**Field
Developed
Program**

Program Description/
Operations Manual

OS DITTO Extensions

Program Number: 5798-CDA

This manual describes the capabilities
of the system and the programs. Dis-
cussion of design assumptions and
potential modification areas are
included. Record and file layouts are
described and primary processing proce-
dures specified. This manual is both
a system description and an installation
and operations reference document.

IBM

PROGRAM SUPPORT PERIOD

Program support is normally provided for a period of nine (9) months from the date the FDP is initially available from PID. If the Program Support Period is modified, all users will be advised via an Error Correction Letter. Further details are included in each FDP Availability Notice (sales brochure GB21-XXXX) under the heading Program Support Period.

Documentation concerning errors in a licensed program may be submitted to:

```
IBM Corporation
Department 73C
1501 California Avenue
Palo Alto, California 94304
Attn:  Mr. L. S. Brammer
```

Further information concerning Programming Services Period for this FDP can be found in the Availability Notice (GB21-1549).

During this period only, IBM through the program author(s) will, without additional charge, send corrections to the customer reporting the problem and/or issue, through PID, corrected code or notice of availability of corrected code. However, IBM does not guarantee service results or represent or warrant that all errors will be corrected. Any on-site program support or assistance will be provided at a charge.

Program documentation updates, distributed as Technical Newsletters (TNL), are to be inserted into the existing manuals. TNLs for the Program Description/Operations Manual are prefixed SN20; TNLs for the Systems Guide are prefixed LN20.

Machine readable material updates, distributed as Error Correction Letters, include:

1. A cover letter, numbered in ascending sequence, beginning with letter number 001.

2. A description of the problem(s) and solution(s).

3. Keypunch coding sheets or cards that identify changed, added, or deleted records.

WARRANTY

EACH LICENSED PROGRAM IS DISTRIBUTED ON AN 'AS IS' BASIS WITHOUT WARRANTY OF ANY KIND EITHER EXPRESS OR IMPLIED.

Requests for copies of IBM publications should be made to your IBM Representative or to the IBM Branch Office serving your locality.

PROGRAM DESCRIPTION/OPERATIONS MANUAL

TABLE OF CONTENTS

INTRODUCTION

OS/DITTO EXTENSIONS (5798-CDA) is an enhancement to Field Developed Program OS/DITTO (5798-ARD). This PDOM replaces the 5798-ARD PDOM to serve as a single reference manual for all functions. The 5798-ARD PDOM should still be referred to for installation instructions for that program and HASP environment considerations.

In today's operating system environment, greater emphasis is being placed on programming and operations productivity. Reductions in the amount of personnel time required to test applications can provide significant benefits to data processing installations. The OS/DITTO program reduces test time requirements by providing the facility to examine, alter, and modify data files used in both a testing and production environment.

A file built incorrectly during testing can significantly impact programmer and system time. Standard utility control card require accurate specifications of file labels, physical and logical record sizes, and physical device characteristics. If the programmer generates the file incorrectly, the standard utility will not access the file. The programmer is then forced into a series of iterations, specifying various file characteristics, until he finally discovers the actual attributes of the generated file.

OS/DITTO provides the user with the ability to access and display any file on any supported device. The user can determine quickly and accurately the file characteristics, and then initiate commands to provide the necessary data to continue testing.

SYSTEM OVERVIEW

DESCRIPTION

OS/DITTO is a general-purpose utility program containing thirty-seven (37) utility functions for Unit Record, Tape, and Disk Input/Output units. Functional utilities available in addition to normal card, disk and tape utilities are:

Tape and Disk Record Scan, Disk and Tape Record Alteration, Disk ID Volume Number Change, Deblocking of Tape Records when printing, and User Tape Error Correction. OS/DITTO is an OS version of the FDP DOS/DITTO (5798-ARN), which is an enhanced version of the Type III program DITTO-360D-00.1.018.

OS/DITTO EXTENSIONS (5798-CDA) offers 22 additional functions, Device Type macro support for obtaining DASD characteristics; a DB/DC Logtape Close capability, Standard Label tape support for DITTO tape output functions, and an optional TSO terminal interface.

Operating Characteristics are:

- Input/Output areas are allocated based on the user's specification at assembly time.

- Operations may be performed entirely from the system console or function card statements may be job-streamed for remote programmer use.

- All console communication is done by initiating OS cataloged procedures. Device selection may be specified by entering optional keyword parameters on the start command.

Significant Enhancements to the Original Type III Version are:

- Support for 3330 (Model I and II), 3340 Models (35 and 70), and 2305 (Models 1 and 2).

- Support for the 3505 Card Reader and 3525 Card Punch, with additional support for the 3525 punch and interpret feature.

- Five new utility functions including support for printing and automatic deblocking of variable length tape records, and tape copying with reblocking of logical records.

- A new string parameter format for console operation allows multiple parameters to be entered in a single Operator response, thus reducing console message time.

- Control card parameters are now free form and variable length format.

- Use of QSAM for all printed output improves printer performance.

- Documented source code, utilizing expanded register notation, facilitates user modification and customizing if desired.

- New "I/O Device Characteristics" table provides the ability to support additional I/O devices.

- A new documentation function provides a program generated document of all utilities and associated parameter requirements. This reduces the need to reference manuals for operational answers.

Highlights of OS/DITTO EXTENSIONS are:

Twelve SYSTEM UTILITY functions save operator, programmer, and especially system programmer time by avoiding the typical cycle of reading the utilities manual, key-punching JCL and control cards, and iterative machine runs to debug JCL and control card format or logic errors. OS/DITTO SYSTEM UTILITY function code routines prompt for keyword values, build and display the appropriate control cards, ask for authorization to execute (or cancel if desired) and then LINK to the appropriate SYSTEM UTILITY.

UTILITY functions supported are Partitioned Data Set member(s) print or punch, DASD volume Dump or Restore, List available free space on a DASD volume, List the VTOC (all options supported), List a catalog, List the directory of a PDS, Catalog or Uncatalog a Data Set name, Scratch a VTOC or Data Set or Member, Rename a Data Set or Member.

DASD Device characteristics are now obtained from the System Control Program DEVICE TYPE macro. This allows new DASD Device support for DITTO EXCP disk routines to be provided by the SCP instead of hand-coded program changes.

In a DB/DC environment after a system failure occurs, the need
arises to somehow CLOSE the LOG TAPE before REWINDing it and
removing it from the drive. Often the awkward procedure of mani-
pulating the LOG TAPE drives control unit Customer Engineer panel
in order to write a tape mark is attempted by system operators.
This elaborate process is time-consuming and delays restarting
the system. A special DITTOLOG procedure is provided to write a
tape mark (without rewinding) on the LOG TAPE drive after Re-IPL.
The procedure can be integrated into installation DB/DC restart
procedures. This allows faster restarts. Also the END of FILE
indicator on the LOG TAPE avoids data check read errors or reading
off the End of Reel by Log Tape Terminator/Recovery programs
that must read LOG TAPES.

Card to QSAM (re-sequencing is optional), QSAM to Card, QSAM to
printer, and QSAM to QSAM functions are provided. The Device
Independence, default buffering, and multiple Record Format support
features of these QSAM functions offer the advantages of DITTO
support for Standard Label Tape (Input or output), Variable Blocked
Record Formats (DASD or Tape), and buffered Tape to Tape copying.
In addition, a Partitioned Data Set member may be selectively
listed in DITTO print format. A message summarizing the number
of logical records read and written is issued to the user.

VSAM Key-Sequence and Entry-Sequence Data Set selective print
functions are provided in the familiar DITTO format. This
format saves eye strain and confusion by using a vertical character
and hexadecimal orientation in addition to a byte position counter
scale. This allows a user to easily relate a character to its
hexadecimal equivalent and determine its exact byte count position
in the record. The DITTO print format is thus easier to read
than the core dump print formats supplied by system utilities.

The Tape Record Scan and Disk Record Scan functions are executable
in the jobstream (without operator intervention). New control
card parameters and a continuation card capability is provided to
allow programmers to eliminate asking operators to initiate these
functions from the console. Also, more selective jobstream
tape printing and copying can be done by scanning the record desired
before initiating prints and copies.

An optional TSO terminal interface is provided for operating
"TSO/DITTO" in console operator mode from a TSO terminal. This
allows TSO programmers to use DITTO facilities to selectively list
large data sets, alter or scan data sets (in character or hexadecimal),
and use DITTO tape control commands. This interface is an independent
installation option if desired.

OS/DITTO EXTENSIONS source code has been re-structured to allow
easier user additions. 5798-ARD Source Statement Sequence numbers
have not been changed.

The disk address operand has been eliminated from console disk
functions thus reducing the number of required operator replies.
Function code Disk Record Scan in console mode allows record size
and scan-type to be entered as string parameters. Also, the disk
record scanned for is printed when found. These features make
Disk Record Scan faster to invoke and can save issuing a print command.

The documentation printout function has been updated to instruct
the user on the use of the new functions. In addition, the new
control card functions have improved control card error diagnostics.

Card Functions

DITTO provides seven card utility functions. These functions are:

- Card to Punch

- Card to Punch with resequencing for Job Control decks,
  Assembler Language, COBOL, RPG, and FORTRAN Source
  decks.

- Card to Printer

- Card to Printer in character and vertical hexadecimal
  format (for listing object decks, etc.)

- Card to Tape blocked 1 to 400 card images

- Card to Tape with resequencing

- Tape to Card

DITTO also has the ability to recognize /+ as the end of card
input data following a $$DITTO control card in jobstream mode.
This allows the user to submit multiple $$DITTO control card
operations in a given job step. The End-of-File Key is recog-
nized as end of card input during console operation. Entire
jobstreams with embedded /* cards may be listed or reproduced
by using the DATA and DLM parameters on the SYSIN DD card;
e.g., //SYSIN DD DATA, DLM=AA followed by the job stream and an
AA delimiter card. Card punch output may be interpreted if
the user's system is configured with the 3525 punch interpret
feature.

Card input devices and card punch devices supported are the same
as those supported by QSAM. Tape devices supported are the 2400
series, the 2420 series, and the 3420 series tape units. Tape may
also be substituted for a card input device, a card punch device,
or as SYSPRINT.

Tape Functions

DITTO provides thirteen tape utility functions and eight tape control
functions. These functions are:

- Six Tape to Printer functions providing two print
  formats for fixed blocked tape records, fixed
  unblocked tape records, and variable length blocked
  or unblocked logical records.

- Tape Record Load function to alter records resident
  on tape.

- Tape Record Scan function to locate a specified
  logical record on a tape file.

- Initialize Tape.

- Tape to Tape Copy for multiple files and multiple
  output reels.

- Tape to Tape Copy with reblocking of logical records.

- Eight Tape Control functions provide the ability to
  forward or back space past multiple records or files,
  to rewind, and to erase entire data set security tapes.
  The erase tape function is valid for 3420 tape com-
  ponents only.

DITTO additionally provides a tape error correction routine. This
routine will allow the operator to copy a tape with parity or block
length errors. User options are provided to correct records in
error before copying onto an output file.

Disk Functions

DITTO provides ten disk utility functions. These functions are:

- Data Set Extent Display

- Disk Character Print (no deblocking)

- Disk Character Print Deblocked into logical records

- Disk Character and Hexadecimal Print (no deblocking)

- Disk Character and Hexadecimal Print deblocked into
  logical records

- Disk Record Scan

- Disk Print and Record Scan for Split Cylinder file
  organizations

- Write Disk End-of-File Record

- Disk Record Load

- Alter Disk Volume Serial Number

Disk Devices supported are 2311, 2314, 3330 (Models I and II),
3340 (Models 35 and 70), and 2305 (Models 1 and 2). Support for
additional disk devices is facilitated by a "Disk Device Character-
istics Table". This allows the user to provide entries which should,
in most cases, enable support for the desired device.

OS/DITTO EXTENSIONS automatically fill in an additional DASD character-
istics table entry by issuing a DEVICE Tape macro for the DD name
DISKIN. This will make new DASD devices automatically supported
by the System Control Program.

All disk utilities function within a Data Set protected environment and
require Data Definition statements. In console mode of operation,
the user must provide the physical hardware address, the data set
name, and the volume serial number in order to access the data set.
Printed output reflects the physical and logical addresses and
file characteristics of the device.

Multiple extent data sets may be printed from the console or by
control cards. In console mode, use the DSE function to determine
valid, absolute CCCHH (BEGIN and END). When END of Extent is
reached, the next extent is made available until all are exhausted.
In control card mode, CCCHH is relative to the first CCCHH of the
data set. When END of Extent is reached, the next extent is made
available. In both modes the last extent remains available for
subsequent disk functions. The End of Extent message is also issued
for Out of Extent conditions when the user specifies an excessive
BEGIN=CCCHH.

-8-

In control card mode the "BEGIN=CCCHH" will be added to the data set CCCHH extent start address. This allows the user to begin printing at any track within the data set. This also holds true for data sets not allocated on a full cylinder boundary. The HH portion of "BEGIN=CCCHH" is added to the data set HH. If the result exceeds Head Maximum for the device, the CCC portion of "BEGIN=CCCHH" is incremented by 1 and the HH is decremented by Head Maximum.

The description of the new function codes for OS/DITTO EXTENSIONS is as follows:

SYSTEM UTILITY FUNCTIONS

PRT   PRINT ONE OR ALL MEMBERS OF A PDS
PUN   PUNCH ONE OR ALL MEMBERS OF A PDS
DUM   DUMP A DASD VOLUME TO TAPE
RES   RESTORE A DUMP TAPE TO DASD
LS   LIST SPACE AVAILABLE ON A DASD VOLUME
LV   LISTVTOC OF A DASD VOLUME
LC   LIST CATALOG (SYSCTLG - IPLVOL OR CVOL)
LP   LIST PDS DIRECTORY
CAT   CATALOG DSNAME
UNC   UNCATALOG DSNAME
SCR   SCRATCH VTOC, DATA SET, OR MEMBER OF A PDS
REN   RENAME DATA SET OR MEMBER

QSAM (CARD, DASD, AND SL/BLP TAPE) FUNCTIONS

CQ   CARD TO QSAM
CQS   CARD TO QSAM WITH SEQ. NUMBERS AND DECK NAME
QC   QSAM TO CARD (USE FUNC=I FOR 3525 INTERPRET)
QQ   QSAM TO QSAM (COMPATIBLE RECFM'S ASSUMED)

QSAM/VSAM PRINT FUNCTIONS

QP   QSAM TO PRINTER DEBLOCKED IN CHAR. FORMAT
QD   QSAM TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP
VP   VSAM TO PRINTER DEBLOCKED IN CHAR. FORMAT
VD   VSAM TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP

MISCELLANEOUS FUNCTIONS

TRS   CONTROL CARD - TAPE RECORD SCAN
DRS   CONTROL CARD - DISK RECORD SCAN (KEY/DATA/EOF)

---   S DITTOLOG - WRITE TAPE MARK ON DB/DC LOG TAPE WITHOUT
                REWIND AFTER SYSTEM FAILURE (SEE COMMENT CARDS
                IN DITTOLOG PROC FOR OPERATIONAL DETAILS)

SYSTEM CONFIGURATION

The program is designed for any System 360/370 capable of supporting
OS, PCP, MFT, MVT, VS1 and VS2 (R1, R2, and R3). Devices supported
are 2311, 2314, 3330-1, 3330-11, 3340-35, 3340-70, 2305-1 and
2305-2 disk subsystems, 2400, 2420, and 3420 tape subsystems, 1403,
or 3211 printer systems and all QSAM supported card input/output
subsystems.

The card punch and print option requires the 3525 card punch with
either the multiline card print feature (#5272) or the two line
card print feature (#8338).

The minimum S/360 or S/370 configuration required is an OS system
and a 1403 or 3211 printer with at least 132 print positions.

The program as distributed requires a 38K OS partition (the documenta-
tion overlay requires an additional 36K when being executed).

PROGRAMMING SYSTEMS

The program consists of one core resident module (OS/DITTO) and
six OS load modules. Storage required for these load modules is
discussed in the Systems Guide.

OS/DITTO requires a S/360 or S/370 OS or VS system with a minimum
partition size of 26K (when only a 2K disk/tape buffer is specified).
A 74K partition is required to utilize the program generated documenta-
tion function.

The program is written in Assembler Language. It was assembled
under OS/VS Assembler (5741-SC1-03) Release 1.0 and tested under the
following systems:

1. OS/MFT Release 21.0

2. OS/MVT Release 20.7 with HASP

3. OS/VS1 Release 1.0, 2.0, 3,0, and 4.0

4. OS/VS2 Release 1.0, 1.6, and 1.7 (with HASP)

5. OS/MFT Release 21.6

6. OS/VS2 Release 2.0, and 3.0 (with JES2)

PROGRAM DESCRIPTION

For detailed formats and operating tips refer to the "XXX" documentation printout in Appendix A in the back of this manual.

CARD UTILITIES

Card to Card

Two card to punch options are available. CC is an 80/80 reproduction of the input cards. CCS is a card-to-card copy with sequence numbers and deck identification name added. Three decktypes are available.

- DECKTYPE=COB is used to sequence and name COBOL source decks. Sequence numbers will be placed in CC 1-6 and an 8 position deckname placed in CC 73-80.

- DECKTYPE=RPG is used for RPG source decks. Sequence numbers will be placed in CC 1-5 and a 6 position deckname placed in CC 75-80.

- DECKTYPE=BAL is used for Assembler and FORTRAN source decks, Job Control decks, etc. Sequence numbers will be placed in CC 76-80 and a 3 position deckname placed in CC 73-75.

