 "D" OR "G"
FILELBL     DS      CL44          FILE LABEL
APPLIDDS    CL8      APPLID
TRANCODE    DS      CL4 "FCTD" OR "FSRV"
                DS      CL5          FILLER
                SPACE 3
FCTDSECX    CSECT
                USING   *,R11
                LR      R11,R14          LOAD BASE REG
                L       R9,TWACOBA      ADDRESS PASSED PARM
                USING   PARMAREA,R9
                SPACE
GETTERM     EQU      *
                LA     R1,TERMCNT      R1 = TERMINAL TABLECOUNT
                LA     R2,TERMTBL      R2 = TERMINAL TABLE
                SPACE
SEARCH     EQU      *
                CLC    TERMID,0(R2)    LOCATE TERMINAL IN TABLE
                BE     TERMFND        BR IF FOUND
                LA     R2,4(R2)        BUMP TABLE REG
                BCT    R1,SEARCH        CONTINUE SEARCH

```

B
SPACE

NOTFND

BR IF NOT IN TABLE

```

TERMFND      EQU      *           TERMINAL IS AUTHORIZED
             MVI      FUNCTION,C'0' SET AUTHORIZED CODE
             B        RETURN      EXIT
             SPACE
NOTFND       EQU      *           TERMINAL NOT IN TABLE
             MVI      FUNCTION,C'2' SET USER MESSAGE CODE
             MVC      MSGAREA,USERMSG MOVE THE MESSAGE
             SPACE
RETURN       EQU      *
             DFHPC    TYPE=RETURN  RETURN TO AUTOMON/BATCH
             EJECT

*
****        TABLE OF TERMINALS AUTHORIZED FOR AUTOMON/BATCH
*
TERMTBL      EQU      *
             DC       C'T001'
             DC       C'T002'
             DC       C'T003'
             DC       C'T004'
TERMCNT      EQU      (*-TERMTBL)/4
             SPACE
USERMSG      DC       CL75'USER NOT AUTHORIZED FOR AUTOMON/BATCH ADDS'
             SPACE
             LTORG
             END      FCTDSECX

```

SECURITY CONTROL FILE MAINTENANCE

In order to use any of the three methods of security control, you must define security control records for any datasets that need security control. Security maintenance is also used to define global security exits.

Entry to the security maintenance system can be accomplished by pressing PF8 from the FILE SERVICE MANAGER menu. Upon doing so, the following message will display:

ENTER PASSWORD FOR SECURITY MAINTENANCE ACCESS

To access the Security Control File, the system requires a one to eight character password be entered. This is the password that is coded in the SECPSW operand of the customization table. If SECPSW is not coded, the default password will be in use (for more information, see Section 12 - *CUSTOMIZATION*. It is suggested that these passwords be changed as described in that section).

Enter the password in the non-display field provided. The following is the next message to display:

ENTER UP TO 8 BYTES OF DATASET LABEL

If this is the first time security control records have been defined, simply press ENTER.

If this is a subsequent access after previously adding security control records, any entry will cause the file to be positioned to the dataset name in the Security Control File closest to your entry.

After responding, the following screen will be displayed:

```
          C I C S   F I L E S E R V                DATE   : 11/
          D A T A S E T   S E C U R I T Y          TIME   : 16:
          DATASET LABEL (DSNAME)                   PASSWORD SECURITY KEYS  U
          AND OPTION INQ UPD ADD DEL  E

TO DELETE, POSITION CURSOR ON DSNAME
TO UPDATE, MODIFY AND PRESS ENTER ==>
CICS-FILESERV RELEASE 3.0      SECURITY MAINTENANCE
1= (H) ELP      3= (E) XIT      7= (P) REVIOUS      9= (F) NEXT      10= (N) EW      11= (D) ELE
```

At this point, you may ADD, MODIFY or DELETE security control records as described on the following pages.

ADDING SECURITY CONTROL RECORDS

If additional records need to be added, press PF10, or change the command code to 'N' and press ENTER.

The Security Control Maintenance screen will display with the following message on row 20:

S1801. KEY DATA FIELDS AND PRESS ENTER

Each row of fields on the screen represents one security control record. Up to 12 records may be entered from one screen. Following is a description of each data field of the record:

DATASET LABEL (DSNAME)

Enter the full dataset label of the file to be secured; or, enter from one to 43 characters followed by an asterisk (*).

In the latter case, the asterisk indicates a generic dataset label, which means that the security attributes assigned to this entry apply to all files which have the same letters in their label, up to the asterisk.

PASSWORD AND OPTION

If password entry is to be required for access to this file, enter the 1 to 8 character password that will be required for this file. If the password is to be required for Inquiry functions of AUTOMON/BATCH (Inquire, Open, Close, Backout) as well as file modification functions (New, Update, Delete), enter 'A' (for ALL) in the one-byte Option field following the password.

If PASSWORD was omitted or if the password is to apply only to file modification functions, leave the OPTION field blank.

SECURITY KEYS

For each of the four fields, INQUIRE, UPDATE, NEW and DELETE, enter the CICS sign-on security key to be associated with this function for this file. Note that Inquire Security includes the open, close, activate and backout command.

The use of security keys is described in more detail under ASSIGNING CICS SECURITY KEYS, earlier in this section. If any or all of the functions are to be left unsecured, enter zeros for the field or leave it blank.

USER EXIT If a local User Security Exit program is used for FCTD or FSRV access to this file, enter the exit program name here. Otherwise leave the field blank.

Once all desired records have been keyed onto the screen, press ENTER to Add the records.

You may enter any of the commands listed at the bottom of the screen.

MODIFYING SECURITY CONTROL RECORDS

To make changes to security control records, simply tab to the record and key the modifications.

DELETING SECURITY CONTROL RECORDS

To delete a security control record you can key the 'D' command in the command area, position the cursor to the record to be deleted, then press ENTER; or, you can position the cursor to the record to be deleted then press PF11. The screen will re-display with the record highlighted and the following message on row 24:

PRESS ENTER TO DELETE RECORD. ANY OTHER KEY TO IGNORE.

Upon pressing ENTER, the security control record will be deleted.

ADDITIONAL COMMANDS IN SECURITY MAINTENANCE

The remaining commands in security maintenance mode work in a similar fashion to normal AUTOMON/BATCH dataset maintenance. These commands are described below:

<u>Command</u>	<u>PFKey</u>	<u>Description</u>
H	PF1	Display help for the screen.
E	PF3	Return to FILE SERVICE MANAGER menu
P	PF7	Browse backward 12 records at a time
F	PF8	Browse forward 12 records at a time
N	PF10	Add a new record
D	PF11	Delete the security control record to which the cursor is pointing when ENTER (or PF11) is pressed

INSTALLATION OF THE AUTOMON/BATCH PRODUCT

This section describes the procedures for installing AUTOMON/BATCH in your CICS environment.

CICS RELEASE LEVEL REQUIREMENTS

If you are running a release of CICS prior to 3.2.1, the programs on this tape will not run. Please call UNICOM Systems Technical Support to discuss receiving a version that will run on your release.

INSTALLATION STEPS

Installation of AUTOMON/BATCH consists of the following steps:

- 1) Link-edit the installation program, STSINST from the first file of the tape.
- 2) Run the STSINST program to print the full installation instructions and create the installation JCL.
- 3) Tailor the installation JCL for your library names.
- 4) Run the installation JCL to load all programs and files.
- 5) If this is a subsequent installation of AUTOMON/BATCH, you will need to run the conversion utility at this time.
- 6) Define the required CICS table entries. If this is a subsequent installation of AUTOMON/BATCH, for the PCT entries FCTD, FCTT, FSRV and FSPR, you must reduce the TWA size from 1500 to 20.
- 7) If this is a subsequent installation of AUTOMON/BATCH, you must code and assemble a new FCTDTBL macro. The previous FCTDTBL will not operate with this release of AUTOMON/BATCH.
- 8) If this is a subsequent installation and you have been starting the FCTD1010 program in the PLT, the entry must be changed to FCTD1000.
- 9) Add the necessary JCL statements to the CICS start-up JCL.

THE INSTALLATION TAPE

The installation tape consists of two files.

FILE 1:

The first file is the object records for program modules, which are used both to finish the installation process and to provide the temporary expiration password. These programs are:

- 1) STSINST - The installation program.
- 2) STS0100 - The password processor program.
- 3) STSPASS
temporary - The password table. This module is pre-loaded with a password that will expire in approximately 30 days.
- 4) STSCORE - Core display module. Useful for problem resolution.

[Note]: The product control password requires the following:

- 1) STS0100, STSPASS and STSHELP must reside in a CICS LOADLIB.
- 2) DFHCSD program definition for STSPASS, STSCORE, STS0100.
- 3) DFHCSD transaction definition for STSC.

Initialization of any UNICOM software product will first process the password (STSPASS). If an error condition results, the following message will be displayed:

F1019. AUTOMON/BATCH PASSWORD ERROR ==> CODE(x)

The "x" in the above message will be one of the following error codes:

- 1 - The current date is greater than the password expiration date.
- 2 - Module "STSPASS" is not defined to CICS or could not be found in a CICS library.
- 3 - The product ID could not be located in STSPASS.
- 4 - The password in STSPASS is invalid.
- 5 - Module "STS0100" is not defined to CICS or could not be found in a CICS library.

- 6 - STSPASS contains a CPU ID that is invalid for the machine that it is installed on.
- 7 - The password is not valid for the requested product.
- 9 - An undetermined error has been encountered. Please call UNICOM Systems Technical Support at 1-818-838-0606.

FILE 2:

The second file contains the object, documentation members and VSAM file records for seven of the products available from UNICOM Systems. This file contains blocked records with control headers separating the various products. You must use the STSINST program to extract and deblock the product to be installed.

The products present in file 2 are:

- 1) CICS-WINDOWS
- 2) CICS-JUGGLER
- 3) AUTOMON/BATCH
- 4) HELP-WINDOWS
- 5) VTAM-WINDOWS
- 6) VTAM-EXPRESS

Even though you are ready to install the AUTOMON/BATCH product, feel free to install any of the other products on the tape. Each product has an associated documentation member that describes both the installation procedure and a brief overview of the product.

If you choose to install one of the other products and need the complete documentation in order to evaluate it properly, a technical reference guide for that product can be obtained from your sales representative at UNICOM Systems.

DOS/VSE INSTALLATION

For DOS/VSE, you must prepare initial JCL for two steps:

Step 1. Load the installation program and password table

The following JCL can be used to install the first file of the tape for DOS/VSE users:

```
* $$ JOB JNM=STSINST,CLASS=C
* $$ PRT CLASS=A,DISP=D
// JOB LOAD
// ASSGN SYSIPT,XXX           [Note 1]
// MTC REW,SYSIPT
// LIBDEF PHASE,CATALOG=L.S   [Note 2]
// OPTION CATAL
// INCLUDE
// EXEC LNKEDT
// RESET SYSIPT
/*
```

[Notes]:

1. Assign SYSIPT to a tape drive that can read the BPI of your installation tape.
2. This LIBDEF is for DOS/VSE/SP, where L.S = Library, Sub-library

If you are running DOS/VSE, the LIBDEF would be:

```
// LIBDEF CL,TO=xxxxxx
```

where xxxxxx = the core-image library to be used.

The library for the install program, password processor and password table must be a CICS library.

This step will link edit four programs, which are:

STSINST	-The installation processor.
STS0100	-The password processor program.
STSPASS	-The password table containing an initial temporary password.
STSHelp	-The UNICOM internal help module.

Step 2. Print installation instructions and punch installation JCL

The following JCL executes the STSINST program to print the full installation documentation and punch the installation JCL:

```
// ASSGN SYS011,XXX          [Note 1]
// LIBDEF *,SEARCH=L.S      [Note 2]
// EXEC STSINST
PRODUCT=ABATCH
MODE=PRINT
OPSYS=DOS
CICS=X.X                   [Note 3]
LINES=56                   [Note 4]
JCL=INSTALL                [Note 5]
/*
/&
* $$ EOJ
```

[Notes]:

1. Assign SYS011 to a tape drive that can read the BPI of your installation tape.
2. The LIBDEF must identify the library where STSINST was link-edited as a search library. L.S = Library, Sub-library.
3. This is the release level of your CICS system. Valid entries are 321 and 330.
4. The LINES keyword may be omitted if desired, in which case 56 lines per printed page is the default.
5. If this is the first installation of the product, or if you intend to define and load new VSAM files, you should specify JCL=INSTALL. If you are installing a subsequent version of the product using existing VSAM files, you should specify JCL=REINSTALL. If you prefer to key all of the installation JCL yourself, omit the JCL keyword altogether. In this case, only the required JCL will be printed.

At this point, the full installation instructions will be printed. The installation JCL will be punched into the POWER punch queue. Retrieve or view the printed installation documentation and follow the instructions provided there to complete the installation of AUTOMON/BATCH.

MS INSTALLATION

For MVS, you must prepare initial JCL for two steps:

Step 1. Load the installation program and password table

The following JCL can be used to install the first file of the tape for MVS users:

```
//LOADSTS          JOB1,'ACCOUNT-ID',MSGCLASS=X
//STEP1           EXEC      PGM=IEWL,PARM='LIST,LET,XREF'
//SYSPRINT        DD SYSOUT=*
//SYSLIB          DD DSN=CICS.STEPLIB,          [Note 1]
//                DISP=SHR
//SYSLIN          DD DSN=MASTER.TAPE,
//                UNIT=TAPE,
//                VOL=(,RETAIN,,,SER=INPUT),
//                LABEL=(1,NL),
//                DCB=(RECFM=FB,LRECL=80,
//                BLKSIZE=80),
//                DISP=(SHR,PASS)
//SYSUT1          DD UNIT=SYSDA,
//                SPACE=(1024,(20,20))
//SYSLMOD         DD DSN=CICS.STEPLIB,          [Note 1]
//                DISP=SHR
```

[Notes]:

1. The library for the install program, password processor and password table must be a CICS load library.

This step link edits three programs. They are:

STSINST - The installation processor.
STS0100 - The password processor program.
STSPASS - The password table containing an initial temporary password.
STSHelp - The UNICOM internal help module.

Step 2. Print installation instructions and create installation JCL

The following JCL executes the STSINST program to print the full installation documentation and create the installation JCL:

```
//STEP2          EXEC      PGM=STSINST
//SYSPRINT      DD SYSOUT=*
//STEPLIB       DD DSN=CICS.STEPLIB,   [Note 1]
//              DISP=SHR
//MPRDIN        DD DSN=MASTER.TAPE,
//              UNIT=TAPE,
//              VOL=SER=INPUT,
//              LABEL=(2,NL),
//              DCB=BLKSIZE=32000,
//              DISP=(SHR,PASS)
//SYSPCH        DD DSN=????.????.(STSJCL), [Note 2]
//              DCB=BLKSIZE=???,DISP=SHR
//SYSIN         DD *
PRODUCT=ABATCH
MODE=PRINT
OPSYS=MVS
CICS=X.X       [Note 3]
LINES=56      [Note 4]
JCL=INSTALL   [Note 5]
//
```

[Notes]:

1. The STEPLIB must identify the library where STSINST was link-edited in step 1.
2. This is any PDS where you want to load the installation JCL. You must specify a member name for your JCL and you must specify the blocksize of the PDS on the DCB parameter.
3. This is the release level of your CICS system. Valid entries are 3.2.1 or 3.3.0.
4. The LINES keyword may be omitted if desired, in which case 56 lines per printed page is the default.
5. If this is the first installation of the product, or if you intend to define and load new VSAM files, you should specify JCL=INSTALL. If you are installing a subsequent version of the product using existing VSAM files, you should specify JCL=REINSTALL. If you prefer to key all of the installation JCL yourself, omit the JCL keyword altogether. In this case, only the required JCL will be printed.

At this point, the full installation instructions will be printed. The installation JCL will be loaded into the member name of the PDS specified by SYSPCH. Retrieve or view the printed installation documentation and follow the instructions provided there to complete the installation of AUTOMON/BATCH.

ADDITIONAL INSTALLATION NOTES, MS AND DOS/ VSE

- 1) Be sure to specify ASSEMBLER as the language.

- 2) If the Batch Interface is to be used, you must not code LOG=YES on the FCT entry. An invalid request condition will occur if it is present.

- 3) Do not put transaction security on the entry for FSPR. This tranocode is used for the on-line print function and if security is present, it will require the on-line printer to be signed-on in order to print a record.

- 4) If you plan to use suffixed option tables, the FCTDBCHT program must be link-edited into an authorized library with AC=1 as a link-edit PARM statement.

THE CONVERSION UTILITY PROGRAM

An additional program, FCTDUTIL, is provided for conversion of the AUTOMON/BATCH Control File from one release of AUTOMON/BATCH to another.

For MVS the JCL required to run the program is as follows:

```
//CONVERT                JOB1,'ACCOUNT',MSGCLASS=X
//STEP1                  EXEC      PGM=FCTDUTIL
//STEPLIB                DD DSN=CICS.SDFHLOAD,
//                        DISP=SHR      [Note 1]
//FCT$FIL                DD DSN=OLD.CICS.ABATCH.FILE,
//                        DISP=SHR      [Note 2]
//PCHFIL                DD DSN=YOUR.TSO.LIB(YOURFIL),
//                        DISP=SHR      [Note 3]
//RDRFIL                DD *
//PRTFIL                DD SYSOUT=*
RDO,GROUP=XXXXXXXX      [Note 4]
//
```

[Notes]:

1. This is the loadlib containing the FCTDUTIL program.
2. This is the AUTOMON/BATCH control file input to this process.
3. This is the TSO library and member name that you want the punched output to be stored in. The member can then be edited and customized to your installation's needs.
4. This is the DFHCSD group name that will be assigned to each file definition. If group names other than this one are desired, the output from this process will need to be edited and the correct group name entered for those files targeted for groups other than the default.

The output from this process is included in a DFHCSDUP batch run to add the file definitions to the DFHCSD file.

THE CICS STARTUP JCL

Add the following DLBL (DOS users) or DD (MVS users) statement to the CICS Start-up JCL or add it to the Partition Standard or Standard Labels:

```
// DLBL FCT$FIL,'CICS.ABATCH.CONTROL.FILE',,VSAM
//FCT$FIL                DD DSN=CICS.ABATCH.CONTROL.FILE,
//                        DISP=SHR
```

If you are not currently using a Start-up System Initialization Program List Table, add the following parameter(s) to the System Initialization Table overrides, or add them to the System Initialization Table (DFHSIT):

