DECUS

PROGRAM LIBRARY

DECUS NO.	8-178
TITLE	Reverse Assembler
AUTHOR	Henry G. Dupont
COMPANY	St. George's School Newport, Rhode Island
DATE	November 25, 1968
SOURCE LANGUAGE	PAL III

DECUS Program Library Write-up   DECUS No. 8-178

ABSTRACT

The Reverse Assembler accepts a paper tape in binary format
and produces either a printed listing or a paper tape that is
acceptable to the PAL Assembler as a symbolic tape. It pro-
duces the mnemonics for almost all input-output devices as
well as PAL III and Floating Point instructions.

REQUIREMENTS

Storage

The program is loaded into locations 0-5400. The character
output tables extend from 1200-5400.

Equipment

Basic PDP-8, 8/S, or 8/I. Input is on either the high speed
reader or ASR-33. Output is on either the high speed punch or
ASR-33. Input and Output devices are selected by switch set-
tings. The type 182 EAE is not used.

USAGE

Loading

The Reverse Assembler is supplied as a binary tape and is loaded
with the BIN loader.

Switch Settings

During initialization the switch register is read to specify the
mode of operation and the input-output devices.

        BIT 0 specified the output mode of the program
        BIT 1 specifies the input device
        BIT 2 specifies the output device
        BIT 3 specifies the presence of Floating Point instructions

	BIT 0	BIT 1	BIT 2	BIT 3
ON	Output Mode 1	High Speed Reader Input	High Speed Punch Output	Presence of Floating Point Instructions
OFF	Output Mode 2	ASR-33 Input	ASR-33 Output	Absence of Floating Point Instructions

Mode 1 gives leader, a symbolic tape acceptable to the PAL III
Assembler, a "PAUSE", and trailer.
Mode 2 gives a printed listing.

Start

1. Load program with binary loader.

2. Set 0200 in the switch register, press LOAD ADDRESS.

3. Set desired modes of operation and input-output devices
    in Bits 0-3 of switch register.

4. Put binary program tape in appropriate reader.

5. Press "START"
    Program will read the binary tape and produce a listing or
    a symbolic tape until checksum is reached. At this point,
    either "PAUSE" or the checksum is typed depending on the
    output mode.

6. Restart program at 0200.

Error Messages

If the program is started in the middle of a binary tape, or the
reader misses a character, the program will be out of phase with
the tape and the first 6 bits of an instruction will be interpreted
as the last 6 bits and vice versa. When the presence of a char-
acter with channel 7 or 8 appears in the second half of an in-
struction, "READER ERROR" is typed, and the computer halts.

Recovery

To recover from reader error, restart the program.

DESCRIPTION

Discussion

The Reverse Assembler is a utility program designed to accept a
tape in binary format and produce an output in either of two modes.
Mode 1 gives leader, punches a symbolic tape, punches "PAUSE" when
the checksum is encountered, and punches trailer. The symbolic
tape is acceptable to the PAL III Assembler. Mode 2 produces a
listing that is easily read and differs from Mode 1 in the format
of output. The "PAUSE" is replaced by an octal number followed
by "(CHECKSUM)". If an address is encountered on the tape, it
is punched out in the form *XXXX where X is an octal digit.

Applications

The Reverse Assembler is used when only the binary tape of a pro-
gram is available. It makes a listing (Mode 2) and/or a symbolic
tape (Mode 1). This is especially useful when modifying a binary

2

tape that would be difficult to modify with a patch. (i.e. when changing ASR-33 input to high speed reader input because a typical high speed reader subroutine needs one more instruction.) This is done by making a symbolic tape (using Mode 1), making the desired modifications with the symbolic tape editor, and using the PAL III Assembler to produce a new modified binary tape. The Reverse Assembler may also be used to produce a revised copy of a program that has been modified by the DDT, ODT, or any other debugging program.

