HP 3000 SERIES II
COMPUTER SYSTEM
MANUAL OF STAND-ALONE DIAGNOSTICS

STAND-ALONE
DIAGNOSTIC UTILITY PROGRAM II
(SDUPII)

Diagnostic D417

03000-90125
Printed in U.S.A.

3/77

NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another program language without the prior written consent of Hewlett-Packard Company.

Copyright © 197?-7 by HEWLETT-PACKARD COMPANY

TABLE OF CONTENTS

PRINTING HISTORY

New editions incorporate all update material since the previous edition. Update packages, which are issued between editions, contain additional and replacement pages to be merged into the manual by the customer. The date on the title page and back cover changes only when a new edition is published. If minor corrections and updates are incorporated, the manual is reprinted but neither the date on the title page and back cover nor the edition change.

First Edition 10/76
Second Edition 3/77

The second edition documents the ASCII and BINARY procedures.

1.0 INTRODUCTION

The purpose of the SDUPII diagnostic utility program is to build an I/O stand-alone diagnostic tape, a stand-alone CPU diagnostic tape, and to enable stand-alone diagnostics to be loaded from a magnetic tape unit. The SDUPII diagnostic utility program is used with an HP 3000 Series II Computer System. All diagnostic and on-line verification programs for Series II Systems are in the group HP32230.SUPPORT.

2.0 MINI OPERATING INSTRUCTION

2.1 I/O Stand-Alone - Type 2

```
:RUN SDUPII

3000 DIAGNOSTIC UTILITY PROGRAM(SDUP)D417A.01.01

DO YOU WANT INSTRUCTIONS?
ANSWER 'YES' OR 'NO'

NO
INPUT DIAGNOSTIC TYPE
2
PROGRAM NAME?
PD411A
PROGRAM NAME?
/
INPUT DRT OF THE LINE PRINTER
A CARRIAGE RETURN ASSUMES NO LINE PRINTER
%12
INPUT LINE PRINTER TYPE
MODELS-2607A,2613A,2617A,AND 2618A ARE TYPE Ø
MODELS 2610A, AND 2614A ARE TYPE 1
A CARRIAGE RETURN ASSUMES TYPE = Ø
1
MOUNT TAPE ON TAPE UNIT

TAPE REQUEST HAS BEEN ISSUED
OPERATOR MUST NOW REPLY TO REQUEST

Ø1 PD411A

LOGICAL SEG PHYSICAL SEG  PB
    ØØØØØ1  ØØØØØ3  Ø1ØØØØ
    ØØØØØØ  ØØØØØ4  Ø3Ø441
    ØØØØØ2  ØØØØØ5  Ø31146
    ØØØØØ3  ØØØØØ6  Ø51457

END OF PROGRAM
```

2.2 CPU Stand-Alone - Type 1

```
:RUN SDUPII

3000 DIAGNOSTIC UTILITY PROGRAM(SDUP)D417A.01.01

DO YOU WANT INSTRUCTIONS?
ANSWER 'YES' OR 'NO'

NO
INPUT DIAGNOSTIC TYPE
1
PROGRAM NAME?
PØ
PROGRAM NAME?
/
MOUNT TAPE ON TAPE UNIT

TAPE REQUEST HAS BEEN ISSUED
OPERATOR MUST NOW REPLY TO REQUEST

Ø1 PØ
END OF PROGRAM
```

2.3 Cold Loading an I/O Stand-Alone Diagnostic

Switch Register - %3006
Press ENABLE and LOAD
Switch Register = Program Number
Press RUN
Execution begins

2.4 Cold Loading a CPU Stand-Alone Diagnostic

Switch Register = %3006
Press ENABLE and LOAD
Press RUN
Execution begins

3.0 REQUIREMENTS

3.1 Hardware

This diagnostic utility program requires an HP 3000 Series II Computer System with a minimum memory of 64K. To run the stand-alone diagnostics, one magnetic tape drive is required. A line printer is optional.

3.2 Software

The utility program will run under MPE II. The relocating library (RL) is a code segment which becomes resident when a stand-alone diagnostic tape is cold-loaded.

3.3 Prerequisite to Running the Diagnostic Utility

A stand-alone diagnostic program library must be created under MPE II. All programs must be in program file format. Each file is identified by a file name (up to eight alphanumeric characters beginning with an alpha character).

