

**DOPSY LOADING PROCEDURE
MANUAL**

December, 1970

DOPSY LOADING PROCEDURE

Order of card decks:

- 1) DICL - Disc Initialization Card Loader
- 2) \$ARR (8, 12 or 16K system) - Automatic Restart Routine
- 3) Deck of Assign Cards
- 4) OPEN
- 5) NSOPEN
- 6) CLOSE
- 7) SWAPOUT
- 8) DFILEN
- 9) SRCH
- 10) GET
- 11) PUT
- 12) GETW
- 13) PUTW
- 14) READ
- 15) WRITE
- 16) ADRXLATE
- 17) SCAN
- 18) DEBUG
- 19) INREC
- 20) GFREC
- 21) CRIO
- 22) TTRIO
- 23) OUTREC
- 24) PFREC
- 25) LPIO

- 26) TTPIO
- 27) DISCIO
- 28) MTIO
- 29) BINDEC
- 30) BINOCT
- 31) OCTBIN
- 32) CRASC
- 33) MONMP
- 34) TVECT
- 35) \$JOB
- 36) \$SET
- 37) \$PATCH
- 38) \$NOTE
- 39) \$UPDATE
- 40) \$CREA2
- 41) \$DELETE
- 42) \$ASSIGN
- 43) \$EXEC
- 44) \$CREATE
- 45) \$MONMP
- 46) \$ASM
- 47) ASMI
- 48) EDIT
- 49) PCNVT
- 50) FDUMP
- 51) UTILITY
- 52) DUMP

- 53) MTAP
- 54) TAPLP
- 55) CRDTAP
- 56) LISTC
- 57) FCOMP
- 58) FCOMP1
- 59) DBUP
- 60) TWO CARD BOOT

OPERATOR FUNCTIONS

- A) Start with Deck 1 (DICL) in the card reader hopper.
- B) Depress ON and START on the card reader. Wait for ready condition.
- C) Check the disc 'write inhibit' switch and turn the switch off (down).
- D) Set 'SIC' switch up.
- E) Set the switch register to 01000100_8 and load the command register with the 'LDC' switch.
- F) Depress CPU RESET.
- G) Depress LOAD CDR (one card will be read).
- H) Set 'SIC' switch down.
- I) Depress CPU start. Cards should start loading.
- J) On first CPU halt, set console switches 2, 3, and 4 and depress CPU start.
- K) The teletype will output a core map of the following programs and then halt.
 - MAINPR, SWAPOU, DISCIO, DINT, BINDEC, ADRXLA,
WRITE, PUTW, CRIO, CRINT, TTPIO, TTPINT
- L) Set console switches 2 and 6 and depress CPU start. The disc will be erased (about 25 seconds) and CPU will stop.
- M) Set only console switch 2 and depress start. Cards will be read and then the teletype will respond with an asterisk (*) indicating the monitor is loaded.

N) Type the system job I.D.

```
// JOB '←←←←'
```

followed by a carriage return.

O) Optional: To verify that DICL is correct, type

```
// FDUMP DIRECTORY
```

The following list of names should be printed on the teleprinter together with the file types, the disc space assigned and the disc space used:

```
$DIRCT, $MONMP, $CREAT, $EXEC, $CREA2, $UPDAT,  
$PATCH, $SET, $DELET, $ASSIG, $FDUMP, $RENAM,  
$NOTE, $JOB, $DEBUG
```

P) If step O is correct or assumed to be okay, type

```
// SET CR
```

The card reader will read the control card for \$ARR and the teleprinter will type

```
PASSWORD
```

Q) The operator responds by depressing control, shift and N simultaneously on the teletype keyboard.

R) The card reader will start reading cards and after \$MONMP is created, instructions for loading the 2 card boot will be printed and the CPU will then halt.

S) The operator loads the 2 card boot by repeating steps D) through I) above.

The teletype will respond with the system monitor asterisk. DOPSY has been loaded.

Error recovery - none.

'Error in record sequence' when \$MONMP is created: Replace complete DOPSY deck with a new copy.