```
PLTPI=xx
```

Where xx is the suffix code used in the system initialization DFHPLT TYPE=INITIAL macro.

SUMMARY

At this point, the basic AUTOMON/BATCH product is installed. It can be activated by entering "FCTD" and pressing enter or (if you are starting it in the PLTPI) it will automatically activate on the next CICS startup.

You may want to tailor the product to fit your environmental needs, as described in the section on *CUSTOMIZATION*, however the product is operational without doing so.

KEYWORDS OF THE UNCOMINSTALLATION PROGRAM

The installation program, STSINST, which is provided on the first file of the installation tape, can be used to print the documentation or perform a complete install for any of the products on the master installation tape.

The individual installation instructions of each product provide the necessary JCL and keyword parameters that are specific to that product.

This topic describes all of the available keywords of the STSINST program, plus the coding format used.

FORMAT OF THE STSINST KEYWORDS

All keywords and operands of the STSINST program are coded in the SYSIN (MVS) or SYSIPT (DOS) dataset.

Each keyword and operand occupies one statement. You cannot concatenate multiple keywords on one line.

Each keyword must begin in position one.

Following are all of the possible keywords and operands of the STSINST program, listed in alphabetical order. A detailed explanation of each keyword and operand follows.

Underlined values listed are the default values if the associated operand is omitted. Values separated by a bold vertical bar (|) indicate mutually exclusive values that may be coded.

KEYWORD DESCRIPTIONS

DIRECTORY

The DIRECTORY keyword requires no operands. If present, it will causes STSINST to pass the master installation tape and print a directory of its contents.

CICS=3.2.1,3.3.0

The CICS keyword defines the release level of CICS where the specified product is to operate. Valid entries are 3.2.1 and 3.3.0 for MVS.

FSRVFILE=xxxxxx

The FSRVFILE keyword identifies the name of the DLBL or DD statement in the job control statements for this execution of STSINST that names the VSAM file for AUTOMON/BATCH. The xxxxxx is the 1-7 byte file ID of the DLBL statement (DOS) or 1-8 byte DDNAME of the DD statement (MVS).

HELPSCRN=xxxxxx

The HELPSCRN keyword identifies the name of the DLBL or DD statement in the job control statements for this execution of STSINST that names the VSAM screen file for AUTOMON/BATCH. The xxxxxx is the 1-7 byte file ID of the DLBL statement (DOS) or 1-8 byte DDNAME of the DD statement (MVS).

HELPTEXT=xxxxxx

The HELPTEXT keyword identifies the name of the DLBL or DD statement in the job control statements for this execution of STSINST that names the VSAM text file for AUTOMON/BATCH. The xxxxxx is the 1-7 byte file ID of the DLBL statement (DOS) or 1-8 byte DDNAME of the DD statement (MVS).

JCL=INSTALL|REINSTALL

The JCL keyword defines the action to be taken for the installation job control statements. Valid operands are:

INSTALL Create JCL that will delete and redefine the VSAM file(s).

REINSTALL Create JCL that will neither delete nor redefine the VSAM file(s).

If the JCL keyword is omitted, the installation JCL will be printed only. If either INSTALL or REINSTALL is specified, the JCL will be punched to the POWER punch queue (DOS) or written to the PDS member identified by the SYSPCH DD statement (MVS).

JUGLFILE=xxxxxx

The JUGLFILE keyword identifies the name of the DLBL or DD statement in the job control statements for this execution of STSINST that names the VSAM file for CICS-JUGGLER or CICS-JUGGLER/SVT. The xxxxxx is the 1-7 byte file ID of the DLBL statement (DOS) or 1-8 byte DDNAME of the DD statement (MVS).

LINES=56|nn

The LINES keyword defines the number of lines to print on the system printer before skipping to a new page. If omitted, 56 is the default.

MODE=PRINT|CREATE|INSTALL|REINSTALL|DOCUMENT

The MODE keyword defines the action to be taken by STSINST. Valid mode operands are:

PRINT	Print the installation instructions for the specified product.
CREATE	Create an installation file on the work tape for the specified product.
INSTALL	Load the VSAM files associated with the specified product. MODE=INSTALL indicates that this is a new installation, loading to an empty VSAM file.
REINSTALL	Load the VSAM files associated with the specified product. MODE=REINSTALL indicates that this is a subsequent installation, reloading the system records to an existing VSAM file that contains user data.
DOCUMENT	Print the overview documentation for the specified product. The overview document is a short preview of the product, telling you how to get started using it. If you need the complete reference guide, that can be obtained from your sales representative.

OPSYS=DOS|MVS|MVS/XA|MVS/SP

The OPSYS keyword defines the operating system where the specified product is to reside. For all products except VTAM-EXPRESS, you need only to specify DOS or MVS. For the MVS version of EXPRESS you must specify either MVS/XA or MVS/SP. FOR MVS/ESA, specify MVS/XA.

PRODUCT=WINDOWS|JUGGLER|SVT|HELP|ABATCH|VWINDOWS|EXPRESS

The PRODUCT keyword identifies the product to be installed, or for which the documentation is to be printed. A brief description of each of the products is contained in the appendix of this document.

The full product name corresponding to each keyword is:

WINDOWS	-	CICS-WINDOWS
JUGGLER	-	CICS-JUGGLER
SVT	-	CICS-JUGGLER/SVT
HELP	-	HELP-WINDOWS
ABATCH	-	AUTOMON/BATCH
VWINDOWS	-	VTAM-WINDOWS
EXPRESS	-	VTAM-EXPRESS

WNDOWFILE=xxxxxx

The WNDOWFILE keyword identifies the name of the DLBL or DD statement in the job control statements for this execution of STSINST that names the VSAM file for CICS-WINDOWS. The xxxxxx is the 1-7 byte file ID of the DLBL statement (DOS) or 1-8 byte DDNAME of the DD statement (MVS).

VTWOWFILE=xxxxxx

The VTWOWFILE keyword identifies the name of the DLBL or DD statement in the job control statements for this execution of STSINST that names the VSAM file for VTAM-WINDOWS. The xxxxxx is the 1-7 byte file ID of the DLBL statement (DOS) or 1-8 byte DDNAME of the DD statement (MVS).

INSTALLING THE PRODUCT CONTROL PASSWORD

The product control password is a special code that controls the authorized use of any product from UNICOM Systems. During the trial evaluation period, the product control password defines the date at which the trial version of the product will expire and can no longer be used.

For licensed users, the product control password defines all the CPUs where the product may be used. Passwords for lease or lease-purchase licenses contain both an expiration date and CPU ID.

When you initially receive an installation tape, whether for a trial evaluation or not, the tape contains a temporary password that will keep the product from expiring for approximately 30 days. If you are a licensed user, part of the re-install procedure is to assemble your permanent password table after installing the new tape.

If you are evaluating the product and have received a new temporary password from your sales representative, you must assemble the password table with the new password to keep the product from expiring.

The following job control is used to assemble and catalog the password control table, STSPASS.

DOS/VSE USERS

```
// JOB PASSWORD ASSEMBLY
// LIBDEF PHASE,CATALOG=L.S      [Note 1]
// OPTION CATAL
  PHASE STSPASS,*              [Note 2]
// EXEC ASSEMBLY

                                DC CL6'000000'   CPU ID
                                DC CL4'xxxx'     Product ID
                                DC CL6'mmddy'    Expiration Date
                                DC CL16'xxxxxxxx' Password
                                .
                                .
                                DC CL6'000000'   CPU ID
                                DC CL4'xxxx'     Product ID
                                DC CL6'mmddy'    Expiration Date
                                DC CL16'xxxxxxxx' Password
                                END

/*
// EXEC LNKEDT
/ &
```

MVS USERS

```
//PASSTBL          JOB1,'ACCOUNT DATA',MSGCLASS=X
//ASM              EXEC      PGM=IEV90,PARM='DECK'
//SYSPRINT         DD SYSOUT=*
//SYSLIN          DD DUMMY
//SYSUT1          DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPUNCH        DD UNIT=SYSDA,DISP=(,PASS),
//                SPACE=(CYL,(1,1)),
//                DSN=&&TEMP1,
//                DCB=BLKSIZE=3200
//SYSIN           DD *
                  DC CL6'000000'   CPU ID
                  DC CL4'xxxx'     Product ID
                  DC CL6'mmddy'    Expiration Date
                  DC CL16'xxxxxxxxxx' Password
                  .
                  .
                  DC CL6'000000'   CPU ID
                  DC CL4'xxxx'     Product ID
                  DC CL6'mmddy'    Expiration Date
                  DC CL16'xxxxxxxxxx' Password
                  END
/*
//LINK            EXEC      PGM=IEWL,PARM=(LIST,XREF),
//                COND=(4,LT)
//SYSLIB          DD DUMMY
//SYSPRINT         DD SYSOUT=*
//SYSLIN          DD DSN=&&TEMP1,DISP=(OLD,DELETE)
//SYSLMOD         DD DSN=CICS.STEPLIB(STSPASS), [Note 1]
//                DISP=SHR        [Note 2]
//SYSUT1          DD UNIT=SYSDA,SPACE=(CYL,(1,1))
```

[Notes]:

1. The library containing the link-edited module must be a CICS load library.
2. The link-edited module name must be STSPASS

DATA FIELDS OF THE PASSWORD CONTROL TABLE

The password table is a series of define constant instructions. The same password table is used for all products from UNICOM Systems.

There are four DC statements required for each product, or for each occurrence of the same product on multiple CPUs. There is no limit to the number of product entries that may be present. The four DC instructions define the following values:

- 1) The first entry in the table is a 6-position CPU ID. For temporary passwords, code this as all zeros. For licensed users, code the full 6-position CPU ID where the product will operate.

If you are a VM user and have multiple guest machines on the same CPU, you can get by with a single entry if the last five positions of the CPU ID on each guest machine is the same. If this is the case, code zero as the first digit, followed by the five identifying digits. If each guest CPU ID is different, you must make an entry for each unique CPU ID.

- 2) The second entry is a 4-position product ID. The valid product IDs are:

WNDO	- CICS-WINDOWS
JUGL	- CICS-JUGGLER
SVT	- CICS-JUGGLER/SVT
FSRV	- AUTOMON/BATCH
HELP	- HELP-WINDOWS
VTWO	- VTAM-WINDOWS
EXPR	- VTAM-EXPRESS

- 3) The third entry is the expiration date. For temporary passwords or for lease and lease-purchase licenses, you will be given a date when the product will expire along with your password. That date must be coded here in MMDDYY format. For perpetual licenses, code this as all zeros.
- 4) The last entry is the password itself. You must define it as a 16-byte field, although the total number of characters in the password may not reach 16. Code the password that you have been given which corresponds to the CPU ID, product and expiration date of this entry.

CUSTOMIZATION

Certain default options in AUTOMON/BATCH may be modified by the System Programmer, if desired, and features of the product may be enabled or disabled.

Customization is done by coding the applicable operands of the FCTDTBL macro.

The options that can be tailored by the FCTDTBL macro are as follows:

- 1) Activate support for the Auto-backout feature (AUTBOUT).

- 2) Specify a 'wake-up' time interval for the Auto-Backout program (AUTBINT).

- 3) Activate support for the Batch Interface (BCHINTF).

- 4) Specify whether messages are to be logged to the system console for functions performed on-line to satisfy batch processing requests (BINTMSG).

- 5) Specify the name of a list that contains all RDO groups which have file definitions for the CICS which will be running this AUTOMON/BATCH.

- 6) Define the 'wake-up' time interval for the Batch interface (BCHINTV).

- 7) Define the CICS APPLID to be used by the Batch Interface.

- 8) Suppress or enable Security checking (SECURTY).

- 9) Define a Global Security Exit (SECGBL).

- 10) Change the default password for Security Control Maintenance (SECPSW).

- 11) Adjust the size of the Generic Label Security Table (SECGEN).

12) Specify a different control file name to be used, rather than FCT\$FIL.

13) Specify whether a message is to be displayed on the system console every time a file is manually opened using an FCTD command (OPENMSG).

14) Specify whether a message is to be displayed on the system console every time a file is manually closed using an FCTD command (CLOSMSG).

15) Specify whether operators may use the CEMT inquire function (PF12) from the File Definition Display (CEMTINQ).

16) Specify the maximum number of files that will be defined in the CICS region (unless your CICS is very static and no new files will defined using RDO, it is a good practice to add 10 to 20% expansion).

CODING THE USER OPTION TABLE

The format of the User Option Table macro, FCTDTBL, is as follows:

FCTDTBL	X
APPLID=[<u>CICS</u> FCTD symbol],	X
AUTBOUT=[NO <u>YES</u>],	X
AUTBINT=[<u>0030</u> mmss],	X
BCHINTF=[NO <u>YES</u>],	X
BCHINTV=[<u>0020</u> mmss],	X
BCHPLT=[NO <u>YES</u>],	X
BINTMSG=[NO <u>YES</u>],	X
CEMTINQ=[NO <u>YES</u>],	X
CLOSMSG=[NO <u>YES</u>],	X
FCT=[<u>FCT\$FIL</u> symbol],	X
FILELST=[list name]	
OPENMSG=[NO <u>YES</u>],	X
RELEASE=[<u>30</u>],	X
SECURTY=[<u>NO</u> YES],	X
SECGEN=[<u>29</u> number],	X
SECGBL=[symbol],	X
SECPSW=[<u>UNICOM</u> symbol],	X
SUFFIX=[number],	X
TOTFILS=[<u>300</u> nnnn]	
VARY=[<u>NOWAIT</u> WAIT FORCE]	

One of the source members loaded from file 3 during installation is called 'FCOPT'. This member is a skeleton user option table with all keywords present. If desired, you may copy this member into your editor and use it as a basis for your version of the user option table.

The operands of the FCTDTBL macro may be coded as follows:

APPLID This operand only applies to DOS users only, if BCHINTF is coded YES. This is a one to eight position name used by program FCTDBCHT to identify the CICS system where AUTOMON/BATCH is running. The value coded for APPLID corresponds to the name used in the CICS= operand of a AUTOMON/BATCH command submitted in a batch partition or region.

Any alphanumeric value may be coded for APPLID, but a recommended value, for consistency's sake, is the value coded for APPLID in the CICS System Initialization Table (SIT), or the SYSIDNT value from the Terminal Control Table (TCT).

If BCHINTF=NO or is omitted, the APPLID is ignored. If BCHINTF=YES and APPLID is omitted, the default APPLID is 'CICSFCTD'.

AUTBOUT This keyword is used to activate or suppress the automatic backout feature of AUTOMON/BATCH. The auto-backout feature can be used to automatically backout specified files after a designated period of inactivity.

Code YES (or omit this operand) if you want the auto-backout feature to be activated, and code the AUTBINT keyword to specify the desired wake-up interval. Code 'NO' if you do not want the auto-backout feature to be active.

For more information on the auto-backout feature, see *Section 5 - Automatic Oprn/Close*.

- AUTBINT** Code this keyword if AUTBOUT is coded YES. This is the time interval (in MMSS format) to be used by the auto-backout program. This is the length of time that the on-line FCTT program will wait before 'waking up' to check for any files to be deactivated. If omitted, the default is 3000, which is 30 minutes. This operand does not apply if AUTBOUT is coded 'NO'. See *Section 5 - Automatic Open/Close* for details of this option.
- AUTOMSG** Code YES for this operand if you want a message to be displayed on the system console any time an automatic open is performed for a file. The message displayed will be number 1019. If omitted, NO is the default.
- BCHINTF** Code YES for this operand if you want to use the BATCH INTERFACE to AUTOMON/BATCH. The Batch Interface allows CICS file commands (Open, Close, Backout, etc.) to be issued from a jobstream running in a Batch partition or region. See *Section 6 - THE BATCH INTERFACE TO AUTOMON/BATCH* for details of this option.

If Batch Interface support is not desired, code NO or omit this operand.

- BCHINTV** This is the time interval (in MMSS format) to be used by the 'timer' version of the Batch Interface. This is the length of time that the on-line AUTOMON/BATCH program will wait before 'waking up' to check for batch activity. If omitted, the default is '0020'. This operand does not apply if BCHTIMR is coded 'NO'. See *Section 6 - THE BATCH INTERFACE TO AUTOMON/BATCH* for details of this option.
- BCHPLT** Code NO if you do not want the batch interface to be activated by AUTOMON/BATCH in the PLTPI initialization.
- BINTMSG** Code NO if you do not want messages to be logged to the system console for functions performed on-line to satisfy batch processing requests. Code 'YES', or omit this operand, if you do want the messages to be logged to the system console.
- CEMTINQ** Code NO if you do not want the CEMT inquire function (PF12) to be allowed from the File Definition Display. If omitted, the default is YES.

APPLID (MVS users only)

This operand only applies to MVS users only, if BCHINTF is coded YES. This is a one to eight position name used by program FCTDBCHT to identify the CICS system where AUTOMON/BATCH is running. The value coded for APPLID corresponds to the name used in the CICS= operand of a AUTOMON/BATCH command submitted in a batch partition or region.

The value coded for APPLID corresponds to the name used in the MVS JOB card used to initiate this CICS.

If BCHINTF=NO or is omitted, the APPLID is ignored. If BCHINTF=YES and APPLID is omitted, the default name is 'CICSFCTD'.

- CLOSMG** Code NO for this operand if you do not want a message displayed on the system console every time a file is manually closed using an FCTD command. If omitted, the default is 'YES'.
- FCT** The FCT keyword can be used to specify a different file name to be used, rather than FCT\$FIL, for the AUTOMON/BATCH VSAM file. If coded, there must be an entry in the DFHCSD file with the same name.

If FCT\$FIL is to be used as the file name, omit this keyword.

- FILELST** Enter the name of a DFHCSD list that contains all groups that reference files. The FILELST is used at initialization time to process the DFHCSD file and table all files with their corresponding

group names. If FILELST is omitted, AUTOMON/BATCH must process all groups in the DFHCSD file, regardless of whether they contain files or not; creating a DFHCSD list of all groups that reference files and specifying the list to AUTOMON/BATCH allows AUTOMON/BATCH to more efficiently acquire the names of all files.

OPENMSG	Code NO for this operand if you do not want a message to be displayed on the system console every time a file is manually opened using an FCTD command. If omitted, the default is 'YES'.
RELEASE	Code this with the release of AUTOMON/BATCH that will be utilizing this option table. This is used as a check point to ensure that AUTOMON/BATCH does not attempt to use an outdated option table. If omitted, the default is 30.
SECGEN	Code a number that represents the maximum number of generic dataset label security records that have been created with the Security Control System. Generic dataset label security records are those security records where the label ends with an asterisk(*). If omitted, 29 is the default. If generic dataset label security records are not in use, or if SECURITY is coded NO, omit this operand.
SECGBL	Code the 1 to 8-character name of the Global Security User exit program, if one is in use. The Global Security Exit program will be invoked for every access to a File Definition with the FCTD transaction. See <i>Section 10 - SECURITY CONTROL SYSTEM</i> for instructions for writing a Security Exit program. If a Global Security Exit program is not desired, omit this operand.
SECPSW	Code the 1 to 8-character password that is to be required for access to the Security Control System. This is the password that is needed in response to message S120. If omitted, the default is 'UNICOM'. If the default password is acceptable or if SECURITY=NO, omit this operand.
SECURITY	Code YES, if AUTOMON/BATCH is to perform security checking using security control records created with the Security Control Maintenance system (see <i>Section 10 - SECURITY CONTROL SYSTEM</i>), or if a Global Security Exit is in use. Code NO or omit to suppress security checking. This will result in a slightly improved response time for the FCTD transaction, particularly on the Multiple File Display.
SUFFIX	Code the two digit suffix name that is to be used for this table. If you intend to have multiple FCTDTBL macros, you may name the table FCTDTBxx, where xx is a suffix. In turn, if AUTOMON/BATCH encounters a problem with the table, AUTOMON/BATCH will generate a message describing the problem included with the correct table name. If you do not want to use table suffixing, omit this operand.
TOTFILS	Specify the maximum number of files that this CICS will have defined to it (normally it is a good procedure to allow for a 10 to 20% expansion). 300 is the default value.
VARY	Code NOWAIT or omit, if a VARY command is issued from the Batch Interface and you want the vary command to take effect immediately. Code WAIT if AUTOMON/BATCH is to defer the vary process until the file is available. Code FORCE if the vary process is to override all exceptions or waits.

DEFAULT VALUES FOR THE USER OPTION TABLE

If the User Option Table is not installed, or if any of the operands are omitted, the following default values will apply:

FCTDTBL		X
APPLID=CICSFCTD,	X	
AUTBOUT=YES,	X	
AUTBINT=0030,	X	
BCHINTF=YES,	X	
BCHINTV=0020,	X	
BCHPLT=YES,	X	
BINTMSG=YES,	X	
CEMTINQ=YES,	X	
CICSNAM=CICSFCTD,	X	
CLOSMSG=YES,	X	
FCT=FCT\$FIL,	X	
OPENMSG=YES,	X	
RELEASE=30,	X	
SECGEN=29,	X	
SECPSW=UNICOM,	X	
SECURTY=NO,	X	
TOTFILS=300,	X	
VARY=NOWAIT	X	

INSTALLING THE USER OPTION TABLE

The User Option Table is installed by assembling and link-editing it into the CICS Core-Image or Load library.

The following example is a DOS assemble and link-edit of the User Option Table:

```
//          JOB FCTDTBL ASSEMBLY
//          LIBDEF CL,TO=XXXXXXXXX          Core-Image library
//          OPTION CATAL
//          PHASE FCTDTBL,*[Note 1]
//          EXEC ASSEMBLY
//          FCTDTBL X
//          SECURITY=YES,          X
//          SECGEN=50,          X
//          SECPSW=SYSTEM,          X
//          TOTFILS=250
//          END
/*
//          EXEC LNKEDT
/ &
```


Following is an MVS example for installing the User Option Table:

