RECOMP II USERS' PROGRAM NO. 1133

PROGRAM TITLE: FLIP-FLOP IMPLEMENTATION FOR RECOMP II

PROGRAM CLASSIFICATION: Service

AUTHOR: Lt. D. Brown, USASCS,
Fort Monmouth, New Jersey

PURPOSE: Provides the Recomp with 38 addressable conditional flip-flops. Any one or combination of these can be set or reset by other programs to "remember" the presence or absence of special conditions.

DATE: 1 March 1962

Published by

RECOMP Users' Library

at

AUTONETICS INDUSTRIAL PRODUCTS

A DIVISION OF NORTH AMERICAN AVIATION, INC.

3400 East 70th Street, Long Beach 5, California

PROGRAM NO.
ORIGIN DATE: 1 March 62
REVISION DATE:
PROGRAMMER: Lt. D. Brown, USASCS,
Fort Monmouth, N. J.

PROGRAM TITLE: FLIP-FLOP IMPLEMENTATION FOR RECOMP II

1. PURPOSE: Provides the Recomp with 38 addressable conditional flip-flops. Any one or combination of these can be set or reset by other programs to "remember" the presence or absence of special conditions.

2. RESTRICTIONS

Flip-flop designations range from 01_8 to 46_8. Attempts to set or interrogate a higher-numbered flip-flop will be ignored.

3. METHOD

Bit positions 2 - 39 in a memory word are treated as flip-flops. A binary one is placed in a designated bit position to "set" it. Zero in any bit position indicates a "reset" state.

4. USAGE

4.1 Calling Sequence

The Routine may be called in two ways:

4.1.1 By a direct transfer to location XXX1, where XXX0 is the first location of the routine. See 4.5, Options available

4.1.2 Through the trapping mode. Whenever the flip-flop designation of the calling instruction is negative, no additional transfer instruction is needed. The computer will trap and go into the routine. See 4.5, Options available

4.2 Explanation of Symbols

See 4.5, Options available

4.3 Extent of Storage

Occupies 56 octal locations

4.4 RETURNS

4.4.1 There is no special error return. Addressing nonexistent flip-flop will be ignored, and return will be made to the next instruction after the calling sequence.

4.4.2 Whenever a flip-flop is being unconditionally set or reset, return is made to the next instruction after the calling sequence.

4.4.3 Whenever a flip-flop is interrogated and found to be in the reset state, return is made to the next instruction after the calling sequence.

4.4.4 Whenever a flip-flop is interrogated and found to be in the set state, return is made to the transfer address specified in the calling instruction.

4.5 Options Available

When a direct transfer is made to the routine, the calling instruction must occupy a single full word in memory. The first half of the word is a transfer instruction to XXX1.0, where XXX0.0 is the start location of the routine. The second half of the calling instruction word is a pseudo-operation that may be constructed according to the following options:

4.5.1

```
                          A B     C
+57 XXX1 0   +27   0000   0
```

When B designates a flip-flop, C is zero, and A is:

positive: set flip-flop 27 and return to the calling program.
negative: reset flip-flop 27 and return to the calling program.

4.5.2

```
                          A B     C
+57 XXX1 0   +27   3754   1
```

When B designates a flip-flop, C gives a transfer option, and A is:

positive: sense flip-flop 27, but leave as-is. If set, transfer to location 3754.1 Otherwise, return to calling program.
negative: sense flip-flop 27 and leave in reset state. If initially set, transfer to location 3754.1. Otherwise, return to the calling program.

4.5.3

```
                          A B     C
+57 XXX1 0   +00   3754   1
```

When B is zero, A is:

positive: set all flip-flops
negative: reset all flip-flops

and,

C is:

0000.0: return to program
3754.1: go to location 3754.1

4.5.4 Whenever the flip-flop designation sign is negative, the +57 XXX1 0 instruction may be used, but it is not needed. If it is not used, a designation such as -27 3754 1 may be placed in either half of an instruction word. It will cause the RECOMP to enter the trapping mode and then enter the flip-flop subroutine.