Decknames will be left-justified and padded with blanks if less than the required length. /* Cards may be interspersed in the input data (see Operating Suggestions). Punched output may be automatically interpreted (see Operating Suggestions). Card count statistics will be printed on SYSPRINT for control card operation, and on the Console for console operation.

Card to Printer

Cards may be listed on SYSPRINT in either character or character and vertical hex format. The vertical hex format is useful for listing object or binary decks. Job Control cards including /* statements may be interspersed in the input data (see Operating Suggestions). Card count statistics will be printed on SYSPRINT for control card operation, and on the console for console operation.

Card to Tape

Card image to tape allows the user to specify a tape blocking
factor of 1 to 400. Job control cards, including /* cards, may
be interspersed in the input data (see Operating Suggestions).
A tape mark will be written after the last card block on the
tape file. Card count statistics will be printed on SYSPRINT
for control card operation, and on the console for console
operation.

Card to Tape with Deckname and Sequencing

This function will transfer source cards to tape with a deck-
name and card resequencing added. The function combines the
capabilities of CCS and card to tape. Options are identical to
CCS (see Card to Card). Card count statistics will be printed
on SYSPRINT for control card operation, and on the console for
console operation.

Tape to Card

Tape to card will transfer blocked or unblocked 80 character
logical input records to SYSPUNCH. Leading tape marks are not
bypassed if present. If the input tape record is not a multiple
of 80 bytes or equal to 81 byte records, the function will
terminate. If the tape is unblocked with 81 character records,
DITTO will assume the first data position to be "carriage con-
trol" and will punch positions 2-81 into the output card. Punched
output may be automatically interpreted (see Operating Suggestions).
Card count statistics will be printed on SYSPRINT for control
card operation, and on the console for console operation.

OS/DITTO EXTENSIONS QSAM AND VSAM UTILITIES

Card to QSAM

Card image to QSAM (tape or DASD) provides a way of creating
Standard Label tape and DASD data sets from card input. Two
cataloged procedures are provided. DITTOCQT allocates TAPE and
DITTOCQD allocates SYSDA. If neither RECFM, BLKSIZE, nor LRECL
are specified on the QSAMOUT DCB, the function recognizes this
condition via the READ JFCB macro and forces RECFM=FBS, BLKSIZE=3120,
and LRECL=80. To override these defaults submit all three parameters.
A card count message is issued to the user.

Card to QSAM with Deckname and Sequencing

This function transfers source or data cards to the QSAMOUT DCB
with Deckname and card resequencing added. The Options are
identical to Card to Card with Sequencing. A card count message
is issued to the user.

QSAM to Card

This function transfers 80 byte logical records to SYSPUNCH. QSAM
to Card provides DITTO support for Standard Label tapes and
DASD data sets. The optional STARTAFT and NLRECS parameters allow
selective card punching if desired. A logical records read and
written message is issued to the user. A PDS member can also be
punched by specifying IDSN=PDS (MEMBNAME) on the DITTOQC or
DITTOQCI cataloged procedure.

QSAM to QSAM

This copy function provides a Tape to Tape (buffered), Tape to DASD,
DASD to DASD, and DASD to Tape capability. Compatible RECORD FORMATS
are assumed. For example, RECFM=V to RECFM=VB can be copied, but
not RECFM=F to RECFM=V. Four cataloged procedures are provided
for each combination (DITTOQTT, DITTOQTD, DITTOQDD, and DITTOQDT).
The optional STARTAFT and NLRECS parameters allow selective copying
if desired. A logical records read and written message is issued to
the user.

QSAM to Printer

Two print functions provide a logical record print in either character
or character and vertical hex Format. The optional STARTAFT and
NLRECS parameters allow selective printing if desired. A PDS member
may be printed by specifying IDSN='PDS (MEMBNAME)' on the DITTOQP
cataloged procedure. A logical records read and written message
is issued to the user.

VSAM to Printer

Two print functions provide a logical record print in either character
or character and vertical hex format. The optional STARTAFT or
STARTKEY and NLRECS parameters allow selective printing if desired.
The VSAMIN DD statement is opened as an ESDS and processed as such
unless a TESTCB indicates the presence of a KSDS in which case a
MODCB changes to KSDS processing. The STARTKEY value is left justi-
fied and padded on the right with binary zeros. A Point macro is
issued for the Key value to position the KSDS data Set. A logical
records read and written message is issued to the user.

TAPE UTILITIES

Tape Print

Six printing formats are available for input tapes. Tape data may be
printed in character format only, or in character and vertical hex
format. Fixed length tape records may be printed as they exist or
may be deblocked into logical records. Variable length tape record
formats will be automatically deblocked. Block and record counts
are printed with the associated records. The presence of tape marks
will be indicated on the output listing.

For control card operation, and optional NBLKS=nnnn parameter may
be specified (see Operating Suggestions). "nnnn" specifies the
maximum number of physical tape record blocks to be printed. In
console mode, the optional parameter "NRECS" may be entered to limit
the number of blocks printed. A leading tape mark will be printed
and the function will terminate.

Tape Record Scan

This function allows the user to specify a logical record size and a
1 to 35 position scan argument for an input tape. DITTO will read
and internally deblock input records until the desired record is
located. At that time the desired logical record with its associated
block number will be printed and the operation terminated. This
function supports fixed and variable length records.

OS/DITTO EXTENSIONS allows this function to be executed from control
cards. The new control card parameters ARGUMENT and SCANPOS are
used for scan data.

Tape Record Load

This function is used to copy and selectively alter existing tape
records. The user need not know the exact location (block#) of the
input tape record to be altered.

The first portion of the Tape Record Load function locates the desired
record. The user enters an estimated block location and the parameter
"F" to denote copying of tape in a forward position. When the
desired record is found, it is printed on SYSPRINT and an option
given to alter the record. All input records prior to the record
to be altered have been copied onto the output tape. If the printed
record is not the desired record, the user can again enter the number
of blocks and direction to reposition the input and output tapes.
If the user has gone past the desired record, a parameter of "B"
(back) should be entered with a block count.

This will cause both input and output tapes to be backspaced the
required number of blocks. This count includes the block just printed.
When the desired record is located, the user may alter the length
and/or data content in a manner similar to the Disk Record Load
function. After all alterations are completed, replies of "9999" and
"F" (9999 blocks to be copied in a forward direction) will usually
be sufficient to finish copying the tape file.

Initialize Tape

Initialize tape will build 80 character volume and header labels
using the standard OS format. The new tape volume serial number is
supplied by the user. Existing tape labels or data on the output
tape is not checked before creating the new label set.

Tape to Tape

This function will copy from 1 to 99 tape files per volume as re-
quested via the console. One file or a specified number of blocks
will be copied per control card if console operation is not being
used. A leading tape mark will be automatically copied and will
count as the first end of file. Block count statistics will be
printed on SYSPRINT Operation, for each tape file copied.

Tape to Tape Reblocked

This function will reblock an input tape file to an output tape file.
Input blocking factor will be determined by dividing the first input
tape record by the user supplied logical record length. Output
block length will be determined by multiplying the logical record
size by the user supplied output blocking factor. Input tape labels
should be either copied separately or bypassed. A leading tape mark
will be automatically copied and will count as the first end of file.
Logical record count statistics will be printed on SYSPRINT for
control card operation, and on the console for console operation.

Tape Control

Eight tape control functions are available with console communica-
tions. Seven tape control functions are available with control
card operations. These control functions are: write tape mark;
rewind tape; rewind and unload tape; forward space or backspace
file; forward space or backspace records; and erase tape. The
erase tape function will write a tape mark, erase the remaining tape
data through the end of reel, and rewind the tape to its original
position. The erase tape function is valid for 3420 series tape
units only.

DISK UTILITIES

Data Set Extent Display

This function must be used in colsole mode to determine valid CCCHH
within data set extent. Absolute CCCHH Begin and End are printed on
the console.

Disk Print

These functions provide a disk print in either character or character
and vertical hex format within specified limits. Disk records are
"separated" into key and data format and printed with their associated
cylinder, head, and record addresses. When a defective track is
detected, the asigned alternate track with its respective data
is printed. This provides a sequential listing of the disk file in
"logical" sequence.

Four Disk to Printer formats are available: Disk Print and Disk Dump
unblocked and Disk Print and Disk Dump with Deblocking. The unblocked
format lists the actual records as they exist on the file. The
Deblock function allows the user to submit a logical record size.
This size parameter should be equivalent to the actual logical record
length only, and should not include separate key length. Imbedded
keys should, however, be included. The Deblock function lists the
record key and then deblocks the data portion of the record into
logical lengths. This provides a more easily used listing of large
physical record files. The Dump option provides character and hex.
The Print option provides character only.

Disk Record Load

This function will alter the contents of the key and/or data portion
of an existing Disk Record. After the required disk unit and
record addresses have been entered, the record will be retrieved
and "separated" into key and data format and printed in SYSPRINT.

At this point, DITTO will request the number of bytes to be changed
and the field type (Key or Data portion of the record) to be altered.
After specifying these parameters, a starting position within the
file will be requested. This starting position is based on using the
first position of the key or data as position "0001". The user can
verify this position by referencing the printout of the record
and its associated scale on SYSPRINT. Data may be entered via the
console in character (1 character per byte of change), or hexadecimal
(2 characters per byte of change data). Valid hexadecimal "characters"
are 0-9 and A through F.

After all alterations to the record in core have been made, the altered record will again be printed on SYSPRINT and a message issued "ARE CHANGES COMPLETE - Y or N". A response of "N" (No) will allow the user to re-enter the alter routine. A response of "Y" (Yes) will rewrite the record onto disk.

Disk Record Scan

Disk Record Scan provides the capability of scanning a disk file and recording the locations of all records which match a given scan argument. Three (3) scan argument types are available.

"Scan on Key" allows the user to scan for a match on any portion of the disk record key field. A 1 to 35 position scan argument, and its associated starting position within the key field, are entered via the console. Beginning and ending disk limits are also entered. DITTO will scan the specified disk area and log all "hits" in the form of cylinder, head and record numbers on the console.

"Scan on Data" provides a similar capability for the data portion of disk records. A logical record length must be specified to allow internal deblocking for scanning blocked files.

"Scan on EOF" will log all End of File record locations. Data entry for "EOF" is not required.

Scan areas can be restricted by using the split cylinder option (see Split Cylinder Disk Operations at end of Disk UTILITIES). This option allows the user to limit the scan to selected tracks of each cylinder.

OS/DITTO EXTENSIONS allows this function to be executed from control cards. The new control card parameters ARGUMENT, SCANPOS, and SCANTYPE are used for scan data. The record found is printed on SYSPRINT in console and control card mode.

Write Disk End-of-File Record

This function will retrieve any existing Disk Record and rewrite the record as an end-of-file record (Key and data length of zero). The user must supply the cylinder, head, and record address of the record to be altered.

Disk Identification Change

This function will allow the user to enter on the console a new volume serial number. The old volume serial number and the new volume serial number entered are printed on the console after the function completes.

Split Cylinder Disk Organizations

Split Cylinder data file organizations are supported by DITTO for
the Disk Print unblocked (DP), Disk Print (Deblocked (DPD), Disk
Dump Unblocked (DD), Disk Dump Deblocked (DDD), and the Disk Record
Scan (DRS) functions. Split cylinder operations are denoted by re-
placing the first D in the above function codes with an S. For
these split cylinder operations, the beginning and ending head
addresses will determine the lower and upper head boundaries for
the function.

MISCELLANEOUS UTILITIES

List DITTO Functions

The operation provides a list of functions, control card formats,
console procedures, and parameter requirements. It can be used as
a reference document for operators and programmers.

End of Job

In console mode the user must issue the function code "EOF" to end
DITTO execution. In jobstream mode EOJ is valid and may be issued
in the last control card. This is not necessary as the same
End-of-Job routine will be executed when an End-of-File condition
on the SYSIN data set is encountered.

OS/DITTO EXTENSIONS SYSTEM UTILITY FUNCTIONS

Print or Punch One or All Members of a PDS

This function invokes IEBPTPCH for Partitioned Data Set support.
It may be used from the console or control cards. The cataloged
procedure DITTOPDS supports both printing and punching.

Dump or Restore a DASD Volume

IEHDASDR is invoked from the console by starting either the DITTODUM
or DITTORES procedure. Refer to the XXX listing in Appendix A for
detailed operating directions. The user is prompted for appropriate
key word values.

List Space Available on a DASD Volume

The List Space SVC is issued to obtain the total number of free
cylinders, tracks, extents, and the largest contiguous free area
in cylinders and tracks. This data is formatted and displayed
on the console.

IEHLIST Functions (LISTVTOC, LISTCTLG, and LISTPDS)

This function prompts the console user for the key word values and
options desired. An IEHLIST control card is built and displayed
to the user. The user can then reply with 'E' to execute the con-
trol card or 'U' to cancel the function. IEHLIST is linked to and
fed the control cards on a temporary disk data set. Upon com-
pletion, the IEHLIST return code is displayed to the user. The
cataloged procedure DITTOUTL is used to allocate the desired DASD
volume. A suggested default value to be installed in the procedure
would be the systems' residence volume or some other most frequently
used volume. All options of LISTVTOC, LISTCTLG and LISTPDS are
supported.

IEHPROGM Functions (CATALOG, UNCATALOG, SCRATCH, and RENAME)

These functions operate similarly to the IEHLIST functions. All
options are supported. The SCRATCHVTOC function requires authoriza-
tion according to the rules of Authorized Program Facility.

WRITE TAPE MARK ON DB/DC LOGTAPE WITHOUT REWIND AFTER SYSTEM FAILURE
(OS/DITTO EXTENSIONS)

This function operates independently from OS/DITTO because it re-
quires KEY=0 operation. There is no function code in OS/DITTO for
this feature. The facility is invoked by starting OS/DITTO cataloged
procedure DITTOLOG. Detailed operating instructions are provided
in the comment cards in the procedure. See Appendix C (Job 'ADDPROC')
for a listing of the DITTOLOG procedure. The requirements for
operating in KEY=0 are discussed in the installation instructions.
If KEY=0 status is not achieved the program ABENDS with a System
Completion Code of 300.

UTILITY FUNCTION CODES

See Appendix A for details on the various function utility codes. The
OS/DITTO EXTENSIONS are indicated as such at the top of each page.

TSO TERMINAL INTERFACE

This optional TSO interface consists essentially of changing the
WTOR and WTO sub-routines to TPUT/TGET and TPUT SVC sub-routines.
This allows a TSO terminal to operate OS/DITTO in console mode. The
DITTO print formats have been shortened (80 character maximum
length) and left justified for better use of terminal print/display
maximums.

To invoke TSO/DITTO for DDname DISKIN functions enter the
following commands:

```
ALLOC   FI(SYSPRINT)  DA(*)

ALLOC   FI(DISKIN) DA ('CATLD.OS.DATA.SET')

CALL    'SYS1. LINKLIB (TSODITTO)'
```

Enter DITTO Disk function DSE when the message 'DITTO
FUNCTION ?' is received to obtain CCCHH BEGIN and END values.

DITTO functions TRL, DRS, and DRL work on a physical blocksize
basis. To avoid excessive terminal output, it may be desirable
to re-block files to a small blocksize (or perhaps unblocked)
before using these functions from a terminal. Another alternative
would be to temporarily allocate SYSPRINT as a NULLFILE if
viewing SYSPRINT output at the terminal is unnecessary.

It is not recommended that function XXX be invoked from
the terminal. It has not been optimized for TSO terminal
printing or display.

DDnames SYSPRINT, SYSIN, DISKIN, TAPEIN, TAPEOUT, QSAMIN,
QSAMOUT, VSAMIN, DITTODD1, DITIN, DITPRINT, SYSUT2 and
SYSUT1 can be allocated as needed for the particular function
to be used. Refer to the Cataloged Procedures and XXX to
correlate a function with DDname.

If tapes are to be allocated, use a special logon PROC that
contains DD statements for TAPEIN and/or TAPEOUT. Under MVS you
may dynamically allocate a tape drive by updating the User Attribute
Data Set with the MOUNT capability and an allocation UNITNAME of
SYSSQ. The tape Data Set Name must also be cataloged. An allocate
command can then be issued:

```
ALLOC   FI(TAPEIN) DA('TAPE DATA') LAB(BLP) SHR
```

Coordinate with the system operator for the drive unit address re-
quested in DITTO Tape functions.

GENERAL INSTALLATION INSTRUCTIONS (5798-CDA)

The installation of 5798-CDA assumes that 5798-ARD has been in-
stalled. 5798-ARD Modules OS/DITTO and DITTOXXX are replaced.
5798-ARD Module, DITTODID and all cataloged procedures are left
intact. If 5798-ARD is being installed, run only Jobs ADDREPL,
DIDMFMVT or DIDVS1 #2, and HASP environment job DITTORDI.

PID TAPE DESCRIPTION

The program source material is distributed on magnetic tape
(no leading Tape Mark). The tape contains 10 files. Each file
consists of the following jobs (CLASS=G):

FILE #	JOB NAME	JOB DESCRIPTION AND PURPOSE
1.	INS#LIB	IEHPROGM and IEFBR14 to scratch and re-allocate SYS1. DITTOLIB.
	INS#PROC	IEBUPDTE to SYS1. PROCLIB to provide a means of controlling the DATA SETS and UNITNAMES used by the installation jobs.
	ADDPROC	IEBUPDTE to install new DITTO cataloged procedures. These PROCS are in addition to 5798-ARD PROCS.
2.	DITTOMAC	IEBUPDTE to establish an ASSEMBLER copy library on the same volume as SYS1. DITTOLIB. It will be concatenated to MACLIB.
	ARDUPDTE	IEBUPDTE to Version 2 Modification 0 PID TAPE for 5798-ARD. A new source tape (CDA001) is created, assembled, and Link Edited (NCAL) into SYS1. DITTOLIB.

