

UNITED STATES DEPARTMENT OF THE INTERIOR
GEOLOGICAL SURVEY

On-Line Operation of Disc Backup Utility Program !DSKUP

by

David V. Fitterman

18 October 1979

Open-File Report 79-1607

1. Introduction

This report describes program DSKUP which allows the on-line running of the HP (Hewlett-Packard*) disc backup utility !DSKUP. !DSKUP can be used to save, restore, and copy the contents of a disc platter. The restore function can be used to change operating systems. The usual means of changing operating systems involves loading a paper tape version of !DSKUP. This can be rather inconvenient if the operating system is changed frequently. DSKUP reads the binary absolute program !DSKUP from a File Manager file into memory, moves the program to its proper memory locations, and then executes the program. The method has been successfully used with the Hewlett-Packard RTE-II operating system, but should work equally well with RTE-IV.

The technique described in this report could be used to allow running other binary absolute programs on-line.

2. Program DSKUP

Program DSKUP is written in HP FORTRAN IV and uses File Manager routines to input a binary absolute file. The program opens a type 7 file called !DSKUP which contains a copy of binary absolute program !DSKUP. If the program cannot open the file, it terminates with a STOP 0001. This situation can result if file !DSKUP has been purged or has not been stored in a File Manager file. After satisfactorily opening the file, each

*Any tradenames and trademarks found in this report are used for descriptive purposes only and do not constitute endorsement by the U.S. Geological Survey.

record of the program is read into array IBUF. The length of this array (11190 words) is sufficient to hold the copy of !DSKUP supplied with RTE-II Revision 1913. This length may have to be changed if substantial changes are ever made in !DSKUP. The word following the last word of !DSKUP in IBUF is set to zero to serve as a flag to subroutine RELOC that all instructions have been moved to their proper locations.

3. Subroutine RELOC

Subroutine RELOC is written in HP assembly language, and makes use of the following HP-21MX Extended Instruction Group commands: CAX, CXA, ISX, LAX, LBX, LDX, MVW. Use of this routine on an older HP-2100 CPU will require some reprogramming to simulate these commands.

After the absolute address of IBUF is determined by system library routine .ENTR, the interrupt and memory protect hardware are deactivated with a call to system routine \$LIBR. RELOC then determines the select code associated with LU 1, the system terminal, by using the first word available (FWA) of the device reference table (DRT) and the equipment table (EQT). The select code is loaded into the S-register. Next the binary absolute records are relocated. The word count of the binary absolute records is checked to be sure it is non-zero. A zero word count is used to signal there are no more binary absolute records to relocate in memory. The word count is shifted from the left hand to the right hand character and stored in COUNT. The destined address of the binary absolute record is loaded and stored in DEST. The source address is computed and saved. The source address less the destination address is computed. If this number is less than zero, relocating the binary

absolute record will result in an over-write of unrelocated records. This condition results in a HLT 13B. The highest address used by !DSKUP is 41165B, so as long as the first word available of the disc resident background area exceeds this number the user should have no problems with over-writes. The absolute code is moved using the Move Word command.

When the last record has been relocated, the program exits from the relocating loop and simulates a PRESET instruction. This is done by issuing a clear control (CLC) and set flag (STF) instruction to the first twenty select codes beginning with S.C. 6B. The program begins execution at the location stored in START.

4. Program Loading

Program DSKUP is loaded as a permanent background program using the following File Manager commands:

```
:LG,2
:MR,%DSKUP
:MR,%RELOC
:RU,LOADR,99,6,9,10,2
```

The modules %DSKUP and %RELOC are the relocatable versions of the routines DSKUP and RELOC.

5. Inputting !DSKUP

The binary absolute tape version of !DSKUP must be stored in a File Manager file named !DSKUP. This is done by loading the paper tape into the photoreader and issuing the following command.

```
:ST,5,!DSKUP:sec:-2:7:-1,BA
```

The term "sec" stands for the file security code. It should be a positive number to prevent the file from being accidentally purged, while allowing the file to be read without knowledge of the code.

6. Operation

Program DSKUP is run using the command:

*RU,DSKUP

Since the operation of DSKUP turns off the operating system and is non-recoverable, all system activity should be terminated before running the program. It is preferable to exit from File Manager and issue the command at system level.

