

JAN 29 1960

**SDC** FIELD NOTE A WORKING PAPER

*M. Blauer*  
FN-10-204

The views, conclusions, and recommendations expressed herein do not necessarily reflect the official views or policies of either the Air Force or the System Development Corporation.

Although this working paper contains no classified information it has not been cleared for open publication by the Department of Defense. Open publication, wholly or in part, is prohibited without prior approval of the System Development Corporation.

AUTHOR  
*E. Foote*  
E. Foote

APPROVED  
*Martin Blauer*  
M. Blauer

(Produced under System Development Corporation sub-contract No. 202 issued by International Electric Corporation in performance of contract AF-30(635)-11583)

PAGE 1 OF 8 PAGES

SYSTEM DEVELOPMENT CORPORATION • 2500 COLORADO AVENUE • SANTA MONICA, CALIFORNIA



PROGRAM DESIGN SPECIFICATION  
FOR THE  
INTERPRETER SECOND PASS (JOLIZ)



PROGRAM DESIGN SPECIFICATION FOR  
THE INTERPRETER SECOND PASS (JOLIZ)  
A SUB-PROGRAM OF THE JOVIAL INTERPRETER SYSTEM

TABLE OF CONTENTS

Introduction . . . . .	.Page 3
Environment . . . . .	.Page 3
Input . . . . .	.Page 3
Functions . . . . .	.Page 4
Output . . . . .	.Page 4
Message Printouts. . . . .	.Page 5
Flow Diagram . . . . .	.Page 7

PROGRAM DESIGN SPECIFICATION  
FOR THE  
INTERPRETER SECOND PASS (JOLIZ)

INTRODUCTION

The Second Pass of the Interpreter (hereinafter referred to as JOLIZ) has as its purpose the interpretation of the source program. To do this it uses the internal tables formed by JALLZ and the initial data set up by the Data Simulation Program JSTRZ. JOLIZ performs all of the operations specified in the Intermediate Language Table (ILT). It will continue to interpret in this fashion until it finds a STOP instruction in the Intermediate Language Table, at which time it returns to the Control Program. During the execution of every step in ILT, JOLIZ will note floating operation trapping errors, division by zero, and certain logical errors inherent to the Object Program; log them as errors and continue.

An additional feature is the production of a buffer tape containing dynamic "snapshots" of certain interpretive functions. This is later used by the Data Processing Program in the preparation of the "TRACE" output.

ENVIRONMENT

Part of the environment of JOLIZ is the tables ILT, SWT, STAT, VAT, and CON. These tables are described in the section "OUTPUT" of FN-LO-201, "Program Design Specification for the Interpreter First Pass". In addition, the initial data, which has been stored in the proper locations by the Data Simulation Program exists in core memory. One communication item, HPARAM, which contains the number of parameter items in VAT, exists in core memory for JOLIZ to use. In addition, the five communication items MATTBL, MATTBL+1, MATTBL+2, MATTBL+3, and MATTBL+4 are used by JOLIZ to locate the five tables used by JOLIZ.

INPUT

There is no dynamic input to JOLIZ.

### FUNCTIONS

The only function of the second pass is to perform the operations on the variables, constants, and subscripts as indicated by the OPERI in the ILT table. The OPERI are: EQ, NQ, LS, GR, GQ, LQ, (+), (-), (\*), (/), END, SET, ^, STOP, GOTO, TERM, procedure declaration, switch declaration, and close declaration. The variables and constants can be of five forms: status (not constants), floating, integer, Hollerith, fixed mixed fraction. Since the Interpreter should operate with a certain degree of speed it would be undesirable to analyze all the arithmetic operations of a complete statement before interpreting it. Instead, all the integers and fixed mixed fractions are floated when they are used and the interim data is retained as floating numbers, thereby allowing the mixing of any type of number (floating, integer, fixed mixed fraction) in an arithmetic expression. The results are unfloatd if need be at the time of deposition. Fixed integer arithmetic is performed on all items which are status, Hollerith, bits, and bytes. The subscripts are used as 15 bit items with complement arithmetic when used as subscripts, but when used as variables they are floated as any other integer.