4.6 Utilization of L & V Loops

High speed loops are not tied up by this routine.

5. CODING INFORMATION

5.1 Constants

5.1.1 After relocating the routine to the desired place in memory, the following instruction must be placed in location 0000:

+15 XX56 0 +57 XXX3 1

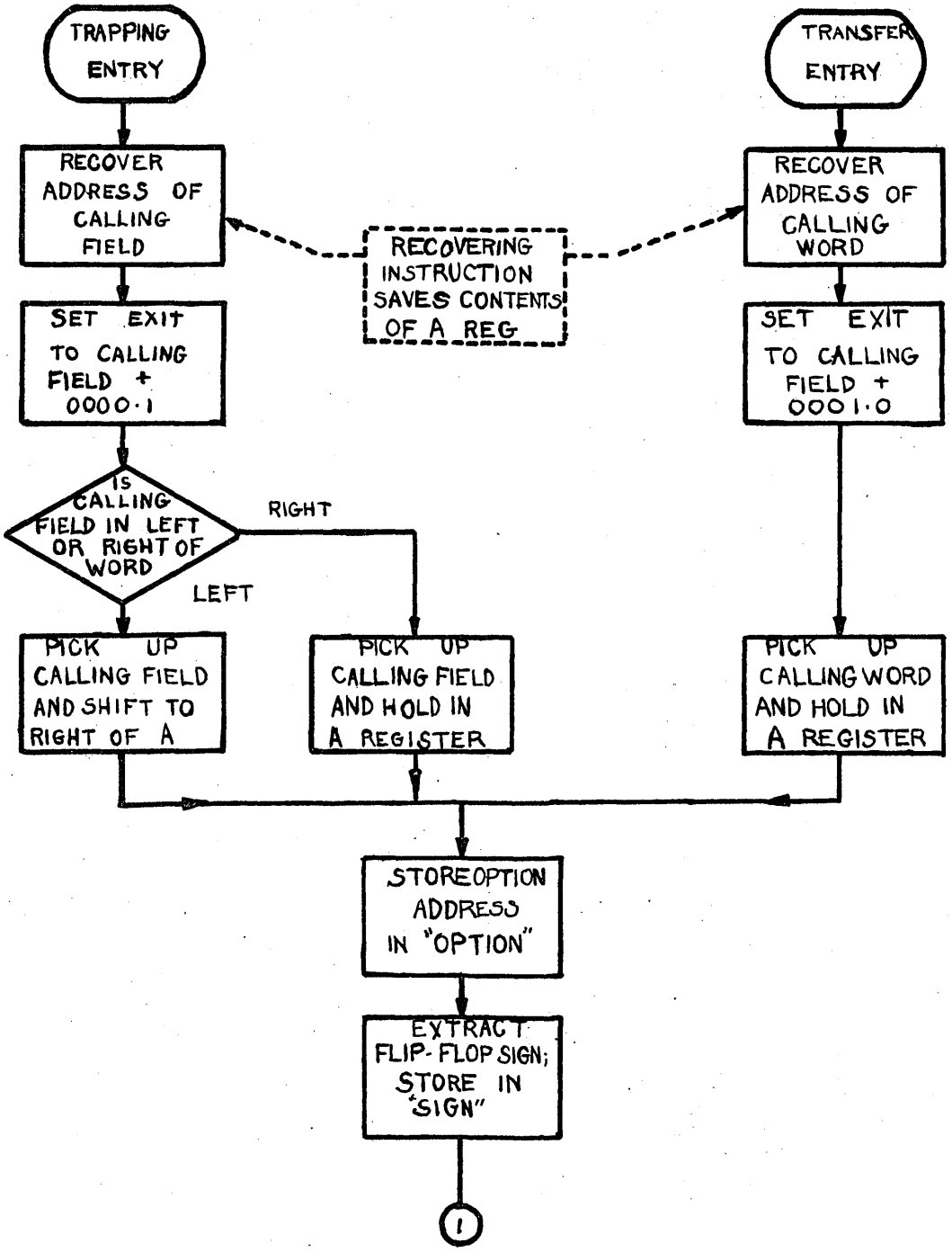
(XXX0 is the relocated start of the flip flop routine.)