3.4 Preparing a Diagnostic for SDUPII

A. A User Subprogram Library (USL) file must be prepared using the Relocating Library in the group HP32230.SUPPORT

   Example: :PREP UD321B, PD321B; PMAP; RL=RL.HP32230.SUPPORT

B. The RL will contain the following procedures:

   1. PUT
   2. PUT FAST
   3. GET
   4. HELP
   5. BINARY
   6. ASCII

4.0 DETAILED OPERATING INSTRUCTIONS

4.1 Making an I/O Stand-Alone Tape (Type 2)

4.1.1 Log on under "FIELD.SUPPORT,HP32230"

4.1.2 Run the Diagnostic Utility Program by typing "RUN SDUPII" 'CR' (CR= Carriage Return).

4.1.3 The terminal will print the title message and ask if you want instructions. (Refer to paragraph 4.6.1c for a list of the instructions.)

Example: 3000 DIAGNOSTIC UTILITY PROGRAM(SDUP)D417A.01.01

DO YOU WANT INSTRUCTIONS?

ANSWER 'YES' OR 'NO'

4.1.4 Reply with either a "YES" or "NO".

4.1.5 The terminal will now request the diagnostic type:

1 = CPU Diagnostic
2 = I/O Stand-Alone

Example: INPUT DIAGNOSTIC TYPE

4.1.6 Type a "2" (I/O) and terminal will request the name of the program file you wish to use.

Example: PROGRAM NAME?

4.1.7 Reply with the Program File Name. If you want to use the preconfiguration option, then type ";C" after the program file name (refer to paragraph 4-7).

Example: "PD422A;C" CR
"PD427A.PUB.SYS;C" CR
"PD422A.PUB.SYS" CR

4.1.8 Step 4.1.6 will be repeated for each program file needed. To terminate this phase type "/".

4.1.9 The terminal will request the line printer DRT (Device Number).

Example: INPUT DRT OF THE LINE PRINTER
A CARRIAGE RETURN ASSUMES NO LINE PRINTER

4.1.10 Reply with the DRT of the Line Printer. An "%" before the number will denote an octal number; anything else will be considered decimal.

Example: "10" CR
"%12" CR
"7" CR
"Ø" CR (No Line Printer)

4.1.11 The terminal will request the type of line printer to be used.
Type 0 is used for Models 2607A, 2613A, 2617A and 2618A.
Type 1 is used for Models 2610A and 2614A.

A carriage return assumes type=0

Example: INPUT LINE PRINTER TYPE
MODELS 2607A, 2613A, 2617A AND 2618A ARE Type Ø
MODELS 2610A AND 2614A ARE Type 1

4.1.12 The terminal will print

MOUNT TAPE ON TAPE UNIT

TAPE REQUEST HAS BEEN ISSUED

OPERATOR MUST NOW REPLY TO REQUEST

A tape request is issued to the System Console with the name of the tape file being "SDUP". The operator must mount the tape and reply to the request.

4.1.13 As the tape is being written, the terminal will print a map of the program files. This map will contain:

1. Program number and name.
2. Logical Segment Number.
   This is the segment number printed in a PMAP from the segmenter on the program listing.
3. Physical Segment Number.
   This is the Physical Segment Number SDUPII assigns to the Logical Segment while loading.
4. PB of Segment.
   This is the octal address when this segment starts.

This map will be repeated for each program file written.

4.1.14 Terminal will print "END OF PROGRAM".

4.2 Making a CPU Stand-Alone Tape (Type 1)

4.2.1 Log on under "FIELD.SUPPORT,HP32220"

4.2.2 Run the Diagnostic Utility Program by Typing "RUN SDUPII" 'CR' (CR = Carriage Return).

4.2.3 The terminal will print the title message and ask if you want instructions. (Refer to paragraph 4.6.1c for a list of instructions.)

Example:
```
3000 DIAGNOSTIC UTILITY PROGRAM (SDUP)D417A.01.01
DO YOU WANT INSTRUCTIOSN?
ANSWER 'YES' OR 'NO'
```

4.2.4 Reply with either a "YES" or "NO".

4.2.5 The terminal will now request the diagnostic type:

1 = CPU Diagnostics
2 = I/O Stand-Alone

Example: INPUT DIAGNOSTIC TYPE

4.2.6 Type in a 1 (CPU) and terminal will request the name of the program file you wish to use.

Example: PROGRAM NAME?

4.2.7 Reply with the Program File Name. If you want to use the preconfiguration option, then type ";C" after the program file name(refer to paragraph 4-7).

Examples:
```
"PD422A;C" CR
"PD427A.PUB.SYS;C" CR
"PD422A.PUB.SYS"  CR
```

4.2.8 Step 4.2.6 will be repeated for each program file needed. To terminate this phase, type "/".

4.2.9 The terminal will print

```
MOUNT TAPE ON TAPE UNIT
TAPE REQUEST HAS BEEN ISSUED
OPERATOR MUST NOW REPLY TO REQUEST
```

A tape request is issued to the System Console with the name of the tape file being "SDUP". The operator must mount the tape and reply to the request.