```

//FCTDTBL          JOB1.'ACCOUNT-DATA',MSGCLASS=X
//ASM              EXEC      PGM=IEV90,PARM='DECK'
//SYSLIB           DD DSN=CICS.MACLIB,
//                  DISP=SHR                                [Note 2]
//                  DD DSN=SYS1.MACLIB,
//                  DISP=SHR                                [Note 3]
//SYSPRINT         DD SYSOUT=*
//SYSLIN           DD DUMMY
//SYSUT1           DD UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPUNCH        DD UNIT=SYSDA,DISP=(,PASS),
//                  SPACE=(CYL,(1,1)),
//                  DSN=&&TEMP1,
//                  DCB=BLKSIZE=3200
//SYSIN            DD *
                  FCTDTBL                                  X
                  SECURTY=YES,                             X
                  SECGEN=50,                                X
                  SECPSW=SYSTEM,                            X
                  TOTFILS=150,
                  END
/*
//LINK             EXEC      PGM=IEWL,PARM=(LIST,XREF),
//                  COND=(4,LT)
//SYSLIB           DD DUMMY
//SYSPRINT         DD SYSOUT=*
//SYSLIN           DD DSN=&&TEMP1,DISP=(OLD,DELETE)
//SYSLMOD          DD DSN=CICS.LOADLIB(FCTDTBL),           [Note 1]
//                  DISP=SHR                                [Note 4]
//SYSUT1           DD UNIT=SYSDA,SPACE=(CYL,(1,1))

```

[Notes]:

1. If a suffixed table is to be used, the name should be

FCTDTBxx

where xx is the desired suffix. (See *SUFFIXING THE USER OPTION TABLE*, in the section entitled *SPECIAL CONSIDERATIONS*)

2. Macro library for CICS (where FCTDTBL macro was loaded).
3. MVS System macro library.
4. Load library for CICS applications.

THE USER OPTION DISPLAY SCREEN

To display the USER OPTIONS DISPLAY screen, press PF7 from the FILE SERVICE MANAGER menu.

The USER OPTIONS DISPLAY screen will appear generally as follows:

```

          C I C S - F I L E S E R V          DATE   : 11/19/90
          U S E R   O P T I O N   T A B L E    TIME   : 10:08
                                               TERMD  : VOO1
-----
TABLE NAME ==>      FCTDTBCT UTILIZATION ==>      10 %
-----
APPLID      TESTC17      BINTMSG      YES      RELEASE      YES
AUTBINT     0030         CEMTINQ     NO       SECGBL       N/A
AUTBOUT     NO          CLOSMG     YES       SECGEN       35
AUTOFCT     NO          FCT        FCT$FIL   SECURITY     NO
AUTOMSG     YES        FCTSYN     NO       SUFFIX       17
BCHINTF     YES        LABELS     YES      TOTFILS     20
BCHINTV     0030        LABPAR     NO       VARY        NOWAIT
BCHPLT     YES        LABSYS     NO
BCHTIMR     YES        OPENMSG    YES
-----
F1=HELP    F3=RETURN
-----
          (C) SOFTOUCH SYSTEMS, INC. 1986
    
```

THE USER OPTIONS DISPLAY SCREEN FIELD DESCRIPTIONS

The USER OPTIONS TABLE screen is for informational purposes only. Any customization changes will need to be made by recoding and recompiling the customization table (explained earlier in this section).

The following fields are displayed:

DATE	This is the current date in MM/DD/YY format.
TIME	This is the current time in HH/MM format.
TERMD	This is the terminal ID of the terminal in use.
TABLE NAME	This is the name of the customization table that was used to set the options that are currently in effect for this transaction.
APPLID	This is the CICS application ID where AUTOMON/BATCH may be accessed by the Batch Interface. This only applies if BCHINTF=YES.
AUTBINT	This is the time interval (in MMSS format) that the on-line FCTT program will wait before 'waking-up' to check for any files to be deactivated. This only applies if AUTBOUT=YES.

AUTBOUT This specifies whether the auto backout feature of AUTOMON/BATCH is active or not. If AUTBOUT=YES, the auto backout feature can be used to automatically backout specified files after a designated period of inactivity.

BCHINTF	This specifies whether the Batch Interface is available or not.
BCHINTV	This is the time interval (in MMSS format) to be used by the 'timer' version of the Batch Interface. This only applies if BCHINTF is coded as YES.
BINTMSG	This specifies whether messages are logged to the console for functions performed on-line that satisfy batch processing requests.
CEMTINQ	This specifies whether CEMT inquire function (PF12) from the File Definition Display is active or not.
CLOSMG	This specifies whether or not a message will be displayed on the system console any time a manual close is performed on a file.
FCT	This specifies the file to be used for the AUTOMON/BATCH VSAM file.
OPENMSG	This specifies whether or not a message will be displayed on the system console any time a manual open is performed on a file.
RELEASE	This is the release of AUTOMON/BATCH.
SECGBL	This is the 1 to 8 character name of the Global Security User Exit. If no User Exit is present this field will contain 'N/A'.
SECGEN	This number represents the maximum number of generic dataset label security records that have been created with the Security Control System.
SECURTY	This specifies whether or not AUTOMON/BATCH will perform security checking.
SUFFIX	This is the suffix of the FCTDTBL macro that is in use for this CICS.
TOTFILS	This is the maximum number of files in the CICS region.
VARY	This specifies whether the WAIT, NOWAIT or FORCE option is in affect for vary commands issued form the batch interface.

SPECIAL CONSIDERATIONS

This section deals with certain functions of AUTOMON/BATCH operation of which the system programmer should be aware, as well as the limitations of the current version of the package.

PARTITION STANDARD LABEL AREA

For **DOS/VSE** users only, AUTOMON/BATCH uses the Partition Standard (PARSTD) label area to create the labels required to activate and open files. If you are currently using the PARSTD area for other (non-AUTOMON/BATCH) labels, you may experience a longer response time than necessary to open a AUTOMON/BATCH file, due to excessive I/O in the label processing. If the PARSTD area is quite full it can even cause AICA transaction abends.

It may be desirable, therefore, to move the labels currently in the PARSTD area to either the Temporary label area (include them in the CICS start-up JCL) or the Standard Label area.

It is also a good idea to clear the PARSTD area prior to starting CICS. This can be accomplished by inserting the following two statements at the beginning of the start-up JCL:

```
// OPTION PARSTD
/*
```

If the PARSTD area fills up during CICS operation with AUTOMON/BATCH active (if message 1506 is displayed), you can alleviate the problem by coding DLBL statements for some or all of the frequently-used files and including them in the CICS start-up JCL or storing them in the Standard Label area. This will cause AUTOMON/BATCH to use the existing label rather than add another label to the PARSTD area.

PRODUCT LIMITATIONS

Certain types of file definitions are not currently supported by AUTOMON/BATCH and must be coded in the FCT. These are:

- DL/I definitions.
- RECFORM=UNDEF for BDAM files.
- Keyed BDAM files (SERVREQ=KEY).
- INDIRECT ACCESS files definitions.
- SEGMENTED file definitions.
- ICIP files (OS only).

NEWCOPY COMMANDS FOR PROGRAM MODULES

The FCTDTBL module is permanently resident and kept perpetually 'in-use' in order to prevent a new copy of the module from being loaded after the initialization of AUTOMON/BATCH has been completed. Special-purpose commands are provided to 'unhook' AUTOMON/BATCH in order to allow a NEWCOPY command to be executed. These commands are discussed in detail in *Section 14 - SPECIAL PURPOSE COMMANDS*.

FCTD1000 and FSRV1000 can be new copied at any time. FCTDTBL can only be new copied after doing an FCTD Z UNHOOK command. If the Batch Interface is active, issue the 'Z STOPBCH' command prior to the UNHOOK command.

SUFFIXING THE USER OPTION TABLE

It may be desirable to have multiple versions of the user option table in the same CICS library, with different versions accessed by different CICS systems. This can be done by adding a suffix code as the last two characters of the table name. The procedure for using suffixed tables is as follows:

- 1) Link-edit the user option table under the name FCTDTBxx, where xx is any two alpha-numeric characters.
- 2) Add a PPT entry for the table using the suffixed name.
- 3) Run the following job step prior to bringing up the CICS system that is to use the suffixed table. Be sure to point the DLBL or DD statement to the correct AUTOMON/BATCH control file for this CICS system. (Note that the xx in the PARM statement is the 2-character suffix code in the table name).

DOS/VSE:

```
// JOB TBL$SUF$X
// DLBL FCT$FIL,'AUTOMON/BATCH.CONTROL.FILE',,VSAM
// EXEC FCTDBC$HT,SIZE=AUTO,PARM='FTBL=xx'
```

MVS:

```
//SUF$X EXEC PGM=FCTDBC$HT,
// PARM='FTBL=xx'
//FCT$FIL DD DSN=AUTOMON/BATCH.CONTROL.FILE,
// DISP=SHR
```

[Note]: This job step must be part of the regular CICS startup JCL if table suffixing is used.

SPECIAL PURPOSE COMANDS

Some commands that are available for special purposes are not present on the FILE CONTROL TABLE MENU. These are:

- 1) Deactivating the AUTOMON/BATCH product.
- 2) Disabling the Batch Interface program.
- 3) Enabling the Batch Interface program

THE 'Z' COMAND

The alpha command code for the listed items above, is 'Z'. There is no PF key equivalent for it. The command may be entered from a clear screen or a File Control display in the form:

FCTD Z xxxxxxx

where xxxxxxx is one of the following special command words:

- UNHOOK - Deactivate the AUTOMON/BATCH product entirely.
- STOPBCH - Deactivate the Batch Interface.
- STRTBCH - Reactivate the Batch Interface.

DEACTIVATING THE AUTOMON BATCH PRODUCT

There may be times when it is desirable to deactivate AUTOMON/BATCH. The most common reason would be to install a new version of the product or apply a PTF, which requires that a NEWCOPY of the program modules be performed. Another reason might be to restore CICS File Control activity to its normal unaltered state in order to diagnose some sort of system problem.

Whatever the reason, full deactivation of the product can be accomplished by use of the 'Z UNHOOK' command.

FCTD Z UNHOOK

from a clear screen or a File Control display.

The UNHOOK command will perform the following functions:

- 1) The File Control Table is reset to its original status after compilation. Only the fixed FCT entries will be present.
- 2) The 'in-use' condition for module FCTDTBL is reset so that a CEMT NEWCOPY can be performed on it.

REACTIVATING THE AUTOMON BATCH PRODUCT

After deactivation of AUTOMON/BATCH by the UNHOOK command, the first AUTOMON/BATCH command of any kind will automatically reactivate it.

DEACTIVATING THE BATCH INTERFACE

If the Batch Interface is in use (BCHINTF=YES in the User Option Table), there is always a suspended AUTOMON/BATCH transaction present in the system. This is the FCTD1000 program that has started an interval control initialted task (FCTD1000) and is waiting for it to expire.

The Batch Interface can be deactivated and this suspended transaction terminated by issuing the 'STOPBCH' command, in the following form:

FCTD Z STOPBCH

which may be issued from a clear screen or a File Control display.

REACTIVATING THE BATCH INTERFACE

After deactivating the Batch Interface by the STOPBCH command, it may be re-activated by means of the STRTBCH command, in the form:

FCTD Z STRTBCH

which can be entered from a clear screen or a File Control display.

CONTROLLING THE AUTOMATIC BACKOUT FEATURE

The FCTT transaction can be used to control the automatic backout feature, as well as review statistics for the automatic backout feature.

There are three commands that can be entered with the FCTT transaction. They are:

1) FCTT,START

This command will activate the FCTD1000 program, which 'wakes up' at periodic intervals to see if any files need to be backed out. The START command is not necessary if the FCTDTBL was compiled with BCHINTF=YES.

2) FCTT,START,INT=MMSS

This command can be used to start the FCTD1000 program and temporarily change the 'wake-up' interval coded in the user option table. The wake-up interval is coded in the user option table by means of the AUTOBINT keyword, which specifies the interval of time that FCTD1000 will be inactive before checking for anything to process.

To temporarily (until changed back or until CICS comes down) change the time interval, specify a new number of minutes and seconds following the INT= keyword in the command.

Note that if the FCTD1000 program is already active, the START command used in this option has no effect, however, the wake-up time interval will be changed.

3) FCTT,STOP

This command is used to deactivate the FCTD1000 program altogether. When the STOP command is issued, no more automatic backouts will occur until the START command is issued again.

MESSAGES

Each error message produced by AUTOMON/BATCH is prefixed by a 4-character message ID. If the message occurred on the system console, the message number is prefixed by 'FCTD'. The messages are presented in this section in message number sequence for ease of reference. The full message is listed, followed by an explanation of the cause of the message and the recommended action.

BATCH INTERFACE MESSAGES ('FCIDB' PREFIX)

These messages are produced by program FCTDBCHT.

FCTDB001. ERROR OCCURRED ON VSAM REQUEST MACRO xxxxxx, R15=yy, ERROR CODE=zz

While accessing the Batch Interface Control record in the VSAM control file (FCT\$FIL), an error occurred on a VSAM request macro.

In the message, xxxxxx indicates the type of macro, yy indicates the return code in R15 and zz indicates the VSAM error code.

ACTION: Look up the error code in the VSAM MESSAGES manual. Correct the error condition if possible, otherwise call UNICOM Systems Technical Support at 1-818-838-0606.

FCTDB002. JOB CANCELLED BY OPERATOR

This message indicates controlled cancellation of the FCTDBCH program. This could be a result of some error condition which was indicated by a previous message or a response of CANCEL by the system console operator to a AUTOMON/BATCH message.

ACTION: No action required.

FCTDB003. INVALID TRANCODE - xxxx

The transaction code for all batch AUTOMON/BATCH commands must be 'FCTD'. The previous command was entered incorrectly.

ACTION: Re-enter the command with the FCTD transaction code as the first 4 characters.

FCTDB004. INVALID COMMAND CODE - x

In the previous command, the command code was not one of the acceptable commands for program FCTDBCHT. Valid commands are:

- A - ACTIVATE
- B - BACKOUT
- C - CLOSE
- O - OPEN
- V - VARY
- X - ALLOCATE (MVS only)
- Y - DEALLOCATE (MVS only)
- Z - (Only the STOPBCH command is accepted.)

ACTION: Correct and re-enter the command.

FCTDB005. DATASET NAME NOT PRESENT OR INVALID

The dataset name in the previous AUTOMON/BATCH command was omitted or entered without a preceding space.

ACTION: Correct and re-enter the command.

FCTDB006. UNKNOWN KEYWORD IN COMMAND - xxxxxxxxxxxxxxxx

The transaction code, command code and dataset name in the previous command was entered correctly but the data following the dataset name (indicated by xxxxxxxxxxxxxxxx) could not be recognized as valid command data.

ACTION: The only acceptable entries following the dataset name are GROUP, GRP and CICS=xxxxxxx. Correct the command and re-enter.

FCTDB007. REPLY CANCEL, IGNORE OR CORRECT STATEMENT

This message appears following any incorrect AUTOMON/BATCH batch command, whether entered on the console or through a jobstream.

ACTION: Enter one of the following on the system console:

- CANCEL - Cancel the FCTDBCH job at this point.
- IGNORE - Ignore the statement in error and process the next statement.

Re-enter the statement in error for reprocessing.

FCTDB008. CICS SYSTEM 'xxxxxxx' NOT WAITING OR INACTIVE

The CICS system designated by the CICS= keyword (either entered on the previous command or previously established as the default) does not have an cross-partition wait outstanding. One of the following conditions is true:

- 1) The CICS= keyword was omitted or entered in error.
- 2) The designated CICS system is inactive.
- 3) The AUTOMON/BATCH Batch Interface is not active in the designated CICS system.

ACTION: If the application ID has not been entered correctly by means of the CICS=xxxxxxx keyword, either re-enter the command with the CICS= keyword on it or establish a default CICS ID by a stand-alone CICS=xxxxxxx command.

If the designated CICS system is not active, no action is required since the batch jobstream to be run will have no interference from CICS.

If the designated CICS system is up, the Batch Interface should be activated by means of an on-line FCTD Z STRTBCH command, then the FCTDBCH command can be re-entered.

This message will always be followed by messages FCTDB009 and FCTDB010.

FCTDB009. CICS INACTIVE=[BYPASS | CANCEL | IGNORE] IS IN EFFECT

The specified CICS is inactive or the Batch Interface Control Record for the partition or region could not be located. This message specifies the action taken by FCTDBCHT. For more information see *ESTABLISHING A DEFAULT CICS APPLICATION ID* in *Section 06 - THE BATCH INTERFACE TO AUTOMON/BATCH*.

ACTION: No action is required.

FCTDB010. BYPASSING FOLLOWING COMMANDS

FCTDB010. - - - - END OF BYPASS - - - -

The specified CICS is inactive or the Batch Interface Control Record for the partition or region could not be located. This message displays the commands that were not executed, then displays the *END OF BYPASS* message. FCTDBCHT will take the action specified by the INACTIVE keyword that is in effect. For more information see *ESTABLISHING A DEFAULT CICS APPLICATION ID* in *Section 06 - THE BATCH INTERFACE TO AUTOMON/BATCH*.

ACTION: No action is required.

FCTDB011. "INACTIVE=" PARM MUST BE "IGNORE", "BYPASS" OR "CANCEL", USING DEFAULT

A CICS=xxxxxxx, INACTIVE=[BYPASS | CANCEL | IGNORE] control statement was encountered with an invalid specification for the INACTIVE operand.

Processing will continue with the default action for your operating system. For MVS, the default action is CANCEL. For VSE, the default action is BYPASS. For more information see *ESTABLISHING A DEFAULT CICS APPLICATION ID* in section 06 - *THE BATCH INTERFACE TO AUTOMON/BATCH*.

ACTION: Correct the keyword in the FCTDBCHT jobstream.

FCTDB012. ENTER FCTD COMMAND, EOB TO END

When using console input to FCTDBCH (// UPSI 1), this message indicates that FCTDBCH is ready to accept a command.

ACTION: Either enter a valid FCTDBCH command or terminate the program by pressing the ENTER/END key. Note that you may also terminate FCTDBCH by entering a /*, /& or //. You may also enter CANCEL to cancel the program.

FCTDB014. MESSAGE QUEUE FULL FOR xxxxxxxx

All available message records for the CICS region with APPLID xxxxxxxx are in use and no more AUTOMON/BATCH commands can be issued from the FCTDBCHT program. There are 256 message queue records available for each CICS region. You can not issue more than 256 commands from any one execution of FCTDBCHT.

ACTION: If this message occurs, it probably means that the on-line Batch Interface program (AUTOMON/BATCH) was terminated by some means other than by a STOPBCH command. If this is the case, restart the Batch Interface with a STRTBCH command. This will cause the unused message queue records to be freed.

This condition could also occur if several executions of FCTDBCHT are running concurrently and the total number of commands for all executions exceed 256. If this is the case, let some of the the other partitions or regions complete, then restart this one.

FCTDB015. THE FOLLOWING COMMANDS HAVE NOT COMPLETED

FCTDB016. DESTINATION CICS SYSTEM = xxxxxxxx

These two messages appear when the FCTDBCHT program has waited for two minutes without further processing of the remaining commands for CICS region xxxxxxxx. All unprocessed commands are listed following these messages. This probably means that the on-line Batch Interface program (AUTOMON/BATCH) was terminated by some means, including a STOPBCH command.

ACTION: If the CICS region is down, reply CANCEL to message FCTDB017. If CICS is up, issue a STRTBCH command to restart the Batch Interface, then respond with RETRY or CONTINUE (depending on your needs) to message FCTDB017.

FCTDB017. REPLY CANCEL, EOJ, RETRY, DISP ? OR HELP FOR INSTRUCTIONS

This message is requesting directions for the condition reported by messages FCTDB015 and FCTDB016. 'CANCEL' will cause FCTDBCHT to ignore the unprocessed commands and go to end of job with a return code of eight. 'EOJ' will cause FCTDBCHT to ignore the unprocessed commands and go to end of job with a return code of zero. 'RETRY' will cause FCTDBCHT to wait for up to two more minutes for CICS to complete the listed commands. 'DISP ?' (where ? is 'A', 'U', 'S' or 'P') will redisplay 'A'll, 'U'nsuccessful, 'S'uccessful or 'P'ending messages. And 'HELP' will display a brief description of the available commands.

ACTION: Reply with the desired command.

FCTDB018. TRANID NOT SUPPLIED WITH START CMD

When using the START command from the Batch Interface, you must code the transaction ID to be started with the TRANID keyword.

ACTION: Correct the statement and re-execute the command.

FCTDB019. STOPBCH IS IN EFFECT FOR CICS - xxxxxxxx

The Batch Interface command could not complete because the CICS system indicated by xxxxxxxx has received a STOPBCH command, thereby deactivating the Batch Interface.

ACTION: A STRTBCH command must be issued for this CICS system before any Batch Interface commands will be accepted.

FCTDB020. NO DATA PASSED IN PARM STATEMENT

The PARM statement was found on the EXEC statement for the FCTDBCHT program, but no data was found between the quotes. The PARM is used to designate the user option table suffix code when the program is run as part of the CICS start-up JCL. The format of the PARM statement is

PARM='FTBL=xx'

where xx is the user option table suffix code.

ACTION: Correct the statement and re-execute the program.

FCTDB021. (FTBL=) KEYWORD NOT FOUND IN PARM STMT

The PARM statement on the EXEC command for the Batch Interface program did not contain the proper keyword for establishing the user option table suffix. The PARM is used to designate the user option table suffix code when the program is run as part of the CICS start-up JCL. The format of the PARM statement is:

PARM='FTBL=xx'

where xx is the user option table suffix code.

ACTION: Correct the statement and re-execute the program.

FCTDB022. FAMILY RCD (xxxxxxx) NOT FOUND

The Batch Interface command could not complete because the family record denoted by xxxxxxxx could not be found in the AUTOMON/BATCH control file.

ACTION: Ensure that the family name is a valid family and that it exists in the CICS that the batch interface was searching.

FCTDB023. B/I CNTL RCD FOR (xxxxxxx) NOT FOUND

The Batch Interface process could not complete because the control record for the CICS Application ID denoted by xxxxxxxx could not be found in the AUTOMON/BATCH control file.

ACTION: Ensure that the APPLID is a valid and that it exists in the AUTOMON/BATCH control file that the batch interface was searching.

ON LINE OPERATIONAL MESSAGES ('F' PREFIX)

The following messages are produced by program FCTD1000.

F0104. REQUESTED SCREEN RECORD xxxxxxxx NOT ON FILE

This message will display if there is a problem with the installation of AUTOMON/BATCH. One of the screen maps, identified by xxxxxxxx, that AUTOMON/BATCH uses could not be found in the AUTOMON/BATCH control file.

ACTION: Re-check the installation for the following possibilities:

- 1) The AUTOMON/BATCH control file was not defined correctly either in the IDCAMS definition or in the DFHCSD file.

- 2) The IDCAMS REPRO step did not complete normally.

F0105. FAMILY RECORD NOT FOUND - xxxxxxxx

The requested Family record record identified by the xxxxxxxx could not be found in the AUTOMON/BATCH control file.

ACTION: Check to see that you are requesting a family record that has been defined to AUTOMON/BATCH after entering the FAMILY PROCESSING selection from the main menu.

F0108. INVALID FUNCTION REQUEST

The command issued or the PF key pressed is not valid at this point in AUTOMON/BATCH operation.

ACTION: Review the available commands at whatever AUTOMON/BATCH screen is currently displayed and enter a valid command.

F0118. VSAM ERROR, RC=rr, ERRCODE=X'cc'

A VSAM request error has occurred in the Dataset Definition File or the requested dataset. The VSAM return code is indicated by rr, and the error code is indicated in hex by cc.

ACTION: Look up the VSAM error code in the VSAM Messages Manual. If the problem is not obvious, review the installation parameters for the AUTOMON/BATCH control file. If everything is correct, run an IDCAMS LISTCAT to see if VSAM is reporting an error. If nothing can be determined, attempt to REPRO the file to a tape or another disk area, then delete and redefine the file and reload it.

If the problem is persistent, call UNICOM Systems Technical Support at 1-818-838-0606.

F0132. SECURITY VIOLATION, TRANSACTION TERMINATED

The operator does not have the necessary security clearance to perform the requested command on the designated file definition. This may be due to one of the following types of security checks:

- 1) The operator security key in the CICS Sign-on Table does not match the security key specification in the Security Control Record for this dataset label.
- 2) The requested password is not valid for the dataset label.
- 3) The user-written Security Exit has returned a "not-authorized" code for the designated dataset label.

ACTION: If you should have access to this file definition, consult your System Programmer or review *Section 10 - THE SECURITY CONTROL SYSTEM*.

F0141. [ENABLE | DISABLE] COMPLETE FOR xx RESOURCES. (yy FAILURES)

In response to an ENABLE or DISABLE function of a Transaction/Program Family, this message verifies that xx transaction or programs were enabled/disabled, and that the enable/disable of yy transactions/programs failed.

ACTION: If there were zero failures, no action is required. If failures did occur, then the PPT or PCT entries could not be located.

F0142. MUST HAVE QUALIFIER OF "TRAN", "PROG" OR "GR???"

A family command was issued without the trailing qualifier.

ACTION: See Section 3 - Grouping Files, Transactions and Programs for correct command format.

F0905. ERROR xx HAS OCCURRED ON I/O FOR FCTDBCH

When accessing the Control Record (record key XFCTDBCH) in FCT\$FIL, error code XX was returned from CICS.

ACTION: The error code can be found in the CICS MACRO LEVEL PROGRAMMER'S REFERENCE MANUAL under Return Codes From File Control Operations. Determine the meaning of the code and correct the error condition, if possible. Otherwise call UNICOM Systems.

F0908. DFHIC ERROR, DEC RC=xxx

An error return code has been returned by CICS in response to an interval control command by AUTOMON/BATCH. The return code can be found in the MACRO LEVEL CICS APPLICATION PROGRAMMERS REFERENCE MANUAL under INTERVAL CONTROL RESPONSE CODES.

ACTION: Determine what the error code means, then if it is not clear what the problem is, call UNICOM Systems Technical Support at 1-818-838-0606.

F0909. xxxxxxxx IS CLOSED, B/I INTERVAL SET TO 120

When the batch interval program wakes up and finds the VSAM control file closed it issues this message. It then restarts the batch interface time interval at 120 seconds. It will continue to wake-up every 120 seconds and check the file until either the file is opened or a STOPBCH command is issued to deactivate the batch interface.

ACTION: Either open the file or do an FCTD Z STOPBCH command to deactivate the batch interface.

F0911. BATCH INTERFACE SUCCESSFULLY [ACTIVATED | DEACTIVATED]

This is an informational message confirming that the requested function was performed successfully.

ACTION: No action is required.

F0912. BATCH INTERFACE IS NOT CURRENTLY ACTIVE

A request to deactivate the Batch Interface was issued; however, it was not performed because the Batch Interface is already inactive.

ACTION: No action is required.

F0913. BATCH INTERFACE IS ALREADY ACTIVE

A request to activate the Batch Interface was issued; however, it was not performed because the Batch Interface is already active.

ACTION: No action is required.

F0914. CEMT R/C => (xxx)

CEMT has returned an error condition for the request.

ACTION: The decimal return code can be found in the CICS MESSAGES MANUAL in the DFH30xx (DFHMTPA) MESSAGES section.

F0915. CEMT ABENDED ON FILE=xxxxxxx. BCHTIMER CONTINUING.

CEMT encountered an abnormal termination for the file xxxxxxxx. The batch program will continue processing.

ACTION: Perform the command that failed in an on-line partition to receive the full error message.

F0916. MUST HAVE QUALIFIER OF "TRAN", "PROG" OR "GR???"

The command was not executed because no valid qualifier was found.

ACTION: Review the appropriate section in this manual for correct command format.

F0917. xxxxxxxx FAMILY FILE RECORD NOT FOUND

The file family record xxxxxxxx could not be located in the AUTOMON/BATCH control file that was being searched.

ACTION: Ensure that the family does exist by viewing the list of families on the Family Directory Display and reissue the command.

F0918. xxxxxxxx FAMILY TRAN/PROG RECORD NOT FOUND

The transaction/program family record xxxxxxxx could not be located in the AUTOMON/BATCH control file that was being searched.

ACTION: Ensure that the family does exist by viewing the list of families on the Family Directory Display and reissue the command.

F1002. xxxxxxxx FILE NOT OPEN

The AUTOMON/BATCH Control File denoted by xxxxxxxx, is closed.

ACTION: Determine why the file is closed or use CEMT or CSMT to open it.

F1011. ABATCH HAS BEEN UN-HOOKED

This message is merely to confirm that AUTOMON/BATCH has been successfully un-hooked, in response to a FCTD Z UNHOOK command.

ACTION: No action is required.

F1016. ** WARNING ** AUTOMON/BATCH EXPIRES IN xx DAYS

This message indicates that the end of the trial period for AUTOMON/BATCH is approaching. This message will start appearing within seven days of the expiration date.

ACTION: No action is required at this time; the product is still operational. If it is allowed to expire, however, it can no longer be used. Notify your sales representative at UNICOM Systems.

F1019. AUTOMON/BATCH PASSWORD ERROR ==> CODE(x)

This message indicates that the product has expired or cannot be used because of an installation error. The product is now inactive and cannot be used.

The "x" in the above message will be one of the following error codes:

- 1 - The current date is greater than the password expiration date.
- 2 - Module "STSPASS" could not be found in a library in the search string.
- 3 - The product ID could not be located in STSPASS.
- 4 - The password in STSPASS is invalid.
- 5 - Module "STS0100" could not be found in a library in the search string.
- 6 - STSPASS contains a CPU ID that is invalid for the machine that it is installed on.
- 7 - The password entered is not valid for the requested product.
- 9 - An undetermined error has been encountered. Please call UNICOM Systems Technical Support at 1-818-838-0606.

ACTION: Correct the error if possible, or call UNICOM Systems Technical Support at 1-818-838-0606.

F1020. INCOMPATIBLE CICS RELEASE, WRONG VERSION OF ABATCH INSTALLED

This release of AUTOMON/BATCH does not support the version of CICS in which this message occurred.

ACTION: Contact your UNICOM Systems Sales Representative to make arrangements for receiving a version of AUTOMON/BATCH that will operate in your version of CICS.

F1021. CANNOT ALLOCATE/DEALLOCATE OPEN FILE

Open files are already allocated and cannot be de-allocated while the file is open.

ACTION: In order to de-allocate the file it must first be closed; or, instead of deallocation, you may want to backout the file, which can be performed on an open file.

F1031. ABATCH SUCCESSFULLY UNHOOKED

This message is merely to confirm that AUTOMON/BATCH has successfully removed its interception points from CICS, in response to a FCTD Z UNHOOK command.

ACTION: No action is required. A CEMT NEWCOPY command may be performed if desired. Note that any AUTOMON/BATCH command will cause AUTOMON/BATCH to replace the interception points within CICS again.

F1034. ABATCH CANNOT BE INITIALIZED

One or more serious errors have occurred during the initialization step of AUTOMON/BATCH, which prevents successful initialization. These conditions are indicated by message appearing on the system console prior to this message.

ACTION: Refer to the previous message for an explanation of the error condition. Correct the condition(s) according to the recommended action for the message(s), then restart CICS.

F1037. ABATCH SUCCESSFULLY RE-HOOKED

This message is merely to confirm that AUTOMON/BATCH has been successfully re-hooked, in response to any AUTOMON/BATCH command while AUTOMON/BATCH is de-activated.

ACTION: No action is required.

F1040. FILE xxxxxxxx [OPENED | CLOSED] AT hh:mm BY TERM tttt

This message appears on the system console when a file is opened or closed by a terminal operator using the FCTD transaction.

ACTION: No action is required.

F1501. INVALID LABEL PARAMETER LIST

This is an internal error which occurred during file label activation. It indicates some kind of problem in the AUTOMON/BATCH package, or some unique situation which cannot be properly handled.

ACTION: Notify UNICOM Systems Technical Support at 1-818-838-0606.

F1502. NO GETVIS STORAGE AVAILABLE FOR LABEL

There is insufficient GETVIS storage available to add another label.

ACTION: You can free some GETVIS storage by closing one or two CICS files. To prevent this situation, either decrease the SIZE parameter on the EXEC statement in the CICS start-up JCL or increase the partition size for CICS.

F1503. EQUAL FILE LABEL IN LABEL AREA

For BDAM files, there is another label in either the standard label area or the User Label area which has the same seven-character file ID as the Dataset Name of this file definition, but with a different 44-character file label. If AUTOMON/BATCH creates a label for this file definition in the Partition Standard label area, there would be a label conflict which could effect CICS.

ACTION: Either remove the other label or change the dataset name of this definition to something else by deleting and re-creating this definition.

F1504. REQUEST MUST FOLLOW ENDLBL FUNCTION

This is an internal error which occurred during file label activation. It indicates some kind of problem in the AUTOMON/BATCH package, or some unique situation which can not be properly handled.

ACTION: Notify UNICOM Systems Technical Support at 1-818-838-0606.

F1505. REQUEST MUST FOLLOW MODGRPL FUNCTION

This is an internal error which occurred during file label activation. It indicates some kind of problem in the AUTOMON/BATCH package, or some unique situation which can not be properly handled.

ACTION: Notify UNICOM Systems Technical Support at 1-818-838-0606.

F1506. NO SPACE AVAILABLE IN LABEL AREA

The Partition Standard Label area is full. No more labels can be added.

ACTION: Run LSERV to list all label areas. If you have permanent (NON-AUTOMON/BATCH) labels in the PARSTD area, move them to the Standard Label area or include them in the CICS start-up JCL to free up space in the PARSTD area. If all the PARSTD labels are AUTOMON/BATCH labels, provide DLBL statements for some or all of the AUTOMON/BATCH file and put the labels for those files somewhere other than the PARSTD area.

If This message is appearing when running the AUTOMON/BATCH Print Utility program (FCTDPRT), you may provide additional DLBL statements in the JCL for some or all of the AUTOMON/BATCH files, or you may specify // UPSI 1 to suppress the VSAM statistics on the report. This will stop FCTDPRT from opening the VSAM files, thus eliminating the need for labels.

You may optionally increase the size of the PARSTD label area by means of the DLA (Define Label Area) command in the ASI procedure at IPL. See the SYSTEM CONTROL STATEMENTS manual for an explanation of the DLA command.

F1507. BAD RETURN FROM ENDLBL FUNCTION

This is an internal error which occurred during file label activation. It indicates some kind of problem in the AUTOMON/BATCH package, or some unique situation which can not be properly handled.

ACTION: Notify UNICOM Systems Technical Support at 1-818-838-0606.

F1508. DEVICE xxx NOT DEFINED IN PUB TABLE

The disk address defined in The Channel Units field for this BDAM file is not a valid device. xxx indicates the device in error.

ACTION: Correct The Channel Units field and issue the command again.

F1509. NO FREE LUB ENTRY FOUND

All of the Logical Unit Blocks in the CICS partition are in use, the disk address specified in Channel Units is not assigned and the assignment can not be made.

ACTION: Either assign the device using a temporary or permanent assignment or increase the number of LUBS for the CICS partition.

F1510. DEVICE xxx NOT DEFINED IN PUB TABLE

The disk address defined in The Channel Units field for this BDAM file is not a valid device. xxx indicates the device in error.

ACTION: Correct The Channel Units field and issue the command again.

F1511. DEVICE xxx NOT A DISK DEVICE

The device indicated by xxx is a valid device address but is not a disk.

ACTION: Correct the Channel Units field and issue the command again.

F1512. DEVICE xxx IS DOWN

The disk at Channel Unit address xxx is off-line or disconnected.

ACTION: If the device can not be made ready you must use a different disk.

F1513. NO JIBS AVAILABLE FOR TEMP ASSIGN

One of the disk addresses in the Channel Units field must be assigned and there are no more Job Information Blocks available.

ACTION: Either assign the device using a temporary or permanent assignment or increase the number of JIBS for the CICS partition.

F1514. UNEXPECTED RETURN CODE FROM ASSIGN

This is an internal error which occurred during file label activation. It indicates some kind of problem in the AUTOMON/BATCH package, or some unique situation which cannot be properly handled.

ACTION: Notify UNICOM Systems Technical Support at 1-818-838-0606.

F1515. DEVICE xxx RESERVED OR MOUNT PENDING

The disk at Channel Unit address xxx is off-line or disconnected.

ACTION: If the device can not be made ready you must use a different disk.

F1516. NO GETVIS STORAGE AVAILABLE FOR ASSIGN

There is insufficient GETVIS storage available to add another label.

ACTION: You can free some GETVIS storage by closing one or two CICS files. To prevent this situation, either decrease the SIZE parameter on the EXEC statement in the CICS start-up JCL or increase the partition size for CICS.

F1517. I/O ERROR VTOC

An I/O error has occurred while reading the VTOC of the device specified in the Channel Units field.

ACTION: Try the command again. If the problem persists, it indicates a hardware error.

F1518. NO FORMAT1 LABEL FOR xxx-----xxx

There is no file present on the disk(s) specified in the Channel Units field with the label specified in the LABEL field.

ACTION: Insure that the label is specified correctly. If so, run a VTOC list on the specified disk(s). Either the file has been deleted or the disk address is incorrect.

F1519. UNEXPECTED RETURN CODE FROM VTOC READ xx

This is an internal error which occurred during the label activation. It indicates some kind of problem in the AUTOMON/BATCH package, if some unique situation can not be properly handled.

ACTION: Notify UNICOM Systems Technical Support at 1-818-838-0606.

F1520. EXISTING LABEL USING SYSxxx ASSIGNED TO yyy, NOT zzz

There is an existing label with the same file ID and file label already present for this BDAM file. However, the existing label is using a sys number which is assigned to a different device than that specified in Channel Units. In the message, xxx indicates the sys number, yyy indicates the current assignment, zzz indicates the requested assignment.

ACTION: If the assignment for the existing label is correct, change the Channel Units field to the same address (yyy). Otherwise the existing label must be removed.

F1522. ERROR OPENING VTOC ON xxx, RETURN CODE=yy

Some kind of error occurred during VTOC processing. In the message, xxx refers to the disk device and yy is the error code. These error codes are documented in the DOS/VSE message manual in Appendix X.

ACTION: Determine what the error code means and if the problem is not obvious, notify UNICOM Systems Technical Support at 1-818-838-0606.

F1523. ERROR CLOSING VTOC ON xxx, RETURN CODE=yy

Some kind of error occurred during VTOC processing. In the message, xxx refers to the disk device and yy is the error code. These error codes are documented in the DOS/VSE message manual in Appendix X.

ACTION: Determine what the error code means and if the problem is not obvious, notify UNICOM Systems Technical Support at 1-818-838-0606.

F1524. ONLY VSAM FILES ARE ALLOWED ON FBA DEVICES.

You can not define a BDAM file on an FBA disk device since these are not supported by IBM.

ACTION: Re-define the file as VSAM.

F1525. THERE RE MORE CHANNEL UNITS FOR THIS FILE THAT ARE NOT DEFINED

This BDAM file indicates multiple extents in the VTOC, but not all of the extents have been defined in the Channel Units field.

ACTION: Define the additional disk addresses and retry the command.

F1526. EXISTING SAM LABELS ARE NOT SUPPORTED.

There is an existing label present with the same file ID and file label but the type is 'SD' rather than 'DA'. You can define SAM files as BDAM, but you must either let AUTOMON/BATCH create the label or use a type 'DA' label.

ACTION: Either remove the 'SD' label or change the Dataset name by deleting and re-creating the file definition.

F1527. UNSUPPORTED DEVICE TYPE

The specified disk device is not supported by AUTOMON/BATCH. Supported disk types are 2311, 2314, 2319, 3330, 3350, and 3380.

ACTION: Determine your disk device type and notify UNICOM Systems Technical Support at 1-818-838-0606.

F1528. REQUEST MUST FOLLOW GETLBL OR INVALID CHANGE

This error can occur on BDAM or SAM files if you open or activate the file, then either increase or decrease the number of disk device addresses in the CHANNEL-UNITS field, then open or activate the file again, after deactivating the file.

This error can also occur if there is an existing label with the same file ID as this file but a different file type. A BDAM label exists for a VSAM file, for instance.

ACTION: If this is the situation, you must either create the file definition again using a different DATASET NAME, then open or activate it, or you can bring CICS down, clear the conflicting label, then bring CICS back up and then open or activate the file again.

F1529. DSN CONFLICT-xxxx.....xxxx

The following situation has been encountered, while attempting to dynamically allocate the dataset:

1) An existing allocation was found with the same DDNAME but a different DSNNAME.

2) An attempt was made to deallocate the conflicting label but the attempt failed, thus preventing allocation of this dataset.

ACTION: To determine the cause of the deallocation failure, define a file using AUTOMON/BATCH with the same DDNAME and the conflicting DSNNAME allocation (XXXX.....XXXX in the message). Issue a DEALLOCATE command for this file, which will return the error message from the deallocation routine.

If the conflicting allocation can not be cleared, you must define this dataset using a different DDNAME.

F2001. HI-LIGHTED FIELDS ARE IN ERROR

An error has been detected while editing the input data.

ACTION: Correct the hi-lighted field and press enter.

F2002. FAMILY FILE COULD NOT BE LOCATED

The requested family file record could not be found in the AUTOMON/BATCH control file.

ACTION: Ensure the family file has been defined according to the procedures described in *Section 3 - Grouping Files, Transactions and Programs*.

F2003. END OF FILE ENCOUNTERED.

End of file was encountered while processing the family request.

ACTION: No action required.

F2004. INVALID FUNCTION. PLEASE RE-ENTER

An invalid request was entered for a family file process.

ACTION: Re-enter the correct function code and press ENTER.

F2005. THIS FUNCTION REQUIRES A FAMILY NAME

The entered function requires the entry of a valid family file name.

ACTION: Enter a valid family file name and press ENTER.

F2006. XXXXXXXXX ALREADY EXISTS.

The family name entered on a "NEW" function is already on the AUTOMON/BATCH control file.

ACTION: Enter a new name for the family.

F0707. FILE DEFINITION (xxxxxxx) NOT FOUND

AUTOMON/BATCH could not locate the requested file definition.

ACTION: Ensure that the definition was spelled correctly, or use the multiple file or family name summary screen accordingly, to find the only known definitions.

F2101. HI-LIGHTED FIELDS ARE IN ERROR

An error has been detected while editing the input data.

ACTION: Correct the highlighted field and press ENTER.

F2102. FAMILY FILE COULD NOT BE LOCATED

The requested family file record could not be found in the AUTOMON/BATCH control file.

ACTION: Ensure the correct spelling of the family file, checking the FAMILY DIRECTORY if necessary.

F2103. END OF FILE ENCOUNTERED.

End of file was encountered while processing the family request.

ACTION: No action required.

F2104. XXXX IS AN INVALID FUNCTION

An invalid request was entered for a family file process.

ACTION: Re-enter the correct function code and press enter.

F2105. CURSOR NOT POSITIONED ON A FILE NAME

This function requires the cursor to be positioned within a file name field.

ACTION: Position the cursor within a file name field and reissue the command.

F2106. XXXXXXXXX ERROR WHEN RE-WRITING RECORD

An error was returned when trying to re-write the family file record indicated by xxxxxxxx.

ACTION: Note the error code and call UNICOM Systems Technical Support at 1-818-838-0606.

F2107. CURRENTLY POSITIONED AT FIRST FAMILY FILE RECORD

This is an informational message informing you that the current display is positioned at the first family file record in the AUTOMON/BATCH control file.

ACTION: No action is required.

F2108. CURRENTLY POSITIONED AT LAST FAMILY FILE RECORD

This is an informational message informing you that the current display is positioned at the last family file record in the AUTOMON/BATCH control file.

ACTION: No action is required.

F2201. HI-LIGHTED FIELDS ARE IN ERROR

An error has been detected while editing the input data.

ACTION: Correct the hi-lighted field and press enter.

F2202. FAMILY FILE COULD NOT BE LOCATED

The requested family file record could not be found in the AUTOMON/BATCH control file.

ACTION: Ensure the correct spelling of the family file, checking the FAMILY DIRECTORY if necessary.

F2203. END OF FILE ENCOUNTERED.

End of file was encountered while processing the family request.

ACTION: No action is required.

F2204. XXXXX IS AN INVALID FUNCTION

An invalid request was entered for a family file process.

ACTION: Re-enter the correct function code and press ENTER.

F2205. CURSOR NOT POSITIONED ON A RESOURCE NAME

This function requires the cursor to be positioned within a resource (tran/prog) name field.

ACTION: Position the cursor in a resource (tran/prog) name field and reissue the command.

F2206. XXXXXXXXX ERROR WHEN RE-WRITING RECORD

An error was returned when trying to re-write the family tran/prog record.

ACTION: Note the error code and call UNICOM Systems Technical Support at 1-818-838-0606.

F2207. CURRENTLY POSITIONED AT FIRST FAMILY TRAN/PROG RECORD

This is an informational message informing you that the current display is positioned at the first family tran/prog record in the AUTOMON/BATCH control file.

ACTION: Note the error code and call UNICOM Systems Technical Support at 1-818-838-0606.

F2208. CURRENTLY POSITIONED AT LAST FAMILY TRAN/PROG RECORD

This is an informational message informing you that the current display is positioned at the last family tran/prog record in the AUTOMON/BATCH control file.

ACTION: Note the error code and call UNICOM Systems Technical Support at 1-818-838-0606.

AUTOMN BATCH RDO INTERFACE

F5001. HI-LIGHTED FIELDS ARE IN ERROR

An error has been detected while editing the input data.

ACTION: Correct the hi-lighted field and press enter.

F5002. DFHEDAP (CEDA) LINK ERROR

An error occurred while linking to DFHEDAP to perform the RDO function.

ACTION: Call UNICOM Systems Technical Support at 1-818-838-0606.

F5003. CEDA COMPLETED SUCCESSFULLY

This is an informational message informing you that the RDO function just performed completed successfully.

ACTION: No action is required.

F5004. FILE CANNOT BE LOCATED IN CICS FCT

The target file name could not be found in the existing FCT. Either it has not been defined to CICS via the DFHCSD file or it has not been installed.

ACTION: Define or install the file as required.

F5005. END OF FILE ENCOUNTERED.

End of file was encountered while processing the family request.

ACTION: No action required.