A teletype listing of a complete DOPSY load is attached.

```

01535
MAINPR N      00305
SWAPOU       00571
DISCIO       00655
DINT  N      00754
BINDEC       01042
ADRXLA       01075
WRITE        01164
PUTW         01201
CRIO         01232
CRINT  N     01305
TTPIO        01372
TTPINT N     01464

```

```
*// JOB '-----'
```

```
*// FDUMP DIRECTORY
```

NAME	TYPE	ASSIGNED	USED
\$DIRCT	D	001920	000090
\$MONMP	C	002160	002047
\$CREAT	C	000576	000477
\$EXEC	C	001152	001049
\$CREA2	C	000480	000329
\$UPDAT	C	000336	000215
\$PATCH	C	000432	000305
\$SET	C	000192	000085
\$DELET	C	000480	000366
\$ASSIG	C	000576	000440
\$FDUMP	C	000768	000664
\$RENAM	C	000432	000291
\$NOTE	C	000288	000157
\$JOB	C	000288	000158
\$DEBUG	C	000480	000340

```
*// SET CR
```

```
// NOTE 'DATE - NOVEMBER 24, 1970.'
```

```
DATE - NOVEMBER 24, 1970.
```

```

// CREATE 'SARR' COREIMAGE CR
PASSWORD
00510
    IS ENTRY PNT
// NOTE 'REBOOT THE SYSTEM' HALT
REBOOT THE SYSTEM
*// JOB '-----'
*// SET CR
// ASSIGN '$DIRCT' 80 SECTORS
// ASSIGN '$MONMP' 3000 WORDS
// ASSIGN '$ASM' 384 WORDS
// ASSIGN '$EDIT' 5150B WORDS
// ASSIGN '$ASM1' 7500B WORDS
// ASSIGN '$UTILITY' 192 WORDS
// ASSIGN '$DUMP' 700 WORDS
// ASSIGN '$FCOMP' 336 WORDS
// ASSIGN '$FCOMP1' 1920 WORDS
// ASSIGN '$PATCH' 550 WORDS
// CREATE 'OPEN' OBJ CR
// CREATE 'NSOPEN' OBJ CR
// CREATE 'CLOSE' OBJ CR
// CREATE 'SWAPOU' OBJ CR
// CREATE 'DFILEN' OBJ CR
// CREATE OBJ CR 'SRCH'
// CREATE OBJ 'GET' CR
// CREATE OBJ 'PUT' CR
// CREATE OBJ 'GETW' CR
// CREATE OBJ 'PUTW' CR
// CREATE 'READ' OBJ CR
// CREATE OBJ 'WRITE' CR
// CREATE 'ADRXLA' OBJ CR
// CREATE OBJ 'SCAN' CR
// CREATE 'DEBUG' OBJ CR
// CREATE 'INREC' OBJ CR
// CREATE 'GFREC' OBJ CR
// CREATE 'CRIO' OBJ CR
// CREATE 'TTRIO' OBJ CR
// CREATE 'OUTREC' OBJ CR
// CREATE 'PFREC' OBJ CR
// CREATE 'LPIO' OBJ CR
// CREATE 'TTPIO' OBJ CR
// CREATE OBJ CR 'DISCIO'
// CREATE 'MTIO' OBJ CR
// CREATE OBJ 'BINDEC' CR
// CREATE OBJ 'BINOCT' CR
// CREATE 'OCTBIN' OBJ CR
// CREATE OBJ 'CRASC' CR
// CREATE 'MONMP' OBJ CR
// CREATE 'TVECT' OBJ CR
// CREATE '$JOB' COREIMAGE CR 6100B MAP CLEAR
06310
$JOB N 06206
$JOB IS ENTRY PNT

```

```

// CREATE '$SET' COREIMAGE CR 6500B MAP CLEAR
06572
$SET N 06520
$SET IS ENTRY PNT
// CREATE '$PATCH' COREIMAGE CR 6000B
06735
MAINPR N 06177
// CREATE '$NOTE' COREIMAGE 6000B CR
06175
$NOTE N 06070
$NOTE IS ENTRY PNT
// CREATE '$RENAME' COREIMAGE CR 6000B MAP CLEAR
06317
RENAME N 06111
RENAME IS ENTRY PNT
// CREATE '$UPDAT' COREIMAGE CR 6016B
06265
$UPDAT N 06102
$UPDAT IS ENTRY PNT
// CREATE '$SCREA2' COREIMAGE CR 6016B MAP CLEAR
06453
$SCREA2 N 06152
$SCREA2 IS ENTRY PNT
// CREATE '$DELETE' COREIMAGE CR 6000B MAP CLEAR
06473
$DELET N 06000
$DELET IS ENTRY PNT
// CREATE '$ASSIGN' COREIMAGE CR 6000B MAP CLEAR
06610
$ASSIG N 06000
$ASSIG IS ENTRY PNT
// CREATE '$EXEC' COREIMAGE CR 6016B
07777
$EXEC N 06016
$EXEC IS ENTRY PNT
// CREATE '$CREAT' COREIMAGE CR 6000B MAP CLEAR
06727
$CREAT N 06000
$CREAT IS ENTRY PNT

