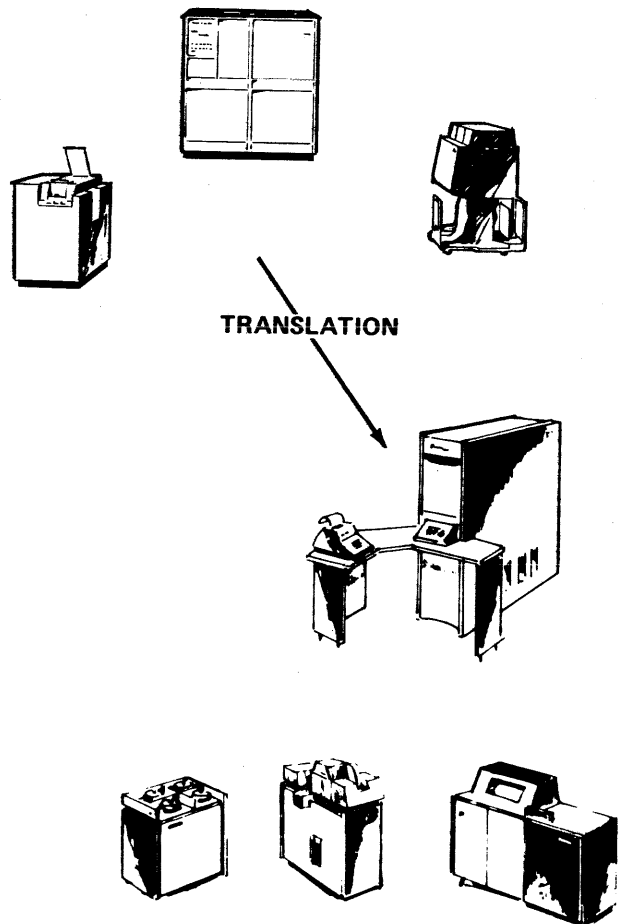




Burroughs

# B 2500 and B 3500



IBM 1401 ASSEMBLY LANGUAGE TRANSLATOR  
INFORMATION MANUAL

```

+++++
+++++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
++
+++++
+++++

```

Burroughs

B 2500 AND B 3500

IBM 1401 ASSEMBLY LANGUAGE TRANSLATOR  
INFORMATION MANUAL

DESCRIPTION AND OPERATING INSTRUCTIONS

MARCH 25, 1968

BUSINESS MACHINES GROUP  
SALES TECHNICAL SERVICES  
SYSTEMS DOCUMENTATION

COPYRIGHT © 1968  
BURROUGHS CORPORATION

BURROUGHS CORPORATION  
DETROIT, MICHIGAN 48232

Burroughs Corporation believes the program described in this manual to be accurate and reliable, and much care has been taken in its preparation. However, the Corporation cannot accept any responsibility, financial or otherwise, for any consequences arising out of the use of this material. The information contained herein is subject to change. Revisions may be issued to advise of such changes and/or additions.

## TABLE OF CONTENTS

INTRODUCTION. . . . .	PAGE	1
SECTION 1 TRANSLATOR FEATURES . . . . .	PAGE	3
1-1. GENERAL: . . . . .	PAGE	3
SECTION 2 INPUT CAPABILITIES . . . . .	PAGE	4
2-1. GENERAL: . . . . .	PAGE	4
2-2. CONTROL CARDS: . . . . .	PAGE	4
2-3. 1400 LABEL RESTRICTIONS: . . . . .	PAGE	4
SECTION 3 OUTPUT CAPABILITIES . . . . .	PAGE	6
3-1. GENERAL: . . . . .	PAGE	6
3-2. PROGRAM LISTING: . . . . .	PAGE	6
3-3. B3500 SYMBOLIC ASSEMBLY FILE: . . . . .	PAGE	6
3-4. SYMBOLIC MODIFICATION: . . . . .	PAGE	7
3-5. B3500 ASSEMBLE ERRORS: . . . . .	PAGE	9
SECTION 4 METHOD OF SPECIFICATION . . . . .	PAGE	10
4-1. GENERAL: . . . . .	PAGE	10
4-2. THE DOLLAR SIGN CARD: . . . . .	PAGE	10
4-2-1. INPUT OUTPUT MEDIA DESIGNATIONS: . . . . .	PAGE	10
4-2-2. 1400 PROGRAM PARAMETERS: . . . . .	PAGE	11
4-3. THE DEFINE CARD: . . . . .	PAGE	12
SECTION 5 FUNCTIONAL DESCRIPTION. . . . .	PAGE	16
5-1. GENERAL: . . . . .	PAGE	16
5-2. PHASE 1: . . . . .	PAGE	16
5-3. PHASE 2: . . . . .	PAGE	16
5-4. PHASE 3: . . . . .	PAGE	16
5-5. PHASE 4: . . . . .	PAGE	17
5-6. PHASE 5: . . . . .	PAGE	17
SECTION 6 OPERATING INSTRUCTIONS. . . . .	PAGE	19
6-1. GENERAL: . . . . .	PAGE	19
6-2. TRANSLATION: . . . . .	PAGE	19
6-3. LIBRARY MODIFICATION: . . . . .	PAGE	20
APPENDIX. . . . .	PAGE	24
A-1. TRANSLATOR EXECUTE DECK: . . . . .	PAGE	24
A-2. "AUTING" FILE: . . . . .	PAGE	25
A-3. OPERATION CODE LISTING. . . . .	PAGE	57

## INTRODUCTION

THE 1400/B3500 TRANSLATOR IS DESIGNED TO TRANSLATE PROGRAMS WRITTEN FOR IBM 1400 SERIES SYSTEMS TO B3500 ASSEMBLY LANGUAGE PROGRAMS. LANGUAGES ACCEPTABLE AS INPUT TO THE TRANSLATOR ARE AUTOCODER, BASIC AUTOCODER, AND SPS, AS IMPLEMENTED ON THE SYSTEMS 1401, 1440, AND 1460. THE 1400/B3500 TRANSLATOR IS INTENDED TO BE USED AS A TOOL IN MINIMIZING THE TIME AND COST OF CONVERTING PROGRAMS FROM 1400 OPERATION TO B3500 OPERATION. ALTHOUGH TRANSLATION DOES NOT ELIMINATE ALL THE PROBLEMS OF CONVERSION, IT CAN BE USED EFFECTIVELY TO SUBSTANTIALLY DIMINISH THE PROBLEMS OF REPROGRAMMING, CODING, AND KEYPUNCHING WHICH ACCOMPANY MANUAL TRANSLATION. THE TOTAL AMOUNT OF TIME REQUIRED TO CONVERT A PROGRAM FROM THE 1400 TO THE B3500 USING THIS TRANSLATOR DEPENDS ON PROGRAMMING TECHNIQUES EMPLOYED BY THE 1400 PROGRAMMER.

THE MINIMUM SYSTEM CONFIGURATION REQUIRED FOR USING THE 1400/B3500 TRANSLATOR IS AS FOLLOWS:

- |    |                           |   |
|----|---------------------------|---|
| A. | CORE                      | 25,000 CHARACTERS<br>PLUS MCP REQUIREMENTS.   |
| B. | DISK OR<br>SYSTEMS MEMORY | THE AREA REQUIRED IS DEPENDENT<br>UPON SIZE OF THE INPUT DECK AND<br>THE TYPE OF LANGUAGE. THESE ARE<br>THE VARIABLES:<br>500 SEGMENTS INITIALLY<br>THE WORK FILE IS BLOCKED 1000<br>1. 1 SEGMENT PER SPS STATEMENT<br>2. 2 SEGMENTS PER AUTOCODER<br>STATEMENT.<br>SYMDIN REQUESTED.<br>1. 1 SEGMENT PER ASSEMBLER<br>STATEMENT. |
| C. | CARD READER               | 1 FOR CONTROL CARD AND SPEC<br>CARD INPUT--ALSO FOR SOURCE  |

PROGRAM INPUT IF ON CARDS,

D. PRINTER

1 FOR LISTING.

E. MAGNETIC TAPE

1 FOR INPUT IF  
SOURCE PROGRAM ON TAPE.  
1 FOR OUTPUT IF SYMBOLIC  
OUTPUT TAPE REQUESTED.

F. CARD PUNCH

1 FOR OUTPUT IF SYMBOLIC  
CARD DECK REQUESTED.

THE SOFTWARE ENVIRONMENT REQUIRES AN MCP. IN ADDITION  
THE B3500 ADVANCE ASSEMBLER MAY BE REQUESTED DURING A  
TRANSLATION. THESE PROGRAMS ARE MAINTAINED ON THE STANDARD  
B3500 SYSTEMS TAPE.

