

AD671491

BR-90 ASSEMBLY PROGRAM - BRASS

JUNE 1968

J. A. Terrasi

and

J. C. Penney

Prepared for

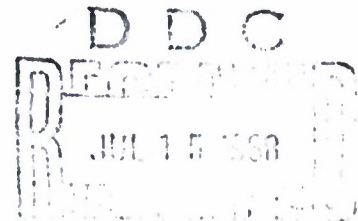
DIRECTORATE OF PLANNING AND TECHNOLOGY

ELECTRONIC SYSTEMS DIVISION

AIR FORCE SYSTEMS COMMAND

UNITED STATES AIR FORCE

L. G. Hanscom Field, Bedford, Massachusetts



This document has been approved for public release and sale; its distribution is unlimited.

Project 512V

Prepared by

THE MITRE CORPORATION

Bedford, Massachusetts

Contract AF19(628)-5165

Reproduced by the
CLEARINGHOUSE
for Federal Scientific & Technical
Information Springfield Va. 22151

ESD ACCESSION LIST

ESTI Call No. AL 61290

Copy No. 1 of 2 cys.

MTR-597

*10P2
ESVF*

ESD-TR-68-132

ESD RECORD COPY

RETURN TO
SCIENTIFIC & TECHNICAL INFORMATION DIVISION
(ESTI), BUILDING 1211

BR-90 ASSEMBLY PROGRAM - BRASS

JUNE 1968

J. A. Terrasi

and

J. C. Penney

Prepared for

DIRECTORATE OF PLANNING AND TECHNOLOGY

ELECTRONIC SYSTEMS DIVISION

AIR FORCE SYSTEMS COMMAND

UNITED STATES AIR FORCE

L. G. Hanscom Field, Bedford, Massachusetts



This document has been approved for public release and sale; its distribution is unlimited.

Project 512V
Prepared by
THE MITRE CORPORATION
Bedford, Massachusetts
Contract AF19(628)-5165

AD671491

ESD-TR-68-132
ESTI FILE COPY

When U.S. Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Do not return this copy. Retain or destroy.

BR-90 ASSEMBLY PROGRAM - BRASS

JUNE 1968

J. A. Terrasi

and

J. C. Penney

Prepared for

DIRECTORATE OF PLANNING AND TECHNOLOGY

ELECTRONIC SYSTEMS DIVISION

AIR FORCE SYSTEMS COMMAND

UNITED STATES AIR FORCE

L. G. Hanscom Field, Bedford, Massachusetts



This document has been approved for public release and sale; its distribution is unlimited.

Project 512V

Prepared by

THE MITRE CORPORATION

Bedford, Massachusetts

Contract AF19(628)-5165

FOREWORD

The work reported in this document was performed by The MITRE Corporation, Bedford, Massachusetts, for the Directorate of Planning and Technology, Electronic Systems Division, of the Air Force Systems Command under Contract AF 19(628)-5165.

REVIEW AND APPROVAL

This technical report has been reviewed and is approved.

WILLIAM F. HEISLER, COL, USAF
Chief, Command Systems Division
Directorate of Planning & Technology

ABSTRACT

This document describes the features, user procedures, and program/coding specifications for the BR-90 Assembly Program (BRASS) which is operational on an IBM 1410 computer.

TABLE OF CONTENTS

		<u>Page</u>
SECTION I	BACKGROUND	1
SECTION II	USER PROCEDURES	2
	2.0 INTRODUCTION	2
	2.1 INPUT CARD FORMATS	2
	2.1.1 Title Card	2
	2.1.2 Instruction Card	2
	2.1.3 Label	2
	2.1.4 Operation Code	3
	2.1.5 Address Option	3
	2.1.6 Operand/Operand Address	3
	2.1.7 Comments	3
	2.2 COMMENT CARD	3
	2.3 END CARD	4
	2.4 OBJECT DATA CARD	4
	2.5 OBJECT END CARD	4
	2.6 ASSEMBLER ERROR INDICATORS	4
	2.7 OPERATING PROCEDURES	5
	APPENDIX A LEGAL VALUES	7
	APPENDIX B PSEUDO OPERATIONS	13
	APPENDIX C SAMPLE ASSEMBLY	17
SECTION III	PROGRAM/CODING SPECIFICATIONS	22
	3.0 PROGRAM DESCRIPTION	22
	3.1 ROUTINE DESCRIPTION	24
	3.2 FLOW CHARTS	25

SECTION I

BACKGROUND

BRASS, a BR-90 assembly program, was designed for use at the AFICCS Support Facility specifically to provide a vehicle for experimental program development under the Display Console Technology Task. Outstanding features of the assembler are the following: 1) tape (not disk) oriented; 2) AFICCS independent; and 3) operational on third-generation computer systems (emulation).

During the BRASS implementation phase, the standard BR-90 Normal Mode Control Program (N-mode) was used as part of the test package and the following highlights were noted: 1) N-mode consists of 4000 source statements with 800 user-labels; and 2) BRASS assembled N-mode in 12 minutes using 1/5 of the allowable user-label area for N-mode labels.

SECTION II
USER PROCEDURES

2.0 INTRODUCTION

BRASS, the BR-90 (AN/FYQ-45) Assembly Program, is an IBM 1410 program which assembles BR-90 source statements into BR-90 executable code. Input is fixed format, one card per source statement; output is to cards, each card containing a maximum of 34 BR-90 words (in BCD format). The assembly is a two-pass operation using tape as intermediate storage.

2.1 INPUT CARD FORMATS

A BRASS input deck is defined as a TITLE card followed by one or more instruction and/or comment cards terminated by an END card.

2.1.1 TITLE CARD

The TITLE card, the mandatory first card in the deck, has the following format:

Columns	1-6:	program identity
	7-11:	TITLE
	29-78:	remarks

where program identity is an optional 1 through 6 character identification.

Program identity and remarks will be printed in the header line of each page of the assembly listing. In addition, the program identity will be punched in columns 75 through 80 of each object card.

2.1.2 INSTRUCTION CARD

Each instruction card has the following format:

Columns	1-6:	label
	7-10:	operation code
	11-12:	address option
	14-28:	operand/operand address
	29-80:	comments

2.1.3 LABEL

A label consists from 1 to 6 characters left-justified in the label field. At least one character of the label must be

non-numeric and the label must not be defined more than once. Note that plus signs, minus signs, or commas should not be used in labels since these symbols are used within an expression as delimiters. Note also that an asterisk (*) should be avoided as the first label character since it would indicate a comment card.

2.1.4 OPERATION CODE

An operation code must be one of the 16 machine codes or one of the 12 psuedo-operations defined in Appendix A. The entry must be left-justified.

2.1.5 ADDRESS OPTION

For machine codes, an address option must be one of the 9 options listed in Appendix A. For psuedo-operations, the address option field must be blank.

2.1.6 OPERAND/OPERAND ADDRESS

The operand/operand address may be one of the following:

- blank;
- an octal number;
- a decimal number;
- a label; or
- an expression, a combination of labels and/or decimal numbers.

The entry must be left-justified.

2.1.7 COMMENTS

This field is ignored by the assembler but is carried on the assembly listing.

2.2 COMMENT CARD

Each comment card has the following format:

Column 1: *
Columns 2-80: remarks

The comment card is ignored by the assembly processor, but is carried on the assembly listing.

2.3 END CARD

The END card has the following format:

columns 7-9: END
columns 14-28: location of first executable instruction.

Columns 14-28 may be blank or may contain a decimal number, a label, or an expression. A non-blank entry must be left-justified. If this field is blank, the location of the first executable instruction will be assumed to be the lower limit of BR-90 user core defined as 0062₈.

2.4 OBJECT DATA CARD

The Object Data Card has the following format:

columns 1-4: octal location of the first BR-90 word punched in the card.
columns 5-6: decimal number of BR-90 words punched in this card ($01 \leq \text{COUNT} \leq 34$).
columns 7-74: consecutive BR-90 words where a BR-90 word is represented by two BCD characters.
columns 75-80: program identity.

Object Data cards are filled with sequential BR-90 word representations. Any break in the sequential flow of the BR-90 source program will cause a new object data card to be generated. Thus, the object data cards are independent of one another, and the order of the object deck is not significant.

2.5 OBJECT END CARD

The Object End card has the following format:

columns 1-3: END.
columns 4-7: octal location of the first executable instruction of the BR-90 program.
columns 75-80: program identity

2.6 ASSEMBLER ERROR INDICATORS

A source statement may generate one or more assembly errors. These error conditions are noted by character error flag codes carried on the assembly listing to the right of the source statement image. These error flags and their associated meanings are as follows:

<u>Error Flag</u>	<u>Meaning</u>
U	Illegal label field
M	Multiple-defined label
A	Illegal operation and/or address option field
O	Illegal operand/operand address field
L	Location counter overflow ($>7777_8$)

Error types U, M, A, and O will cause a BR-90 "all zeros" word to be generated. Error type L is only a warning indication noting that overflow has occurred and that the location counter was reset to 0062_8 at the first instance of overflow.

2.7 OPERATING PROCEDURES

BRASS, a non-AFICCS dependent program, may be called by one of two methods:

- a) by COPS within AFICCS, if SAP assembled the BRASS AUTOCODER source statements; or
- b) by an object deck loader, if PR108 assembled the BRASS AUTOCODER source statements.

BRASS non-AFICCS hardware requirements are as follows:

- a) minimum of 20K 1410 core;
- b) one tape drive;
- c) card reader/punch;
- d) console typewriter; and
- e) printer.

To assemble BR-90 source programs using BRASS:

- a) mount a blank tape on channel 1, drive 2;

- b) ready card punch;
- c) if in AFICCS, load BR-90 source program deck(s) in reader and call BRASS via console typewriter.

If non-AFICCS, load object deck loader, BRASS object deck, and BR-90 source program deck(s) in reader, and bootstrap the loader.
- d) enter date of BR-90 assembly after typewriter cue message;
- e) upon completion of the assembly process, AFICCS may be reloaded by placing the AFICCS system tape on channel 1, drive 1 and pressing COMPUTER RESET and START.

BRASS accepts batch assemblies. If no END card is used, the assembler generates an object end card and a dummy "end card" printer line with the "A" flag set.

APPENDIX A
LEGAL VALUES

OPERATIONS

The following are the legal operations and their associated meanings.

<u>MNEMONIC</u>	<u>OPERATION</u>
BU	Branch unconditional
ST	Store X-register
XF	External function
IO	Input/output
BV	Branch on overflow
LD	Load X-register
DS	Display
XM	Extract to memory
BL	Branch link
EX	Extract to X
AD	Add to X
SU	Subtract from X
BZ	Branch on X equal to zero
MG	Merge to X
SH	Shift X right
MM	Merge to memory

ADDRESS OPTIONS

OPERATIONS

For legal operations, with the exception of the SH operation, the following are the legal address options and their associated meanings.

<u>ADDRESS OPTION</u>	<u>DESCRIPTION</u>
DS	Direct from scratchpad
IS	Indirect from scratchpad
DM	Direct from memory
IM	Indirect from memory
DR	Direct relative
IR	Indirect relative

For the SH operation, the following are the legal address options and their associate meanings.

<u>ADDRESS OPTION</u>	<u>DESCRIPTION</u>
OL	Open logical right shift
CL	Closed logical right shift
ON	Open numeric right shift

OPERAND/OPERAND ADDRESS

For legal operations, with the exception of the SH operation, the operand/operand address must be either blank or an EXPRESSION depending upon the address option selected.

For address options DM or IM, the operand/operand address must be blank.

For address options DS or IS, the operand/operand address must be an EXPRESSION whose octal value is in the range 0 through 77.

For address options DR or IR, the operand/operand address must be an EXPRESSION whose value is within 77_8 positions of the location of the instruction; that is, the value [EXPRESSION - 1 - location of the instruction] must be between 0 through 77_8 . This value is a relative address modifier and is placed in the actual machine instruction.

For the SH operation, the operand/operand address must be an unsigned decimal number whose range is 1 through 15.

An EXPRESSION is defined as a combination of labels and decimal numbers. If S means label and I means a decimal number, the following are legal EXPRESSIONS.

I
+I
 S
+S
 I+S
 S+I
+I+S
+S+I
+S+S
+I+I

PSEUDO OPERATIONS

The following are the legal pseudo operations and their associated meanings. Appendix B contains more detail on the pseudo operations.

<u>MNEMONIC</u>	<u>PSEUDO OPERATION</u>
ORG	Origin
RES	Reserve
PZE	Plus Zero
Blank Field	Same as PZE
OCT	Octal
EQU	Equals
EQUB	Equals octal
VCO	Vector coordinates
SCO	Symbol coordinates
CIR	Circle
BCIC	Binary coded information console
PAGE	Page eject

PREDEFINED LABELS

The assembler defines a number of labels prior to any assembly. These labels may be used by any source BR-90 program. The user should be aware of what labels are predefined since the error M will be set if a user attempts to define a predefined label in his program. The following are the predefined labels and their associated octal values.

<u>LABEL</u>	<u>OCTAL VALUE</u>
*	Current value of the location counter
**	0
\$EOM	37
\$STOP	0
\$START	1
\$P1	2
\$P2	3
\$P3	4
\$P4	5
\$P5	6
\$P6	7
\$I1	10
\$I2	11
\$I3	12
\$I4	13
\$I5	14
\$I6	15
\$I7	16
\$I8	17

<u>LABEL</u>	<u>OCTAL VALUE</u>
\$NIA	20
\$LG	21
\$LH	22
\$T1	23
\$T2	24
\$T3	25
\$T4	26
\$T5	27
\$T6	30
\$T7	31
\$T8	32
\$T9	33
\$T10	34
\$T11	35
\$T12	36
\$T13	37
\$T14	40
\$T15	41
\$T16	42
\$T17	43
\$T18	44
\$T19	45
\$T20	46
\$T21	47
\$T22	50
\$T23	51
\$T24	52
\$T25	53
\$T26	54
\$T27	55

<u>LABEL</u>	<u>OCTAL VALUE</u>
\$T28	56
\$T29	57
\$T30	60
\$T31	61
\$T32	62
\$T33	63
\$T34	64
\$T35	65
\$T36	66
\$T37	67
\$T38	70
\$T39	71
\$T40	72
\$MTWO	73
\$MONE	74
\$TWO	75
\$ONE	76
\$ZERO	77
\$POW	100
\$INA	104
\$INB	110
\$DIS	114

APPENDIX B

PSEUDO OPERATIONS

ORIGIN - (ORG)

The ORG pseudo op causes the location counter to be set to the value given in the operand/operand address field. The operand/operand address field must contain an expression whose value is in the range 0_8 to 7777_8 . Labels, if used in the expression, must be previously defined. ORG may not be a labelled statement.

RESERVE - (RES)

The RES pseudo op causes the assembler to reserve one or more consecutive BR-90 words within the object program commencing with the current value of the location counter. The operand/operand address field must contain an unsigned decimal number within the range 1 to 4095. RES may be labelled, in which case, the label refers to the first word of the block. A reserve statement resulting in a location counter overflow is considered illegal.

PLUS ZERO - (PZE)

The PZE pseudo op causes the assembler to reserve one BR-90 word whose value is equal to the octal value of the expression in the operand/operand address field. A symbol appearing in the expression need not be defined prior to the PZE statement. A blank operation field is equivalent to the PZE pseudo op. PZE may be a labelled statement.

OCTAL - (OCT)

The OCT pseudo op causes the assembler to reserve one BR-90 word whose value is equivalent to the signed or unsigned octal number in the operand/operand address field. The octal number must be between -7777_8 and $+7777_8$. OCT may be a labelled statement.

EQUALS - (EQU)

The EQUALS pseudo op causes the label in the label field to be equated to the octal value of the expression in the operand/operand address field. A symbol appearing in the expression must be defined prior to the EQU statement. EQU must be a labelled statement.

VECTOR COORDINATES - (VCO)

The VCO pseudo op causes the assembler to generate two BR-90 words containing the x,y coordinates as defined by the EXPRESSIONS in the operand/operand address field. The operand/operand address field may be one of two formats:

- a) EXPRESSION, EXPRESSION;
- b) EXPRESSION, EXPRESSION, COMBINATION where COMBINATION may be the letters U, B, W, E in any order. These letter codes mean the following:

U = generate vector;

B = blink vector;

W = wide vector;

E = end of string.

A label in an EXPRESSION need not be defined prior to the VCO pseudo op. VCO may be a labelled statement.

SYMBOL COORDINATES - (SCO)

The SCO pseudo op causes the assembler to generate two BR-90 words containing the x,y coordinates as defined by the EXPRESSIONS in the operand/operand address field. The format of the operand/operand address field is EXPRESSION, EXPRESSION. A label in an EXPRESSION need not be defined prior to the SCO pseudo op. SCO may be a labelled statement.

CIRCLE - (CIR)

The CIR pseudo op causes the assembler to generate one BR-90 word containing information necessary to generate a circle. The operand/operand address field may be one of two formats:

- a) EXPRESSION;
- b) EXPRESSION, COMBINATION

where COMBINATION may be the letters W, B, N in any order. The meaning of these letter codes is as follows:

W = wide circle,
B = blink circle;
N = blank display.

A label in the EXPRESSION need not be defined prior to the CIR pseudo op. CIR may be a labelled statement.

BINARY CODED INFORMATION CONSOLE - (BCIC)

The BCIC pseudo op causes the assembler to generate a number of BR-90 words containing the character representations defined in the operand/operand address field. The operand/operand address field may have one of three formats:

- a) XX...X\$
1-14 characters
- b) XX...X
15 characters
- c) XX....X\$COMBINATION

where \$ is the string terminator and COMBINATION may be the letters E, B, L, M, N in any order. The meanings of the letter codes are as follows:

E = end of string;
B = blink symbol;
L = large symbol;
M = generate marker;
N = blank symbol.

BCIC may be a labelled statement, and if labelled, the label refers to the first BR-90 word generated.

PAGE EJECT - (PAGE)

The PAGE pseudo op directs the assembler to generate no BR-90 words, but causes the printing portion of the assembler to start a new printed page. The PAGE pseudo op will be printed as the first line of the new page.

EQUALS OCTAL - (EQUB)

The EQUALS OCTAL pseudo op causes the label in the label field to be equated to the octal value of the signed or unsigned octal number in the operand/operand address field. The octal number must be between -7777_8 to $+7777_8$. The EQUB must be a labelled statement.

APPENDIX C

SAMPLE

BRASS ON-LINE RECORD OF RUN

R IDLOOP Ø2 IBM

I BRASS

R TYPE IN TODAYS DATE AS DD MMYY

I 26 FEB 68

R END OF RUN--TO LOAD AFICCS PLACE BOOTSTRAP SYSTEM TAPE ON 1 AND COMPUTER RE
SET AND START

R IDLOOP Ø2 IBM


```

TEST1 TITLE
1LABLEQU 100
*
2LABLEQU 1LABEL+10
*
OCT77 EQU8 +77
*
ORG +512
*
PZE 0
*
ORG ++10
ST DM
*
PZE $ONE
*
ST DS $START
*
ST OR ++45
*
RES 100
*
MIN71 OCT -71
*
VCO OCT77+1,2,E
*
SCO 5,1+1LABEL
*
CIR 15,M
*
BCIC TEST1,$EB
PAGE
*
*
* ERRORS
*
1LABLEPZE 0
99 PZF 0
POZ 0
ST DA 10
PZE +5000
ST OR ++200
OCT 1099
ORG A
PZE 1LABEL+1+2
ORG 4095
PZE $STOP
END MIN77

```

```

THIS IS AN EXAMPLE OF A BRASS ASSEMBLY
EQUATE
COMMENT CARD
COMMENT CARD
COMMENT CARD
COMMENT CARD
COMMENT CARD
INSTRUCTION
SYSTEM PRE-DEFINED LABEL
COMMENT CARD
COMMENT CARD
COMMENT CARD
COMMENT CARD
RESERVE
BLANK OPERATION=PZE
OCTAL
VECTOR COORDINATES
SYMBOL COORDINATES
CIRCLE
BINARY CODED INFORMATION CONSOLE
MULTI-DEFINED LABEL
ILLEGAL LABEL
ILLEGAL OPERATION
ILLEGAL ADDRESS MODIFIER
ILLEGAL O/D ADDRESS
LEGAL INSTRUCTION, BUT OVERFLOW SET ON
START EXECUTING AT MIN77 LOCATION

```

**

APPENDIX C (Continued)
BRASS Assembly Listing

26 FEB 68 PAGE 001

THIS IS AN EXAMPLE OF A BRASS ASSEMBLY

TEST1

LOC	O C T A L VALUE	LABEL	OP	ADD OPT	O/O ADDRESS	EQUATE	COMMENTS	ERRCR
0144		1 LABEL	EQU		100		COMMENT CARD	
0156		2 LABEL	EQU		1 LABEL+10		COMMENT CARD	
0077		OCT77	EQU8		+77	EQUAL TO OCTAL	COMMENT CARD	**
		*	ORG		+512	ORIGIN	COMMENT CARD	
1000	0000	*	PZE		0	PLUS ZERO	COMMENT CARD	
1013	0500	*	ORG ST	DM	**10	INSTRUCTION	COMMENT CARD	
1014	0076	*	PZE		\$CNE	SYSTEM PRE-DEFINED LABEL	COMMENT CARD	
1015	0401	*	ST	DS	\$START		COMMENT CARD	
1016	0554	*	ST	DR	**45		COMMENT CARD	
1017		*	RES		100	RESERVE	COMMENT CARD	
1163	0000	*			0	BLANK OPERATION=PZE	COMMENT CARD	
1164	7707	MIN71	OCT		-71	OCTAL	COMMENT CARD	
1165	0400	*	VCO		OCT77+1,2,E	VECTOR COORDINATES	COMMENT CARD	
1166	0011	*						
1167	0026	*	SCC		5,1+1 LABEL	SYMBOL COORDINATES	COMMENT CARD	
1170	0624	*						
1171	1710	*	GIR		15,W	CIRCLE	COMMENT CARD	
1172	2320	*	BCIC		TFST1.5EB	BINARY CODED INFORMATION CCNSCLE	COMMENT CARD	
1173	6520							
1174	2220							
1175	2320							
1176	0120							
1177	7321							

LOC	O C T A L VALUE	LABEL	OP	ADD OPT	C/O ADDRESS	COMMENTS	ERRCR
-----	-----------------	-------	----	---------	-------------	----------	-------

PAGE

*
*
*
ERRORS

1200	0000	1LABEL	PZE		0	MULTI-DEFINED LABEL	M
1201	0000	99	PZE		0	ILLEGAL LABEL	U
1202	0000		POZ		0	ILLEGAL OPERATION	A
1203	0000		ST	DA	10	ILLEGAL ADDRESS MODIFIER	A
1204	0000		PZE		+5000	ILLEGAL O/O ADDRESS	C
1205	0000		ST	DR	**200		C
1206	0000		OCT		1099		O
			DRG		A		C
1207	0000		PZE		1LABEL+1+2		O
			ORG		4C95		
7777	0000		PZE		\$STOP	LEGAL INSTRUCTION, BUT OVERFLOW SET ON	L
			END		MIN77	START EXECUTING AT MIN77 LOCATION	L

00009 = NUMBER OF FATAL ASSEMBLY ERRORS

APPENDIX C (Continued)

BRASS Object Code (80-80 List)

100001
1013055
116321
777701
END00062

415*
74 9 W6U 8T E S T I . /

TESTI
TESTI
TESTI
TESTI
TESTI

SECTION III
PROGRAM/CODING SPECIFICATIONS

3.0 PROGRAM DESCRIPTION

The BRASS assembler was designed for use in the 1410 AFICCS system or on a 1410 computer which has no AFICCS system and has the following minimum hardware requirements:

20K core memory;
card reader/punch;
one channel, 1 tape drive;
printer; and
console typewriter.