An example follows:

This is the output from the PAL III Assembler. It is a program
to test the Reverse Assembler.

```
                    /PAL III ASSEMBLER PASS 3 LISTING
                    /REVERSE ASSEMBLER TEST PROGRAM
0200    6046    START,  TLS
0201    4406            JMS I INPUT
0202    4407            JMS I 7
0203    6103            FPUT SAVE
0204    0400            FEXT
0205    4406            JMS I INPUT
0206    4407            JMS I 7
0207    4003            FSIN
0210    3103            FMPY SAVE
0211    6103            FPUT SAVE2
0212    0000            FEXT
0213    1103            TAD SAVE+3
0214    7006            RTL
0215    1107            TAD NUM2
0216    7004            RAL
0217    3103            DCA SAVE+3
0220    1106            TAD NUM1
0221    3110            DCA COUNT
0222    2103            ISZ SAVE+3
0223    2110            ISZ COUNT
0224    5222            JMP .-2
0225    4407            JMS I 7
0226    5103            FGET SAVE
0227    0000            FEXT
0230    4405            JMS I OUTPUT
0231    7402            HLT
                    /VARIABLES AND CONSTANTS FOLLOW:
                        *5
0005    7200    OUTPUT, 7200
0006    7400    INPUT,  7400
0007    5600            5600
                        *100
0100    0000    SAVE,   0
0101    0000            0
0102    0000            0
0103    0000    SAVE2,  0
0104    0000            0
0105    0000            0
0106    0000    NUM1,   0
0107    6424    NUM2,   6424
0110    0000    COUNT,  0

COUNT   0110
INPUT   0006
NUM1    0106
NUM2    0107
OUTPUT  0005
SAVE    0100
SAVE2   0103
START   0200
```

4

This is an output from the Reverse Assembler in MODE 1.
In MODE 1, the Reverse Assembler punches leader, the text,
and trailer. The Text is acceptable to the PAL III Assembler.

```
*0200
6046  /000  -1732  ILS                    /Addresses are in standard format
4406  /001  -3372  JMS I 0 006            /IOT instruction
4407  /002  -3371  JMS I 0 007            /This is a JMS indirectly to page 0
6100  /003  -1700  FPUT    0 100
0000  /004  +0000  FEXT                   /Floating Point instructions are
4406  /005  -3372  JMS I 0 006            /triggered by a JMS I 7
4407  /006  -3371  JMS I 0 007
0003  /007  +0003  FSIN
3100  /010  +3100  FMPY    0 100          /Floating Point Package D instructions
6103  /011  -1675  FPUT    0 103          /are assumed
0000  /012  +0000  FEXT
1103  /013  +1103  TAD     0 103
7006  /014  -0772  RTL
1107  /015  +1107  TAD     0 107
7004  /016  -0774  RAL
3103  /017  +3103  DCA     0 103
1106  /020  +1106  TAD     0 106
3110  /021  +3110  DCA     0 110
2103  /022  +2103  ISZ     0 103
2110  /023  +2110  ISZ     0 110
5222  /024  -2556  JMP       022
4407  /025  -3371  JMS I 0 007
5100  /026  -2700  FGET    0 100
0000  /027  +0000  FEXT
4405  /030  -3373  JMS I 0 005
7402  /031  -0376  HLT

*0005
7200  /005  -0600  CLA
7400  /006  -0400
5600  /007  -2200  JMP I     000          /This is data but is assigned
                                          /a CLA because of the octal code
*0100
0000  /100  +0000  AND     0 000
0000  /101  +0000
0000  /102  +0000                         /Because it is impracticable to
0000  /103  +0000                         /have two or more AND instructions
0000  /104  +0000                         /in a sequence, the program assumes
0000  /105  +0000                         /that the second and following
0000  /106  +0000                         /AND instructions are data
6424  /107  -1354  ITXON
0000  /110  +0000  AND     0 000
PAUSE                                     /This is data, but the octal
                                          /code is the same for the Data
                                          /Communications Systems Type 680
```