3.	WTORSUB	WTOR Sub-routine	)	
			)	These jobs are
	WTOSUB	WTO Sub-routine	)	
			)	Assemblies and
	CNCDSUB	Control Card Sub-routine	)	
			)	LINK EDITS (NCAL)
	FUNCSUB	Function Code Sub-routine	)	
			)	into SYS1. DITTOLI
	DISKOPEN	DEVICE TYPE and OPEN Sub-routine for DASD	)	

FILE #	JOB NAME	JOB DESCRIPTION AND PURPOSE	
	CDRDSUB	Card Read Sub-routine	)
	CDPNSUB	Card Punch Sub-routine	)
	PRNTSUB	Print Sub-routine	)
	HEADING	OS/DITTO Heading Page Print Sub-routine	)
4.	DITTOPGM	IEHPROGM Interface routines	)
	DITTOLST	IEHLIST Interface routines	)
	DITTOPDS	IEBPTPCH Interface routines	)
	DITTODAS	IEHDASDR Interface routines	) These jobs are
	DITTOQV	QSAM and VSAM function routines	) Assemblies and
	DITTOLOG	DB/DC LOGTAPE CLOSE routine is Assembled and Link Edited (AC=1) into SYS1. LINKLIB	) LINK EDITS (NCAL)) into SYS1. DITTOLIB.
5.	DITTOXXX	'XXX' documentation listing module	)
	DITTOE11	'XXX' for Extensions Page 1	)
	DITTOE12	'XXX' for Extensions Page 2	)
	DITTOE13	'XXX' for Extensions Page 3	)
	DITTOE14	'XXX' for Extensions Page 4	)

FILE #	JOB NAME	JOB DESCRIPTION AND PURPOSE	
6.	DITTOLE	OSDITTO)	Module OSDITTO will replace 5798-ARD
		)	module OSDITTO
	PGMLE	DITTOPGM)	
		)	
	LSTLE	DITTOLST)	These Load modules are Link Edited
		)	
	PDSLE	DITTOPDS)	(AC=1) into SYS1. LINKLIB for
		)	
	DASLE	DITTODAS)	execution.
		)	
	QVLE	DITTOQV)	
	XXXLE	DITTOXXX)	Module DITTOXXX will replace
		)	
		)	5798-ARD module DITTOXXX.
7.	SAMPXXX		
	to	This jobstream consists of 20 sample jobs each demonstrating the use of OS/DITTO EXTENSIONS new function control card	
	SAMPDRS	formats.	
8.	TSOUPDTE	IEBUPDTE to the new source tape (CDA001) is created by job ARDUPDTE.	
		TSO changes are made to create a second new source tape (TSO001).	
		The TSO tape is then assembled and Link Edited (NCAL) into SYS1. DITTOLIB.	
9.	TSOFUNC	Function Code Sub-routine)	
		)	These jobs are
	TSOPRNT	Print Sub-routine)	Assemblies and
		)	Link Edits (NCAL)
	TSOWTOR	TPUT and TGET Sub-routine)	into SYS1. DITTOLIB
		)	Necessary changes
	TSOWTO	TPUT Sub-routine)	for TSODITTO are
		)	contained in these
	TSOQV	QSAM and VSAM function)	modules.
		routines)	

FILE #	JOB NAME	JOB DESCRIPTION AND PURPOSE	
10.	TSODITLE	TSODITTO)	
	TSOPGMLE	DITSOPGM)	
	TSOLSTLE	DITSOLST)	These modules are Link
	TSOPDSLE	DITSOPDS)	Edited (AC=1) into SYS1.
	TSODASLE	DITSODAS)	LINKLIB for execution.
	TSOQVLE	DITSOQV)	
	TSOXXXLE	DITSOXXX)	

Each Jobstream file may be read in for execution by issuing an
OS START Reader command as follows:

```
S RDR. P2, 280, DCB=(BLKSIZE=1600, RECFM=FB,
LRECL=80, BUFL=1600, BUFNO=1), LABEL=(n,BLP)
```

Vary the n value in LABEL=(n,BLP) from 1 to 10 to read in the
appropriate file. When installing under OS/MFT, P2 must be defined
as a Resident Reader Partition to support blocked input.

The modules Link Edited in File 6 and File 10 must be in an Author-
ized Library for VS1, SVS, and MVS. The AC=1 parm field is ignored
by MFT and MFT Linkage Editors. The authorized library used is
controlled by INS#PROC member DITTODSR. VS1 Release 4 Authorization
requires OS/DITTO in the Program Properties Table (See Appendix D).

In addition, the DITTOLOG module in File 4 must run KEY=0. For
MFT, MVT, VS1, and SVS this requires that the module name DITTOLOG
be placed in the first section of the LINK TABLE (member IEEVLNKT)
of SYS1. LINKLIB or SYS1. LPALIB). For MVS an entry for DITTOLOG
is required in the PROGRAM PROPERTIES TABLE (csect IEFSDPPT of
scheduler module IEFSD060 in SYS1. LPALIB). The easiest way to
satisfy this requirement is to SUPERZAP in DITTOLOG over presently
unused entries or dummy entries maintained by your installation
for this purpose. DITTOLOG is the only module that requires
KEY=0 when started from the console.

Consult the release documentation for the particular System Control
Program being used in your installation to determine exact procedures
for Authorization and KEY=0 for Started Tasks. Refer to Appendix D
for a profile of these requirements by operating system.

All jobstreams should be run single thread to avoid SYSLMOD Link
Edit conflicts.

DETAILED INSTALLATION INSTRUCTIONS
FOR
OS/DITTO EXTENSIONS   (5798-CDA)

1.   Review the jobs contained in File 1 (see Appendix C) for
      these considerations:

      INS#LIB   Allocates data set SYS1. DITTOLIB on VOL=REF=SYS.1.
                 LINKLIB. SPACE required is approximately four
                 cylinders of 2314. SPACE parameter is coded
                 SPACE=(Cyl, (2, 1, 10)). SYS1. DITTOLIB is
                 CATALOGED.

      INS#PROC   Provides a PROC to control Data Sets Referenced
                 (member DITTODSR) by the Assemblies and Link
                 Edits. This allows easy control (from one place)
                 of where NCAL and Executable Load Modules are
                 Link Edited. All Link Edit jobs use this PROC
                 in the first step and refer back to it for SYSLMOD.
                 Job ADDPROC uses refer back for the SYSUT2 output.

                 A second PROC is included for use by jobs ARDUPDTE and
                 TSOUPDTE. The tape drive default unitname is 2400.
                 This PROC (member DITUPDTE) will be used during the
                 error correction period to update the source for
                 OSDITTO and TSODITTO modules.

      ADDPROC   This job adds the new OS/DITTO PROCS to SYS1. PROCLIB.
                 The tape drive unitname default is TAPE. DASD drives
                 are default allocated by SYSDA. Distribution Region
                 sizes are 100K to allow adequate room for SYSTEM
                 UTILITIES to be linked in and for QSAM buffers. A
                 smaller Region size could be used depending on the
                 size generated for OS UTILITIES and QSAM blksizes used.

2.   Make changes to File 1 jobs if needed, then run them as prepara-
      tory steps for Files 2 through 10. Verify successful completion
      before proceeding.

3.   Run File 2 jobs DITTOMAC and ARDUPDTE. DITTOMAC copies the DCB
      parameters from SYS1. MACLIB to allow concatenation to SYS1.
      DITTOMAC during Assemblies. Most Assemblies will copy members

DITTODS1 and DITTODS2. The 'XXX' EXTENSION modules will copy
DITTOXE1. ARDUPDTE will call for VOL=SER=ARDTAP. This must
be the PID tape for OS/DITTO 5798-ARD Version 2, Modifica-
tion 0. Another version may not complete IEBUPDTE success-
fully. SECOM maintenance to Version 2, Modification 0,
may be present. It is included in the CHANGE stream. An
output tape VOL=SER=CDA001 is called for. This tape contains
the source for OS/DITTO EXTENSIONS and should be retained
for future maintenance. Check Assembly for Condition Code zero.

4. Now that PROC DITTODSR and Data Sets SYS1. DITTOLIB and SYS1.
    DITTOMAC are established the installation of OS/DITTO EXTEN-
    SIONS is ready for completion. Simply run all the Assemblies
    and Link Edits contained in Files 3 through 6 in single thread
    fashion via a single CLASS=G INITIATOR.

5. Review Assembly Condition Codes for zero. NCAL Link Edits
    will be Condition Code zero or four. Final AC=1 Link Edits
    into SYS1. LINKLIB should be Condition Code zero.

6. Take the necessary steps to allow the DITTOLOG module to run
    KEY=0 when started from the console. If the use of this
    facility is not antiticpated do not bother. The PROC DITTOLOG
    executes the module DITTOLOG and is totally separate from OS/DITTO
    PROCS and modules.

7. To verify proper installation the Sample Jobstream may be run.
    In particular, the first job should be run to obtain an 'XXX'
    listing which documents the new OS/DITTO EXTENSION function
    codes, control card formats, cataloged procedures, and console
    operation using string parameters.

INSTALLATION INSTRUCTIONS FOR TSO/DITTO
(TSO TERMINAL INTERFACE TO OS/DITTO)

1. Run the job contained in File 8 (TSOUPDTE). This job will call for input tape VOL=SER=CDA001 (created by Job ARDUPDTE). Output tape VOL=SER=TSO001 is called for. This tape contains the source for TSODITTO and should be retained for future maintenance.

2. The jobs in File 9 and 10 may now be run to complete the installation of TSODITTO. Review the Assemblies for Condition Code Zero. NCAL Link Edits will be Condition Code zero or four. Final AC=1 Link Edits into SYS1. LINKLIB will be Condition Code zero.

3. To verify the installation of TSODITTO:

    a. LOGON and ALLOCATE DDnames and Data Sets as needed.

       - SYSPRINT to the Terminal

       - TAPEIN, TAPEOUT, DISKIN, QSAMIN, QSAMOUT, and VSAMIN to drives/data sets.

    b. CALL 'SYS1. LINKLIB (TSODITTO)'

    c. Issue function codes desired as if operating in console mode DITTO.

4. No Command Procedures are distributed. It will be desirable to build those that would be convenient for installation use of TSODITTO functions. Another possibility would be to provide a special LOGON procedure that allocates TAPE drives or Data Sets most frequently used.

SAMPLE JOBSTREAM

File Number 7 contains sample jobs showing how to use OS/DITTO EXTENSIONS from control cards. The jobstream contains card data to build files. Review for JCL sensitive items such as the tape unit parameter of 2400 and a DASD Unit of SYSDA. Job SAMPQCI asks for SYSPUNCH online at address 007. Job SAMPQP160 will request the PID TAPE as input. Job SAMPVPEK asks for VOL=SER of 'STEVEB' and contains VSAM AMS control cards that refer to this volume. Job SAMPTRS will use the PID TAPE as input. Job SAMPTRSV will have to be changed to scan for a valid data field on a variable blocked tape provided by your installation.

OPERATING INSTRUCTIONS

CONTROL CARD OPERATION

If the user wishes to remove the decision requirements from
the console operator, he may prepare control cards and submit
his task in a normal Jobstream environment.

To denote control card operation, the user must submit on
an Exec card a PARM='JOBSTREAM' parameter. DITTO will test
the PARM field to determine whether control card or console
operation is desired.

Most required control card information can be contained on one
card. A control card may be continued by a trailing comma and
a non-blank character in Column 72. There is no limit to the
number of continuations. For details see Appendix A and the control
card format section of the XXX Documentation listing. The con-
tinued card must begin with $$DITTO in card column one. Each
control card set (if continuations exist) is treated as a new
operation. The control card format is free form using keyword
parameters. At least one blank must separate the $$DITTO field
and the Function Code, and at least one blank must separate the
Function Code and the first parameter.

```
    CC 1-7      $$DITTO                              CC. 72
                ...at least one blank...
                Function Code
                ...at least one blank...
                Parameter1,.....,Parameter N(,)          (X)
```

Parameters are in standard keyword format. Each parameter must
be separated with a comma with no embedded blanks. A blank
stops the card scan. Refer to Appendix B for control card ex-
amples of the OS/DITTO EXTENSIONS control card functions.

Control Card Jobstream Examples

```
1.    //COPY TAPE    JOB     MSGLEVEL=1
      //             EXEC    PGM=OSDITTO,PARM='JOBSTREAM'
      //TAPEIN        DD      UNIT=2400,DISP=OLD,DSN=COPYIN,LABEL=(,BLP)
```

```
//                      VOL=SER=INTAPE
//TAPEOUT     DD        UNIT=2400,DISP=OLD,DSN=COPYOUT,LABEL=(,BLP)
//                      VOL=SER=OUTAPE
//SYSPRINT    DD        SYSOUT=A
//SYSIN       DD        *
$$DITTO       TT        INPUT=TAPEIN,OUTPUT=TAPEOUT
$$DITTO       REW       OUTPUT=TAPEIN
$$DITTO       TPD       INPUT=TAPEIN,RECSIZE=80,NBLKS=5
$$DITTO       EOJ
/*


2.   //PRNTDISK    JOB       MSGLEVEL=1
     //            EXEC      PGM=OSDITTO,PARM='JOBSTREAM'
     //DISKIN      DD        UNIT-3330,DISP=OLD,DSN=DISKDATA,
     //                      VOL=SER=333001
     //SYSPRINT    DD        SYSOUT=A
     //SYSIN       DD        *
     $$DITTO       DD        INPUT=DISKIN,BEGIN=00000,END=00000
     $$DITTO       DD        INPUT=DISKIN,BEGIN=00018,END=00018
     $$DITTO       DDD       INPUT=DISKIN,BEGIN=00100,END=00101,
                            RECSIZE=80
     $$DITTO       EOJ
     /*


3.   //DUPCARDS    JOB       MSGLEVEL=1
     //            EXEC      PGM=OSDITTO,PARM='JOBSTREAM'
     //SYSPRINT    DD        SYSOUT=A
     //SYSPUNCH    DD        SYSOUT=B
     //SYSIN       DD        *
     $$DITTO       CC
     DATA CARDS    (followed by /+)
     /+
     $$DITTO       CP
     DATA CARDS    (followed by /+)
     /+
     $$DITTO       EOJ
     /*
```

NOTE: The DCB parameter is <u>not</u> required. For further
        Jobstream examples, please reference the sample
        jobs supplied on the distribution tape. Also
        additonal information is on the XXX documentation
        listing (Appendix A).

CONSOLE OPERATIONS

All operator communication may be in upper or lower case. The
cancel key is also implemented allowing retyping of replies
in error. The end of message may be denoted by the EOB/END"
key.

Replying "IRPT" to the "*** OS/DITTO active ***" message will
cause the existing function to be terminated and a new opera-
tion request displayed on the console This is an outstanding
(un-answered) WTOR message technique. The ECB is checked in all
input/output routines for an answer. The interrupt will not occur
if no I/O activity is present.

Tape address replies are entered in the form of "CUUMM" where
C=Channel, UU=unit address, and MM=the density and mode setting.
For nine-track single density tapes the MM parameter may be
omitted.

For console operation, card input end-of-file is indicated by
physical end-of-file on the input device. End-of-file is denoted
by depressing the E-O-F key on the card reader, or the presence
of a tape mark if SYSIN is assigned to tape. This provides
the operator with the ability to include embedded /* cards in
the input data. If the operator desires to cancel a program
with card data input, he should interrupt DITTO and enter "CCL"
on the console. This will cause the input stream to be flushed
to end-of-file.

DITTO may be executed entirely from the console by entering an
OS start command for the desired OS/DITTO console procedure. Each
procedure allocates a unique group of I/O devices via data definition
cards. This console allocation technique provides the required I/O
devices for a given set of DITTO functions. The use of standard OS
allocation also insures program operating stability and release
independence. The supplied procedures and how to use them are
described in the XXX documentation listing.

The DITTO functions request operator information in a tutorial
sequence. Messages specify the length and format of each required
response, and will re-request a response if in error. To minimize
console messages, the operator may enter up to five responses
with one reply. The operator should reference the XXX pages
entitled "Console Procedures" and "Console Operations" for specific
parameters required.

OPERATING SUGGESTIONS

Printing Tape Files

Programmers who require listings of tape files will find the Tape
Hex formats more useful than the character print functions. The Tape
Hex functions provide both character and vertical hexadecimal for-
mat with a position scale for each tape record. Programmers using
these functions under the remote control card operation should also
make use of the NBLKS parameter. This optional parameter gives
the programmer the capability of obtaining "partial" listings of tape
files.

Tape Errors

All tape errors will initially be handled by Input/Output Supervisor.
For Input tape errors that are uncorrectable, DITTO will transfer
control to a console routine when in console operation mode. This
routine will display the record as read on SYSPRINT, and allow the
Operator to accept, bypass, or alter the data and/or length of the
tape record before returning control to the original operation. This
option is not available for remote programmer control card operation.

