**DATA GENERAL CORPORATION**

Southboro,
Massachusetts 01772
(617) 485-9100

PROGRAM

Double Precision Negate

TAPES

ASCII Source: 090-000019

ABSTRACT

This routine negates a double precision, two's complement number.

093-000014-00

1. REQUIREMENTS

   1.1 Memory

   1K or larger alterable memory

   1.2 Equipment

   NOVA central processor

   1.3 External Subroutines

   None

   1.4 Other

   None

2. OPERATING PROCEDURE

   2.1 Calling Sequence

   ```
   JSR   .DNEG
   return
   ```

   2.2 Input Format

   A double precision, two's complement number in AC∅ (high order), AC1 (low order).

   2.3 Output Format

   The negative of the input is returned in AC∅ (high order), AC1 (low order).

   2.4 Error Returns

   None

   2.5 State of Active Registers upon Exit

   AC∅, AC1, AC3, and Carry are destroyed by .DNEG. AC2 remains unchanged.

## 2.6 Cautions to User

The negative of -2**31 cannot be represented and is returned unchanged.

## 3. DISCUSSION

### 3.1 Algorithms

If the input is

$$2**16*A + B$$

then the negative of the input

$$= 2*32 - (2**16*A + B)$$

$$= 2**16*(2**16 - A) + 2**32 - B$$

Now 2**32 - B is equivalent to a two's complement of B which always causes a "borrow" from the high order unless B = $\emptyset$. Therefore, if B = $\emptyset$, the two's complement of A is required. Otherwise,

$$2**16 - A - 1$$

is required (which is the one's complement).

### 3.2 Limitations and Accuracy

The routine is exact.

### 3.3 Size and Timing

The routine is 4 words in length.

Execution time is 13.8 $\mu$ seconds.

### 3.4 References

See section 2.2 of "How to Use the NOVA" for a further discussion of double precision arithmetic.

### 3.5 <u>Flow Diagrams</u>

None

## 4. <u>EXAMPLES AND APPLICATIONS</u>

An ASCII source tape of .DNEG is provided with the NOVA software.  The tape should be edited into user software that requires double precision negate.

## 5. <u>PROGRAM LISTING</u>

A listing of .DNEG follows.  No origin has been given, enabling the user to edit the source anywhere within his routines.

```
; DOUBLE NEGATE
; COMPUTES -D WHERE D IS A DOUBLE PRECISION TWO'S
;         COMPLEMENT INTEGER.

; INPUT:          D IN AC0, AC1 (HIGH ORDER, LOW ORDER)

; OUTPUT:         -D IN AC0, AC1

; CALLING SEQUENCE:
;        JSR     .DNEG
;        RETURN

; CAUTION:        THE NEGATIVE OF -2**31 CANNOT
;                 BE REPRESENTED AND IS RETURNED UNCHANGED

; DESTROYED:      AC0, AC1, AC3, CARRY
; UNCHANGED:      AC2


00000 124404 .DNEG:  NEG 1,1,SZR     ; LOW ORDER NEGATE
00001 100001         COM 0,0,SKP     ; IF AC1 NON-ZERO
                                     ; AC0 IS A ONE'S COMPLEMENT
00002 100400         NEG 0,0         ; IF AC1 IS ZERO
                                     ; AC0 IS A TWO'S COMPLEMENT
00003 001400         JMP 0,3         ; RETURN
```