Once !DSKUP has been loaded, it will ask questions about the I/O channel configuration and the task to be performed. Refer to the Hewlett-Packard RTE Utility Programs Reference Manual (Part No. 92060-90017) for operation details. After !DSKUP has been run, the operating system must be restarted.

7. Listings

Included in this section are a compiler listing for DSKUP (Table 1), an assembler listing for RELOC (Table 2), and the loader map for DSKUP (Table 3).

Table 1 Compilation listing of program DSKUP.

PAGE 0001 FTH. 8:55 AM THU., 18 OCT., 1979

```

0001 FTH,L. PROGRAM DSKUP,3,80
0002 C
0003 C
0004 C PROGRAM TO RUN ABSOLUTE PROGRAM IDSKUP ON LINE.
0005 C PERFORMS THE FOLLOWING OPERATIONS:
0006 C
0007 C SAVE - DISC TO TAPE
0008 C RESTORE - TAPE TO DISC
0009 C COPY - DISC TO DISC
0010 C
0011 C THE TAPE FORMATS USED BY IDSKUP ARE COMPATIBLE WITH
0012 C ON-LINE RTE ROUTINES WHICH PERFORM THE SAVE FUNCTIONS.
0013 C
0014 C THIS PROGRAM READS A TYPE 2 (BINARY ABSOLUTE) FILE
0015 C NAMED IDSKUP WHICH IS THE ABSOLUTE VERSION OF UTILITY
0016 C PROGRAM IDSKUP. THIS IS STORED IN ARRAY IBUF.
0017 C THIS PROGRAM WILL EXECUTE A STOP: 0001 IF IT CAN NOT
0018 C OPEN FILE IDSKUP AND WILL EXECUTE A STOP: 0002 IF IT
0019 C CAN NOT READ A RECORD OF THE SAME FILE.
0020 C AFTER READING THE PROGRAM, ASSEMBLY ROUTINE RELOC IS USED
0021 C TO MOVE THE ABSOLUTE CODE TO ITS PROPER LOCATION. RELOC
0022 C CHECKS TO BE CERTAIN THAT IT IS NOT OVER WRITING ITSELF.
0023 C IF THIS CONDITION IS DETECTED, THE PROGRAM HALTS (HLT 13B).
0024 C
0025 C THE SELECT CODE OF THE SYSTEM CONSOLE IS DETERMINED BY
0026 C ROUTINE RELOC BY LOOKING AT LU 1'S ENTRY IN THE DRT.
0027 C
0028 C WRITTEN BY D. V. FITTERMAN
0029 C U.S. GEOLOGICAL SURVEY
0030 C BOX 25046, MAIL STOP 964
0031 C DENVER, COLORADO 80225
0032 C
0033 C WRITTEN JULY 1979
0034 C MODIFIED 25 JULY 1979
0035 C
0036 C DIMENSION IDCB(144),NAME(3),IBUF(11190)
0037 C EQUIVALENCE(NAME(1),IBUF(1))
0038 C DATA IPT/1,NAME/2HID,2HSH,2HUP/
0039 C
0040 C OPEN FILE
0041 C CALL OPEN(IDCB,IER,NAME)
0042 C IF(IER .LT. 0) STOP 1
0043 C
0044 C READ A RECORD
0045 C 10 CALL READF(IDCB,IER,IBUF(IPT),512,LEN)
0046 C IF(IER .LT. 0) STOP 2
0047 C IF(LEN .EQ. -1) GO TO 20
0048 C IPT=IPT+LEN-1
0049 C GO TO 10
0050 C 20 IBUF(IPT)=0
0051 C CALL RELOC(IBUF)
0052 C END

```

Table 1 -continued

PAGE 0002 DSKUP 8:55 AM THU., 18 OCT., 1979

FTN4 COMPILER: HP92060-16092 REV. 1913 (790206)