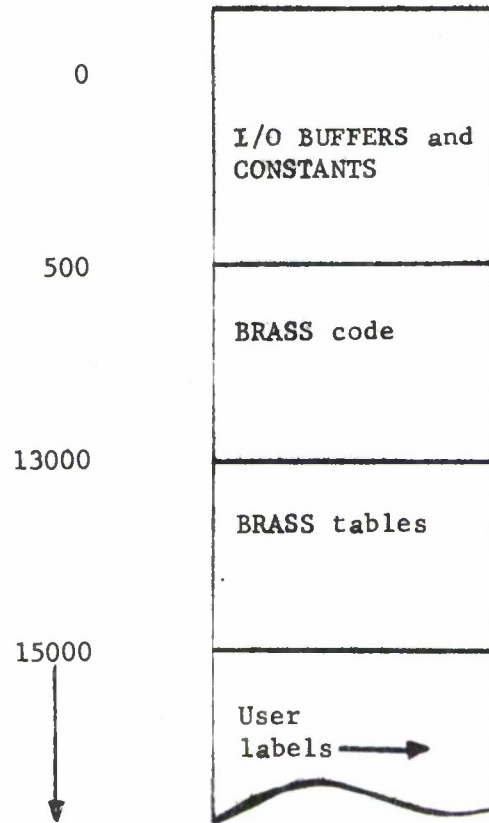
To enable BRASS to maintain this duality, the following programming conventions were used:

- a) origin at 500;
- b) no pre-defined group mark - word marks; and
- c) all I/O references imbedded within the program.

Because of these conventions, BRASS 1410 Autocoder source code may be assembled by either the SAP assembler or by the PR108 assembler. If assembled by SAP within AFICCS, BRASS will integrate on disk and will be assigned the COPS calling name BRASS. If assembled by PR108, a 1410 object deck will be punched which may be loaded into a 1410 computer by a standard bootstrap loader.

BRASS is a two-pass assembler. The first pass reads, partially assembles, and saves the BR-90 source code on tape. The second pass reads the stored instructions from tape and completes the assembly, prints the assembly listing and punches the BR-90 object card deck. BRASS has within it predefined system labels which may be used by any BR-90 source program.

BRASS's core map is as follows:



Lower core (0-499) is initialized immediately upon execution of the BRASS code. The user label table is built up from about 15000 and remains in core throughout the entire assembly process. Thus the limiting factor to the number of labels is the amount of core (up to 40K).

3.1 ROUTINE DESCRIPTION

TITLE: BR-90 Assembly Program (BRASS)

CLASS: Program

SIZE: 15,000 characters

ORIGIN: 500

DATE: 27 February 1968

FUNCTION: This program assembles BR-90 source code statements into executable BR-90 octal machine codes.

CALLING SEQUENCE: Within AFICCS, type in BRASS. Otherwise, load BRASS 1410/PR108 object deck with 1410 object deck loader.

INPUT: BR-90 source deck(s)

OUTPUT: Assembly listing(s) and BR-90 object deck(s)

NORMAL RETURN: At normal exit, BRASS types out cue message and halts. At such time, AFICCS may be reloaded by mounting the AFICCS system tape on channel 1, drive 1 and pressing COMPUTER RESET and START.

ERROR RETURN: Program halts on unrecoverable errors.

ERROR PRINTOUTS: Subroutines TAPECOMMON, READRPUNCH, and PRINTERR type out indicative hardware errors.
"SYMBOL TABLE FULL - DUMP 40K CORE" - if assembly labels fill core from 15000 to 40000.

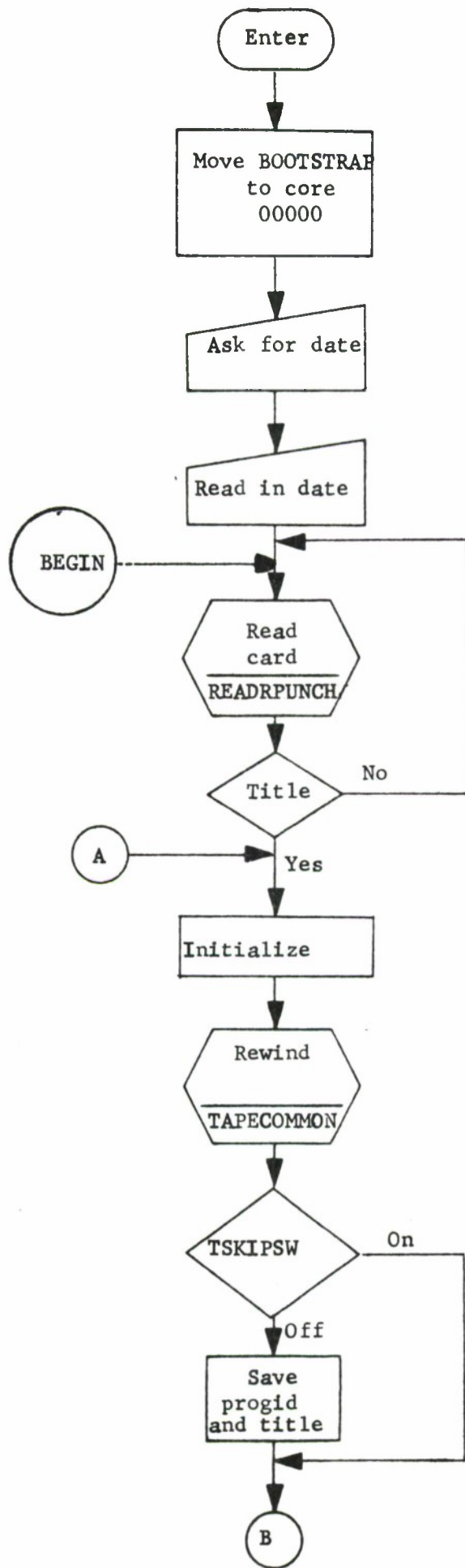
SUBROUTINES: TAPECOMMON and READRPUNCH are explained in PCS-AF-1, Volume 2, Part 2. All other significant subroutines of BRASS are flow-charted in the following sections.

ACCESSED SDA's: None.

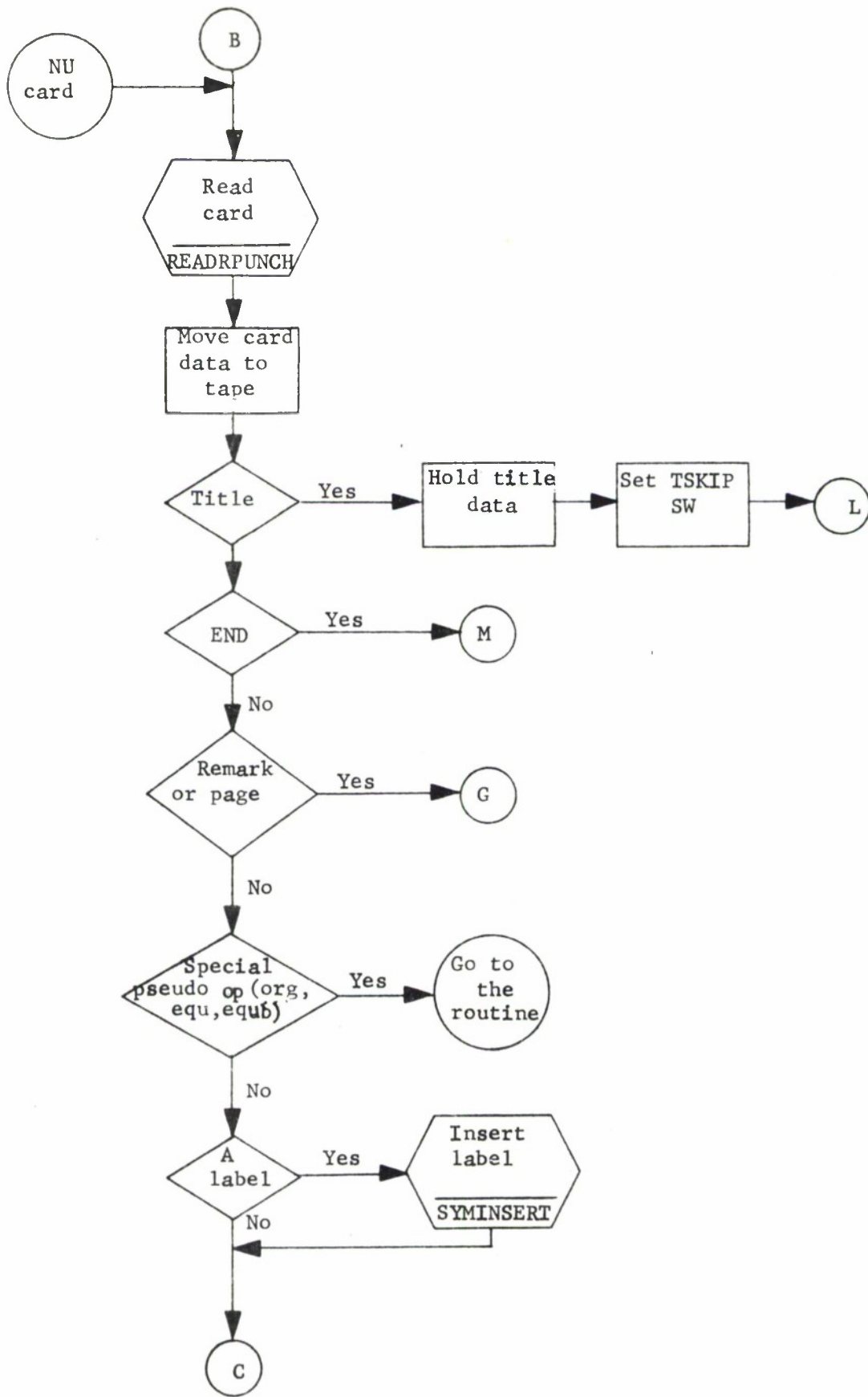
REMARKS: BRASS is programmed using 1410 Autocoder such that it could be assembled by either SAP or PR108. BRASS, therefore, is independent of AFICCS and may be used at a 1410 facility which has no AFICCS system.

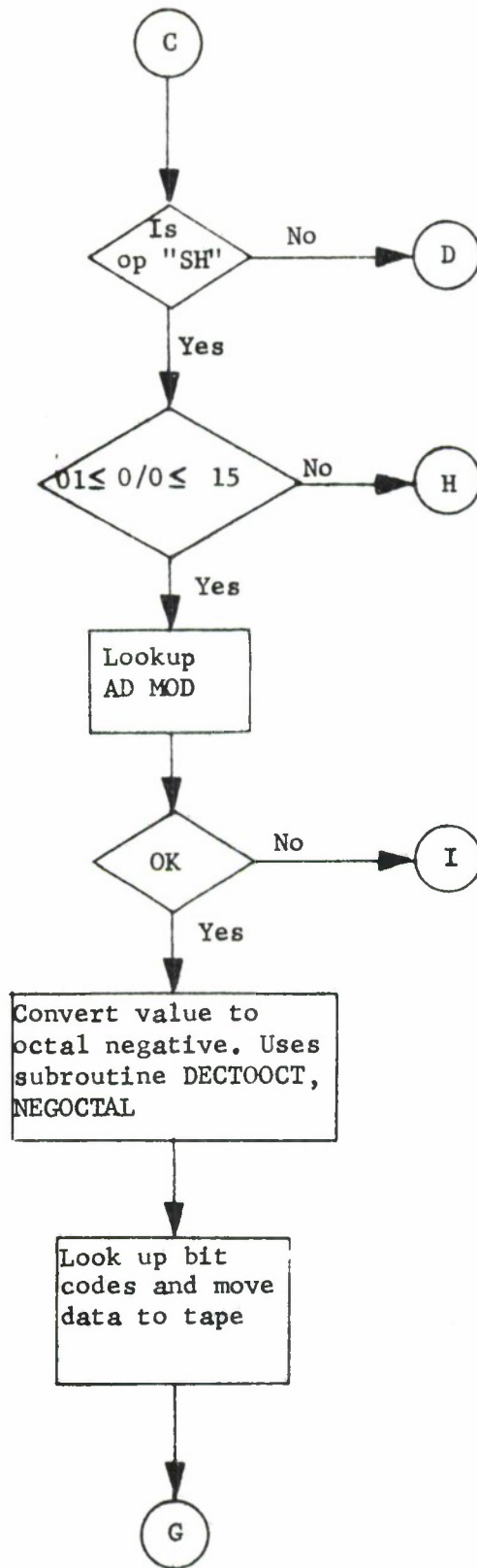
3.2 FLOW CHARTS

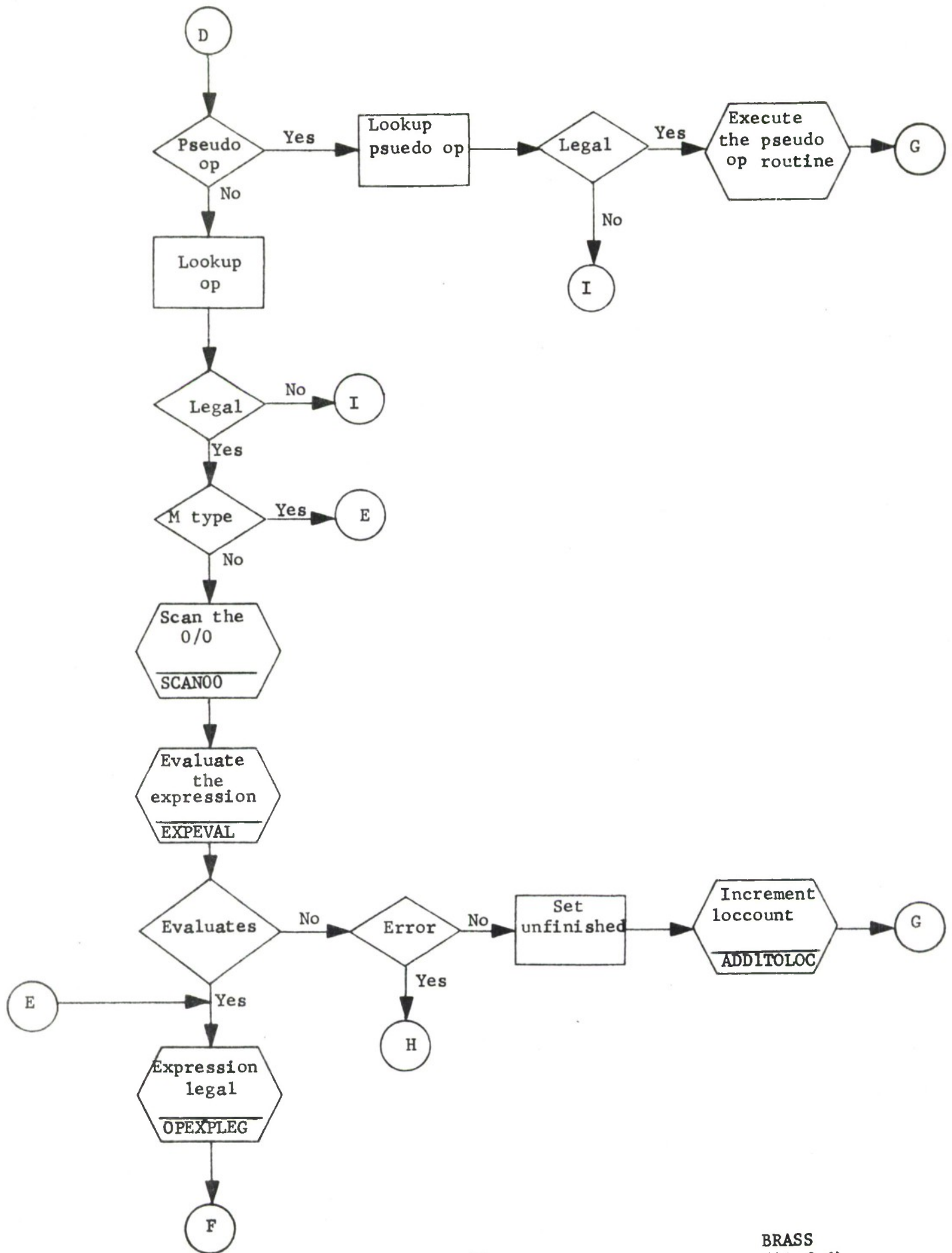
The following flow charts describe the program flow of the BRASS Assembly Program.

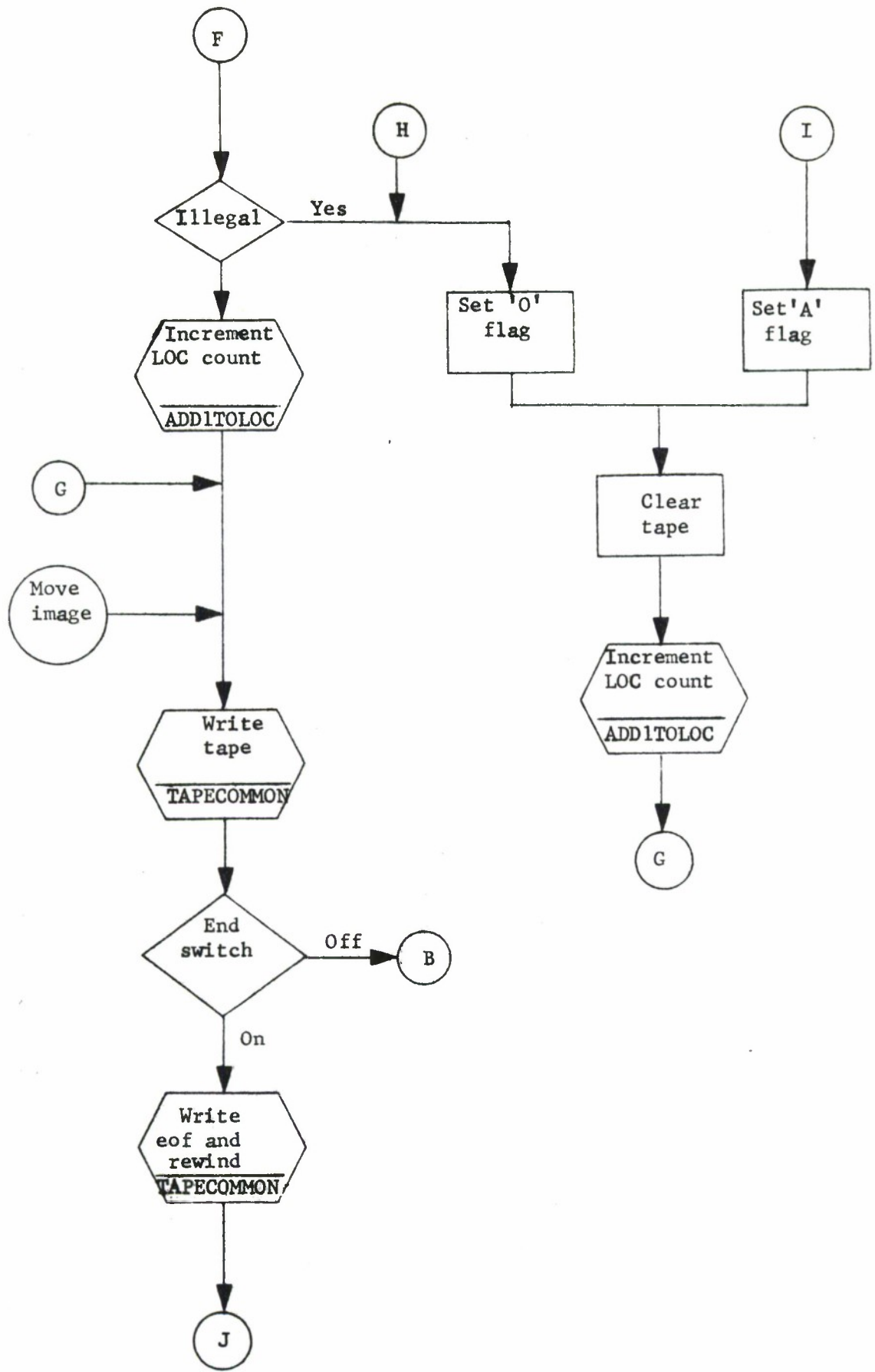


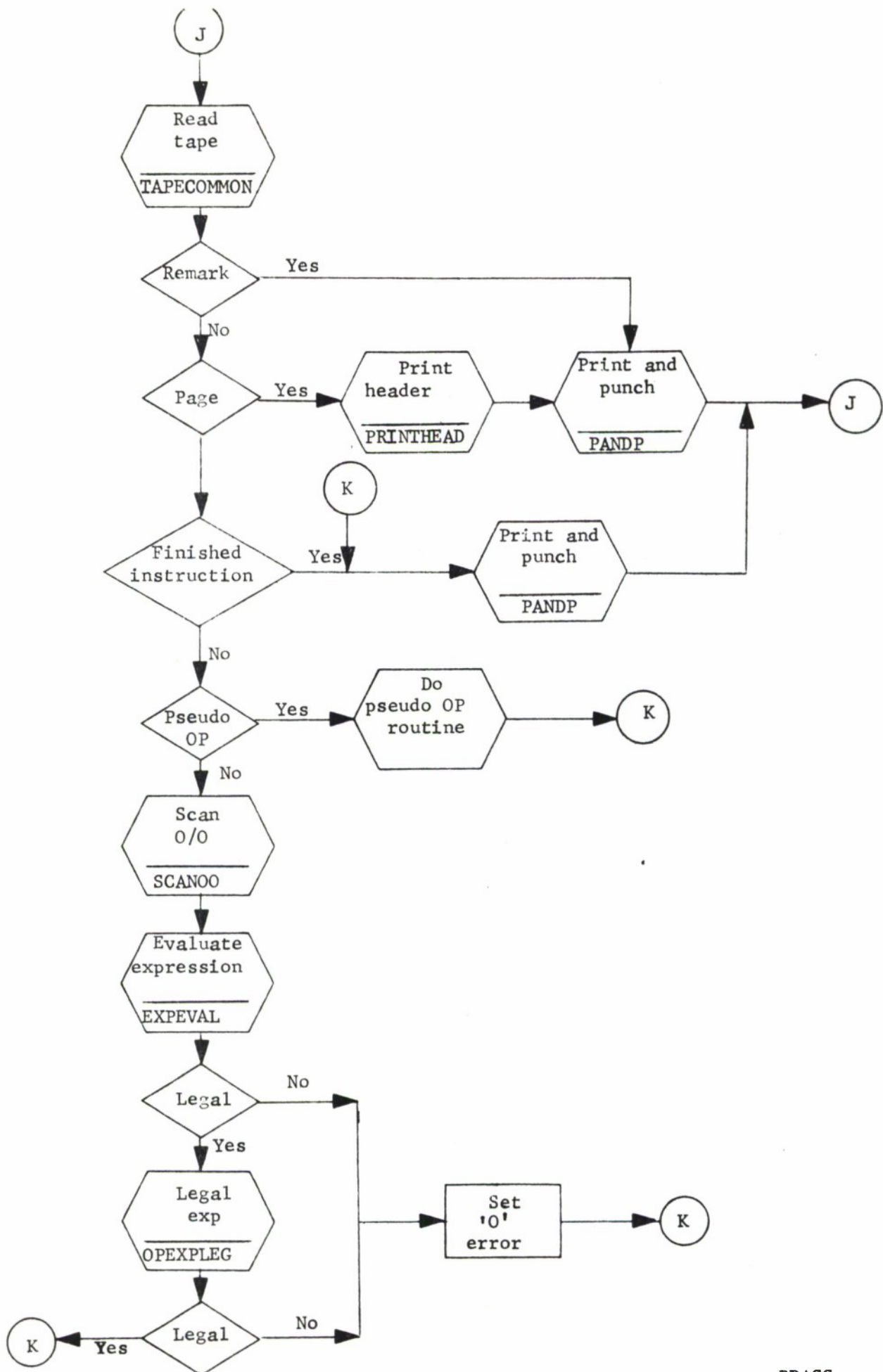
BRASS FLOW CHART
(1 of 6)

