ICL

SYSTEM TEN

DMF II

SYSTEM ANALYSIS GUIDE

THIS DOCUMENT IS CLASSIFIED

'COMPANY RESTRICTED'

Published by the Publication Centre, S&TS,
ICL, 85 Upper Richmond Road,
London SW15 2TE (LON 15).

Previously issued as RP4007

S&TS Issue 1/0

December 1977

CONTENTS

Title: ISSUE CONTROL SHEET Section: 0

Sec	Page Range	Issue Level of Page										Inserted Pages	
		0	1	2	3	4	5	6	7	8	9		
0	000-009	1/0					1/0						
1	100-109	1/0									1/0		
2	200-209	1/0									1/0		
	210-219	1/0			1/0								
3	300-309	1/0			1/0								
4	400-409	1/0					1/0						
5	500-509	1/0					1/0						
6	600-609	1/0									1/0		
	610-619	1/0				1/0							

LIST OF TABLES

TABLE PAGE

PREFACE

This guide will aid the programmer in solving a DMF II system problem because it contains descriptions of the internals of DMF II. The fields for label structures, locks, jobstream control blocks, common core constants and configuration tables must not be used by applications programmers because the definitions and locations of these fields may be changed in future releases of DMF II.

SECURITY

This document has "Company Restricted" status. Under no circumstances may this Guide be shown to ICL personnel without proper authorisation or issued to System Ten customers without proper authorisation.

Section 1
DMF II LOCKS

INTRODUCTION

A lock is defined as a field which restricts access to a specific resource. The partition which sets the lock can access the resource but no other partition can use the resource until the lock is set "off". The resource may be a disc drive, a data file record, a linked sequential pool free sector list, a volume label, etc. When the lock is set "on", the field always contains the number of the partition which set the lock. This partition is referred to the "locking partition".

Locks may be in Common Core or on disc.

COMMON-CORE RESIDENT LOCKS

The following paragraphs describe the use and setting of locks which reside in Common Core. When DMF II Standard Common is initialised (i.e., _LCOM > 0), Common Core resident locks may be used to control physical access to disc resources.

When DMF II Standard Common is not initialised (i.e., _LCOM = 0), the disc resources normally controlled by Common Core resident locks may be accessed.

SYSTEM LOCK , _SYSLK

Field: Location 358 in Common Core

Name: System Lock

Description: The System Lock is a field in DMF II Standard Common. This field is set before altering any unlocked pool or volume label and it is reset after the altered label has been rewritten. The System Lock should be set for short periods of time only. Overlays should not be loaded while this lock is set.

System Lock is set by using the following procedure:

1 Compare _SYSLK with _LPCD. If equal, the lock has already been set by the partition doing the check.

2 Compare _SYSLK against blank. If higher than blank, the lock has already been set by another partition. If the System Lock has been set by another partition, this compare operation must be repeated (i.e., the lock must be retryed). In order to retry the test, a counter must be incremented; if the limit has not been reached, the lock may be retried after a partition switch; if the limit has been reached, the lock is not retried

3 If the system lock is blank, it is set with _LPCD. This move must follow the unsuccessful branch in step 2.

There cannot be a partition switch beween steps 2 and 3; coding step 2 with the instructions:

```
C _SYSLK,=C' '
BNE
```

allows for a partition switch on equality, and is incorrect

System Lock may be set to blank at any time by the partition which set the lock. To allow contention, a partition switch must be allowed to occur before the System Lock is set again.

DISC DRIVE LOCK TABLE, _DLTAB

Field: Locations 510 through 525 of Common Core

Name: Drive Lock Table

Description: The Drive Lock Table contains a lock position for each disc drive (0-15). The first location (510) is for drive zero, the second is for drive one, etc. Whenever DMF II Standard Common is initialized, Drive Lock must be set for any alteration of either a linked sequential file directory or the contents of the configuration tables on a SYSRES volume. Drive Lock is also used to prevent access to the drive involved by other partitions during multisector operations (e.g., multisector records, overlay loading).

See _DLTAB in Table 5.1. The following steps outline the procedure for setting a drive lock:

1 Test the drive lock position for the host partition (this partition) code; if equal, the lock is already set

2 Test the appropriate Drive Lock Table position for blank. If it is not blank, a partition switch must be executed before the Drive Lock can be tested again

3 Set this partition's code in the position of the Drive Lock Table for the drive to be locked. The restriction on branches from step 2 to step 3 when setting _SYSLK applies to _DLTAB also. The lock position for a drive in _DLTAB may be reset to blank at any time by the partition which set the drive lock. When the lock position for a given drive is reset to blank, a partition switch must be executed before that drive can be locked again

Disc Drive Status Table (_DSTAB)

A disc Drive Status Table is stored in common core positions 530 through 545. The disc Drive Status Table contains a position for each possible disc drive (0-15). The Drive Status Table is initialized from location V_DSTB of the Volume Label of the primary resident system when DMF II Standard Common is initialized by MAINT.

A nonblank status table entry (normally an installation defined code) allows access to that drive only by partitions which have that drive as either their System Residence drive or which have been assigned that drive by the operator via CONI in response to a mount message on CONO. If its corresponding status position is blank, or if DMF II Standard Common is not initialized, the drive may be accessed by any partition (available).

The contents of the disc drive status table may be modified with the MAINT SET D() command.

RECORD LOCKS, _RPTAB ENTRIES

Field: Locations 410 through 509 of Common core; ten entries, each of ten digits.

Name: Record Protect Table

Description: The record protect feature protects a file which may be updated from more than one partition; it prevents more than one partition from attempting to update the same file at the same time.

Record locks are implemented by a Record Protect Table (_RPTAB) located in common core (starting at location 410). This table contains ten record lock entry fields, each entry field consists of the following three fields:

1 the address of the first sector in the block being locked

2 the identification code of the locking partition

3 a link to the next entry field in the table

A record lock may be set on (an entry is made into _RPTAB) by either a GETUP or READUP macro call; a lock is set off (_RPTAB entry is erased) by the next macro call for that file which had issued the locking macro call.

When a record is locked, the address of the block containing the record and the partition ID are placed into an available _RPTAB record lock entry field. The number of blocks which can be locked at any one time is 10, the size of the _RPTAB table. If the _RPTAB table is full when a GETUP or READUP call is issued, the partition issuing the call will switch until either a table entry becomes available, or until 1000 switches have been made (see the LOCKTEST system generation option).

Entries in the record protect table may also be set and cleared with the ENQUE and DEQUE macro calls.

LABEL RESIDENT LOCKS

The following paragraphs describe the locks which reside in system labels (i.e., Volume, Pool, File).

VOLUME LOCK, V_LPTN

Field: Location 17 in Volume Label

Name: Locking Partition Code

Description: This lock is set while performing volume initialization and reformatting the pool directory.

If the volume is unlocked, V_LPN is blank; if the volume is locked, V_LPN contains the partition code (i.e., it equals _LPCD of the _LCDEF macro which defines partition low core) of the locking partition. The partition codes are:

0 - 9 for partitions 00-09

P - Y for partitions 10-19

If DMF II Standard Common is initialized, system lock must be set before reading the volume label when intending to set volume lock. If volume lock is set to another partition, system lock must be cleared, and a partition switch must occur before retrying.

Before testing volume lock for blank, it must be tested against the host partition number. If equal, the volume lock is already set.

POOL LOCK, P_LOCK

Field: Location 78 in Pool Label

Name: Pool Lock Flag

Description: This flag is used for linked, relative, and scratch pools to indicate if the pool is locked and the nature of the lock. The following codes may be used for a pool lock flag:

1 L, this code specifies a long-term lock; it is set for work pool access, exclusive access to both direct and relative pools and the non-contention use of a linked sequential pool

2 S, this code specifies a short-term lock; it is set, for example, while allocating sectors to a linked file

3 Any other character may be used to signify an unlocked state

The character U is placed in this field when the pool is being unlocked, however, only the value S or L is used when testing for the pool lock status.

For relative pools only, a pool lock may be set by LIOCS (for exclusive use) as a substitute for a file lock. This type of operation is similar to the locking of relative pools at a pool level by the assembler when it is using a pool as work space.

If this field contains L or S, the P_LPTN must be set to the locking partition number. If this field contains other values, P_LPTN should be set to blank.

LOCKING PARTITION CODE, P_LPTN

Field: Location 79 in Pool Label

Name: Locking Parition Code

Description: This field is used only when the pool is locked (i.e., P_LOCK contains an L or S). It contains the partition code of the locking partition. When unlocking a pool, system programs set this field to blank but it should not be tested for a blank state except by such functions as MAINT UNLOCK. The partition codes are:

1 0-9 for partitions 00-09

or

2 P-Y for partitions 10-19

The contents of this field will be the same as the contents of field _LPCD in low partition core which is defined by the _LCDEF macro.

FILE LOCKING PARTITION CODE, F_LPTN

Field: Location 56 in File Label

Name: Locking Partition Code

Description: The file lock is set by LIOCS when the user program specifies exclusive use for a file (specified by the USE= parameter of the FCB macro). When the file is opened the file label is updated to indicate the locking partition. When the file is closed, the file is unlocked (i.e., the locking partition code is blanked out in the file label).

When a partition opens a file for exclusive use, no check is made to determine if the file is currently opened by other partitions for shared use. When a file is opened for shared use, no action is taken to indicate this condition anywhere in the system. A file, therefore, could be opened for shared use by several partitions at the same time and subsequently be opened by a partition for exclusive use. Once a file is opened for exclusive use, no new partitions will be able to open the file. This characteristic dictates the need for careful program scheduling to ensure that the file lock is effective.

The file lock is intended to restrict the use of a file to only one program at a time.

If a file is opened for shared use and another file opens it for exclusive use, the first partition will not be able to close the file until after the second partition has closed the file. LIOCS will give control to the error exit of the CLOSE macro call with an 'E' status code until the partition with exclusive use has closed the file.

When the F_LPTN field is set it will contain the code of the locking partition; the codes are:

0 - 9 for partitions 00 through 09,

P - Y for partitions 10 through 19.

When cleared, it contains blank.