** NO WARNINGS ** NO ERRORS ** PROGRAM - 11400 COMMON - 00000

PAGE 0003 DSKUP 8:55 AM THU., 18 OCT., 1979

SYMBOL TABLE

| NAME | ADDRESS | USAGE | TYPE | LOCATION |
|-----------|---------|-----------------------------|---------|----------|
| Q10 | 26125R | STATEMENT NUMB | | |
| Q20 | 26162R | SUBPROGRAM | | EXTERNAL |
| CLRIO | 00001X | SUBPROGRAM | REAL | EXTERNAL |
| EXEC | 00000X | SUBPROGRAM | REAL | EXTERNAL |
| IBUF | 00000R | ARRAY(*) | INTEGER | LOCAL |
| IDCB | 00000R | ARRAY(*) | INTEGER | LOCAL |
| IJOB | 25000R | VARIABLE | INTEGER | LOCAL |
| IERR | 26176R | VARIABLE | INTEGER | LOCAL |
| IPT | 26175R | VARIABLE | INTEGER | LOCAL |
| LEN | 26203R | ARRAY(*) | INTEGER | LOCAL |
| NAME | 00000R | ARRAY(*) | INTEGER | LOCAL |
| OPEN | 00002X | SUBPROGRAM | REAL | EXTERNAL |
| READF | 00004X | SUBPROGRAM | REAL | EXTERNAL |
| RELOC | 00005X | SUBPROGRAM | REAL | EXTERNAL |
| PAGE 0004 | FTN. | 8:55 AM THU., 18 OCT., 1979 | | |

0053

END#

Table 2 Assembly listing of routine RELOC

PAGE 0001

ASMB,L,T,C

```
0001 RELOC R 0000001
0002 #LIBR X 0000001
0003 ENTER X 0000002
0004 TRUF R 0000000
0005 LOOP R 0000002
0006 PREST R 0000003
0007 R 0000000
0008 ENTH 0010000
0009 DET 0010002
0010 START R 0000002
0011 COUNT R 0000003
0012 DFTST R 0000004
0013 NPER R 0000005
0014 ** NO ERRORS PASS#1 **RTE ASMB 760924**
0015 PAGE 0002 #01
```

ASMB,L,T,C

ROUTINE RELOC

THIS ROUTINE MOVES BINARY ABSOLUTE CODE TO ITS PROPER MEMORY LOCATION. FIRST THE SELECT CODE ASSOCIATED WITH LU1, THE SYSTEM CONSOLE, IS DETERMINED AND LOADED INTO THE S-REGISTER. THE PROGRAM CODE IS IN BINARY ABSOLUTE RECORD FORM. THE CHECK SUM WORD IS NOT VERIFIED. THE LAST RECORD IS FOLLOWED BY A RECORD WITH A WORD COUNT OF ZERO. THE ROUTINE OF CHECKS FOR OVER WRITING UNMOVED SECTIONS OF CODE. IF THIS CONDITION WILL RESULT THE PROGRAM WILL HALT (HLT 13). NEXT A PRESET INSTRUCTION IS SIMULATED ON NPER PERIPHERALS STARTING WITH SELECT CODE 100. THE PROGRAM THEN JUMPS TO THE STARTING ADDRESS AND BEGINS EXECUTION.

THIS ROUTINE USES SEVERAL 21-MX INSTRUCTIONS WHICH WILL HAVE TO BE SIMULATED IF RUN ON A HP2100.

WRITTEN BY D. V. FITTERMAN
U.S. GEOLOGICAL SURVEY
BOX 25046, MAIL STOP 964
DENVER, COLORADO 80225

WRITTEN JULY 1979
MODIFIED 27 JULY 1979

```
0001
0002*
0003*
0004*
0005*
0006*
0007*
0008*
0009*
0010*
0011*
0012*
0013*
0014*
0015*
0016*
0017*
0018*
0019*
0020*
0021*
0022*
0023*
0024*
0025*
0026*
0027*
```