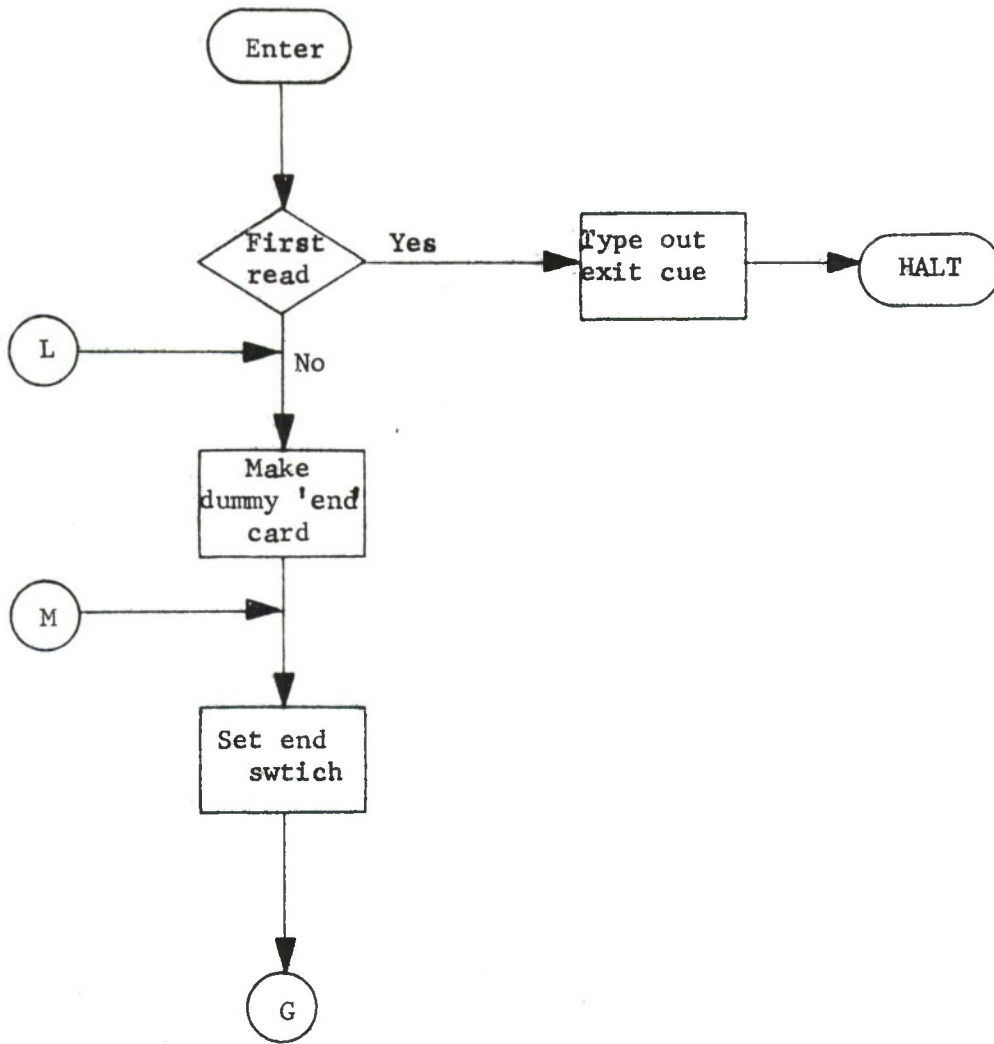


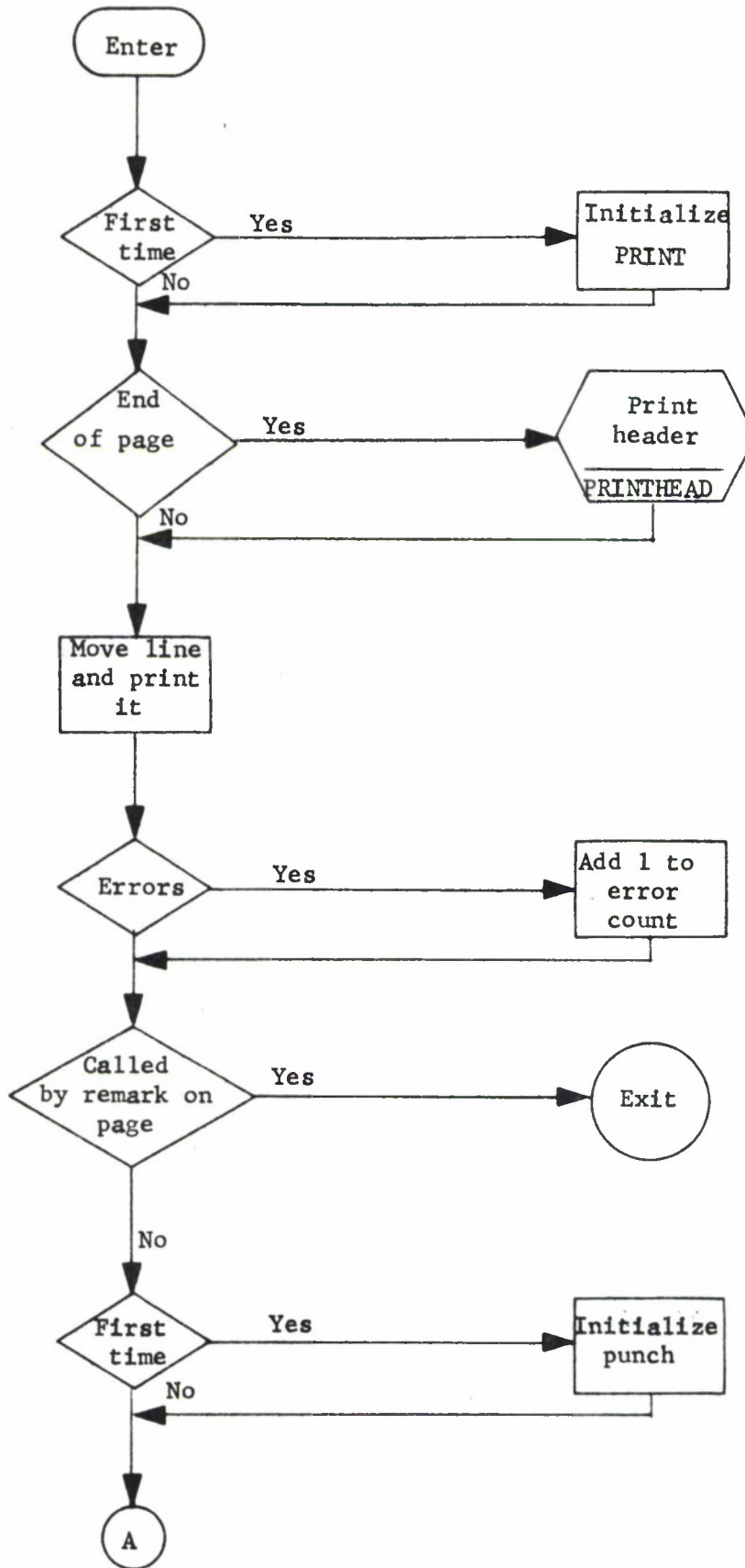


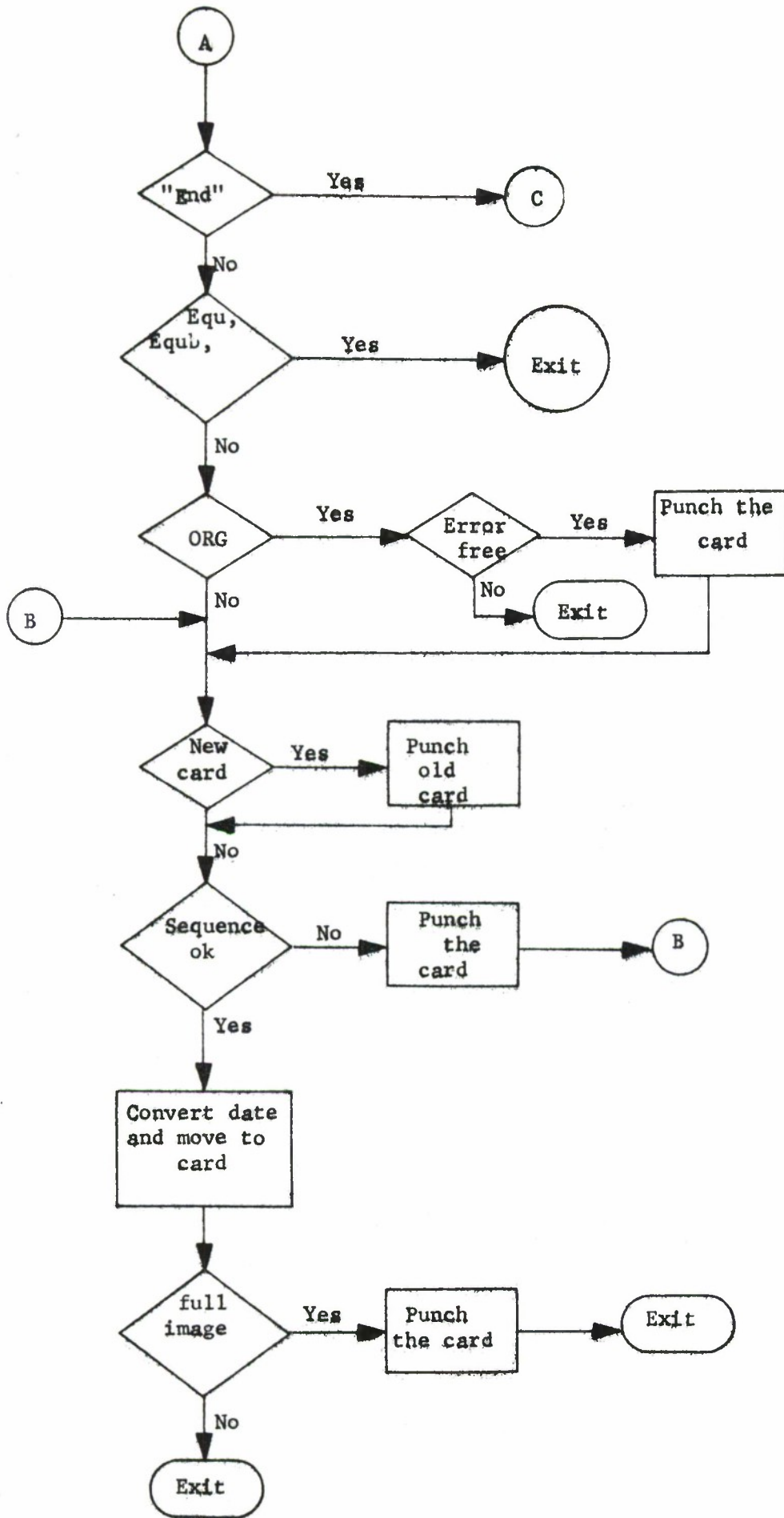


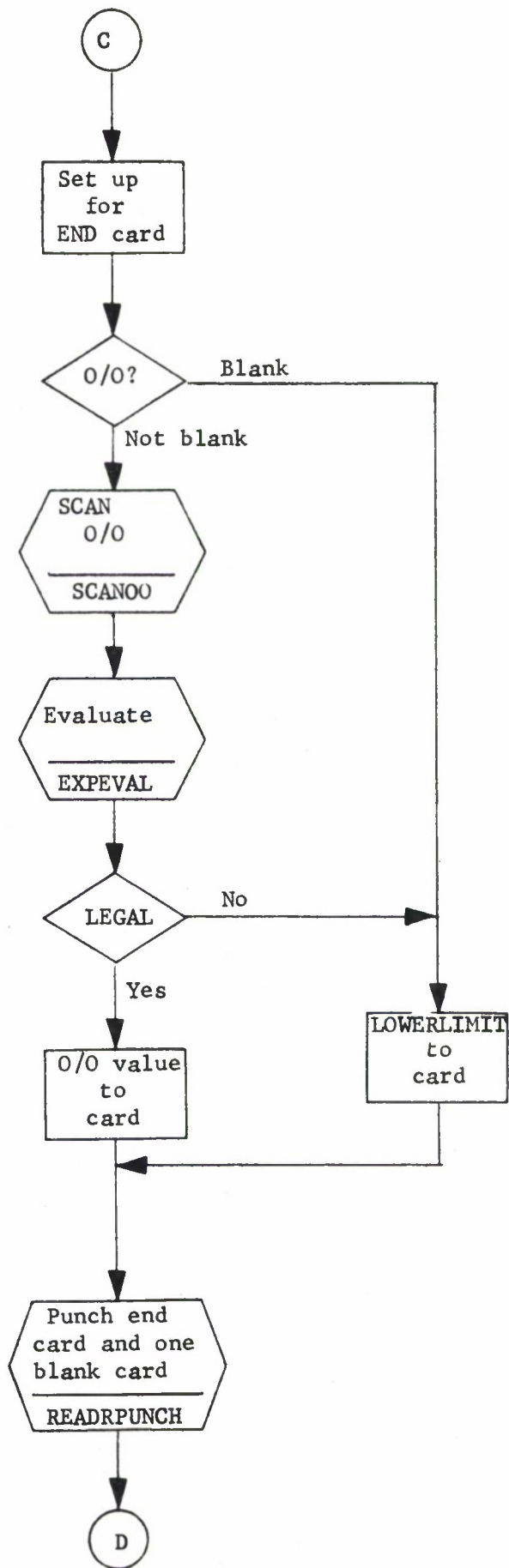


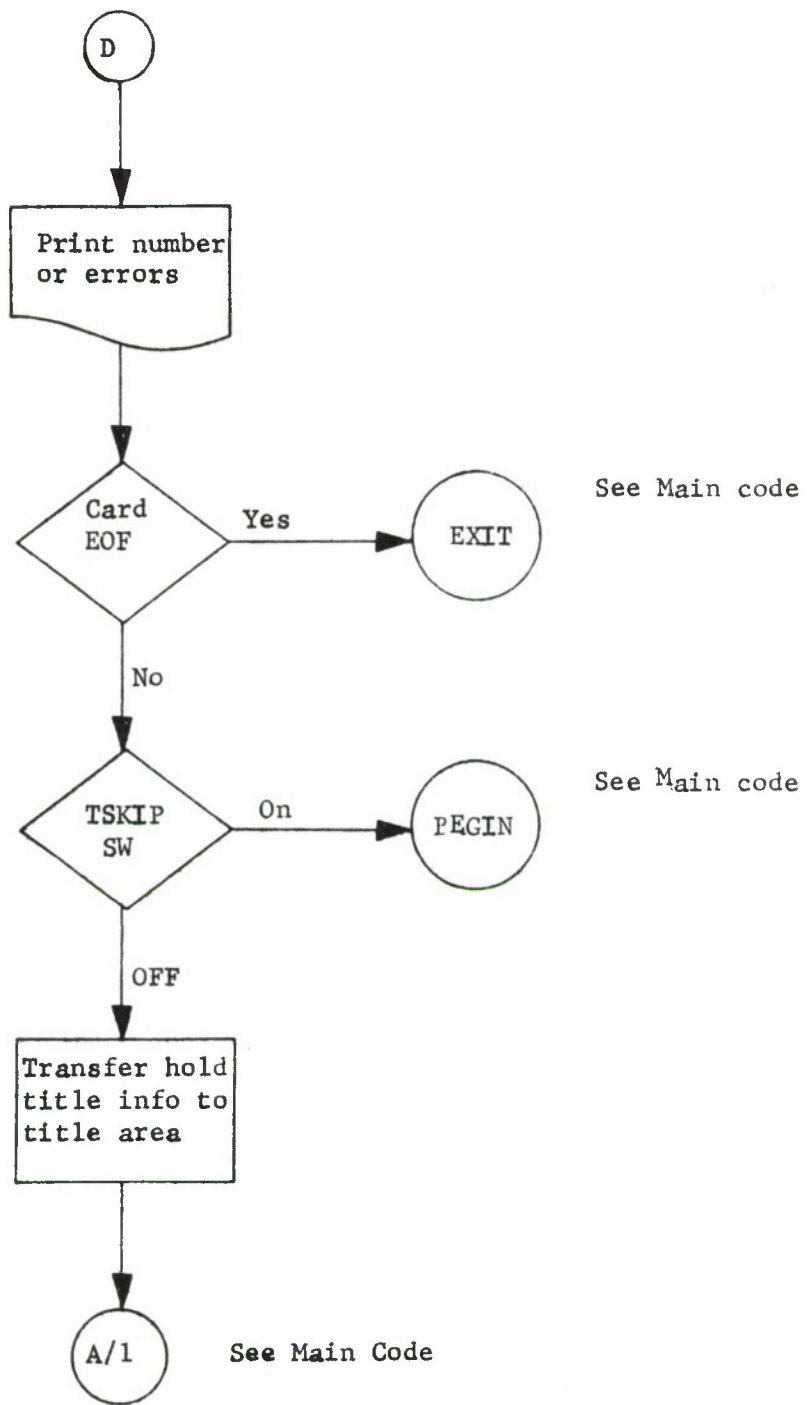




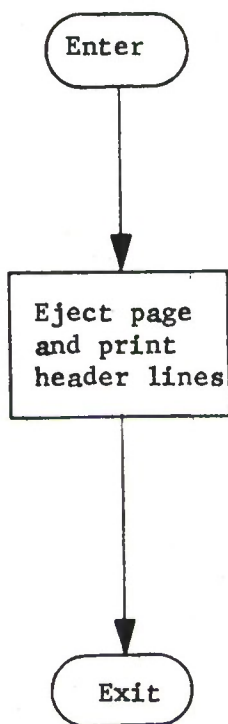


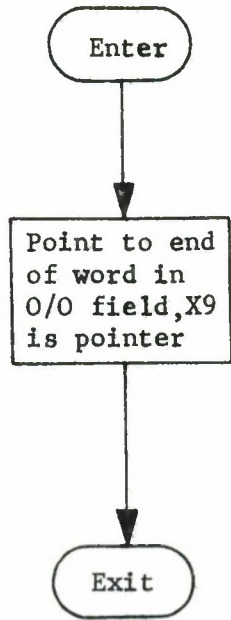


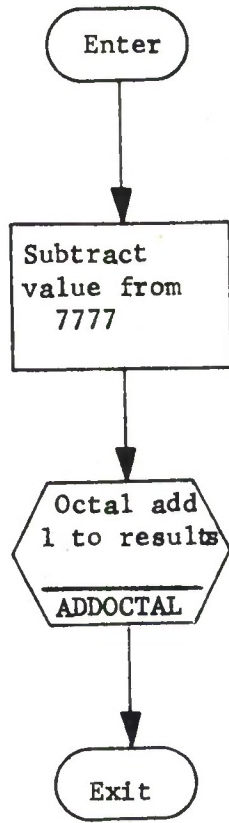


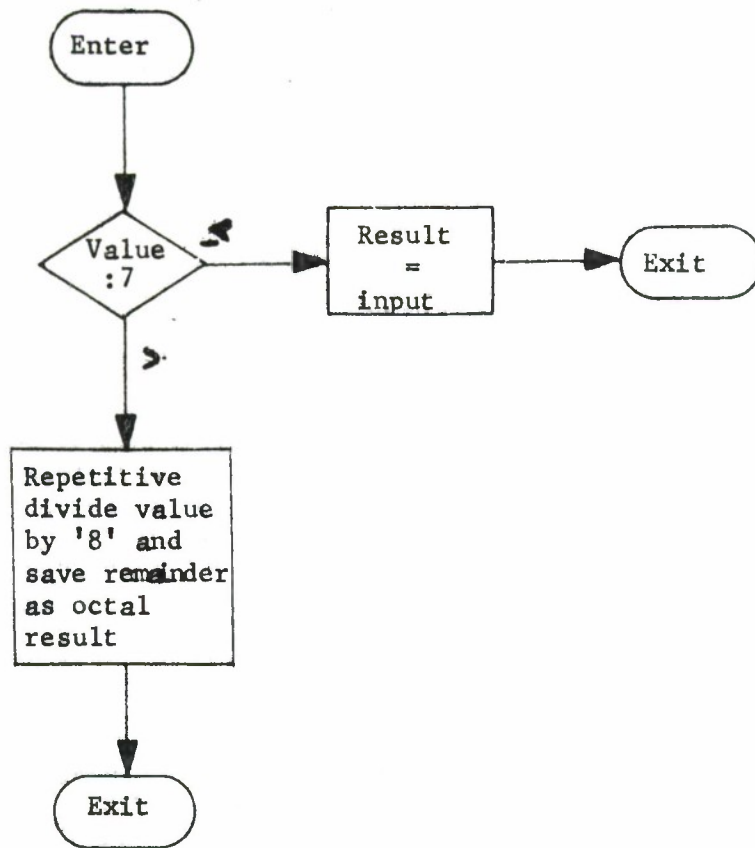


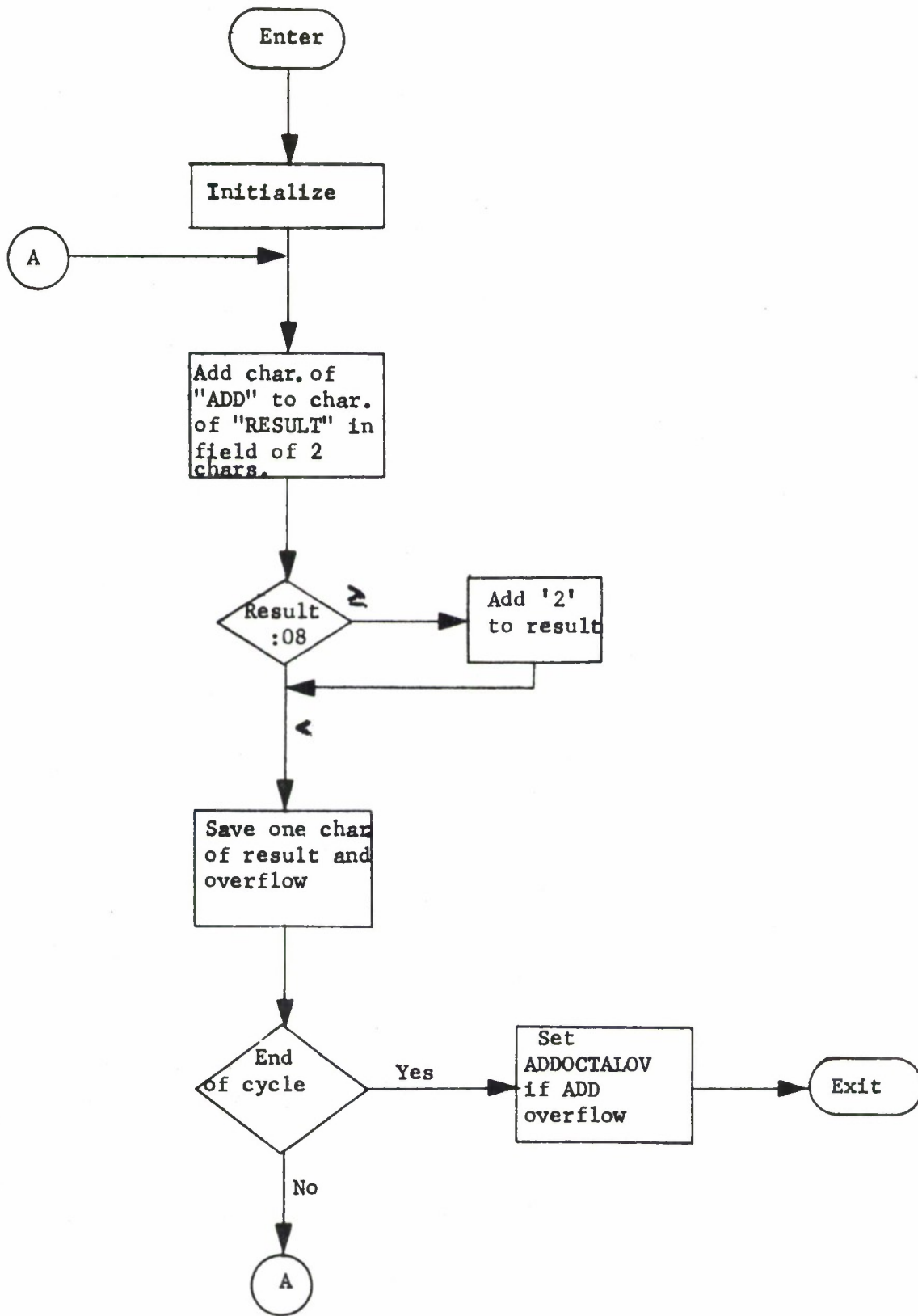
PRINTHEADER