## SECTION 1 TRANSLATOR FEATURES

1-1. GENERAL:  
-----

## FEATURES:

- A. ACCEPT AS INPUT EITHER AUTOCODER, BASIC AUTOCODER, OR SPS SOURCE PROGRAMS.
- B. SOURCE PROGRAM MAY BE READ IN FROM CARD READER OR MAGNETIC TAPE.
- C. SOURCE PROGRAMS MAY BE STACKED ON MAGNETIC TAPE FOR INPUT TO THE TRANSLATOR.
- D. THE TRANSLATOR B3500 SYMBOLIC OUTPUT MAY BE SPECIFIED AS PUNCHED CARDS, A SYMBOLIC MAGNETIC TAPE FILE LABELLED "SYMTIN", OR A SYMBOLIC DISK FILE LABELLED "SYMDIN".
- E. ALLOWS FOR OPTIONAL COMPILE OF THE TRANSLATED PROGRAM ON THE B3500 ADVANCED ASSEMBLER USING THE MCP ZIP COMMUNICATE.
- F. PROVIDES OPTIONAL INTERSPERSED LISTINGS OF THE 1400 AND B3500 SYMBOLIC RECORDS ON OUTPUT PLUS AN OPTIONAL LISTING OF THE SOURCE PROGRAM ON INPUT.
- G. ALLOWS USER TO APPEND LABELS TO MACHINE LANGUAGE ADDRESSES USED IN THE SOURCE PROGRAM MEMORY ALLOCATION AT TRANSLATE TIME THROUGH USE OF DEFINE "DFN" CARDS.
- H. PROVIDES ERROR FLAGS AND WARNINGS ON THE TRANSLATED PROGRAMS.
- I. SEQUENCES THE B3500 SYMBOLIC FILE IN INCREMENTS OF 100.
- J. SIMULATES 1400 COMPARISON INDICATOR AND OVERFLOW CONVENTIONS.
- K. ALLOWS FOR USER MODIFICATIONS OF TRANSLATOR EMITTED SUBROUTINES.
- L. RETAINS INTEGRITY OF 1400 CARD READER, CARD PUNCH, PRINTER MULTIPLE I-O CAPABILITY AND EXTENDS THIS CAPABILITY TO MAGNETIC TAPE.

## SECTION 2 INPUT CAPABILITIES

2-1. GENERAL:

THE 1400/B3500 TRANSLATOR PROCESSES A SYMBOLIC CARD DECK OR CARD-IMAGE TAPE FORMATTED ACCORDING TO THE STANDARDS OF THE LANGUAGE-AUTOCODER, SPS, OR BASIC AUTOCODER. THE AUTOCODER ASSEMBLER WILL ACCEPT AND/OR PRODUCE A SYMBOLIC TAPE WHICH MAY CONTAIN MULTIPLE AUTOCODER PROGRAMS. THIS "STACKED" AUTOCODER TAPE IS ACCEPTABLE AS INPUT TO THE 1400/B3500 TRANSLATER. THE TAPE PRODUCED BY THE AUTOCODER ASSEMBLER IS UNLABELLED AND OPERATOR INTERVENTION THROUGH THE SPD IS REQUIRED TO READ THE TAPE FILE.

2-2. CONTROL CARDS:

THE FOLLOWING CARDS ARE REQUIRED TO INITIATE THE TRANSLATION OF A 1400 PROGRAM,

? EXECUTE TR1400

? DATAB AUTO

SPECIFICATION CARDS (DESCRIBED IN SECTION 4)

SOURCE PROGRAM FOR TRANSLATION (OPTIONAL)

? END

2-3. 1400 LABEL RESTRICTIONS:

THE 1400/B3500 TRANSLATOR RETAINS THE ORIGINAL LABELS USED IN THE SOURCE 1400 PROGRAM. THEREFORE, THE APPEARANCE OF LABELS IN THE SOURCE 1400 PROGRAM WHICH HAVE A RESERVED MEANING TO THE B3500 ASSEMBLER OR ARE USED IN THE SYMBOLIC SUBROUTINES GENERATED BY THE TRANSLATOR MAY LEAD TO SYNTAX ERRORS DURING THE COMPILE (ASSEMBLY) ON THE B3500. THESE RESERVED IDENTIFIERS ARE:

IX1

IX2

IX3



BASE

ANY LABEL CONTAINING "X\*", "F\*", OR "R\*" AS ITS  
FIRST TWO CHARACTERS MAY ALSO LEAD TO LABEL  
QUALIFICATION PROBLEMS

## SECTION 3 OUTPUT CAPABILITIES

**3-1. GENERAL:**  
-----

THE 1400/B3500 TRANSLATOR ALLOWS THE OUTPUT MEDIA TO BE PRINTED COPY, MAGNETIC TAPE, PUNCHED CARDS, OR DISK. THE PRINTED COPY (PROGRAM LISTING) MAY BE SUPPRESSED, SPECIFIED EXCLUSIVELY, OR SPECIFIED IN COMBINATION WITH THE TRANSLATED SYMBOLIC FILE GOING TO ANY ONE OF THE OTHER OUTPUT MEDIA.

**3-2. PROGRAM LISTING:**  
-----

THE 1400/B3500 TRANSLATOR PROVIDES THREE OPTIONS IN SPECIFYING A PROGRAM LISTING. THE SOURCE 1400 PROGRAM MAY BE LISTED ON INPUT. THE TRANSLATED B3500 SYMBOLIC FILE AND/OR THE INPUT PROGRAM MAY BE LISTED ON OUTPUT. SHOULD LISTING OF BOTH OF THE LATTER BE SPECIFIED, A LISTING OF SOURCE 1400 SYMBOLIC RECORDS INTERSPERSED WITH THE TRANSLATED SYMBOLIC RECORDS WILL BE PRODUCED. THE INTERSPERSED LISTING WILL PRECEDE THE LABEL FIELD OF THE SOURCE RECORD BY AN "\*" TO ENHANCE THE READABILITY OF THE LISTING. THE FORMAT OF THESE LISTINGS IS SIMILAR TO THOSE DESCRIBED IN THEIR RESPECTIVE ASSEMBLER MANUALS.

**3-3. B3500 SYMBOLIC ASSEMBLY FILE:**  
-----

THE B3500 SYMBOLIC ASSEMBLY FILE MAY BE RECORDED ON DISK, MAGNETIC TAPE, OR PUNCHED CARDS. THE LABEL ID FOR THE DISK FILE IS "SYMDIN" AND FOR THE TAPE FILE, "SYMTIN". THE FORMAT OF THESE FILES IS GIVEN IN THE B3500 ADVANCED ASSEMBLER MANUAL.