F5101. FCTDTBL "TOTFILS" SPECIFICATION HAS BEEN EXCEEDED

The "TOTFILS" value specified the last time the FCTDTBL macro was assembled has been exceeded. Although processing by FCTD can continue, it will not be able to track activity on any file installed or opened after this message.

ACTION: Increase the FCTDTB "TOTFILS" value and re-assemble the FCTDTBL macro.

F5301. HI-LIGHTED FIELDS ARE IN ERROR

An error has been detected while editing the input data.

ACTION: Correct the hi-lighted field and press enter.

F5302. DFHEDAP (CEDA) LINK ERROR

An error occurred while linking to DFHEDAP to perform the RDO function.

ACTION: Call UNICOM Systems Technical Support at 1-818-838-0606.

F5303. CEDA COMPLETED SUCCESSFULLY

This is an informational message informing you that the RDO function just performed completed successfully.

ACTION: No action required.

F5304. FILE CANNOT BE LOCATED IN CICS FCT

The target file name could not be found in the existing FCT. Either it has not been defined to CICS via the DFHCSD file or it has not been installed.

ACTION: Define or install the file as required.

F5305. END OF FILE ENCOUNTERED.

End of file was encountered while processing the family request.

ACTION: No action required.

F5401. HI-LIGHTED FIELDS ARE IN ERROR

An error has been detected while editing the input data.

ACTION: Correct the highlighted field and press enter.

F5403. FUNCTIONS COMPLETED SUCCESSFULLY

This is an informational message informing you that the function just performed completed successfully.

ACTION: No action required.

F5404. FILE CANNOT BE LOCATED IN CICS FCT

The target file name could not be found in the existing FCT. Either it has not been defined to CICS via the DFHCSD file or it has not been installed.

ACTION: Define or install the file as required.

F5405. END OF FILE ENCOUNTERED.

End of file was encountered while processing the family request.

ACTION: No action required.

AUTOMON BATCH INITIALIZATION STAGE

F6001. DFHEDAP (CEDA) LINK ERROR

An error occurred while linking to DFHEDAP to perform the RDO function.

ACTION: Call UNICOM Systems Technical Support at 1-818-838-0606.

F6002. CEDA COMPLETED SUCCESSFULLY

This is an informational message informing you that the RDO function just performed completed successfully.

ACTION: No action required.

F6003. FILE CANNOT BE LOCATED IN CICS FCT

The target file name could not be found in the existing FCT. Either it has not been defined to CICS via the DFHCSD file or it has not been installed.

ACTION: Define or install the file as required.

F6004. PROCESSING GROUP (XXXXXXXX)

This is an informational message informing you of the current DFHCSD group being processed by the FCTD initialization stage.

ACTION: No action required.

F6005. ABATCH FILE TABLE PREPARATION COMPLETE

This is an informational message informing you that the AUTOMON/BATCH file table has been successfully built.

ACTION: No action required.

F6006. "TOTFILS" ALLOCATION IN FCTDTBL IS TOO SMALL

This is an informational message informing you that the AUTOMON/BATCH file table has been successfully built, but certain files are inaccessible to AUTOMON/BATCH. The files exist in the DFHCSD file, but the total number of files in the DFHCSD file exceeds the number of files allocated to AUTOMON/BATCH.

ACTION: Increase the number of "TOTFILS" using the FCTDTBL macro, then deactivate AUTOMON/BATCH using the UNHOOK command as described in *Section 14 - Special Purpose Commands*. AUTOMON/BATCH may then be reactivated by issuing any valid AUTOMON/BATCH command.

LOCATE FUNCTION MESSAGES ('L' PREFIX)

L1000. END OF FILE HAS BEEN REACHED

The end of the input file has been reached during the record selection operation. The LOCATE function is terminated.

ACTION: Proceed with any terminal activity. If you re-enter the 'L' command at this point, the LOCATE specifications from the previous operation will be filled-in on the specifications screens.

L1001. PRESS ENTER TO CONTINUE LOCATE, PF3 TO END LOCATE

For output disposition 1, DISPLAY THE RECORD AND OPTIONALLY CONTINUE THE SEARCH, this message indicates that a qualifying record has been found. The record is displayed, with the qualifying data highlighted.

ACTION: Press ENTER to continue the record selection operation with the next sequential record, or press PF3 to terminate the LOCATE operation. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation.

L1002. PRESS ENTER TO ADD THIS RECORD, PF3 TO BYPASS

For one of the output dispositions involving optional record addition, this message indicates that a qualifying record has been found. The record is displayed, with the qualifying data highlighted.

ACTION: Press ENTER to add this record to the output file and continue the record selection operation with the next sequential record, or press PF3 to skip this record and continue. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation. If you wish to terminate the LOCATE operation, press CLEAR.

L1003. PRESS ENTER TO REWRITE THIS RECORD, PF3 TO BYPASS

For the output dispositions involving optional record modification and rewrite, this message indicates that a qualifying record has been found. The record is displayed, with the qualifying data highlighted and the modifications in place.

ACTION: Press ENTER to rewrite this record in the input file and continue the record selection operation with the next sequential record, or press PF3 to skip this record and continue. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation. If you wish to terminate the LOCATE operation, press CLEAR.

L1004. PRESS ENTER TO DELETE THIS RECORD, PF3 TO BYPASS

For the output disposition involving optional record deletion, this message indicates that a qualifying record has been found. The record is displayed, with the qualifying data highlighted and the modifications in place.

ACTION: Press ENTER to rewrite this record from the input file and continue the record selection operation with the next sequential record, or press PF3 to skip this record and continue. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation. If you wish to terminate the LOCATE operation, press CLEAR.

L1005. DUPLICATE FOUND, ENTER TO REPLACE, PF3 TO BYPASS

For one of the output dispositions involving record addition, this message indicates that a duplicate record has been found in the output file. The record from the output file is displayed. Note that you can avoid this message by selecting BYPASS or REPLACE as the DUPLICATE RECORD OPTION in the OUTPUT DISPOSITION INSTRUCTIONS screen.

ACTION: Press ENTER to replace this record with the input record and continue the record selection operation with the next sequential record, or press PF3 to skip this record and continue. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation. If you wish to terminate the LOCATE operation, press CLEAR.

L1006. xxxx RECORDS REACHED WITH NO QUALIFIERS, ENTER TO CONTINUE, PF3 TO END

This message indicates that the number of consecutive records to read without pausing when no records match the input query has been reached. The xxxx in the message is the number of consecutive records which did not match the query. The current record is displayed.

ACTION: Press ENTER to continue the record selection operation with the next sequential record, or press PF3 to terminate the LOCATE operation. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation.

L1007. LOCATE FUNCTION TERMINATED

This message displays then you press PF3 or enter 'R' at any Locate message where PF3 is an option to terminate the Locate operation.

ACTION: Proceed with any terminal activity. If you re-enter the 'L' command at this point, the LOCATE specifications from the previous operation will be filled-in on the specifications screens.

L1008. NON-NUMERIC DATA AT POSITION xxxx, LENGTH yy, ENTER= CONTINUE, PF3=END

This message indicates that a numeric value was entered for the COMPARISON DATA on the Input Selection Specifications screen and during the record selection operation, a record was found which contained invalid numeric data at the specified position. The record in question is displayed. The xxxx in the message indicates the record position and yy indicates the length. AUTOMON/BATCH considers this record a non-qualifying record for the selection.

ACTION: Press ENTER to continue the record selection operation with the next sequential record, or press PF3 to terminate the LOCATE operation. You may also perform any valid FSRV command at this time, then change the command back to 'L' and press ENTER to continue the LOCATE operation. If you correct the invalid data with a CHANGE command at this point and want the record to be re-evaluated for selection, you must back up one record in the file using the 'B' command (or PF2), then change the command to 'L' and press ENTER.

L1009. xxxx RECORDS SELECTED, PRINT STARTED

When using the record print option of the Output Dispositions, this message indicates that a qualifying record was found and a task has been started to print it on the on-line printer. The xxxx in the message is a running count of the number of records sent to the printer.

ACTION: No action is possible. The terminal is not available for input and will not be until end of file or a Locate pause occurs.

L1010. UNABLE TO START PRINT TASK, CODE =xx

When using the record print option of the Output Dispositions, this message indicates that an error occurred while attempting to start a task to print a record on the on-line printer. The xx in the message is the CICS return code for DFHIC commands.

ACTION: Look up the DFHIC code in the MACRO-LEVEL APPLICATION PROGRAMMERS REFERENCE MANUAL under INTERVAL CONTROL RESPONSE CODES. If the reason is not obvious, call UNICOM Systems Technical Support at 1-818-838-0606.

L1011. ERROR HAS OCCURRED USING TEMP-STORAGE, CODE=xx

An error return code has been returned by CICS in response to a Temporary Storage GET or PUT command by AUTOMON/BATCH. This could be caused by using Auxiliary Temporary Storage with an insufficient record size or control interval size defined for the Auxiliary file. The return code can be found in the Macro Level CICS Application Programmers Reference Manual under Temporary Storage Response Codes.

ACTION: Determine what the error code means, then if it is not clear what the problem is, call UNICOM Systems Technical Support at 1-818-838-0606.

FILE SERVICE UTILITY MESSAGES ('R' PREFIX)

The following messages (those starting with 'R') pertain to the File Service Utility functions described in *Section 8 - THE FILE SERVICE FUNCTION OF AUTOMON/BATCH* of this guide. These messages are produced by program FSRV1000.

R0001. KEY DATA AND PRESS ENTER TO ADD RECORD

This message is produced when an 'N' command has been issued to add a new record to the file.

ACTION: Key data in the character side, or issue the 'T'oggle command and key data in the hexadecimal side, then press ENTER to add the record to the file.

R0002. TO MODIFY, MAKE CHANGES AND PRESS ENTER

This message is produced when a 'C' command has been issued to modify a file record.

ACTION: Make changes and press ENTER.

R0003. PRESS ENTER TO DELETE THIS RECORD

This message is produced when a 'D' command has been issued to delete a record from a file.

ACTION: Press ENTER to delete the record, or change the command line.

R0004. REQUESTED SCREEN RECORD xxxxxxxx NOT ON FILE

This message will display if there is a problem with the installation of AUTOMON/BATCH. One of the screen maps that AUTOMON/BATCH uses, identified by xxxxxxxx, could not be found in the Dataset Definition File.

ACTION: Re-check the installation for the following possibilities:

- 1) The Dataset Definition File was not defined correctly either in the IDCAMS definition or in the CICS FCT definition.
- 2) The IDCAMS REPRO step did not complete normally.

R0005. DATA RECORD NOT FOUND - xxxxxxxx

The requested data record identified by the xxxxxxxx could not be found in the File.

ACTION: If the record key was mis-keyed, correct the command line and press ENTER. If it was keyed properly, ensure that the record has not been previously deleted.

R0006. END OF FILE REACHED

This message occurs when moving forward or backward in the file and the end of the file is reached.

ACTION: Move forward or backward accordingly.

R0007. REQUESTED MENU OPTION NOT DEFINED ON MENU

This message occurs when requesting an option that is not valid for this menu.

ACTION: Review the options of the menu.

R0008. INVALID FUNCTION REQUEST

The command issued or the PF key pressed is not valid at this point in AUTOMON/BATCH operation.

ACTION: Review the available commands at whatever AUTOMON/BATCH screen is currently displayed and enter a valid command.

R0009. RECORD SUCCESSFULLY DELETED

This message occurs in response to a delete request, to inform that no errors were encountered.

ACTION: No action is required.

R0010. DUPLICATE DATA RECORD ON FILE

While attempting to add a data record, it could not be added because this would have created a record key that duplicated an already existing key.

ACTION: For the record to be added, it must possess a key that is distinctive. Change the key and press ENTER.

R0011. UNSUPPORTED CICS FUNCTION FOR THIS FILE

This message indicates that one of the required service request parameters in the SERVREQ operand of the DFHFCT macro for the Dataset Definition File is not present. This is usually an installation problem.

ACTION: Review the CICS FCT definition for the Dataset Definition File, or display FCT\$FIL using CEMT to insure that all required service requests are present. The required options are UPD, ADD, BRO and DEL.

If the problem persists, call UNICOM Systems Technical Support at 1-818-838-0606.

R0012. I/O ERROR HAS OCCURRED

CICS has returned an I/O error return code in response to some kind of file request by AUTOMON/BATCH. This would usually indicate a damaged File, although it could also be caused by incorrectly defining the file either to VSAM or to CICS.

ACTION: Review the installation parameters for the file. If everything is correct, run an IDCAMS LISTCAT to see if VSAM is reporting an error. If nothing can be determined, attempt to REPRO the file to a tape or another disk area, then delete and redefine the file and reload it.

R0013. DATA FILE IS FULL

No more space is available for record addition to this file.

ACTION: Increase the size of the file.

R0014. FILE IS CLOSED TO ONLINE USERS

The designated dataset is closed to CICS.

ACTION: Determine why the file is closed or use CEMT or CSMT to open it.

R0015. FCT ENTRY NOT DEFINED FOR xxxxxxxx

This message indicates that the requested dataset is not known to CICS.

ACTION: Verify that the correct dataset name was used and if so, define the dataset to CICS with an FCTD NEW command.

R0016. RECORD ADDED SUCCESSFULLY

This message is issued in response to successfully completing an 'N' command to add a new record.

ACTION: No action is required.

R0017. UNKNOWN FILE RESPONSE, DEC. CODE=xxx

VSAM returned an unknown response.

ACTION: The decimal code can be found in the VSAM MESSAGES MANUAL for request macros.

R0018. VSAM ERROR, RC=rr, ERRCODE=X'cc'

A VSAM request error has occurred in the requested dataset. The VSAM return code is indicated by rr, and the error code is indicated in hex by cc.

ACTION: Look up the VSAM error code in the VSAM MESSAGES MANUAL. If the problem is not obvious, review the parameters for the file. If everything is correct, run an IDCAMS LISTCAT to see if VSAM is reporting an error. If nothing can be determined, attempt to REPRO the file to a tape or another disk area, then delete and redefine the file and reload it.

If the problem is persistent, call UNICOM Systems Technical Support at 1-818-838-0606.

R0019. INVALID HEXADECIMAL DATA ENTERED

Data specified as hexadecimal (using X'...' delimiters), data was entered that was not valid hexadecimal data.

ACTION: Reenter the data using numbers 0-9 and letters A-F.

R0020. TEMPSTRNG ERROR, DEC RC=xxx

An error return code has been returned by CICS in response to a Temporary Storage GET or PUT command by AUTOMON/BATCH. This could be caused by using Auxiliary Temporary Storage with an insufficient record size or control interval size defined for the Auxiliary file. The return code can be found in the MACRO LEVEL CICS APPLICATION PROGRAMMERS REFERENCE MANUAL under TEMPORARY STORAGE RESPONSE CODES.

ACTION: Determine what the error code means, then if it is not clear what the problem is, call UNICOM Systems Technical Support at 1-818-838-0606.

R0021. ENTER DESIRED FILE NAME

An FSRV command has been entered but the dataset name was omitted

ACTION: Enter the dataset name.

R0028. ENTER BLOCK AND RELATIVE RECORD

R0028. ENTER RBA IN HEX , I.E. X'0050'

R0028. ENTER RECORD KEY

R0028. ENTER RELATIVE RECORD NUMBER

An FSRV command and dataset name have been entered but the record key was omitted. Depending on the type of file involved, you must enter a record key, a record number a relative byte address, or a combination block and relative record number.

ACTION: Enter the desired record key data. Refer to the subject entitled *GENERAL OPERATION* in *Section 8 - THE FILE SERVICE FUNCTIONS OF AUTOMON/BATCH*.

R0029. ENTER SCAN DATA (X'...' IF ENTERED IN HEX)

This message is issued in response to a 'S' command.

ACTION: Enter the data that AUTOMON/BATCH is to search for in this record.

R0030. SCAN DATA NOT FOUND IN THIS RECORD

The data that AUTOMON/BATCH searched for was not contained in this record as it was keyed.

ACTION: No action is required.

R0031. INVALID LENGTH VALUE SPECIFIED

The entered length value is invalid.

ACTION: Re-enter a valid length.

R0032. SECURITY VIOLATION, TRANSACTION TERMINATED

The operator does not have the necessary security clearance to perform the requested command on the designated dataset. This may be due to one of the following types of security checks:

1) The operator security key in the CICS Sign-on Table does not match the security key specification in the Security Control Record for this dataset label.

2) The requested password is not valid for the dataset label.

3) The user-written Security Exit has returned a "not-authorized" code for the designated dataset label.

ACTION: If you should have access to this dataset, consult your System Programmer or review *Section 10 - THE SECURITY CONTROL SYSTEM*.

R0033. INVALID SPECIFICATION FOR BDAM, RRDS, OR ESDS FILE

The specification is not valid for the specified type of file.

ACTION: No action is required.

R0034. UNSUPPORTED DISK DEVICE TYPE

The disk device containing the requested file is not supported by AUTOMON/BATCH.

ACTION: The file should be moved to a disk device that is supported.

R0035. SUPPLY LOCATE SPECIFICATIONS AND PRESS ENTER

This message is displayed in response to an 'L' command.

ACTION: Key all required fields and press ENTER.

R0036. PURGE LOCATE SPECIFICATIONS ? (Y/N)

This message occurs when exiting AUTOMON/BATCH while locate specifications are active.

ACTION: If you intend to use the current locate specifications upon returning to AUTOMON/BATCH, change the field to 'N' and press ENTER. Otherwise simply press ENTER to delete the specification.

R0037. ENTER ON-LINE PRINTER ID

This message is displayed in response to the 'Q' command.

ACTION: Enter the ID of the terminal printer as it is known to CICS.

R0039. RECORD PRINT SUCCESSFULLY STARTED

This message is merely to confirm that the record(s) have been sent to the printer.

ACTION: No action is required.

R0040. INVALID TERMINAL ID REQUESTED

An invalid terminal ID was entered.

ACTION: Enter a terminal ID as it is known to CICS.

R0044. *** WARNING *** AUTOMON/BATCH WILL EXPIRE IN xx DAYS

This message indicates that the end of the trial period for AUTOMON/BATCH is approaching. This message will start appearing within seven days of the expiration date.

ACTION: No action is required at this time; the product is still operational. If it is allowed to expire, however, it can no longer be used. Notify your sales representative at UNICOM Systems.

R0045. YOU MUST BE IN NEW OR CHANGE MODE TO USE THIS COMMAND

The entered command is only valid when in data entry mode.

ACTION: No action is required.

R0046. UPPER CASE TRANSLATION IS [ON | OFF]

In response to the X command, this message verifies the translation setting in the TCT.

ACTION: No action is required.

R0047. RESET UPPER CASE TRANSLATE ? (Y/N)

During the operation of AUTOMON/BATCH, UPPER CASE TRANSLATE for this terminal was turned off, and upon exiting, AUTOMON/BATCH is asking if you wish to turn it back on.

ACTION: If for some reason you want to leave UPPER CASE TRANSLATE off, change the field to 'N' and press ENTER, otherwise simply press ENTER.

R0048. KEYLEN GT 64 BYTES NOT SUPPORTED

Files with key lengths of greater than 64 bytes are not supported by AUTOMON/BATCH.

ACTION: No action is required.

R0049. INVALID BLOCK/RECNO SPECIFIED

The entered block or record number is not valid.

ACTION: Re-enter a valid specification.

R0061. PGM FCTD1010 NOT FOUND OR DISABLED

The FCTD1010 program is either disabled or is not present in the PPT.

ACTION: Enable or install the program properly.

R0062. TRANID FCTD NOT FOUND

The transaction ID 'FCTD' could not be located in the PCT.

ACTION: Ensure that the transaction ID is available in the PCT and that it is enabled.

R0063. FUNCTION NOT ALLOWED FOR REMOTE DATASET

The attempted function is not valid for a dataset that is defined as remote.

ACTION: To perform the function, enter the command from a partition or region in which the desired dataset resides locally.

R0064. STSHELP MODULE NOT AVAILABLE FOR HELP DISPLAY

HELP has been requested; however AUTOMON/BATCH cannot access the STSHELP program.

ACTION: Ensure that STSHELP is enabled and available to CICS.

SECURITY MAINTENANCE MESSAGES ('S' PREFIX)

The following messages (those starting with 'S') pertain to the Security Control maintenance functions described in *Section 10 - SECURITY CONTROL SYSTEM* of this guide. These messages are produced by program FCTD1000.

S1801. KEY DATA FIELDS AND PRESS ENTER

In the New Security Record function, the screen is ready to receive the record data.

ACTION: Enter the desired data in hex or character mode and press ENTER to add the security control record.

S1803. "ENTER" TO DELETE RECORD OR "PF3" TO IGNORE

Security Control Record deletion has been requested and the requested record is displayed prior to deleting it.

ACTION: If deletion is not desired, press PF3. Otherwise, press ENTER to delete the record.

S1805. SECURITY CONTROL RECORD NOT FOUND - xxxxxxx

The requested security control record indicated by the record key replacing xxxxxxx in the message is not present in the data file.

ACTION: Change the record key or dataset name and try the command again.

S1806. END OF FILE REACHED

During forward or backward browse mode, the end (forward) or beginning (backward) of the file has been reached.

ACTION: No action is required. Issue any AUTOMON/BATCH command.

S1807. CURSOR NOT POSITIONED ON VALID DSNAME ENTRY

A delete request has been issued, however it could not be completed because the security control record to be deleted was not specified by tabbing the cursor to that entry before issuing the command.

ACTION: Point to the security control record with the cursor and re-issue the delete request.

S1808. INVALID FUNCTION. PLEASE RE-ENTER

An invalid function has been entered.

ACTION: Re-enter the correct function.

S1809. SECURITY CONTROL RECORD SUCCESSFULLY DELETED

The security record at the cursor was successfully deleted.

ACTION: No action is required.

S1810. MORE THAN 256 RECORDS WITH THE SAME KEY

Because dsnames can be upto 44 bytes in length and the control file is a 10 byte key length, the first 8 bytes of the dsname are used in conjunction with an "S" prefix and a one byte binary suffix. For this reason only 256 security records can be entered with the same starting 8 bytes of dsname.

ACTION: Assign a unique dsname.

S1811. HI-LIGHTED FIELDS ARE IN ERROR

Errors have been detected while editing the input.

ACTION: Correct the hi-lighted fields and press ENTER.

S1813. END OF FILE ENCOUNTERED

While processing the file, End of File was detected.

ACTION: No action is required.

S1815. THIS FUNCTION REQUIRES A DSNAME NAME

Security records must have a dsname.

ACTION: Enter a dsname and reissue the command.

S1816. XXXXXXXXX ALREADY EXISTS

This security record key is already on the file.

ACTION: Assign a new security record key.

S1821. OPERATOR NOT AUTHORIZED FOR SECURITY MAINTENANCE

The correct security maintenance keyword must be entered by the operator.

ACTION: Enter the correct security maintenance keyword.

S1822. GENERIC LABEL SECURITY TABLE IS FULL

The number of generic security table entries entered in the FCTDTBL macro have been exceeded.

ACTION: Increase the entry and re-compile the table.

S1901. NOT AUTHORIZED FOR THIS FILE/FUNCTION

Your security authorization is inadequate for accessing this file or for the function you are trying to perform.

ACTION: Request another function or file.

S1912. I/O ERROR HAS OCCURRED.

An I/O error has occurred while accessing the AUTOMON/BATCH control file.

ACTION: Call UNICOM Systems Technical Support at 1-818-838-0606.

S1914. FILE IS CLOSED TO ON-LINE USERS.

The AUTOMON/BATCH control file has been closed and is unavailable for processing security checks.

ACTION: Open the AUTOMON/BATCH control file using CEMT.

S1918. VSAM ERROR, RC = xx, ERRCODE = x'yy'

CICS has returned the ILLOGIC return code from an I/O request and this is a VSAM file, indicating a VSAM error from a Request Macro.

ACTION: Look up the error in the VSAM Error Messages Manual. xx is the Return Code, yy is the VSAM Error Code.

S1923. USER SECURITY EXIT xxxxxxxx NOT FOUND

Either the global security exit program specified in the SECGBL keyword of the user option table or a local security exit program specified in the security control file could not be found in the library.

ACTION: Install the proper exit program, correct the name reference, or remove the name reference.

S1924. PASSWORD REQUIRED FOR xxxxxxxx

The command being issued for this file definition is password protected in the security control file.

ACTION: Enter the correct password to continue operation.

AUTOMON BACKOUT MESSAGES ('T' PREFIX)

The following messages (those starting with 'T') pertain to the Automatic Backout functions described in *Section 5 - Automatic Open/Close* of this guide. These messages are produced by program FCTD TIME.

T0001. TIMER BACKOUT SUCCESSFULLY STARTED

This message is issued in response to an attempt to start the Auto-backout program, and that Auto-backout was successfully started.

ACTION: No action is required.

T0002. TIMER BACKOUT SUCCESSFULLY STOPPED

This message is issued in response to an attempt to stop the Auto-backout program, and that Auto-backout was successfully stopped.

ACTION: No action is required.

T0004. TIMER BACKOUT NOT CURRENTLY ACTIVE

This message is issued in response to an attempt to stop the Auto-backout program, and that Auto-backout was already inactive.

ACTION: No action is required.

T0005. INTERVAL VALUE MUST BE NUMERIC

This message is issued after an unsuccessful attempt to change the Auto-backout interval if the entered interval contained non-numeric data.

ACTION: Re-attempt changing the interval using the format MMSS where MM is the total number of minutes and SS is the number of seconds.

T0006. DFHIC ERROR... DEC R/C = xxx

An error occurred while attempting to cancel the timer task.

ACTION: RC is the EIBR CODE returned from the HANDLE CONDITION before the cancel.

T0007. TRANID (FCTT) NOT FOUND OR DISABLED

The transaction ID 'FCTT' could not be located in the CICS PCT.

ACTION: Ensure that the transaction ID is available in the PCT and that it is enabled.

T0009. SCREEN RCD NOT FOUND ==> CALL UNICOM SYSTEMS

This message will display if there is a problem with the installation of AUTOMON/BATCH. One of the screen maps that AUTOMON/BATCH uses could not be found in the Dataset Definition File.

ACTION: Re-check the installation for the following possibilities:

1) The Dataset Definition File was not defined correctly either in the IDCAMS definition or in the DFHCD file.

2) The IDCAMS REPRO step did not complete normally.

T0010. ERROR ON IC PUT R/C = xx

An error occurred while attempting to start the timer task.

ACTION: RC is the EIBR CODE returned from the HANDLE CONDITION.

DYNAMIC ALLOCATION MESSAGES: ('X' PREFIX)

For MVS systems, all messages beginning with 'X' pertain to the dynamic allocation/deallocation function. These errors can occur during an OPEN, ACTIVATE, BACKOUT, ALLOCATE or DEALLOCATE command.

Following the 'X' prefix for each message is a four-position hexadecimal code which corresponds directly to the return code from the DYNALLOC Supervisor Call, followed by the message text. If the message text is not self-explanatory, these messages can be found in the MVS SYSTEM MACROS AND FACILITIES reference manual, under the heading SVC 99 RETURN CODES.

If an 'X' series message appears, it means that the allocation or deallocation function which was being attempted could not be completed. You must clear the error condition before attempting the command again.

UNNUMBERED MESSAGES

Certain error messages in AUTOMON/BATCH do not contain a prefixed number. These are editing errors that can occur as input fields are keyed for some input screens. Most of them concern the LOCATE function.

When these messages occur, one or more fields on the screen will be highlighted. The messages correspond to the highlighted fields.

These messages are listed below in alphabetical order. In every case, the action to be taken is to correct the field and re-enter, or press CLEAR to terminate the function.

BASE NAME CAN NOT EQUAL FILE ID

The value entered in the BASE CLUSTER field (MVS only) is the same as the DATASET NAME (DDNAME) of this file.

Operation cannot continue until the field is corrected and ENTER is pressed. If you cannot correct the error at this time, press CLEAR to exit the AUTOMON/BATCH function.

BOTH FIELDS MUST BE SAME LENGTH

The Low/High limits format is being used for the COMPARISON DATA on the Input Selection Specifications screen. Both the fields (on either side of the slash) must be the same number of characters.

BOTH FIELDS MUST BE SAME TYPE

The Low/High limits format is being used for the COMPARISON DATA on the Input Selection Specifications screen. Both the fields (on either side of the slash) must be the same type. That is, they must both be a Character constant, Delimited character constant, Hexadecimal data or @FIELD specification.

CANNOT = INPUT FILENAME

An output file name has been entered on the Output Dispositions screen which is the same as the input file name.

CANNOT ADD TO BDAM FILE

Adding records to a BDAM file is invalid.

CANNOT DELETE FROM BDAM FILE

Deleting records from a BDAM file is not valid.

CLOSE TOD MUST BE 0000 TO 2359

The CLOSE TIME OF DAY field on the Definition screen must contain a valid time.

FIELD LENGTH REQUIRED

Data must be supplied for FIELD LENGTH for this entry.

FILE IS NOT VARIABLE LENGTH

The record length change options on the Output Dispositions screen can only be used for variable length files.

FILENAME DOES NOT EXIST

The specified output file name could not be located in CICS.

FUNCTION NOT SUPPORTED FOR VAR/BLK BDAM

This function is not supported for variable blocked BDAM files.

INPUT POS ENTERED

You cannot enter both COMPARISON DATA and INPUT POSITION on one line of the Input Selection Specifications screen. Choose one or the other.

INVALID

The highlighted data is an invalid value for the field.

INVALID BLKSIZE/RECSIZE

The value entered in the RECORD SIZE field must divide into the value entered in the BLOCK SIZE field evenly.

INVALID DESTID

The destination ID is invalid.

INVALID HEX

The entered hexadecimal data is invalid. Valid hex digits are 0-9 and A-F.

INVALID KEYLEN

The key length is invalid.

NO OUTPUT OPTION SELECTED

On the Output Dispositions screen, one of the output options must be selected by keying a 'Y' in the field to the left of the option description.

NUMERIC

Non-numeric data was entered for a numeric field.

ONLY ONE OUTPUT OPTION CAN BE SELECTED

On the Output Dispositions screen, more than one of the output options has been selected. Only one option can be chosen.

OPTION NOT 'Y'

On the Output Dispositions screen, one of the output options contains a character other than 'Y'.

POSITION EXCEEDS RECORD LENGTH

The specified POSITION field is greater than the maximum record length for this file.

POSITION + LENGTH EXCEEDS RECORD LENGTH

The specified POSITION field plus the corresponding LENGTH will exceed the maximum record length for this file.

PRINTER ID DOES NOT EXIST

The specified printer ID on the Output Dispositions screen can not be located in CICS.

RANGE ERR

The value (usually a number) is outside the range of acceptable values for this field.

REQUIRED

The highlighted field is a required entry for this function.

RSL MUST BE 00-24 OR "PU"

The Resource Security Level must be left blank or contain one of the following values:

00 = NO ACCESS
nn = NUMBER FROM 1-24
PU = PUBLIC (CICS 1.6 and above)

STRNOG EXCEEDS STRNO

For MVS users, the value entered in GET-ONLY STRINGS (STRNOG) must be less than the value in MAXIMUM STRINGS (STRNO).

SYNTAX ERROR FOR LOW-HIGH INPUT

The Low/High limit data in the COMPARISON DATA field is entered incorrectly.

TABLE ERR

The value in the field is not one of the acceptable values for this field.

TOO LONG

The highlighted numeric value is too long for the requested function. Numeric fields in the COMPARISON DATA can not exceed 16 bytes, and the FIELD LENGTH of the Modify/Reformat specifications can not exceed 35. The value entered in DATASET NAME can not exceed 7 characters for DOS/VSE users.

UNPAIRED QUOTE FOR X' MISSING

For hexadecimal specification, the ending quote is omitted.

APPENDIX A SUMMARY OF AUTOMON/BATCH COMMANDS

The following table describes the valid commands for the FILE CONTROL transaction. The alphabetic command code, its PF key equivalent and the description of the command are listed in alphabetic command code sequence.

<u>ALPHA</u> <u>COMMAND</u>	<u>PF</u> <u>KEY</u>	<u>DESCRIPTION</u>
ALT	PF13	Alter a file definition.
ALL	PF9	Allocate a file.
ACT	PF5	Activate a file.
BACK	PF9	Backout a file.
DEL	PF11	Delete a File Definition.
DEF	PF14	Define a new file definition.
DIRE	PF16	Display a directory of all files and their group names.
EXIT	PF3	Return to the previous logical level.
H	PF1	Display a Help screen.
*INQ	none	Inquire (display) a File Definition.
INST	PF15	Install a file or group.
M	PF12	Transfer to CEMT to display a AUTOMON/BATCH file as it exists in the CICS FCT.
*NEW	PF10	Add a New File Definition.
PREV	PF7	Browse backward in the Dataset Definition file.
STRTBCH	none	Activate the Batch Interface
STOPBCH	none	Deactivate the Batch Interface
UNHOOK	none	Deactivate the AUTOMON/BATCH product
†VARY	none	Vary the Service Request Values of a file.
VIEW	PF2	View a file definition.
Z	none	Special-purpose commands.

* These commands may be entered using the 'ALL' or generic dataset name, in addition to a single dataset name.

† This command is only valid through the Batch Interface.

The following table describes the valid commands for the FILE SERVICE transaction. The alphabetic command code, its PF key equivalent and the description of the command are listed in alphabetic command code sequence.

ALPHA COMMAND	PF KEY	DESCRIPTION
B	PF7	Browse backward through the file.
C	PF9	Change (modify) a data record.
D	PF11	Delete a data record.
F	PF8	Browse forward through the file.
H	PF1	Display a Help screen.
I	PF17	Inquire (display) a data record.
L	PF13	Locate and optionally modify data records.
N	PF10	Add a New record to the file.
P	PF4	Page through a large record.
Q	PF15	Print a data record on an on-line printer.
R	PF3	Return to the previous logical level.
S	PF12	Scan a data record for a hex or character string.
T	PF5	Toggle between hexadecimal and character mode.
U	PF2	Scroll the record display up one line.
W	PF6	Scroll the record display down one line.
X	PF14	Toggle the uppercase translate off and on for this terminal.

APPENDIX B SYSTEM REQUIREMENTS

This appendix describes the system environment in which AUTOMON/BATCH will operate, the program modules involved, the CICS table entries required and the fixed and dynamic storage utilization involved.

OPERATING SYSTEM:

MVS, CICS/ESA and above

TERMINAL TYPES:

Any terminal using IBM 3270 data-stream technology. This includes personal computers accessing the mainframe via the IRMA board or some other interface. The terminal may be in a local or remote environment. Both BTAM and VTAM are supported.

PROGRAM MODULES:

Core-Image/Load modules - FCTD1000, FCTDTBL, FCTDBCHT,
FSRV1000, STS0100,
STSHELP, STSCORE

Relocatable modules - none

Source modules - FCTDTBL macro, copybooks FCTDPCT, FCTDPPT, FCTDOPT and
FCTDREC

FILES: VSAM file FCT\$FIL

CICS TABLE ENTRIES:

PCT - Five entries: FCTD, FCTT, FSRV, FSPR, and STSC transactions.

PPT - Seven required entries: FCTD1000, FSRV1000, FCTDTBL, STS0100,
STSPASS, STSHELP, STSCORE

CSD - One entry - FCT\$FIL

PLT - One entry - FCTD1000 (optional)

STORAGE REQUIREMENTS:

Resident CICS storage -
FCTDTBL - 300 bytes + 11 bytes per generic dataset label security record
16 bytes per file entry in TOTFILS= option table entry.

Dynamic storage - Approximately 4K per FCTD transaction.

VSAM file space - UNICOM screen display records plus user's family and security records. (One cylinder on a 3380 would be exceptionally large).

APPENDIX C FILE DEFINITION RECORD FORMAT

There are multiple record types in the FCT\$FIL control file. Each record contains an identifier in the first byte. These are:

- G - File family record
- H - Transaction/program family record
- K - Security record
- S - Screen mapping record
- x'00' - Batch interface control record

When browsing the file sequentially, a change in record type code indicates end-of-file for the previous record type.

The first 10 bytes are the VSAM record key.

APPENDIX D INSTALLATION DOCUMENTATION

* SYSTEM REQUIREMENTS *

OPERATING SYSTEM: MVS, CICS/ESA and above.

TERMINAL TYPES: ANY TERMINAL USING IBM 3270 DATA-STREAM TECHNOLOGY. THIS INCLUDES PC'S ACCESSING THE MAINFRAME VIA THE IRMA BOARD OR SOME OTHER INTERFACE. THE TERMINAL MAY BE IN A LOCAL OR REMOTE ENVIRONMENT. BTAM AND VTAM ARE SUPPORTED.

PROGRAM MODULES: FCTD1000, FCTDTBL, FCTDBCHT, FCTDUTIL, FSRV1000, STS0100, STSHELP, STSPASS, STSCORE

SOURCE MODULES: E.FCTDTBL, A.FCTDPCT, A.FCTDPPT, A.FCTDREC, A.FCTDOPT

FILES: 1 VSAM FILE (FCT\$FIL)

* CICS REQUIREMENTS *

TABLE ENTRIES: PCT -FCTD, FCTT, FSRV, FSPR, STSC

PPT - FCTD1000, FSRV1000, FCTDTIME, FCTDCEMT, FCTDTBL, STS0100, STSPASS, STSHELP, STSCORE

CSD - FCT\$FIL

PLT - FCTD1000

STORAGE REQUIREMENTS: RESIDENT CICS STORAGE - 16 BYTES PER file FILE defined in CICS.

DYNAMIC STORAGE: APPROX 4K PER FCTD TRANSACTION

TEMPORARY STORAGE: APPROX 4K WHILE FCTD TRANSACTION IS ACTIVE, DELETED WHEN THE TRANSACTION ENDS

INSTALLATION STEPS:

* PREPARATION OF THE AUTOMON/BATCH INSTALLATION JCL *

USE WHATEVER MEANS ARE PROVIDED BY YOUR SYSTEM TO GET THE PUNCHED JCL MEMBER FROM THE INITIAL LINK/PRINT/PUNCH PROCESS. THIS IS THE OUTPUT FROM THE JCL KEYED FROM THE UNICOM SYSTEMS INSTALLATION COVER LETTER.

DOS USERS WHO HAVE A FACILITY SUCH AS "GETP" IN ICCF FOR RETRIEVING MEMBERS FROM THE PUNCH QUEUE CAN MAKE USE OF THE JCL=INSTALL OR JCL=REINSTALL PARAMETER.

MVS USERS CAN PUNCH THE INSTALLATION JCL INTO A PDS MEMBER BY SUPPLYING THE PDS NAME, MEMBER NAME, AND BLOCKSIZE OF THE PDS IN STEP 2 OF THE INSTALLATION COVER LETTER.

IF FOR SOME REASON YOU CANNOT OR DO NOT WANT TO USE THE INSTALLATION JCL FROM STEP 2 OF THE INSTALLATION INSTRUCTION, USE THE PRINTED OUTPUT FROM STEP 2 TO CREATE THE JCL IN ANY FASHION DESIRED.

AFTER CREATING THE JCL, BY WHAT EVER MEANS, MODIFY IT TO FIT YOUR SYSTEM'S SPECIFIC REQUIREMENTS. THE OPTIONAL ENTRIES ARE NOTED AND FILLED WITH QUESTION MARKS. WHEN THE MODIFIED JCL IS READY, SUBMIT IT ALONG WITH THE NECESSARY TAPE MOUNTS. MODIFY THE CICS TABLES AS INSTRUCTED AND AUTOMON/BATCH WILL BE READY FOR EXECUTION.