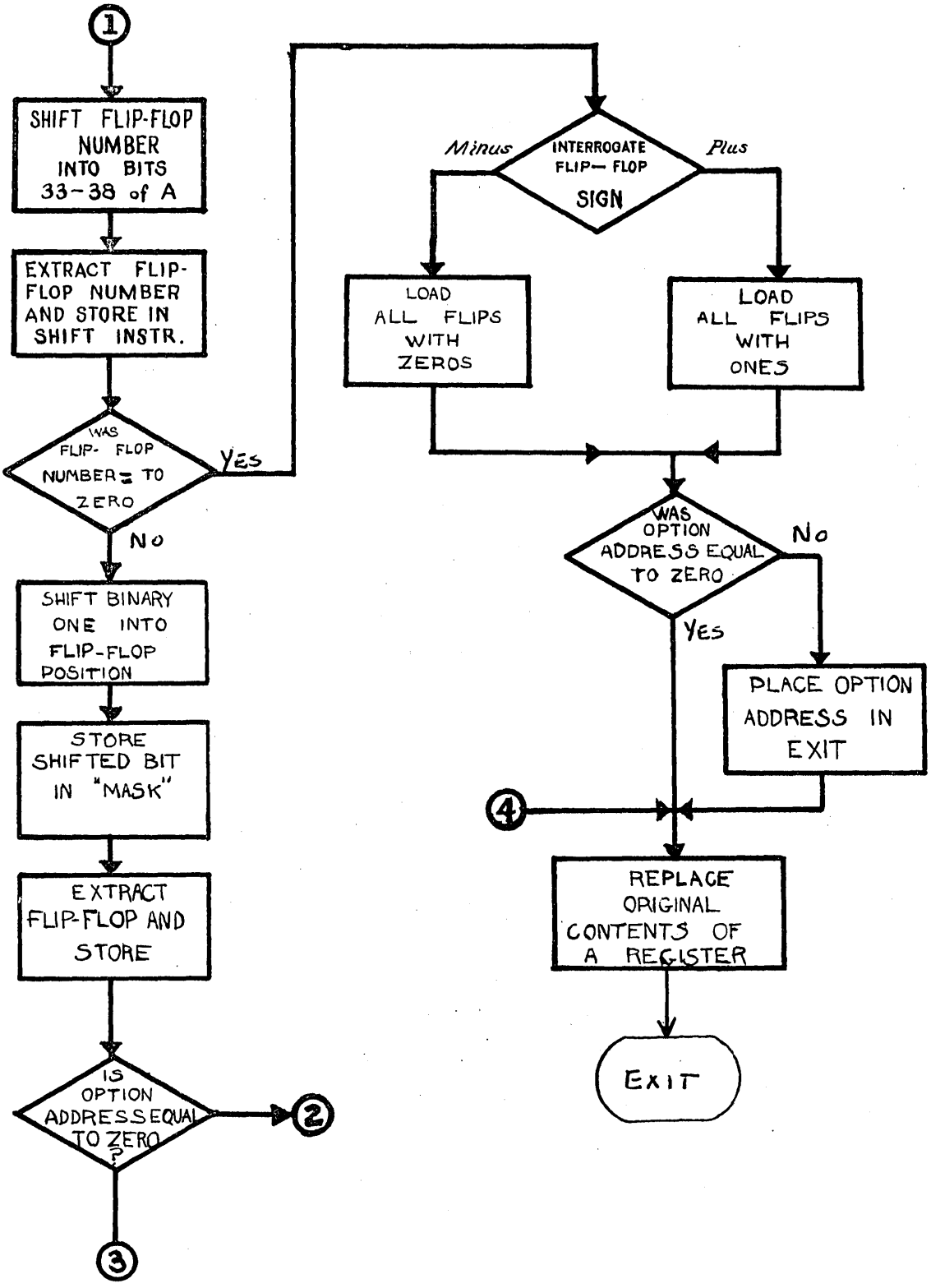
5.1.2 The flip-flop holding register may be examined at location XX50.