Multiple Console Support and Route Codes

All WTO's and WTOR's contain DESCRIPTOR and ROUTECODE parameters.
DESC=6 and ROUTCDE=2 are the current values. To change for installa-
tion standards reference the statement numbers listed. This allows
an installation with Multiple Console Support to isolate OS/DITTO
messages and replies to a given console.

WTO/WTOR statement numbers by installation job:

//OSDITTO	//DIDMFMVT	//DIDVS1#2
OS013801	D1469700	D1490900
	D1476900	D1495900
	D1477810	D1496810
	D1478200	D1497200
	D1478500	D1497500

//WTORSUB	//WTOSUB
WR006900	WO016600

Card Input Assignment

For control card operations, the end of the card input for a given
$$DITTO control card is signaled with a/+ card. This will denote
a unique "software" end-of-file for the particular function. It
also will allow multiple sets of $$DITTO Cnn control cards and
associated card data to be read by an OS Reader until the /* state-
ment or equivalent is encountered.

The last $$DITTO operation must be a $$DITTO EOJ card and should
be followed by a /* card or it's equivalent. Reference the
sample control card jobstream for examples. Jobstreams consist-
ing of JCL and embedded /* cards may be supplied as card input
data by using a //SYSIN DD DATA DLM=AA statement with an AA card
(AA in C.C. 1 and 2) delimiting all SYSIN data.

For console operation, card input end-of-file is indicated by
physical end-of-file on the input device. End-of-file is denoted
by depressing the E-O-F key on the card reader, or the presence
of a tape mark if SYSIN is assigned to tape. Since the input
device is allocated online to DITTO, jobstreams with embedded
/* cards may be read as card input data.

3525 Card Punch Printing Feature

DITTO supports either the two line (3525 feature #8338) or the
multi-line (3525 feature #5272) print feature for punched output
via the OS JCL FUNC=I parameter. DITTO will print the first
64 card columns of punched data on line one of the card, and the
remaining 16 card columns of data on line three. The user initiates
this punch and print operation by providing a SYSPUNCH DD card
of the following form when using functions CC, CCS, and TC,
E. G.,//SYSPUNCH DD UNIT=00D,DCB=(BLKSIZE=81,RECFM=FA,FUNC=I).

This punch device must be allocated online. The OS writer will
not interpret SYSOUT=B output data. FUNC=I JCL support is in
OS Release 21.0 and current OS/VS releases.

Standard Tape Labels

OS/DITTO issues Open/Close/EOV for all Tape, Disk, and Unit Record
data sets. It is feasible to process standard labeled tapes as
such. However, OS/DITTO physical block and Tape Mark handling is
done on an EXCP basis without the use of standard OS Access Methods.

It is recommended that the LABEL=(,BLP) parameter be used on
the TAPEIN and TAPEOUT DD cards. Standard label tapes may be
processed on a physical basis. The user must allow for standard
Label format by issuing additional DITTO tape functions to print
and/or forward space by Standard Labels. Card to Tape and Tape to
Tape functions write a Tape Mark. This is in addition to the Tape
Mark written by CLOSE. If creating a standard Label tape, issue
a Back Space File (BSF) to back up over the extra Tape Mark
(written by the DITTO function) before EOJ. CLOSE will then create
a Tape Mark before the Trailer Labels. The standard Label Tape
will lack RECFM, BLKSIZE, and LRECL information but this may be
provided by a JCL DCB parameter at creation time or input time.

Standard Label Tape with OSAM Functions

The new OS/DITTO EXTENSIONS QSAM Input/Output functions are the
best way to use DITTO functions with Standard Label information.
Functions CQ, CQS, QP, QD, and QQ along with cataloged procedures
provide an easy way of creating and reading the Standard Label
information on Tape or Disk.

User Abends

OS/DITTO issues user abends when unusual conditions occur. The
abend codes and conditions are as follows:

-        U100 - Control card error. The XXX listing is printed
         for error correction reference. No Dump is taken.

-        U101 - The SYSPRINT DCB did not OPEN successfully.
         No Dump is taken.

-        U200 - A physical input block has exceeded the
         input/output buffer. This buffer is fixed at assembly
         time to insure VS page-fixing efficiency. To in-
         crease buffer size beyond 14K as distributed, reference
         the Systems Manual - "Reducing Storage Requirements
         Section". A Dump is taken to verify this condition.

- U300 - Unusual tape input/output error. Reference the tape ECB's in the SYSUDUMP listing for cause.

- U400 - Unusual disk input/output error. Reference the DISKIN ECB in the SYSUDUMP listing for cause.

- U500 - During function TTR, an input block has been read with one of the following conditions:

  (1) Exceeds the first input block read from data set or

  (2) Not multiple of logical record size.

  Reference the "DTPIO" input/output buffer in the SYSUDUMP listing for verification.

- U604, U608, U612, U616, U620 - are TPUT SVC non-zero return code designations used in the TSOWTOR and TSOWTO sub-routines.

- U704, U708, U712, U716, U720 - are TGET SVC non-zero return code designations used in the TSOWTOR sub-routines.

OPEN/CLOSE/EOV Considerations

The TAPEOUT data set is OPEN'ed OUTPUT. A CLOSE is issued by functions EOJ and RUN. This causes the OS unlabeled tape END-OF-DATA SET sequence to be written. This sequence consists of two tape marks. Therefore, when copying tapes, WTM commands for END-OF-DATA SET are unnecessary if EOJ follows the copy function.

Also, tape editing of the TAPEOUT data set should be followed by one or more FSF's before the EOJ function to prevent writing tape marks in a data area. Caution should be used with the RUN function for the same reason. It is recommended that most tape-to-printer functions be performed from the TAPEIN data set. EOV is issued only when END-OF-REEL is sensed on the TAPEOUT data set.

Tape Mark and Load Point Message

Unlike DOS/DITTO, the OS/DITTO program issues a message to the operator/programmer when the "at Load Point" or "Tape Mark sensed" condition is recognized during operations subsequent to a BSF, BSR, or FSR operation.

Tape Density and Recording Technique (Mode)

In console mode, the operator may specify (if needed) the "MM" suffix when entering CUUMM to set density, parity, and track. The default is CO. It may be changed by reordering the Mode Table at statements OS042500 and OS045400. The tape mode appearing on the TAPE to PRINTER heading output is the mode supplied by the operator of JCL. The Density printed is the actual density of the input tape (obtained from sense bytes set by the first read). The "XXX Listing" contains specifications on the "MM" mode settings. Programmers may submit the DCB parameter with the DEN and TRTCH sub-parameters to control these settings.

DITTODP and DITTODSN Console Procedure Consideration

The DITTODP procedure requires the symbolic parameters DSNM and SER to provide data set name and volume DD parameters. The procedure will fail with a JCL error if DSNM and SER are not provided on the START command. The procedure DITTODSN provides the same parameters as DITTODP as well as QUAL= to specify DSN for single-level Qualified Data Set Names.

Initiating Console Mode DITTO Through the Jobstream to Obtain SMF Data

If the user requires that console mode OS/DITTO be initiated through the jobstream reader as opposed to using a START command so that SMF Data for DITTO will be written, the following steps should be taken:

(1)   Remove the PARM="JOBSTREAM" parameter from the EXEC card

(2)   Submit the DD cards that will be needed (TAPEIN, TAPEOUT, and/or DISKIN)

This could also be done via a S RDRJOB, PRC=DITTO--technique similar to the HASP internal reader technique. RDRJOB would be an OS reader procedure instead of a HASP internal reader.

DOS SYSLST Tapes

DITTO functions TFA and TFD have been removed from the OS version. If the user requires the capability of printing DOS SYSLST tapes under OS, he may employ the same technique used to print OS Direct Systems Output tapes.

Under OS, DSO tapes are printed by using the OS utility IEFPRINT.
This utility may also be used to print DOS SYSLST tapes. BLKSIZE,
record format, and carriage control (ASA or machine may be speci-
fied on the input DCB parameter).

Using DITTO Console Procedures in the Jobstream

Most OS/DITTO cataloged Procedures may be invoked from the job-
stream. This saves the programmer the task of supplying the
necessary DD statements. For example, the statements needed to
execute procedure DITTODP and DITTOTP.

```
//DISKPRINT JOB MSGLEVEL=1
//EXEC DITTODP,PARM="JOBSTREAM",SER=WORK01,DSNM=MYDATA
//STEP 1.SYSIN DD *
$$DITTO DP BEGIN=00000,END=00000,INPUT=DISKIN
/*
//TAPEPRNT JOB MSGLEVEL=1
//EXEC DITTOTP,PARM="JOBSTREAM"
//STEP1.SYSIN DD *
$$DITTO TP INPUT=TAPEIN
```

All procedures contain STEP 1 on the EXEC card to allow over-
riding and adding DD cards.

The Effect of BSF, FSF, and ERT on Tape Control Units

The Back Space File, Forward Space File, and Erase Tape command
are executed in the Tape Control Unit as are all other tape drive
commands. If the execution of the command requires that a major
portion of the tape reel be passed, the tape control unit is
temporarily unavailable to other jobs in the system. This is
only significant in full reel cases. If the case arises, the control
unit can be made available by taking the drive out of ready, cancel-
ing DITTO, and then ready to drive.

Accessing Multiple Extent Data Set Using Control Cards

OS/DITTO provides access to each extent of a multiple extent data
set serially beginning with the first through the last. The
first extent is available for printing until the End-of-Extent
(or invalid extent is reached). At that time the second extent
is made available for printing. This process may be continued until
all extents are exhausted at which time the final extent remains
available. The following is an example for accessing the first

and last track of each extent of a Data Set. The Data Set was
allocated with "UNIT=3330,SPACE=(CYL,(5.1)" and Secondary allocation
was entered 2 times.

```
-1st TRACK(1st EXTENT)       $$DITTO DP  INPUT=DISKIN,BEGIN=0000,END=00000
-LAST TRACK(1st EXTENT)      $$DITTO DP  INPUT=DISKIN,BEGIN=00418,END=00500
-1st TRACK(2nd EXTENT)       $$DITTO DP  INPUT=DISKIN,BEGIN=00000,END=00000
-LAST TRACK(2nd EXTENT)      $$DITTO DP  INPUT=DISKIN,BEGIN=00018,END=00100
-PRINT TO EOF(LAST EXTENT)   $$DITTO DP  INPUT=DISKIN,BEGIN=00000,END=00018
```

   *   END=00500 and END=00100 causes Out of Extent and Next
       Extent is made available.

APPENDIX A

```
OCOCOGOUOO    SSSSSSSSSS          *   DDDCDDDCDDD   IIIIIIIIIIII   TTTTTTTTTTTT   TTTTTTTTTTTT   OOOOOOOOOOO
OOCOLOOOCOOO  SSSSSSSSSSSS        *   DDDDDDDDDDDD  IIIIIIIIIIII   TTTTTTTTTTTT   TTTTTTTTTTTT   OOOOOOOOOOOO
OO        OO  SS        SS       *    DD        DD       II            TT             TT         OO        OO
OO        OO  SS                *     DD        DD       II            TT             TT         OO        OO
OU        OO  SS               *      DD        DD       II            TT             TT         OO        OO
CO        OO  SSSSSSSSSSS      *      DD        DD       II            TT             TT         OO        OO
OU        OO   SSSSSSSSSSS    *       DD        DD       II            TT             TT         OO        OO
OO        OO          SS     *        DD        DD       II            TT             TT         OO        OO
OO        OO          SS    *         DD        DD       II            TT             TT         OO        OO
OO        OO  SS        SS  *         DD        DD       II            TT             TT         OO        OO
OOCOOOOOOOOO  SSSSSSSSSSSS  *         DDDDDDDDDDDD  IIIIIIIIIIII       TT             TT         OOOOOOOOOOOO
OOOOOOOOOO    SSSSSSSSSS    *         DDDDDDDDDD    IIIIIIIIIIII       TT             TT         OOOOOOOOOO
```

