BDRUM: FASTRAND DRUM DATA-FILE CONTROL

1.0.0 Contents:

2.0.0 General: Data-files on the FASTRAND drum(s).

3.0.0 BDRUM Input/Output Control PROCedure:

- 3.1.0 BDRUM call line.
- 3.1.1 Example - BDRUM call line.

4.0.0 BDRUM Directives:

- 4.1.0 GIVE directive: Request A Data-File Area
  - 4.1.1 Entrance requirements.
  - 4.1.2 Normal exit conditions.
  - 4.1.3 Example - GIVE directive.
  - 4.1.4 Additional notes: GIVE directive.
- 4.2.0 DFIND directive: Find An Existing Data-File Area
  - 4.2.1 Entrance requirements.
  - 4.2.2 Normal exit conditions.
  - 4.2.3 Example - DFIND directive.
- 4.3.0 DCLOS directive: Close An Opened File
  - 4.3.1 Entrance requirements.
  - 4.3.2 Normal exit conditions
  - 4.3.3 Example - DCLOS directive.
- 4.4.0 DREL directive: Close An Opened File And Release Drum Area
  - 4.4.1 Entrance requirements.
  - 4.4.2 Normal exit conditions
  - 4.4.3 Example - DREL directive.

4.5.0 DREAD directive: Read A Sector(s)

4.5.1 Entrance requirements.

4.5.2 Normal exit conditions.

4.5.3 Example - DREAD directive.

4.6.0 DWRIT directive: Write A Sector(s)

4.6.1 Entrance requirements.

4.6.2 Normal exit conditions.

4.6.3 Example - DWRIT directive.

4.7.0 DSRCH directive: Search And Read Sector(s)

4.7.1 Entrance requirements.

4.7.2 Normal exit conditions.

4.7.3 Example - DSRCH directive.

4.8.0 DHEAD directive: Position Head-Bar

4.8.1 Entrance requirements.

4.8.2 Normal exit conditions.

4.8.3 Example - DHEAD directive.

4.9.0 DFUNC directive: Deliver User Specified Drum Packets To OPR.

4.9.1 Entrance requirements.

4.9.2 Normal exit conditions.

4.9.3 Example - DFUNC directive.

2.0.0 General:

The BDRUM input/output routines and its directives provide for the control of data-files on the FASTRAND drum(s), and must be used with the operating system (OPR). Control is provided for files on FASTRAND I, II, and MODULAR drum(s) configurations.

Before these routines can be used, the Initial-Drum-Set-Up routine (IDMS) must be run. Basically, IDMS "maps" the drum(s) by writing a series of directory entries, and provides the drum-locator-loader (DLL) which is written in the first track of drum zero (0). Through the use of parameter cards, the user can "map" the drum(s) according to his needs. IDMS must be run before the FASTRAND drum software can be used.

The smallest addressable unit of data which can be accessed by the BDRUM routines is one FASTRAND drum sector (168 characters). (an exception to this: see DFUNC directive). If "chaining" of data-sectors becomes necessary, the following should be carefully observed:

(1) Only single level chaining is to be used.

(2) The last eight characters of all data-sectors cannot be used for data-storage; this limits the number of characters available within each sector to 160.

(3) The last eight characters (160-167) of each sector are used as follows:

(a) Characters 160-163:

If unused sector:	.binary zeroes.
If 1st sector of chain	.077777777.
If last sector of chain. . . .	.the drum address of previous link.
If within a sector chain . . .	.the drum address of previous link.

(b) Characters 164-167:

If unused sector:	.binary zeroes.
If 1st sector of chain:	.the drum address of 1st link.
If last sector of chain:	.binary zeroes.
If within a sector chain:	.the drum address of next link.

All software development will follow this method of single level chaining.

3.0.0 BDRUM Input/Output Control PROCedure:

The BDRUM call directs the assembler to include the drum input/output control routines in the worker program, with a linkage to the operating system. These routines conform to the FASTRAND drum data-file conventions established in the FASTRAND drum software specifications document.