4.2.10 As the tape is being written, the terminal will print a map containing the program number and name of the program files.

4.2.11 Terminal will print "END OF PROGRAM".

4.3 Procedure for Cold-Loading an I/O Stand-Alone Diagnostic

4.3.1 Select Tape Unit 0 and mount the general stand-alone diagnostic tape as illustrated on the front panel of the tape unit.

4.3.2 Place other tape units off-line (by pressing the OFF button), or set them to any unit number but 0.

4.3.3 Press the RESET button, then press the LOAD button on the tape unit. The tape winds forward until the load point is reached, and stops. The LOAD light on the tape unit comes on.

4.3.4 Press the ON LINE button on the tape unit.

4.3.5 Set the Control Panel Switch Register to %3006.

4.3.6 Press ENABLE and LOAD simultaneously. The tape will Cold-Load the Stand-Alone Loader and Halt. (The CIR is unchanged). If the Cold-Load fails, the system halt light may be lit.

4.3.7 Set the Switch Register to the diagnostic number for the appropriate diagnostic. This was the number printed in the map in paragraph 4.2.10.

Press RUN.

The loader will search and load the appropriate diagnostic on the tape and transfer control to the beginning of the diagnostic.

4.3.8 Press the RETURN key on the console to speed-sense the terminal.

4.4 Procedure for Cold-Loading a CPU Diagnostic

4.4.1 Select Tape Unit 0 and mount the CPU stand-alone diagnostic tape as illustrated on the front panel of the tape unit.

4.4.2 Place other tape units off-line (by pressing OFF button), or set them to any unit number but 0.

4.4.3 Press the RESET button, then press the LOAD button on the tape unit. The tape winds forward until the load point is reached, and stops. The LOAD light on the tape unit comes on.

4.4.4 Press the ON LINE button on the tape unit.

4.4.5 Set the Control Panel Switch Register to %3006.

4.4.6 Press ENABLE and LOAD simultaneously.

4.4.7 Press RUN

Control will be transferred to the diagnostic.

4.4.8 The CPU diagnostics are written on the tape, each with their own loader. To load the fifth CPU diagnostic on the tape you will have to perform five cold loads (4.4) to load the diagnostic you want.

4.5 Coded Halts - Stand-Alone Section

4.5.1 I/O Stand-Alone Diagnostics

Halt	Status	Meaning
%1	%100004	Program number in the Switch Register is greater than the maximum program on the tape. Place correct number in Switch Register and push RUN.
%2	%100004	SIO program failed to complete for tape index record or CST record. Fatal error. Must re-cold load.
%3	%100004	IXIT failed to launch diagnostic. Fatal error. Must re-cold load.
%5	%100004	TIO failed to the mag tape. Condition code C7. Push RUN to try again. TOS has mag tape status.
%6	%100004	Time out for SIO or status on mag tape. Push RUN to try again.
%7	%100004	SIO to mag tape failed condition code C7. Returned fatal error. Must re-cold load.
%11	%100004	CIO to mag tape failed. Condition code C7 returned. Fatal error. Must re-cold load.
%12	%100004	Tape error status return from mag tape. Fatal error. Must re-cold laod.
%1	%100001	CPU unexpected trap RA = Trap # X = Instruction that caused trap. May be fatal error.
%0	Unknown	I/O failure in PUTFAST. Push RUN to try again.
%14	Unknown	Line printer not ready. Push RUN to try again.

4.5.2 Coded Halts - CPU Diagnostics

Halt	Status	Meaning
%1	%100001	Unexpected CPU Trap. RA = Trap # May be a fatal error.
%2	%100002	Unexpected external interrupt trap. May be a fatal error.