Section 2
LABEL STRUCTURES

Table 2-1 Volume Label Format

Label	*Location & Length*	*Descriptor*	*Comments*
V_LAB	0(3)	Volume Label Indicator	This field always contains the literal "VOL". The presence of this value in disc sector 00001 (the disc sector is 60001 on Model 41B packs) indicates that volume has been initialised by DMF II. If not initialised, the volume cannot contain DMF II pools or files and is not protected from initialisation by DMF II utilities such as DMFSRV. If initialised, a volume cannot be reinitialised without operator intervention, and is subject to the protections offered by V_LPTN and V_PASS.
V_SYS	3(1)	Volume SYSRES Indicator	A value of "1" in this field indicates a SYSRES volume; a value of "2" indicates an Interchangeable volume.
V_SNAM	4(6)	System Name	This name identifies all volumes associated with the SYSRES. When used as a volume identifier (as in the source and target specific ations for COPY), the field contains the name of the SYSRES volume of that system. This field matches the P_SNAM field of any volume pointers which point to any pools or volume pointers located on this volume.
V_SEC	10(1)	Volume Security	This field is defined for SYSRES volumes and indicated whether or not the system is accessible for physical disc dumps. The field is also used to indicate whether or not execution of TESTER is permitted when this volume is the primary SYSRES. The values in this field are set by MAINT and may either be blank (access by TESTER permitted) or non-blank (TESTER access not permitted).
V_PTP	12(5)	Pool Tree Pointer	This field contains the address of the root of the pool directory for this volume. The five digits of this address are used with the physical drive number of the volume to begin search for pools resident on this volume. If this volume is a SYSRES, the resultant six-digit disc address is stored in _LPTP for all partitions using that SYSRES.
V_LPTN	17(1)	Locking Partition Code	This field is used to lock the entire volume during volume initialisation and during pool-directory reformatting operations. If the volume is unlocked, this field is blank. If the volume is locked, this field contains the partition code of the locking partition (same as _LPCD).
V_DMF	18(1)	DMF II Indicator	This field indicates if this label is a DMF II volume label. A double quotation mark (") indicates a DMF II volume label; any other value indicates DMF I.
V_LEXT	19(6)	Lower System Extent	This field specifies the low boundary and addressing mode of this volume. If the volume was recorded in a surface mode, this field will contain a disc address in the form '*dsssss*' where *d* is the logical drive number and *sssss* is a five-digit sector address. If the volume was recorded in a cylinder mode, this field specifies the low boundary in the form '*dccch*X'; *d* is the logical drive number, *ccch* is the lowest addressable track in the system and X is literally 'X' to signify the cylinder mode. This field is recorded when the volume is initialised by a MAINT INIT or COPY command or by the DMFSRV program.

Table 2-1 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
V_LDR	19(1)	Logical Drive Number	This field is the first position of V_LEXT and contains the logical drive number for this volume. Values range 0-?. This drive number is used in all volume pointers which reference this volume.
V_HEXT	25(6)	Upper System Extent	This field contains the upper system extent address for this volume, recorded in the same format as V_LEXT. The pool directory and all pools on the volume are recorded within the boundaries defined inclusively by V_LEXT and V_HEXT. The maximum size of the volume must never exceed the physical capability of a logical volume. This field does not contain a drive number.

V_TYPE	31(1)	Volume Type	This field contains a code which indicates the type of disc pack in use. The permitted codes are as follows:
			blank = Model 41
			L = Model 41A
			U = Model 41B

V_CMSZ	32(3)	Common Size	For a SYSRES volume, this field contains the size of Common Core in thousands of characters. This field is set by Volume Update and System Copy commands and by the MAINT and DMFSRV program. The MAINT program uses this field when processing the STARTUP command. The field is blank on Interchangeable Volumes.
V_SCNT	35(3)	System Volume ID Counter	For a SYSRES volume, this field contains the current value of a counter used by the MAINT INIT command to assign default V_ID fields in newly initialised volumes. This field is always numeric and is incremented only when used by INIT. On non-SYSRES volumes, this field is blank.
V_ID	38(3)	Volume ID	This field should contain a unique identification for a particular volume within a system. On a SYSRES volume, this field is always zero. On Interchangeable Volumes, this field is normally assigned a sequential number from V_SCNT, but the field can also contain a user-specified value. The contents are stored left-justified and are blank-filled. This field matches P_VLID of any volume pointers on a SYSRES volume which represent pools on this volume. The field may contain any System Ten character except '@'.
V_DESC	41(14)	Volume Address/ Description	The contents of this field are used with V_SNAM and V_ID to uniquely identify the volume. This field is likely to have the greatest mnemonic significance to the user. The contents are stored left-justified and are blank-filled. This field matches P_VLDS of any volume pointer on a SYSRES volume which represent pools on this volume.
V_PASS	55(6)	Volume Password	If the password is protected, this field is non-blank and must be specified by the user in order to reinitialise the volume. This field is never displayed when requesting an operator decision to overwrite the volume. The field is set at initialisation time.

Table 2-1 (concluded)

Label	*Location & Length*	*Descriptor*	*Comments*
V_DSTB	61(16)	Permanent Drive Status Table	This field is meaningful only when used on a SYSRES volume; it is used by MAINT to set the Drive Status Table (_DSTAB) in Standard Common. This field contains a single character entry for each possible physical drive; a blank entry indicates that the corresponding drive is available for use.
V_ORG	77(1)	Original System	This field contains "0" for a Model 20 processor, "1" for a Model 21 and "2" for a Model 22.
V_DDSP	78(1)	Displacement in Multi-Volume Drive	This field contains a single digit which indicates this volume's displacement from the lowest drive number which shares the same access aim mechanism. Only the numeric portion is used. This fiald may contain blank or 0 to 3.
V_DRIV	79(3)	Drive on Pack Model Code	This field contains a disc drive type and pack type identifier, where Drive Type — Code for type of disc pack 40 — 41 42 upper — 41A 42 lower — 41B 43 — 43 44 — 44 44-1 — 441 44-3 — 443
V_CYL	82(3)	Pack Size in Cylinders	This field contains the physical count of cylinders which can be accessed by the System Ten processor.
V_LTRK	85(2)	Lower Track Address(Head Member)	This field contains the number of the lowest head on the drive
V_UTRK	87(2)	Reserved Area	-
V_SMSZ	89(4)	Reserved Area	-
V_SMRK	93(3)	Reserved Area	-
V_HTSZ	96(2)	Reserved Area	-
V_TMTC	98(1)	Time out/MTIOC	This field contains two parts: the lower four bits contain the time out value for a Model 22 processor and the upper two bits are a reserved area.
	99(1)	Reserved Area	-

Table 2-2 Pool Label Format

Label	*Location & Length*	*Descriptor*	*Comments*
P_NAME	0(6)	Pool Name	The contents of this field are defined by the contents of the P_TYPE field. The P_NAME field may contain the label (address) of the root (entry point) of a binary tree structure which forms any one of several types of pools. Valid values are as follows: If PTYPE is: — P_NAME contains: blank — the user-assigned name of a linked pool @ — Normally N@@@@@ the label of a pool directory, however, if the volume containing the pool had originally been created under DMF I and was subsequently converted to run in a DMF II environment, this field would contain @@@@@@. R — the user-assigned name of a relative pool W — the user-assigned name of a work pool V — the user-assigned name of a volume pointer
P_LOAD	7(5)	Pointer to Less-Than Pool	This field contains the physical disc address (less drive number) of the next pool root label entered into the pool directory tree structure whose name is less than (alphabetically precedes) the name of the current label. If no less-than label exists, this field is set to '/////'.
P_HIAD	13(5)	Pointer to Greater-Than Pool	This field contains the physical disc address (less drive number) of the next pool root label entered into the pool directory tree structure whose name is greater than (alphabetically follows) the name of the current label. If no greater than label exists, this field is set to '/////'.
P_BKUP	19(5)	Pointer to Parent Pool	This field contains the five-digit address of the pool label which precedes the current label in the pool directory tree structure (refered to as the parent label). This field in the pool directory root label (entry point) contains the address of the Volume label (i.e., 00001).
P_FTP	25(5)	File Tree Pointer	The contents of this field depends on the type of pool in which it is located, <u>For a</u>: — <u>P_FTP contains</u>: Linked pool — the address of the file directory root label Relative pool — the address of the file label All other pool types and null pools — slashes (i.e. '/////')
P_PASS	30(6)	Pool Password	This field contains the password for linked, scratch and relative pools; it is blank for all other pool types or if the pool is not password protected. If this field is blank, any password given will be ignored.
P_FSP	37(5)	Free Sector Pointer	This field contains the address of the first free sector in a linked pool.
P_NAS	48(6)	Sector/Data Sector Allocated	This field contains the total number of sectors allocated to this pool.

Table 2-2 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
P_NFS	54(6)	Number of Free Sectors	For: — This field contains: Linked pools — the length of the free sector list (maintained but not used by system software for logical purposes) Pool root — the number of sectors available for creation of pool labels Relative pools — the number of unused records Work pools — the number of sectors
P_NDS	60(6)	Number of Discarded Sectors	This field is developed by CREATE for linked pools; it contains the number of sectors skipped (linked around) because of FLAG status. It serves the same purpose in the Pool Directory root label. This field in relative pools is divided into the following: Field — Label — Contents 60 & 61 — P_NBAD — Number of bad (flagged) sectors 62 & 63 — P_BLOT — Size of blot table 64 & 65 — P_BS — User speciifed bucket size
P_LDR	66(1)	Logical Drive Number	This field contains the logical drive number assigned the volume on which the pool is located.
P_LLIM	67(5)	Pool Lower Limit	This field indicates the low boundary and addressing mode of the pool; it applies to all pool types except Volume Pointers. If the pool is recorded in surface mode, this field contains the five-digit address of the first sector in the pool and is always a positive number. If the pool is recorded in cylinder mode, its low limit is recorded in the format *ccch*X, where *ccc* is the cylinder number (000-199), *h* is the head number (0-9), and X is literally 'X', causing the field to have a negative value when tested with the form numeric instruction. The first sector allocated to a cylinderised pool is the first sector of the track specified by *ccch*, (i.e., sector 00 if *h* is even, sector 50 if *h* is odd).
P_HLIM	73(5)	Pool Upper Limit	This field contains the pool's upper boundary and is recorded in exactly the same format as P_LLIM. The last sector allocated to a cylinderised pool is the last sector of the track specified by *ccch*, i.e., sector 49 if *h* is odd; 99 if *h* is even.
P_LOCK	78(1)	Pool Lock Flag	For linked and scratch pools. The contents of this field indicates whether or not the pool is locked, and the nature of the lock. The codes which may be used are: L for long-term lock S for short-term lock (valid for linked pools only) Any other character indicates an unlocked pool The system standard value placed in this field when unlocking a pool is "U", but this value is not used when testing for an unlocked pool. Both "S" and "L" must be tested explicitly.