5798-ARD COPYRIGHT IBM CORPORATION 1973

DATE=75.342 VERS.2 MOD.0

5798-CDA COPYRIGHT IBM CORPORATION 1975

EXTENSIONS VERS.1 MOD.0

```
****************************** O S / D I T T O ********************************
*                                                                            *
*                            CARD FUNCTIONS                                  *
*          CC        CARD TO CARD                                            *
*          CCS       CARD TO CARD WITH SEQ. NUMBERS AND DECK NAME            *
*          CP        CARD TO PRINTER IN CHARACTER FORMAT                     *
*          CD        CARD TO PRINTER IN CHARACTER AND HEX DUMP FORMAT        *
*          CT        CARD TO TAPE BLOCKED 1 TO 400                           *
*          CTS       CARD TO TAPE RESEQUENCED                                *
*          CCL       CANCEL CARD INPUT (FLUSH TO EOF)                        *
*                                                                            *
*                            TAPE FUNCTIONS                                  *
*          TC        TAPE TO CARD BLOCKED OR UNBLOCKED                       *
*          TP        TAPE TO PRINTER UNBLOCKED IN CHAR. FORMAT               *
*          TPD       TAPE TO PRINTER DEBLOCKED IN CHAR. FORMAT               *
*          TD        TAPE TO PRINTER UNBLOCKED IN CHAR. AND HEX DUMP         *
*          TDD       TAPE TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP         *
*          TPV       TAPE TO PRINTER VARIABLE RECDS CHAR. FORMAT             *
*          TDV       TAPE TO PRINTER VARIABLE RECDS CHAR. AND HEX DUMP       *
*          TRS       TAPE RECORD SCAN                                        *
*          TRL       TAPE RECORD LOAD                                        *
*          INT       INITIALIZE TAPE                                         *
*          TT        TAPE TO TAPE  (01 TO 99) FILES                         *
*          TTR       TAPE TO TAPE REBLOCKED                                  *
*          WTM       WRITE TAPE MARK                                         *
*          REW       REWIND TAPE                                            *
*          RUN       REWIND AND UNLOAD TAPE                                  *
*          FSR       FORWARD SPACE RECORD                                    *
*          BSR       BACK SPACE RECORD                                       *
*          FSF       FORWARD SPACE FILE                                      *
*          BSF       BACK SPACE FILE                                         *
*          ERT       ERASE TAPE  (DATA SECURITY ERASE - 3420 ONLY)          *
*                                                                            *
*                            DISK FUNCTIONS                                  *
*          DSE       DISPLAY DATA SET EXTENT ON CONSOLE, USE FOR CCCHH       *
*          DP        DISK TO PRINTER UNBLOCKED IN CHAR. FORMAT               *
*          DD        DISK TO PRINTER UNBLOCKED IN CHAR. AND HEX DUMP         *
*          DPD       DISK TO PRINTER DEBLOCKED IN CHAR. FORMAT               *
*          DDD       DISK TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP         *
*          DRL       DISK RECORD LOAD - KEY AND/OR DATA                      *
*          DRS       DISK RECORD SCAN - PARTIAL KEY OR DATA OR EOF           *
*          EOF       WRITE DISK EOF RECORD                                   *
*          DID       ALTER DISK IDENTIFICATION VOLUME LABEL (OFFLINE)        *
*                                                                            *
*          XXX       LIST FUNCTIONS ON SYSPRINT                              *
*          EOJ       END OF JOB                                              *
*                                                                            *
*     CONTROL CARD OPERATION - /* IS SYSIN EOF FOR CONTROL CARDS & DATA.     *
*                              IF /* IN DATA, //SYSIN DD DATA,DLM=AA.        *
*                              /+ IS OS/DITTO EOF FOR DATA CARD INPUT.       *
*     CONSOLE OPERATION - READER EOF KEY IS END OF FILE FOR CARD INPUT.      *
*     FOR SPLIT CYLINDER DISK FUNCTIONS: USE SP, SD, SPD, SDD, AND SRS.      *
*     PUNCHED OUTPUT MAY BE INTERPRETED BY USING //SYSPUNCH DD UNIT=00D,     *
*     DCB=(BLKSIZE=81,RECFM=FM,FUNC=I) - 3525 PRINT FEATURE REQUIRED.        *
*                                                                            *
*               L. S. BRAMMER       IBM CORPORATION  1973                    *
*****************************************************************************
```

```
*************************** EXTENSIONS  *****************************************
*                                                                             *
*                    SYSTEM UTILITY FUNCTIONS                                  *
*                                                                             *
*       PRT       PRINT ONE OR ALL MEMBERS OF A PDS                            *
*       PUN       PUNCH ONE OR ALL MEMBERS OF A PDS                            *
*       DUM       DUMP A DASD VOLUME TO TAPE                                   *
*       RES       RESTORE A DUMP TAPE TO DASD                                  *
*       LS        LIST SPACE AVAILABLE ON A DASD VOLUME                        *
*       LV        LISTVTOC OF A DASD VOLUME                                    *
*       LC        LIST CATALOG (SYSCTLG - IPLVOL OR CVOL)                      *
*       LP        LIST PDS DIRECTORY                                           *
*       CAT       CATALOG DSNAME                                               *
*       UNC       UNCATALOG DSNAME                                             *
*       SCR       SCRATCH VTOC, DATA SET, OR MEMBER OF A PDS                   *
*       REN       RENAME DATA SET OR MEMBER                                    *
*                                                                             *
*                 QSAM (CARD, DASD, AND SL/BLP TAPE) FUNCTIONS                 *
*                                                                             *
*       CQ        CARD TO QSAM                                                 *
*       CQS       CARD TO QSAM WITH SEQ. NUMBERS AND DECK NAME                 *
*       QC        QSAM TO CARD (USE FUNC=I FOR 3525 INTERPRET)                 *
*       QQ        QSAM TO QSAM (COMPATIBLE RECFM'S ASSUMED)                    *
*                                                                             *
*                 QSAM/VSAM PRINT FUNCTIONS                                    *
*                                                                             *
*       QP        QSAM TO PRINTER DEBLOCKED IN CHAR. FORMAT                    *
*       QD        QSAM TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP              *
*       VP        VSAM TO PRINTER DEBLOCKED IN CHAR. FORMAT                    *
*       VD        VSAM TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP              *
*                                                                             *
*                 MISCELLANEOUS FUNCTIONS                                      *
*                                                                             *
*       TRS       CONTROL CARD - TAPE RECORD SCAN                              *
*       DRS       CONTROL CARD - DISK RECORD SCAN (KEY/DATA/EOF)               *
*                                                                             *
*       ---       S DITTOLOG - WRITE TAPE MARK ON DB/DC LOG TAPE              *
*                             WITHOUT REWIND AFTER SYSTEM FAILURE              *
*                             (SEE COMMENT CARDS IN DITTOLOG PROC              *
*                             FOR OPERATIONAL DETAILS)                         *
*                                                                             *
*   SYSTEM UTILITY FUNCTIONS PROMPT FOR KEYWORD VALUES, DISPLAY SYSIN         *
*    CARDS, ASK IF EXECUTION DESIRED, AND LINK TO IEBPTPCH, IEHDASDR,         *
*    IEHLIST, AND IEHPROGM. THE UTILITY RETURN CODE IS DISPLAYED.             *
*                                                                             *
*   CQ AND CQS RECOGNIZE RECFM=V AND VB ON QSAMOUT AND THE 4 BYTE RDW         *
*    (RECORD DESCRIPTOR WORD) IS BUILT. IF QSAMOUT RECFM, BLKSIZE, AND        *
*    LRECL NOT SUBMITTED: RECFM=FBS,BLKSIZE=3120,LRECL=80 IS USED.            *
*                                                                             *
*   QC, QP, AND QP RECOGNIZE RECFM=V AND VB ON QSAMIN. THE 4 BYTE RDW         *
*    FIELD IS USED FOR LRECL. A PDS MEMBER IS PROCESSED BY DSN(MEMBER)        *
*                                                                             *
*   QQ ASSUMES COMPATIBLE RECFM'S (V TO VB, F TO FBS) ON QSAMIN AND           *
*    QSAMOUT. NO ADJUSTMENT FOR VAR LEN RDW  FIELDS IS MADE. IF               *
*    QSAMOUT RECFM, BLKSIZE, AND LRECL NOT SUBMITTED: QSAMIN IS USED.         *
*                                                                             *
*   ALL QSAM TAPE PROCS DEFAULT TO STANDARD LABELS.                           *
*                                                                             *
*            L. S. BRAMMER       IBM CORPORATION  1975                        *
*******************************************************************************
```

```
* * * * * * * * *        EXTENSIONS  CONTROL CARD FORMAT    * * * * * * * * *

   CC 1-7      FUNCTION           PARAMETERS                                              CC 72

   $$DITTO      PRT         (MEMBER=MEMBNAME)(,RECSIZE=N...N)
   $$DITTO      PUN         (MEMBER=MEMBNAME)(,RECSIZE=N...N)
   $$DITTO      CQ
   $$DITTO      CQS         DECKTYPE=XXX,DECKNAME=X...X
   $$DITTO      QC          (STARTAFT=N...N)(,NLRECS=N...N)
   $$DITTO      QQ          (STARTAFT=N...N)(,NLRECS=N...N)
   $$DITTO      QP          (STARTAFT=N...N)(,NLRECS=N...N)
   $$DITTO      QD          (STARTAFT=N...N)(,NLRECS=N...N)
   $$DITTO      VP          (STARTKEY=KEYVAL)OR(STARTAFT=N...N)(,NLRECS=N...N)
   $$DITTO      VD          (STARTKEY=KEYVAL)OR(STARTAFT=N...N)(,NLRECS=N...N)
   $$DITTO      TRS         INPUT=DDNAME,RECSIZE=N...N,ARGUMENT=ARGVAL(,SCANPOS=N...N)
   $$DITTO      DRS         INPUT=DDNAME,BEGIN=CCCHH,END=CCCHH,RECSIZE=N...N,              X
   $$DITTO                  SCANTYPE=X,ARGUMENT=ARGVAL(,SCANPOS=N...N)

          PARAMETER                      DESCRIPTION

       INPUT=DDNAME             MUST BE TAPEIN,DISKIN, OR TAPEOUT
       SCANTYPE=X               SCAN FOR (K=KEY, D=DATA, E=EOF)
       BEGIN=CCCHH              CCCHH REL. TO 1ST TRK OF DATA SET
       END=CCCHH                CCCHH REL. TO 1ST TRK OF DATA SET
       SCANPOS=N...N            SCAN STARTING POSITION IN LREC
       ARGUMENT=ARGVAL          ARGVAL IS C'CHAR' OR X'HEX'
       RECSIZE=N...N            LOGICAL RECORD SIZE  1-99999 OR V (TRS)
       DECKTYPE=XXX             DECKTYPE (BAL, RPG, COB)
       DECKNAME=X...X           DECKNAME (0-3, 0-5, 0-8) CHAR.
       STARTKEY=KEYVAL          KEYVAL IS CHAR 'CHAR' C'CHAR' OR X'HEX'
       STARTAFT=N...N           START AFTER NTH LREC
       NLRECS=N...N             USE TO LIMIT NUMBER OF LRECS PROCESSED
       MEMBER=MEMBNAME          1-8 CHAR. OMIT TO GET ALL MEMBERS

NOTE: CONSOLE PROCEDURES (NEXT PAGE) MAY ALSO BE USED IN CONTROL CARD MODE. SUBMIT
      PARM.STEP1='JOBSTREAM' ON PROC EXEC AND //STEP1.SYSIN DD * PLUS CONTROL CARDS.
      PRT AND PUN RECSIZE= IS OPTIONAL(DEFAULT=80). SCANPOS= IS OPTIONAL(DEFAULT=1).
      DDNAMES USED BY PRT TO VD ARE FIXED AND CONTAINED IN PROCS. NO INPUT= NEEDED.
      ( ) DENOTES OPTIONAL PARAMETER. TRAILING , AND XIN72 ALLOWS PAPMS ON NEXT CARD.
      KEYVAL AND ARGVAL CHAR LIMIT IS 35. HEX LIMIT IS 50. CODE '' FOR ' IN VALUE.
      SYSPRINT, SYSPUNCH, AND QSAMOUT ARE NOT CLOSED TILL EOJ. FUNCTION CONTROL CARDS
      CAN SELECTIVELY PROCESS QSAMIN/VSAMIN DATA SETS. EOF CLOSES QSAMIN/VSAMIN AND
      ENDS THE FUNCTION. QSAMIN/VSAMIN ARE RE-OPENED WHEN A FUNCTION REQUESTS THEM.
      VSAMIN SUPPORTS BOTH KSDS AND ESDS VSAM DATASET FORMATS.

JOBSTREAM EXAMPLES

      //PRTPUN    JOB MSGLEVEL=1
      // EXEC DITTOPDS,PARM.STEP1='JOBSTREAM',
      //        DSNM='SYS1.PARMLIB'
      //STEP1.SYSIN DD *
      $$DITTO PRT
      $$DITTO PUN MEMBER=IEASYS00

      //DRS       JOB MSGLEVEL=1
      // EXEC  DITTODP,PARM.STEP1='JOBSTREAM',
      //        DSNM='SYS1.DITTOMAC'
      //STEP1.SYSIN DD *
      $$DITTO DRS INPUT=DISKIN,BEGIN=00000,END=00000,        X
      $$DITTO     SCANTYPE=D,RECSIZE=80,SCANPOS=10,           X
      $$DITTO     ARGUMENT=C'EJECT'
```

TO INITIATE OS/DITTO FROM THE CONSOLE ENTER A START COMMAND OF THE FORM; 'S DITTO---.PN'. REPLACE --- WITH OS/DITTO
PROCEDURE NAME SUFFIX; E.G. S DITTOUTL.P2,SER=IPLVOL WILL ALLOCATE DASD VOLUME 'IPLVOL'. THIS ALLOCATION SUPPORTS
FUNCTIONS LS THROUGH REN FOR THAT VOLUME. USE THE FOLLOWING PROCEDURE NAME, SYMBOLIC PAREMETER, AND FUNCTION USE
GUIDELINES TO CHOOSE THE CORRECT PROCEDURE AND FUNCTION.

PROCEDURE NAME	SYMBOLIC PARAMETER DEFAULT VALUES	FUNCTION USE GUIDELINES
DITTOPDS	IN=SYSDA,SER=,DSNM=,CLS=B,PUN=,FUN=	PRT,PUN
DITTODUM	DUNIT=SYSDA,DSER=,TUNIT=TAPE,TLAB=BLP,TSER=DUMTAP	DUM
DITTORES	DUNIT=SYSDA,DSER=,TUNIT=TAPE,TLAB=BLP,TSER=RESTAP	RES
DITTOUTL	IN=SYSDA,SER=,	LS,LV,LC,LP,CAT,UNC,SCR,REN
DITTOCQT	OUT=TAPE,CDSN=CQT.OUT,RDR=00C	CQ,CQS
DITTOCQD	OUT=SYSDA,ODSN=CQD.OUT,RDR=00C	CQ,CQS
DITTOQC	IN=SYSDA,CLS=B	QC
DITTOQCI	IN=SYSDA,CLS=,PUN=00D,FUN=I	QC
DITTOQTT	IN=TAPE,OUT=TAPE,ODSN=QTT.OUT	QQ
DITTOQTD	IN=TAPE,OUT=SYSDA,ODSN=QTD.OUT	QQ
DITTOQDD	IN=SYSDA,OUT=SYSDA,ODSN=QDD.OUT	QQ
DITTOQDT	IN=SYSDA,OUT=TAPE,ODSN=QDT.OUT	QQ
DITTOQP	IN=SYSDA,CLS=A,PRT=	QP,QD
DITTOVP	IN=,IDSN=,ISHR=SHR,ISER=,IAMP=,CLS=A,PRT=	VP,VD
DITTOLOG	OUT=TAPE,ODEN=,OTRTCH=	NOT A FUNCTION CODE

QSAMIN SYMBOLIC PARAMETER DEFAULT VALUES	QSAMOUT SYMBOLIC PARAMETER DEFAULT VALUES	UNIT RECORD SYMBOLIC PARAMETERS	
IDSN=,ISHR=SHR,IKEEP=KEEP	ONEW=NEW,CKEEP=KEEP,	RDR=CUU	SELECT SYSIN UNIT
ISER=,IFILENO=1,ILABEL=SL	OSER=,OFILENO=1,OLABEL=SL	PUN=CUU	SELECT SYSPUNCH UNIT
IBLKSIZ=,IRECFM=,ILRECL=	OBLKSIZ=,ORECFM=,OLRECL=,	CLS=	SYSOUT CLASS=NULL
	(DASD ONLY)	CLS=A	SYSOUT CLASS=A OR B
	OSPACE=TRK,OPRI=1,OSEC=1	PRT=	SYSPRINT UNIT=NULL
		PRT=CUU	SELECT SYSPRINT UNIT

SAMPLE START COMMANDS USING OPTIONAL SYMBOLIC PARAMETERS TO OVERRIDE DEFAULT VALUES.

COMMAND	EXPLANATION
S DITTOPDS.P0,DSNM=PDSLIB,CLS=,PUN=00D,FUN=I	SYSPUNCH ON 00D WITH INTERPRET
S DITTODUM.P1,DSER=SYSRES	DUMP SYSRES VOLUME TO TAPE (BLP)
S DITTORES.P2,DSER=NEWRES	RESTORE TAPE (BLP) TO NEWRES
S DITTOUTL.P3,SER=WORK01	SELECT WORK01 VOLUME FOR LS THROUGH REN
S DITTOCQT.P4,CSER=000000	CARD TO STANDARD LABEL TAPE NUMBER 000000
S DITTOCQD.P5,OSPACE=CYL	CARD TO DISK WITH CYL ALLOCATION
S DITTOQC.P6,IDSN=CARDS,ISER=111111	DASD TO CARD DSN=CARDS,VOL=SER=111111
S DITTOQCI.P7,IDSN=SCRCTAPE	TAPE TO CARD WITH INTERPRET(SL CATALOGED)
S DITTOQTT.P8,IDSN=CATLGDSN,ODSN=DATAPESL	TAPE TO TAPE (OSER=SCRTCH BY DEFAULT)
S DITTOQTD.P9,IDSN=CATLGDSN,OKEEP=CATLG	TAPE TO DASD (DISPOSITION=CATLG)
S DITTOQDD.P10,IDSN=CATLGDSN,OBLKSIZ=12000, ORECFM=FBS,OLRECL=80	DASD TO DASD (RE-BLOCK TO 12000)
S DITTOQDT.P11,IDSN=CATLGDSN,ODSN=TAPEDATA	DASD TO TAPE (SUPPLY OUTPUT DSNAME)
S DITTOQP.P12,IDSN=PDS(MEMBNAME)	DASD TO PRINTER (USE QSAM PDS SUPPORT)
S DITTOVP.P13,IDSN=KSDS	VSAM TO PRINTER (SELECT KSDS)
S DITTOLOG.P14,OUT=280	TAPEMARK LOGTAPE REEL ON 280 (NO REWIND)

NOTE: IF DSNAME (IDSN, ODSN, OR DSNM) CONTAINS SPECIAL CHARACTERS, ENCLOSE IN QUOTES AND CAPITALIZE ALPHA CHARACTERS
      WHEN KEYING THE START COMMAND. E.G. S DITTOPDS.P0,DSNM='SYS1.MACLIB'

TO MINIMIZE CONSOLE MESSAGES, YOU MAY ENTER WITH THE FUNCTION ALL OR PART OF THE REQUIRED PARAMETERS. A PARAMETER
ERROR WILL CAUSE DITTO TO REVERT TO THE TUTORIAL ROUTINES AND REQUEST REMAINING PARAMETERS IN A SINGLE REQUEST MODE.
A FUNCTION MAY BE ENDED BY REPLYING 'U' TO THE FIRST WTOR MESSAGE. A REPLY OF 'EOJ' WILL END THE FUNCTION AND DITTO.

SYSTEM UTILITY FUNCTIONS   PARAMETER REQUIREMENTS

```
PRINT ONE OR ALL MEMBERS OF A PDS               PRT,MEMBER/U,MEMBLRECL/U,EXECUTE
PUNCH ONE OR ALL MEMBERS OF A PDS               PUN,MEMBER/U,MEMBLRECL/U,EXECUTE
DUMP A DASD VOLUME TO TAPE                       DUM,EXECUTE
RESTORE A DUMP TAPE TO DASD                      RES,CPYVOLID/U,PURGE/U,EXECUTE
LIST SPACE AVAILABLE ON A DASD VOLUME           LS
LISTVTOC OF A DASD VOLUME                        LV,FORMATORDUMP/U,DSNAME/DDYYY/U,EXECUTE
LIST CATALOG (SYSCTLG - IPLVOL OR CVOL)         LC,NODE/U,EXECUTE
LIST PDS DIRECTORY                              LP,DSNAME,FORMATORDUMP/U,EXECUTE
CATALOG DSNAME                                  CAT,DSNAME,CVOL/U,DEVICE,VOLSER,EXECUTE
UNCATALOG DSNAME                                UNC,DSNAME,CVOL/U,EXECUTE
SCRATCH VTOC, DATA SET, OR MEMBER OF A PDS      SCR,DSNAMEORVTOC,PURGE/U,MEMBERORSYS/U,EXECUTE
RENAME DATA SET OR MEMBER                       REN,DSNAME,NEWNAME,MEMBNAME/U,EXECUTE
```

QSAM (CARD, DASD, AND SL/BLP TAPE) FUNCTIONS

```
CARD TO QSAM                                     CQ
CARD TO QSAM WITH SEQ. NUMBERS AND DECK NAME    CQS,DECKTYPE,DECKNAME
QSAM TO CARD (USE FUNC=I FOR 3525 INTERPRET)    QC,STARTAFT/U,NLRECS/U
QSAM TO QSAM (COMPATIBLE RECFM'S ASSUMED)       QQ,STARTAFT/U,NLRECS/U
```