```
// JOB STSINST                                PRINT INSTRUCTIONS AND PUNCH JCL
* *****
* STEP 1. - CREATE AUTOMON/BATCH INSTALL TAPE *
* *****
// ASSGN SYS011,???                          INPUT MASTER PRODUCT TAPE
// LIBDEF *,SEARCH=L.S                        * NOTE 1 *
// EXEC STSINST
PRODUCT=ABATCH
MODE=CREATE                                (CREATE A PRODUCT INSTALL TAPE)
OPSYS=DOS                                  (DOS, MVS, MVS/SP)
CICS=?.?                                    (1.6, 1.7, 2.1, ...)
LINES=56                                    (CAN BE FROM 1 TO 99)
/*

* *****
* STEP 2. - DEFINE AUTOMON/BATCH VSAM FILE *
* *****
// EXEC IDCAMS,SIZE=AUTO
DEFINE CLUSTER                               -
(NAME(ABATCH.VSAM.FILE)                     -
RECSZ(250 7200)                             -
KEYS(10 0)                                   -
SHR(4)                                       -
VOLUMES(?????)                             -
(NAME(ABATCH.VSAM.FILE.DATA)                -
CISZ(8192),CYL(5 1))                        -
INDEX                                       -
(NAME(ABATCH.VSAM.FILE.INDX)                -
CISZ(512),TRK(2 1))                         -
CATALOG(???????)
/*