Table 2-2 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
P_LPTN	79(1)	Locking Partition Code	This field is meaningful only when P_LOCK contains either "S" or "L". The contents of this field is the same as the partition code(_LPCD) of the locking partition. When unlocking a pool, system programs set this field to blank, but it is <u>not</u> tested for blank except by such functions as MAINT UNLOCK. The codes which may appear in P_LPTN are: 0-9 for partitions 00 to 09 P-Y for partitions 10 to 19 blank specifies that no lock is present
P_TYPE	80(1)	Pool Type	The contents of this field indicates the type of the pool label. The codes which may be used are: blank for linked pools @ for pool directory root label W for work or scratch pool R for relative, direct or variable pools V for volume pointer M for work pool Note: In automatic volume search mode, this field is tested to determine whether or not a given label is a volume pointer. It should <u>not</u> normally be tested for "@" by traverse algorithms (see _LPTP).
P_FDEF	81(1)	Default File Type/Attribute	This field is used for linked pools only; it may be speciifed at CREATE time as the FTYPE parameter; it may also be modified by RENAME. The value of P_FDEF is used by FILE when cataloging a new linked file for which no TYPE parameter is specified.
P_PROT	83(1)	Pool Protection Code	For linked pools, the contents of the field indicates whether the files in this pool are Read-and-Write or Write-protected by the pool password. If the pool is password-protected in READ mode, the password must be specified whenever the pool's files <u>are accessed as a group</u>, including read-only access as in the source pool for COPY=MERGE. When the pool protection mode is WRITE, the password (P_PASS) need be specified only when altering or deleting files in the pool. In either case, the pool password overrides password protection on individual files whenever MAINT commands access all of the pool's files in a group. (Note: This does not include generic operations, which use file passwords). For Relative pools, P_PROT is always equal to F_PROT; either may be used to determine the mode of protection provided by P_PASS and F_PASS. For Scratch pools, P_PROT indicates the mode of protection offered by P_PASS over the entire pool area. The codes which may be used are: "R" Read and Write protection "W" Write protection The correct test is always for equality or non-equality to "W".

Table 2-2 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
P_TEXT	84(16)	Pool Textual Description	This field contains descriptive information provided by the user through a CREATE or RENAME command. It is displayed but not used by the system.

NOTE: THE FOLLOWING FIELDS ARE FOR RELATIVE AND DIRECT ACCESS POOLS

Label	*Location & Length*	*Descriptor*	*Comments*
P_NBAD	60(2)	Number of Bad Sectors	This field contains the number of bad (flagged) sectors encountered while formatting a type "R" pool.
P_BLOT	62(2)	Maximum Number of Alternative Sectors	This field contains the number of alternative sectors allocated for blot recovery. This value is determiend by the user and is specified in the NBLOTS Parameter in the CREATE command. The permitted range of values is 00-20. When NBLOTS=0 a single sector is allocated for a blot table and no alternate sectors are provided. When NBLOTS Is greater than zero, a blot table, an alternate blot table and NBLOTS alternate sectors are allocated. NBLOTS default to a value of 5 in the MAINT CREATE command.
P_BS	64(2)	Bucket Size	This field is set by CREATE for direct access files only; it contains the user-specified bucket size when a nonstandard, bucket-oriented direct access scheme is in use. When the user does not specify bucket size, this field is set to blanks.

NOTE: THE FOLLOWING FIELDS ARE FOR VOLUME POINTERS ONLY

Label	*Location & Length*	*Descriptor*	*Comments*
P_VLSN	48(6)	System Name of Linked Volume	The P_VLSN field contains the name of the Interchangeable volume on which the pool (that this label identifies) is located. The field is one of three fields which form a volume pointer. The volume pointer is used during automatic volume search procedures. The volume pointer identifies and locates the pool for which this label was constructed. In order for a volume search to be successful, all of the following must match: *Volume Pointer Fields* — *Linked Volume Label Fields* P_VLSN — V_SNAM P_VLID (see field 81 below) — V_ID P_VLDS (see field 84 below) — V_DESC
P_SYN	54(6)	Pool Name on Linked Volume	This field contains the actual pool name (P_NAME) of the link to another pool label volume pointer in SYSRES which is referenced by this volume pointer. This name is used to lock up the pool label after the interchangeable volume (which is referenced by P_VLSN, P_VLID and P_VLDS) has been found. The P_NAME and P_SYN fields of a volume pointer are identicial unless a synonym was specified in the LINK command which created the volume pointer.

Table 2-2 (concluded)

Label	*Location & Length*	*Description*	*Comments*
P_VLOC	60(6)	Volume Label Disc Address	This field contains the *logical* disc address of the volume label referenced by this volume. The first position is the logical drive number and provides automatic volume search with the number of the first drive to be accessed in looking for the desired interchangeable volume. The second position is a zero, if the linked volume is a Model 41 or a 41A pack, or a 6 if the linked volume in a Model 41B pack. The remaining positions contain "0001".
P_VLID	81(3)	Volume ID on Linked Volume	This field contains the value of the Volume ID field (V_ID) for the interchangeable volume referenced by this volume pointer. This field is intentionally contiguous with P_VLDS so that the two fields may be used as a single 17-character field for moves, compares, etc. The P_VLID field is one of three fields which form a volume pointer that is used during automatic volume search procedures. This volume pointer identifies and locates the pool for which this label was constructed. In order for a volume search to be successful, all of the fields must match as described for P_VLSN (field 48 above).
P_VLDS	84(14)	Volume Address/ Description	This field is one of three fields which form a volume pointer that is used during automatic search procedures. This volume pointer identifies and locates the pool for which this label was constructed. The P_VLDS field provides the V_DESC value for the interchangeable volume referenced by this volume pointer. In order for volume search to be successful, *all* of the fields must match as described for P_VLSN (field 48 above).

Table 2-3 File Label Format

Label	*Location & Length*	*Descriptor*	*Comments*
F_NAME	0(6)	File Name	The field contains the user-assigned file name. The file name may consist of from one to six alphabetic or numeric characters, however, the first character in a file must be an alphabetic character. The name of a file must be unique within the pool in which it is stored.
F_LOAD	7(5)	Pointer to Less-Than File	This field contains the five-digit disc address of the next entered file label which has a name that is less than (alphabetically precedes) the name of this (current) label. If no less-than label exists in the file label tree structure, this field will contain slashes.
F_HIAD	13(5)	Pointer to Greater-Than File	This field contains the five-digit disc address of the next entered file label which has a name that is greater than (alphabetically follows) the name of this (current) label. If no greater-than label exists in the file label tree structure, this field will contain slashes.
F_LDR	18(1)	Logical Drive Number	This field is no longer supported.
F_BKUP	19(5)	Pointer to Parent File	This field contains the address of the file label which precedes this (current) label in the file label tree structure. If the current label is either the root (entry point) of its pool's directory or if the pool label pool type field (P_TYPE) is 'R', this field contains the address of the pool label.
F_BEG	35(5)	File Begin Address/Index Root	For linked files, this field contains the address of the first sector in the file; it contains "/////" if the file is null. For index (F_TYPE=I) files, this field contains the address of the root of the index tree; it contains "/////" if the index contains no entries. For Relative, Direct and Variable files, this is the file base address (i.e., the disc address of the first record in the file).
F_PASS	30(6)	File Password	This field contains the password for all file types; it is blank if no password is used. For Relative, Direct and Variable files the contents of this field always matches the contents of P_PASS. The contents of F_PROT determines how the password is used.
F_TYPE	36(1)	File Type	This field contains a code which identifies the file type; the codes are: "O" for linked Sequential File "S" for Doubly Linked Sequential File "I" for Indexed-Linked Sequential File "R" for Relative File "D" for Direct File "V" for Variable Sequential File

Table 2-3 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
F_LPTN	56(1)	Locking Partition Code	If non-blank, this field indicates that the file has been locked for Exclusive Use. When set, this field contains the partition code (from _LPCD) of the locking partition; valid codes are as follows: 0-9 for partitions 00-09 P-Y for partitions 10-19 blank not locked (Note: This field has an alternate use for index files; see F_DPN for Index Files.)
F_ATTR	57(1)	File Attribute	Used for linked sequential files only. Codes are: blank Data File T Absolute program text R Relocatable program text A Assembler source (including macros) C LIOCS control file G RPG source file U MAINT USE file J Job stream file
F_PROT	58(1)	Password Protection Indicator	If F_PASS is non-blank, this field indicates whether the password protects the file from all access (Read protection) or from modification only (Write protection). R = Read and write protection W = Write protection only This field is *always* tested for W (i.e., all other values are interpreted as Read protection).
F_STAT	59(1)	Record Status Indicator	If the STATS= parameter is not given in the FCB, the F_STAT Indicator specifies the defaults: 1 for STATS=NO (no record status characters) 2 for STATS=YES (record status characters present) 3 for STATS=USER (record status characters present)
F_UPD	62(6)	Date of Last Update	Date file was last updated, in format *yymmdd*. Set by MAINT FILE command (when replacing an existing file), by Text Editor, and by LIOCS (when access is not Read-Only). Copies are made without change from the source file during COPY, MOVE, BACKUP and RESTORE operations.
F_CREA	68(6)	File Creation Date	Date file was created, in format *yymmdd*. This field is blank if no creation date was available.

Table 2-3 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
F_FORM	37(1)	Record Format	This field contains a code which specifies "F" (Fixed) for Linked Sequential and 'R' and 'D' type files. "V" (Variable) for 'V' type files. For index files, this field is used for F_IATR.
F_BL	38(4)	Block Length	Number of characters per block in this file. Not used for Indexed Linked Sequential. Set by the MAINT CPEATE and FILE command and by LIOCS OPEN. For other than variable sequential, must be an even multiple of record length. For variable sequential, contains the maximum block length. *For the file type:* — *This field specifies:* Linked sequential — Block length (9306) Indexed — Not used Relative — Block length (9899) Direct — Block length (9899) Variable Sequential — Maximum Block length (9899)
F_RL	42(4)	Record Length	This field specifies either the actual or maximum number of characters per record in this file - permitted range is from 10 to 9899. *For the file type:* — *This field specifies:* Linked Sequential — Record length (9306) Indexed — Not used Relative — Record length (9899) Direct — Record length (9899) Variable Sequential — Maximum record length (9895)
F_KL	47(2)	Key Length	When used for indexed-linked sequential files and for direct access files. For direct files, the numeric range of this field is 01-99 and the alpha-numeric range is 01-49. For indexed-linked sequential files, the range is 01-66.
F_KD	51(2)	Key Displacement	Displacement of an indexed or a direct-access key within the record. Displacement of 00 means that the key begins in the first position of the record. For indexed linked sequential files, the acceptable range is 00-99. For direct-access files, the range is 01-99. The key must be contained within the record and within the first sector of a multi-sector record.
F_REV	53(3)	Revision Count	This field contains a value for the number of times the file has been replaced by the MAINT FILE command or by the Text Editor. The low-order two digits are recorded by the EDIT Program in the session number field of edited sectors. This field is *not* used by LIOCS and is always copied without change from the source file during COPY, MOVE, BACKUP and RESTORE operations.

blank

Table 2-3 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
F_SCNT	74(6)	Sector Count	For linked files, including index-linked sequential files, this field contains a value for the total number of *sectors* allocated to the file; this value does not include the file label itself. For relative files, this field contains a value for the *number of records* in the file, and thus defines the end-of-file. All records with relative record numbers equal to or greater than F_SCNT are considered to be "available". For variable files, this field contains a value for the *number of sectors* of data in the file. When F_SCNT is zero, the file is empty.
F_EOFA	81(5)	End of File Address	For linked files, this field contains the sector address of the last sector in the file (i.e., the one containing slashes in its linkage). This field is not used for index, relative or variable files. If the file is null, this field contains slashes.
F_TEXT	86(14)	Textural Description	Text and description of file or a user-comment area. Used only by linked-sequential and indexed-linked sequential files.