4.6 Messages and Meaning

4.6.1

	Message	Meaning
a.	3000 DIAGNOSTIC UTILITY PROGRAM (SDUP)D417A.01.01	Title message.
b.	DO YOU WANT INSTRUCTIONS?	Request for instructions.
c.	THE DIAGNOSTIC UTILITY PROGRAM (SDUP) IS A PROGRAM THAT CONVERTS DIAGNOSTIC PROGRAM FILES TO A COLD LOADABLE TAPE. THIS TAPE WILL BE USED TO COLD LOAD STAND-ALONE DIAGNOSTICS. THERE ARE 2 TYPES OF STAND-ALONE DIAGNOSTICS: TYPE 1 IS A CPU DIAGNOSTIC WHICH HAS ONLY ONE SEGMENT AND USES A SIMPLIFIED LOADER. TYPE 2 IS A STAND-ALONE DIAGNOSTIC WHICH MAY HAVE MORE THAN ONE SEGMENT AND WILL BE LOADED WITH A RELOCATING LOADER. THIS TYPE DIAGNOSTIC MAY ALSO HAVE DIALOGUE TO THE SYSTEM CONSOLE. THE PROGRAM WILL REQUEST DIAGNOSTIC TYPE. REPLY '1' FOR CPU OR '2' FOR STAND-ALONE. THE PROGRAM WILL REQUEST THE NAME OF THE PROGRAM FILE. REPLY WITH THE NAME OF THE PROGRAM FILE. IF YOU WANT TO USE THE PRECONFIGURATION OPTIONS, TYPE A ';C' AFTER THE PROGRAM NAME. A '/' TERMINATES THE PROGRAM INPUT PHASE.	Instructions.
d.	TYPE ANY CHARACTER TO CONTINUE	A request to continue.
e.	INPUT DIAGNOSTIC TYPE	Request for type of diagnostic 1) for CPU 2) for I/O Stand-Alone
f.	TYPE NOT 1 OR 2 --- INPUT ERROR TRY AGAIN	Type entered not 1 or 2.
g.	PROGRAM NAME?	Request for the name of the program file to be written to tape.
h.	FOPEN ERROR ON PROGRAM FILE	A FOPEN to the program file requested returned condition code <>. Non-fatal error.

4.6.1 (con't)

	Message	Meaning
i.	FREAD ERROR ON PROGRAM FILE	A FREAD to the program file requested returned condition code <>0. Non-fatal error.
j.	CPU TYPE DIAGNOSTIC HAS GLOBAL AREA EXCEEDING %6000 WORDS	CPU type diagnostic has too big a global area declared.
k.	UNRESOLVED EXTERNALS IN PROGRAM FILE	Check listing for unresolved externals. Program cannot be written to tape.
l.	LEVEL OF CHECKING <> 0 - BAD PROGRAM FILE	SDUP II does not allow procedures declared with level of checking <>0. Revise program.
m.	CPU TYPE DIAGNOSTIC HAS MORE THAN 1 SEGMENT	Only 1 segment programs are allowed as CPU type.
n.	NO EXTERNALS FOUND	Bad program file. Revise program.
o.	CPU TYPE DIAGNOSTIC HAS UNRESOLVED EXTERNALS	No unresolved externals are allowed in a CPU diagnostic. Check listing and revise program.
p.	FATAL TAPE WRITE ERROR	A fatal FREAD error has occurred on the mag tape. SDUPII will abort.
q.	FATAL FREAD ERROR ON PROGRAM FILE	A fatal FREAD error has occurred. SDUPII will abort.
r.	FATAL DIRECTORY ERROR	This occurs when the directory routine in SDUPII cannot find the information it needs to write a tape. SDUPII will abort.
s.	FOPEN ERROR ON MAG TAPE	FOPEN returned condition code <>. Fatal error. SDUPII will abort.
t.	FATAL FCONTROL ERROR OR TAPE UNIT	FCONTROL failed to tape unit. SDUP II will abort.
u.	MORE THAN ONE REFERENCE TO PUTADDR FOUND IN PROGRAM FILE FATAL ERROR	SDUPII only allows one reference to "PUTADDR" and that is in the RL. Possible bad RL. SDUPII will abort.
v.	TOO MANY SEGMENTS IN PROGRAM FILE REVISE PROGRAM -- MAX # OF SEGMENTS = %12	Number of segments in program file exceeds %12.

4.6.1 (con't)