```


// CREATE 'SMONMP' COREIMAGE 'MONMP' 550B MAP CLEAR CTRL XPND
UPDATE OF EXISTING FILE?(Y/N)

Y

LOAD TVECT
NOLOAD CPIO ASCBIN

05762

TVECT 0

MAINPR N 00652

EROREC 01114

INREC 01211

SCAN 01406

OUTREC 01547

DFILEN 01770

TTRIO 02065

TTRINT N 02200

TTPIO 02337

TTPINT N 02431

LPID 02561

LPINT N 02765

CRID 03076

CRINT N 03151

CRASC 03247

DISCID 03357

DINT N 03456

READ 03540

WRITE 03552

GET 03577

PUT 03645

GETW 03702

PUTW 03713

OPEN 03731

NSOPEN 03762

SRCH 04174

WRITDS 04260

ENTRFN 04271

CLOSE 04345

SWAPOU 04435

BINDEC 04500

MTIO 04524

TAPINT N 05002

GFREC 05367

OCTBIN 05471

BINOCT 05536

PFREC 05616

ADRXL 05675

// NOTE 'ASSEMBLER'

// CREATE 'SASM' COREIMAGE CR 11000B

11433

ASM N 11145

ASM IS ENTRY PNT

// CREATE '=ASMI' OBJ CR

```

// CREATE '$ASMI' COREIMAGE CTRL XPND
NOLOAD SRCH WRITDS OCTBIN
LOAD BINDEC CLOSE SWAPOUT DISCIO GETW READ WRITE ADRXLATE TPIO MTIO
LOAD LPIO CRASC GFREC CRIO TTRIO GET PUT INREC '=ASMI'
07455
=ASMI 0
BINDEC 00224
CLOSE 00261
SWAPOU 00351
DISCIO 00435
DINT N 00534
GETW 00616
READ 00626
WRITE 00640
ADRXLA 00663
TPIO 00771
TTPINT N 01063
MTIO 01136
TAPINT N 01414
LPIO 02037
LPINT N 02243
CRASC 02367
GFREC 02471
CRIO 02604
CRINT N 02657
TTRIO 02753
TTRINT N 03066
GET 03215
PUT 03263
INREC 03335
MAINPR N 07421
// CREATE '=EDIT' OBJ CR

```

// CREATE '\$EDIT' COREIMAGE '=EDIT' XPND CTRL
NOLOAD CPIO LPIO BINOC ASCBIN OCTBIN GETW PUTW
NOLOAD MAINPR TTPINT TTRINT DINT CRINT ENTRFN

05155
MAINPR 00460
OPEN 01111
INREC 01151
GET 01333
DFILEN 01400
SCAN 01472
OUTREC 01633
CLOSE 02061
TPIO 02165
TTPINT 02257
SRCH 02504
WRITDS 02570
ENTRFN 02601
NSOPEN 02652
GFREC 02731
CRIO 03044
CRINT 03117
CRASC 03215
TTRIO 03327
TTRINT 03442
MTIO 03560
TAPINT N 04036
READ 04404
PUT 04421
PFREC 04472
SWAPOU 04547
WRITE 04610
DISCIO 04651
DINT 04750
ADRLA 05041
BINDEC 05132

// CREATE 'PCNVT' COREIMAGE CR 5100B CTRL

NOLOAD CPIO ASCBIN

05552
MAINPR N 05105
TTRIO 05247
TTRINT N 05362
BINOC 05506

// CREATE '\$FDUMP' COREIMAGE 6000B CR

UPDATE OF EXISTING FILE?(Y/N)

Y
07274
FDUMP N 06372
FDUMP IS ENTRY PNT

// CREATE '\$UTILI' COREIMAGE 6100B MAP CLEAR CR

06133
MAINPR N 06105

// CREATE '\$DUMP' COREIMAGE CR 6400B

07620
MAINPR N 06400

```

// CREATE '$MTAP' COREIMAGE CR 6000B
06213
MAINPR N    06026
// CREATE '$DEBUG' COREIMAGE 'DEBUG' 16500B MAP CLEAR
UPDATE OF EXISTING FILE?(Y/N)
Y
17213
DEBUG N     16542
// CREATE '=TAPLP' OBJ CR
// CREATE 'TAPLP' COREIMAGE '=TAPLP' 1000B MAP CLEAR
02653
MAINPR N     01111
LPIO        01343
LPINT N     01547
MTIO        01642
TAPINT N    02120
TPIO        02507
TTPINT N    02601
// CREATE '=CRDTA' OBJ CR
// CREATE 'CRDTAP' COREIMAGE '=CRDTA' 1000B MAP CLEAR
02457
MAINPR N     01055
CRIO        01212
CRINT N     01265
CRASC       01363
MTIO        01446
TAPINT N    01724
TPIO        02313
TTPINT N    02405
// CREATE 'LISTC' COREIMAGE 6100B CR
07055
MAINPR N     06155
LPIO        06305
LPINT N     06511
CRIO        06622
CRINT N     06675
CRASC       06773
// CREATE 'FCOMP' COREIMAGE CR 6000B
06243
FCOMP N     06131
FCOMP      IS ENTRY PNT