QSAM/VSAM PRINT FUNCTIONS

```
QSAM TO PRINTER DEBLOCKED IN CHAR. FORMAT        QP,STARTAFT/U,NLRECS/U
QSAM TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP  QD,STARTAFT/U,NLRECS/U
VSAM TO PRINTER DEBLOCKED IN CHAR. FORMAT        VP,STARTYPE/U(,STARTKEY/STARTAFT),NLRECS/U
VSAM TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP  VD,STARTYPE/U(,STARTKEY/STARTAFT),NLRECS/U
```

```
/                   SLASH SHOWS OR CONDITION. CHOOSE ONE         *   U               USE THE DEFAULT OPTION, VALUE,OR FORMAT
MEMBER=MEMBNAME     1-8 CHAR. OMIT TO GET ALL MEMBERS            *   MEMBLRECL       NNN - MEMBER LRECL (DEFAULT IS 80)
EXECUTE             E - EXECUTE FUNCTION, U - TO CANCEL          *   CPYVOLID        C - CPYVOLID=YES, U - CPYVOLID=NO
PURGE               P - PURGE UNEXPIRED DSN'S, U - NO PURGE      *   FORMATORDUMP    F - FORMAT, D - DUMP, U - NEITHER(LIST)
DSNAME              44 CHAR MAX DATA SET NAME                    *   DDDYYY          DATE OPTION, * BY DSN'S LESS THAN DDDYY
NODE                LISTCTLG NODE OPTION (E.G. SYS1)             *   CVOL            C - CATLG CVOL OPTION, U - NOT A CVOL
DEVICE              DEVICE TYPE FOR CATALOG ENTRY(E.G.3330)      *   VOLSER          VOLSER LIST (1 - 5 MAX, DO NOT STRING)
DSNAMEORVTOC        'VTOC' - SCRATCH VTOC, DSNAME - SCR DSN      *   MEMBERORSYS     MEMBNAME - SCR MEMBER, S - SCRATCH SYS
NEWNAME             NEWNAME OF DSN OR MEMBER                     *   MEMBNAME        MEMBER TO BE RENAMED
DECKTYPE=XXX        DECKTYPE (BAL, RPG, COB)                     *   DECKNAME=X...X  DECKNAME (0-3, 0-5, 0-8) CHAR.
STARTAFT=N...N      START AFTER NTH LREC COUNT                   *   NLRECS=N...N    USE TO LIMIT NUMBER OF LRECS PROCESSED
STARTYPE            K-STARTKEY, C-STARTAFT, U-FIRST LREC         *   STARTKEY        X...X - KEY VALUE IS 1-35 CHAR MAX
( )                 DENOTES OPTIONAL PARAMETER                   *
```

```
* * * * * * * * *        FUNCTION CONTROL CARD FORMAT       * * * * * * * * * *


    CC 1-7          FUNCTION          PARAMETERS

    $$DITTO          CC
    $$DITTO          CCS       DECKTYPE=XXX,DECKNAME=X...X
    $$DITTO          CP
    $$DITTC          CD
    $$DITTO          CT        OUTPUT=DDNAME,BLKFACTOR=N.N
    $$DITTO          CTS       OUTPUT=DDNAME,BLKFACTOR=N.N,DECKTYPE=XXX,DECKNAME=X...X
    $$DITTO          TC        INPUT=DDNAME
    $$DITTO          TP        INPUT=DDNAME,NBLKS=N..N
    $$DITTC          TPD       INPUT=DDNAME,RECSIZE=N...N,NBLKS=N..N
    $$DITTC          TD        INPUT=DDNAME,NBLKS=N..N
    $$DITTO          TDD       INPUT=DDNAME,RECSIZE=N...N,NBLKS=N..N
    $$DITTC          TPV       INPUT=DDNAME,NBLKS=N..N
    $$DITTC          TDV       INPUT=DDNAME,NBLKS=N..N
    $$DITTC          TT        INPUT=DDNAME,OUTPUT=DDNAME,NBLKS=N..N
    $$DITTO          TTR       INPUT=DDNAME,OUTPUT=DDNAME,RECSIZE=N...N,BLKFACTOR=N.N
    $$DITTO          WTM       OUTPUT=DDNAME
    $$DITTO          REW       OUTPUT=DDNAME
    $$DITTC          RUN       OUTPUT=DDNAME
    $$DITTC          FSR       OUTPUT=DDNAME,NBLKS=N..N
    $$DITTO          BSR       OUTPUT=DDNAME,NBLKS=N..N
    $$DITTO          FSF       OUTPUT=DDNAME
    $$DITTO          BSF       OUTPUT=DDNAME
    $$DITTO          DP        INPUT=DDNAME,BEGIN=CCCHH,END=CCCHH
    $$DITTO          DD        INPUT=DDNAME,BEGIN=CCCHH,END=CCCHH
    $$DITTC          DPD       INPUT=DDNAME,BEGIN=CCCHH,END=CCCHH,RECSIZE=N...N
    $$DITTC          DDD       INPUT=DDNAME,BEGIN=CCCHH,END=CCCHH,RECSIZE=N...N
    $$DITTO          XXX
    $$DITTC          EOJ


          PARAMETER                      DESCRIPTION

        INPUT=DDNAME              MUST BE TAPEIN,DISKIN, OR TAPEOUT
        OUTPUT=DDNAME            MUST BE TAPEOUT OR TAPEIN
        BEGIN=CCCHH              CCCHH REL. TO 1ST TRK OF DATA SET
        END=CCCHH               CCCHH REL. TO 1ST TRK OF DATA SET
        NBLKS=N..N              NUMBER OF TAPE BLOCKS      1-9999
        RECSIZE=N...N           LOGICAL RECORD SIZE       1-99999
        DECKTYPE=XXX            DECKTYPE (BAL, RPG, COB)
        DECKNAME=X...X          DECKNAME (0-3, 0-5, 0-8) CHAR.
        BLKFACTOR=N.N           OUTPUT BLOCKING FACTOR       1-999


    NOTE. SUBMIT (PARM='JOBSTREAM') ON EXEC CARD TO DENOTE CONTROL CARD OPERATION.
          SUBMIT TAPEIN, TAPEOUT, & DISKIN DD CARDS IF USED IN INPUT= OR OUTPUT=.
          TAPEOUT IS OPENED AS OUTPUT; THEREFORE, A FILE PROTECT RING IS REQUIRED.
          TWO TAPE MARKS ARE WRITTEN ON TAPEOUT BY A CLOSE IN FUNCTIONS RUN & EOJ.
          DCB=(DEN=X,TRTCH=XX) MAY BE SPECIFIED ON TAPEIN AND TAPEOUT DD CARDS.
          CCCHH IS RELATIVE TO FIRST CCCHH OF DATA SET EXTENT; E.G. BEGIN=00000,
              END=00418 WILL ACCESS ENTIRE 3330 FIVE CYLINDER 3330 DATA SET.
          FROM CONSOLE CCCHH IS ABSOLUTE. OPERATOR MUST USE DSE FOR VALID CCCHH.
          NBLKS=N..N PARAMETER IS OPTIONAL. LAST PARAMETER MUST BE EOJ.
```

TO INITIATE OS/DITTO FROM THE CONSOLE ENTER A START COMMAND OF THE FORM; 'S DITTO---.PN'. REPLACE --- WITH OS/DITTO
PROCEDURE NAME SUFFIX; E.G. S DITTOTP.P2 WILL ALLOCATE ONE TAPE DRIVE AND SYSPRINT. THIS ALLOCATION WILL SUPPORT
DITTO FUNCTION TP AND ALL OTHER FUNCTIONS THAT REQUIRE NO MORE THAN ONE TAPE DRIVE AND SYSPRINT. TO SELECT INPUT
TAPE DRIVE AND PRINT ONLINE, ENTER OPTIONAL SYMBOLIC PARAMETERS; E.G. S DITTOTP.P2,IN=281,PRT=00F,CLS= . USE THE
FOLLOWING PROCEDURE NAME, SYMBOLIC PARAMETER, AND FUNCTION USE GUIDELINES TO CHOOSE CORRECT PROCEDURE AND FUNCTION.

PROCEDURE NAME	SYMBOLIC PARAMETER DEFAULT VALUES	FUNCTION USE GUIDELINES
DITTOCC	RDR=00C,PUN=00D,CLS=	CC,CCS,CP,CD,CCL
DITTOCCI	RDR=00C,PUN=00D	CC,CCS,CP,CD,CCL
DITTOCP	RDR=00C,PRT=,CLS=A	CP,CD,CCL
DITTOCT	RDR=00C,OUT=TAPE	CT,CTS,CCL
DITTOTC	IN=TAPE,PUN=00D,CLS=	TC,TP,TPD,TD,TDD
DITTOTCI	IN=TAPE,PUN=00D	TC,TP,TPD,TD,TDD
DITTOTP	IN=TAPE,PRT=,CLS=A	TP,TPD,TD,TDD,TPV,TDV,TRS,INT,ERT
DITTOTT	IN=TAPE,OUT=TAPE	TT,TTR
DITTOTRL	IN=TAPE,OUT=TAPE,PRT=00E,CLS=	TRL
DITTOTRS	IN=TAPE,PRT=00E,CLS=	TRS
DITTODP	IN=SYSDA,DSNM=,SER=,PRT=,CLS=A	DSE,DP,DD,DPD,DDD,DPL,DRS,SPS,EOF
DITTODID	NONE	DID,XXX

UNIT RECORD SYMBOLIC PARAMETERS		TAPE AND DISK SYMBOLIC PARAMETERS	
RDR=CUU	SELECT SYSIN UNIT	IN=CUU	SELECT TAPEIN UNIT
PUN=CUU	SELECT SYSPUNCH UNIT	IN=CUU	SELECT DISKIN UNIT
CLS=	SYSOUT CLASS=NULL	OUT=CUU	SELECT TAPEOUT UNIT
CLS=A	SYSOUT CLASS=A OR B	DSNM=	SPECIFY DISKIN DSN
PRT=	SYSPRINT UNIT=NULL	SER=	PROVIDE DISKIN VOLUME
PRT=CUU	SELECT SYSPRINT UNIT		SERIAL NUMBER

SAMPLE START COMMANDS USING OPTIONAL SYMBOLIC PARAMETERS TO OVERRIDE DEFAULT VALUES.

COMMAND	EXPLANATION
S DITTOCC.P0,RDR=00A,PUN=,CLS=B	SYSIN ON 00A, SYSPUNCH TO SYSOUT=B
S DITTOCCI.P1,RDR=00A	SYSIN ON 00A
S DITTOCP.P2,PRT=00F,CLS=	SYSPRINT ON 00F
S DITTOCT.P3,OUT=TAPE7	TAPEOUT ON 7-TRACK (NEED UNITNAME=TAPE7)
S DITTOTC.P4,IN=285,PUN=,CLS=B	TAPEIN ON 285, SYSPUNCH TO SYSOUT=B
S DITTOTCI.P5,PUN=00B	SYSPUNCH ON 00B
S DITTOTP.P6,IN=287,PRT=00F,CLS=	TAPEIN ON 287, SYSPRINT ON 00F
S DITTOTT.P7,IN=284,OUT=281	TAPEIN ON 284, TAPEOUT ON 281
S DITTOTRL.P8,PRT=,CLS=A	SYSPRINT TO SYSOUT=A
S DITTOTRS.P9,IN=270,PRT=,CLS=A	TAPEIN ON 270, SYSPRINT TO SYSOUT=A
S DITTODP.P10,IN=151,DSNM=DATASET,SER=333001	DISKIN ON 151, DSN=DATASET,VOL=SER=333001
S DITTODSN.P11,QUAL=SYS1,DSNM=DUMP	ALLOCATE QUALIFIED DSN=SYS1.DUMP

DITTOCCI AND DITTOTCI INTERPRET CARD OUTPUT. 3525 FEATURE AND OS JCL FUNC=I SUPPORT REQUIRED. SYSPUNCH MUST BE ONLINE.
TAPE CONTROL FUNCTIONS (INT,WTM,REW,RUN,FSR,BSR,FSF,BSF,ERT) ARE VALID WHEN TAPEIN OR TAPEOUT ALLOCATED BY PROCEDURE.
DISKIN IS OPENED AS INOUT TO ALLOW DISK WRITE FUNCTIONS DRL AND EOF.
XXX AND DID FUNCTIONS ARE VALID IN ALL PROCEDURES.

* * * * * * * * * * OS/DITTO CONSOLE OPERATION *** * * * * * * * * ***

TO MINIMIZE CONSOLE MESSAGES, YOU MAY ENTER WITH THE FUNCTION ALL OR PART OF THE REQUIRED PARAMETERS. A PARAMETER
ERROR WILL CAUSE DITTO TO REVERT TO THE TUTORIAL ROUTINES AND REQUEST REMAINING PARAMETERS IN A SINGLE REQUEST MODE.
A FUNCTION MAY BE ENDED BY REPLYING 'U' TO THE FIRST WTOR MESSAGE. A REPLY OF 'EOJ' WILL END THE FUNCTION AND DITTO.

CARD FUNCTIONS
Function	PARAMETER REQUIREMENTS
CARD TO CARD	CC
CARD TO CARD WITH SEQ. NUMBERS AND DECK NAME	CCS,DECKTYPE,DECKNAME
CARD TO PRINTER IN CHARACTER FORMAT	CP
CARD TO PRINTER IN CHARACTER AND HEX DUMP FORMAT	CD
CARD TO TAPE BLOCKED 1 TO 400	CT,CUUMM,BLKFACTOR
CARD TO TAPE RESEQUENCED	CTS,CUUMM,BLKFACTOR,DECKTYPE,DECKNAME
CANCEL CARD INPUT (FLUSH TO EOF)	CCL

TAPE FUNCTIONS
Function	Parameters
TAPE TO CARD BLOCKED OR UNBLOCKED	TC,CUUMM
TAPE TO PRINTER UNBLOCKED IN CHAR. FORMAT	TP,CUUMM,NRECS
TAPE TO PRINTER DEBLOCKED IN CHAR. FORMAT	TPD,CUUMM,RECSIZE,NRECS
TAPE TO PRINTER UNBLOCKED IN CHAR. AND HEX DUMP	TD,CUUMM,NRECS
TAPE TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP	TDD,CUUMM,RECSIZE,NRECS
TAPE TO PRINTER VARIABLE RECDS CHAR. FORMAT	TPV,CUUMM,NRECS
TAPE TO PRINTER VARIABLE RECDS CHAR. AND HEX DUMP	TDV,CUUMM,NRECS
TAPE RECORD SCAN	TRS,CUUMM,RECSIZE,SCANLNTH,SCANPOS
TAPE RECORD LOAD	TRL,CUUMM,CUUMM,NBLKS,COPYD
INITIALIZE TAPE	INT,CUUMM,VOLSER
TAPE TO TAPE (01 TO 99) FILES	TT,CUUMM,CUUMM,NFILES
TAPE TO TAPE REBLOCKED	TTR,CUUMM,CUUMM,RECSIZE,BLKFACTOR
WRITE TAPE MARK	WTM,CUUMM
REWIND TAPE	REW,CUUMM
REWIND AND UNLOAD TAPE	RUN,CUUMM
FORWARD SPACE RECORD	FSR,CUUMM,NBLKS
BACK SPACE RECORD	BSR,CUUMM,NBLKS
FORWARD SPACE FILE	FSF,CUUMM,NFILES
BACK SPACE FILE	BSF,CUUMM,NFILES
ERASE TAPE (DATA SECURITY ERASE - 3420 ONLY)	ERT,CUUMM

DISK FUNCTIONS
Function	Parameters
DISPLAY DATA SET EXTENT ON CONSOLE, USE FOR CCCHH	DSE
DISK TO PRINTER UNBLOCKED IN CHAR. FORMAT	DP,CCCHH,CCCHH
DISK TO PRINTER UNBLOCKED IN CHAR. AND HEX DUMP	DD,CCCHH,CCCHH
DISK TO PRINTER DEBLOCKED IN CHAR. FORMAT	DPD,CCCHH,CCCHH,RECSIZE
DISK TO PRINTER DEBLOCKED IN CHAR. AND HEX DUMP	DDD,CCCHH,CCCHH,RECSIZE
DISK RECORD LOAD - KEY AND/OR DATA	DRL,CCC-HH-RRR
DISK RECORD SCAN - PARTIAL KEY OR DATA OR EOF	DRS,CCCHH,CCCHH,SCANTYPE,RECSIZE,SCANLNTH
WRITE DISK EOF RECORD	EOF,CCC-HH-RRR
ALTER DISK IDENTIFICATION VOLUME LABEL (OFFLINE)	DID

LIST FUNCTIONS ON SYSPRINT	XXX
END OF JOB	EOJ

NBLKS=N..N	NUMBER OF TAPE BLOCKS 1-9999	*	
DECKTYPE=XXX	DECKTYPE (BAL, RPG, COB)	*	
BLKFACTOR=N.N	OUTPUT BLOCKING FACTOR 1-999	*	
COPYD=X	COPY FORWARD OR BACK (F OR B)	*	
NFILES=NN	NUMBER OF FILES (1-99)	*	
NRECS=N..N	OPTIONAL - (LIMITS PRINTING)	*	
SCANPOS=N...N	SCAN STARTING POSITION IN RECORD	*	

RECSIZE=N...N	LOGICAL RECORD SIZE 1-99999
DECKNAME=X...X	DECKNAME (0-3, 0-5, 0-8) CHAR.
SCANLNTH=NN	LENGTH OF SCAN ARGUMENT (1-35)
VOLSER=NNNNNN	NEW VOLUME SERIAL NUMBER LNTH=6
SCANTYPE=X	SCAN FOR (K=KEY, D=DATA, E=EOF)
RECSIZE=V	TRS VARIABLE LENGTH RECORDS ONLY

```
* * * * * * * * *    CONSOLE TAPE DENSITY & MODE VALUES   MM  * * * * * * * * * *
```

TRACK	DENSITY	PARITY	TRANSLATE	DATA CONVERT	MM
7	200	ODD	OFF	ON	10
7	200	EVEN	OFF	OFF	20
7	200	EVEN	ON	OFF	28
7	200	ODD	OFF	OFF	30
7	200	ODD	ON	OFF	38
7	556	ODD	OFF	ON	50
7	556	EVEN	OFF	OFF	60
7	556	EVEN	ON	OFF	68
7	556	ODD	OFF	OFF	70
7	556	ODD	ON	OFF	78
7	800	ODD	OFF	ON	90
7	800	EVEN	OFF	OFF	A0
7	800	EVEN	ON	OFF	A8
7	800	ODD	OFF	OFF	B0
7	800	ODD	ON	OFF	B8
9	800	SINGLE DENSITY			C0
9	1600	SINGLE DENSITY			C0
9	6250	SINGLE DENSITY			D0
9	800	DUAL DENSITY			C8
9	1600	DUAL DENSITY			C0
9	6250	DUAL DENSITY			D0

NOTE. MM PORTION OF CUUMM IS OPTIONAL - VALUE IS CO IF OMITTED

A USER INPUT TAPE ERROR CORRECTION ROUTINE IS AVAILABLE UNDER CONSOLE OPERATION.