	Message	Meaning
w.	TOTAL LENGTH OF PROGRAM +DATA EXCEEDS %177000 WORDS **FATAL ERROR REVISE PROGRAM	Program too long. Fatal error. SDUPII will abort.
x.	TYPE NOT Ø or 1 --- INPUT ERROR TRY AGAIN	Line printer type entered not a Ø or a 1. Enter correct value.
y.	INPUT ERROR - TRY AGAIN	Error on input. Re-enter information.
z.	DB LOCATION REQUESTED EXCEEDS 1023- TRY AGAIN	In preconfiguration, a DB location was requested that was greater than 1023. Re-enter correct value.
aa.	PRECONFIGURATION CHANGES FOR PROGRAM FILE XXXXX	Preconfiguration message for file named XXXX.
ab.	DB LOCATION TO BE CHANGED?	Request for the DB location to be modified.
ac.	DB LOCATION XXX = YYYYYY XXX	XXX = DB location. YYYYYY is the present contents of that location.
ad.	NEW VALUE =	Request for new value to be stored in DB location.
ae.	WARNING -- NO CALL TO PUT FOUND IN PROGRAM FILE XXX	SDUPII could not find a call to put. Non-fatal warning.
af.	TOO MANY CHARACTERS INPUT - TRY AGAIN	Number of characters exceeded 26.
ag.	INPUT DRT OF THE LINE PRINTER A CARRIAGE RETURN ASSUMES NO LINE PRINTER	Request for line printer DRT (Device Number). A "%" before the number denotes octal.
ah.	INPUT LINE PRINTER-TYPE MODELS-2607A, 2613A, 2617A, AND 2618A ARE TYPE Ø MODELS 2610A, AND 2614A ARE TYPE 1	Request the line printer. Type - Reply "Ø", or "1", or CR.
ai.	A CARRIAGE RETURN ASSUMES TYPE = Ø	
aj.	MOUNT TAPE ON TAPE UNIT	Reply to tape.
ak.	TAPE REQUEST HAS BEEN ISSUED OPERATOR MUST NOW REPLY TO REQUEST	Request to write tape.

4.7 Procedure for Preconfiguring a Diagnostic

4.7.1 Preconfiguration allows the user to modify locations in the DB area of a program. Each diagnostic will have DB locations for the Switch Register options, Section Select Register options, DRT of the device to be tested, etc. The user, in order to save time in running diagnostics, may wish to customize his cold-load tape to his system.

4.7.2 When SDUPII requests the "PROGRAM NAME" type in a ";C" after the file name if you wish to preconfigure this program.

Example: PROGRAM NAME?
PD420A;C

4.7.3 Follow the regular procedure for running SDUPII.

4.7.4 After the operator has replied to the tape request in Paragraph 4.1.12 and 4.2.9, SDUPII will start to write the tape. When it gets to a program where preconfiguration has been requested the terminal will print

"PRECONFIGURATION CHANGES FOR PROGRAM FILE XXXX"
"DB LOCATION TO BE CHANGED?"

4.7.5 Reply with the DB location you wish to modify. SDUPII assumes decimal, unless the number is preceded by a "%"; then it will accept the number as octal. Refer to each diagnostic manual for specific preconfiguration information.

A carriage return (CR) will terminate preconfiguration for this program file.

4.7.6 Terminal will print

"DB LOCATION YYY = ZZZZZZ"
"NEW VALUE ="

YYY is the DB location requested.
ZZZZZZ is the contents of the DB location requested.

4.7.7 Reply with value to be entered into DB. A "%" sign preceding the number SDUPII denotes it as octal, otherwise decimal.

4.7.8 The terminal will then repeat Section 4.7.6. A carriage return terminates preconfiguration.

5.0 GENERAL INFORMATION

5.1 Functional and Operational Characteristics

The SDUPII program consists of three major components. (1) the diagnostic tape preparation which prepares stand-alone diagnostic tapes, (2) the relocating loader which loads stand-alone diagnostics, and (3) the RL module which performs input and output to a console device and/or a line printer.

5.2 Create an I/O Stand-Alone Diagnostic Tape

This operation requires one magnetic tape drive. The program requests the user to supply the line printer DRT number and a name for the diagnostic program. The user also specifies whether preconfiguration is required. The diagnostic is then picked up from the diagnostic library and checked to see that it is a legal diagnostic. The program will request preconfiguration changes, if required, and print a segment map. The diagnostic is then written onto the magnetic tape as follows:

a. A 32 word cold-load record.
b. A 256 word cold-load record.
c. Two 1K records containing the stand-alone loader, tables, and halts.
d. A file mark.
e. For each program a %23 word program index.
f. For each program a %64 word CST record.
g. For each program as many 1K records as necessary for the code and global data
h. A file mark.

The tape is rewound.

5.3 Relocating Loader

The relocating loader (also called the stand-alone loader) is placed on the stand-alone UNIT diagnostic tape by the diagnostic utility. It is loaded into memory by the 32 word cold-load program. The user enters the program ID in the Switch Register (obtained from the map printed) and presses RUN. The program is loaded with an origin of %10000, the tape is rewound, and the diagnostic is given control at the primary entry point.

5.4 Communication Procedures

The purpose of the I/O communication procedures is to perform the printing of messages and input of data required by several stand-alone UNIT diagnostics. The procedures are appended to the diagnostic at preparation time and are loaded with the relocating loader. The I/O Communication Procedure GET will SPEEDSENSE. A call to the procedure GET (BUFFER, Ø, K) to determine baud rate must precede any input/output. The response to this call to GET is a carriage return.