The BDRUM directives provide the functions for:

(a) Issuing unused drum area, suitable for writing data-files, to the worker program, (GIVE).
(b) Locating existing data-files on the drum(s), (DFIND).
(c) Closing data-files without releasing the associated drum area, (DCLOS).
(d) Closing data-files with release of the associated drum area, (DREL).
(e) Reading data-files, (DREAD).
(f) Writing data-files, (DWRIT).
(g) Searching data-files, (DSRCH).
(h) Positioning the head-bar, (DHEAD).
(i) Issuing user generated FASTRAND drum packets to the operating system, without the use of the above directives, (DFUNC).

All addresses submitted to the BDRUM directives are validated to ensure that they are within the bounds of existing opened files. This is to prevent the inadvertant destruction or accessing of files not intended. This is done even though the user submits a packet directly by the DFUNC directive.

All data-files on the FASTRAND drum(s) are referenced by two names:

1. CLASS name - this is the name which was supplied to the Initial-Drum-Set-Up routine (IDMS), within which the file name exists.

2. FILE name - this is the name supplied to the GIVE directive when opening a new data-file. All future references to this file must include this name, and it must be unique within its CLASS.

Both the CLASS and FILE names referencing data-files must be eight characters in length; the left most character must be alphabetic.

3.1.0 The BDRUM call line (must appear only once in the worker program).

LABEL	OP'N	OPERANDS
	BDRUM	P1, P2, P3, P4, P5

Where: P1 the number of files to be used, at any one time, by the worker program (max. 63).

The BDRUM PROCedure maintains a table in core memory which serves as an index for the files to be used by this worker program. This table can accommodate up to 63 files. An entry is made to this table whenever a GIVE or DFIND directive is executed (if a DFIND is executed for a file which is already existing in the table, the entry will not be duplicated); an entry is deleted whenever a DCLOS or DREL directive is executed. If it becomes necessary to have more than 63 different files available to the worker program, it is suggested that the most active of these files be left in the table. The files which are less frequently used can be opened (DFIND) and closed (DCLOS) when needed; however, the user should be aware that this is time consuming, because of the additional FASTRAND drum accesses required for these directives.

Refer to the sections of this document for detailed descriptions of the DFIND, GIVE, DCLOS, and DREL.

P2,P3 two index registers which can be used by the BDRUM directives. Whenever a directive is executed the contents of these index registers will be changed (1,2...not X1,X2,...).

P4 the label of a subroutine to which a JR will be executed in the event of an unrecoverable FASTRAND drum error; the contents of tetrads 56, 57, 58, and 59 will be in AR1. If control is returned, the order will be considered successful.

P5 RT, if this program's FASTRAND drum orders are to be given priority (when running concurrently). Both programs in core cannot be so designated.
Otherwise, this parameter is omitted.

3.1.1 Example: (as the BDRUM call line might appear in a worker program).

LABEL	OP'N	OPERANDS	COMMENTS
(1)	BDRUM	20, 6, 7, ERROR, RT	

. Comments: (1) 20 - provide for 20 files.

. 6,7 - index registers (X6 and X7).

. ERROR -User's subroutine which will be entered in the event of an unrecoverable FASTRAND drum error.

. RT - this program's FASTRAND drum orders are to be given priority.

4.0.0 BDRUM Directives:

4.1.0 To request an unused FASTRAND drum area suitable for writing a data-file: (to open a new file).

LABEL	OPCN	OPERANDS
	BAl	mmmm, 4
	GIVE	P1, P2, P3, P4

where:

mmmm the address of the LSC of a 4 character field which contains the number of sectors, in binary, requested.

P1 the address (LSC) of the file name which is to be assigned to this file.

P2 the address to which program control will be transferred if:

a. the file directory is full (1)
b. the data area is full (2)
c. the class (P4) cannot be located on the drum(s) (3)

at the time program control is transferred to P2, the LSC of AR1 will contain, in binary, 1, 2, or 3 indicating the reason. (value in parenthesis above).

P3 HEAD, if the file area is to start at head zero (this minimizes head movement during data-file processing).

Ø, if the file area is to start at any head.

P4 the address (LSC) of the class name, within which this file is to be located.

Note: these classes were created by IDMS from user supplied parameter cards.

4.1.1 Entrance requirements:

(1) the 4 LSC of AR1 must contain the number of sectors requested (in binary).

4.1.2 Normal exit conditions:

(1) The 4 LSC of AR1 contain the address of the first sector of the file area supplied (absolute drum address). The file is open.