When the Interpreter is sent to a procedure, it maintains a level count so that subscripts and interim data won't be destroyed. At present procedures can be stacked four deep.

### OUTPUT

1. The object program's tables (answers).
2. If requested:
  - A. All effective GOTOs.
  - B. All SETs.

#### Output Format

- |         |               |  |
|---------|---------------|--|
| A) GOTO | 1st Word: MZE | In the address is the two's complement of the present location in ILT.<br>In the decrement is the two's complement of the ILT location to which control is being sent. |
| B) SET  | 1st Word: PZE | In the address the two's complement is the present location in ILT.<br>In the decrement is the two's complement of the relative location of the subscript modulo 26.   |

Output Format Cont.

2nd Word: PZE	The value of the item being set in binary.
3rd Word: PZE	Hollerith name of the item being set. If it is a subscript this location has a zero.
4th Word: PZE	Signed value of subscript in binary.

MESSAGE PRINTOUTS

The printouts from JOLIZ consist of error messages only.

A. The basic message form is:

1. Identification EF##
2. Relative location of the error in ILT.
3. If the error is a floating point overflow or underflow, its location in the Interpreter plus one is given in octal, otherwise the area is blank.

Example  $\Delta \Delta \Delta \Delta \Delta \Delta \Delta$  EF## $\Delta \Delta \Delta \Delta \Delta \Delta \Delta$   
IDENT ILT LOC. OV.UN. LOC.

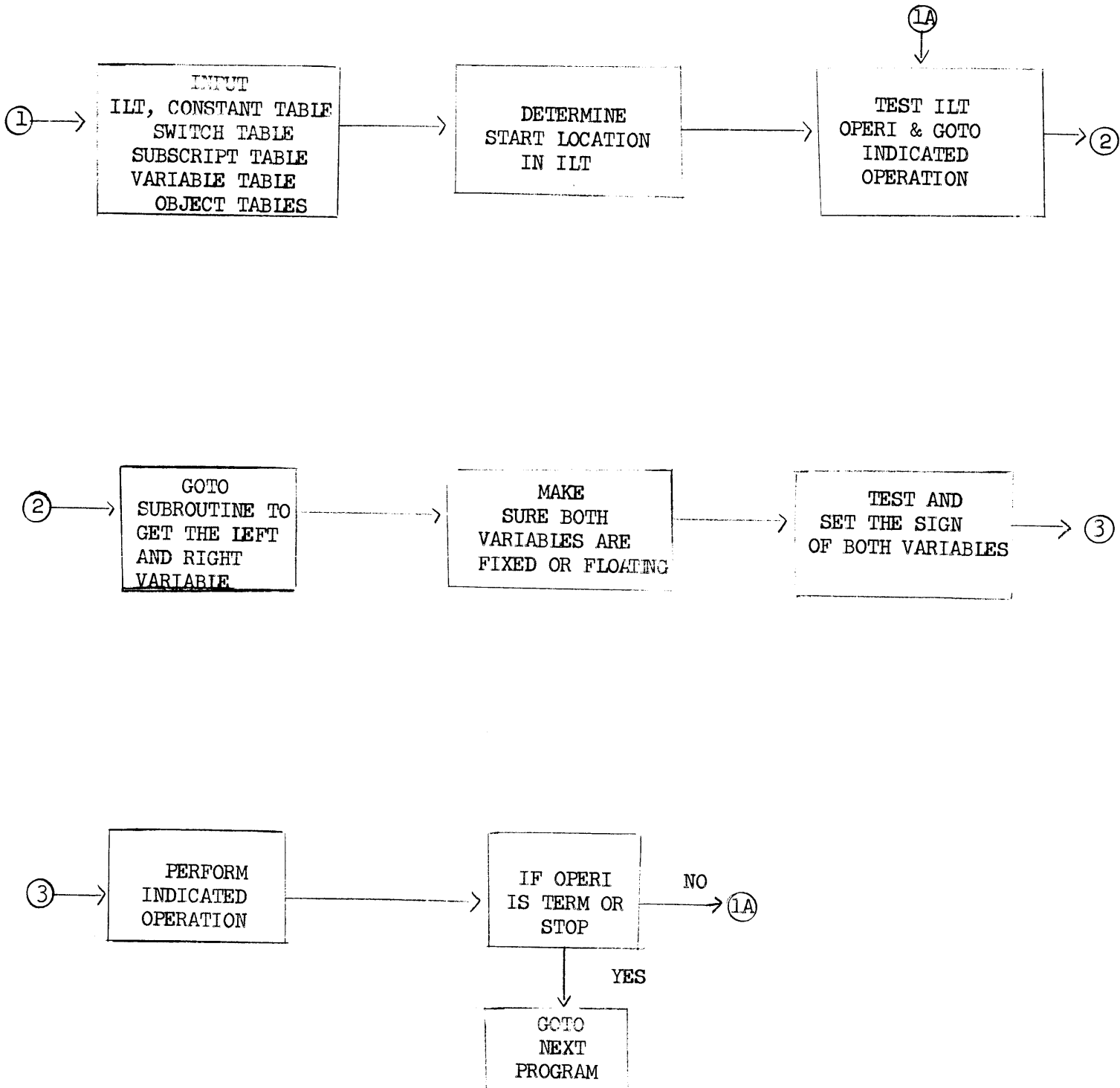
B. Messages:

1. EF1: Floating add, subtract, multiply; either an overflow or underflow occurred in the MQ. Control is returned to the error location plus one.
2. EF2: Division by zero; MQ is destroyed and control is returned to the error location plus one.
3. EF3: Floating add, subtract, multiply; either an overflow or underflow occurred in the MQ or AC. Control is returned to the error location plus one.
4. EF4: Negative argument to logarithm subroutine control is returned to the next ILT entry.
5. EF5: Argument too large for the exponential subroutine. Control is returned to the next ILT entry.

B. Messages: Cont.

6. EF6: Floating add, subtract, multiply; either an overflow or underflow occurred in the MQ or AC. Control is returned to the error location plus one.
7. EF7: Floating add, subtract, multiply; either an overflow or underflow occurred in the MQ or AC. Control is returned to the error location plus one.
8. EF8: The object program is setting a status, Hollerith, bit or byte information into a fixed mixed number. Control is returned to the next ILL entry.
9. EF9: Floating divide; either an overflow or underflow occurred in the MQ. Control is returned to the error location plus one.
10. EF10: Floating divide; either an overflow or underflow occurred in the AQ. Control is returned to the error location plus one.
11. EF11: Floating divide; either an overflow or underflow occurred in the AC or MQ. Control is returned to the error location plus one.
12. EF12: The object program is setting a non-status, Hollerith, bit or byte information into a status or hollerith item. Control is returned to the next ILL entry.
13. EF13: Floating divide; either an overflow or underflow occurred in the AC or MQ. Control is returned to the error location plus one.
14. EF14: The object program is setting a status, Hollerith, bit or byte information into a floating number. Control is returned to the next ILL entry.
15. EF16: The number of entries of each table of the ENT instruction are not equal. Control is returned to the next ILL entry.
16. EF17: The left form of this ILL entry was ENT and the right form was not. Control is returned to the next ILL entry.
17. EF18: The operator of this ILL entry is not a set with the form of ENT. Control is returned to the next ILL entry.

FLOW DIAGRAM



A listing of the program symbolic deck will be issued as the first supplement to this document (FN-LO-204,S-1).

Distribution:

SDC (Lodi)

Division Staff (1 ea.)  
Programming Branch Staff (1 ea.)  
Program Production Group (1 ea.)  
Program Design Group - M. Mineart (20)  
Program Requirements Group - F. Diaz (5)  
CUSS Project - J. I. Schwartz (10)

SDC (Santa Monica)

J. D. Madden  
R. Bosak  
J. Matousek  
B. Morriss  
G. Dobbs (10)  
E. Gordon  
C. M. Lawson  
D. E. Henley  
G. Jacobs

IEC

Standard Distribution (35)

:lb