5.5 Procedure GET (Buffer, Maxcount, K);

```
VALUE MAXOUNT:
INTEGER MAXOUNT, K;
BYTE ARRAY BUFFER;
OPTION EXTERNAL;
```

Outputs an X-ON to start the tape reader, then receives and stores data in the buffer until the operator sends a CR. Program responds with an LF (line feed).

The following characters are ignored during input:

Code	
%00	NULL, CONTROL @
%12	line feed, CONTROL J
%23	DC3, X-OFF, CONTROL S
%31	CONTROL Y
%177	rubout, DEL

A CONTROL X results in the last line being deleted; this is acknowledged by "!!!" CRLF.

A CONTROL H results in the last character being deleted; this is acknowledged with "/".

On return, K contains a count of the characters received.

5.6 Procedure PUT (Message, Count, Type);

```
VALUE COUNT, TYPE;
INTEGER COUNT, TYPE;
BYTE ARRAY MESSAGE;
OPTION EXTERNAL;
```

Writes a line with optional CRLF. User supplies buffer pointer, byte count, and type parameter for CRLF. If TYPE = Ø, the program will append a CRLF. An X-OFF procedes all carriage returns.

5.7 Procedure PUTFAST (Message, Count, Type);

```
VALUE COUNT, TYPE:
INTEGER COUNT, TYPE;
BYTE ARRAY MESSAGE;
OPTION EXTERNAL;
```

User supplies buffer pointer, byte count, and a type parameter for format control. If count is not Ø the message string is put in the line printer's buffer and if TYPE = Ø then the line printer's buffer is printed using a single space format. If the DRT for the line printer is Ø, PUTFAST will call PUT instead of using the line printer.

5.8 Integer Procedure numchar:=ASCII(word,base,string);

```
VALUE WORD, BASE;
LOGICAL WORD;
INTEGER BASE;
BYTE ARRAY STRING;
OPTION EXTERNAL;
```

Converts a 16-bit binary number (word) to a different base and represents it as a numberic ASCII string. The procedure returns the number of characters contained in the string to numchar.

```
base =  8    Octal conversion
base = 10    Decimal conversion (left justified)
base =-10    Decimal conversion (right justified)
```

Refer to the MPE Intrinsics Manual for more information about this procedure.

5.9 Logical Procedure bineqv:=BINARY(string,length);

```
VALUE LENGTH
BYTE ARRAY STRING;
INTEGER LENGTH;
OPTIONAL EXTERNAL;
```

Converts an ASCII string to its equivalent binary value and returns the value in bineqv.

If the character string is preceded by %, the string is treated as an octal value; if by +, -, or a number, it is treated as a decimal value.

If length = 0, the procedure returns 0 to the calling program.

Refer to the MPE Intrinsics Manual for more information about this procedure.

APPENDIX A

TABLES

A.1 CPU DIAGNOSTIC

INITIALIZATION FOLLOWING COLD LOAD

Absolute Memory Address (Octal)	Contents (Octal)	Code	Comments
1231	040014	Load CØ	<<TOS:=S-DB>>
1232	040014	Load C1	<<TOS:=Q-DB>>
1233	040014	Load C2	<<TOS:=Z-DB>>
1234	040014	Load C3	<<TOS:DL-DB>>
1235	000600	Zero	<<TOS:=DB BANK>>
1236	040013	Load C4	<<TOS:-DB>>
1237	027563	SETR %163	<<Set DB,DBBANK,DL,Z>> <<Q,S>>
1240	020012	LOAD C5	<<Form Exit Market X=Ø>>
1241	040012	LOADC6	<<ΔP>>
1242	040012	LOAD C7	<<STATUS>>
1243	100012	LOAD C10	<<DQ>>
1244	031400	EXIT Ø	<<Go to start of Diag.>>
1245	007200	CØ CON %4500	<<S-DB>>
1246	007204	C1 CON %6504	<<Q-DB>>
1247	007660	C2 CON %5000	<<Z-DB>>
1250	177770	C3 CON %-10	<<DL-DB>>
1251	002000	CR CON %2000	<<DB>>
1252	000000	C5 CON %0	<<X>>
1253	000000	C6 CON %0	<<ΔP>>
1254	100003	C7 CON %100003	<<STATUS>>
1255	000004	C10CON %4	<<ΔQ>>

A.2 CPU DIAGNOSTIC

CORE IMAGE AFTER COLD-LOADING

Octal Location

0	CSTP - %1340
1-3	0
4	0
5	QI1 = %7100
6	ZI1 = %7776
7	0
10	0
11	Q12 = %7100
12	Z12 = %7776
13	0
14-177	DRT TABLE
1340-1357	CST TABLE
1060	START OF SEGMENT 1
1430	START OF SEGMENT 2