NOTE: THE FOLLOWING FIELDS ARE USED FOR INDEX FILES (F_TYPE=I)

Label	*Location & Length*	*Descriptor*	*Comments*
F_IATR	37(1)	Autoscan Indicator	This field contains a "1" if Autoscan feature can be used
F_DPN	56(6)	Pool Name of Data File	This field contains the name of the pool in which the data file resides.
F_DFN	80(6)	File Name of Data File	Used only by indexed-linked sequential files, and is the file name of the subject data file.

NOTE: THE FOLLOWING FIELDS ARE USED FOR DIRECT FILES(F_TYPE=D)

Label	*Location & Length*	*Descriptor*	*Comments*
F_DVRL	86(1)	Prime Divisor Length	Only used for direct-access files. Indicates number of significant digits in the prime divisor field. Contains '@' if not used.
F_DVR	87(6)	Prime Divisor	Used only for direct-access files. The prime divisor is left-justiifed with trailing blanks whose length of significant digits is defined F_DVLR. This field is used by the LIOCS direct-access algorithm.
F_ANAM	93	Direct Algorithm Name	This field is defined as the direct access algorithm name. The default is the standard DMF II algorithm, _PRIME.

Table 2-3 (concluded)

Label	*Location & Length*	*Descriptor*	*Comments*
F_NKEY	99(1)	Non-Numeric Key Indicator	Applies to Direct files only "N" or "." (period) = numeric "A" or "!" = alpha-numeric (The characters "." and "!" denote that the block is optimised.)

Section 3
JOB STREAM CONTROL BLOCK

INTRODUCTION

The JSCB provides a medium of communication between programs and the Conversational Loader at both load and exit times. When a program receives control, the JSCB provides information such as the pool and file name under which it was stored. If the program was loaded from a job stream, the JSCB includes the name of the JSF and the disc address of the next sector following the call to this program.

Several macros are provided to enable users to access parameters in a JSF; these assume the presence of the JSCB. User should ensure that their programs ORG at least as high as _JTOP; they should also be aware that LIOCS OPEN and CLOSE operations destroy the JSCB.

Some JSCB fields may be used at program exit to pass information to C_LOAD. C_LOAD does not expect information to be left in the JSCB from the previous load. The EXIT macro enables the user to set values in the JSCB indicating one of several C_LOAD actions, such as cancelling the current job stream or fetching a successor program. These options are described under _JOPC, below.

The _JSCB macro, which is accessible to both user and system programs, defines the standard labels for the fields of the JSCB. The fields contained by the JSCB are described in Table 3.1

Table 3-1 Job Stream Control Block

Label	*Location & Length*	*Descriptor*	*Comments*
_JOPC	300(2)	Operation Code	This field is used to pass information to the Conversational Loader at program exit time; it is set by C_LOAD at program load time. At program termination, an equal sign in the first position indicates to C_LOAD that a Control Block exists, otherwise remaining fields are ignored. The codes set by the EXIT macro have the following meanings:
		"=="	Initiate a new job stream whose name is in _JSFPN and _JSFFN.
		"= C"	Cancel the job stream
		"= F"	Fetch the program inidcated by _JPPN and _JPFN. Provide a simple passage of control from one program called via the job stream, the stream will be suspended until the fetched program exits
		"= b̸"	Pass on error communication code _JCODE to the next program in the job stream. Ignored if no job stream
		"= W"	Wait for a program to be called from the console. When the program thus called exits, control returns to the Job Stream File. This code is meaningful only when in a Job Stream File.
		"= *n*"	Give control to the console. *n* is a single digit in the range 0-9 and is the device number of the console device which is to receive control. If a job stream file is active, this code interrupts it. *Temporary* assignments to device *n* are made for the LOAD, COMMAND, CONI and CONO logical units. If the user wishes to establish a new logical unit configuration and/or a new JSF, he should call MAINT immediately from the console and enter the appropriate ASSIGN command. Note: Program is loaded from SYSRES drive only. If a program is loaded as a result of "= F" or "= W", C_LOAD will set fields _JSFPN, _JSFFN and _JSFDA and _LJSF to blank.
_JPPN	302(6)	Program Pool Name	Contains the Pool Name of the program just loaded. This field is also used when a program exits to fetch another program.
_JPFN	308(6)	Program File Name	File name of the program just loaded, or of the program to be fetched when _JOPC contains "= F"
_JCODE	314(1)	Error or Communications Code	The value of this field will be used by C_LOAD to update the System Configuration Table, except if _JOPC is "= C". This field is set by C_LOAD Prior to loading a program. If no job stream is active, or if no code has been posted, this position is blank. System programs will post an error severity code, if an error was detected within a system program or another user program. The severity code is reset to blank each time a JOb Stream File is called from the console, is initiated via MAINT ASSIGN, or calls itself. The code will be updated, if the code currently in the System Configuration Table is less severe than the current error encountered. Higher lettered codes are more severe; they are: "E" Error; the program has taken corrective action "S" Severe Error; a program execution step was aborted "T" Terminal Error; execution of the program was aborted User programs may pass error codes by using the EXIT macro. The severity of the detected error is based on the code's position within the System Ten character set, so that users can define codes above, below or within the range of values defined by the system. Note that "W" (warning) is *not* posted by system programs and is not part of the severity scheme

Table 3-1 (concluded)

Label	*Location & Length*	*Descriptor*	*Comments*
_JSFPN	315(6)	Job Stream File Pool Name	At program load time this field contains the pool name of the current Job Stream File. The existence of a JSF is indicated by a "J" in field _LJSF of low core; this field is blank if no job stream file is active. At program termination, operation code == in _JOPC indicates the job stream file identified by the contents of this field and field _JSFEN.
_JSFFN	321(6)	Job Stream File File Name	File name of the JSF from which the current program was called,or which is to be initiated by C_LOAD is contained by this field. When a Job Stream File is initiated, C_JSPN and C_JSEN will be set by C_LOAD to contain the qualified name of the JSF. The fields C_JCOD through C_JSDA will be moved into the Job Stream Control Block (JSCB) prior to loading a program.
_JSFDA	327(6)	Job Stream Current Disc Address	When a job stream is active, this field is set to the physical disc address of the sector which follows the sector in the Job Stream File that contains the program name, otherwise it is set to '/////'.
_JLOG	333(1)	Job Stream Log Flag	This flag indicates whether programs called are to log their commands on CONO. A "1" indicates commands are to be logged; a "0" indicates that commands are not logged.
_JTOP	334(1)	First Available Unused Position	This field contains the address of the next available store location. It provides a reference to the unused portion of store which may be used (as a label) in ORG-type statements.
_JORG	335(1)	Next Available Instruction Boundary	This field contains the address of the next available instruction boundary. It acts as a label which provides reference to the first available instruction boundary which follows the JSCB. The user may ORG his program to this label.

Section 4
CONFIGURATION TABLE

INTRODUCTION

A configuration table is provided in each System Resident Volume for each of the twenty partitions. Each partition configuration table specifies:

1 The attribute of each input device

2 logical unit assignment and operation characteristics

3 identification of any Job Stream File assignment made to the partition

Configuration tables are stored in volume sectors 00015 through 00024 (60015-60024 of a Model 41B disc pack); each table occupies half of a sector. Partitions 0 through 9 have their tables in the first 50 characters of the sector; partitions 10 through 19 in the last 50 characters.

To illustrate:

SECTOR	SECTOR LOCATIONS 0 - 49 CONTAINS TABLE FOR PARTITION	SECTOR LOCATIONS 50 - 99 CONTAINS TABLE FOR PARTITION
00015	0	10
00016	1	11
.	.	.
.	.	.
.	.	.
0023	8	18
0024	9	19

The field contained by each partition table are listed and their contents described in TABLE 4.1

Table 4-1 Configuration Table

Label	*Location & Length*	*Descriptor*	*Comments*
C_ATTR	0(10) and 50(10)	Attributes for Device 0-9	This field contains ten, single position sub-fields. The relative position of each sub-field corresponds to a device assignemnt number. Device assignment numbers are assigned for each hardware device attached to the partition. Since Fast Access Channel (FAC) devices are not partition exclusive devices, magnetic tape and disc devices are not included. Each sub-field contains a device attribute code for its associate device. These attribute codes are set by the MAINT ASSIGN command. See table 4-2 for definition of the device attribute codes. The character codes for the different catagories of hardware are as follows: *Type of device* — *Solid character codes* Input-only device — blank, !, ", £, $, %, &, ', (,), *, +, comma, -, period and / Devices which are capable of both input and output — 0,1,2,3,4,5,6,7,8,9, colon, semicolon, <, =, > and ? Output devices only — @, A, B, C, D, E, F, G, H, I, J, K, L, M, N and O Special purpose device and terminals — P, Q, R, S, T, U, V, W, X, Y, Z, [, \,], ^ and _
C_RDR	10(1) and 60(1)	Command Assignment	
C_IPT	11(1) and 61(1)	Read Assignment	
C_LST	12(1) and 62(1)	Print Assignment	
C_PCH	13(1) and 63(1)	Punch Assignment	
C_CNI	14(1) and 64(1)	CONI Assignment	
C_CNO	15(1) and 65(1)	CONO Assignment	This field contains a one-character partition assignemnt for one DMF II logical unit. These assignments may be set individually by the MAINT ASSIGN MTIOC command (in the PERM mode). The C_LOAD Program moves these values from the configuration table of the partition's SYSRES into the corresponding positions of partition low core. (A move character instruction is used, because the bit codings for _LRDR, etc., are the same as here.) When a logical unit has been set to IGNORE or does not exist, this field is encoded as a slash ("/"). Otherwise the logical unit fields are encoded as follows: Bit 1-4 — Numeric portion of device number, ranging 0-9. Using this number as an index into C_ATTR gives the actual device attribute for this logical unit Bit 5 — Always *on* (i.e., always set = 1) Bit 7 — Conversation device

Table 4-1 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
C_LOD	16(1) and 66(1)	Load Assignment	If the character in this position is greater than "/" (ASCII code 0101111), the character indicates the assignment of the MTIOC LOAD Logical unit for this partition; this character is encoded in the same manner as C_RDR, etc. (The LOAD logical unit is the device from which program or JSF names are accepted when no Job Stream File is active. If the device is conversational, the message "A)ENTER PROGRAM NAME." is issued from this device. This field is set only by the MAINT ASSIGN command. If the character in this position is less than or equal to "/", the character indicates a partition type code for non-MTIOC (or "blind") partitions. Values in this range indicate that no DMF II logical units are assigned on this partition. (Logical unit assignments are all set to "/"; program names may only be entered by using a job stream. Partition type codes are: *C_LOD Code* — *Partition Type* blank — (No partition exists) ! — MDS (MDIOC) " — SCA # — ATA $ — ACA % — ADCA & — ACA-ICF ' — Digital Clock / — NODEV (any value greater-than a slash) — MTIOC
C_PLFN	17(4) and 67(4)	Print Logical Unit-Length of Form	This field is normally set by the PAGE = parameter of the MAINT ASSIGN command. This field contains a four-digit number which specifies the size of printer forms in the number of lines per inch per single form. For example at six lines per inch on eleven-inch forms would require C_PLEN to have a value of 0066. Note: The value 0066 is the default when size is not specified. C_PLEN is useful when simulating form feeds on a device which does not support "skip-to-top-of-form".
C_LPP	21(4) and 71(4)	Print Logical Unit Lines per Page	This field contains the maximum number of lines to be printed on each page of printed output. A program using this field (such as Assembler II) will print no more than the specified number of lines before skipping to a new page. This field is set by the MAINT ASSIGN command. When C_PLEN is set but C_LPP is not specified, default is 6 less than the value for Length of Form. The initial value set by volume update is 0060.

