IDENTIFICATION: DECIMAL OUTPUT I
Single Precision, Fixed Point Subroutine

PURPOSE: To convert a binary-scaled, single precision number to decimal form and print out the decimal number.

RESTRICTIONS:

1. The magnitude of the number, ignoring decimal point, must be less than 2,097,152.

2. The output number, ignoring decimal point, can have a maximum of 7 decimal digits.

3. If the output number is entirely fractional, the maximum number of decimal digits possible is 6.

4. The binary scaling of the number to be converted must be positive and less than 21.

STORAGE: Sectors 000-306 of line 05 for the routine, plus all of line 00 for temporary storage.

TIMING: The timing of the program is dependent on the output typing speed of the Flexowriter -- about 10 characters/second.

ACCURACY: The output number, N, is correctly rounded:

$$|X - N| \leq 5 \times 10^{-(R+1)}$$

Where X = Binary scaled number in A

N = Decimal output number typed

R = Number of digits printed to the right of the decimal point

USE:

1. Calling Sequence

LDA	number to be converted @ Q
LDB	keyword (see below)
LDC	return instruction
TRU	015 05

The keyword is of the form

+ Q Q L L R R X

2 octal digits each — 1 octal digit

Where:
- + = sign of keyword (must be positive)
- Q = binary scaling of number to be converted
- L = number of digits to be printed to the left of the decimal point
- R = number of digits to be printed to the right of the decimal point
- X = spare character (must be zero)

2. Output

The decimal number will be typed in the form

S I . F

Where S is the sign of the number (space = plus, - = minus), I is the integer part of the number (L digits), and F is the fractional part of the number (R digits). The sign and decimal point will always be printed.

Leading zeroes before the decimal point are suppressed, except that if the number being converted equals 0, there is no zero-suppression.

If any of the above restrictions are not met, the program will not provide any alarms, but will output erroneous data.

METHOD:

The binary scaled number, X @ Q, is transformed to a positive, decimally-scaled number, $|X| . 10^{R}$ @ 21, and rounded. Successive division by 10 will produce as remainders the decimal digits, least significant digit first, with the quotient providing the new dividend. When L + R remainders (digits) have been generated, these then provide the linkage to a corresponding WOC instruction. Zero suppression and decimal output are determined by counting each place during output, and comparing with L and R.