* *****
* STEP 3. - LNKEDT THE AUTOMON/BATCH PHASES *
* *****
// ASSGN SYSIPT,???                          => OUTPUT TAPE FROM STEP 1 * NOTE 3 *
// MTC REW,SYSIPT
// LIBDEF PHASE,CATALOG=L.S                  * NOTE 1 *
// OPTION CATAL
INCLUDE
```

// EXEC LNKEDT

* *****
* STEP 4. - LOAD THE AUTOMON/BATCH MACROS *
* *****

// MTC REW,SYSIPT
// MTC FSF,SYSIPT,1
// EXEC LIBR,PARM='A S=L.S'
/*
// RESET SYSIPT

* NOTE 4 *

* *****
* STEP 5. - LOAD THE AUTOMON/BATCH VSAM FILE *
* *****

// ASSGN SYS011,??? => OUTPUT TAPE FROM STEP 1
// DLBL FCT\$FIL,'ABATCH.VSAM.FILE',,VSAM
// LIBDEF *,SEARCH=L.S
// EXEC STSINST
MODE=INSTALL|REINSTALL
PRODUCT=ABATCH
FSRVFILE=FCT\$FIL|BYPASS
/*

* NOTE 3 *

* NOTE 5 *

* NOTE 1 *

* NOTE 6 *

* NOTE 7 *

* JCL NOTE EXPLANATIONS *

- NOTE
- 1) DEPENDING ON YOUR DOS SYSTEM :
L.S = LIBRARY.SUB LIBRARY TO CONTAIN THE MODULES.
CL = CORE IMAGE LIBRARY TO CONTAIN THE MODULES.
 - 2) IF THE BATCH INTERFACE IS NOT TO BE USED, SHR(2) MAY BE USED. IN VM ENVIRONMENT , IF FILE IS SHARED ACROSS GUEST MACHINES, THEN SET TO SHR(3). (SEE THE BATCH INTERFACE TO AUTOMON/BATCH FOR DETAILS)
 - 3) SYSIPT SHOULD BE ASSIGNED TO THE TAPE DRIVE THAT THE PRODUCT INSTALL TAPE IS MOUNTED. (OUTPUT FROM THE "CREATE" RUN).
 - 4) DEPENDING ON YOUR DOS SYSTEM :
L.S = LIBRARY.SUB LIBRARY TO CONTAIN SOURCE MODULES.
SL = SOURCE LIBRARY TO CONTAIN THE SOURCE MODULES.
 - 5) SHOULD BE THE CLUSTER NAME USED IN THE DEFINE STEP. CODE CLUSTER NAME ACCORDING TO USER STANDARDS.
 - 6) MODE=INSTALL FOR A NEW INSTALLATION OF THE AUTOMON/BATCH PRODUCT. VSAM FILE SHOULD BE EMPTY. MODE=REINSTALL FOR THE RE-INSTALLATION OF THE AUTOMON/BATCH PRODUCT.
 - 7) FSRVFILE=FCT\$FIL - MUST BE THE 7-BYTE VSAM FILENAME.
FSRVFILE=BYPASS - WILL BYPASS THE FILE.

//STSINST JOB 1,UNICOM-SYSTEMS,MSGCLASS=X,CLASS=A
/*


```

//*          CREATE THE AUTOMON/BATCH PRODUCT INSTALL TAPE
//*
//STEP1      EXEC  PGM=STSINST
//SYSPRINT   DD  SYSOUT=*
//STEPLIB    DD  DSN=????????,          * NOTE 1 *
//           DISP=SHR
//MPRDIN     DD  DSN=MASTER.TAPE,UNIT=TAPE,      * NOTE 2 *
//           VOL=SER=MASTER,LABEL=(2,NL),
//           DCB=BLKSIZE=32000,DISP=SHR
//MOBJOT     DD  DSN=FSRV.LIB,UNIT=TAPE,          * NOTE 3 *
//           VOL=(,RETAIN,,,SER=FSRV),LABEL=(1,NL)
//MMACOT     DD  DSN=FSRV.LIB,UNIT=TAPE,          * NOTE 3 *
//           VOL=(,RETAIN,,,SER=FSRV),LABEL=(2,NL)
//MTXTOT     DD  DSN=FSRV.LIB,UNIT=TAPE,          * NOTE 3 *
//           VOL=(,RETAIN,,,SER=FSRV),LABEL=(3,NL)
//SYSIN      DD  *
PRODUCT=ABATCH      (PRODUCT=ABATCH)
MODE=CREATE          (CREATE A PRODUCT INSTALL TAPE)
OPSYS=MVS            (OPERATING SYSTEM)
CICS=?.?            (1.6, 1.7, 2.1, ...)
LINES=56             (CAN BE 1 TO 99)
/*
/*
/*          LINK EDIT THE AUTOMON/BATCH MODULES
/*
//STEP2      EXEC  PGM=IEWL,PARM='LIST,LET,XREF'
//SYSPRINT   DD  SYSOUT=*
//SYSLIB     DD  DSN=????????,          * NOTE 4 *
//           DISP=SHR
//SYSLIN     DD  DSN=FSRV.LIB,UNIT=TAPE,          * NOTE 5 *
//           VOL=(,RETAIN,,,SER=FSRV),LABEL=(1,NL),
//           DCB=BLKSIZE=3200,DISP=(SHR,PASS)
//SYSUT1     DD  UNIT=SYSDA,SPACE=(1024,(20,20))
//SYSLMOD    DD  DSN=????????????,          * NOTE 6 *
/*
/*
/*          CATALOG THE MACROS
/*
//STEP3      EXEC  PGM=IEBUPDTE
//SYSPRINT   DD  SYSOUT=*
//SYSUT1     DD  DSN=????????????,          * NOTE 7 *
//           DISP=SHR
//SYSUT2     DD  DSN=????????????,          * NOTE 7 *
//           DISP=SHR
//SYSIN      DD  DSN=FSRV.LIB,UNIT=TAPE,          * NOTE 5 *
//           VOL=(,RETAIN,,,SER=FSRV),LABEL=(2,NL),
//           DCB=BLKSIZE=3200,DISP=(SHR,PASS)
/*
/*
/*          DEFINE THE AUTOMON/BATCH VSAM FILE
/*
//STEP4      EXEC  PGM=IDCAMS
//SYSPRINT   DD  SYSOUT=*
//SYSIN      DD  *
DEFINE CLUSTER

```

```

                                (NAME(ABATCH.VSAM.FILE)
                                KEYS(10 0)
                                RECSZ(250 7200)
                                SHR(4,3)
                                VOL(??????)
                                DATA(NAME(ABATCH.VSAM.FILE.DATA)
                                CISZ(8192)
                                CYL(5 1))
                                INDEX(NAME(ABATCH.VSAM.FILE.INDEX)
                                CISZ(512)
                                TRACKS(2 1))
/*
//*
//*                                LOAD ABATCH SCREEN CONTROL RECORDS
//*
//STEP5                                EXEC PGM=STSINST
//SYSPRINT                            DD SYSOUT=*
//STEPLIB                            DD DSN=???????,
//                                DISP=SHR
//FCT$FIL                            DD DSN=ABATCH.VSAM.FILE,
//                                DISP=SHR
//MPRDIN                            DD DSN=FSRV.LIB,UNIT=TAPE,
//                                VOL=(,RETAIN,,,SER=FSRV),LABEL=(3,NL),
//                                DCB=BLKSIZE=3200,DISP=(SHR,PASS)
//SYSIN                                DD *
PRODUCT=ABATCH                                (PRODUCT=ABATCH)
MODE=INSTALL|REINSTALL                        (CREATE FSRV FILES)
OPSYS=MVS                                    (OPERATING SYSTEM)
LINES=56                                      (CAN BE 1-99)
FSRVFILE=FCT$FIL
/*

```

```

*****
*                                JCL NOTE EXPLANATIONS                                *
*****

```

- NOTE
- 1) MUST BE THE LIBRARY THAT "STSINST" PHASE WAS LINK EDITED TO IN THE INITIAL JCL PROCESS FROM THE INSTALLATION GUIDE.
 - 2) THIS TAPE DRIVE SHOULD HAVE THE MOUNTED PRODUCT MASTER TAPE RECEIVED FROM UNICOM SYSTEMS.
 - 3) THIS TAPE DRIVE SHOULD BE A SCRATCH TAPE. THE AUTOMON/BATCH INSTALLATION FILES WILL BE WRITTEN TO THIS DRIVE. IT THEN BECOMES THE INPUT FOR THE REMAINDER OF THE INSTALL PROCESS.
 - 4) THIS IS THE CICS LIBRARY WHICH WOULD BE USED TO LINK EDIT A COMMAND LEVEL ASSEMBLER PROGRAM. IT MUST CONTAIN THE COMMAND LEVEL INTERFACE MODULE "DFHEAI".
 - 5) THIS TAPE DRIVE SHOULD HAVE THE MOUNTED AUTOMON/BATCH INSTALLATION TAPE. (OUTPUT FROM STEP 1)
 - 6) THIS IS THE LOAD LIBRARY WHERE CICS PROGRAMS ARE KEPT. IT MUST BE IN THE CONCATENATION FOR DFHRPL IN THE CICS STARTUP JCL.

- 7) SYSUT1 AND SYSUT2 SHOULD POINT TO A SOURCE PDS FOR MACROS. IF THE MACROS ALREADY EXIST, YOU SHOULD DELETE THEM BEFORE THE INSTALL/REINSTALL PROCESS.
- 8) THIS IS THE CLUSTER NAME USED IN DEFINING THE AUTOMON/BATCH VSAM FILE.
- 9) ASSIGN THE VOLUME THAT THE AUTOMON/BATCH VSAM FILE IS TO RESIDE.
- 10) MODE=INSTALL MEANS YOU ARE INSTALLING AUTOMON/BATCH FOR THE FIRST TIME OR YOU ARE INITIALIZING THE AUTOMON/BATCH VSAM FILE AND LOADING THE SCREEN CONTROL RECORDS. THE FILE SHOULD BE EMPTY.

MODE=REINSTALL MEANS YOU ARE REINSTALLING THE AUTOMON/BATCH PRODUCT. REINSTALL WILL REPLACE ANY EXISTING SCREEN RCDS BUT WILL LEAVE ANY EXISTING DATA RCDS IN TACT.

```

*****
*           F C P C T           *
*   AUTOMON/BATCH PCT TABLE ENTRIES   *
*****

```

```

FCTD  DFHPCT TYPE=ENTRY,TRANSID=FCTD,           X
      PROGRAM=FCTD1000,TWASIZE=1500
FSRV  DFHPCT TYPE=ENTRY,TRANSID=FSRV,           X
      PROGRAM=FSRV1000,TWASIZE=1500
FCTT  DFHPCT TYPE=ENTRY,TRANSID=FCTT,           X
      PROGRAM=FCTDIME,TWASIZE=1500
FSPR  DFHPCT TYPE=ENTRY,TRANSID=FSPR,           X
      PROGRAM=FSRV1000,TWASIZE=1500
STSC  DFHPCT TYPE=ENTRY,TRANSID=STSC,           X
      PROGRAM=STSCORE,TWASIZE=0

```

```

*****
*           F C P P T           *
*****

```

```

FCTD1000 DFHPPT TYPE=ENTRY,PROGRAM=FCTD1000
FCTD1010 DFHPPT TYPE=ENTRY,PROGRAM=FCTD1010
FSRV1000 DFHPPT TYPE=ENTRY,PROGRAM=FSRV1000
FCTDIME DFHPPT TYPE=ENTRY,PROGRAM=FCTDIME
FCTDCENT DFHPPT TYPE=ENTRY,PROGRAM=FCTDCENT
DFHPLTXX DFHPPT TYPE=ENTRY,PROGRAM=DFHPLTXX      * NOTE 1 *
FCTDTBL DFHPPT TYPE=ENTRY,PROGRAM=FCTDTBL      * NOTE 2 *
STS0100 DFHPPT TYPE=ENTRY,PROGRAM=STS0100
STSPASS DFHPPT TYPE=ENTRY,PROGRAM=STSPASS
STSHLP DFHPPT TYPE=ENTRY,PROGRAM=STSHLP
STSHLP DFHPPT TYPE=ENTRY,PROGRAM=STSCORE

```

```

*****
*           F C F C T           *
*   FILE CONTROL TABLE ENTRY   *
*****

```

```

*           F C T D   C O N T R O L   F I L E           *
*           C I C S   R E L E A S E   1 . 6   U S E R S           *
* NOTE 1 : DO NOT PUT FCT$FIL IN LSR POOL.           *
*           *           *           *           *           *
* NOTE 2 : KEY VALUE ON ACCMETH MAY BE OMITTED IF THE *
*           BATCH INTERFACE FEATURE IS NOT BEING USED. *
*           *           *           *           *           *
*****

```

```

DFHFCT TYPE=DATASET,DATASET=FCT$FIL,           * NOTE 3*           X
      ACCMETH=(VSAM,KSDS,KEY),                   X
      FILSTAT=(ENABLED,OPENED),                   X
      RECFORM=(VARIABLE,BLOCKED),                 X
      SERVREQ=(UPDATE,NEWREC,BROWSE,DELETE),      X
      STRNO=6

```

```

*****
*   FCTD CONTROL FILE                               *
*   CICS REL 1.7 AND ABOVE *
*****

```

```

DFHFCT TYPE=DATASET,DATASET=FCT$FIL,          * NOTE 3*           X
      FILSTAT=(ENABLED,OPENED),                X
      RECFORM=(VARIABLE,BLOCKED),              X
      SERVREQ=(UPDATE,ADD,BROWSE,DELETE),      X
      LSRPOOL=NONE,      X
      STRNO=6

```

```

*****
*   FCTD POOL MACRO                               *
*   NOTE : PLACE POOL MACRO PRIOR TO SHRCTL MACRO IF LSR *
*   (LOCAL SHARED RESOURCES) IS BEING USED.      *
*****

```

```

      SPACE
      FCTDTBL TYPE=POOL,          X
      TOTFILS=50,      TOTAL FILES ACTIVE IN CICS           X
      VSAMFIL=40,      TOTAL VSAM FILES                    X
      BDAMFIL=5,      TOTAL BDAM FILES                     X

```