```

```
// CREATE 'FCOMP1' COREIMAGE CR XPND CTRL
NOLOAD CRIO CRASC TTRIO OCTBIN PFREC
NOLOAD CPIO ASCBIN
```

```
03620
```

```
FCOMP1 N    00440
INREC      00667
TPIO       01061
TTPINT N   01153
BINDEC     01232
OUTREC     01275
GFREC      01531
MTIO       01626
TAPINT N   02104
BINOCT     02460
LPIO       02601
LPINT N    03005
GET        03111
PUT        03157
GETW       03214
READ       03224
WRITE      03236
ADRXLA     03261
SWAPOU     03353
DISCIO     03437
DINT N     03536
```

```
FCOMP1 IS ENTRY PNT
```

```
// CREATE 'DBUP' COREIMAGE CR 1000B
```

```
02015
```

```
MAINPR N   01124
DISCIO     01417
DINT N     01516
TPIO       01621
TTPINT N   01713
BINDEC     01772
```

```
// NOTE 'REBOOT SYSTEM TO LOAD NEW MONITOR.' HALT
```

```
*// JOB '-----'
```

```
*// FDUMP DIRECTORY ' /++'
```

NAME	TYPE	ASSIGNED	USED	JOB#
\$DIRCT	D	003840	000378	-----
\$MONMP	C	003024	002778	-----
\$CREAT	C	000576	000519	-----
\$EXEC	C	001152	001057	-----
\$CREA2	C	000480	000332	-----
\$UPDAT	C	000336	000215	-----
\$PATCH	C	000576	000525	-----
\$SET	C	000192	000106	-----
\$DELET	C	000480	000363	-----
\$ASSIG	C	000576	000440	-----
\$FDUMP	C	000768	000748	-----
\$RENAM	C	000432	000255	-----
\$NOTE	C	000288	000173	-----
\$JOB	C	000288	000184	-----
\$DEBUG	C	000480	000379	-----
\$ASM	C	000384	000331	-----
\$EDIT	C	002688	002573	-----
\$ASM1	C	003936	003789	-----
\$UTILI	C	000192	000075	-----
\$DUMP	C	000720	000704	-----

FCOMP	C	000336	000211	++++
FCOMP1	C	001920	001840	++++
OPEN	0	000048	000042	++++
NSOPEN	0	000096	000056	++++
CLOSE	0	000144	000105	++++
SWAPOU	0	000096	000061	++++
DFILEN	0	000096	000062	++++
SRCH	0	000192	000171	++++
GET	0	000096	000066	++++
PUT	0	000096	000059	++++
GETW	0	000048	000023	++++
PUTW	0	000048	000028	++++
READ	0	000048	000023	++++
WRITE	0	000048	000031	++++
ADRXLA	0	000144	000100	++++
SCAN	0	000192	000158	++++
DEBUG	0	000480	000435	++++
INREC	0	000240	000194	++++
GFREC	0	000144	000128	++++
CRIO	0	000144	000133	++++
TTRIO	0	000240	000238	++++
OUTREC	0	000288	000254	++++
PFREC	0	000096	000094	++++
LPIO	0	000336	000316	++++
TTPIO	0	000192	000163	++++
DISCIO	0	000192	000192	++++
MTIO	0	000528	000519	++++
BINDEC	0	000048	000038	++++
BINOCT	0	000096	000061	++++
OCTBIN	0	000096	000055	++++
CRASC	0	000096	000096	++++
MONMP	0	000384	000351	++++
TVECT	0	000192	000164	++++
=ASM1	0	002736	002705	++++
=EDIT	0	000624	000603	++++
PCNVT	C	000480	000346	++++
\$MTAP	C	000288	000187	++++
=TAPLP	0	000240	000225	++++
TAPLP	C	001104	000987	++++
=CRDTA	0	000192	000163	++++
CRDTAP	C	000960	000863	++++
LISTC	C	000672	000541	++++
DBUP	C	000672	000573	++++

WORKING STORAGE HAS 726768 WORDS

*