Table 4-1 (concluded)

Label	*Location & Length*	*Descriptor*	*Comments*
C_SRD	25(2) and 75(2)	System Residence Drive/Partition Protection	This field defines the current SYSRES Volume for this partition. If the partition is assigned to the *primary* SYSRES volume, the numeric portion of the first character position is not used and the second character position contains an asterisk. If common store is initialised, the primary SYSRES volume is stored in _IPLDR: otherwise the volume is 00. If the partition is assigned to a *secondary* SYSRES volume, the numeric porition of the first position of this field contains the drive number of the secondary SYSRES volume. The drive number is coded in binary with a range of 0-15. The second position of the field is the second digit of the disc address used to access the SYSRES volume label, bootstraps, etc. This position is 0 (zero) if the SYSRES volume is a Model 41 or 41A pack, 6 if the SYSRES is a Model 41B pack. This 2-character field is used by C_LOAD to determine whether or not the SYSRES was loaded from the proper drive and, if not, to bootstrap the appropriate volume. This field is set by the SYSRES=parameter of the MAINT ASSIGN command. The five-bit in the first character position is always *on*. The seven-bit in the first character position of this field is the Partition Protection Flag. If this flag is on, an operator override is required to change assignments for this partition (including SYSRES and Partition Protection). If the flag is off, assignments may be made for this partition without confirmation from the operator. This flag is set by the PROTECT=parameter of the MAINT ASSIGN command. The values in this field are meaningful only in the configuration table for the *primary* SYSRES volume. The primary SYSRES volume is defined by _IPLDR if common store is initialised; otherwise a Model 41 or 41A pack is assumed to be on drive zero.
C_JCOD	27(1) and 77(1)	Job Stream Error Code	Termination codes are set by user programs upon EXIT. The highest code set during the execution of a jobstream is stored in this field.
C_JSPN	28(6) and 78(6)	JSF Pool Name	This field indicates whether or not a Job Stream is currently assigned for this partition. If not, C_JSPN is blank. If the partition has a JSF assigned, this field contains the pool name of the JSF. This field is set by the MAINT ASSIGN command or by the C_LOAD Program when a JSF is initiated or cancelled.
C_JSFN	34(6) and 84(6)	JSF File Name	If C_JSPN is non-blank, this field contains the file name of the partition's current JSF. Otherwise, C_JSFN is blank.
C_JSDA	40(6) and 90(6)	JSF Current Disc Address	This field is only meaningful when C_JSPN is non-blank (otherwise, C_JSDA is also blank). When a JSF is initiated by MAINT ASSIGN, the pool and file names are set in C_JSPN and C_JSFN, however, C_JSDA is left blank. This allows C_LOAD to detect that the JSF must be located and its first sector accessed for a program name. Thereafter, C_LOAD uses C_JSDA to maintain the physical address of the sector following the sector containing the current program name.
C_JLOG	46(1) and 96(1)	Job Stream Log Flag	This field contains a single digit to indicate whether job stream commands are to be logged on the CONO device.
None	47(2) and 97(2)	Reserved Fields	This field is used for communications LIOCS and LTERM program.

Table 1-2 Device Attribute Codes

Code	*Attribute(s)*	*Device*
b̸	NONE	No device assigned
!	CR,M30	Model 30 Card Reader
"	M31	Model 31 Card Reader
+	PTR,M60	Model 60 Paper Tape Reader
/	TPTR	Model 7102 Terminal Paper Tape Reader
0	WS,M70	Model 70 Workstation
8	VDU,M80,CRT	Model 80 CRT
9	M81	Model 81 CRT
?	7102	Model 7102 Terminal Console
@	CP,M35	Model 35 Card Punch
A	M36	Model 36 Card Punch
F	PTP,M65	Model 65 Paper Tape Punch
I	TPTP	Model 7102 Terminal Paper Tape Punch
J	HSP,M50	Model 50 Line Printer
K	LP,M52	Model 52 Line Printer
L	M54	Model 54 Line Printer
P	M900*	Model 900 MDTS Terminal
Q	M902*	Model 902 MDTS Terminal
R	M925*	Model 925 MDTS Terminal
S	M950*	Model 950 MDTS Terminal
V	RTS, M120*	Model 120 Remote Terminal Scanner
W	JIS, M100*	Model 100 Job Information Station
X	TAS, M105*	Model 105 Attendance Station
Y	LSU,M191*	Model 191 Line Switching Unit

* Terminal attributes are not supported by ASSIGN MTIOC and are not permitted with ASSIGN MDS.

Section 5

STANDARD COMMON

Table 5-1 Definitions for the system constants area of common core (Standard Common)

Label	*Location & Length*	*Descriptor*	*Comments*
_NDATE	300C(6)	Numeric Date	This field contains the current date in the format: *yymmdd*. (For example: 740704 for 4 July 1974).
_UDATE	306C(8)	User Date	This field contains the current date in the format: *mm/dd/yy*. (For example: 07/04/74 for July 4 1974).
_SCTIM	314C(8)	Software Date Time Block	This field contains three subfields which store the Julian date (_JDATE) day of week (_DAYW) and software clock (_SCHTH). The _SCTIM field is divided as follows:
_JDATE	314C(3)	Julian Date	This field contains the day of the year in the format: DDD (for example, 285 for this 285th day in the calendar year. (Note that this field is preceded by the year in the _UDATE field.)
_DAYW	317C(1)	Day of Week	This field contains a character within the range of 1 to 7 which represents the day of the week.
_SCHTH	318C(4)	Software Clock	This field contains two subfields which define time in hours and in tenths and hundreds of an hour. When_SCHTH is referenced, the output is in the format: *hhth*. (For example: 1335 is for 35/100 of an hour past 1pm.)
_SCHRS	318C(2)	Software Hours (24 Hr)	This field contains two digits for time in hours where 00 is midnight and 23 is 11pm.
_SCTH	320C(2)	Software Clock Tenths and Hundreths	This field contains two digits for tenths and hundreths of an hour.
(None)	322C(2)	(None)	Reserved locations
_WCTIM	324C(6)	Wall Clock (12 Hr)	This field contains three 2-digit subfields which contain hours, minutes and seconds in the format: *hhmmss*. These subfields are maintained by the DMF II Clock Partition software. The _WCTIM field is subdivided as follows:
_WCHRS	324C(2)	Wall Clock: Hours	This field contains two digits for time in hours in the format *hh*. (For example: 01 is 1am or 1pm and 12 is 12noon or 12 midnight)
_WCMIN	326C(2)	Wall Clock: Minutes	This field contains two digits for the minutes count in the format *mm*. (For example: 00 is zero minutes past the hour and 59 is 59 minutes past the hour.)
_WCSEC	328C(2)	Wall Clock: Seconds	This field contains two digits for the seconds count in the format *ss*. (For example: 00 is zero seconds past the minute and 59 is 59 seconds past the minute.)
_WCAP	330C(1)	AM/PM Indicator	This field contains the character A is the wall clock time is AM or the character P is the wall clock time is PM. This field is maintained by the standard DMF II Clock Partition software.

Table 5-1 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
_COMV1	331C(1)	First Validation Flag	This field is a validation flag no.1. When Standard Common is loaded by the MAINT program, this field is set to a value of < (less than sign). If this field or the other two validation flags are found to be incorrect, the contents of Standard Common have been overwritten and should be initialised by the MAINT STARTUP command.
_CALOC	332C(4)	Address of first User Allocation	This field contains a 4-digit pointer to the last allocated Common Allocation Block (CAB) of common store which was reserved for user allocation. The _CALOC field must contain an address within the range 1000 to 79990, if there is a current allocation, or the field must contain ////, if there is no current allocation. Each CAB is the last 10-character portion of each allocation. The CAB is established and linked by the CALOC and CFREE Macros. (These macros enable the user to establish allocations for program use and to release these allocations at the end of the job.) (See also _CFREE)
_CFREE	336C(4)	Next Available Common Address	This field contains a pointer to the CAB for the next available block of free reserved Common Core. If Common Core has been reserved for user allocation, _CFREE contains a value in the range 1000 to 79990. If all available Common is currently allocated to user programs or if no user allocation space has been reserved, _CFREE contains ////. (The free reserved Common Core list is established by the MAINT program, where areas to be reserved are specified in the syntax of the STARTUP command.) (See also _CALOC)
_UPS	340C(10)	User Program Switches	This field consists of ten one-character positions. The label _UPS has a defined length of one character. The user program switches are referenced individually using the form: _UPS+*n*, where *n* is the switch number. For example, switch 6 is referenced as _UPS+6. DMF II software does not test or modify the user program switches; it only provides, via MAINT, the ability to set or to clear these switches.
(None)	350C(5)	(None)	Reserved locations.
_CPROC	355C(1)	Current Processor	This field identifies the model number of processor on which DMF II is currently executing; where 0 (zero) indicates a Model 20 processor with am MDC I type disc controller; 1 indicates a model 21 processor with an MDF I type disc controller and A indicates a Model 21 processor with an MDF II type disc controller; and B indicates a Model 22 processor.
_COMSZ	356C(2)	Common Size in thousands	This field contains the user-specified size of common (specified in 1K units). Typically, this value is specified using the MAINT SET COM=command.
_SYSLK	358C(1)	System/Pool Directory Lock	This field contains the partition code of the partition that is performing an operation which requires a lock. This field contains a blank when an unlocked state exists.
_COMV2	359C(1)	Second Validation Flag	This field is validation flag no.2. When Standard Common is loaded by the MAINT program, this field is set to a value of ; (semicolon). (See _COMV1 and COMV3)

Table 5-1 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
_IOCS	360C(10)	LIOCS Central Entry Point	This field contains a branch instruction. The first five locations contain an unconditional branch that points to the LIOCS entry point, or to the NO LIOCS HERE error routine (see _CLIAR). The last four locations of the _IOCS contain an address that points to a LIOCS control block or, if there is no LIOCS, to 0000. (The LIOCS control block is labelled _LIOCB in the LIOCS assemblies.)