CONTROL WILL BE PASSED TO THE ROUTINE FOR ALL INPUT TAPE DITTO FUNCTIONS IF OS

ERROR RECOVERY IS UNSUCCESSFUL. THE CONSOLE OPERATOR MAY EITHER IGNORE THE

ERROR (ACCEPTING THE RECORD AS READ), BYPASS (DROP) THE ERROR RECORD, OR PERFORM

CONSOLE ALTERATION TO THE RECORD LENGTH AND/OR DATA, BEFORE RETURNING CONTROL TO

THE ORIGINAL DITTO FUNCTION.

APPENDIX B

JS/DITTO EXTENSIONS JOBSTREAM CONTROL CARD EXAMPLES:


```
//SAMPXXX   JOB ('A=SX82'),'PAS3 BRAMMER',
// CLASS=A,REGION=46K,TIME=0010,MSGLEVEL=1
//   EXEC PGM=DSDITTO,PARM='JOBSTREAM'
//SYSPRINT DD SYSOUT=A
//SYSIN DD *
$$DITTO XXX
```

1. XXX JOBSTREAM EXAMPLE


```
//PRT#PUN   JOB ('A=SX82'),'PAS3 BRAMMER',
//   MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTOPDS,
//        PARM.STEP1='JOBSTREAM',
//     PUN=,CLS=B,FUN=I,
// DSNM='SYS1.DITTOMAC'
//STEP1.SYSIN DD *
$$DITTO PRT
$$DITTO PRT MEMBER=DITTODS1
$$DITTO PRT RECSIZE=40
$$DITTO PRT MEMBER=DITTODS1,RECSIZE=40
$$DITTO PUN MEMBER=DITTODS1
$$DITTO PUN MEMBER=DITTODS1,RECSIZE=40
$$DITTO PUN MEMBER=DITTODS1,RECSIZE=80
$$DITTO PUN
$$DITTO PUN                    RECSIZE=40
```

2. PRT PUN JOBSTREAM EXAMPLE


```
//CQ#CQS    JOB ('A=SX82'),'PAS3 BRAMMER',
//   MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTOCQD,
//        PARM.STEP1='JOBSTREAM',
//        OBLKSIZ=400,ORECFM=FB,OLRECL=80
//STEP1.SYSIN DD *
$$DITTO CQ
1234567890                      DATA CARD                 1234567890
1234567890                      DATA CARD                 1234567890
 /+ (IN C.C. 1)
$$DITTO  CQS DECKTYPE=BAL,DECKNAME=TES
1234567890                      DATA CARD                 1234567890
1234567890                      DATA CARD                 1234567890
```

3. CQ CQS JOBSTREAM EXAMPLE (CARD TO DASD)

```
//CQ#VB#SL JOB ('A=SX82'),'PAS3 BRAMMER',
//   MSGLEVEL=1,CLASS=G,REGION=100K
// EXEC DITTOCQT,
//       PARM.STEP1='JOBSTREAM',
//       OUT=2400,
//       OSER=004734,
//       ODSN='VARIABLE.BLOCKED',
//       OBLKSIZ=3128,ORECFM=VB,OLRECL=80
//STEP1.SYSIN DD *
$$DITTO  CQ
1234567890                          DATA CARD            1234567890
1234567890                          DATA CARD            1234567890
```

4. CQ JOBSTREAM EXAMPLE (CARD TO STANDARD LABEL TAPE - VARIABLE BLOCKED)


```
//QCI#VB   JOB ('A=SX82'),'PAS3 BRAMMER',
//   MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTOQCI,
//       PARM.STEP1='JOBSTREAM',
//       IDSN='VARIABLE.BLOCKED',
//       IN=2400,
//       ISER=004734
//STEP1.SYSIN DD *
$$DITTO QC NLRECS=5
$$DITTO QC STARTAFT=5,NLRECS=5
$$DITTO QC STARTAFT=5
$$DITTO QC
```

5. QC JOBSTREAM EXAMPLE (INTERPRET FUNC=I DEFAULT IN PROC - VAR BLK INPUT)


```
//QQ#TT    JOB ('A=SX82'),'PAS3 BRAMMER',
//   MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTOQTT,
//       PARM.STEP1='JOBSTREAM',
//       IDSN='VARIABLE.BLOCKED',
//       ISER=004734,
//       IN=2400,
//       OUT=2400,
//       OBLKSIZ=6248,ORECFM=VB,OLRECL=80
//STEP1.SYSIN DD *
$$DITTO QQ NLRECS=2
$$DITTO QQ NLRECS=2,STARTAFT=3
$$DITTO QQ
```

6. QQ JOBSTREAM EXAMPLE (VB TO VB RE-BLOCKED TO STANDARD LABEL SCRTCH TAPE)

```
//QP#QD    JOB ('A=SX82'),'PAS3 BRAMMER',
//  MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTOQP,
//      PARM.STEP1='JOBSTREAM',
// IDSN='SYS1.DITTOMAC(DITTODS2)'
//STEP1.SYSIN DD *
$$DITTO QP NLRECS=15
$$DITTO QP STARTAFT=15,NLRECS=20
$$DITTO QP STARTAFT=100
$$DITTO QP
$$DITTO QD NLRECS=2
$$DITTO QD STARTAFT=10,NLRECS=3
$$DITTO QD STARTAFT=400


7. QP QD JOBSTREAM EXAMPLE (PDS MEMBER INPUT)


//VP#VD    JOB ('A=SX82'),'PAS3 BRAMMER',
//  MSGLEVEL=1,CLASS=G,REGION=100K
// EXEC DITTOVP,
//      PARM.STEP1='JOBSTREAM',
// IDSN='ESDS06'
//STEP1.SYSIN DD *
$$DITTO VP
$$DITTO VP NLRECS=15
$$DITTO VP STARTAFT=10
$$DITTO VP STARTAFT=10,NLRECS=20
$$DITTO VD NLRECS=2
$$DITTO VD NLRECS=2
// EXEC DITTOVP,
//      PARM.STEP1='JOBSTREAM',
// IDSN='KSDS06'
//STEP1.SYSIN DD *
$$DITTO VP NLRECS=5
$$DITTO VP STARTAFT=10
$$DITTO VP STARTAFT=01,NLRECS=20
$$DITTO VD NLRECS=2
$$DITTO VD NLRECS=2
$$DITTO VP
$$DITTO VP STARTKEY=005
$$DITTO VP STARTKEY=009,NLRECS=1
$$DITTO VP STARTKEY=01,NLRECS=1
$$DITTO VP STARTKEY=X'F0F1F6',NLRECS=1
$$DITTO VP STARTKEY='019'
$$DITTO VP STARTKEY='0''0'


3. VP VD JOBSTREAM EXAMPLE (ESDS AND KSDS)
```

```
//SAMPTRS  JOB ('A=SX82'),'PAS3 BRAMMER',
//  MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTOTP,PARM.STEP1='JOBSTREAM',IN=2400
//STEP1.SYSIN DD *
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,ARGUMENT=C'CONSOLE',SCANPOS=1
$$DITTO REW OUTPUT=TAPEIN
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,ARGUMENT=C'CONSOLE'
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,ARGUMENT=C'PL,X''50''',SCANPOS=16
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,ARGUMENT=C'********************',X
$$DITTO     SCANPOS=1
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,                              X
$$DITTO ARGUMENT=X'C3',SCANPOS=1
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,                              X
$$DITTO              SCANPOS=1,                                   X
$$DITTO ARGUMENT=X'40404040404040404040'
$$DITTO   TRS   INPUT=TAPEIN,RECSIZE=80,ARGUMENT=C'END',SCANPOS=10
$$DITTO REW OUTPUT=TAPEIN
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,ARGUMENT=X'C3'
$$DITTO TRS INPUT=TAPEIN,RECSIZE=80,ARGUMENT=X'C3C'
```

9. TRS JOBSTREAM EXAMPLE (TAPE RECORD SCAN FROM CONTROL CARDS)


```
//SAMPTRSV JOB ('A=SX82'),'PAS3 BRAMMER',
//  MSGLEVEL=1,CLASS=G,REGION=100K
// EXEC DITTOTP,PARM.STEP1='JOBSTREAM',IN=2400
//STEP1.SYSIN DD *
$$DITTO TRS INPUT=TAPEIN,RECSIZE=V,ARGUMENT=C'DE37999',SCANPOS=10
```

10. TRS VARIABLE BLOCKED JOBSTREAM EXAMPLE (CONTROL CARD SCAN)


```
//SAMPDRS  JOB ('A=SX82'),'PAS3 BRAMMER',
//  MSGLEVEL=1,CLASS=G,REGION=128K
// EXEC DITTODP,PARM.STEP1='JOBSTREAM',DSNM='SYS1.DITTOMAC'
//STEP1.SYSIN DD *
$$DITTO DRS BEGIN=00000,END=00000,RECSIZE=80,SCANPOS=1,SCANTYPE=D,  X
$$DITTO        INPUT=DISKIN,                                        X
$$DITTO ARGUMENT=C'MAIN'
$$DITTO DRS BEGIN=00000,END=00000,RECSIZE=80,SCANTYPE=D,INPUT=DISKIN, X
$$DITTO ARGUMENT=C'MAIN'
$$DITTO DRS BEGIN=00000,END=00003,SCANTYPE=E,INPUT=DISKIN
$$DITTO DRS BEGIN=00000,END=00003,SCANTYPE=E,INPUT=DISKIN,
$$DITTO DRS BEGIN=00000,END=00003,RECSIZE=80,SCANPOS=1,SCANTYPE=E,  X
$$DITTO    INPUT=DISKIN
$$DITTO DRS BEGIN=00000,END=00000,RECSIZE=80,SCANPOS=1,SCANTYPE=K,  X
$$DITTO        INPUT=DISKIN,                                        X
$$DITTO ARGUMENT=X'FFFFFFFFFFFFFFFF'
$$DITTO DRS BEGIN=00000,END=00000,                                 X
$$DITTO        INPUT=DISKIN,                                       X
$$DITTO  SCANPOS=1,                                                X
$$DITTO  SCANTYPE=K,                                               X
$$DITTO ARGUMENT=X'FFFFFFFFFFFFFFFF'
```