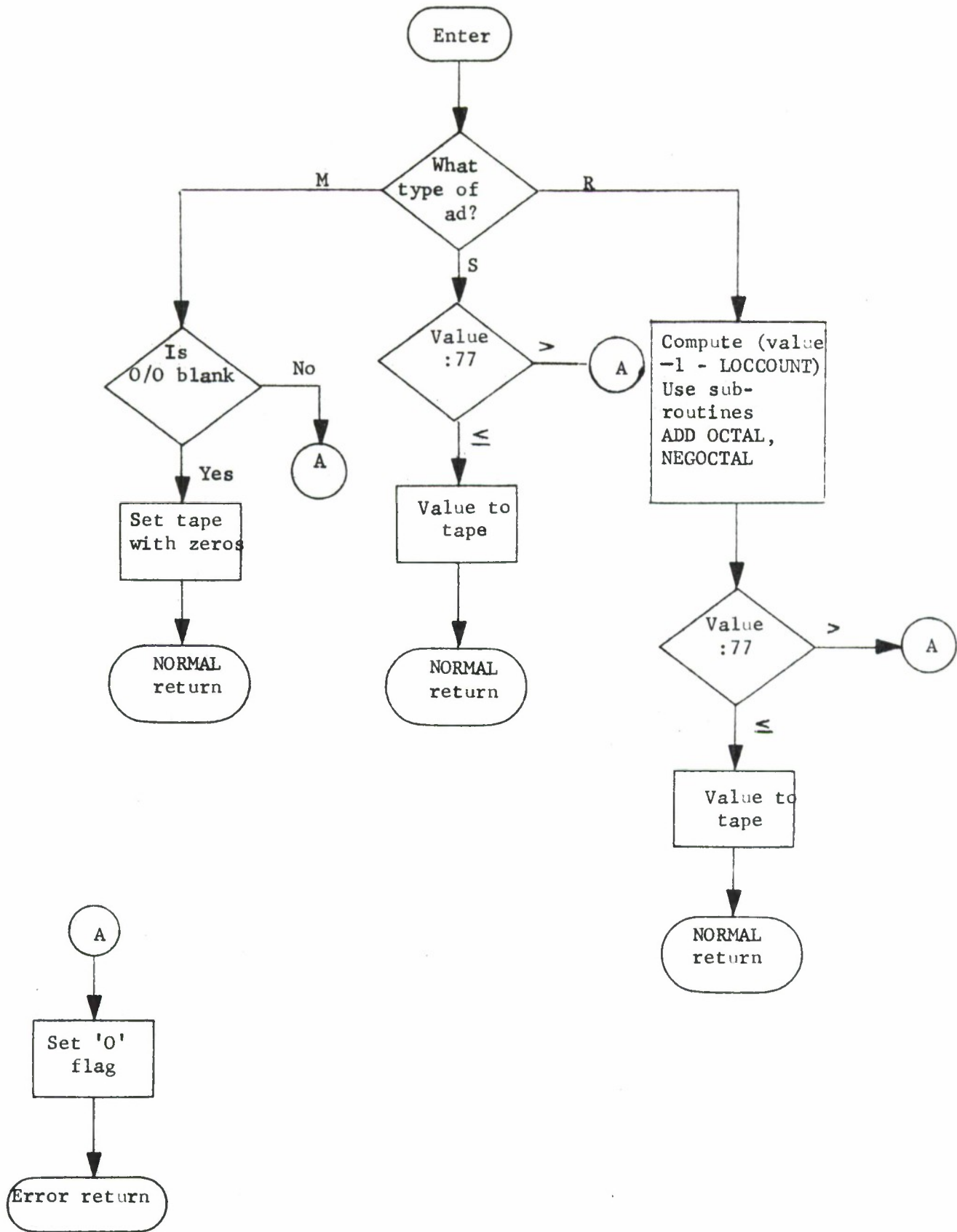


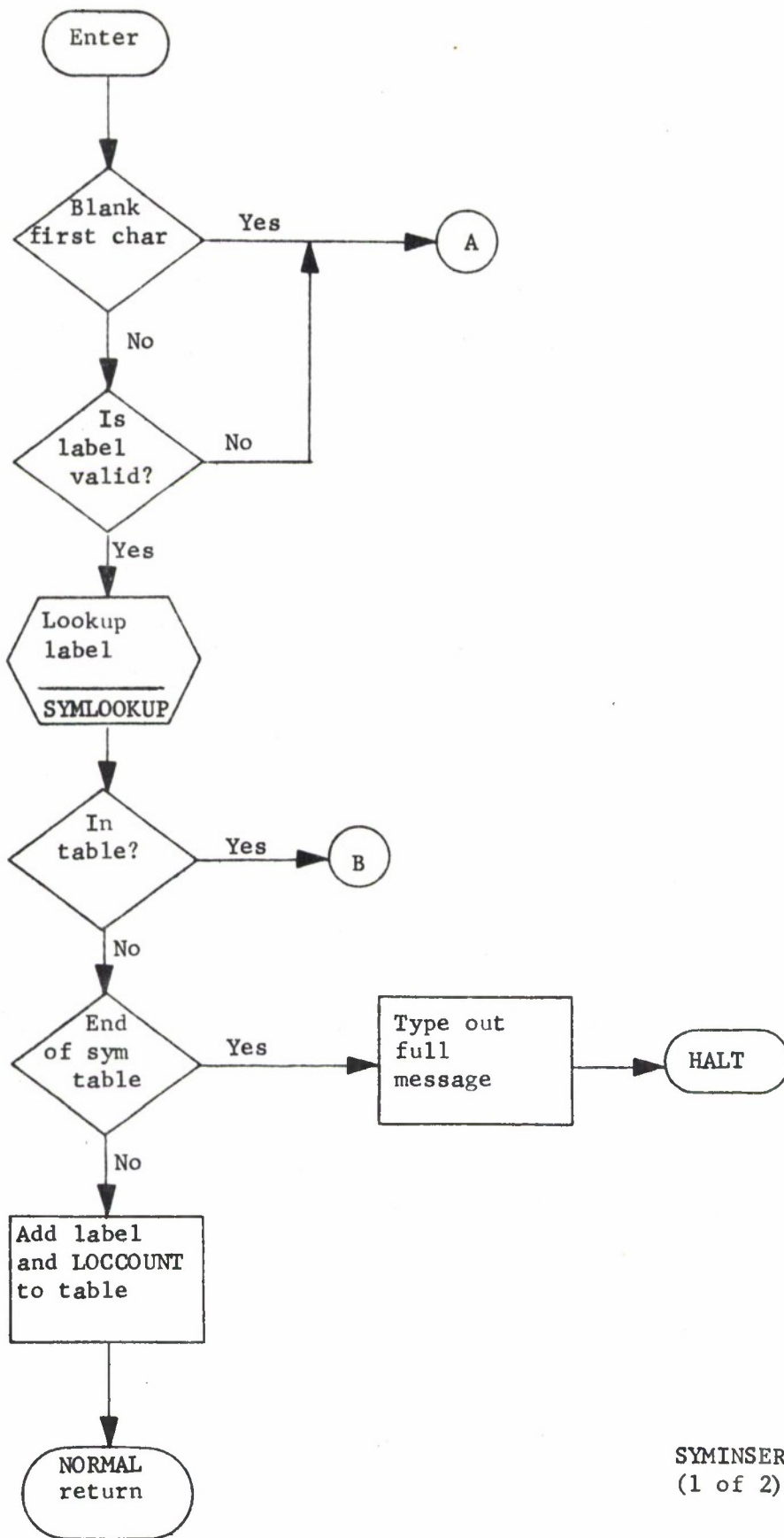




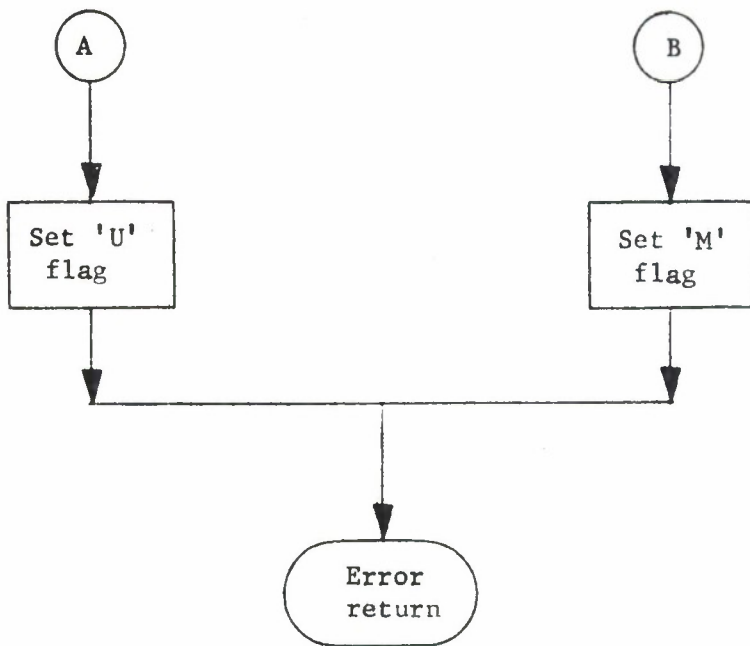


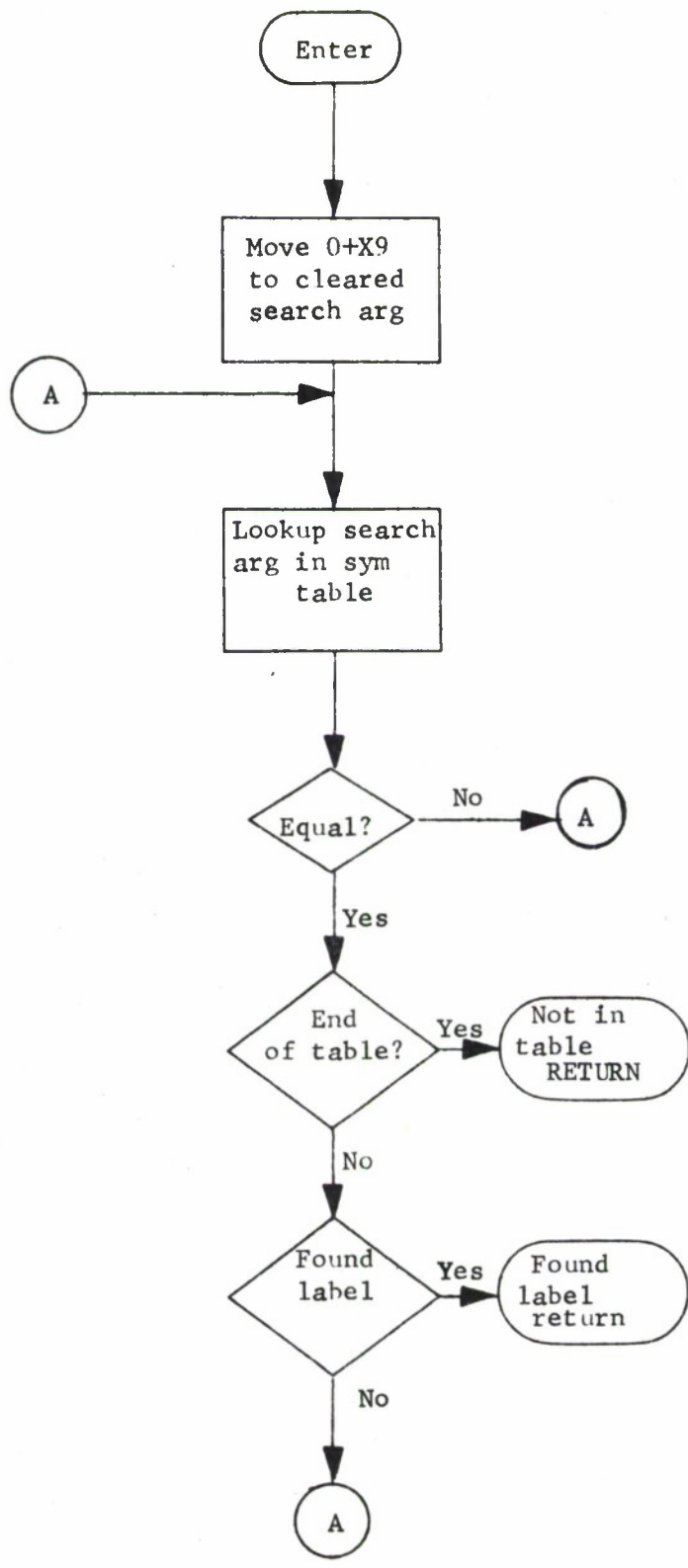




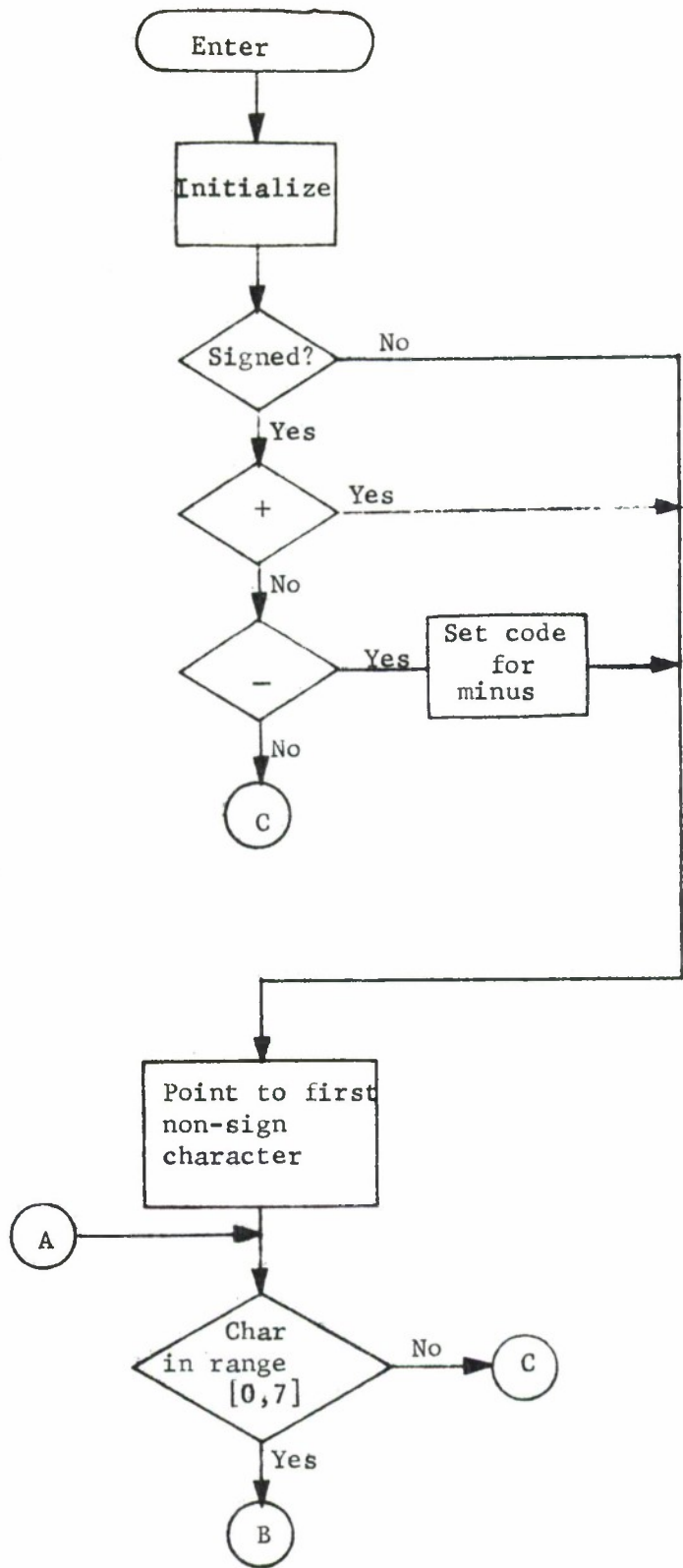


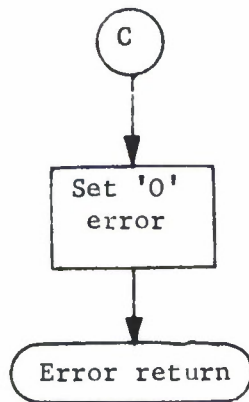
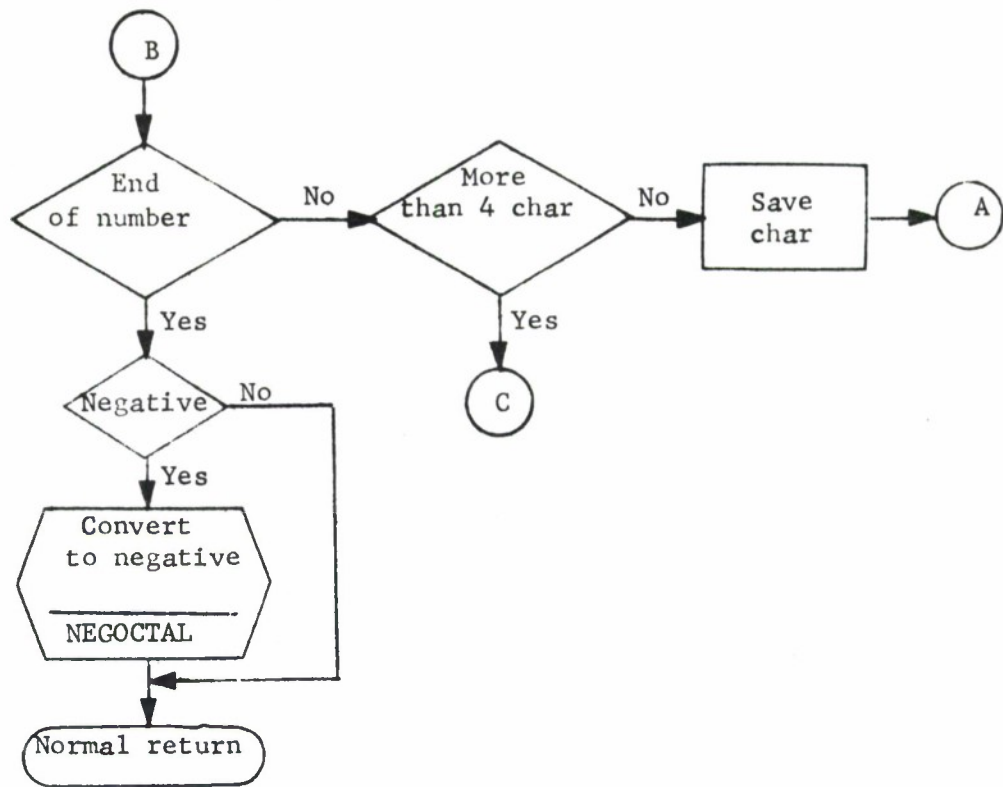
SYMINSERT FLOW CHART
(1 of 2)

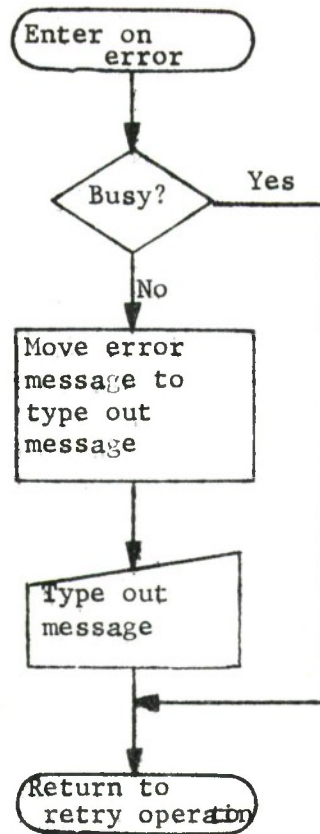




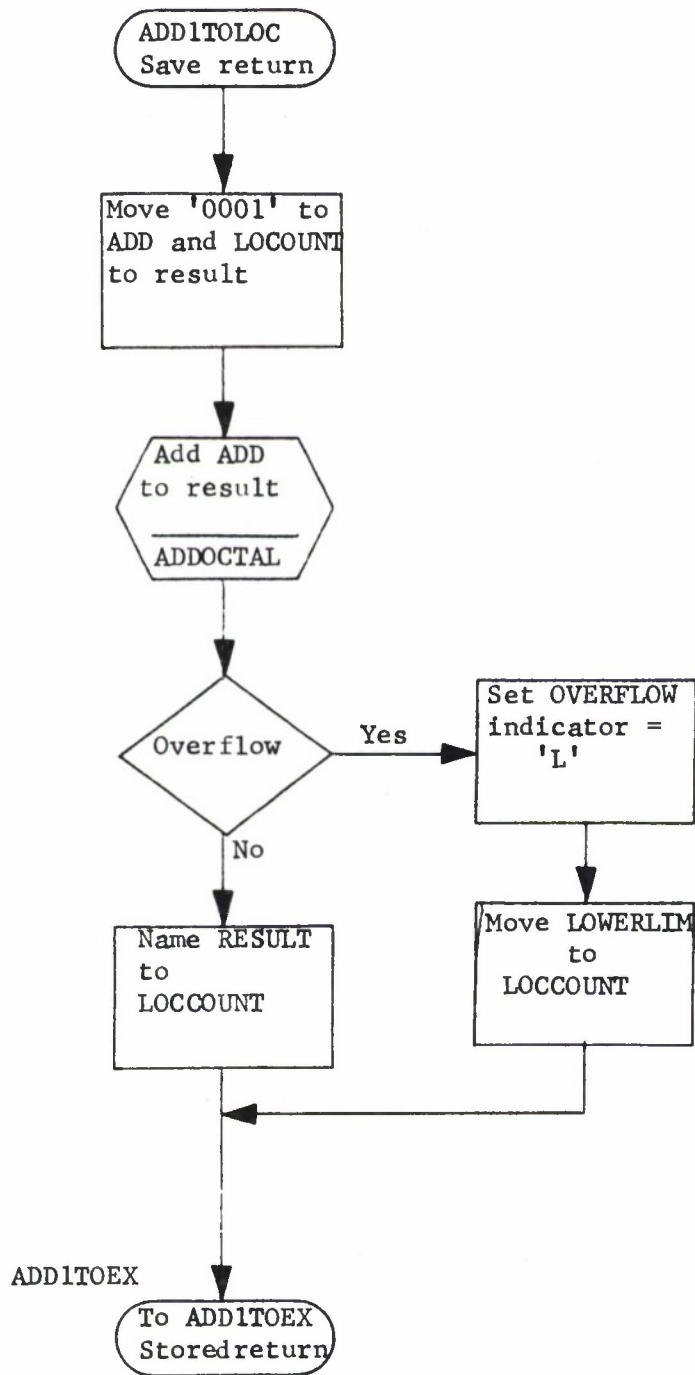
SYMLOOKUP FLOW CHART
(1 of 1)