3-4, SYMBOLIC MODIFICATION:

THE B3500 SYMBOLIC PROGRAM GENERATED BY THE 1400/B3500 TRANSLATOR MAY CONTAIN VARIOUS STANDARD SUBROUTINES EMITTED BY THE TRANSLATOR. THESE SUBROUTINES ARE GENERATED TO ACCOMODATE TRANSLATION OF INSTRUCTIONS WHICH WOULD BE UNWIELDY TO TRANSLATE INTO IN-LINE CODE. PROVISION HAS BEEN MADE FOR USER MODIFICATION OF THE LIBRARY OF EMITTED SUBROUTINES. THIS LIBRARY IS ON THE 1400/B3500 SYSTEM TAPE AND MUST BE LOADED BEFORE BEGINNING A TRANSLATION. THE IDENTIFICATION OF THE SYMBOLIC LIBRARY FILE IS "AUTIMG". A LISTING OF THIS FILE APPEARS IN THE APPENDIX.

THE FIRST THREE CHARACTERS OF EVERY RECORD ON THE AUTIMG FILE IS A SEQUENTIAL IDENTIFICATION NUMBER. THE SUBROUTINES TO BE EMITTED FROM THIS FILE AND THEIR STARTING AND ENDING IDENTIFICATION NUMBERS ARE LISTED BELOW. CARDS WITH AN ASTERISK IN COLUMN 4 ARE TREATED AS FILLER AND IGNORED DURING TRANSLATION.

001	THRU	099	ERROR, WARNING, AND DOCUMENTATION MESSAGES
100	THRU	120	CARD READ SUBROUTINE
121	THRU	200	ADD-SUB SUBROUTINE
201	THRU	220	HALT SUBROUTINE
221	THRU	230	RESERVED CORE CONSTANTS
231	THRU	300	COMPARE SUBROUTINE
301	THRU	330	MOVE-CHAR-TO-RECORD- MARK SUBROUTINE
331	THRU	360	MOVE SUBROUTINE (MOVES OVER 100)
361	THRU	395	MOVE CHARACTERS AND SUPPRESS ZEROS ROUTINE
396	THRU	505	MOVE CHARACTERS AND EDIT SUBROUTINE

506	THRU	530	PRINT ROUTINE FOR OTHER THAN SINGLE SPACING
531	THRU	560	PRINT ROUTINE FOR TRIPLE SPACING
561	THRU	630	MULTIPLY SUBROUTINE
631	THRU	700	DIVIDE SUBROUTINE
701	THRU	740	CONVERT 1400 ADDRESS TO DECIMAL ADDRESS
741	THRU	770	DETERMINE CORRECT INDEX REGISTER VALUE
771	THRU	790	POSITION PRINTER-NO TRIPLE SPACING
791	THRU	820	POSITION PRINTER-WITH TRIPLE SPACING

TO MODIFY OR REPLACE A SUBROUTINE LISTED ABOVE THE "AUTIMG" FILE MUST BE CONVERTED TO CARDS. THE DESIRED CHANGES ARE THEN MADE TO THE CARDS. ALL SUBROUTINES MUST FALL WITHIN THE RANGE GIVEN ABOVE. WHEN CHANGES TO THE CARD DECK ARE MADE THE "AUTIMG" DISK FILE IS RECREATED. THERE IS A PROGRAM CALLED "TRNLIB" ON THE 1400/B3500 SYSTEM TAPE TO FACILITATE THIS TYPE OF MODIFICATION. THE INSTRUCTIONS FOR USING "TRNLIB" ARE GIVEN IN THE CHAPTER ON OPERATING INSTRUCTIONS.

WHEN MODIFYING THE B3500 SYMBOLIC ASSEMBLY FILE THE MEMORY ALLOCATION OF THE ASSEMBLED PROGRAM MUST BE CONSIDERED. THE RESERVED MEMORY ALLOCATION IS GIVEN BELOW:

CORE LOCATION	USE
1-7	INDIRECT FIELD LENGTHS
8-15	INDEX REGISTER 1
16-23	INDEX REGISTER 2
24-31	INDEX REGISTER 3
32-39	INDIRECT FIELD LENGTHS

40-45	PROGRAM STACK POINTER
46-47	HALT BREAKPOINT MASK
48-64	EDIT TABLE
65-71	SENSE SWITCHES; A (LAST CARD SWITCH), B THROUGH G.

**3-5, B3500 ASSEMBLE ERRORS:**  
-----

THE B3500 SYMBOLIC FILE GENERATED BY THE 1400/B3500 TRANSLATOR MAY WHEN COMPILED (ASSEMBLED) GIVE SYNTAX ERRORS. CONDITIONS LEADING TO THESE SYNTAX ERRORS ARE LISTED BELOW:

1. REFERENCE TO LABELS WITHIN NON-TRANSLATED MACROS.
2. REFERENCE TO FILE NAMES NOT DETECTED BY TRANSLATOR.

## SECTION 4 METHOD OF SPECIFICATION

4-1. GENERAL:

THE 1400/B3500 TRANSLATOR REQUIRES A SET OF SPECIFICATION CARDS FOR EACH 1400 PROGRAM TO BE TRANSLATED. THERE ARE TWO TYPES OF SPECIFICATION CARDS REQUIRED BY THE TRANSLATOR, DOLLAR SIGN CARDS AND DEFINE CARDS.

4-2. THE DOLLAR SIGN CARD:

THE DOLLAR SIGN CARD RECEIVES ITS NAME FROM THE "\$" IN CARD COLUMN ONE USED TO IDENTIFY IT. THE 1400/B3500 TRANSLATOR DOLLAR SIGN CARD IS USED FOR DESIGNATION OF INPUT OUTPUT MEDIA AND 1400 PROGRAM PARAMETERS. IF THERE ARE MULTIPLE "\$" CARDS ONLY THE LAST ONE IS CONSIDERED. THE DOLLAR SIGN CARD IS A FREE FORM SPECIFICATION CARD.

4-2-1. INPUT OUTPUT MEDIA DESIGNATIONS:

THE INPUT OUTPUT MEDIA DESIGNATION OPTIONS ARE LISTED BELOW WITH MUTUALLY EXCLUSIVE OPTIONS LISTED ON THE SAME LINE. IF MORE THAN ONE OF THESE MUTUALLY EXCLUSIVE OPTIONS APPEARS ON A DOLLAR SIGN CARD ONLY THE LAST IS CONSIDERED. THE DEFAULT OPTIONS ARE INDICATED IN PARENTHESES.

1. CARDIN TAPEIN (CARDIN ASSUMED IF NEITHER IS SPECIFIED).
2. CARDOT TAPEOT DISKOT (NO FILE IS GENERATED IF NO FILE IS SPECIFIED).
3. LISTINPUT (ASSUMED OFF ON DEFAULT).
4. 1400 (ASSUMED OFF ON DEFAULT).
5. B3500 (ASSUMED OFF ON DEFAULT).
6. ASSEMBLE (NO ASSEMBLE ON DEFAULT).

THE INPUT MEDIA OPTIONS, CARDIN AND TAPEIN, SPECIFY THE INPUT MEDIA FOR THE SOURCE 1400 PROGRAM. CARDOT, TAPEOT, AND DISKOT LIKEWISE SPECIFY THE OUTPUT MEDIA FOR THE TRANSLATED B3500 SYMBOLIC ASSEMBLY FILE. THE APPEARANCE OF LISTINPUT ON THE DOLLAR SIGN CARD CALLS FOR LISTING OF SOURCE PROGRAM ON INPUT, 1400 AND B3500 ON THE DOLLAR SIGN CARD INDICATE LISTING OF THE REARRANGED 1400 CODE AND THE B3500 SYMBOLIC OUTPUT RESPECTIVELY. IF BOTH 1400 AND B3500 APPEAR ON THE DOLLAR SIGN CARD AN INTERSPERSED LISTING WILL BE PRODUCED.

IF THE "ASSEMBLE" OPTION IS SPECIFIED THE TRANSLATOR WILL "ZIP" TO THE ADVANCED ASSEMBLER FOR A COMPILATION TO LIBRARY OF THE B3500 SYMBOLIC RECORDS. THE ID OF THE ASSEMBLED PROGRAM IS SET TO THE ID SPECIFIED FOR THE PROGRAM ON THE DOLLAR SIGN CARD. IF NO ID IS SPECIFIED THE PROGRAM WILL BE ASSEMBLED UNDER THE ID "XX".