11. DRS JOBSTREAM EXAMPLE (DISK RECORD SCAN FROM CONTROL CARDS)

APPENDIX C

=ILE # 1 OF 5758-CDA PID TAPE (JOBS INS#LIB, INS#PROC, AND ADDPROC).

```
//INS#LIB   JOB ('A=SX82'),'PAS3 BRAMMER',                          LI000100
//  MSGLEVEL=1,CLASS=G,REGION=100K                                  LI000200
//*    THIS JOB ALLOCATES A LIBRARY ON VOL=REF=SYS1.LINKLIB. IT WILL BE   LI000300
//*      USED TO CONTAIN NCAL LOAD MODULES.                         LI000400
//*    A COPY LIBRARY (SYS1.DITTOMAC) WILL BE ALLOCATED ON VOL=REF=  LI000500
//*      SYS1.DITTOLIB BY JOB 'DITTOMAC' LATER IN THE INSTALL STREAM  LI000600
//SCRSTEP EXEC  PGM=IEHPROGM                                        LI000700
//DD1 DD UNIT=SYSDA,DISP=SHR,                                       LI000800
//         VOL=REF=SYS1.LINKLIB                                     LI000900
//SYSPRINT  DD  SYSOUT=A                                            LI001000
//SYSIN  DD  *                                                      LI001100
  SCRATCH DSNAME=SYS1.DITTOLIB,VOL=3330=VSIRES,PURGE               LI001200
  SCRATCH DSNAME=SYS1.DITTOLIB,VOL=3330=VSLIBS,PURGE               LI001210
  SCRATCH DSNAME=SYS1.DITTOLIB,VOL=2314=STEVEB,PURGE               LI001300
 UNCATLG DSNAME=SYS1.DITTOLIB                                      LI001400
/*                                                                 LI001500
// EXEC PGM=IEFBR14                                                LI001600
//LKED.SYSLMOD DD DSN=SYS1.DITTOLIB,UNIT=SYSDA,                    LI001700
//          DISP=(NEW,CATLG),SPACE=(CYL,(2,1,10)),                 LI001800
//       VOL=REF=SYS1.LINKLIB                                      LI001900
//                                                                 LI002000
//INS#PROC JOB ('A=SX82'),'PAS3 BRAMMER',                          IN002100
//  MSGLEVEL=1,CLASS=G,REGION=128K                                 IN002200
//      EXEC PGM=IEBUPDTE,PARM=NEW                                 IN002300
//SYSUT2 DD DSN=SYS1.PROCLIB,DISP=SHR                              IN002400
//SYSPRINT DD SYSOUT=A                                             IN002500
//SYSIN DD DATA,DCB=BLKSIZE=80,DLM=AA                              IN002600
./ ADD NAME=DITTODSR,LIST=ALL                                     IN002700
//EXECSTEP EXEC PGM=IEFBR14                                        IN002800
//*         THIS PROC IS REFERENCED BY THE DITTO INSTALLATION STREAM   IN002900
//*         ITS USE IS TO CONTROL (FROM ONE SOURCE) THE DATA SET NAMES  IN003000
//*         REFERRED TO AND STORED INTO BY THE INSTALLATION JOBSTREAM  IN003100
//DITTOMAC DD DSN=SYS1.DITTOMAC,DISP=SHR                           IN003200
//*         CONCATENATE TO MACLIB FOR COPY FACILITY                IN003300
//LOADLIB  DD DSN=SYS1.DITTOLIB,DISP=SHR                           IN003400
//*         LOAD MODULE LIBRARY                                    IN003500
//*         ALSO STORE OBJECT CODE ( NCAL ) FOR LATER RE-LINKEDIT  IN003600
//LINKLIB  DD DSN=SYS1.LINKLIB,DISP=SHR                            IN003700
//*         WILL BE USED FOR APF MODULES THAT MUST BE IN AN        IN003800
//*         AUTHORIZED LIBRARY                                     IN003900
//PROCLIB  DD DSN=SYS1.PROCLIB,DISP=SHR                            IN004000
//*         PROCLIB: USED BY JOB 'ADDPROC'                         IN004100
./ ADD NAME=DITUPDTE,LIST=ALL                                     IN004200
//IEFPROC  PROC TUNIT=2400,ISER=ARDTAP,OSER=CDA001,                IN004300
//         IDSN='OSDITTO.N5758ARD.V02M00',                         IN004400
//         ODSN='OSDITTO.N5758CDA.V01M00'                          IN004500
//STEP1 EXEC PGM=IEBUPDTE,PARM=MOD                                 IN004600
//SYSPRINT DD SYSOUT=A                                             IN004700
//SYSUT1 DD DSN=&IDSN,UNIT=&TUNIT,                                 IN004800
//       DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80),                     IN004900
//       DISP=(SHR,KEEP),                                          IN005000
//       VOL=SER=&ISER,                                            IN005100
//       LABEL=(,BLP)                                              IN005200
//SYSUT2 DD DSN=&ODSN,UNIT=&TUNIT,                                 IN005300
//       DCB=(BLKSIZE=1600,RECFM=FB,LRECL=80),                     IN005400
//       DISP=(NEW,PASS),                                          IN005500
```

```
//        VOL=SER=&DSER,                                              IN005600
//        LABEL=(,BLP)                                                IN005700
./ ENDUP                                                             IN005800
AA                                                                   IN005900
//                                                                   IN006000
//ADDPROC   JOB ('A=SX82'),'PAS3 BRAMMER',                           AD006100
// MSGLEVEL=1,CLASS=G,REGION=100K                                    AD006200
//PROCSTEP EXEC DITTODSR                                             AD006300
//STEP1 EXEC PGM=IEBUPDTE,PARM=NEW                                   AD006400
//SYSUT2 DD DSN=*.PROCSTEP.EXECSTEP.PROCLIB,DISP=SHR                 AD006500
//SYSPRINT DD SYSOUT=A                                               AD006600
//SYSIN DD DATA,DCB=BLKSIZE=80,DLM=AA                                AD006700
./ ADD NAME=DITTOPDS,LIST=ALL                                        AD006800
//IEFPROC  PROC IN=SYSDA,SER=,DSNM=,CLS=B,PUN=,FUN=                  AD006900
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                             AD007000
//DITIN    DD UNIT=SYSDA,SPACE=(TRK,(1,1)),DSN=&CTLCARD              AD007100
//DITPRINT DD SYSOUT=A                                               AD007200
//SYSUT1   DD  DISP=SHR,UNIT=&IN,VOL=SER=&SER,DSN=&DSNM              AD007300
//SYSUT2   DD SYSOUT=A                 USE FOR PRINT                 AD007400
//SYSPUNCH DD SYSOUT=&CLS,                        USE FOR PUNCH      AD007500
//            DCB=FUNC=&FUN,                USE FOR 3525 INTERPRET   AD007600
//            UNIT=&PUN                 USE FOR ONLINE               AD007700
//*          (REMOVE DCB=FUNC IF IT CAUSES JCL ERROR)               AD007800
//SYSPRINT DD SYSOUT=A                                               AD007900
//SYSUDUMP DD SYSOUT=A                                               AD008000
./ ADD NAME=DITTODUM,LIST=ALL                                        AD008100
//IEFPROC  PROC DUNIT=SYSDA,DSER=,TDISP=NEW,TUNIT=TAPE,             AD008200
//            TLAB=BLP,TDEN=,TDSN=BACKUP,TSER=DUMTAP                 AD008300
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                             AD008400
//DITPRINT DD SYSOUT=A                                               AD008500
//DITIN    DD UNIT=SYSDA,SPACE=(TRK,(1,1)),DSN=&CTLCARD              AD008600
//DISK     DD DISP=OLD,UNIT=&DUNIT,VOL=SER=&DSER                     AD008700
//TAPE     DD DISP=(&TDISP,KEEP),UNIT=&TUNIT,VOL=SER=&TSER,         AD008800
//            LABEL=(1,&TLAB),DCB=DEN=&TDEN,DSN=&TDSN               AD008900
//SYSPRINT DD SYSOUT=A                                               AD009000
//SYSUDUMP DD SYSOUT=A                                               AD009100
./ ADD NAME=DITTORES,LIST=ALL                                        AD009200
//IEFPROC  PROC DUNIT=SYSDA,DSER=,TDISP=SHR,TUNIT=TAPE,             AD009300
//            TLAB=BLP,TDEN=,TDSN=BACKUP,TSER=RESTAP                 AD009400
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                             AD009500
//DITPRINT DD SYSOUT=A                                               AD009600
//DITIN    DD UNIT=SYSDA,SPACE=(TRK,(1,1)),DSN=&CTLCARD              AD009700
//DISK     DD DISP=OLD,UNIT=&DUNIT,VOL=SER=&DSER                     AD009800
//TAPE     DD DISP=(&TDISP,KEEP),UNIT=&TUNIT,VOL=SER=&TSER,         AD009900
//            LABEL=(1,&TLAB),DCB=DEN=&TDEN,DSN=&TDSN               AD010000
//SYSPRINT DD SYSOUT=A                                               AD010100
//SYSUDUMP DD SYSOUT=A                                               AD010200
./ ADD NAME=DITTOUTL,LIST=ALL                                        AD010300
//IEFPROC  PROC IN=SYSDA,SER=                                        AD010400
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                             AD010500
//DITTODD1 DD  DISP=OLD,UNIT=&IN,VOL=SER=&SER                        AD010600
//DITPRINT DD SYSOUT=A                                               AD010700
//DITIN    DD UNIT=SYSDA,SPACE=(TRK,(1,1)),DSN=&CTLCARD              AD010800
//SYSPRINT DD SYSOUT=A                                               AD010900
//SYSUDUMP DD SYSOUT=A                                               AD011000
./ ADD NAME=DITTOCQT,LIST=ALL                                        AD011100
//IEFPROC  PROC ODSN='CQT.OUT',OUT=TAPE,ONEW=NEW,OKEEP=KEEP,OSER=,  AD011200
//            OFILENO=1,OLABEL=SL,                                   AD011300
```

```
//           ORECFM=,OLRECL=,OBLKSIZ=,                                    AD011400
//           RDR=OOC                                                      AD011500
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                  AD011600
//QSAMOUT  DD DSN=&ODSN,UNIT=&OUT,DISP=(&ONEW,&OKEEP),VOL=SER=&OSER,      AD011700
//           LABEL=(&OFILENO,&OLABEL),                                    AD011800
//           DCB=(RECFM=&ORECFM,LRECL=&OLRECL,BLKSIZE=&OBLKSIZ)          AD011900
//SYSPRINT DD SYSOUT=A                                                    AD012000
//SYSUDUMP DD SYSOUT=A                                                    AD012100
//SYSIN DD UNIT=&RDR,DCB=(BLKSIZE=80,RECFM=F)                             AD012200
./ ADD NAME=DITTOCQD,LIST=ALL                                            AD012300
//IEFPROC  PROC ODSN='CQD.OUT',OUT=SYSDA,ONEW=NEW,OKEEP=KEEP,OSER=,       AD012400
//           OSPACE=TRK,OPRI=1,OSEC=1,                                    AD012500
//           ORECFM=,OLRECL=,OBLKSIZ=,                                    AD012600
//           RDR=OOC                                                      AD012700
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                  AD012800
//QSAMOUT  DD DSN=&ODSN,UNIT=&OUT,DISP=(&ONEW,&OKEEP),VOL=SER=&OSER,      AD012900
//           SPACE=(&OSPACE,(&OPRI,&OSEC)),'                             AD013000
//           DCB=(RECFM=&ORECFM,LRECL=&OLRECL,BLKSIZE=&OBLKSIZ)          AD013100
//SYSPRINT DD SYSOUT=A                                                    AD013200
//SYSUDUMP DD SYSOUT=A                                                    AD013300
//SYSIN DD UNIT=&RDR,DCB=(BLKSIZE=80,RECFM=F)                             AD013400
./ ADD NAME=DITTOQC,LIST=ALL                                             AD013500
//IEFPROC  PROC IDSN=,IN=SYSDA,ISHR=SHR,IKEEP=KEEP,ISER=,                 AD013600
//           CLS=B,PUN=,                                                  AD013700
//           IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=                AD013800
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                  AD013900
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,       AD014000
//           LABEL=(&IFILENO,&ILABEL),                                    AD014100
//           DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)          AD014200
//SYSPUNCH DD SYSOUT=&CLS,                         USE FOR PUNCH          AD014300
//           UNIT=&PUN                             USE FOR ONLINE         AD014400
//SYSPRINT DD SYSOUT=A                                                    AD014500
//SYSUDUMP DD SYSOUT=A                                                    AD014600
./ ADD NAME=DITTOQCI,LIST=ALL                                            AD014700
//IEFPROC  PROC IDSN=,IN=SYSDA,ISHR=SHR,IKEEP=KEEP,ISER=,                 AD014800
//           CLS=,PUN=I,PUN=OOC,                                          AD014900
//           IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=                AD015000
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                  AD015100
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,       AD015200
//           LABEL=(&IFILENO,&ILABEL),                                    AD015300
//           DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)          AD015400
//SYSPUNCH DD SYSOUT=&CLS,                         USE FOR PUNCH          AD015500
//           DCB=FUNC=&FUN,                        USE FOR 3525 INTERPRET AD015600
//           UNIT=&PUN                             USE FOR ONLINE         AD015700
//*          (REMOVE DCB=FUNC IF IT CAUSES JCL ERROR)                     AD015800
//SYSPRINT DD SYSOUT=A                                                    AD015900
//SYSUDUMP DD SYSOUT=A                                                    AD016000
./ ADD NAME=DITTOQTT,LIST=ALL                                            AD016100
//IEFPROC  PROC IDSN=,IN=TAPE,ISHR=SHR,IKEEP=KEEP,ISER=,                  AD016200
//           IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=,               AD016300
//           ODSN='QTT.OUT',OUT=TAPE,ONEW=NEW,OKEEP=KEEP,OSER=,           AD016400
//           OFILENO=1,OLABEL=SL,                                         AD016500
//           ORECFM=,OLRECL=,OBLKSIZ=                                    AD016600
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                  AD016700
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,       AD016800
//           LABEL=(&IFILENO,&ILABEL),                                    AD016900
//           DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)          AD017000
//QSAMOUT  DD DSN=&ODSN,UNIT=&OUT,DISP=(&ONEW,&OKEEP),VOL=SER=&OSER,      AD017100
```

```
//            LABEL=(&OFILENO,&OLABEL),                                    AD0172C0
//            DCB=(RECFM=&ORECFM,LRECL=&OLRECL,BLKSIZE=&OBLKSIZ)          AD017300
//SYSPRINT DD SYSOUT=A                                                    AD017400
//SYSUDUMP DD SYSOUT=A                                                    AD017500
./ ADD NAME=DITTOQTD,LIST=ALL                                            AD017600
//IEFPROC  PROC IDSN=,IN=TAPE,ISHR=SHR,IKEEP=KEEP,ISER=,                 AD017700
//            IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=,              AD017800
//            ODSN='QTD.OUT',OUT=SYSDA,ONEW=NEW,OKEEP=KEEP,OSER=,        AD017900
//            OSPACE=TRK,OPRI=1,OSEC=1,                                   AD018000
//            ORECFM=,OLRECL=,OBLKSIZ=                                    AD018100
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                 AD018200
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,      AD018300
//            LABEL=(&IFILENO,&ILABEL),                                   AD018400
//            DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)         AD018500
//QSAMOUT  DD DSN=&ODSN,UNIT=&OUT,DISP=(&ONEW,&OKEEP),VOL=SER=&OSER,     AD018600
//            SPACE=(&OSPACE,(&OPRI,&OSEC)),                              AD018700
//            DCB=(RECFM=&ORECFM,LRECL=&OLRECL,BLKSIZE=&OBLKSIZ)         AD018800
//SYSPRINT DD SYSOUT=A                                                    AD018900
//SYSUDUMP DD SYSOUT=A                                                    AD019000
./ ADD NAME=DITTOQDD,LIST=ALL                                            AD019100
//IEFPROC  PROC IDSN=,IN=SYSDA,ISHR=SHR,IKEEP=KEEP,ISER=,               AD019200
//            IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=,              AD019300
//            ODSN='QDD.OUT',OUT=SYSDA,ONEW=NEW,OKEEP=KEEP,OSER=,        AD019400
//            OSPACE=TRK,OPRI=1,OSEC=1,                                   AD019500
//            ORECFM=,OLRECL=,OBLKSIZ=                                    AD019600
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                 AD019700
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,      AD019800
//            LABEL=(&IFILENO,&ILABEL),                                   AD019900
//            DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)         AD020000
//QSAMOUT  DD DSN=&ODSN,UNIT=&OUT,DISP=(&ONEW,&OKEEP),VOL=SER=&OSER,     AD020100
//            SPACE=(&OSPACE,(&OPRI,&OSEC)),                              AD020200
//            DCB=(RECFM=&ORECFM,LRECL=&OLRECL,BLKSIZE=&OBLKSIZ)         AD020300
//SYSPRINT DD SYSOUT=A                                                    AD020400
//SYSUDUMP DD SYSOUT=A                                                    AD020500
./ ADD NAME=DITTOQDT,LIST=ALL                                            AD020600
//IEFPROC  PROC IDSN=,IN=SYSDA,ISHR=SHR,IKEEP=KEEP,ISER=,               AD020700
//            IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=,              AD020800
//            ODSN='QDT.OUT',OUT=TAPE,ONEW=NEW,OKEEP=KEEP,OSER=,         AD020900
//            OFILENO=1,OLABEL=SL,                                        AD021000
//            ORECFM=,OLRECL=,OBLKSIZ=                                    AD021100
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                 AD021200
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,      AD021300
//            LABEL=(&IFILENO,&ILABEL),                                   AD021400
//            DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)         AD021500
//QSAMOUT  DD DSN=&ODSN,UNIT=&OUT,DISP=(&ONEW,&OKEEP),VOL=SER=&OSER,     AD021600
//            LABEL=(&OFILENO,&OLABEL),                                   AD021700
//            DCB=(RECFM=&ORECFM,LRECL=&OLRECL,BLKSIZE=&OBLKSIZ)         AD021800
//SYSPRINT DD SYSOUT=A                                                    AD021900
//SYSUDUMP DD SYSOUT=A                                                    AD022000
./ ADD NAME=DITTOQP,LIST=ALL                                             AD022100
//IEFPROC  PROC IDSN=,IN=SYSDA,ISHR=SHR,IKEEP=KEEP,ISER=,               AD022200
//            CLS=A,PRT=,                                                 AD022300
//            IFILENO=1,ILABEL=SL,IRECFM=,ILRECL=,IBLKSIZ=              AD022400
//STEP1    EXEC  PGM=OSDITTO,REGION=100K                                 AD022500
//QSAMIN   DD DSN=&IDSN,UNIT=&IN,DISP=(&ISHR,&IKEEP),VOL=SER=&ISER,      AD022600
//            LABEL=(&IFILENO,&ILABEL),                                   AD022700
//            DCB=(RECFM=&IRECFM,LRECL=&ILRECL,BLKSIZE=&IBLKSIZ)         AD022800
//SYSPUNCH DD SYSOUT=B                                                    AD022900
```

```
//SYSPRINT  DD  SYSOUT=&CLS,UNIT=&PRT                                    AD023000
//SYSUDUMP DD SYSOUT=A                                                   AD023100
./ ADD NAME=DITTOVP,LIST=ALL                                            AD023200
//IEFPROC   PROC IDSN=,IN=,ISHR=SHR,ISER=,IAMP=,CLS=A,PRT=              AD023300
//STEP1     EXEC  PGM=OSDITTO,REGION=100K                               AD023400
//VSAMIN    DD DSN=&IDSN,UNIT=&IN,DISP=&ISHR,VOL=SER=&ISER,AMP=&IAMP    AD023500
//SYSPRINT DD SYSOUT=&CLS,UNIT=&PRT                                     AD023600
//SYSUDUMP DD SYSOUT=A                                                   AD023700
./ ADD NAME=DITTOLOG,LIST=ALL                                           AD023800
//IEFPROC   PROC OUT=TAPE,ODEN=3,OTRTCH=                                AD023900
//STEP1 EXEC PGM=DITTOLOG                                               AD024000
//*        DITTOLOG RUNS KEY=0 WHEN AUTHORIZED TO DO SO BY:             AD024100
//*                 1. PLACING MODULE NAME DITTOLOG IN THE IEELINKT     AD024200
//*                    LIST FOR KEY=0 MODULES                           AD024300
//*                 2. IS STARTED FROM THE CONSOLE                      AD024400
//*        KEY=0 IS REQUIRED TO ALLOW WRITING A TAPEMARK ON A LOG TAPE  AD024500
//*            WITHOUT REWINDING THE TAPE. THIS IS USEFUL FOR PROVIDING AD024600
//*            AN END OF FILE ON LOGTAPE REELS THAT ARE NOT CLOSED      AD024700
//*            DUE TO A SYSTEM FAILURE.                                 AD024800
//*        THE OPERATIONAL PROCEDURE FOLLOWS:                           AD024900
//*                 1. AFTER A FAILURE AND BEFORE RE-IPL, TAKE THE      AD025000
//*                    LOGTAPE DRIVE OUT OF READY STATE (DO NOT REWIND) AD025100
//*                    THIS CAN BE DONE BY DEPRESSING RESET.            AD025200
//*                 2. RE-IPL THE SYSTEM                                AD025300
//*                 3. S DITTOLOG,OUT=280 (WHERE 280 IS THE LOGTAPE)    AD025400
//*                    ODEN DEFAULTS TO 3 (1600 BPI)                    AD025500
//*                        TO OVERRIDE USE ODEN=4 (E.G. 6250 BPI)       AD025600
//*                        AN ODEN VALUE IS REQUIRED (NULL = ERROR)     AD025610
//*                    OTRTCH DEFAULTS TO NULL (NO 7 TRK)               AD025700
//*                        TO OVERRIDE USE OTRTCH=ET (EVEN & TRAN.)     AD025800
//*                 4. AFTER DEVICE ALLOCATION AND A MOUNT MESSAGE HAVE AD025900
//*                    OCCURED, READY THE LOGTAPE DRIVE. THIS CAN BE    AD026000
//*                    DONE BY DEPRESSING START.                        AD026100
//*                 5. A SINGLE TAPEMARK WILL BE WRITTEN ON THE LOGTAPE.AD026200
//*                 6. IF A WRITE ERROR OCCURS THE OPERATOR WILL BE     AD026300
//*                    NOTIFIED AND GIVEN THE OPTION TO RETRY OR ABORT. AD026400
//LOGTAPE DD UNIT=&OUT,VOL=SER=LOGTAP,DSN=LOGTAPE,DISP=SHR,             AD026500
//           LABEL=(,BLP),DCB=(DEN=&ODEN,TRTCH=&OTRTCH)                 AD026600
//SYSUDUMP DD SYSOUT=A                                                   AD026700
./ ENDUP                                                                AD026800
AA                                                                      AD026900
//                                                                      AD027000
```

APPENDIX D

AUTHORIZED PROGRAM FACILITY AND DITTOLOG (KEY=0) REQUIREMENTS

OPERATING SYSTEM AND RELEASE	AC=1 Exec. Modules	Reside in Auth. Library	PROGRAM PROPERTIES AND LINK TABLE ENTRIES					
			Module Name	Csect Name	Section	Module(s)	Library	Entry Function
MFT and MVT	NO	NO	DITTOLOG	IEEVLNKT	First	IEEVLNKT	LINKLIB	KEY=0
VS1 (R1-R3)	YES	YES	DITTOLOG	IEEVLNKT	First	IEEVLNKT	LINKLIB	KEY=0
VS1 (R4)	YES	YES	DITTOLOG OSDITTO	IEEVLNKT IEFSDPPT	First Fourth	IEEVRCTL IEFSD161 IEF161SD	LINKLIB LINKLIB LINKLIB	KEY=0 APF APF
VS2 (R1.7)	YES	YES	DITTOLOG	IEEVLNKT	First	IEEVLNKT	LPALIB	KEY=0
VS2 (R2-R3)	YES	YES	DITTOLOG	IEFSDPPT	Set Unique Key and KEY=0 Bits	IEFSD060	LPALIB	KEY=0

SB21-1550-0

IBM