This is an output from the Reverse Assembler in MODE2. It is
slightly faster than mode 1 and is easire to read because there
is no slash and the output order is different.

```
*0200
000  6046  -1732  ILS
001  4406  -3372  JMS  I  0  006
002  4407  -3371  JMS  I  0  007
003  6100  -1700  FPUT     0  100
004  0000  +0000  FEXT
005  4406  -3372  JMS  I  0  006
006  4407  -3371  JMS  I  0  007
007  0003  +0003  FSIN
010  3100  +3100  FMPY     0  100
011  6103  -1675  FPUT     0  103
012  0000  +0000  FEXT
013  1103  +1103  TAD      0  103
014  7006  -0772  RTL
015  1107  +1107  TAD      0  107
016  7004  -0774  RAL
017  3103  +3103  DCA      0  103
020  1106  +1106  TAD      0  106
021  3110  +3110  DCA      0  110
022  2103  +2103  ISZ      0  103
023  2110  +2110  ISZ      0  110
024  5222  -2556  JMP         022
025  4407  -3371  JMS  I  0  007
026  5100  -2700  FGET     0  100
027  0000  +0000  FEXT
030  4405  -3373  JMS  I  0  005
031  7402  -0376  HLT

*0005
005  7200  -0600  CLA
006  7400  -0400
007  5600  -2200  JMP  I      000

*0100
100  0000  +0000  AND      0  000
101  0000  +0000
102  0000  +0000
103  0000  +0000
104  0000  +0000
105  0000  +0000
106  0000  +0000
107  6424  -1354  TTXON
110  0000  +0000  AND      0  000
2474     (CHECK SUM)
```

/MODE 2 gives Checksum

METHODS

Discussion

The program initializes by checking the switch register for the
input-output devices and the operating mode. It then reads a
binary tape. If channel 7 is punched (indicating an address)
the program punches *XXXX where XXXX is an octal number. The
current page address and the octal instruction are punched next
followed by the signed equivalent of the octal instruction. If
not an address, the first three bits of the octal instruction
are examined to find the operation code. A JMS I 0007 raises a
flag which assumes that Floating Point instructions are used
unless Bit 3 in the switch register is raised. This disables the
Floating Point instruction subroutine and prevents the Reverse
Assembler from interpreting instructions as being Floating Point
after a JMS I 0007. Operation codes 0-5 are treated as memory
reference instructions and are always typed out. If the instru-
tion has an operation code of 6 or 7, the Reverse Assembler com-
pares the word to a table on pages 5-8. If the word is found
in the table, the mnemonic code is typed out and the next char-
acter on the paper tape is read. If the word is not found on
the tables, it is assumed the word is not an instruction, but
data, and the computer reads the next character having typed
only the address and the octal code. The IOT list contains every
IOT instruction except the ones for the Serial Magnetic Drum
System Type 251 and Type RM08 as well as the instructions for the
Garded Scanning Digital Voltmeter Type AF04A. This is because
the same octal codes are used by two or more different input-
output devices. The program checks for the checksum and punches
either "PAUSE" (in Mode 1) or "XXXX (CHECKSUM)" (in Mode 2).
The Reverse Assembler punches leader-trailer in Mode 1.

FORMAT

Input Data

On input, the program accepts a paper tape in binary format. A
tape in RIM format may be read, however, an address will be typed
on every other line.

Output Data

The output depends on the preselected output mode. In mode 2,
the computer types a 3-digit word which indicates the address of
the instruction on the current page. Next comes a four-digit
octal code indicating the code as read off the paper tape. This
is followed by a signed 4-digit number which is the equivalent
of the octal number. (i.e. 7773 is equivalent to -5.) If the
octal number represents a memory reference instruction, the
mnemonic code is typed. If it is an indirect reference, an "I"
is typed. If it is a reference to page 0, a "0" is typed. If
a 0 is not typed, the reference is to the current page.

EXECUTION TIME

The speed of the Reverse Assembler is input-output limited. The ASR-33 types a line in approximately 2.5 seconds. The high speed punch punches a line in approximately 0.5 seconds.

PROGRAM

Core Map

        0-1200 : executable program
        1200-1400 : Reference table for operation code 7
        1400-2000 : Reference table for operation code 6 (IOT)
        2000-5400 : ASCII table of all mnemonic codes.