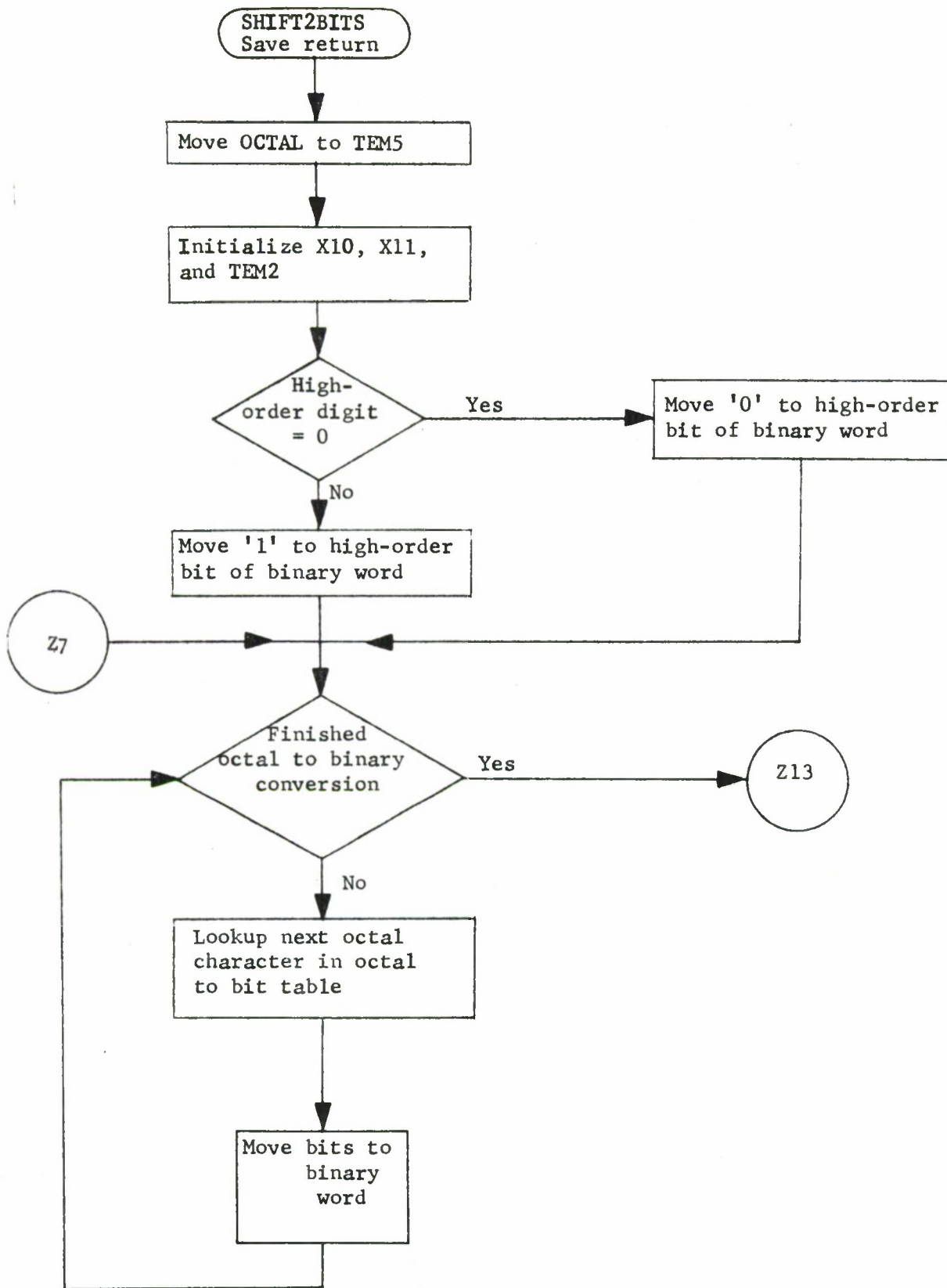


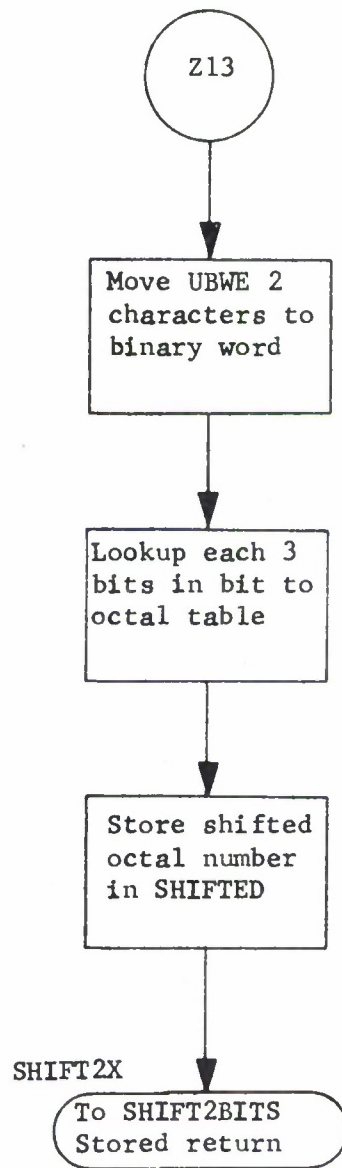


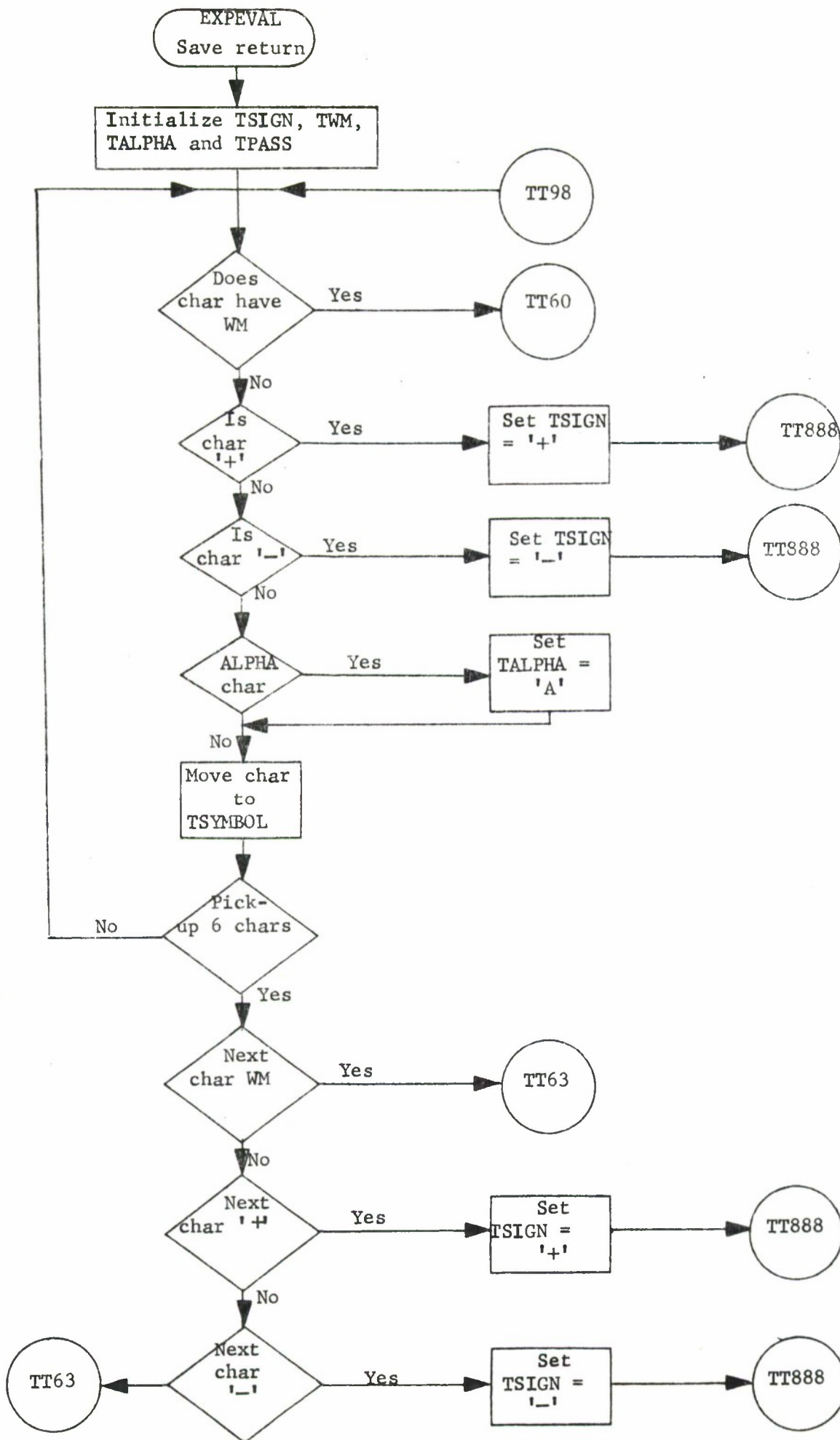
PRINTERR FLOW CHART
(1 of 1)

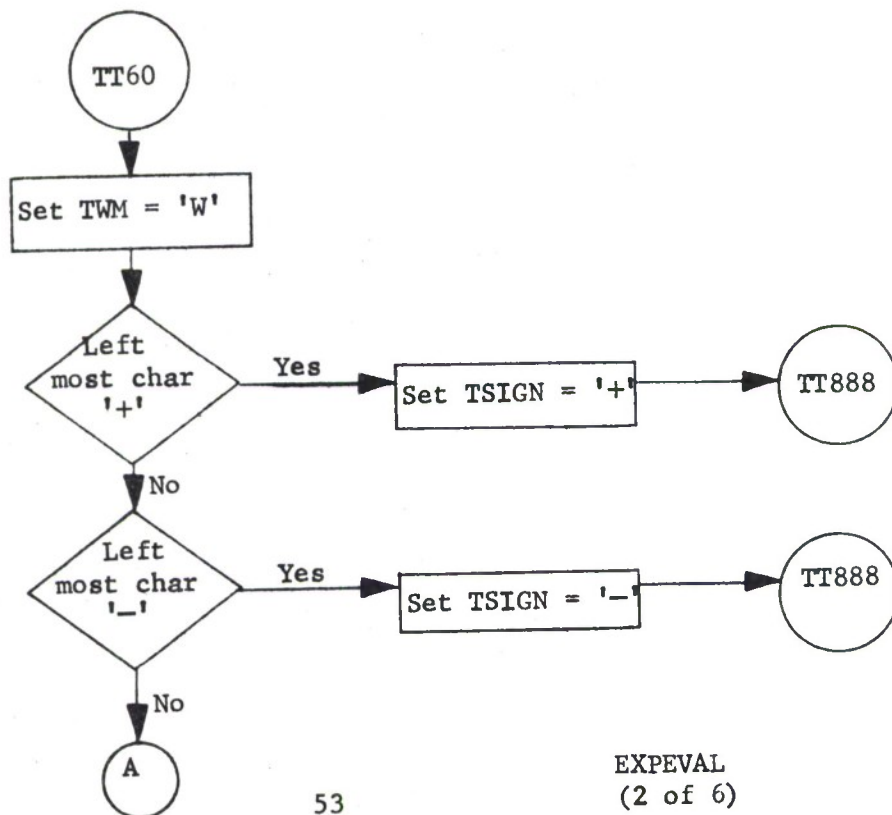
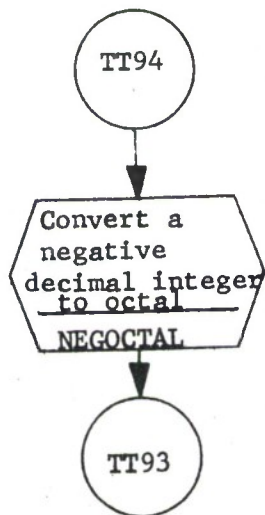
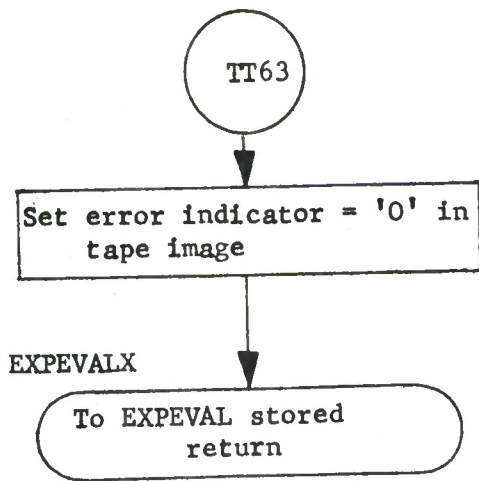


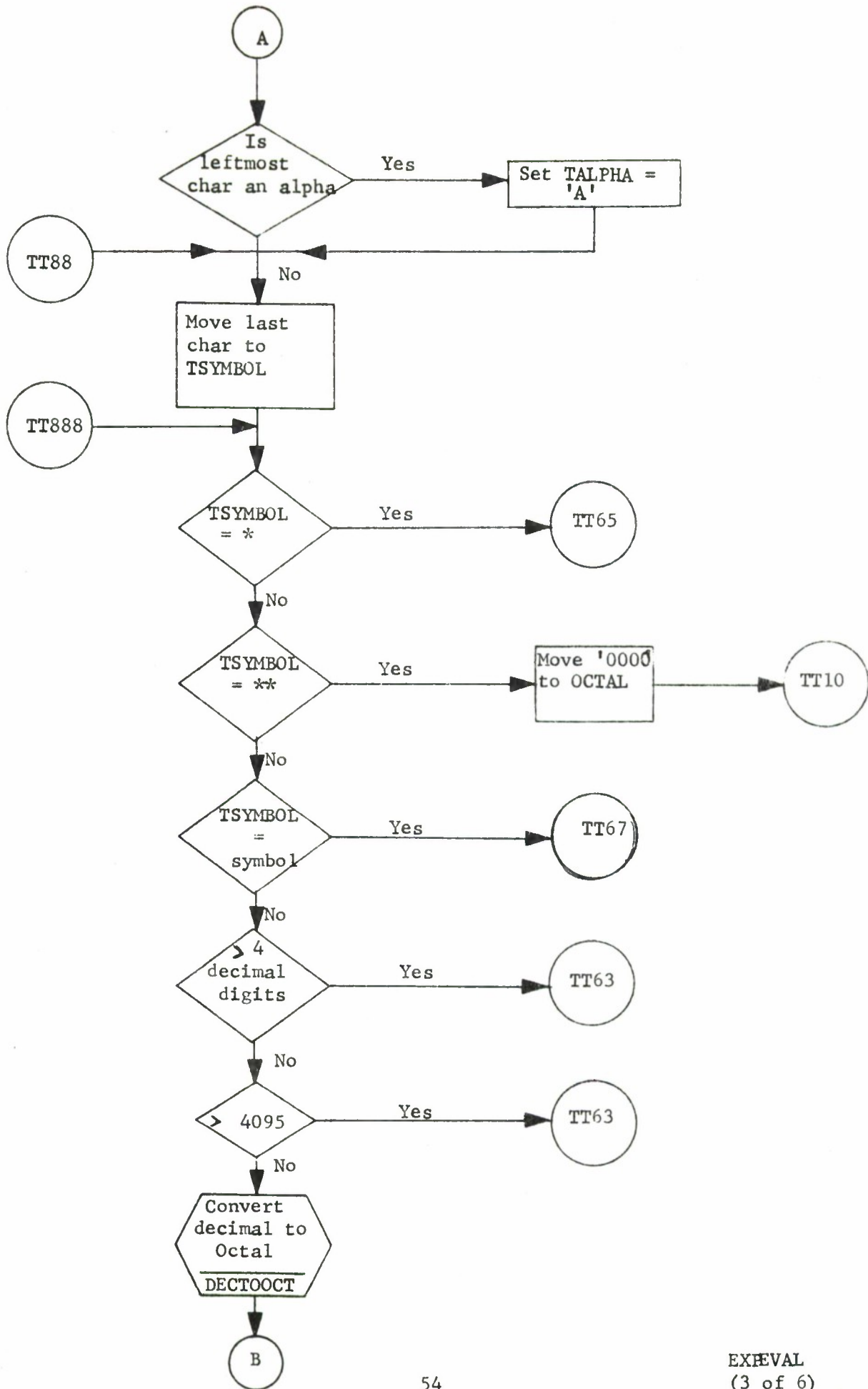
ADDITOCLOC FLOW CHART
(1 of 1)

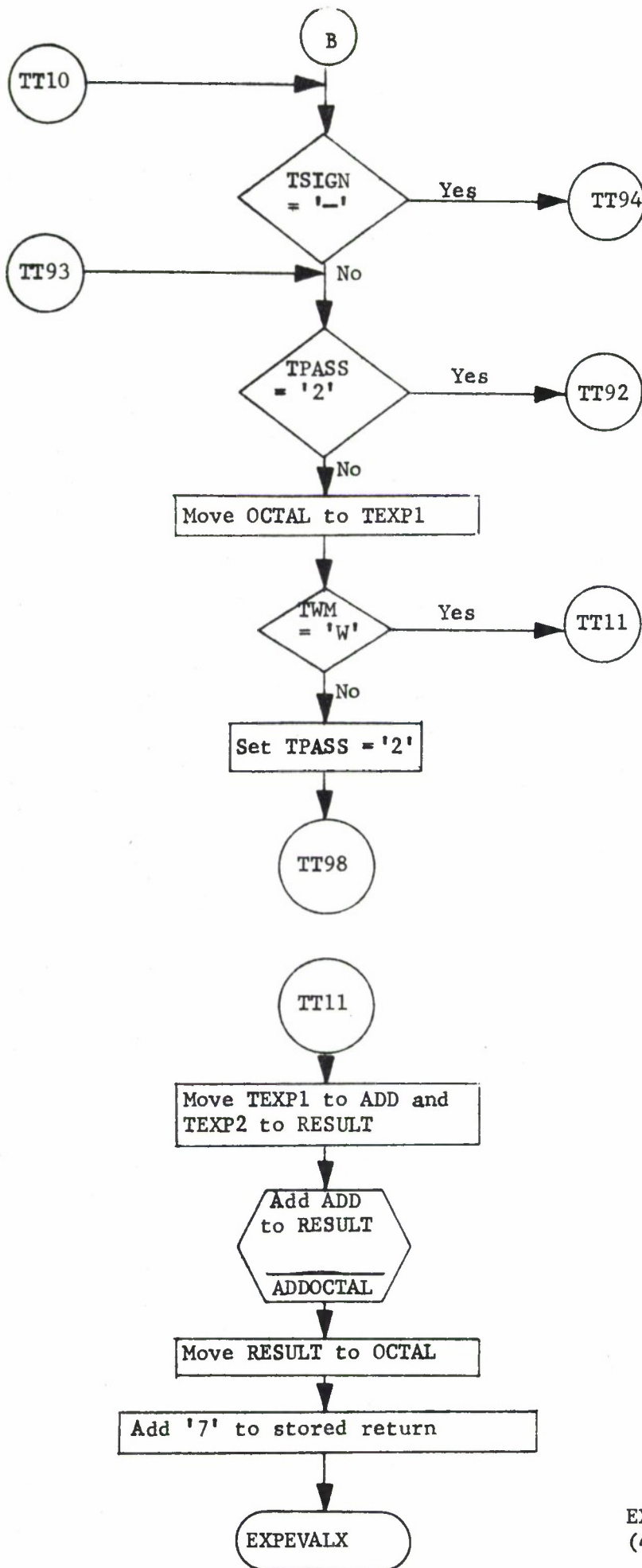




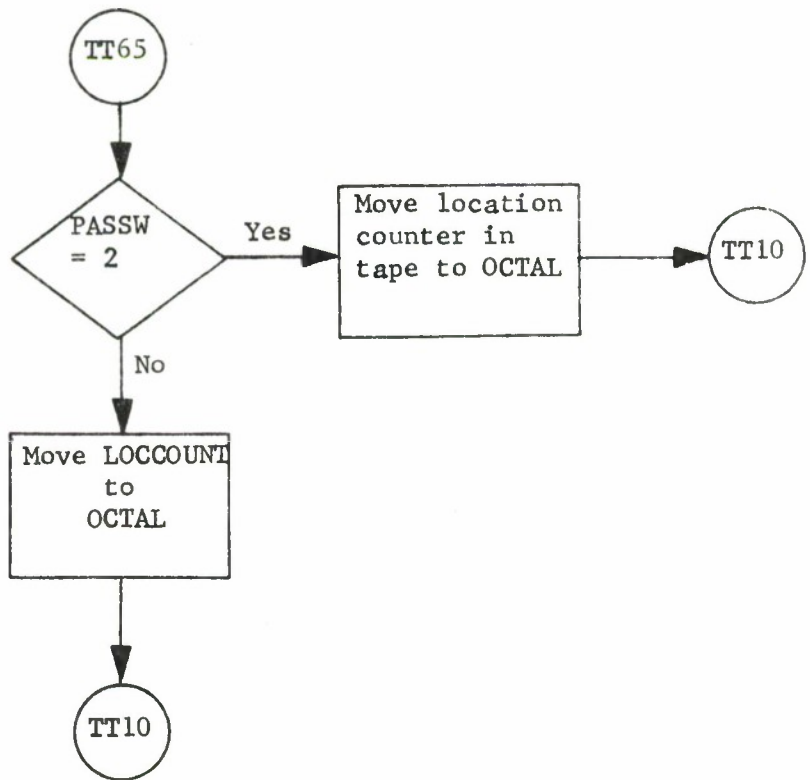
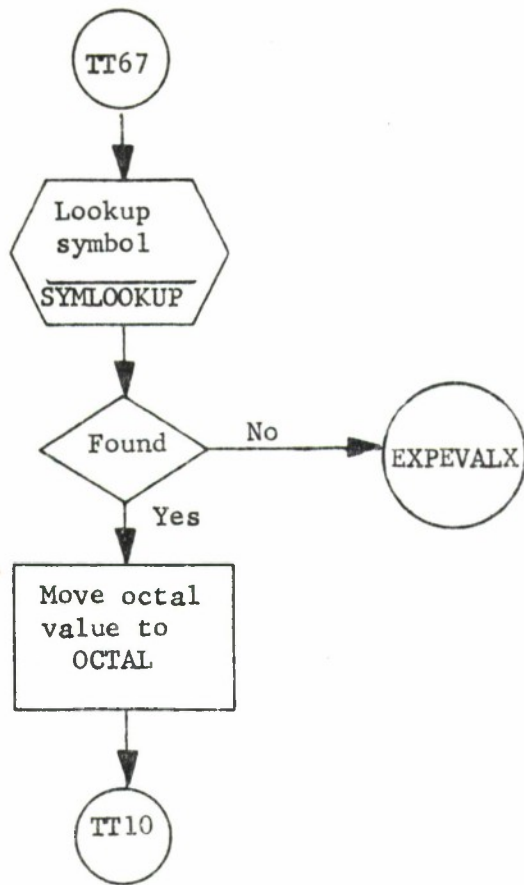