REGISTERS FOLLOWING INITIALIZATION ROUTINE

Data Stack	Code Segment
DL = %1770	PB = %10000
DB = %2000	P = %10000
S=Q %6500	PL = Code Dependent
Z = %7000	

A.3 I/O STAND-ALONE

INITIALIZATION FOLLOWING COLD-LOAD

Absolute Memory Address (Octal)	Contents (Octal)	Code	Comments
1231	040014	Load CØ	<<TOS:=S-DB>>
1232	040014	Load C1	<<TOS:=Q-DB>>
1233	040014	Load C2	<<TOS:=Z-DB>>
1234	040014	Load C3	<<TOS:=DL-DB>>
1235	000600	Zero	<<TOS:=DB BANK>>
1236	040013	Load C4	<<TOS:=DB>>
1237	027563	SETR %163	<<Q,S>>
1240	040012	LOAD C5	<<Form Exit market X=Ø>>
1241	040012	LOAD C6	<<ΔP>>
1242	040012	LOAD C7	<<STATUS>>
1243	100012	LOAD CIO	<<DQ>>
1244	031400	EXIT Ø	<<Go to start of Stand-Alone Loader>>
1245	007200	CØ CON %7100	<<S-DB>>
1246	007204	C1 CON %7100	<<Q-DB>>
1247	007660	C2 CON %7776	<<Z-DB>>
1250	177770	C3 CON %Ø	<<DL-DB>>
1251	002000	C4 CON %Ø	<<DB>>
1252	000000	C5 CON %Ø	<<X>>
1253	000000	C6 CON %0	<<ΔP>>
1254	100003	C7 CON %100004	<<STATUS>>
1255	000004	C10CON %4	<<ΔQ>>
1256			

A.4 I/O STAND-ALONE MEMORY IMAGE AFTER LOADING

Octal Location	
0	CSTP = %1340
1	LP DRT
2	PUT LABEL
3	CONTROL A FLAG
4	CONSOLE DRT
5	QI1 = %7100
6	ZI1 = %7776
7	Launch Delta P
10	Baud rate
11	Q12 = %7100
12	Z12 = %7776
13	0
14-777	DRT TABLE
1340-1357	CST TABLE
1060	Start of Segment 1
1430	Start of Segment 2

APPENDIX B

RULES AND CAUTIONS ON USING SDUPII

B.1 CPU TYPE DIAGNOSTICS

1. DB is limited to %6000 words.

2. CPU type diagnostics run on the interrupt control stack.

3. The trap segment (Segment 1) is constructed as follows:

   ```
   LDI (Trap Number)
   Halt 1
   Exit or IXIT (depending if trap is an ICS interrupt)
   ```

4. The External Interrupt Segment (Segment 2) contains a Halt 2 and an IXIT.

B.2 I/O STAND-ALONE PROGRAMS

1. User is set up on his own stack. Q-Z is allocated by checking the prepared program file for the initial stack value.

2. There is a %600 word interrupt control stack that the user will run on during his interrupt procedure. This will be a split stack.

   ```
   DL = -1
   Q  = QI
   Z  = ZI
   DB = DB in DRT Entry (DBI)
   ```

3. User must not change the DRT entries for the line printer or the console. These are handled by SDUPII.

4. Control A is implemented in SDUPII. Memory location 3 contains a Control A flag. This flag is set to zero by SDUPII. If the user wishes to use the Control A feature, he must set the flag to a non-zero state. If the user is running with interrupts enabled and a Control A is received, SDUPII will restart the program from the main segment.

5. Programs are limited to %12 code segments and 1 data segment. The sum of code and data cannot exceed %177000 words.

6. SDUPII also supplies a general interrupt procedure. This procedure will print the DRT of the interrupting device. Therefore, it is recommended that the user fill in only the DRT entries he will be using and let SDUPII handle the others.

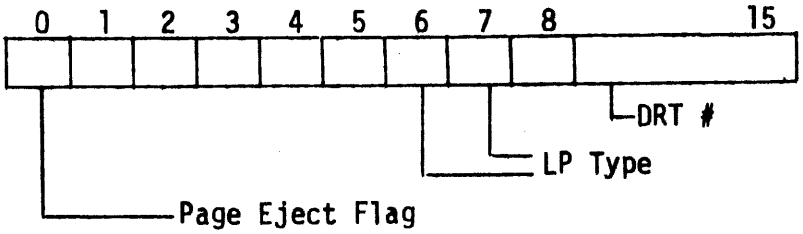
7. Power fail-auto restart must be handled by the user (if needed).