Table 2 -continued

| | | | |
|-------|-------|-------------------|--|
| 0028X | 00000 | NAM RELOC,7,80 | |
| 0029 | 00000 | ENT RELOC | |
| 0030 | 00000 | ENT \$LIBR, .ENTR | |
| 0031 | 00000 | BSS 1 | ADDRESS OF BUFFER TO RELOCATE |
| 0032 | 00000 | NOP | |
| 0033 | 00001 | JSB .ENTR | RESOLVE INDIRECT ADDRESSES |
| 0034 | 00002 | DEF IBUF | |
| 0035 | 00003 | JSB \$LIBR | TURN OF INTERRUPTS AND MEMORY PROTECT |
| 0036 | 00004 | NOP | |
| 0037 | 00005 | LDA DRT, I | LOAD FIRST WORD OF DRT (LU 1) |
| 0038 | 00006 | AND =B77 | MASK OFF EQT ENTRY NUMBER |
| 0039 | 00007 | ADA =D-1 | SUBTRACT ONE |
| 0040 | 00010 | MPY =D15 | MULTIPLY BY 15 |
| 0041 | 00011 | NOP | |
| 0042 | 00012 | ADA EQTA | ADD FWA EQT |
| 0043 | 00013 | ADA =D3 | ADDRESS OF EQT4 |
| 0044 | 00014 | LDA A, I | LOAD EQT4 |
| 0045 | 00015 | AND =B77 | MASK OFF SELECT CODE |
| 0046 | 00016 | OTA 1B | OUTPUT TO "S" REGISTER |
| 0047 | 00017 | LDX =B0 | CLEAR "X" REGISTER |
| 0048 | 00021 | LAX IBUF, I | LOAD WORD COUNT |
| 0049 | 00022 | SZA, RSS | ZERO? |
| 0050 | 00023 | JMP, PREST | YES, JUMP TO PRESET |
| 0051 | 00024 | ISX | INCREMENT "X" REGISTER |
| 0052 | 00025 | ALF, ALF | MOVE COUNT TO LOWER HALF OF "A" REGISTER |
| 0053 | 00026 | STA COUNT | SAVE WORD COUNT |
| 0054 | 00027 | STA COUNT | |
| 0055 | 00030 | STA COUNT | |
| 0056 | 00031 | STA COUNT | |
| 0057 | 00032 | STA COUNT | |
| 0058 | 00033 | STA COUNT | |
| 0059 | 00034 | STA COUNT | |
| 0060 | 00035 | STA COUNT | |
| 0061 | 00036 | STA COUNT | |
| 0062 | 00037 | STA COUNT | |
| 0063 | 00038 | STA COUNT | |
| 0064 | 00039 | STA COUNT | |
| 0065 | 00040 | STA COUNT | |
| 0066 | 00041 | STA COUNT | |
| 0067 | 00042 | STA COUNT | |
| 0068 | 00043 | STA COUNT | |
| 0069 | 00044 | STA COUNT | |
| 0070 | 00045 | STA COUNT | |
| 0071 | 00046 | STA COUNT | |
| 0072 | 00047 | STA COUNT | |
| 0073 | 00048 | STA COUNT | |
| 0074 | 00049 | STA COUNT | |
| 0075 | 00050 | STA COUNT | |
| 0076 | 00051 | STA COUNT | |
| 0077 | 00052 | STA COUNT | |
| 0078 | 00053 | STA COUNT | |
| 0079 | 00054 | STA COUNT | |
| 0080 | 00055 | STA COUNT | |
| 0081 | 00056 | STA COUNT | |
| 0082 | 00057 | STA COUNT | |
| 0083 | 00058 | STA COUNT | |
| 0084 | 00059 | STA COUNT | |
| 0085 | 00060 | STA COUNT | |
| 0086 | 00061 | STA COUNT | |
| 0087 | 00062 | STA COUNT | |
| 0088 | 00063 | STA COUNT | |
| 0089 | 00064 | STA COUNT | |
| 0090 | 00065 | STA COUNT | |
| 0091 | 00066 | STA COUNT | |
| 0092 | 00067 | STA COUNT | |
| 0093 | 00068 | STA COUNT | |
| 0094 | 00069 | STA COUNT | |
| 0095 | 00070 | STA COUNT | |
| 0096 | 00071 | STA COUNT | |
| 0097 | 00072 | STA COUNT | |
| 0098 | 00073 | STA COUNT | |
| 0099 | 00074 | STA COUNT | |
| 0100 | 00075 | STA COUNT | |