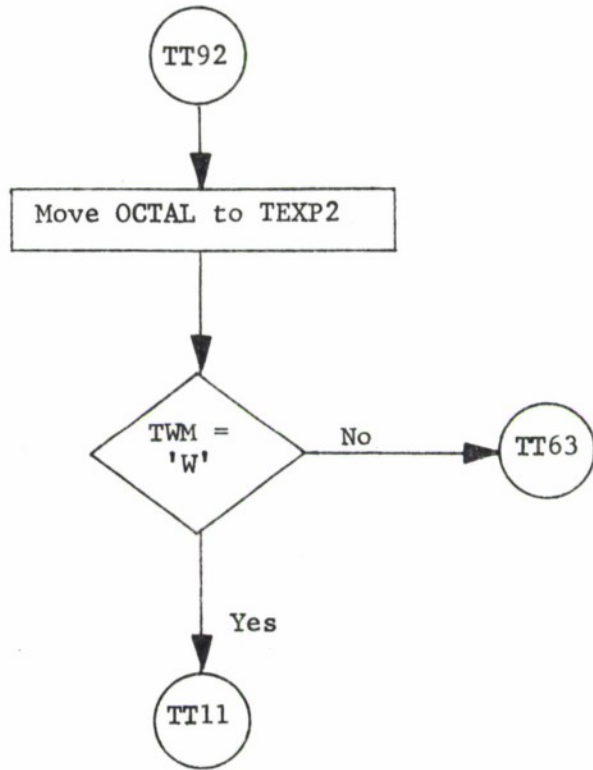


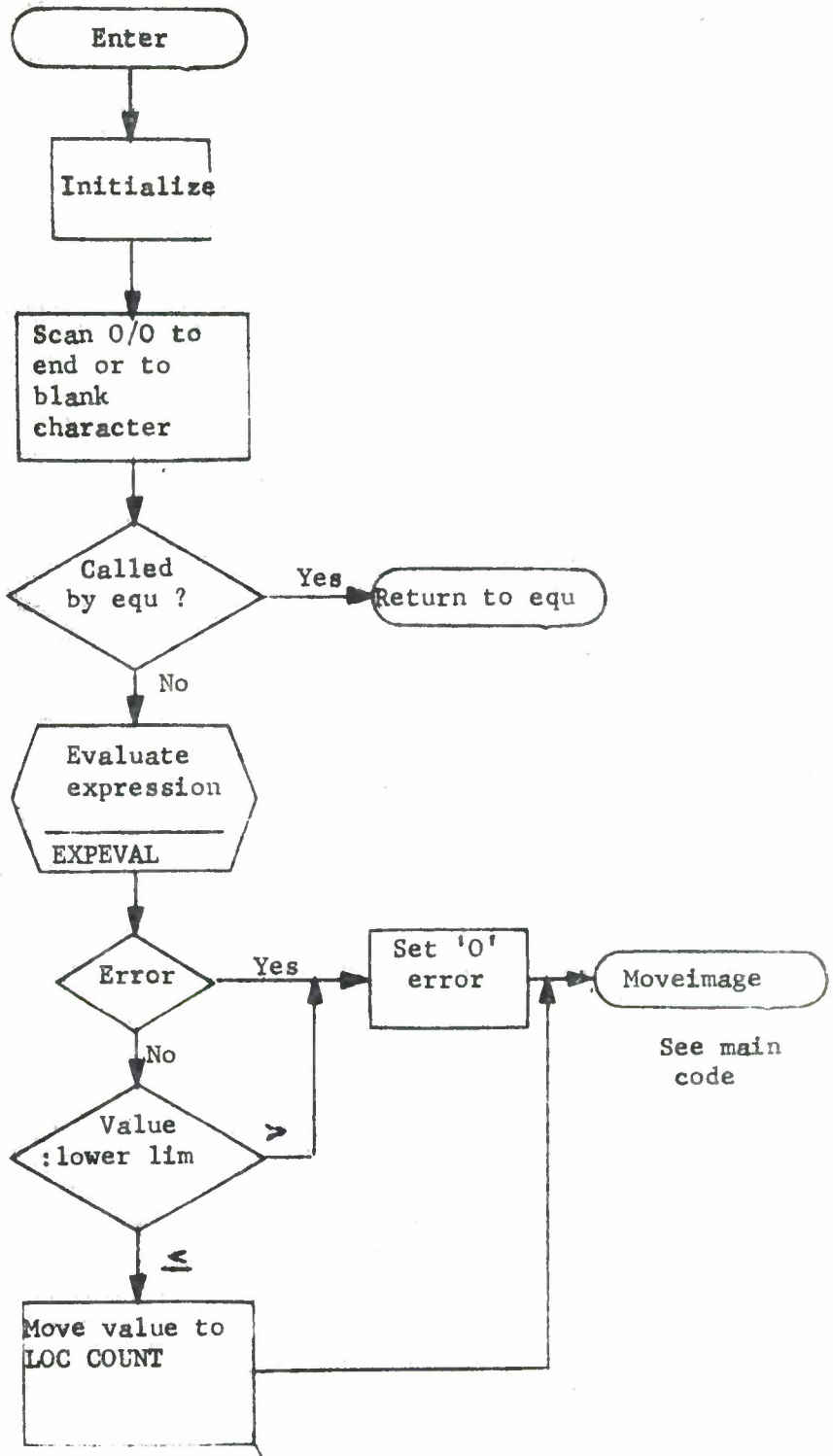


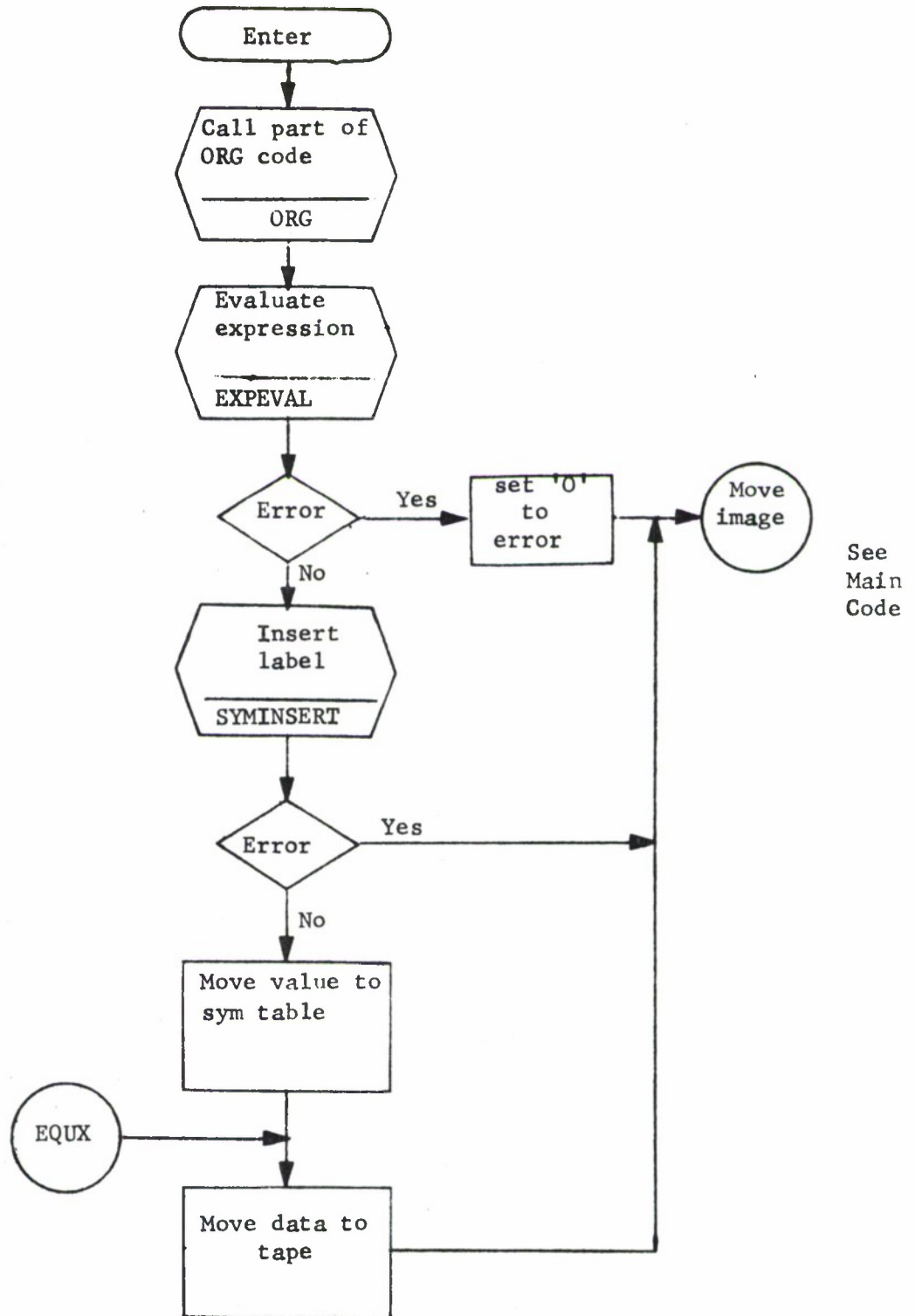


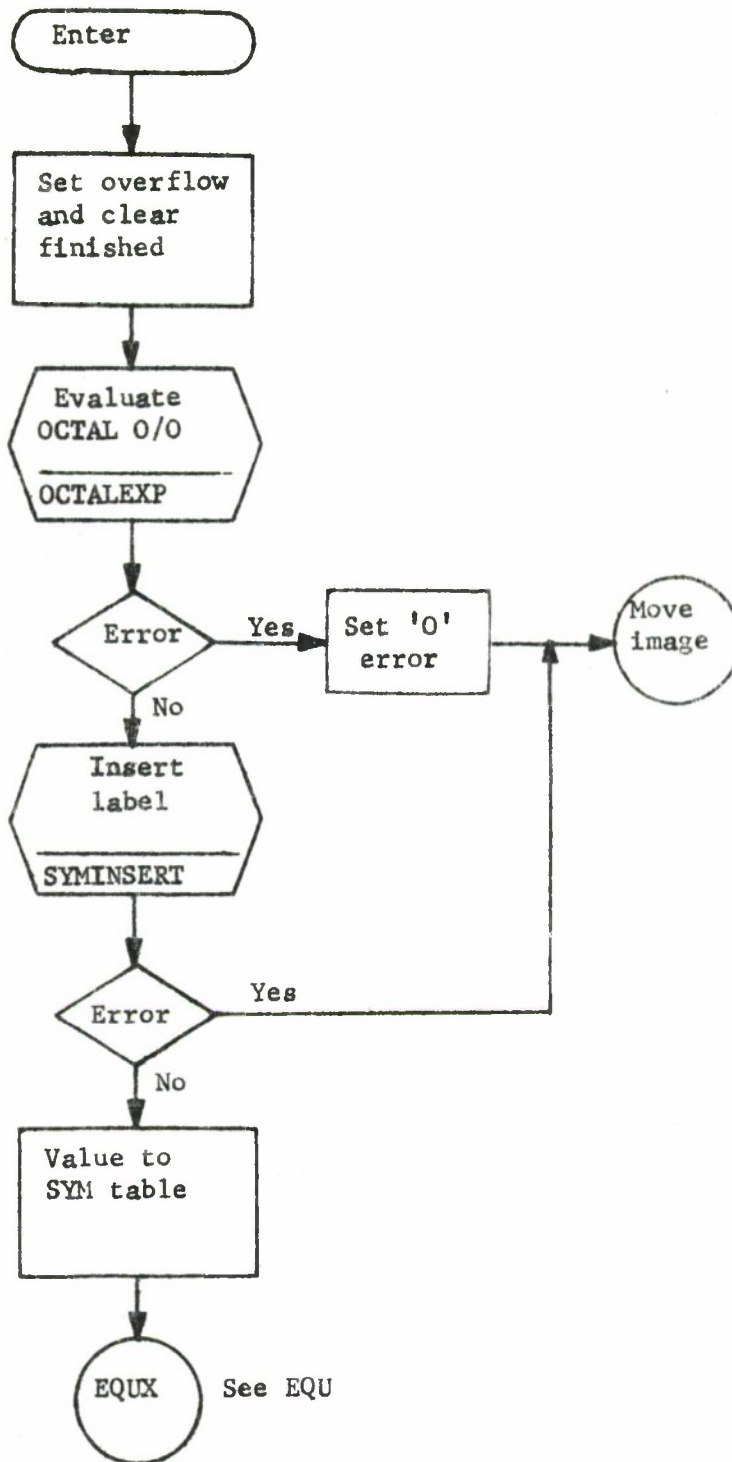
EXPEVAL
(4 of 6)



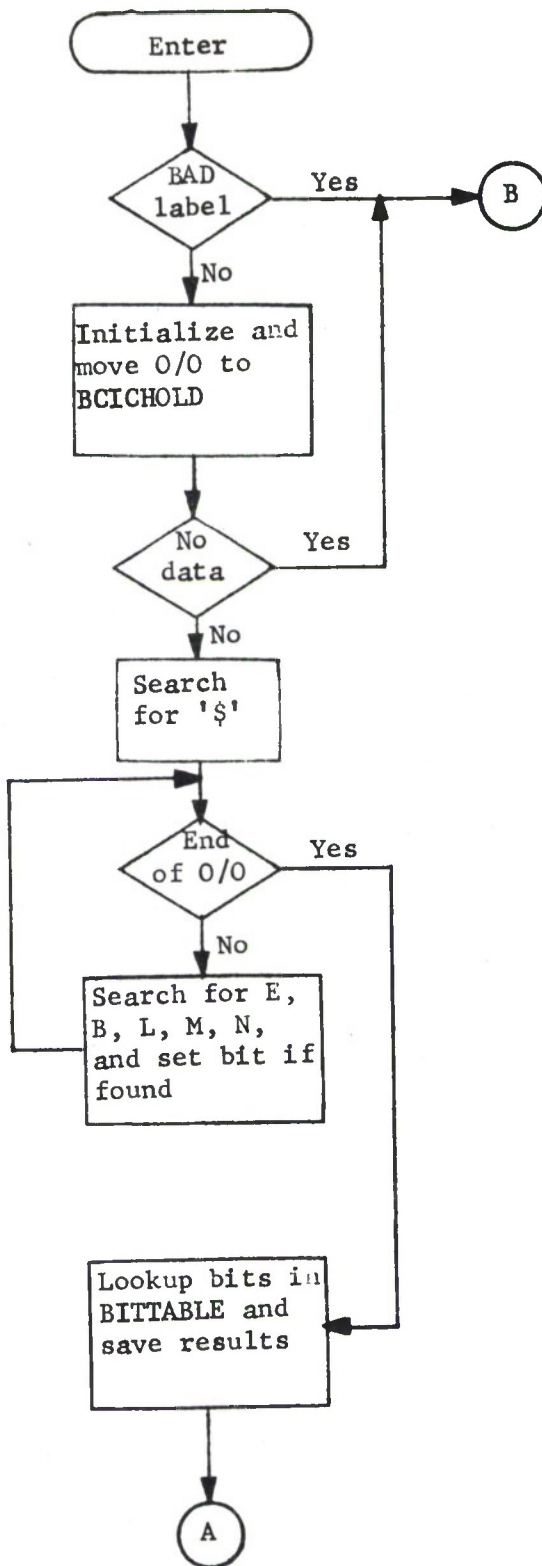


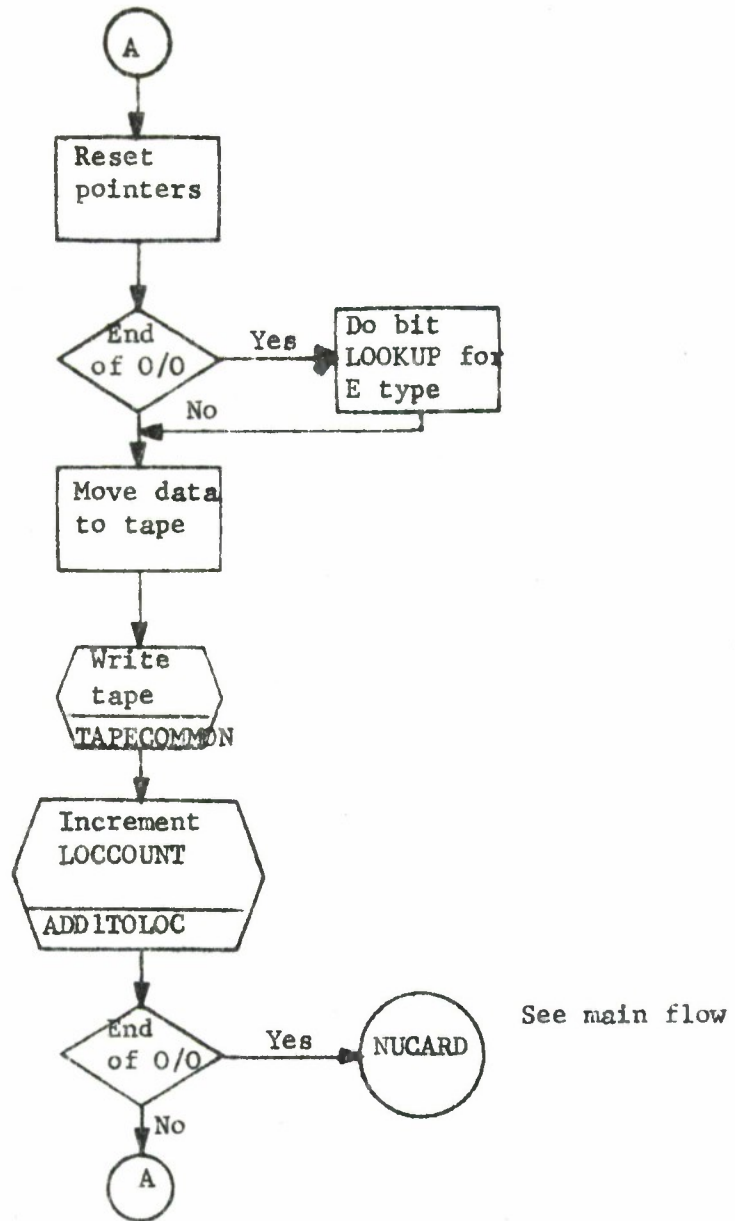


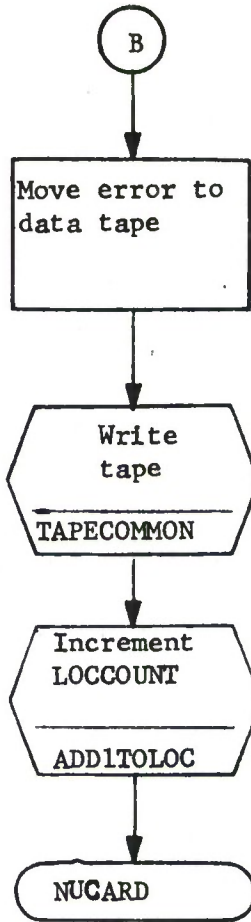


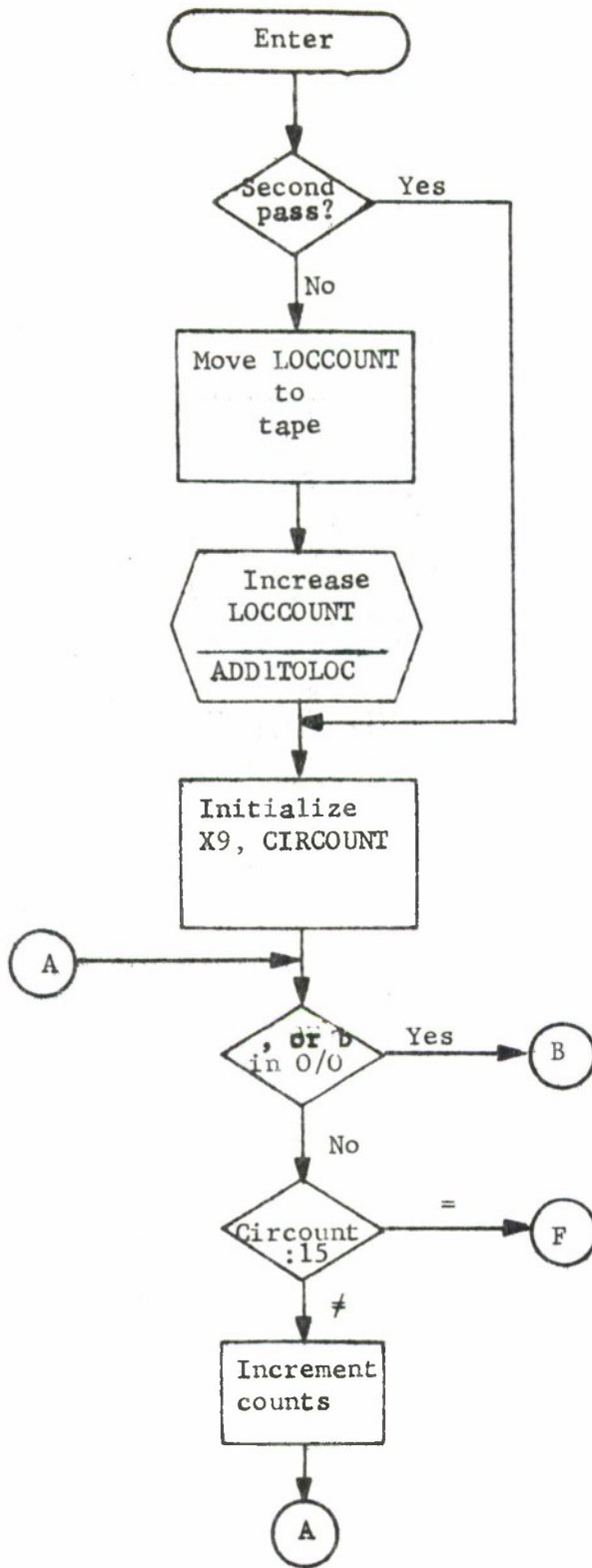


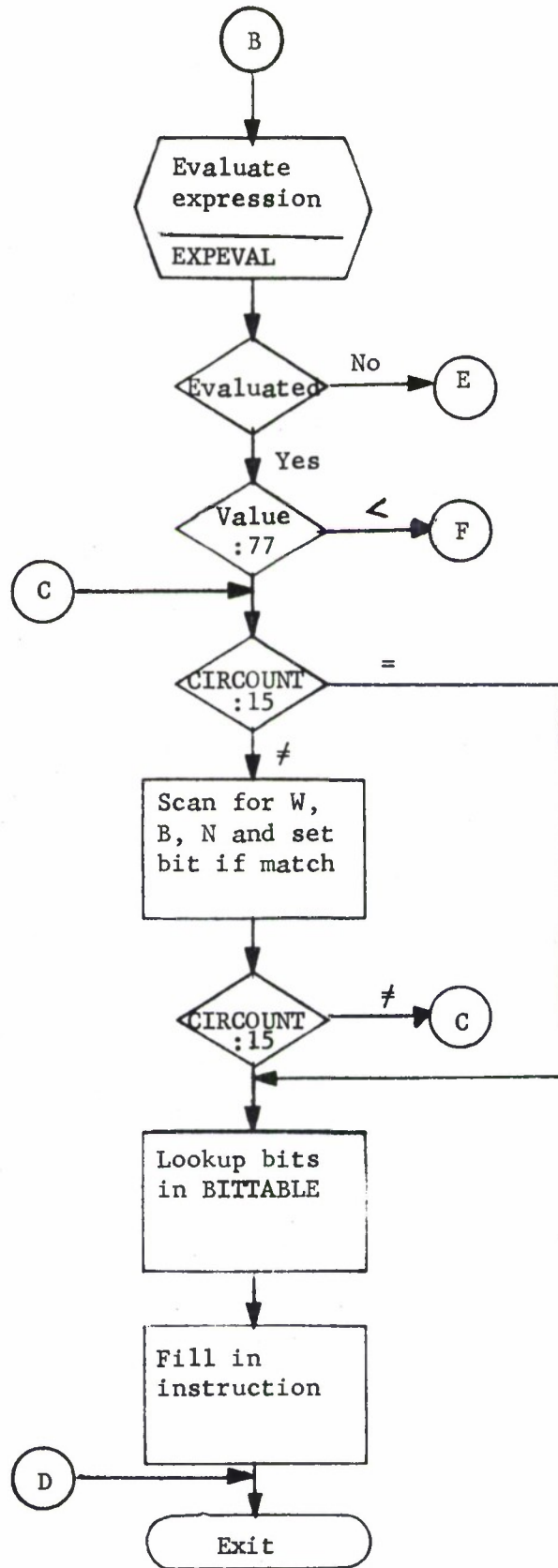
EQUB FLOW CHART
(1 of 1)

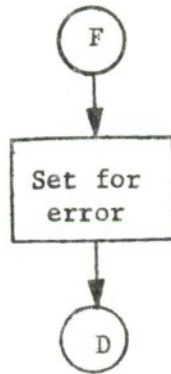
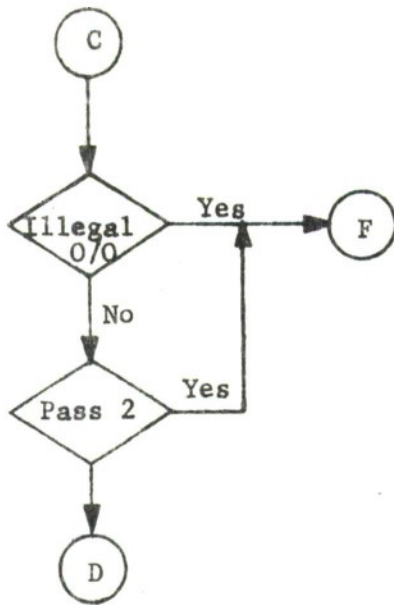


