IDENTIFICATION

Product Code: DIGITAL-8-28-U-SYM.

Product Name: Single Precision Decimal-to-Binary Conversion
and Input ASR 33, Signed or Unsigned

Date Created: January 14, 1966

Maintainer: Software Service Group

1.   ABSTRACT

This routine accepts a string of up to four decimal digits (single precision for the PDP-8) from the Teletype keyboard and converts it to the corresponding 2's complement binary number.

The string may contain as legal characters a sign (+, −, or space) and the digits from 0 - 9. If the first legal character is not a sign, the conversion is <u>unsigned</u>. A back arrow (←) at any point in the string erases the current string and allows the operator to reenter the correct value. Any character after the first, other than another digit or back arrow causes the conversion to terminate and is found in location SISAVE within the subroutine.

2.   PRELIMINARY REQUIREMENTS

2.1   <u>Storage</u>

This subroutine requires 74 core locations.

2.2   <u>Equipment</u>

Basic PDP-8 with ASR 33

3.   LOADING OR CALLING PROCEDURE

3.1   <u>Loading</u>

The symbolic tape provided may be assembled with the user's main program by either PAL III or MACRO-8. The symbolic tape has neither an origin setting nor a terminating "$", but does have a PAUSE pseudo-instruction at the end.

3.2   <u>Calling Sequence</u>

The subroutine is called by an effective JMS to location SICONV. Return is to the location immediately following the calling JMS with the binary number in the AC (accumulator).

4.   USING THE ROUTINE

4.1   <u>Errors in Usage</u>

If a sign (+, −, or space) precedes the string of decimal digits, the maximum decimal number correctly accepted is 2047 ($2^{11}-1$). The sign, if any, must appear first. If a sign does not precede the string of decimal digits, the maximum decimal number correctly accepted is 4095 ($2^{12}-1$).

4.2   <u>Recovery from Such Errors</u>

If either of these maxima is exceeded, the results are unspecified.

5.   RESTRICTIONS

5.1   <u>Status Active Registers</u>

The status of the AC and link is not preserved.

5.2   <u>Status Hardware</u>

This subroutine should not be used when the interrupt is on.

5.3   <u>Miscellaneous</u>

The magnitude restrictions on numbers are described in section 4.1.

1

6. DESCRIPTION

6.1 Discussion

This subroutine converts to the binary equivalent a signed or unsigned string of decimal numbers read from the console keyboard of the PDP-8. If a minus sign is specified, the results are in 2's complement negative form. The first character is examined and, if it is a sign (+, −, or space), a switch is set to provide the correct sign for the conversion. Regardless, a switch is set after the first character to terminate conversion if a character other than a decimal digit or rub out appears. If a back arrow appears at any time, the conversion is reinitialized and the subroutine waits for the correct entry.

The last four bits of the ASCII code for each of the decimal digits are identical to the standard 8-4-2-1 BCD code. Thus, the BCD digit is extracted from the 8-bit code by the AND instruction with a "mask" of 17_8. When the first BCD digit comes in, it is added to a cleared location (SJHOLD) in memory and stored back in that location. When the next legal character comes in, location SJHOLD is multiplied by 10, _then_ added to the BCD code of the character and returned to location STORE. This sequence holds true for a decimal number of any arbitrary length.

6.2 Example and/or Application

Since the PDP-8 can add and shift easily, the multiplication by 10 can be accomplished in three instructions. Since a shift left is equivalent to a multiplication by 2, a double shift left is equivalent to a multiplication by 4. Assume that the number currently in STORE is 5, and the new code just coming in is the number 1 stored in HOLD. The program sequence to perform the multiplication and storage is as follows:

Instruction Sequence	Comment	Contents of AC
CLA		
TAD STORE	/Load C(STORE) into AC	000 000 000 101
CLL RTL	/Multiply C(STORE) by 4	000 000 010 100
TAD STORE	/Add STORE giving C(STORE) by 5	000 000 011 001
CLL RAL	/Multiply by 2 giving C(STORE) by 10	000 000 110 010
TAD HOLD	/Add in the next number	000 000 110 011
DCA STORE	/Store back into STORE and return to wait for next character	000 000 000 000

The number residing in location STORE is 0063_8 or 0051_{10}.

If the next number to come in were "9," using the same sequence and conditions, the result would be 001 000 000 111, the binary equivalent of 519.

6.3   Scaling

This subroutine assumes an integral decimal number (signed or unsigned) and yields an integral binary equivalent (signed or unsigned respectively).

7.   METHOD

The algorithm used is illustrated above (6.2) with details shown in the listing (10.1).

8.   FORMAT

8.1   Input

The input string may or may not contain a sign (+, −, or space). Any character other than a sign, 0 - 9, or back arrow causes the subroutine to terminate, as does a sign in any but the first position.