**4-2-2. 1400 PROGRAM PARAMETERS:**  
 -----

THERE ARE ESSENTIALLY FOUR 1400 PROGRAM PARAMETERS WHICH MUST BE FURNISHED TO THE 1400/B3500 TRANSLATOR. THESE PARAMETERS ARE LISTED BELOW WITH MUTUALLY EXCLUSIVE OPTIONS APPEARING ON THE SAME LINE. THE DEFAULT OPTIONS ARE INDICATED IN PARENTHESES.

1. "XXXXX" (ON DEFAULT, ID = SPACES).
2. AUTO SPS BASIC (AUTOCODER ASSUMED IF NONE SPECIFIED).
3. 1401 1440 1460 (1401 ASSUMED IF NONE SPECIFIED).
4. BYPASS (OFF ON DEFAULT).

THE PARAMETER REPRESENTED BY "XXXXX" IS THE PROGRAM

IDENTIFICATION FOR THE 1400 PROGRAM AND THE TRANSLATED B3500 ASSEMBLY PROGRAM. THIS PARAMETER MUST BE CODED AS A FIVE CHARACTER IDENTIFIER ENCLOSED IN QUOTES. THE PARAMETERS AUTO, SPS, AND BASIC INDICATE THE LANGUAGE THE 1400 PROGRAM IS CODED IN; AUTOCODER, SPS, OR BASIC AUTOCODER RESPECTIVELY. 1401, 1440, AND 1460 SPECIFY WHICH OF THE THREE SYSTEMS THE 1400 PROGRAM WAS DESIGNED TO RUN ON. BYPASS IS A SPECIAL PARAMETER WHICH IS APPLICABLE ONLY WHEN TAPE INPUT HAS BEEN SPECIFIED. WHEN BYPASS IS SET ON, THE TRANSLATOR BYPASSES THE REMAINDER OF THE SPECIFICATION CARDS (IF SPECIFICATION CARDS ARE PRESENT) FOR THAT PROGRAM, POSITIONS THE INPUT TAPE AT THE BEGINNING OF THE NEXT PROGRAM, AND BEGINS READING SPECIFICATION CARDS FOR THE NEW PROGRAM. THE PROGRAM BYPASS FEATURE HAS BEEN IMPLEMENTED TO PERMIT SELECTIVELY TRANSLATING PROGRAMS FROM A STACKED INPUT TAPE.

#### 4-3. THE DEFINE CARD: -----

THE DEFINE CARD IS A CODING CONVENTION IMPLEMENTED IN THE 1400/B3500 TRANSLATOR TO SERVE TWO BASIC PURPOSES. FIRST- THE DEFINE CARD IS USED TO PASS INFORMATION TO THE TRANSLATOR ABOUT THE PERIPHERALS USED BY THE PROGRAM TO BE TRANSLATED AND, SECOND- THE DEFINE CARD IS USED TO EQUATE A LABEL TO A BLOCK OF MEMORY IN THE 1400 PROGRAM. THE DEFINE CARD IS PUNCHED IN THE STANDARD FORMAT OF SPS. ONE DEFINE ENTRY IS REQUIRED FOR EACH I-O DEVICE USED BY THE 1400 PROGRAM. THE FORMAT OF THE DEFINE CARD IS GIVEN BELOW.

DEFINE CARD ENTRIES WHICH ARE INTENDED TO APPEND A LABEL TO A BLOCK OF MEMORY CONTAIN ALL BLANKS EXCEPT:



COL 08-13	ANY LABEL
COL 14-16	LITERAL "DFN"
COL 17-21	STARTING LOCATION OF AREA TO BE DEFINED
COL 28-32	ENDING LOCATION OF AREA TO BE DEFINED

**NOTE:**

-----

THE PURPOSE OF THE AREA-DEFINING ENTRY IS TO EQUATE 1400 PROGRAM AREAS WHICH HAVE BEEN ADDRESSED EXCLUSIVELY IN MACHINE LANGUAGE TO A LABEL SO THEY MAY BE TRANSLATED. THE TRANSLATOR WILL CREATE A DATA DECLARATION (CNST) FOR EACH LABEL SO DEFINED. THUS ALL MACHINE LANGUAGE REFERENCES TO THIS AREA WILL BE TRANSLATED TO THE FORMAT LABEL + INCREMENT. THIS FUNCTION MAY ALSO BE USEFUL WHERE EQU OR DS STATEMENTS HAVE BEEN USED TO EQUATE AUTOCODER OR SPS LABELS TO MACHINE LANGUAGE ADDRESSES WITHOUT RESERVING CORE. IN THIS CASE THE "DFN" ENTRY WILL CAUSE A DATA DECLARATION TO BE INSERTED IN THE PROGRAM. SUBSEQUENT DS OR EQU STATEMENTS REFERENCING MACHINE LANGUAGE ADDRESSES WITHIN THIS DEFINED AREA WILL THEN BE EQUATED TO THE "DEFINED" LABEL PLUS AN APPROPRIATE INCREMENT. IN THIS WAY REFERENCE TO THE EQU OR DS LABELS MAY BE PROPERLY TRANSLATED.

DEFINE CARD ENTRIES WHICH DEFINE FILES ARE FORMATTED AS FOLLOWS:

COL 08-13	ANY RESERVED FILE NAME
COL 14-16	LITERAL "DFN"
COL 17-21	STARTING LOCATION OF FILE I-O AREA
COL 28-32	ENDING LOCATION OF FILE I-O AREA
COL 40-45	FILE LABEL ID

**NOTE:**  
**-----**

THE RESERVED FILE NAME MENTIONED ABOVE MUST BE FORMATTED AS FOLLOWS:

THE FIRST TWO CHARACTERS IDENTIFY THE DEFINE CARD TO BE DEFINING A FILE AND MUST BE CODED AS "F\*". CHARACTERS 3-5 OF THE FILE NAME ARE THE B3500 HARDWARE MNEMONIC DESIGNATION FOR THE PERIPHERAL DEVICE ASSOCIATED WITH THE FILE, I.E.

MTP = MAGNETIC TAPE

DSK = DISK

CRD = CARD READER

CPU = CARD PUNCH

PRN = PRINTER

THE LAST CHARACTER IN THE FILE NAME IS THE 1400 LOGICAL UNIT NUMBER OF THE I-O DEVICE.

**4-4. THE DEFINE END CARD:**  
**-----**

EACH SET OF PROGRAM SPECIFICATION CARDS (DOLLAR SIGN AND/OR DEFINE CARDS) MUST BE FOLLOWED BY A DEFINE END CARD. THE DEFINE END CARD (IDENTIFIED BY "DFE" IN COLUMNS 14-16) IS USED TO TERMINATE READING OF SPECIFICATION CARDS AND INITIATE PROCESSING OF THE 1400 PROGRAM FROM EITHER CARDS OR MAGNETIC TAPE.

**4-5. SPECIFICATION CARD ORDER:**  
**-----**

THE ORDER OF PROGRAM SPECIFICATION CARDS IS AS FOLLOWS:

- A. DOLLAR SIGN CARD(S). IF MORE THAN ONE IS PRESENT ONLY THE LAST ONE IS EFFECTIVE.
- B. DEFINE CARD(S).
- C. THE DEFINE END CARD.

IF THE 1400 PROGRAM TO BE TRANSLATED IS TO BE INPUTTED FROM THE CARD READER THE PROGRAM

DECK(S) MUST IMMEDIATELY FOLLOW ITS RESPECTIVE PROGRAM SPECIFICATION CARDS. THERE IS NO SPECIAL CARD REQUIRED TO INDICATE END OF TRANSLATOR INPUT. THIS FUNCTION IS SIMULATED THROUGH THE NORMAL END OF FILE ACTION ON THE CARD READER. THE 1400 LANGUAGE END CARD SIGNALS THE END OF THE INPUT SYMBOLIC DECK. IT MUST BE PRESENT FOR EACH SYMBOLIC DECK TO BE TRANSLATED.