_IDLE	370C(20)	System Idle Logs	This 20-character field contains the following instructions:
			`_IDLE   C    _IDLE(1),OC`
			`        BESW  lOP ,_IDLE`
			which cause the partition to branch and switch. Partitions enter this loop when the MAINT STARTUP, CLEAR=SYSTEM command is executed, and also when a partition has no job stream file and the LOAD logical unit is unassigned. MAINT ASSIGN changes the instructions to start partitions.

_SYS	390C(20)	Drive-Independent EXIT/IPL	This 20-character field contains the instructions that will cause a disc IPL (Initial Program Load) from any drive. A transfer to made to this field either by entering E SYS in response to a load on IOC device 0, or by executing an EXIT macro in a program. This field also contains the IPL drive number field _IPLDR.
_IPLDR	404C(2)	IPL Drive Number/SYSRES Pack Type	This two-character field is the 15th and 16th locations of the _SYS field. The _IPLDR field contains the IPL drive number followed by either the digit 6 if the SYSRES volume is a 41B disc pack or the digit 0 if the SYSRES volume is any other type of disc pack. This field is initialised by MAINT to point to the same drive as pointed to by _LSRD. The contents of this field may be reset to any other drive number by entering the following instructions on IOC device 0 in response to a load (assuming that Common is initialised): 0040T50002 (this issues a two-character read from IOC device 0) NP (where N = drive number, P = pack type) The preceding instructions will set the drive number into the first character of _IPLDR and the pack type into the second character; the entry of E SYS in response to a load on IOC device 0 will bootstrap SYSRES from the selected drive. If the character at _IPLDR+2 (within _SYS field) is a P, the program C_LOAD will idle until MAINT Is used to change the primary SYSRES drive number during a STARTUP operation (unless _JOPC is set to an equals sign followed by a number within the range 0 to 9).
_CLIAR	410C(100)	LIOCS Expand Pivot Area and Reserved Area (Formerly Record Protect Table _RPTAB)	When LIOCS is loaded, this area contains special entry points which are defined by the DEFINE macro. The entry points are _CPLDR, _CPRSM and _CPSNP. If no LIOCS is loaded, this area contains an error routine which indicates "NO LIOCS HERE".
_CPLIO	420C(10)	LIOCS Entry Point (_IOCS) Trap	Definition for this field applies only when LIOCS is not loaded.

Table 5-1 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
_CPLDR	410C(10)	LOADER (_LLDR) Entry Point	Definition for this field applies whether LIOCS is loaded or not.
_CPRSM	440C(10)	RESUME Entry Point	Definition for this field applies whether LIOCS is loaded or not.
_CPSNP	500C(10)	Snapshot Dump (SNAPD) Entry Point	Defintiion for this field applies whether LIOCS is loaded or not.
_DLTAB	510C(16)	Drive Lock Table	This table contains 16 one-character entries. Each entry is positionally related to the corresponding drive number (0-15). If the entry is the two characters '∇!' that volume and the next adjacent volume are not locked (i.e. available). If the entry is the four characters '∇!"£' that volume and the next three adjacent three volumes are not locked. A value of 0 (zero) or greater means that the volume and the corresponding adjacent volumes are locked.
_TLTAB	526C(4)	Tape Lock Table	This table contains four one-character entries. Each entry is positionally related to the corresponding tape drive number (1-4). If the entry is blank the drive is unlocked. If the entry is the partition number, the drive is available only to the specified partition.
_DSTAB	530C(16)	Disc Drive Status Table	This table contains 16 one-character entries. Each entry is positionally related to the corresponding disc drive number (0-15). The entry is blank if a drive may be accessed by any partition (i.e., available), or the entry may contain a code (normally installation defined) that indicates that the drive is not to be accessed without operator intervention. MAINT initialises _DSTAB from V_DSTB of the volume label of the primary resident system (pointed to be _IPLDR).
_CBUFF	546C(4)	Address of Communications Buffer	This field contains a pointer to the communications control block. This field is used by the communications access method of DMF II LIOCS. If a buffer exists, _CBUFF must contain an address within the range 1000 to 79990; if there is no buffer, _CBUFF must contain ////.
_RPTOP	550C(3)	Pointer to Top of Record Protect Table	This field contains the first three characters of tens-boundary address of the last entry in the record protect table. If no record protect table exists this field is set to 000.
_RPFRE	553C(3)	Record Protect Table: Free Pointer	This three-location field contains the tens-boundary address of the next available entry in the record protect table. If the record protect table is full or if there is no record protect table, _RPFRE is set to ///.
_RPLOK	556C(1)	Record Protect Table: Usage Lock	This field contains a blank if the record protect table is not being updated. If the table is being updated, the field contains the partition code of the program currently executing a LIOCS operation. This field is also used by the CALOC and CFREE macros when updating common.

Table 5-1 (continued)

Label	*Location & Length*	*Descriptor*	*Comments*
_CPLDR	410C(10)	LOADER (_LLDR) Entry Point	Definition for this field applies whether LIOCS is loaded or not.
_CPRSM	440C(10)	RESUME Entry Point	Definition for this field applies whether LIOCS is loaded or not.
_CPSNP	500C(10)	Snapshot Dump (SNAPD) Entry Point	Defintiion for this field applies whether LIOCS is loaded or not.
_DLTAB	510C(16)	Drive Lock Table	This table contains 16 one-character entries. Each entry is positionally related to the corresponding drive number (0-15). If the entry is the two characters '∇!' that volume and the next adjacent volume are not locked (i.e. available). If the entry is the four characters '∇!"£' that volume and the next three adjacent three volumes are not locked. A value of 0 (zero) or greater means that the volume and the corresponding adjacent volumes are locked.
_TLTAB	526C(4)	Tape Lock Table	This table contains four one-character entries. Each entry is positionally related to the corresponding tape drive number (1-4). If the entry is blank the drive is unlocked. If the entry is the partition number, the drive is available only to the specified partition.
_DSTAB	530C(16)	Disc Drive Status Table	This table contains 16 one-character entries. Each entry is positionally related to the corresponding disc drive number (0-15). The entry is blank if a drive may be accessed by any partition (i.e., available), or the entry may contain a code (normally installation defined) that indicates that the drive is not to be accessed without operator intervention. MAINT initialises _DSTAB from V_DSTB of the volume label of the primary resident system (pointed to be _IPLDR).
_CBUFF	546C(4)	Address of Communications Buffer	This field contains a pointer to the communications control block. This field is used by the communications access method of DMF II LIOCS. If a buffer exists, _CBUFF must contain an address within the range 1000 to 79990; if there is no buffer, _CBUFF must contain ////.
_RPTOP	550C(3)	Pointer to Top of Record Protect Table	This field contains the first three characters of tens-boundary address of the last entry in the record protect table. If no record protect table exists this field is set to 000.
_RPFRE	553C(3)	Record Protect Table: Free Pointer	This three-location field contains the tens-boundary address of the next available entry in the record protect table. If the record protect table is full or if there is no record protect table, _RPFRE is set to ///.
_RPLOK	556C(1)	Record Protect Table: Usage Lock	This field contains a blank if the record protect table is not being updated. If the table is being updated, the field contains the partition code of the program currently executing a LIOCS operation. This field is also used by the CALOC and CFREE macros when updating common.

Table 5-1 (concluded)

Label	*Location & Length*	*Descriptor*	*Comments*
_SMLB	557C(1)	Set Mode LB for Partition Timeout Value	This field contains the current value for use as the LB value in order not to alter the partition timeout on Model 22 processors. It is set with the MAINT SET TIMEOUT command.
_CLKFL	558C(1)	Clock Communications Flag	This field contains codes with the following meanings: *Code* — *Meaning* blank — No clock partition operating; field set by MAINT : — Clock operating normally; field set by CLOCK = — New time has been entered by MAINT ? — Clock operating; time may be unreliable; field set by CLOCK
_COMV3	559C(1)	Validation Flag No.3	This field is validatin flag no.3. When Standard Common is loaded by the MAINT program, this field is set to a value of : (colon). (_COMV1 and _COMV2)
_PSTAT	560C(20)	Partition Status Flags	This field contains 20 one-character entries. Each entry is positionally related to a corresponding partition number (0-19). An entry can contain any of the following characters: _(underscore) — Partition is executing in system idle loop (see _IDLE) blank — Partition status undetermined L — In middle of a C_LOAD operation at EOJ C — Partition is executing with common LIOCS E — Partition is executing RPG compiler N — C_LOAD beginning execution of a program Any other value — User is defined MAINT initialises the partition flags to blanks; MAINT will not load a version of LIOCS without operator intervention if any partition status flag contains either a C or an R.
_MFLAG	580C(1)	Mailbox Availability Flag	This field contains a B if the mailbox is in use. If _MFLAG contains either an A or any other value but B, the mailbox is available. The mailbox macros MBPREP, MBSEND, MBRECV and MBRELS are used to test, set and release the mailbox. Fields _MTO, _MFROM and _MREC also contain mailbox control information.
_MTO	581C(2)	Mailbox 'TO' Partition Number	This field contains a user-defined code that identifies the partition and/or program that is to receive a block of data via the system mailbox. The code may be any two characters except: @,\, [,],^ , or _.
_MFROM	583C(2)	Mailbox "FROM" Partition Number	This field contains a user-defined code that identifies the partition and/or program that is sending a block of data via the system mailbox. The code may be any two characters except: @,\, [,],^ , or _.
_MREC	585C(1)	Mailbox Record Buffer	This field has an implicit length of 1 and is the first location of the system mailbox. The size of the mailbox is 415 locations.

Section 6
DISC RESOURCES ACCESS PROCEDURES

The following paragraphs describe how a volume may be accessed.

VOLUME ACCESS

A partition may only access a volume on a drive which is available (meets any one of the following conditions) to a partition.