(2) Functions performed:

a. the class name (P4) was located in the systems directory.

b. the file directory entry was written on the FASTRAND drum for the file (FNAME).

4.1.3 Example: (as the GIVE directive might appear in a worker program.)

	LABEL	OP'N	OPERANDS
(1)		BAl	SIZE, 4
(2)		GIVE	FNAME, ERR1, HEAD, CNAME
(3)		SAl	FBEGN, 4

Comments:

(1) AR1 is loaded with the number of sectors requested.

(2) FNAME - address of the file name.

ERR1 - address to which control will be transferred if the file area is full, the directory is full, or the class (CNAME) does not exist on the drum.

HEAD - this file is to start at head zero.

CNAME - address of the class name.

(3) the sector address (first sector of area) is stored in FBEGN for worker program reference and use.

4.1.4 Additional information available at normal exit time:

(1) the image of the file directory entry is still in core memory. The MSL of the image is at WRDAR.

(2) At WSARR (LSC of 4 character field) is the address PLUS ONE of the FASTRAND file directory entry.

Note: The information in (1) and (2) should not be used unless the user has a thorough understanding of the file and systems directory entries and their purposes in all software packages. Refer to the FASTRAND drum software specifications document.

4.2.0 To locate an existing data-file (to open an existing data-file):

LABEL	OP'N	OPERANDS
	DFIND	P1, P2, P3

Where:

P1 the address of the file name.

P2 the address to which control will be transferred if the file cannot be found.

P3 the address of the class name.

4.2.1 Entrance requirements:

(1) the file should exist on the FASTRAND drum(s). If it does not exist, control will be transferred to P2.

4.2.2 Normal exit conditions:

(1) THE 4 LSC of AR1 contain the address of the first sector of the data-file area. The file is now open.

(2) refer to 4.1.4 (additional information available to the worker program, as after a GIVE.).

4.2.3 Example: (as the DFIND directive might appear in a worker program).

LABEL	OP'N	OPERANDS
(1)	DFIND	FNAME, NOFLE, CNAME
(2)	SA1	FBEGN, 4

. Comments: (1) FNAME - the address of the file name.

NOFLE - address to which control will be transferred if the file cannot be found on the drum(s).

CNAME - address of the class name.

(2) FBEGN - now contains the address of the first sector of the file area, for use by the worker program.

4.3.0 To close a file without releasing the associated drum area (the file entry in memory is deleted):

LABEL	OP'N	OPERANDS
	DCLOS	P1, P2, P3
	Where:	P1, P2, and P3 have the same meaning as for the DFIND directive: refer to 4.2.0 .

4.3.1 Entrance requirements:

(1) the file should be open (by a previous GIVE or DFIND):

4.3.2 Normal exit conditions:

(1) the table entry, in core memory, has been deleted.

4.3.3 Example: (as the DCLOS directive might appear in a worker program).

LABEL	OP'N	OPERANDS
(1)	DCLOS	FNAME, NOFLE, CNAME
. Comments:		refer to the comments of the DFIND directive, (4.2.3).

4.4.0 To close a file and release the associated drum area for it (the area originally requested by the GIVE directive for this file will be released and made available for re-assignment to another file).

LABEL	OP'N	OPERANDS
	DREL	P1, P2, P3

Where: P1, P2, and P3 have the same meaning as for the DFIND directive: refer to 4.2.0 .

4.4.1 Entrance requirements:

(1) the file should be open (by a previous GIVE or DFIND).

4.4.2 Normal exit conditions:

(1) the table entry, in core memory, has been deleted.

(2) the drum area associated with this file has been released, and is available for re-assignment.

4.4.3 Example: (as the DREL directive might appear in a worker program).

	LABEL	OP'N	OPERANDS
(1)		DREL	FNAME, NOFLE, CNAME

. Comments: refer to the comments of the DFIND directive, (4.2.3).

4.5.0 To read one or more sectors from a data-file:

LABEL	OP'N	OPERANDS	
	SC	XSECN, n	this line is required on multiple sector read directives only.
	BA2	mmmm, 4	
	DREAD	P1, P2, P3	

Where:

n — the number of sectors to be read, if more than one. (XSECN is an addressable label within the BDRUM PROCedure.)

mmmm — the address (LSC) of a 4 character field in core memory which contains the absolute drum address of the first sector to be read from the drum; in multiple reads it is the address of the first sector to be read.