8.2   Core Data

The terminating character is found in location SISAVE.

8.3   Output

Spacing, tabulation, carriage return, etc., are not provided for in this subroutine. See DIGITAL-8-19-U-Sym which contains short subroutines for the latter purposes.

9.   EXECUTION TIME

9.1   Average

This subroutine is input limited at a maximum of 10 hz.

10.   PROGRAM

10.1 Program Listing

/SINGLE PRECISION DECIMAL INPUT FROM KEYBOARD
/CALLING SEQUENCE: JMS SICONV
/ACC IGNORED, RETURN WITH BINARY WORD IN ACC

```
0200    0000    SICONV, 0
0201    7300    CLA CLL
0202    1274    TAD SISET1 + 1          /INITIALIZE PROGRAM SWITCHES
0203    3232    DCA SICTRL
0204    1274    TAD SISET1 + 1
0205    3224    DCA SIXSW1
0206    3310    DCA SIHOLD
0207    3311    DCA SINEG1             /CLEAR NEGATIVE SWITCH
0210    5257    JMP SINPUT
0211    3307    SIPROC, DCA SISAVE
0212    1307    TAD SISAVE             /STORE AND THE PROCESS CHARACTER
0213    1301    TAD SIRBUT
0214    7450    SNA                    /IS IT A "BACK-ARROW" (IE. ERASE) KEY
0215    5201    JMP SICONV + 1         /YES, REINITIALIZE
0216    1302    TAD SIM260
0217    7510    SPA                    /IS IT LESS THAN 260 (IE. "0")
0220    5232    JMP SICTRL             /YES. TRANSFER TO SEE WHAT CHAR. IT IS
0221    1303    TAD SIM271
0222    7740    SMA SZA CLA            /IS IT GREATER THAN 271 (IE. "9")?
0223    5232    JMP SICTRL             /YES, TRANSFER TO SEE WHAT CHARACTER IT
0224    7300    SIXSW1, CLA CLL        /NO, FIRST CHARACTER WAS A DECIMAL DIGIT
0225    1231    TAD .+4                /CLOSE SWITCH TO GO TO "SINMBR" NEXT
0226    3224    DCA .-2
0227    1245    TAD SINMBR -1          /SET SWITCH TO SENSE TERMINATING CHAR.
0230    3232    DCA SICTRL
0231    5246    JMP SINMBR
0232    7300    SICTRL, CLA CLL        /CONTINUE CHECKING
0233    1307    TAD SISAVE
0234    1304    TAD SIMSPC
0235    7450    SNA                    /IS IT A SPACE?
0236    5274    JMP SISET1 + 1         /YES, SET SWITCH TO SENSE TERM. CHAR.
0237    1305    TAD SIMPLS
0240    7450    SNA                    /IS IT A "PLUS"?
0241    5274    JMP SISET1 + 1         /YES, SET SW TO SENSE TERM. CHAR.
0242    1306    TAD SIMMNS
0243    7650    SNA CLA                /IS IT A MINUS?
0244    5273    JMP SISET1             /YES, SET NEGATIVE X SWITCH AND TERM SW.
0245    5264    JMP SIEND              /NO, IT WAS A TERMINATING CHAR.
0246    1310    SINMBR, TAD SIHOLD     /MULTIPLY CURRENT ASSEMBLED NUMBER BY 10
0247    7106    CLL RTL
0250    1310    TAD SIHOLD
0251    7004    RAL
0252    3310    DCA SIHOLD
0253    1307    TAD SISAVE             /PICK UP CURRENT DIGIT
0254    0300    AND SIMASK             /MASK OFF THE HIGH ORDER BITD
0255    1310    TAD SIHOLD             /ADD TO ASSEMBLED NUMBER
```

4

```
0256    3310    DCA SIHOLD              /STORE BACK IN SIHOLD
0257    6031    SINPUT, KSF             /INPUT ROUTINE
0260    5257    JMP .-1
0261    6036    KRB
0262    6046    TLS
0263    5211    JMP SIPROC
/TERMINATING ROUTINE
0264    7300    SIEND, CLA CLL
0265    1311    TAD SINEG1
0266    7010    RAR                     /PUT NEGATIVE SWITCH INTO LINK
0267    1310    TAD SIHOLD
0270    7430    SZL                     /IS THE LINK "1"?
0271    7041    CMA IAC                 /YES, NUMBER NEGATIVE. COMPLEMENT
0272    5600    JMP I SOCONV            /RETURN.
0273    2311    SISET1, ISZ SINEG1      /SET NEGATIVE SWITCH
0274    7300    CLA CLL
0275    1245    TAD SINMBR -1           /CLOSE SW TO TRANSFER TO TERM.
0276    3232    DCA SICTRL
0277    5257    JMP SINPUT
/CONSTANTS AND VARIABLES
0300    0017    SIMASK, 17
0301    7441    SIRBUT, -337            /CODE FOR ERASE
0302    0057    SIM260, 57              /NUMBER USED TO GENERATE CODE "260"
0303    7767    SIM271, -11             /NUMBER USED TO GENERATE CODE "271"
0304    7540    SIMSPC, -240            /CODE FOR SPACE
0305    7765    SIMPLS, -13             /NUMBER USED TO GENERATE CODE "253" (+)
0306    7776    SIMMNS, -2              /NUMBER USED TO GENERATE CODE "255" (-)
0307    0000    SISAVE, 0               /STORAGE LOCATIONS
0310    0000    SIHOLD, 0
0311    0000    SINEG1, 0
```