```

*****
*
*   CICS NOTE EXPLANATIONS
*
*****

```

- 1) IF YOU ALREADY HAVE A STARTUP PROGRAM LIST TABLE IN THE PPT, THIS ENTRY IS NOT NEEDED. IF NOT, THE XX IN THE MODULE NAME IS THE SUFFIX USED IN THE DFHPLT TYPE=INITIAL MACRO.
- 2) THIS ENTRY IS REQUIRED IF YOU DEFINE THE FCTD USER OPTION TABLE, AS DESCRIBED UNDER CUSTOMIZATION. IF YOU USE A SUFFIXED TABLE, THE PROGRAM NAME WILL BE FCTDTBXX ... WHERE XX IS THE SUFFIX .
- 3) FCT\$FIL IS THE DEFAULT FILE NAME USED IN THE USER OPTION TABLE. THIS IS THE FCT FILENAME. (SEE CODING THE USER OPTION TABLE).

APPENDIX E REPORTING PROBLEMS

If a problem is encountered with **AUTOMON/BATCH** which can not be resolved by the information in this document, call UNICOM Systems at (405) 947-8080. You will be referred to a technical support representative who will take the information about the problem and resolve it then and there on the phone, if possible. Should a support person not be available when you call, you will be called back as soon as possible, usually the same day.

If the problem can not be resolved on the phone, the support representative may require that you run certain CICS diagnostic reports and send them to him. This will usually include an Auxiliary Trace of the transaction or series of transactions involved, a coredump if one is available, a listing of the CICS File Control Table and sometimes an IDCAMS PRINT of the Dataset Definition File.

When you call, please have the following information available:

- 1) Your Operating System
- 2) The release of CICS where AUTOMON/BATCH runs.
- 3) The release and PTF level number of AUTOMON/BATCH that you have. (This is displayed on any of the main menus)
- 4) Approximately how long AUTOMON/BATCH has been installed.
- 5) Any message numbers involved in the problem, whether produced by CICS, AUTOMON/BATCH or the Operating System.

UNICOM Systems is committed to effective product support and we will make every effort to resolve your problem as quickly as possible.

APPENDIX F - THE PRODUCT LINE

This appendix briefly describes all of the products currently available from UNICOM Systems, Inc. These products are all available on the master installation tape and may be installed and evaluated if desired.

CICS-WINDOWS

CICS-WINDOWS is our top-of-the-line CICS-based session manager, providing full transaction session management within CICS. In addition, interactive windowing allows up to nine windows on one screen, each capable of operating transactions as if it were in a full screen.

CICS-WINDOWS allows the terminal operator to define their terminal as two, three...up to nine "logical", or "virtual" terminals. This means that a single terminal can operate more than one task at a time. A PF or PA key is used to "toggle" from one virtual terminal to another. When the operators toggles out of a transaction, the transaction screen and all pertinent information to restart the task is saved in off-line storage. When the operator returns, the screen is restored intact and the transaction may be continued as if it were not interrupted.

In addition, it is capable of "interactive windowing". This means that the terminal operator can divide the terminal screen into multiple parts, or windows, displaying a portion of each application task in each window. Operation of each task can continue while in window mode. Windows can be sized for best fit of the application screens, or the screen display can be "panned" left, right, up and down to view additional information. This feature makes it easy for the operator to work in one application while viewing another, thereby eliminating the need for external hard-copy reports or other media.

Two additional features of CICS-WINDOWS are outbound datastream compression, which can provide savings of up to 40 percent for all outbound terminal traffic, whether the terminal is using CICS-WINDOWS or not, and the "session view" feature, which allows an operator at one terminal to view the current display of another terminal in the network.

CICS-WINDOWS offers significant savings in operator time and productivity, due to the elimination of time-consuming transactions which are required to move from one application to another. Most users experience savings on the average of 30 minutes per day, per operator. This translates to thousands of dollars in time savings per day in many installations, where hundreds or even thousands of terminal operators are present.

Three separately licensed options are available with CICS-WINDOWS. These are:

- 1). Menu Generation - Create and maintain on-line menus for controlling application programs.
- 2). Message Broadcasting - Send messages to terminals or users, queue and save messages, display messages in popup windows.
- 3). On-line Help Creation - Build help windows attached to application screens at transaction, screen, or field level.

CICS-JUGGLER

CICS-JUGGLER is our CICS session manager providing full transaction session management within CICS. It contains all of the major features of CICS-WINDOWS except for interactive windowing and outbound data compression.

For users who want session management in CICS but do not care about interactive windowing, this product can be a good fit. It is priced below CICS-WINDOWS, yet still offers support for all CICS environments and provides everything you need to take advantage of transaction session management.

AUTOMON/BATCH

AUTOMON/BATCH is a software tool for the systems programmer in a CICS installation. It enhances some of the basic file-handling features of CICS. It provides the ability to define files to CICS dynamically. That is, a programmer can add an on-line file to the system without bringing the system down, which would normally be required. This means that the operation of all terminal operators can continue undisturbed, whereas it would normally be interrupted for 30 minutes or so each time a new file is defined. In addition, it allows on-line files to be opened and closed from another region, thereby alleviating scheduling bottlenecks in operations. It also provides significant savings in storage utilization within the CICS system.

AUTOMON/BATCH also provides the ability to dynamically access any file in the CICS environment for on-line display or maintenance purposes. This means that a programmer can view or change data records without writing a program, which can result in hours or days of saved time. In addition it offers a powerful file search facility, allowing the formulation of complex boolean queries to extract selected data from a file.

AUTOMON/BATCH provides significant advantage over standard CICS file control services by offering the following features:

- Resource Definition On-line (RDO) for the FCT
- Batch interface to open and close CICS files from a batch region.
- Dynamic allocation and deallocation (both DOS/VSE and MVS).
- Automatic (first-access) file opens for all releases of CICS.
- Automatic file closes (after specified period of inactivity).
- Automatic file closes (at specified time of day).
- Automatic disable/enable of transactions and programs associated with a file.
- Grouping of file names.
- Display, update, add or delete records of any file.
- Powerful file queries to locate and optionally extract, modify or reformat data records.

HELP-WINDOWS

This product addresses the area of on-line documentation. It handles the two most-needed functions in this area, which are:

- a). Full on-line text maintenance. Complete manuals can be created and maintained on-line. Entire manuals or selected subjects may be printed on the system printer or displayed on the terminal in "list" form (as it appears when printed). Automatic page numbering, section titles, running headers and print extraction are supported. In addition, text from other systems can be imported into the HELP-WINDOWS text file.
- b). On-line retrieval of help text from application transactions. Selected sections of a manual, or stand-alone documents may be "keyed" to any application transaction in the on-line environment. A user-defined PF or PA key may be used to retrieve the associated text while operating the transaction. The associated text can be displayed either in full-screen mode or in a "window" on the screen, the position and size of which is defined by the text writer. No modifications to user programs are required to provide on-line help access. Help text can be created for any application - user-written transactions, vendor packages, fourth-generation languages, conversational programs, regardless of the method used by the application for building screens.

HELP-WINDOWS offers a "word-wrap" feature which allows text of any length to fully display in a window of any size. In addition, you can create help text with a user transaction by invoking the transaction screen, then using the cursor to "point" to the fields where help is to be made available. Help text can be associated at the transaction level, the screen level or the field level.

VTAM-WINDOWS

VTAM-WINDOWS is a multiple sessions manager operating at the VTAM level. It has full support for interactive windowing, just like CICS-JUGGLER. With VTAM-WINDOWS, you can toggle and window different VTAM applications, such as CICS, TSO, IMS, etc. A strategic feature of VTAM-WINDOWS is the use of an asynchronous VSAM file for saving screen buffers, which results in a significant reduction in storage overhead over most other VTAM session managers. Because of this feature, VTAM-WINDOWS can run very effectively in DOS/VSE, VSE/SP, VSE/ESA, MVS/SP, MVS/XA and MVS/ESA environments.

In addition to basic VTAM session management and interactive windowing, VTAM-WINDOWS offers the following features and benefits.

- Gate-way processing, provides a single entry-point to the system, with pass-through connection to VTAM applications.
- CICS-JUGGLER interface, automatic activation of CICS-JUGGLER using gate-way.
- Automatic connection to an application in any or all sessions.
- Automatic sign-on to applications.
- Command stream processing, allows a series of transactions to be automatically sent to a VTAM application.
- Cut and paste, provides ability to copy data from one application screen into another.
- Session viewing, lets an operator at one terminal view the current display of any session of another terminal using VTAM-WINDOWS.

- On-line profile changes, allows the operator to alter the choices of PF/PA key usage, window configuration and number of sessions that were set in the customization table.

VTAM-JUGGLER is a multiple sessions manager operating at the VTAM level which offers all of the features of VTAM-WINDOWS except for interactive windowing. It is designed for those users who want sessions management at the VTAM level but can forego the windowing feature in return for a reduced license fee.

VTAM-EXPRESS

VTAM-EXPRESS is a terminal datastream compression product which runs at the VTAM level. Its function is to eliminate, as much as possible, superfluous characters in datastreams that are transmitted to and from the host CPU and the terminals. Since the number of characters in a datastream is the major factor in determining the response time for an on-line transaction, the elimination of extra characters can significantly reduce that time. Most users experience a 55 to 80% reduction in terminal traffic with VTAM-EXPRESS, which results in dramatic savings both in time and cost of the computer installation usage.

With VTAM-EXPRESS you can have datastream compression for any VTAM application. CICS, IMS, TSO, any application system which communicates with terminals using VTAM can benefit. It does not require its own region or partition to operate.

VTAM-EXPRESS performs all three major types of datastream compression, that is:

- Elimination of repetitive duplicate characters.
- Outbound image saving, sending only the data fields which changed on this display.
- Inbound mirroring, removing MDT fields so that they do not re-transmit, unless they are actually modified.

Customization options are available in VTAM-EXPRESS to exclude selected applications or terminals from compression, or control the various types of compression being performed. Statistics can be viewed from an on-line display and/or logged to a disk file for batch reporting.

An additional feature of VTAM-EXPRESS is an enhanced buffer trace, which will capture and print the entire buffer contents of VTAM send and receive requests for a designated terminal. IBM's standard VTAM buffer trace will only show you the first 256 bytes of data.

INDEX

A

ACTIVATE command
for file family 17, 18, 20
from inquiry screen 9
from multiple file display 13
in batch interface 27
in programmer interface 37
addition
for BDAM files 54
in FSRV, supported record types 54
of record with FSRV 52
ALLOCATE command
description of 9
for file family 17, 18, 20
from multiple file display 13
in batch interface 27
in programmer interface 37
ALTER command
description of 10
APPLID keyword
description of 112
AUTBINT keyword
description of 113
AUTBOUT keyword
description of 112
authorized library
requirement for 102
automatic backout
control of 23
controlling 122
AUTOMSG keyword
description of 113

B

BACKOUT command
for file family 18
from multiple file display 13
in batch interface 27
in programmer interface 37
BACKWARD (FSRV)
command
description of 47
batch interface
commands with 27
controlling 33
deactivation of 122

enabling 28
examples of commands 32
programs of 28
re-activation of 122
technical information 36
batch interface statistics screen 33
BCHINTF keyword
description of 113
BCHINTV keyword
description of 113
BCHPLT keyword
description of 113
BDAM files
key entry in FSRV 42
BINTMSG keyword
description of 113
boolean logic (in FSRV) (see **LOCATE** command, input selection)

C

CEMT
new copy command 119
CEMT command
description of 9
CEMTINQ keyword
description of 113
CHANGE (FSRV) command
description of 49
CICS
start-up JCL 103
start-up with suffixed table 120
CICSNAM keyword
description of 113
CLOSE command
for file family 17, 18, 20
from inquiry screen 9
from multiple file display 13
in batch interface 27
in programmer interface 37
CLOSMSG keyword
description of 113
command codes
ACTIVATE 17, 18, 20, 27, 37
ALLOCATE 9, 17, 18, 20, 27, 37

ALTER 10
BACKOUT 17, 18, 20, 27, 37
BACKWARD (FSRV) 47
CEMT 9
CHANGE (FSRV) 49
CLOSE 17, 18, 20, 27, 37
DEALLOCATE 9, 17, 18, 20, 27, 37
DEFINE 10
DELETE 9, 16, 17, 20
DIRECTORY 10
DOWN (FSRV) 47
ENABLE 27, 37
EXIT 9, 11, 17, 20
FORWARD 9, 11, 17, 20
FORWARD (FSRV) 47
INQUIRE 12, 17, 20
INQUIRE (FSRV) 47
INSTALL 10
LOCATE 57
NEW 17, 20
NEXT 9
OPEN 18, 27, 37
PAGE (FSRV) 47
PREVIOUS 9, 11, 12, 17, 20
QUEUE (FSRV) 56
QUIESCE 27, 37
RETURN (FSRV) 48
SCAN (FSRV) 48
START 27
START (FCTT) 23, 122
STATUS 10
STOP 23, 122
STOPBCH 121
STRTBCH 121
TOGGLE (FSRV) 50
TRANSLATE (FSRV) 51
UNHOOK 121
UP (FSRV) 47
VARY (FCTT) 30
VIEW 9, 11
Z 27, 121
commands
file service functions 41
for file families 17, 20
for transaction/program families 19
format for FSRV 40
format of 7
from file family display 17, 20
from inquiry screen 9

- from multiple file display 11, 12
- from transaction/program family screen 20
- in programmer interface 37
- in security file maintenance 94
- special-purpose 121
- summary of for FCTD 161
- summary of for FSRV 162
- conversion utility
 - description of 103

D

- DATA keyword
 - description of 30
- DATASET field
 - in FSRV 41
- DEALLOCATE command
 - description of 9
 - for file family 17, 18, 20
 - from multiple file display 13
 - in batch interface 27
 - in programmer interface 37
- DEFINE command
 - description of 10
- DELETE command
 - description of 9
 - for file family 16, 17, 20
- deletion
 - of file family 16
 - of records in FSRV 55
- DIRECTORY
 - description of 10
- DOWN (FSRV) command
 - description of 47

E

- ENABLE command
 - in batch interface 27
 - in programmer interface 37
- error messages
 - automatic backout 155
 - dynamic allocation 157
 - locate 140, 143
 - operational 128
 - security maintenance 152
 - un-numbered 157
- EXIT command
 - description of 9, 11

- for file family 17, 20
- exits (see user exit)

F

- family definitions
 - explanation of 15
- file families
 - commands available 17, 20
 - description of 17
 - display of 16
 - transaction/program
 - command format 19
 - commands available 19
 - description of 19
- FCT keyword
 - description of 113
- FCT\$FIL
 - SHR(4) requirement 28
- FCTDTBL
 - coding format 112
 - default values of 115
 - installation of 115
 - keywords of (see keywords)
 - skeleton version 112
 - suffixing of 120
- FCTT transaction
 - commands of 23
- file definition
 - duplication of 8
- file families
 - description of 17
- file maintenance (see file service functions)
- file service functions
 - description of 39
- AUTOMON/BATCH
 - benefits of 2
 - customization of 111
 - deactivation of 121
 - limitations of 119
 - programs of 163
 - re-activation of 121
 - security with 83
 - storage requirements 163
 - transaction operation 3
- FORWARD (FSRV) command
 - description of 47
- FORWARD command
 - description of 9, 11
 - for file family 17, 20
- FSRV transaction
 - adding a record 52

- data display screen 44
- displaying a record 47
- locate function 57
- modifying a record 49
- opening menu 40
- record deletion 55
- record printing 56
- record types supported 54

H

- help displays
 - invocation of 5

I

- INQUIRE (FSRV) command
 - description of 47
- INQUIRE command
 - for file family 17, 20
 - for file family 16
 - from multiple file display 12
- INSTALL command
 - description of 10
 - installation
 - DOS/VSE 98
 - MVS 100
 - of other products 97
 - special notes 102
 - summary 104
 - tape contents 96
 - installation documentation 167

J

- job control
 - load install program
 - DOS/VSE 98
 - MVS 100
 - product password install 108
 - running the install program
 - DOS/VSE 99
 - MVS 101

K

keywords
 APPLID 112
 AUTOBINT 113
 AUTOBOUT 112
 AUTOMSG 113
 BCHINTF 113
 BCHINTV 113
 BCHPLT 113
 BINTMSG 113
 CEMTINQ 113
 CICSNAM 113
 CLOSMSG 113
 DATA 30
 DIRECTORY 105
 FCT 113
 FSRVFILE 105
 HELPSCRN 105
 HELPTXT 105
 JCL 106
 JUGLFILE 106
 LINES 106
 MODE 106
 OPENMSG 114
 OPSYS 106
 RELEASE 114
 SECGBL 114
 SECGEN 114
 SECPSW 114
 Keywords (*cont'd*)
 SECURITY 114
 SUFFIX 114
 TERM 30
 VARY 114
 VTWOFILE 107
 WNDOWFILE 107

L

labels
 partition standard 119
 local shared resources (see shared resources)
 LOCATE command
 description of 57
 execution of
 different file types 81
 error conditions 79
 initiating 78
 pause options 80
 terminating and restarting 82
 initial entry of 57
 input selection

comparison data 59
 comparison operator 59
 examples of 61
 field connector 60
 field length 58
 notes on 61
 record position 58
 screen display 58
 modify/reformat
 addition entry of 77
 changing record length 77
 clear fields option 77
 constant data 76
 field length 76
 input position 76
 output position 76
 screen display 75
 output disposition
 add to another file 67, 69
 display and add 72
 display and delete 73
 display and rewrite 70
 handling duplicates 68, 69, 72
 modify and add 71
 modify and rewrite 70
 record deletion 73
 record display 67
 record printing 74
 screen display 66
 LOG option in FCT
 restriction of 102
 lower case letters
 use of in FSRV 50

M

messages
 automatic backout 155
 dynamic allocation 157
 locate 140, 143
 operational 128
 security maintenance 152
 un-numbered 157
 modification
 of file families 16

N

NEW command

for file family 17, 20
 NEWCOPY command
 for AUTOMON/BATCH programs 119
 NEXT command
 description of 9

O

OPEN command
 error codes 13, 18
 for file family 18
 from inquiry screen 9
 from multiple file display 13
 in batch interface 27
 in programmer interface 37
 OPENMSG keyword
 description of 114

P

PAGE (FSRV) command
 description of 47
 PARM statement
 in FCTDBCHT 120
 PARSTD area
 clearing 119
 considerations for 119
 password
 for product
 description of 108
 fields of 110
 installation of
 DOS/VSE 108
 MVS 109
 use of 84
 PF keys
 at file family display 17, 20
 at multiple file display 11, 12
 at security control display 94
 PLTPI
 definition of 103
 PREVIOUS command
 description of 9, 11, 12
 for file family 17, 20
 printing
 of records in FSRV 56
 programmer interface
 examples of 38
 rules for 37
 programs

FCTDBCHT
JCL for 28

Q

query (in FSRV) (see LOCATE
command, input selection
QUEUE (FSRV) command
description of 56
QUIESCE command
in batch interface 27
in programmer interface 37

R

record key
for BDAM files 43
for ESDS files 43, 54
for KSDS files 43, 54
for RRDS files 43, 54
for SAM files 43
format by file type 43
RELEASE keyword
description of 114
RETURN (FSRV) command
description of 48

S

SAM files
key entry in FSRV 42
SCAN (FSRV) command
description of 48
screen
file service manager menu 3

screen display
family directory 15
FCTT statistics display 24
file 17
file definition display 9
FSRV data record display 44
FSRV opening menu 40
locate function
input selection 58
modify and reformat 75

output disposition 66
new file definition 7
security file maintenance 92
transaction/program family 19
searching
of records in FSRV 57
SECGBL keyword
description of 114
SECGEN keyword
description of 114
SECPSW keyword
description of 114
security
file maintenance 92
in AUTOMON/BATCH 83
methods of controlling
passwords 84
security keys 83
user exit 85
security control
addition 93
deletion 94
update 94
security keys
use of 83
SECURTY keyword
description of 114
shared resources
support for 102
START command
in batch interface 27
START command (batch)
description of 30
START command (FCTT)
description of 23, 122
start-up JCL
for CICS DOS/VSE 103
with suffixed table 120
STATUS command
description of 10
STOP command
description of 23, 122
STOPBCH command
description of 121
STRTBCH command
description of 121
STS0100
link-edit of
DOS/VSE 98

STSINST
link-edit of
DOS/VSE 98
MVS 100
STSPASS

link-edit of
DOS/VSE 98
MVS 100
suffix
on DFHPLT entry 103
on user option table 120
SUFFIX keyword
description of 114

T

TERM keyword
description of 30
TOGGLE (FSRV) command
description of 50
transactions
FCTT
statistics description 24
use of 23, 122
AUTOMON/BATCH
fast-path command 7
use of 3
FSRV
use of 40
user
attaching to terminal 31
starting from batch 30
TRANSLATE (FSRV)
command
description of 51

U

UNHOOK command
description of 121
effect of 121
use of 121
UP (FSRV) command
description of 47
updating
of record with FSRV 49
user exit
examples of 88
installing 87
security 85
user option table
default values of 115
format of 112
installation of 115
skeleton version 112
suffixing of 120

V

- V command
 - description of 30
- VARY command
 - description of 30
- VARY keyword
 - description of 114
- VIEW command
 - description of 9, 11
- VSAM local shared resources
 - (see shared resources)

Z

- Z command
 - description of 121
 - in batch interface 27