File # 1.01.01

Operation

Floating Arithmetic Interpretive Routine

A routine to interpret and process floating point orders on a fixed point computer.

FAIR

Use

a) Calling Linkage

327

L :	00d 1 L+1 01c ~~35f~~	ENTER FAIR	$\alpha =]W_1[$
L + 1:	000 5 000 α 2b2		
L :	080 5 000 000 β	EXIT FAIR	(first floating point order)

The entrance linkage gives FAIR control of the program, starting with the order at W_1. The routine retains control until the next exit linkage is reached, when control of the program is yielded to the computer at order β.

b) Adaptation Link Word - Not relocatable

c) Storage

160 words: 2a7 to 346
All opstos self-contained

Requirements and Performance

a. Method of operation Mainly logical (floating point by simulation)

b. Additional routines required None

c. Range and form of data Conventional order format, except for sign.

1) All arithmetic orders to be processed as floating point must be negative (first hexadecimal character must be 2).

2) Arithmetic orders to be processed as straight fixed point should carry a + sign.

3) All non-arithmetic orders (TN, TZ, I, O, E, H) must always be +.

Note that although the comparison in all conditional transfer orders is made between two floating point numbers, all such orders must nevertheless be +.

d. Accuracy In floating point operations, one bit may be lost
See write-up [5 a (5)].

e. Performance time: Overall, about 70 cycles/order = 1.2 sec/order

Remarks

a. All operands for compare orders must be normalized.

b. The result of the subtraction associated with compare orders is never formed.

c. Although short memory may be used just as in normal programming, the result of order N is delivered before operations, with N + 1 , are begun.