1 The drive's _DSTAB (device table) entry is blank or is already locked to this partition, or DMF II common is not initialized

2 The drive is the system residence drive of this partition (_LSRD in DMF II partition low core)

3 The drive has been assigned to this partition by being entered by the operator through CONI in response to the volume mount message

Initial access to a volume other than the System Residence Drive of a partition is through the volume label (sector 1 of a Model 41 and 41A pack; sector 60001 of a Model 41B pack). If a volume does not have a volume label in the appropriate sector (identified by 'VOL' in the first three positions of the label), the volume is considered unlabelled, and all access is unrestricted (provided the drive the volume is available as defined by items 1 and 3 of the above descriptions). Additionally, all access to areas on a labelled volume other than the DMF II area is unrestricted as above under unlabelled volumes.

The DMF II area of a labelled volume is the first 25 sectors (beginning at sector 0 of a Model 41 and 41A pack and at sector 60000 of a Model 41B pack) of a SYSRES volume (location V_SYS of the volume label is "1") or the first two sectors of an interchangeable volume (location V_SYS is "2") and the DMF II extent defined in locations V_LLIM and V_HLIM.

Volume Search Algorithm

This algorithm is used to search all available disc drives for a volume specified by system name, ID, and an address/description (system name only needed for COPY operation). The following procedure is performed:

1 Compute next drive to be tested. For DMF II System Programs, the volume's logical drive (default is 1) is searched first followed by sequentially higher drives

through 15 (drive code is "?") and then beginning with the next lower drive from the volume's logical drive followed by sequentially lower drives through 0. Drives 10 through 15 must not be searched when DMF II Common is not intialized since this will cause a program check on Model 20 processors

2 If DMF II Standard Common is initialized (_LCOM_0), test appropriate position for drive in _DSTAB. If _DSTAB entry is not blank, return to item 1

3 If not Volume Pointer transversal (P_VLOC is volume label sector address, positions 60 ghrough 65) set volume label sector address to 00001

4 Read volume label. If a fault or an error exists, (condition codes 1, 3, and 4), return to item 1

5 Test volume label for "VOL" at B_LAB (volume label indicator) and for match with V_SNAM (system name), V_ID (volumeID), V_DESC, or match with V_SNAM and V_SYS (volume system resident indicator) equal to "1" for COPY with system name specifiction. If not "VOL", return to item 1. If not match and if not Volume Pointer transversal and V_TYPE (volume type) or previously read label is 'L', set volume label sector address to 60001 and go to item 4; otherwise, return to item 1

Operator Volume Mount Message

When the system name, volume ID, and address/description of the volume are known and all available drives have been searched, an operator volume mount message is output on CONO. The message is output in the following format:

A)ENTER DRIVE NUMBER OF VOLUME !!! FROM SYSTEM $$$$$$ — @@@@@@@@@@@@@@

where

!'s = the volume ID.

$'s = the system name.

@'s = the address/description.

The operator must reply to the mount message with a one-character entry(0-?) identifying the disc drive to be tested for the desired volume.

The read instruction is two characters to allow for error key entry.

If CONO or CONI is not assigned, a CONI fault condition is detected (i.e., power off or no device), or a flag condition on CONI is received (i.e., control key preset for console devices or a unit separator is entered for read-only device) the search for the volume is terminated.

The disc drive specified by the operator's response is accessed using the previously described volume search algorithm for only the specified drive (this results in the mount message being reissued during the second execution of item 1).

VOLUME LABEL ACCESS

Access to a volume label on a drive may be performed only if:

1 The drive's status is available

2 The drive is the system resident drive for this partition (_LSRD)

3 The drive has been assigned to this partition by a response to a mount message

The procedures required to either alter (write) or read a volume label are described in the following paragraphs.

Volume Label Write Access

A write to a volume label, other than during volume initialization, can only be performed if the volume is locked. The locked volume label can (only) be altered by the locking partitions.

The steps required to lock an available volume label are:

1 Set System Lock (_SYSLK)

2 Read the Volume Label.

If the Volume Label is locked by another partition, turn off System Lock and optionally re-execute this algorithm. If the attempt to lock the volume is abandoned due to disc fault or disc errors, System Lock must be turned off.

3 Set V_LPTN (Volume Locking Partitions Code) to this partition's code (_LPCD)

4 Rewrite the locked Volume Label

If the attempt to write the altered Volume Label is abandoned due to disc fault or disc errors, System Lock must be turned off (step 5)

5 Turn off System Lock

A Volume Label may be unlocked (V_LPTN is set to blank and the Volume Label is rewritten) at any time by the partition which has it locked.

Volume Label Read Access

Read access to a volume label on a drive which is available to this partition is not restricted.

VOLUME INITIALIZATION

The initialization of a volume (i.e., altering the Volume Label, reformatting the Pool Directory, sector 0 on all volumes, and sectors 2-24 on SYSRES volumes) which already contains a Volume Label must be verified by the operator via CONI while the Volume Lock is set.

The initialization of a volume on a drive available to this partition must be performed using the following steps:

1 If the first three characters of the Volume Label sector are 'VOL', set this partition's code in V_LPTN of Volume Label (see Volume Label Write Access). If the first three characters of the Volume Label sector are not 'VOL', go to step 5 to write

2 If the Volume Label has no password (V_PASS is blank), skip to step 4. Otherwise, the operator must enter the password. The following message is output on CONO:

```
A)ENTER PASSWORD OF VOLUME !!! FROM SYSTEM $$$$$$ -- @@@@@@@@@@@@@
ON DRIVE *.
```

where

!'s = the volume label's ID.

$'s = the system name.

@'s = the address description.

*'s = the physical drive volume it is currently mounted on.

The password is then read with a seven character read from CONI in "fill mode". The first six characters are used to compare with the password. If CONI is a workstation, the read is in "nonprint mode". If CONI is a VDU, the line is erased after the read.

If:

a) CONO or CONI is not assigned,

b) a fault condition is received on CONI (power not on or device doesn't exist) or

c) a flag condition is received on CONI (control key pressed for console devices or unit separator for read only devices),

turn off the Volume Lock (set V_LPTN to blank and rewrite volume label without setting any locks) and abort the operation

3 Compare password entered by operator with volume label password (V_PASS). If no match, re-execute step 2

4 Display the first 55 characters (V_LAB through V_DESC) of the current Volume Label contents on CONO and request verification of volume change via CONI. If the volume change is denied, turn off the Volume Lock (set V_LPTN to blank and rewrite volume label without setting any other locks) and abort the operation

5 Create new volume label. Set V_LPTN to _LPCD and V_PTP to '//////' and write the volume label

6 Reformat Pool Directory and write sector 0 of all volumes and sectors 2-24 of SYSRES volumes

7 Rewrite Volume Label with V_LPTN set to blank and V_PTP properly set to point to the root of the pool tree

DMF II SYSTEM AREA ACCESS OPERATIONS

The DMF II system area is located in sector 0 (sector 60000 of a model 41B volume) on all standard (model 40) volumes and sectors 2 through 24 (sectors 60002-60024 of a model 41B volume) on a standard (model 40) SYSRES volume. The DMF II system area is formatted during volume initialization procedures.

The methods used to write into and to read from the system areas are described in the following paragraphs.

DMF II System Area Write Access

The following changes (write operations) to the DMF II System area on an available SYSRES volume may be performed only when the SYSRES drive is locked (drive lock):

1 altering the DMF II System Configuration Table

2 altering the _LLDA location in "partition low core" on disc sector 3 when C_LOAD file label is added or deleted in SYSPOL

3 altering the _LSYS location in "partition low core" on disc sector 4 when the SYSPOL pool label is either added or deleted by any of the following: MAINT DELETE, RENAME, COPY, MOVE, RESTORE, FILE or CREATE

The alteration of any portion of the DMF II System Configuration Table entry for any partition - other than the host partition - on any SYSRES volume can be performed only if that partition's protection entry is off (i.e., bit 7 of C_SRD - Partition Protection flag - in the Primary System Configuration Table is 0). If DMF II standard common is initialized, the Configuration Table entry for another partition can only be performed if the _PSTST value for that partition is not the character 'L'.

If DMF II standard common is initialized (_LCOM>0) the identity of the drive on which SYSRES is mounted is contained by the common core constant _IPLDR; location _IPLDR+1 indicates whether the volume disc pack is a Model 41B (_IPLDR+1 = 6) or a Model 40 (_IPLDR+1 = 0).

If DMF II standard common is not initialized (_LCOM = 0) the Primary Resident System Volume is mounted on drive 0-with the configuration table located in sectors 15 through 24 (Model 40 disc pack).

The following steps must be performed to alter the DMF II system area:

1 Turn on the drive lock in the disc drive status table (_DSTAB) for the drive containing the system area

2 Perform the desired alterations

3 Turn off the drive lock set on in step 1

DMF II System Area Read Access

Read access to the DMF II Systems Area of any volume on a drive available to this partition is unrestricted.

POOL DIRECTORY ACCESS OPERATION

The procedures used to either write into or read from a pool directory are described in the following paragraphs.

Pool Directory Write Access Operations

The lock of the volume which contains the pool directory to be accessed must be set on before making any pool space allocation or any change to the structure of the pool directory (adding/deleting pools labels, changing the name of the pool, etc.). Unlocked pool labels may be altered (written into) only when the system lock is set on.

The password must be supplied when a pool is password protected. The following steps must be performed to add or delete a pool label, make a pool space allocation or change the name of the pool:

1 Set V_LPTN of the Volume Label to this partition's code (_LPCD) using the algorithm for setting Volume Lock

2 If the operation being performed requires space allocation, search Pool Directory of volume to determine if space is available. If allocation is abandoned due to insufficient space, disc fault, or disc errors, turn Volume Lock off (step 6)

3 If the operation being performed is a MAINT pool RESTORE when the pool label pre-exits, the pool label is locked (P_LOCK set to "L" and P_LPTN to _LPCD) and P_FTP and P_FSP are set to slashes using the Pool Label Write Access Algorithm

4 If new pool space is being created for a Linked Pool or a Relative/Variable Pool/File, format pool space allocated. If the formatting is abandoned due to disc fault or errors, turn Volume Lock off (step 6)

5 Perform alterations to the pool labels and the pool directory. Alteration of any unlocked pool label other than P_TYPE = @ must be performed while System Lock is set. No portion of a pool label locked to another partition may be

altered. To delete a pool label with a password, the password must be supplied

6 Rewrite Volume label with V_LPTN set to blank

Pool Directory Read Access Operations

Read access to the pool directory of a volume mounted on an available drive does not require the use of locks; access is, essentially, unrestricted.

POOL LABEL ACCESS OPERATION

The procedures used to either write into or read from a pool label are described in the following paragraphs.