IN FILLING OUT PROGRAM SPECIFICATION CARDS IT SHOULD BE REMEMBERED THAT WHILE NO "DFN" CARDS ARE REQUIRED BY THE TRANSLATOR, THE TRANSLATOR DOES REQUIRE AT LEAST ONE "\$" CARD AND ONLY ONE "DFE" CARD IN EACH SET OF PROGRAM SPECIFICATION CARDS.

## SECTION 5 FUNCTIONAL DESCRIPTION

**5-1. GENERAL:**  
**-----**

THE 1400/B3500 TRANSLATOR IS A FIVE PHASE PROGRAM TO TRANSLATE AUTOCODER, BASIC AUTOCODER, OR SPS AS WRITTEN FOR THE IBM SERIES 1401, 1440, OR 1460 SYSTEMS TO B3500 ASSEMBLY LANGUAGE. THE 1400/B3500 TRANSLATOR IS PROGRAMMED TO RUN ON THE B3500 SYSTEM. THE FUNCTION OF EACH OF THE PHASES OF THE TRANSLATOR IS DESCRIBED BELOW.

**5-2. PHASE 1:**  
**-----**

THE FIRST PHASE OF THE 1400/B3500 TRANSLATOR CONVERTS THE INPUT PROGRAM TO A FIXED FORM SPS-LIKE LANGUAGE WHICH IS THEN UTILIZED INTERNALLY FOR TRANSLATION. AN OPTIONAL LISTING OF THE INPUT PROGRAM IS AVAILABLE AT THIS TIME.

**5-3. PHASE 2:**  
**-----**

THE SECOND PHASE OF THE TRANSLATOR PASSES THE INTERNAL REPRESENTATION OF THE INPUT PROGRAM CREATED BY PHASE 1 AND BUILDS A MAP OF THE 1400 MEMORY ALLOCATION PLUS A LABEL TABLE. THE LABEL TABLE CONTAINS AN ENTRY FOR EACH LABEL IN THE SOURCE PROGRAM PLUS ANY LABELS GENERATED BY THE TRANSLATOR. PHASE 2 APPENDS TO EACH ENTRY IN THE LABEL TABLE INFORMATION REQUIRED IN LATER PHASES OF TRANSLATION.

**5-4. PHASE 3:**  
**-----**

PHASE 3 DOES A LOOKUP ON THE LABEL TABLE FOR EACH LABEL REFERENCE IT ENCOUNTERS IN THE INPUT PROGRAM

AND AFFIXES PERTINENT INFORMATION IN THE LABEL TABLE TO THE LABEL REFERENCE. THIS IS DONE TO DETERMINE INFORMATION ABOUT THE ADDRESSES REFERENCED IN EACH INSTRUCTION. THE THIRD PHASE ALSO UPDATES THE MEMORY MAP BUILT IN PHASE 2 WITH WORD MARKS IMPLIED BY INSTRUCTION LOCATIONS AND DATA DEFINING STATEMENTS. THIS MAP OF MEMORY ALLOCATION PLUS WORD MARK LOCATIONS IS THEN WRITTEN TO DISK.

5-5.    PHASE 4:  
-----    -----    --

THE FOURTH PHASE OF TRANSLATION PASSES THE INTERNAL SOURCE PROGRAM AND APPENDS FIELD LENGTHS TO THE ADDRESSES REFERENCED BY EACH INSTRUCTION. THIS IS DONE ONLY WHERE IT IS DETERMINED THE OUTPUT B3500 INSTRUCTION WILL REQUIRE FIELD LENGTHS AND IS ACCOMPLISHED BY ANALYZING WORD MARK LOCATIONS IN THE PREVIOUSLY CONSTRUCTED MEMORY MAP.

NOTE:  
-----

IT SHOULD BE REMEMBERED THAT THE 1400/B3500 TRANSLATOR DOES NOT SIMULATE THE LOGIC FLOW OF THE SOURCE PROGRAM BUT RATHER TRANSLATES EACH INSTRUCTION AS ENCOUNTERED IN THE INPUT PROGRAM. THEREFORE SW, CW AND OTHER INSTRUCTIONS WHICH AFFECT WORD MARK LOCATIONS WILL CHANGE WORD MARKS IN THE TRANSLATOR MEMORY ALLOCATION SCHEME AS ENCOUNTERED IN THE INPUT PROGRAM. THE EXTENT TO WHICH THIS AFFECTS THE COMPLETENESS OF TRANSLATION IS DEPENDENT UPON THE PROGRAMMING TECHNIQUES EMPLOYED.

5-6.    PHASE 5:  
-----    -----    --

THE FIFTH PHASE OF THE TRANSLATOR TAKES THE INTERNAL REPRESENTATION OF THE 1400 PROGRAM AS UPDATED BY THE PREVIOUS PHASES OF TRANSLATION

AND PRODUCES B3500 SYMBOLIC CODE. THIS PHASE  
ALSO PRODUCES THE OPTIONAL INTERSPERSED LISTING  
DESCRIBED IN PARAGRAPH 4-2-1.

## SECTION 6 OPERATING INSTRUCTIONS

6-1. GENERAL:

OPERATING INSTRUCTIONS FOR THE 1400/B3500 TRANSLATOR ARE SUBDIVIDED INTO TWO CATEGORIES; THE INSTRUCTIONS NECESSARY TO EFFECT THE ACTUAL RUNNING OF THE TRANSLATOR PROGRAM AND THE INSTRUCTIONS REQUIRED FOR MODIFYING THE LIBRARY OF TRANSLATOR EMITTED WARNING MESSAGES AND SUBROUTINES.

6-2. TRANSLATION:

IN ORDER TO RUN THE 1400/B3500 TRANSLATOR THE FOLLOWING STEPS SHOULD BE FOLLOWED:

- A. FILL OUT PROGRAM SPECIFICATION CARDS AS OUTLINED IN SECTION 4.
- B. MOUNT THE AUTOCODER INPUT TAPE IF TAPE INPUT HAS BEEN SPECIFIED.
- C. PREPARE THE EXECUTE DECK AS SPECIFIED IN PARAGRAPH 2-2. THE OPTIONAL 1400 SOURCE PROGRAM DECK MUST NOT BE PRESENT IF TAPE INPUT HAS BEEN SPECIFIED.
- D. PLACE EXECUTE DECK IN CARD READER AND READ.
- E. IF TAPE INPUT HAS BEEN SPECIFIED FOR THE PROGRAM(S) TO BE TRANSLATED THEN IMMEDIATELY FOLLOWING THE READING OF THE PROGRAM SPECIFICATION CARDS THE FOLLOWING MESSAGE WILL APPEAR ON THE SPO - "NO FILE AUTOTP TR1400 (MIX INDEX)". THE OPERATOR SHOULD THEN SPECIFY THE UNIT ON WHICH HE HAS MOUNTED THE INPUT TAPE BY KEYING IN THE FOLLOWING:  
(MIX INDEX) UL (I-O CHANNEL NO)/(UNIT NO).

- F. IF ONE OR MORE PROGRAMS HAVE SPECIFIED SYMBOLIC TAPE "SYMTIN" OUTPUT THE OPERATOR MUST MAKE AVAILABLE TO THE SYSTEM A SCRATCH TAPE EACH TIME THE SPO MESSAGE "MT RQD SYMTIN TR1400 (MIX INDEX)" APPEARS.
- G. IF MORE THAN ONE PROGRAM CALLS FOR A SYMBOLIC DISK FILE "SYMDIN" AS OUTPUT A "DUPLICATE LIBRARY" MESSAGE WILL BE DISPLAYED ON THE SPO EACH TIME A NEW "SYMDIN" FILE IS CREATED (EXCEPT FOR THE FIRST TIME THE FILE IS CREATED). THIS MESSAGE TAKES THE FORM OF "DUP LIBRARY SYMDIN TR1400 (MIX INDEX)" IF THE EXISTING "SYMDIN" FILE IS NOT IN USE THE OPERATOR SHOULD CHANGE IT TO A NEW NAME, E.G. KEY IN "CC CHANGE SYMDIN TO NEWID" AND THEN KEY IN "(MIX INDEX) OK".
- H. UPON TRANSLATING THE FINAL PROGRAM THE TRANSLATOR WILL GO TO A NORMAL END OF JOB.