5.2 Execution time

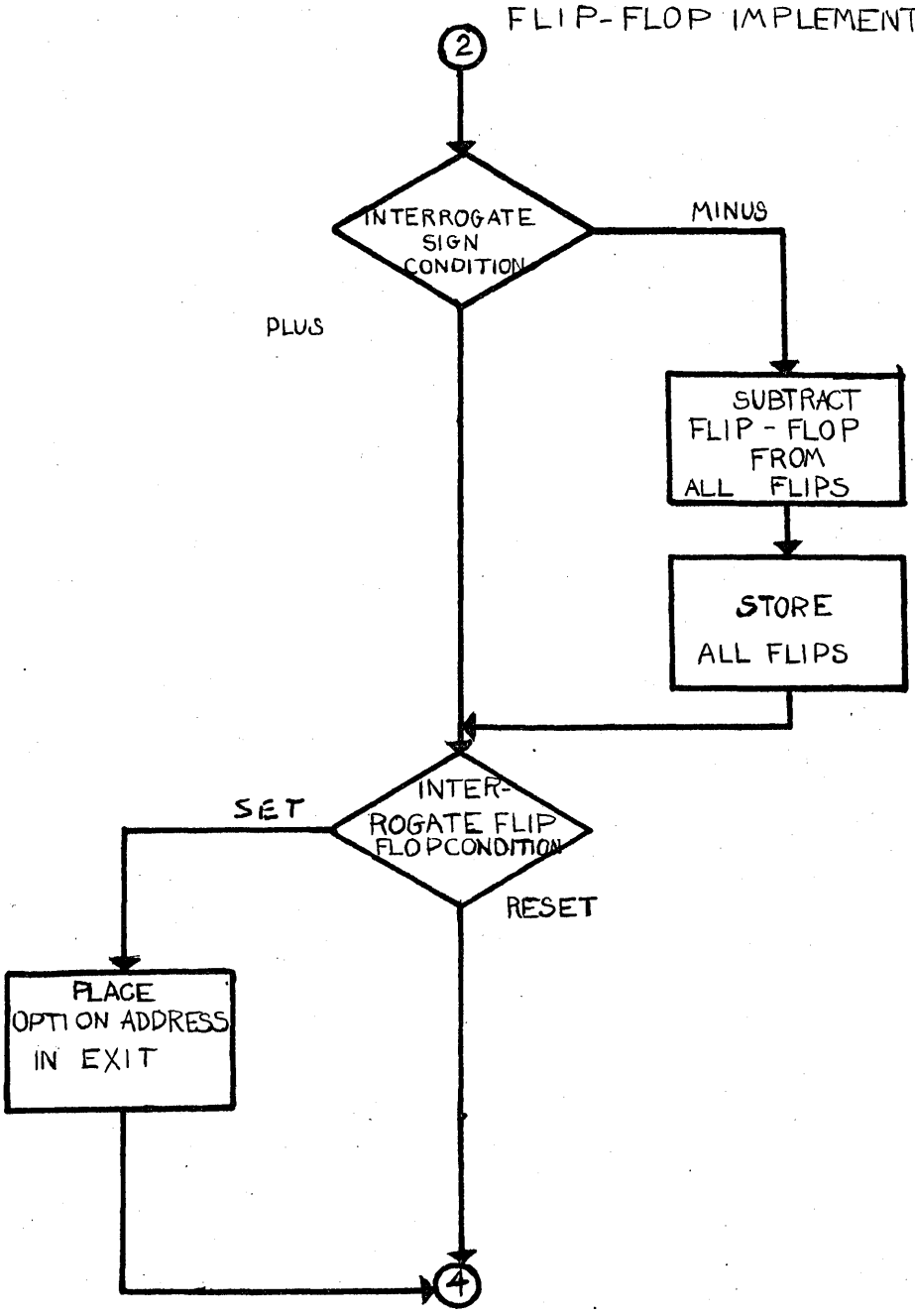
Averages 0.47 seconds.