Pool Label Write Access Operations

An alteration (write operation) of the contents of a pool label (such as setting the pool lock, changing the pool password, etc.) must be performed while the System Lock is set unless the pool label is already locked for this partition. In addition, the password must be supplied to:

1 alter the pool name (P_NAME)

2 alter the password (P_PASS and P_PROT)

3 alter the text fields (P_TEXT)

4 delete the pool label. (See Pool Directory Write Access)

The following steps must be performed to alter a pool label not currently locked by the partition involved:

1 Turn on System Lock (_SYSLK)

2 Search the Pool Directory for pool label. This step optionally follows setting the System Lock and may optionally include reading the Volume Label to access the pool directory and following volume pointers. If the address of the pool label is known, this step may be omitted

3 Read the pool label. If the pool label is locked by another partition (P_LOCK = 'S' or 'L' with P_LPTN not this partition), turn off System Lock and optionally re-execute this algorithm. (Retries may depend on whether the lock is long-term or short-term.) If the attempt to alter the pool

label is abandoned due to disc fault or disc errors, System Lock must be turned off (step 6)

4 Make alterations to the pool label in core

5 Rewrite the altered pool label. If the attempt to write the altered pool label is abandoned due to disc fault or disc errors, System Lock must be turned off (step 6)

6 Turn off System Lock.

Pool Label Read Access

Read access to a pool label does not require locks and is unrestricted except that the pool password may not be written to any System Ten output device (IOC, FAC, SCA, etc.) except as noted in the Pool Data Read Access description.

POOL DATA AREA ACCESS OPERATIONS

Read/Write operations on a pool data area involve the entire contents of the pool (including the file directory of a linked sequential pool).

Deletion of either the file directory of a linked pool or the deletion of any pool label requires the pool password to be supplied. Transmitting an entire pool to any System Ten output device (MAINT BACKUP) requires the pool password to be supplied if the pool is read protected (P_PROT not equal to 'W'). The password may not be transmitted except when a copy of the pool contents with passwords retained is created (new pool creation - MAINT MOVE/COPY).

Linked Pool Free Sector List Access Operations

Returning/removing sectors to/from a linked pool free sector list may only be performed while the pool is locked for this partition.

A short-term pool lock (P_LOCK = 'S') is used for returning/removing a list of one or more sectors to/from the pool's free sector list without any intervening processing. Typical short-term free sector list access operations are allocating/deallocating a private free sector list, or removing/returning a sector to create/delete a file label.

A long-term pool lock (P_LOCK = 'L') is employed for the exclusive use of a pool's free sector list such as Out/Extend/Work access under LIOCS to a linked sequential file in "noncontention" mode (FCB ALL)C=NO).

Access to a pool's free sector list must be performed using the following steps:

1 Set the appropriate lock type ('S' or 'L') in P_LOCK and this partition's code in P_LPTN in the pool label. Use the algorithm for altering a pool label (See Pool Write Access)

2 Perform the short-term or long-term access to the pool's free sector list. Once written with a lock, a pool label may be either kept in core or reread prior to updating

3 Set the updated free sector list pointer in P_FSP, the updated free sector count in P_NFS, P_LOCK to 'U' and P_LPTN to blank in the pool label and rewrite without using any locks. Alterations may be made to the pool label as it was when the lock was set since a locked pool label can't be altered by any other partition

Accessing Relative or Variable Pool or File Work Pool Areas

The methods by which various types of pool or file work areas may be accessed are described in the following paragraphs.

Relative/Variable Pool/File and Relative Work Pool Exclusive Use Access

Any utilization of a Relative Work Pool (use of a Relative/Variable Pool as a Work Pool) or exclusive use access to a Relative Access, Direct Access, Variable Access) must be performed while the Pool Label is long-term locked.

Utilization of a Relative Work Pool or exclusive use access to a Relative/Variable Pool/File must be performed with the algorithm for Scratch Work Pool Access.

Scratch Work Pool Access

Any utilization of a Scratch Work Pool (use of a pool with P_TYPE = "W" or use and conversion of a linked pool with no files to P_TYPE = "W") must be performed while the pool label is long-term locked (P_LOCK = "L").

Utilization of a Scratch Work Pool must be performed with the following algorithm:

Step 1: Set 'L' in P_LOCK and this partition's code in the pool label using the algorithm for altering a pool label (See Pool Label Write Access)

Step 2: Access the pool

Step 3: To release the pool, set P_LOCK to 'U' and P_LPTN to blank in the pool label and rewrite without using any locks

Work or Relative/Variable Pool Write Access

Utilization of a Work Pool or any alteration of the data of a Relative/ Variable Pool/File (Relative Access, Direct Access, Variable/Access) requires the pool password to be supplied if the pool/file is password protected.

Relative/Variable Pool/File Read Access

Read access to a Relative/Variable Pool/File (Relative Access, Direct Access, Variable Access) requires the file password to be supplied if the file label is read protected (F_PROT is not equal to "W").

LINKED POOL FILE DIRECTORY ACCESS OPERATIONS

The procedures used to either write into or read from a linked pool file directory are described in the following paragraphs.

File Directory Write Access Operations

The addition or deletion of a file label in a linked sequential pool file directory must be performed while Drive Lock is set; since these operations require manipulation of the pool free sector list (to retrieve or return the sector used for the file label), the pool must be short-term locked. A file label which is locked but not to this partition cannot be deleted. When the file label is password protected, the password must be supplied to delete the file label.

The following steps must be performed to add or delete a file label in a Linked pool file directory:

1 Set 'S' in P_LOCK and this partition's code in P_LPTN in the pool label using the algorithm for altering a pool label (See Pool Lable Write Access)

2 Turn on Drive Lock for the drive on which the volume containing the pool is mounted. If the attempt to alter the file directory is abandoned because the drive cannot be locked, turn off pool lock (step 5)

3 Make the appropriate alteration to the pool file directory and free sector list. If the attempt to alter the file directory is abandoned due to disc fault, disc errors, inability to delete a locked file label or no free sector

available to create a file label, the last two steps of this algorithm must be executed before remedial action is taken

4 Turn off the drive lock

5 Set the updated free sector list pointer in P_FSP, the updated free sector count in P_NFS, the updated File Directory root if changed in P_FTP, P_LOCK to 'U', and P_LPTN to blank in the pool label and rewrite without setting additional locks. Alterations may be made to the pool label as it was when the lock was set since a locked pool label can't be altered by any other partition

File Directory Read Access Operations

Read access to a Linked pool file directory does not require locks and is unrestricted except as noted on the Linked File Label Read Access description.

LINKED FILE LABEL ACCESS OPERATION

The procedures required to either write into or read from the label of a linked file are described in the following paragraphs.

Linked File Label Write Access

Any alteration of the contents of a Linked file label (such as setting file lock, altering the file's password, changing file beginning address, etc.) must be performed while the drive lock is set, (including when the file label is already locked for this partition). When the file label is password protected, the password must be supplied to alter the file name (F_NAME), password(F_PASS and F_PROT), text (F_TEXT) fields of the file label or the data-oriented portions of the file label as part of Linked Sequential File Data Write Access (F_BEG, F_TYPE, F_IATR, F_BL, F_RL, F_KL, F_KD, F_REV, F_DPN, F_STAT, F_SCNT, F_EOFA, F_DFN) and to delete the file lable (See Linked File Directory Write Access description).

The following steps must be used to alter the label of a linked file:

1 Turn on Drive Lock (_DLTAB) for the drive on which the volume containing the file is mounted

2 Search the file directory for the file label. This step optionally follows setting the Drive Lock. If the address of the file label is known, this step can be bypassed

available to create a file label, the last two steps of this algorithm must be executed before remedial action is taken

4 Turn off the drive lock

5 Set the updated free sector list pointer in P_FSP, the updated free sector count in P_NFS, the updated File Directory root if changed in P_FTP, P_LOCK to 'U', and P_LPTN to blank in the pool label and rewrite without setting additional locks. Alterations may be made to the pool label as it was when the lock was set since a locked pool label can't be altered by any other partition

File Directory Read Access Operations

Read access to a Linked pool file directory does not require locks and is unrestricted except as noted on the Linked File Label Read Access description.

LINKED FILE LABEL ACCESS OPERATION

The procedures required to either write into or read from the label of a linked file are described in the following paragraphs.

Linked File Label Write Access

Any alteration of the contents of a Linked file label (such as setting file lock, altering the file's password, changing file beginning address, etc.) must be performed while the drive lock is set, (including when the file label is already locked for this partition). When the file label is password protected, the password must be supplied to alter the file name (F_NAME), password(F_PASS and F_PROT), text (F_TEXT) fields of the file label or the data-oriented portions of the file label as part of Linked Sequential File Data Write Access (F_BEG, F_TYPE, F_IATR, F_BL, F_RL, F_KL, F_KD, F_REV, F_DPN, F_STAT, F_SCNT, F_EOFA, F_DFN) and to delete the file lable (See Linked File Directory Write Access description).

The following steps must be used to alter the label of a linked file:

1 Turn on Drive Lock (_DLTAB) for the drive on which the volume containing the file is mounted

2 Search the file directory for the file label. This step optionally follows setting the Drive Lock. If the address of the file label is known, this step can be bypassed

3 Read the file label. If the file label is locked by another partition, turn off Drive Lock and optionally re-execute this algorithm. If the attempt to alter the file label is abandoned due to disc fault or disc errors, Drive Lock must be turned off (step 6)

4 Make alterations to the file label in core

5 Rewrite the altered file label. If the attempt to write the altered file label is abandoned due to disc fault or disc errors, Drive Lock must be turned off (step 6)

6 Turn off Drive Lock

Linked File Label Read Access

Read access to a Linked file label does not require locks and is unrestricted except for access to the data-oriented portions of the file label as part of Linked Data Read Access, (F_BEG=F_EOFA).

The file password cannot be transmitted to any System Ten output device except when a copy of the entire pool in which the file resides is created with passwords retained (See Pool Data Area Read Access descriptions) or when a copy of the entire file is made with the password retained (new file-MAINT MOVE/COPY).

LINKED DATA FILE ACCESS

The methods by which a linked data file may be addessed are described in the following paragraphs.

Linked File Data Exclusive Use Access

Exclusive use access to a Linked file must be performed while the file label is locked. Exclusive use access to a Linked file must be performed while the file label is locked. Exclusive use access to a Linked File must be performed using the following steps:

1 Set this partition's code in F_LPTN in the file label using the algorithm for altering a file label (See Linked File Label Write Access description)

2 Perform the required access to the file data

3 Set F_LPTN to blank in the file label using the algorithm for altering a file label

The validity of exclusive use is not guaranteed unless all programs use the file for exclusive use. Otherwise, a program could be using the file for shared use when the file lock was set.

Linked File Data Write Access

Any alteration of the data contents of a Linked file or corresponding file label fields requires the file password to be supplied.

Linked File Data Read Access

Read access to the data contents of a Linked file requires the file password to be supplied if the file label is read protected (F_PROT is not equal to 'W').