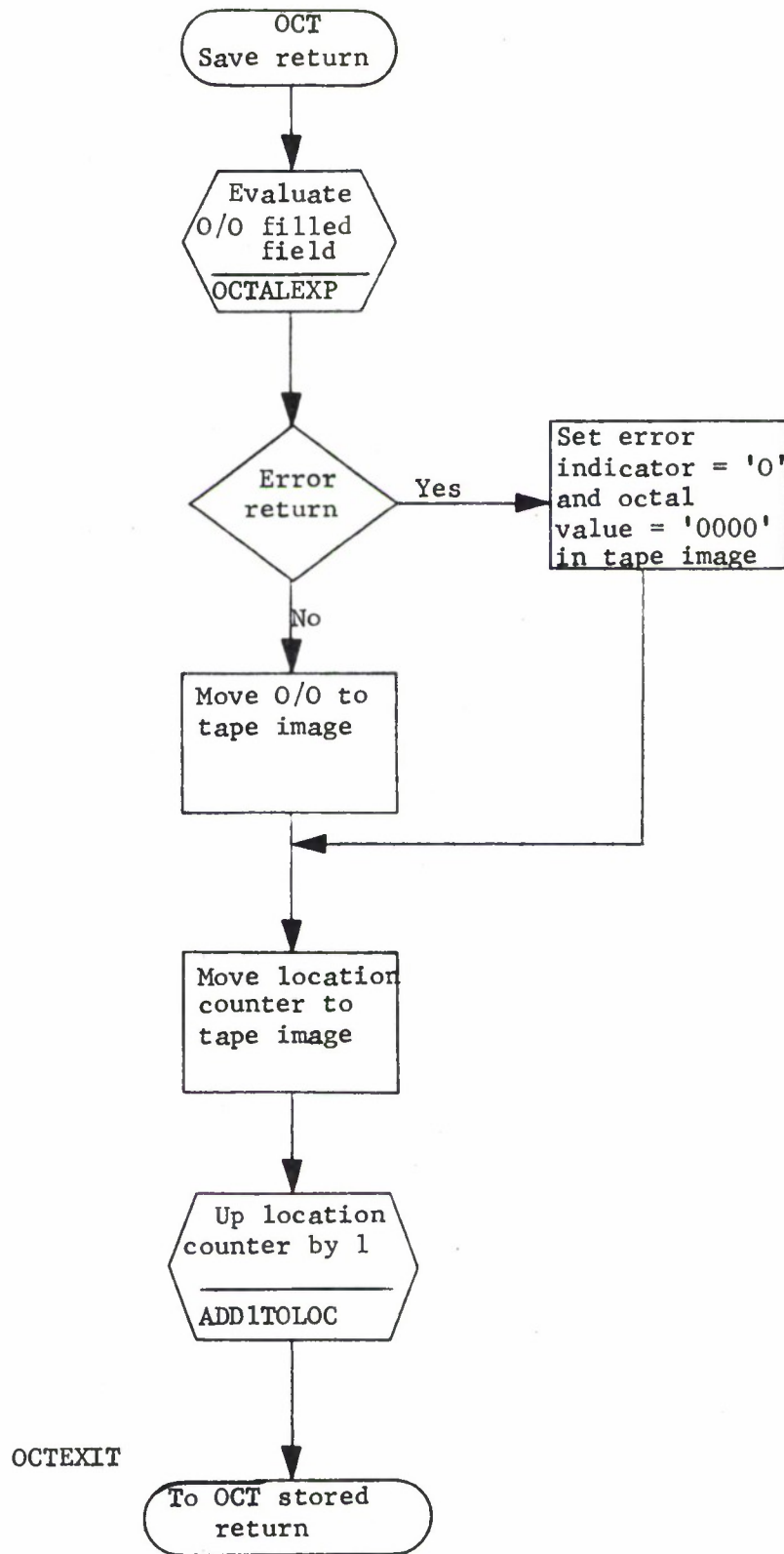


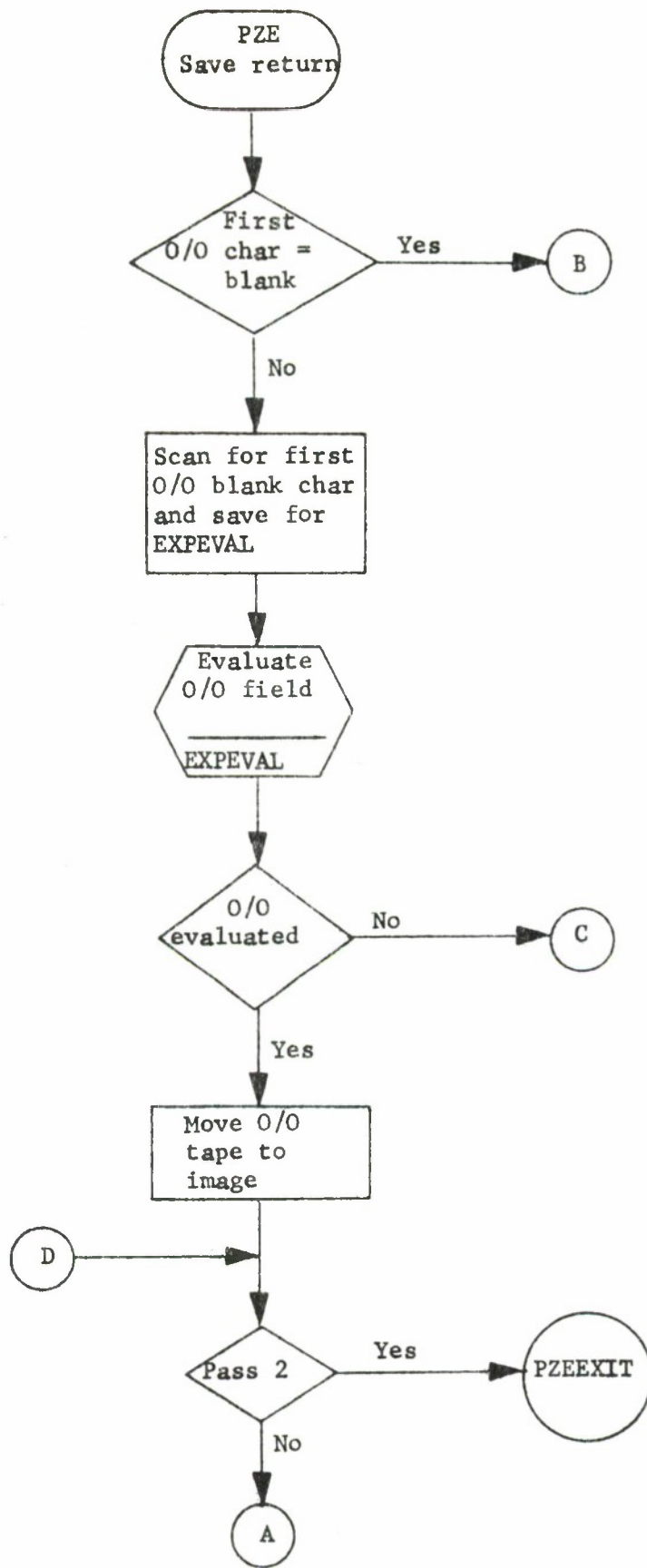


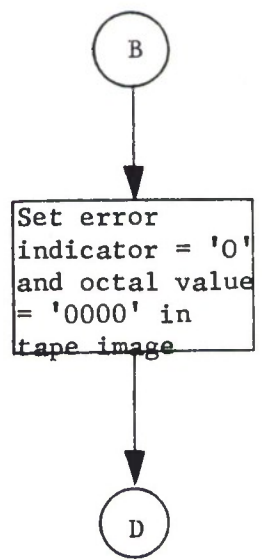
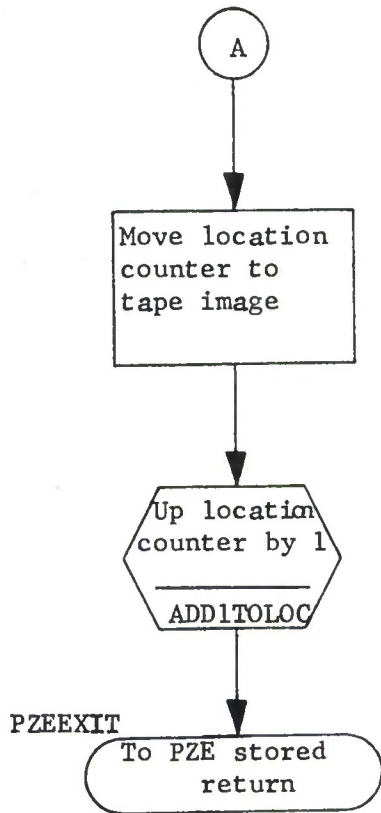


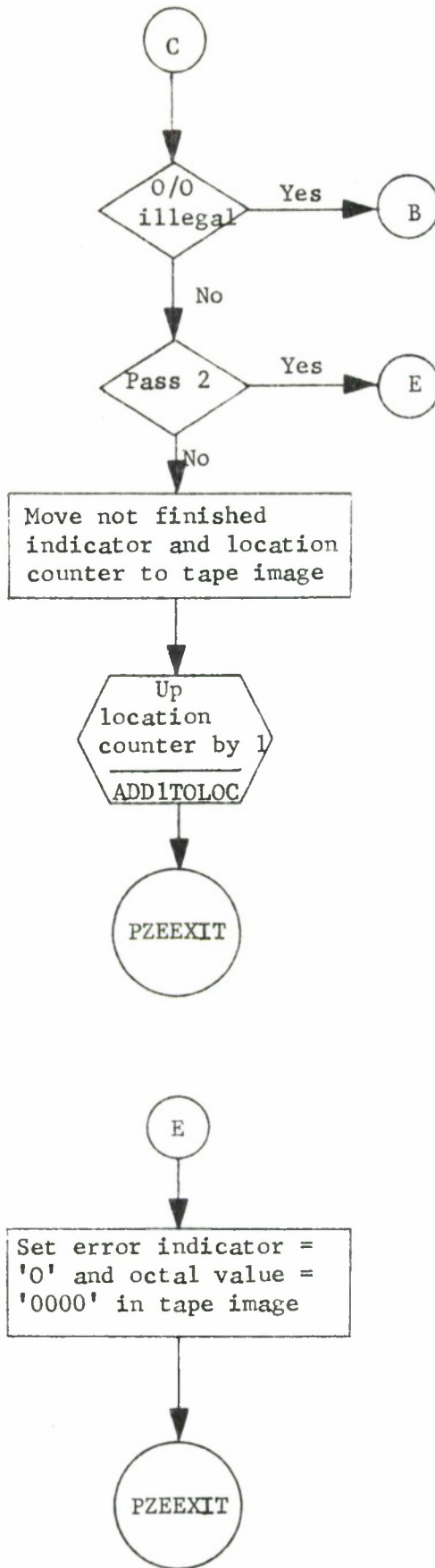


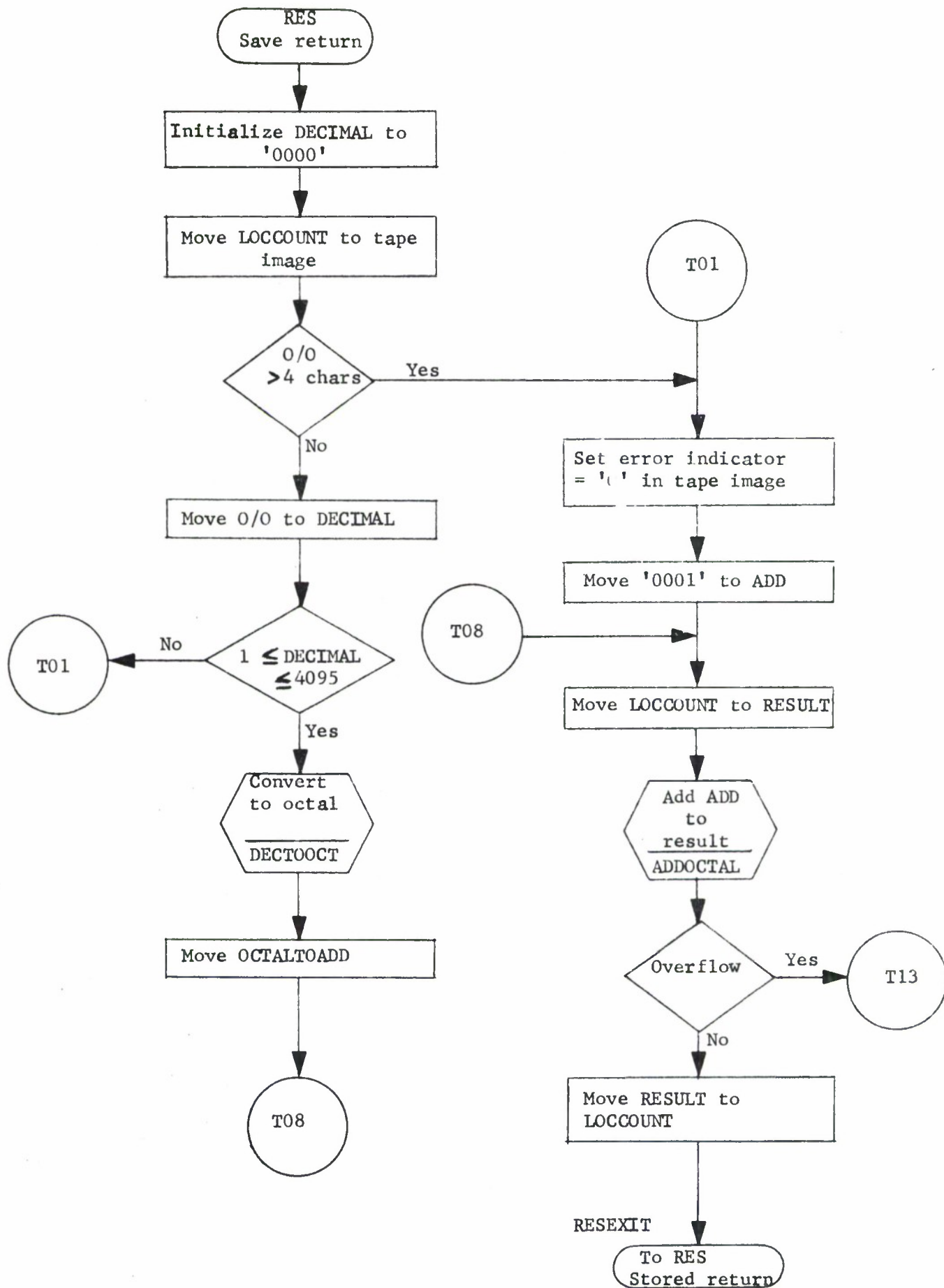


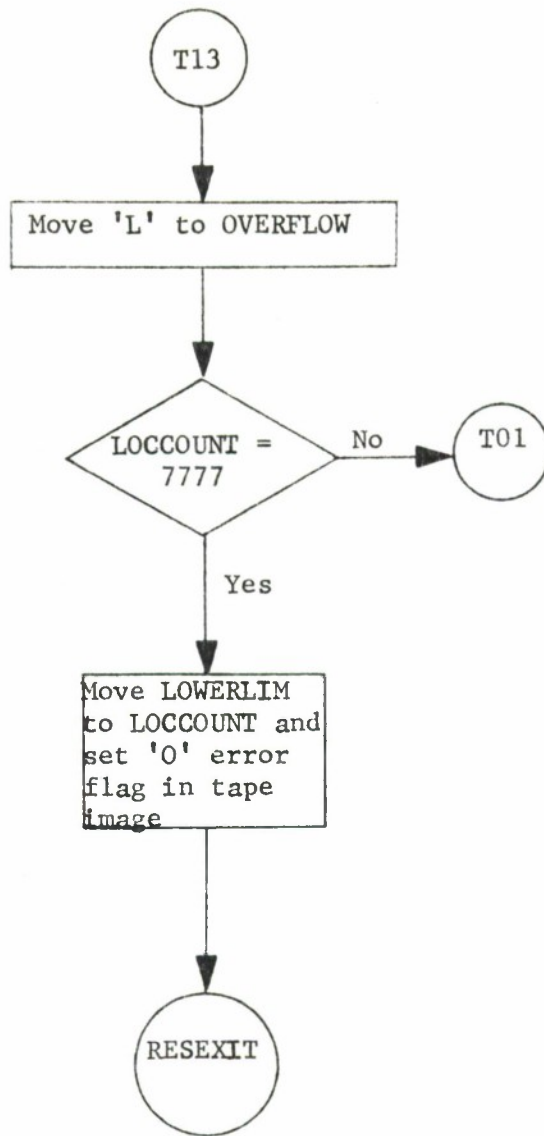




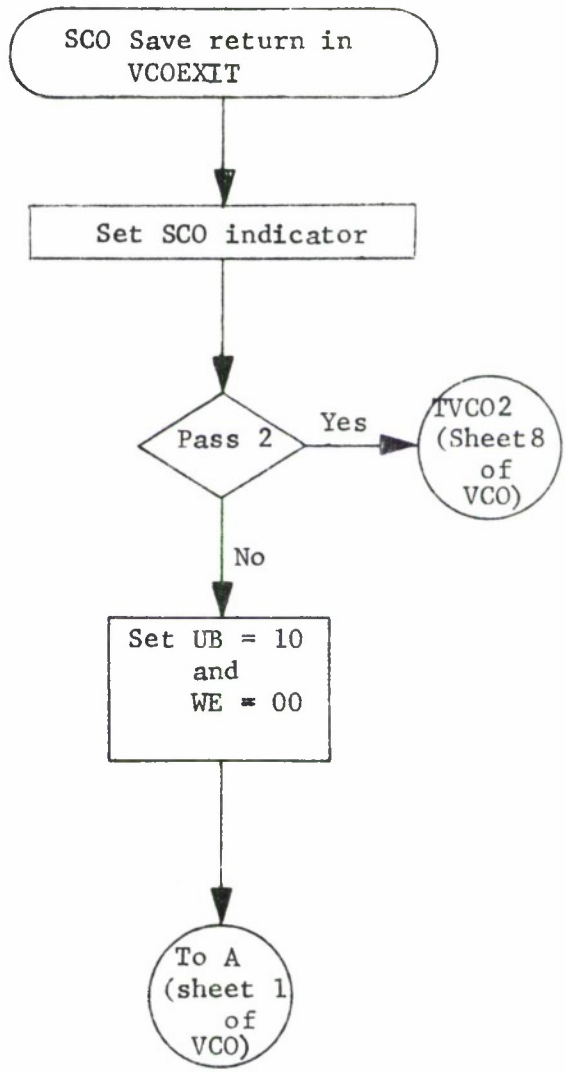




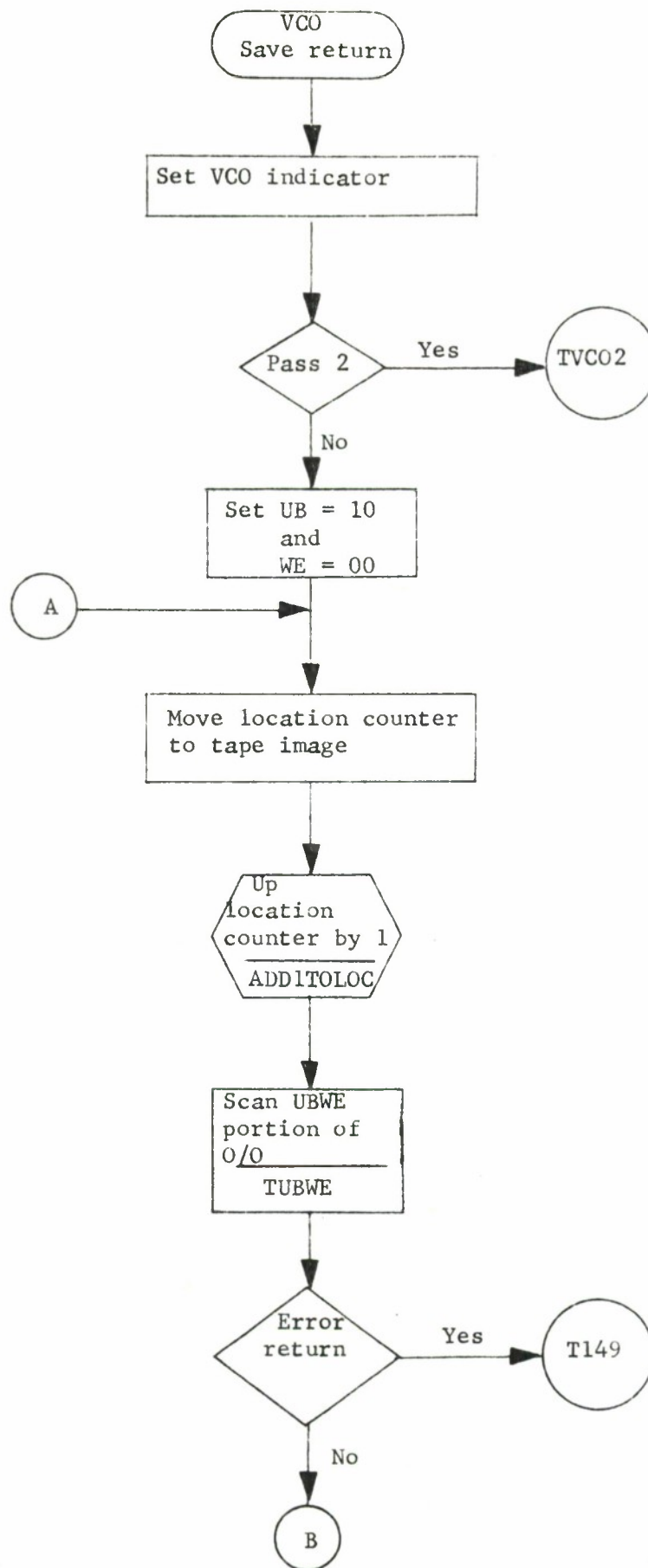




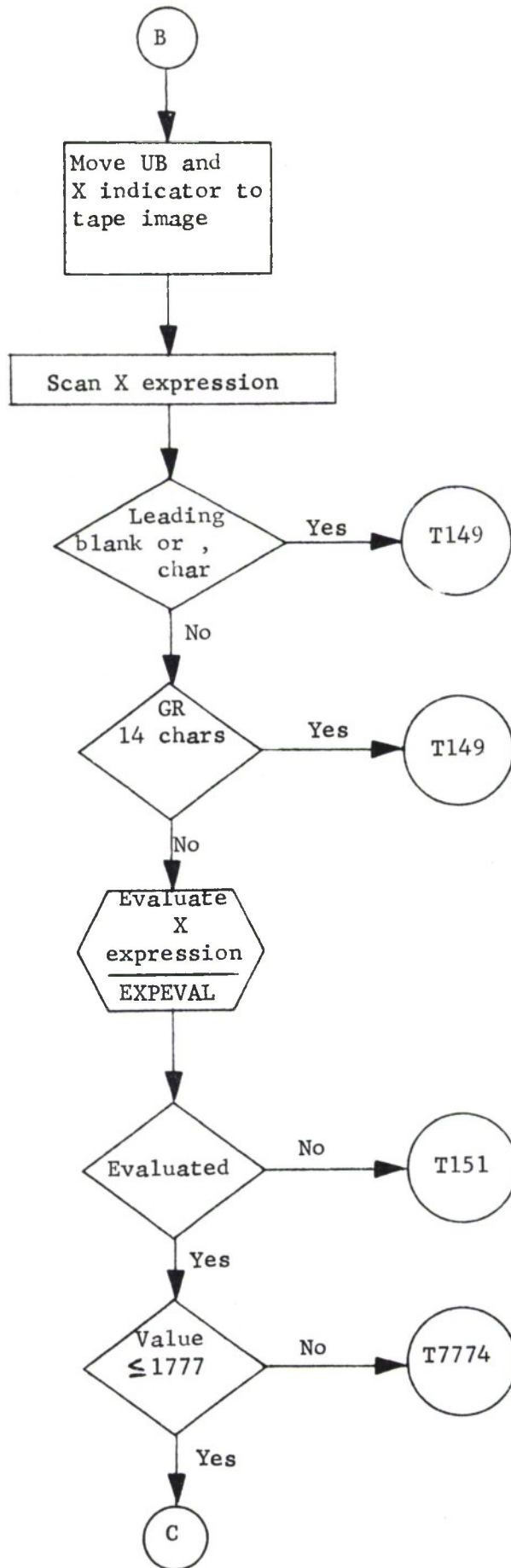
RES
(2 of 2)