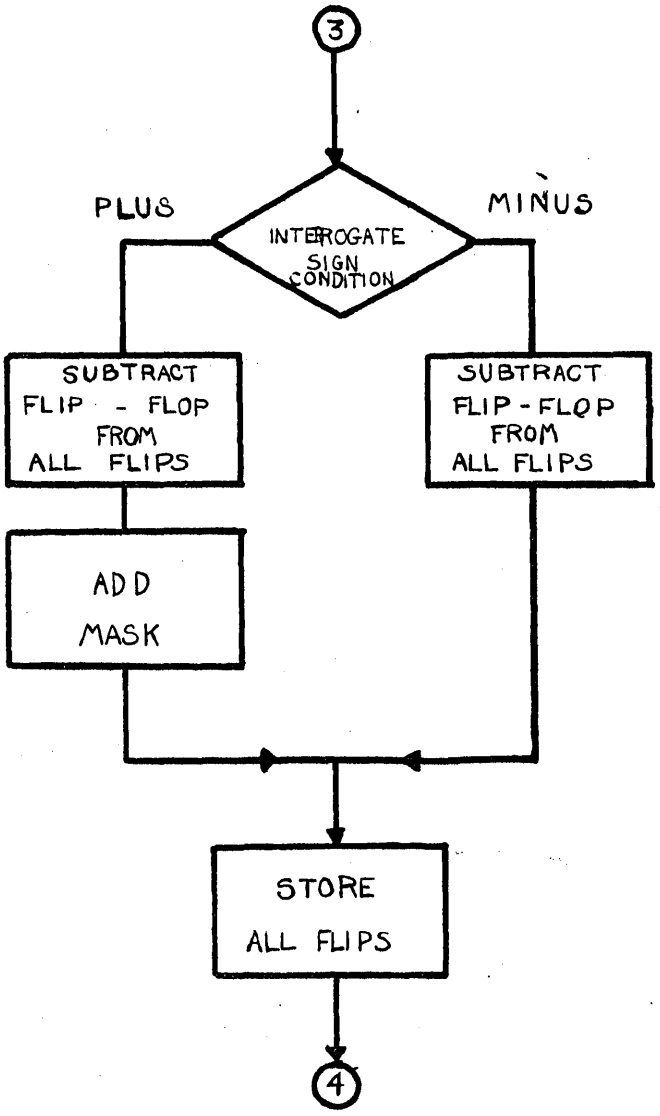
FLIP-FLOP IMPLEMENTATION SUBROUTINE




1
SHIFT FLIP-FLOP NUMBER INTO BITS 33-38 of A
EXTRACT FLIP-FLOP NUMBER AND STORE IN SHIFT INSTR.
WAS FLIP-FLOP NUMBER = TO ZERO
Yes
No
SHIFT BINARY ONE INTO FLIP-FLOP POSITION
STORE SHIFTED BIT IN "MASK"
EXTRACT FLIP-FLOP AND STORE
IS OPTION ADDRESS EQUAL TO ZERO?
2
3
INTERROGATE FLIP-FLOP SIGN
Minus
Plus
LOAD ALL FLIPS WITH ZEROS
LOAD ALL FLIPS WITH ONES
WAS OPTION ADDRESS EQUAL TO ZERO
No
Yes
PLACE OPTION ADDRESS IN EXIT
4
REPLACE ORIGINAL CONTENTS OF A REGISTER
EXIT

FLIP-FLOP IMPLEMENTATION 3




3
PLUS
INTERROGATE SIGN CONDITION
MINUS
SUBTRACT FLIP - FLOP FROM ALL FLIPS
SUBTRACT FLIP - FLOP FROM ALL FLIPS
ADD MASK
STORE ALL FLIPS
4