Table 2 -continued

| | | | | | | |
|------|----------------------|---------|--|--|--|---|
| 0005 | 00047 | 101744 | | | | "X" TO "A" |
| 0006 | 00050 | 042063R | | | | UPDATE "X" POINTER |
| 0007 | 00051 | 101741 | | | | "A" TO "X" |
| 0008 | 00052 | 000022R | | | | |
| 0009 | 00053 | 100706 | | | | SIMULATE PRESET TO PERIPHERALS READY |
| 0010 | 00054 | 102106 | | | | |
| 0011 | 00055 | 000053R | | | | INCREMENT SELECT CODES |
| 0012 | 00056 | 000054R | | | | DONE ENOUGH CHANNELS? |
| 0013 | 00057 | 000065R | | | | |
| 0014 | 00058 | 000053R | | | | START EXECUTION OF ABSOLUTE PROGRAM |
| 0015 | 00059 | 120062R | | | | |
| 0016 | 00060 | | | | | |
| 0017 | 01050 | | | | | FWA EQUIPMENT TABLE |
| 0018 | 01052 | | | | | FWA DEVICE REFERENCE TABLE |
| 0019 | 00062 | | | | | ADDRESS OF FIRST LOCATION TO BE EXECUTED |
| 0020 | 00063 | 000002 | | | | TEMPORARY STORAGE FOR WORD COUNT |
| 0021 | 00064 | 000000 | | | | TEMPORARY STORAGE FOR DESTINATION ADDRESS |
| 0022 | 00065 | 177254 | | | | NUMBER OF PERIPHERAL CHANNELS TO PREEST |
| | 00066 | 000077 | | | | |
| | 00067 | 177277 | | | | |
| | 00070 | 000017 | | | | |
| | 00071 | 000003 | | | | |
| | 00072 | 000000 | | | | |
| 0033 | NO ERRORS *TOTAL | | | | | |
| | END RELOC | | | | | |
| | ** RTE ASMB 760924** | | | | | |

Table 2 -continued

PAGE 0004

RELOC
CROSS-REFERENCE SYMBOL TABLE

| | | | | |
|-------|-------|-------|-------|-------------|
| #LIHR | 00031 | 00036 | | |
| ENTR | 00031 | 00034 | | |
| "D0 | | 00047 | | |
| "377 | | 00039 | 00045 | |
| "D-1 | | 00040 | | |
| "D15 | | 00041 | | |
| "D3 | | 00043 | | |
| A | 00076 | 00044 | 00060 | |
| COUNT | 00080 | 00053 | 00064 | 00066 E |
| DEST | 00081 | 00055 | 00063 | |
| DRT | 00078 | 00038 | | |
| EOTA | 00077 | 00042 | | |
| IBUF | 00032 | 00035 | 00048 | 00054 00058 |
| LOOP | 00048 | 00068 | | |
| NPER | 00082 | 00073 | | |
| PRESI | 00069 | 00050 | 00071 | 00072 00074 |
| RELOC | 00033 | 00030 | 00083 | |
| START | 00079 | 00075 | | |

Table 3 Loader map

| | | | |
|--------|-------|-------|--------------------|
| DSKUP | 46002 | 74211 | |
| RELOC | 74212 | 74304 | |
| OPEN | 74305 | 74472 | 02002-16006 741205 |
| READF | 74473 | 75231 | 02002-16006 770801 |
| RMPAR | 75232 | 75274 | 781106 24998-16001 |
| R/U# | 75275 | 75430 | 02002-16006 740801 |
| P-PAS | 75431 | 75457 | 02002-16006 740801 |
| #OPEN | 75460 | 75666 | 02002-16006 740801 |
| RU#UB | 75667 | 76140 | 02002-16006 750422 |
| RUMD# | 76141 | 76263 | 02002-16006 771121 |
| REUSE | 76264 | 76364 | 771122 24998-16001 |
| CLRIO | 76365 | 76373 | 750701 24998-16001 |
| PRU.E | 76374 | 76374 | 750701 24998-16001 |
| PRHME | 76375 | 76442 | 771121 24998-16001 |
| REIO | 76443 | 76547 | 02001-16005 780212 |
| .DFFER | 76550 | 76621 | 750701 24998-16001 |
| CLOSE | 76622 | 76740 | 02002-16006 771115 |