P1 — core memory address of the read-in area (MSL).

P2 — the address of a subroutine to which a JR will be executed if the address (mmmm) is not within the bounds of an existing opened file. If control is returned, the sector will be read.

P3 — the address of a subroutine to which a JR will be executed as soon as the FASTRAND drum order is initiated, but before it is completed. No other FASTRAND drum orders can be issued until program control is returned to BDRUM by exiting from this routine.

If this parameter is omitted, control will be retained by BDRUM until the order is completed.

4.5.1 Entrance requirements:

(1) the file must be open (see GIVE and/or DFIND).

(2) the 4 LSC of AR2 must contain the absolute drum address of the first sector to be read.

4.5.2 Normal exit conditions:

(1) the sector(s) is in core memory, beginning at the address specified by P1.

4.5.3 Example: (as the DREAD directive might appear in a worker program).

	LABEL	OP'N	OPERANDS	COMMENTS
(1)	EXMP1	BA2	SADDR, 4	read one sector.
(2)		DREAD	FREAD, ERR2, IRL1	

. Comments: (1) SADDR - the address (LSC) of a 4 character field in core memory which contains the address of the sector to be read from the drum.

. (2) FREAD - MSL of the read area in core memory.

. ERR2 - label of subroutine to which a JR will be executed if the address is invalid. (not within the bounds of an existing opened file).

. IRL1 - Immediate Return Line.

(3)	EXMP2	SC	XSECN, 24	read 24 sectors.
(4)		BA2	SADDR, 4	
(5)		DREAD	FREAD, ERR2	

. Comments: (1) XSECN - the contents of this field (XSECN,2) represents the number of sectors to be read. If a head-overflow occurs, BDRUM will execute a second order to complete the multiple sector read. In this example the read is for 24 sectors.

. (5) Note; if no IRL (P3) is given, BDRUM will retain control until the DREAD order is completed.

4.6.0 To write one or more sectors in a data-file:

LAB	OPN	OPERANDS	
	SC	XSECN, n	this line is required on multiple sector write directives only.
	BA2	mmmm,4	
	DWRIT	P1, P2, P3	

Where:

n — the number of sectors to be written, if more than one.

mmmm — the address (LSC) of a 4 character field in core memory which contains the absolute drum address of the first sector to be written on the drum; in multiple writes it is the address of the first sector to be written.

P1 — core memory address of the write-out area (MSL).

P2 — the address of a subroutine to which a JR will be executed if the address (mmmm) is not within the bounds of an existing opened file. If control is returned, the sector will be written.

P3 — Immediate Return Line (IRL). (See 4.5.0, P3, for detailed explanation)

4.6.1 Entrance requirements:

(1) the file must be open (see GIVE and/or DFIND).

(2) the 4 LSC of AR 2 must contain the absolute drum address of the first sector to be read.

4.6.2 Normal exit conditions:

(1) the sector(s) have been written, beginning at the address specified by P1.

4.6.3 Example: (as the DWRIT directive might appear in a worker program).

LABEL	OP'N	OPERANDS	COMMENTS
EXMP1	BA2	SADDR, 4	write one sector.
	DWRIT	FWRIT, ERR3, IRL2	
. Comments:		same basic format as DREAD, see 4.5.3.	
EXMP2	SC	XSECN, 10	write 10 sectors.
	BA2	SADDR, 4	
	DWRIT	FWRIT, ERR3, IRL2	
. Comments:		see DREAD, 4.5.3.	

4.7.0 To search sectors within data-files (search with 8, 16, or 32 character keys; for equal or equal/greater):

LABEL	OP'N	OPERANDS	COMMENTS
	SC	XSECN, n	
	BA2	mmmm, 4	
	DSRCH	P1, P2, P3, P4, P5	

Where:

- n — the number of sectors to be read-in on a find condition.
- mmmm — the address (LSC) of a 4 character field in core memory which contains the absolute drum address of the sector at which the search is to start. (The search is terminated by: sector overflow (no-find), or successful find.)
- P1 — core memory address of the read-in area (MSL).
- P2 — length of search key:
  If search for EQUAL: 8, 16, or 32.
  If search for EQUAL or GREATER-than: 8G, 16G, 32G.
- P3 — the address of a subroutine to which a JR will be executed if the address (mmmm) is not within the bounds of an existing opened file. If control is returned, the search will be executed.
- P4 — address to which control will be transferred on a 'no-find' condition (sector overflow).
- P5 — Immediate Return Line (IRL).
  If this parameter is omitted, control will be retained by BDRUM until the order is completed.