### 6-3. LIBRARY MODIFICATION:

IT IS ANTICIPATED THAT SOME USERS OF THE 1400/83500 TRANSLATOR MAY DESIRE TO MODIFY OR ADD TO THE LIBRARY OF TRANSLATOR GENERATED SUBROUTINES TO FACILITATE TRANSLATION OF PROGRAMS WHICH EMPLOY SPECIALIZED 1400 PROGRAMMING TECHNIQUES. PROVISION HAS BEEN MADE FOR SUCH MODIFICATION AS EXPLAINED IN PARAGRAPH 3-4. THE OPERATING INSTRUCTIONS FOR EFFECTING SUCH A MODIFICATION FOLLOW.

- A. LOAD FROM THE "AUTSYS" SYSTEM TAPE THE AUTIMG (SYMBOLIC CARD) FILE AND THE PROGRAM TRNLIB, E.G. KEY IN SPU



- "CC LOAD FROM AUTSYS AUTIMG, TRNLIB",
- B. EXECUTE PROGRAM CALLED TRNLIB, E,G,  
KEY IN SPO  
"CC EXECUTE TRNLIB".
- C. IMMEDIATELY FOLLOWING BOJ TRNLIB WILL  
DISPLAY THE FOLLOWING MESSAGE ON THE SPO:  
"DUMP,CREATE,DUMP AUT,CREATE AUT,  
DUMP SOLT OR CREATE SOLT"  
THIS MESSAGE INDICATES THE POSSIBLE  
USES OF TRNLIB.  
THE PROGRAM TRNLIB WILL THEN WAIT ON AN  
ACCEPT STATEMENT. SINCE THE PURPOSE OF  
THIS RUN IS TO OBTAIN A CARD DECK PLUS  
LISTING OF THE AUTIMG DISK FILE,  
KEY IN SPO  
"(MIX INDEX) AXDUMP AUT".
- D. AFTER PRODUCING A CARD DECK PLUS LISTING  
OF THE AUTIMG FILE TRNLIB WILL GO TO A  
NORMAL END OF JOB.
- E. AT THIS POINT THE PROGRAMMER IS READY TO  
INSERT HIS OWN MODIFICATION INTO THE  
SUBROUTINES. THE FORMAT OF THE CARDS IN  
THE AUTIMG FILE SUBROUTINES IS THE SAME  
AS THE STANDARD B3500 ADVANCED ASSEMBLER  
REQUIRES EXCEPT:
1. POSITIONS 1-3 CONTAIN THE SEQUENCE  
NUMBER OF THE CARD WITHIN THE AUTIMG  
FILE.
  2. POSITIONS 4-7 ARE BLANK.

**NOTE:**  
-----

AREA WITHIN THE SYMBOLIC  
SUBROUTINES IS OFTEN RESERVED FOR  
MODIFICATION BY INSERTING BLANK  
FILLER CARDS WITHIN THE SUBROUTINES.  
THESE BLANK FILLER CARDS ARE  
IDENTIFIED BY AN ASTERISK IN

## COLUMN 4.

- F. SEQUENCE NUMBERS FOR MODIFYING THE AUTIMG SYMBOLIC FILE SHOULD BE OBTAINED BY EXAMINING THE LISTING OF THE FILE PRODUCED BY TRNLIB IN CONJUNCTION WITH THE GUIDELINES FOUND IN PARAGRAPH 3-4 UN B3500 SYMBOLIC MODIFICATION.
- G. WHEN THE PROGRAMMER HAS FINISHED MUDIFYING THE AUTIMG DECK AND IS PREPARED TO RECREATE THE AUTIMG DISK FILE HE SHOULD
1. EXECUTE TRNLIB AS EXPLAINED IN PARAGRAPH 6-3-B.
  2. KEY IN SPO  
"(MIX INDEX) AXCREATE AUT"  
AS PER 6-3-C.
  3. PLACE NEW AUTIMG CARDS IN READER PRECEDED BY A "DATAB GENCRD" CONTROL CARD AND TERMINATED BY AN "END" CONTROL CARD.
- H. UPON CREATING THE NEW AUTIMG FILE TRNLIB WILL WAIT ON A "DUPLICATE LIBRARY" CONDITION. THE OPERATOR SHOULD REMOVE THE OLD AUTIMG FILE BY KEYING IN SPO  
"(MIX INDEX) RM".  
TRNLIB WILL THEN GO TO A NORMAL END OF JOB.

NOTE:

IF IT IS DESIRED TO CREATE A NEW SYSTEM TAPE AFTER UPDATING "AUTIMG", NORMAL PROCEDURES FOR LOADING AND DUMPING FILES SHOULD BE FOLLOWED. IN THIS INSTANCE, PRIOR TO RUNNING TRNLIB THE ENTIRE 1400/B3500 SYSTEM TAPE SHOULD BE LOADED; E.G. AT REFERENCE 6-3-A KEY IN SPO

"CC LOAD FROM AUTSYS /."

WHEN THE NEW AUTIMG FILE HAS BEEN CREATED A NEW

AUTSYS SYSTEM TAPE MAY BE CREATED THROUGH USE OF  
THE MCP DUMP OPTION. THE PROGRAMS AND FILES ON  
THE SYSTEM TAPE (AUTSYS) ARE:

TR1400	1400 TO B3500 TRANSLATOR
TRNLIB	UTILITY PROGRAM
AUTIMG	SUBROUTINE FILE

## APPENDIX

A-1. TRANSLATOR EXECUTE DECK:

```
? EXECUTE TR1400
? DATAB AUTO
$TAPEIN LISTINPUT
$TAPEIN BYPASS
$TAPEIN LISTINPUT 1400 B3500 DISK "IDENT"
      F*PRN1DFN00201      00333      PRTLIN
      F*CPU1DFN00101      00180      PUNCH
      F*CRD1DFN00001      00080      CARDS
```

DFE

(IF CARD INPUT HAD BEEN SPECIFIED A SOURCE  
PROGRAM DECK WOULD BE INSERTED HERE).

? END

A-2. "AUTIMG" FILE:

(NOTE: LAST EIGHT COLUMNS ARE NOT LISTED HERE).

```

0001 ***** THE ABOVE INSTRUCTION HAS BEEN DELETED
0002 ***** UNIDENTIFIED OP-CODE OR NON-TRANSLATED BRANCH OPTION      *
0003 ***** NON-TRANSLATED -/O COMMAND                                *****
0004 ***** STACKER SELECT NOT TRANSLATED NO EFFECT ON PROGRAM      *
0005 ***** A-ADDR AFFECTS WORD MARK IN CODE - POSSIBLE LOGIC ERROR -
0006 ***** A-ADDR IS DATA - OK UNLESS DYNAMIC FIELD SIZE ADJUSTMENT-
0007 ***** B-ADDR AFFECTS WORD MARK IN CODE - POSSIBLE LOGIC ERROR -
0008 ***** B-ADDR IS DATA - OK UNLESS DYNAMIC FIELD SIZE ADJUSTMENT-
0009 ***** NON-TRANSLATED - INTENT NOT DETERMINED *****
0010 ***** SBR TO BRANCH ZERO - ASSUMED SUB-ROUTINE LINKAGE - - **
0011 ***** INSTRUCTION MODIFICATION -- RESULTS PROBABLY INCORRECT ***
0012 ***** CODE CHARACTER USED AS DATA - PROBABLY INCORRECT --      -
0013 ***** A-ADDR IS M/L NOT DEFINED- INSERT PROPER A-ADDRESS --      *
0014 ***** B-ADDR IS M/L NOT DEFINED - INSERT PROPER B-ADDRESS --      *
0015 ***** MISSING A-ADDR OR NON TRANSLATED A CHAIN-INSERT A LABEL  *
0016 ***** MISSING B-ADDR OR NON TRANSLATED B CHAIN-INSERT B LABEL  *
0017 ***** ADDRESS CONSTANT IS MACHINE LANGUAGE - NOT DEFINED      *
0018 ***** EQUATE FOR AN X-CONTROL OPERAND - PROBALLY NOT NEEDED --  *
0019 ***** EQUATE OF A NON DEFINED MACHINE LANGUAGE ADDRESS          *
0020 ***** LCA USED IN CONJUNCTION WITH MCE -- NOT NEEDED --        *
0021 ***** LCA NOT FOUND FOR THIS MCE - INSERT PROPER B ADDRESS --  *
0022 ***** NON TRANSLATED BRANCH IF INDICATOR PROBALLY ERROR BRANCH *
0023 ***** UN-LABELED DA FIELD OR SUB-FIELD - NOT REQUIRED            *
0024 ***** NON TRANSLATED WM-BRANCH EITHER INDEXED OR ABSOLUTE ZERO *
0025 ***** WARNING - DSA ADDRESS INCREMENT MAY BE INCORRECT        *
0026 ***** MOD 16000 COMPLIMENT ADDRESS ASKED FOR - CHECK REFERENCE *
0027 ***** INCREMENTED BRANCH - LOCATION NOT FOUND - INCORRECT ADDR *
0028 ***** BRANCH TO DATA AREA - CODE PROBALLY INCORRECT ----      *
0029 ***** INDEXED BRANCH - INDEX REGISTER VALUE PROBALLY INCORRECT-*
0030 ***** BRANCH TO ZERO -- ASSUMED SUBROUTINE EXIT                *
0031 ***** BRANCH TO SBR - ASSUMED SUBROUTINE ENTRY -              *
0032 ***** INSTRUCTION MODIFICATION -- RESULTS PROBALLY OK          *

```

```

0033 ***** ORG STATEMENTS NOT TRANSLATED - POSSIBLE LOCATION ERROR *
0034 ***** SEGMENTATION OF CODE IS PROVIDED HERE, HOWEVER, THE POINT*
0035 ***** WHERE OVERLAYING IS ACCOMPLISHED MUST BE DETECTED AND THE*
0036 ***** PROPER OVERLAY CODE INSERTED DEPENDING ON THE ASSEMBLER *
0037 ***** TO BE USED. *****
0038 ***** SKIP TO CHANNEL 12 MUST BE CHANGED TO ANOTHER CHANNEL *
0039 ***** LITERAL TRUNCATION- LIT SIZE TOO LARGE FOR PARAMETER *
0040 ***** ADD OR SUB OF 1400 ADDRESS CONSTANT INVALID ON B3500 *
0041 ***** THIS NOP (MORE THAN ONE-ADDRESS) HAS NO B3500 EQUIVALENT *
0042
0043
0044
0045
0046 ***** NON-TRANSLATED I=O COMMAND ACTION DEPENDENT ON SYSTEM *
0047 ***** WRITE DISK CHECK IS HARDWARE DEPENDENT-LIKELY NOT NEEDED *
0048 ***** SCAN DISK HARDWARE DEPENDENT - PROVIDE SUBROUTINE *
0049 ***** SEEK DISK REQUIRES ARM MOVEMENT - NOT NEEDED *
0050 ***** "GET" AND "PUT" REQUIRE MANUAL CORRECTION, INSERT FILE NAM
0051 BRANCH ADDRESS, ERROR BRANCH ADDRESS AS REQD, IF IOC ENTRY SPECIFI
0052 AREA OTHER THAN THE RECORD AREA ADD THE APPROPRIATE "MOVE" INSTRUC
0053 ***** THE PRECEDING DISK COMMAND REQUIRES THE FOLLOWING:
0054 ***** 1. THE NAME OF THE APPROPRIATE FILE IN COLUMN 22;
0055 ***** 2. A 6-CHARACTER LABEL IN COLUMN 34 DESIGNATING A
0056 ***** LOCATION TO BRANCH TO AT END-OF-FILE ON DISK;
0057 ***** THE EOF LABEL IS OPTIONAL, IF IT IS OMITTED, AN
0058 ***** END-OF-FILE CONDITION WILL RESULT IN A
0059 ***** NON-RECOVERABLE SITUATION DESIGNATED ON THE SPO
0060 ***** BY THE MCP AS "EOF == NO LABEL",
0061 ***** FOR OUTPUT DISK FILES, END-OF-FILE MEANS THAT
0062 ***** ALL REQUESTED DISK AREAS HAVE BEEN FILLED,
0063 ***** DISK WORD-MARK FORMAT NOT COMPATIBLE WITH THE B3500 *
0064 ***** TYPE OF DISK I/O -- SECTOR MODE *
0065 ***** TYPE OF DISK I/O -- FULL TRACK RECORD *
0066 ***** TYPE OF DISK I/O -- SECTOR MODE WITH COUNT OVERLAP *
0067 ***** TYPE OF DISK I/O -- TRACK SECTORS WITH ADDRESSES *
0068 ***** TYPE OF DISK I/O -- TRACK RECORDS WITH ADDRESSES *
0069 ***** DISK FILE REORGANIZATION PROBABLY NECESSARY *

```

0070  
0071  
0072  
0073  
0074  
0075  
0076  
0077  
0078  
0079  
0080  
0081  
0082  
0083  
0084  
0085  
0086  
0087  
0088  
0089  
0090  
0091  
0092  
0093  
0094  
0095  
0096  
0097  
0098  
0099  
0100  
0101  
0102  
0103\*  
0104 X\*R MVW R\*CRD R\*CRD1  
0105 READ F\*CRD1 X\*ECRD  
J106 EXT

## CARD READ ROUTINE

CURRENT RECORD  
READ NEXT RECOR  
RETURN TO PROGR

0107	X*ECRD							
0108	MVN	1 11	NLX*SWA					SET SENSE SWITC
0109	EXT							RETURN TO PROGR
0110								
0111*								
0112*								
0113*								
0114*								
0115*								
0116*								
0117*								
0118*								
0119*								
0120*								
0121								
0122			ADD - SUBTRACT	SUBROUTINE				
0123								
0124								
0125	X*ACHNDEC	2 5BASE	183	BASE	233			FIND CHAINED A-
0126	DEC	2 5BASE	183	BASE	233			
0127	DEC	2 5BASE	203	BASE	293			FIND CHAINED B-
0128	DEC	2 5BASE	203	BASE	293			
0129								
0130	X*A	MVN 1 1BASE	173	+0	1			SET PROPER OP-C
0131		MVA 2 2BASE	93U	BASE	16 UA			STORE IND, FLD
0132		ADD 1 21	NLBASE	34	BASE	36		ADD 1 TO B-FIELD
0133		INFL	BASE	32	BASE	32		
0134		MVN	BASE	223	IAX*WRK1			STORE A-FIELD
0135		MVN 1 1C	NLX*WRK1		UN			SET PLUS SIGN
0136		ADD 2 6BASE	183	BASE	223	+A	12	FIND A-FIELD LS
0137		INC 2 5BASE	183	+A	13			
0138		DEC 1 52	NL+A		13			
0139	.A	SDE 16 1X*Z=B	BASE					IS A-FLD LSD SI
0140		GTR *	26					BRANCH IF NOT
0141		MVN 1 1D	NLX*WRK1		UN			SET MINUS SIGN
0142		INFL	BASE	34	BASE	36		
0143		MVN	BASE	283	IAX*WRK2			STORE B-FIELD