11. DIAGRAMS

11.1 Flow Chart
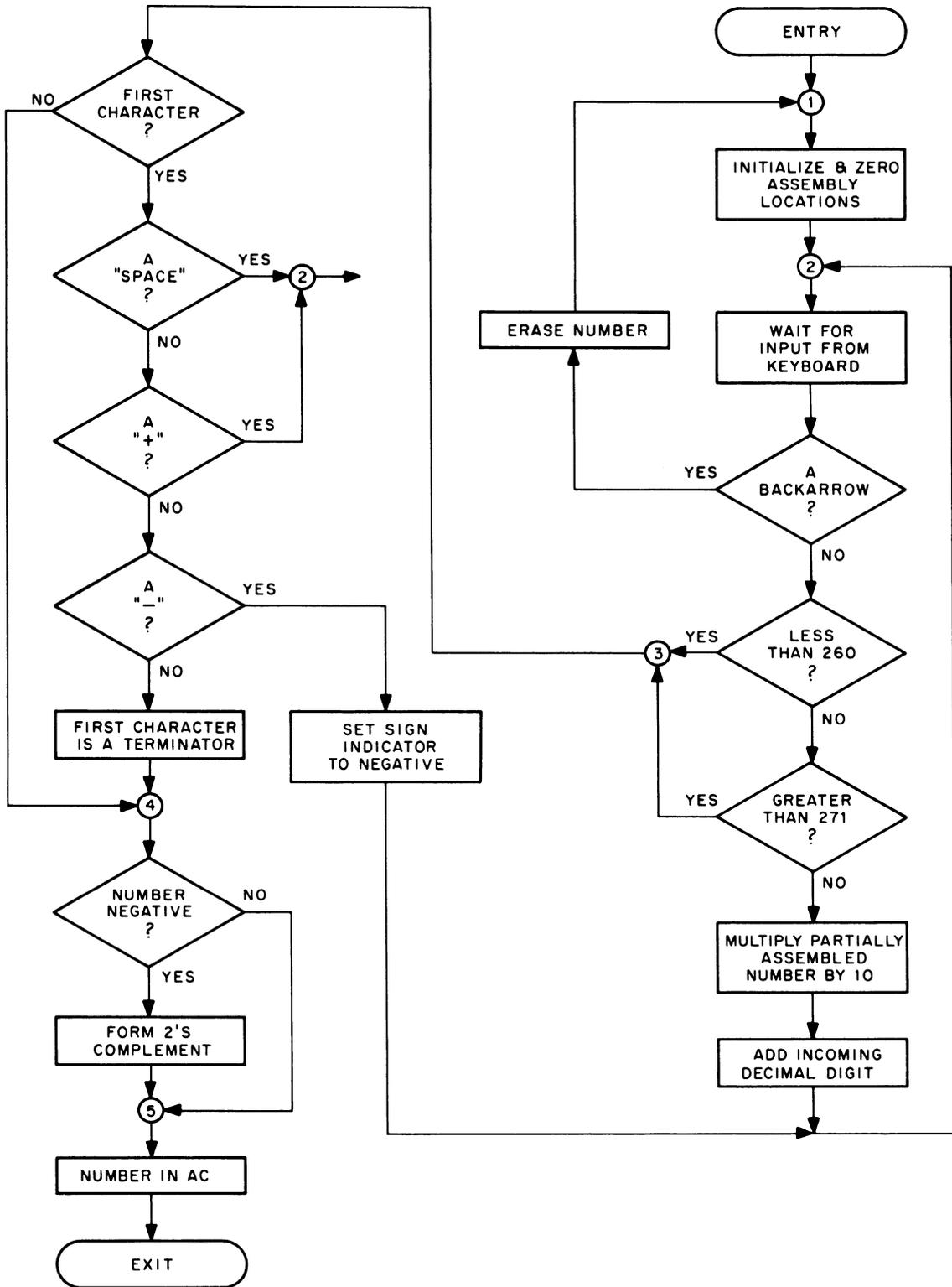
12.	REFERENCES

12.1	Other Library Programs

DIGITAL-8-19-U-Sym