4.7.1 Entrance requirements:

(1) the search key must be stored in the first 8, 16, or 32 locations of the read-in area (in core memory).

(2) the MSL of the read-in area must be a multiple of 64.

(3) the 4 LSC of AR 2 must contain the absolute drum address of the sector at which the search is to begin.

(4) the file must be open (see GIVE and/or DFIND).

4.7.2 Normal exit conditions:

(1) if the search was successful (find), the sector(s) is in core memory, beginning at the address specified by P1.

Note: if a successful find occurs on the search, but outside the address limits of the file being searched, it will be considered as a 'no-find'.

4.7.3 Example: (as the DSRCH directive might appear in a worker program).

	LABEL	OP'N	OPERANDS	COMMENTS
(1)		BA2	SKEY8, 8	transfer search key to
(2)		SA2	FREAD+7, 8	read-in area.
(3)		SC	XSECN, 6	if 'find', read 6 sectors
(4)		BA2	SADDR, 4	
(5)		DSRCH	FREAD, 8, ERR4, NOFND, IRL4	

Comments: (1), (2) an eight character search key is transferred to the first eight characters of the read-in area.

(3) XSECN is set to read 6 sectors, if a successful search occurs.

(4) the absolute drum address of the sector at which the search is to start is loaded in AR2.

(5) FREAD - the core memory read-in area.

8 - the length of the search key.

ERR4 - subroutine (if address is not valid).

NOFND - address to which control will be transferred if the search is unsuccessful. (no-find).

IRL4 - Immediate Return Line.

4.8.0 To position the head-bar:

LABEL	OP'N	OPERANDS
	BAl	mmmm, 4
	DHEAD	

Where: mmmm the address (LSC) of a 4 character field containing the absolute drum address for the positioning.

4.8.1 Entrance requirements:

(1) the 4 LSC of AR1 must contain the absolute drum address for the positioning. (the address returned by the DFIND or GIVE directive).

4.8.2 Normal exit conditions:

(1) the head-bar positioning order has been initiated; the worker program can continue processing, thus overlapping the head-movement time.

4.8.3 Example: (as the DHEAD directive might appear in a worker program).

	LABEL	OP'N	OPERANDS
(1)		BAl	DADDR, 4
(2)		DHEAD	

Comments: (1) drum address to AR1.
(2) initiate drum order.

4.9.0 To issue a user generated FASTRAND drum packet to the operating system:

LABEL	OPCN	OPERANDS
	SC	XSECN, n
	BAl	mmmm, 16
	DFUNC	P1, P2

Where:

mmmm — the address (LSC) of a 16 character field which contains the complete 16 character FASTRAND drum packet as required by the operating system. (see OPR document).

n — required only on multiple sector operations.

P1 — the address to which a JR will be executed if the address in mmmm is not within the bounds of an existing opened file. If control is returned, the packet will be issued as if valid.

If this parameter is omitted, the address validation will be bypassed.

P2 — Immediate Return Line.

If this parameter is omitted, control will be retained by BDRUM until the order is completed.

4.9.1 Entrance requirements:

(1) AR1 must contain the 16 character FASTRAND drum packet as required by the operating system.

(2) the FASTRAND drum handler (in OPR) must be initialized (JR XINFR), unless a GIVE or DFIND (any file) has been previously executed.

4.9.2 Normal exit conditions:

(1) the 16 character packet has been issued.(Refer to the OPR document).

4.9.3 Example: (as the DFUNC directive might appear in a worker program).

LABEL	OP'N	OPERANDS
(1)	SC	XSECN, 8
(2)	BAl	PACKT, 16
(3)	DFUNC	ERR6, IRL6

. Comments: (1) this is a multiple sector function (8sectors)

. (2) PACKT - the address (LSC) of a 16 character FASTRAND drum packet.

. (3) ERR6 - label of subroutine, if address is invalid.

. IRL6 - Immediate Return Line.