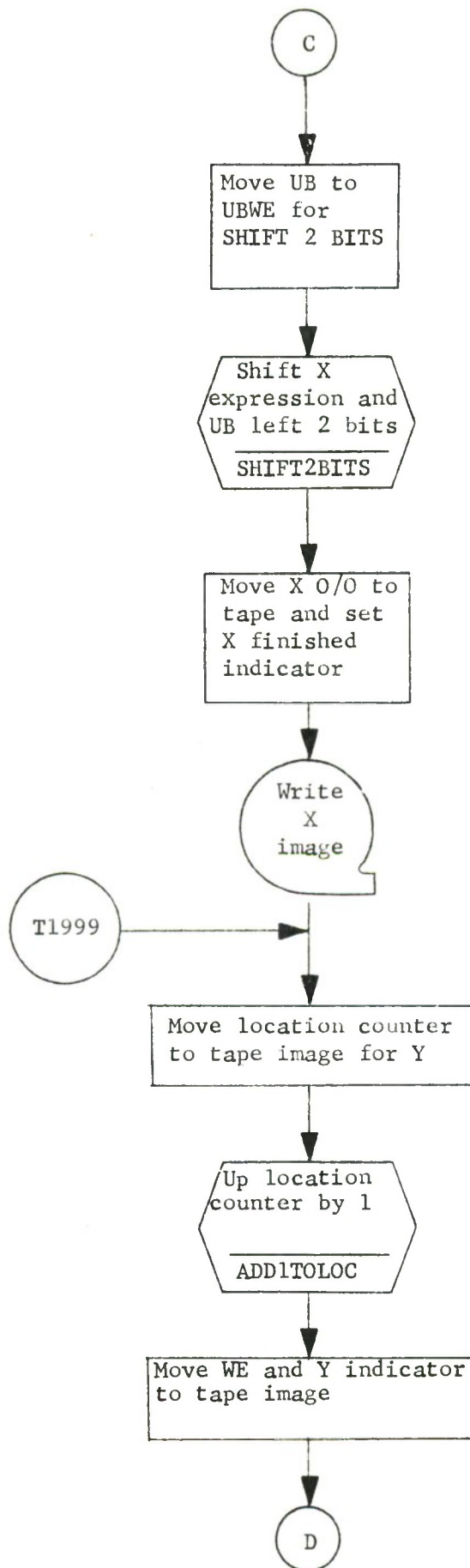


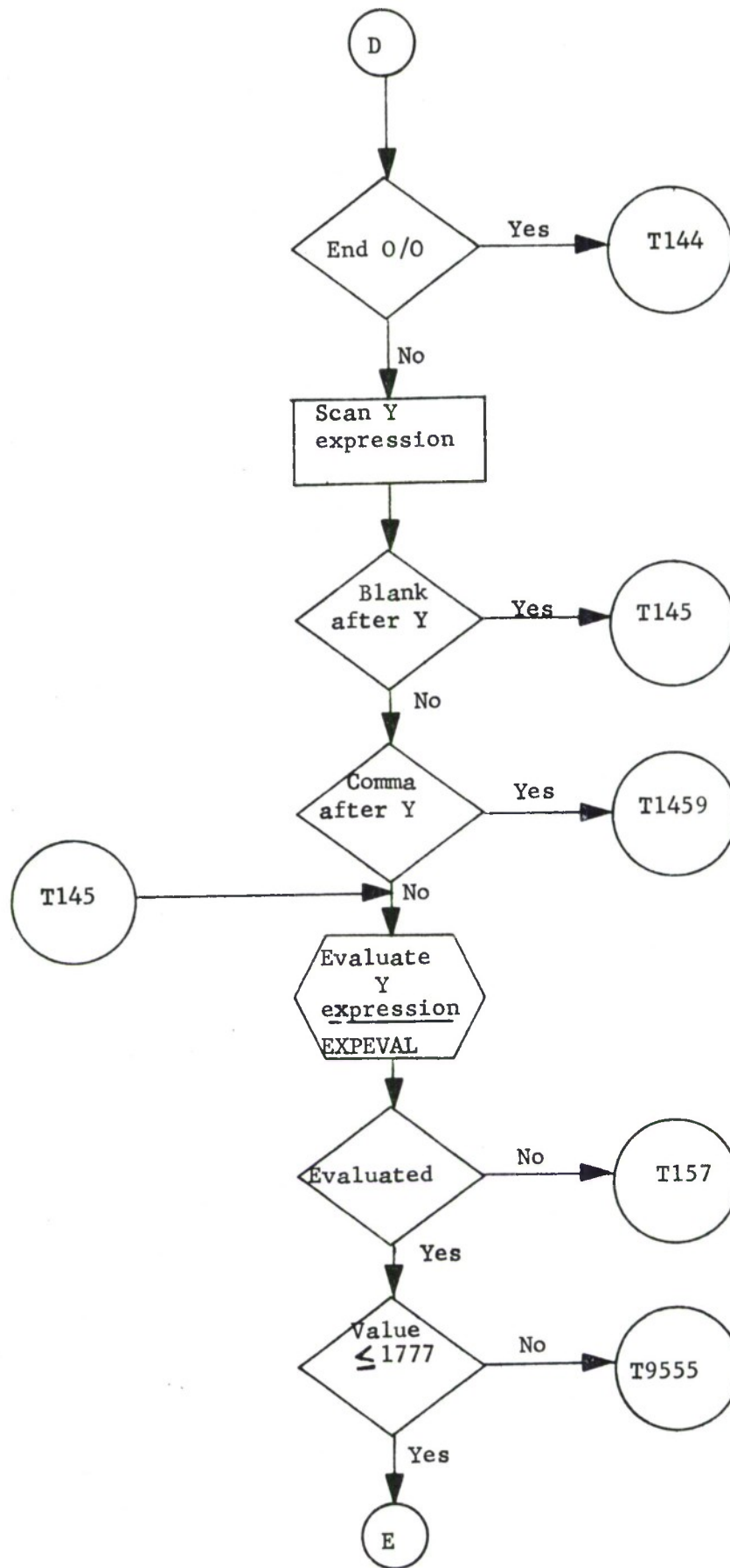
SCO Flow Chart
(1 of 1)

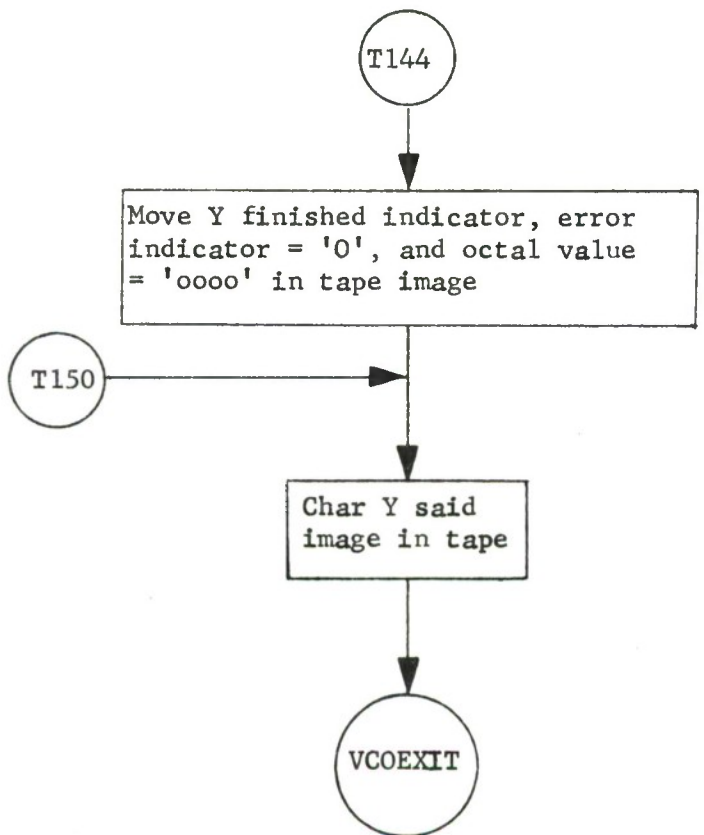
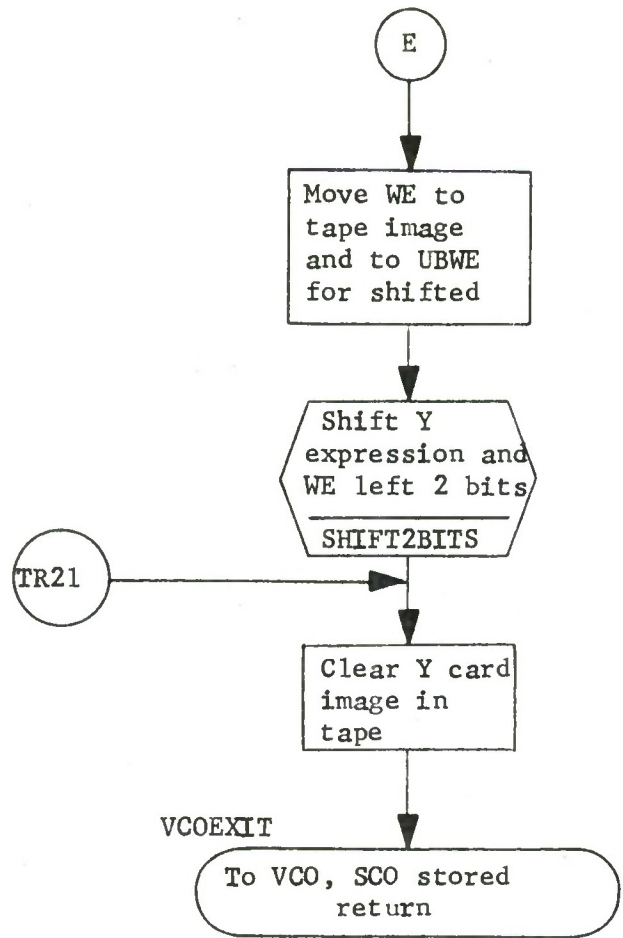


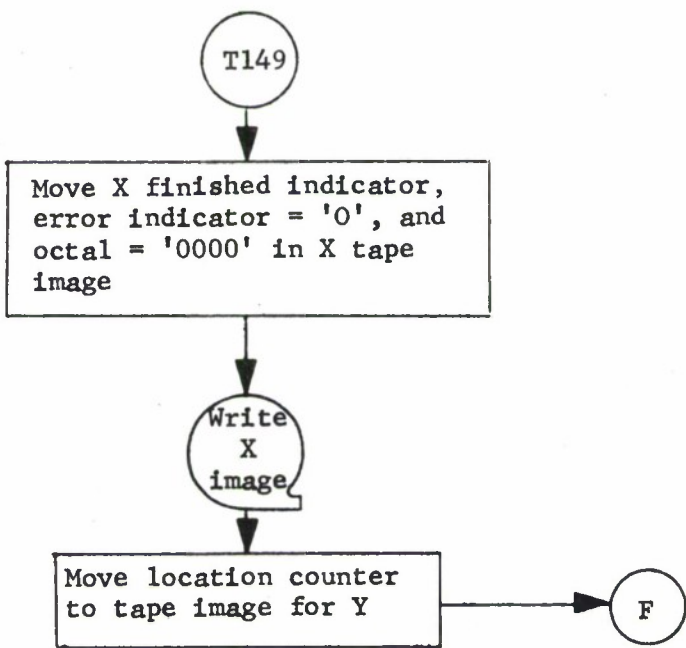
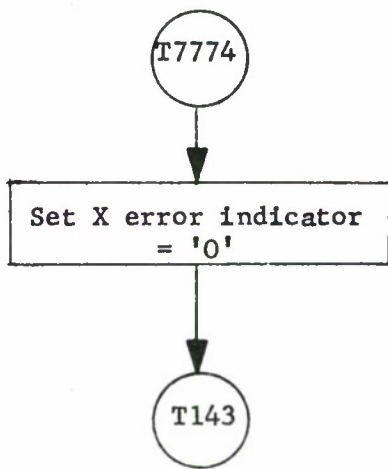
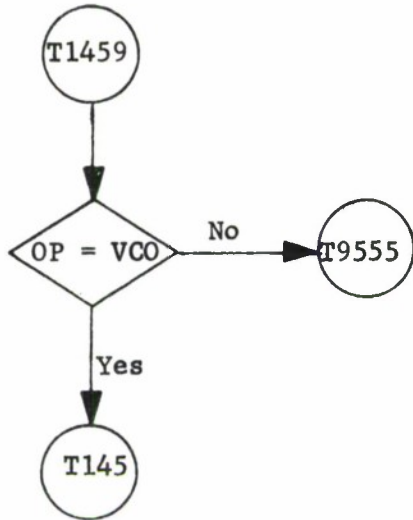
VCO Flow Chart
(1 of 9)

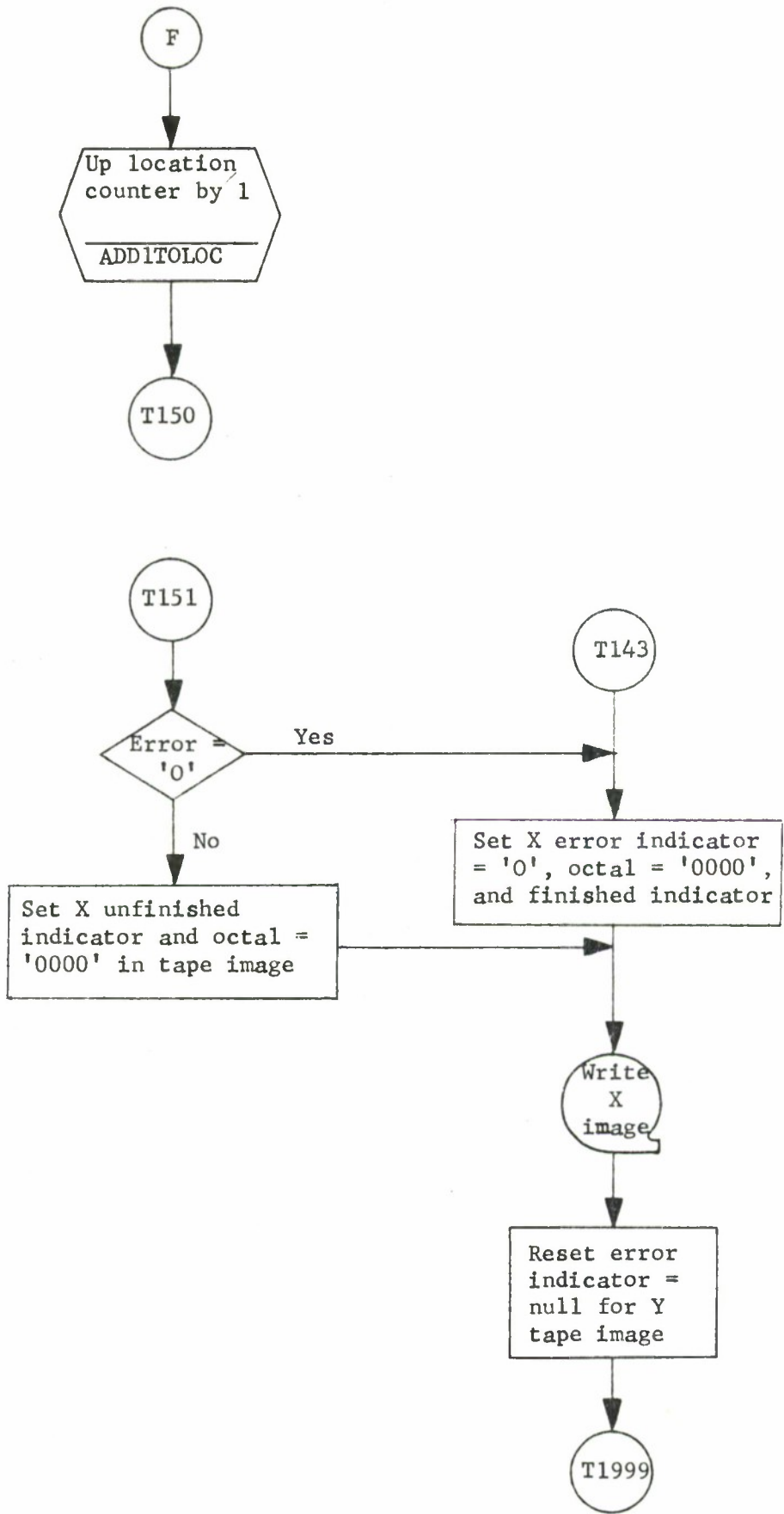


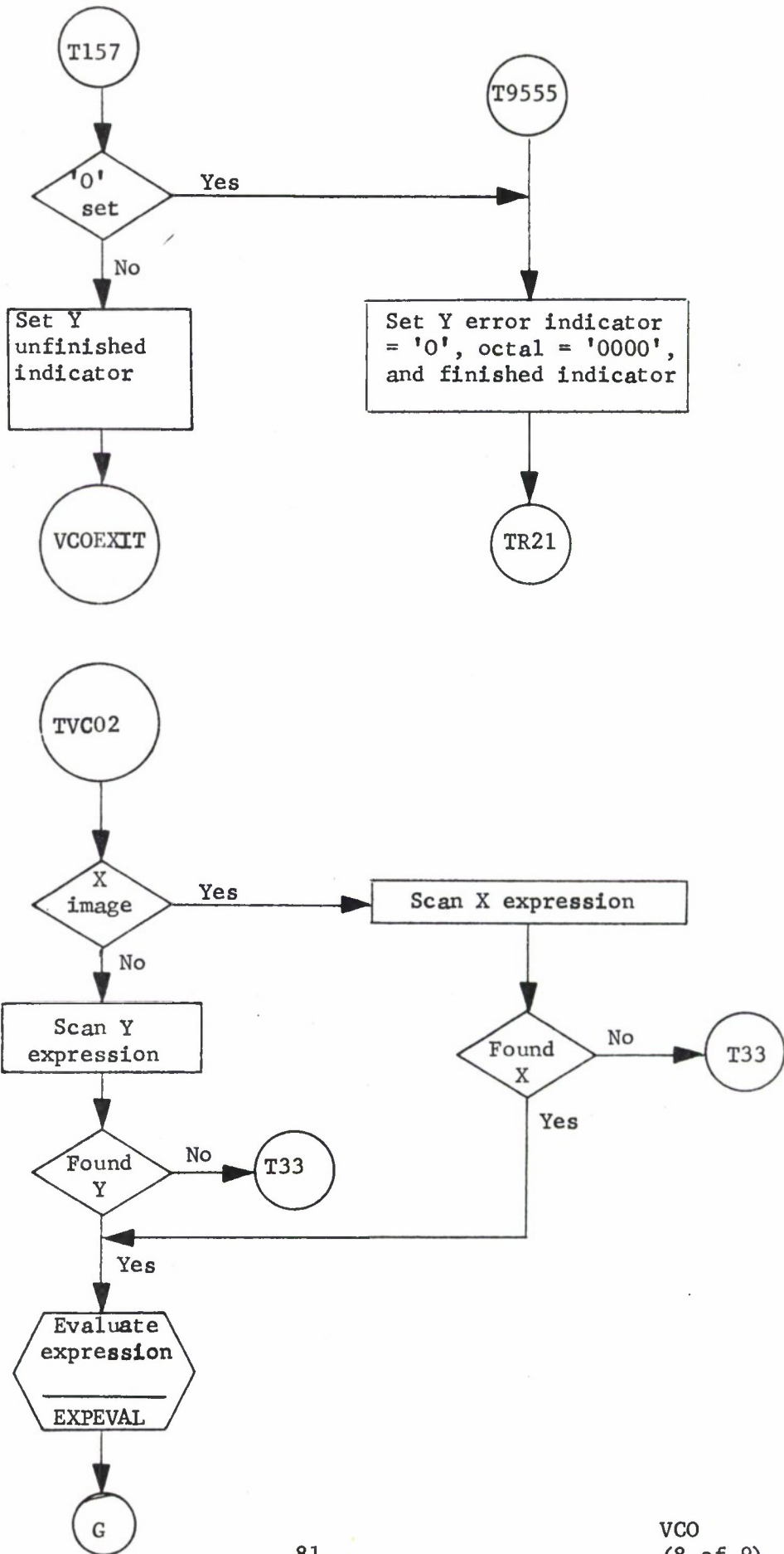


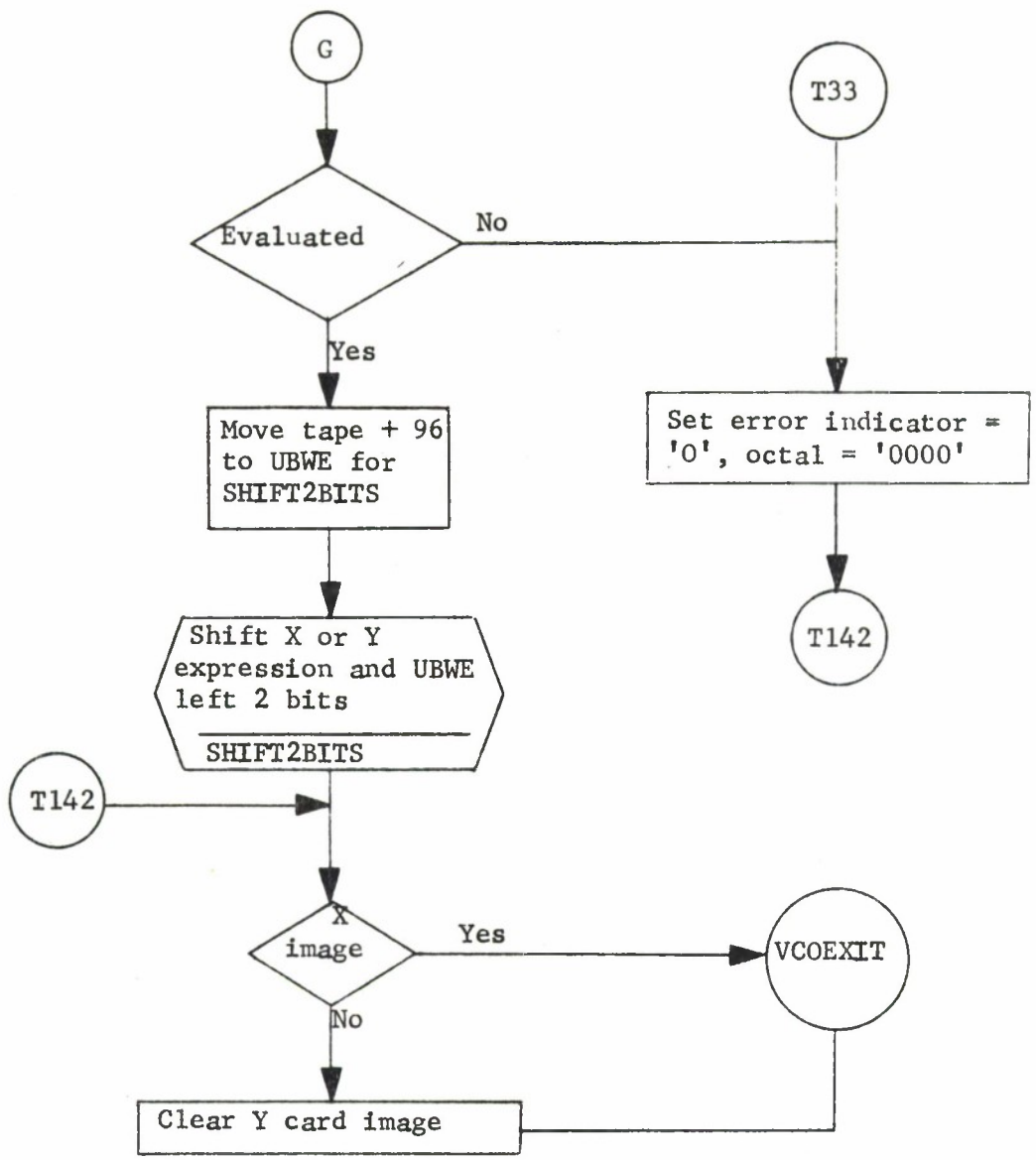












VCO
(9 of 9)

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) The MITRE Corporation Bedford, Massachusetts		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP N/A	
3. REPORT TITLE BR-90 ASSEMBLY PROGRAM - BRASS			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) N/A			
5. AUTHOR(S) (First name, middle initial, last name) J. A. Terrasi and J. C. Penney			
6. REPORT DATE June 1968		7a. TOTAL NO. OF PAGES 86	7b. NO. OF REFS --
8a. CONTRACT OR GRANT NO. AF 19(628)-5165		9a. ORIGINATOR'S REPORT NUMBER(S) ESD-TR-68-132	
b. PROJECT NO. 512V		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) MTR-597	
c.			
d.			
10. DISTRIBUTION STATEMENT This document has been approved for public release and sale; its distribution is unlimited.			
11. SUPPLEMENTARY NOTES N/A		12. SPONSORING MILITARY ACTIVITY Directorate of Planning and Technology, Electronic Systems Division, Air Force Systems Command, L. G. Hanscom Field, Bedford, Mass.	
13. ABSTRACT This document describes the features, user procedures, and program/coding specifications for the BR-90 Assembly Program (BRASS) which is operational on an IBM 1410 computer.			

KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Assembler BR-90 Communications Console IBM 1410 Oriented						