APPENDIX C

SPECIAL MEMORY LOCATIONS

ADDRESS	DEFINITION
%1	Mag tape DRT during loading - Line printer DRT during execution.

LP DRT Format



%2	PUTADDR. The external label of PUT. For SDUP use only.
%3	Control A flag.
%4	Console DRT.
%5	QI
%6	ZI
%7	Launch Delta P.
%10	Baud rate of console.
%17	Maximum program number.

APPENDIX D

TAPE INDEX RECORD

WORD	DEFINITION
%0	Program Number
%1	Q Register setting
%2	Z Register setting
%3	DB Register setting
%4	Putaddress
%5	Line printer DRT
%6	# of Data Records
%7	# of Code Records in Segment 3
%10	# of Code Records in Segment 4
%11	# of Code Records in Segment 5
%12	# of Code Records in Segment 6
%13	# of Code Records in Segment 7
%14	# of Code Records in Segment 10
%15	# of Code Records in Segment 11
%16	# of Code Records in Segment 12
%17	# of Code Records in Segment 13
%20	# of Code Records in Segment 14
%21	Entry PB of main segment

APPENDIX E

SEGMENT 1 TRAPS

Code segment number 1 contains all internal interrupt procedures. Interrupt procedures for I/O devices may be in any segment other than segment number 1.

Table E-1 lists the internal interrupts and traps with their corresponding entry numbers in the Segment Transfer Table (STT) of the internal interrupt code segment. The parameter is a value that is derived by the Interrupt Handler and which passes relevant information about the interrupt to the interrupt routine.

The Device Reference Table (DRT) contains a label for each entry, pointing to the interrupt procedure for each device. Bit 8 of the CPX1 register indicates an external interrupt. The parameter value for an external interrupt is the device number.

Table E-1. Interrupt Types

EXT. PROG. LABEL	STT NO. (%)	INTERRUPT	PARAMETER	EXECUTING STACK**
100401	1	Bounds Violation		
101001	2	Illegal Memory Address		
101401	3	Non-Responding Module		
102001	4	System Parity Error		ICS
102401	5	Address Parity Error		ICS
103001	6	Data Parity Error		ICS
103401	7	Module Interrupt	Module No.	ICS
104001	10	(Unused)		
104401	11	Power Fail		ICS
105001	12	(Unused)		
105401	13	(Unused)		
106001	14	(Unused)		
106401	15	(Unused)		
107001	16	(Unused)		
107401	17	(Unused)		
110001	20	Unimplemented Instruction		
110401	21	STT Violation		
111001	22	CST Violation		

Table E-1. Interrupt Types (Continued)

EXT. PROG. LABEL	STT NO. (%)	INTERRUPT	PARAMETER*	EXECUTING STACK**
111401	23	DST Violation		
112001	24	Stack Underflow		
112401	25	Privileged Mode Violation		
113001	26	(Unused)		
113401	27	(Unused)		
114001	30	Stack Overflow		ICS
114401	31	User Traps		
		a. Integer Overflow	%000001	
		b. Floating-Point Over.	%000002	
		c. Floating-Point Under.	%000003	
		d. Integer Divide by 0	%000004	
		e. Floating-Point Divide by 0	%000005	
		f. Ext. Prec. Floating-Point Overflow	%000010	
		g. Ext. Prec. Floating-Point Underflow	%000011	
		h. Ext. Prec. Floating-Point Divide by 0	%000012	
		i. Decimal Overflow	%000013	
		j. Invalid ASCII digit	%000014	
		k. Invalid Dec. digit	%000015	
		l. Invalid Source Word Count	%000016	
		m. Result Word Count Overflow	%000014	
		n. Decimal Divide by 0	%000020	
115001	32	(Unused)		
115401	33	(Unused)		
116001	34	(Unused)		
116401	35	(Unused)		
117001	36	(Unused)		
117401	37	Absent Code Segment		
		a. On PCAL	P-Label	
		b. On EXIT	N	
		c. On IXIT	0	
120001	40	Trace		
		a. On PCAL	P-Label	
		b. On EXIT	N	
		c. On IXIT	0	
120401	41	STT Entry Uncallable	P-Label	
121001	42	Absent Data Segment	DST No.	
121401	43	Power On		ICS
122001	44	Cold Load		ICS
		a. System I/O (SIO)	0	
		b. Direct I/O (DIO)	Label	

*Unless noted, the parameter is the External Program Label.
**Unless noted, Interrupts are serviced on the User Stack.

All User Traps (STT No. %31) are enabled by the User Traps bit in the Status register.