0181\*  
 0182\*  
 0183\*  
 0184\*  
 0185\*  
 0186\*  
 0187\*  
 0188\*  
 0189\*  
 0190\*  
 0191\*  
 0192\*  
 0193\*  
 0194\*  
 0195\*  
 0196\*  
 0197\*  
 0198\*  
 0199\*  
 0200\*  
 0201  
 0202  
 0203  
 0204  
 0205  
 0206  
 0207  
 0208  
 0209  
 0210  
 0211  
 0212\*  
 0213\*  
 0214\*  
 0215\*  
 0216\*  
 0217\*

HALT ROUTINE

X*H	SUB 2 512	NLBASE	13	X*HALT	8	MOVE HALT LOC T
	DISP13 X*HALT					DISPLAY HALT
	ACPT 2 X*ACPT					ACCEPT REPLY
	CPA 2 2DS	ALX*ACPT				IS IT DS
	NEQ +A					BRANCH IF NOT
	STOP					STOP RUN
.A	EXT					RETURN TO PROGR

0218*						
0219	X*HALTCNST	13UA		HALT AT		
0220	X*ACPTCNST	2UA				
0221	X*OFLSCNST	1UN				STORED OVERFLOW
0222	X*OFL CNST	3UN				OVERFLOW TOGGLE
0223	X*CH12CNST	3UN				CHANNEL 12 BOOL
0224	X*CTOGCNST	1UN				COMPARISON INDI
0225	X*Z=NOCNST	16UA		1234567890 #@:;>		NO-ZONE CHARACT
0226	X*Z=A CNST	16UA		/STUVWXYZ,%=]'^*#		A-ZONE CHARACT
0227	X*ZZ CNST	2UN		4E		
0228		CNST 2UN		C0		
0229		CNST 15UA		ABCDEFGHI&.[(<<		
0230	X*Z=ABEQIV	17X*ZZ	UA			AB-ZONE CHARACT
0231	X*Z=B CNST	16UA		JKLMNOPQR-x\$*)]S		B-ZONE CHARACT
0232*						
0233*						
0234*						
0235*						
0236*						
0237*						
0238*						
0239*						
0240						
0241						
0242				COMPARE SUBROUTINE		
0243						
0244						
0245*						
0246*						
0247*						
0248*						
0249*						
0250	X*CCHNDEC	2 5BASE	163	BASE	213	ADJUST A ADDRES
0251		DEC 2 5BASE	163	BASE	213	FULL CHAIN.
0252	X*CHCNDEC	2 5BASE	183	BASE	273	ADJUST B ADDRES
0253		DEC 2 5BASE	183	BASE	273	FULL OR HALF CH
0254						



```

0292*
0293*
0294*
0295*
0296*
0297  .L   MVN  1 11           NLX*CTOG           SET COMPARISON
0298      EXT                               EXIT WITHOUT BR
0299  .H   MVN  1 14           NLX*CTOG           SET COMPARISON
0300      EXT                               EXIT WITHOUT BR
0301*
0302*
0303*
0304*
0305*
0306      SUBROUTINE TO MOVE CHARACTERS TO RECORD MARK
0307
0308  X*M   MVN  1 40           NLBASE  +36           RESET GM FLAG A
0309      SDE  200#+           ALBASE  +163IA        SCAN FOR REC/M
0310      GTR   +A                BRANCH IF NOT F
0311      MVN  1 11           NLBASE  +36           SET REC/M BOOLE
0312  .A   INC  1 31           NLBASE  +37           ADD 1 TO INFL
0313      INFL  BASE  +38     BASE  +38           MOVE LENGTH AT
0314      MVA   BASE  +163IABASE +223IA        MOVE A-FLD TO B
0315      INC  1 32           NLBASE  +173          ADVANCE A-ADDRE
0316      INC  1 32           NLBASE  +233          ADVANCE B-ADDRE
0317      BOT  0110BASE  +36           WAS REC/M FOUND
0318      NEQ   X*M                REPEAT IF NOT
0319      EXT                               RETURN TO MAIN
0320
0321*
0322*
0323*
0324*
0325*
0326*
0327*
0328*

```

```

0329*
0330*
0331
0332
0333             MOVE SUBROUTINE (MOVES OVER 100)
0334
0335
0336*
0337*
0338*
0339*
0340  X*N  MVN  2 299             NLBASE  32             SET LENGTH TO 9
0341  .A   MVN  5 2BASE  173 BASE  32             STORE SIZE IF N
0342             INFL  BASE  32  BASE  32             MOVE SIZE AT AD
0343             MVA   BASE  223IABASE  283IA         MOVE UP TO 99 C
0344             INC  3 5198             NLBASE  233         ADVANCE A=ADDRE
0345             INC  3 5198             NLBASE  293         ADVANCE B=ADDRE
0346             DEC  2 5BASE  32  BASE  173         SUB NO MOVED FR
0347             GTR   -A                 REPEAT IF RESUL
0348             EXT                               EXIT MOVE ROUTI
0349*
0350*
0351*
0352*
0353*
0354*
0355*
0356*
0357*
0358*
0359*
0360*
0361             MOVE CHARACTERS AND SUPPRESS ZEROS SUBROUTINE
0362
0363  X*Z  MVN  2 2BASE  +163UNBASE  +36 UN         SET LENGTH OF A
0364             INFL  BASE  +36
0365  .A   MVA  1             ALBASE  +263IA         SET B=FIELD TO

```

0366		INFL			BASE	+36				
0367		SDU	2	0,	ALBASE	+203IA				SCAN A COUNTING
0368		GTR		+X						NUMBER OF LEADI
0369		INC	2	5BASE	+38	UNBASE	+213UN			IF ALL ZEROS -
0370		INC	2	5BASE	+38	UNBASE	+213UN			ADJUST BOTH THE
0371		INC	2	5BASE	+38	UNBASE	+273UN			B-ADDRESSES BY
0372		INC	2	5BASE	+38	UNBASE	+273UN			NUMBER OF LEADI
0373		DEC	2	2BASE	+38	UNBASE	+36 UN			ALSO ADJUST SIZ
0374		INFL		BASE	+36	BASE	+36			BOTH OPERANDS.
0375		MVA		BASE	+203IABASE	+263IA				MOVE NON-ZERO P
0376		INFL			BASE	+36				OF A TO B RIGHT
0377		SDE	1	.	ALBASE	+203IA				SCAN FOR . TO R
0378		GTR		+Y						ZERO SUPPERSION
0379		INC	1	21	NLBASE	+38 UN				IF NOT, EXIT.
0380		CPN	2	2BASE	+36	UNBASE	+38 UN			DETERMINE IF PE
0381		EQL		+X						LAST CHARACTER,
0382		INC	2	5BASE	+38	UNBASE	+213UN			IT IS, EXIT.
0383		INC	2	5BASE	+38	UNBASE	+213UN			ADJUST ADDRESSE
0384		INC	2	5BASE	+38	UNBASE	+273UN			POINT PAST THE
0385		INC	2	5BASE	+38	UNBASE	+273UN			AND SET SIZE TO
0386		DEC	2	2BASE	+38	UNBASE	+36 UN			NUMBER OF REMAI
0387		BUN		-A						CHARACTERS,
0388	.Y	CPN	1	21	NLBASE	+36 UN				IF ADDRESSES AR
0389		EQL		+Z						ALREADY POINTIN
0390		DEC	1	21	NLBASE	+36				LAST CHARACTER
0391		INC	2	5BASE	+36	UNBASE	+273UN			FIELD, ADJUST T
0392		INC	2	5BASE	+36	UNBASE	+273UN			DO SO AND REMOV
0393	.Z	MVN	1	1BASE	+263IABASE	+263IA				SIGN
0394	.X	EXT								JUSTIFIED AND E
0395*										
0396*										
0397*										
0398	X*REPLCNST	4UA			0 *					REPLACEABLE CHA
0399	X*FLOT CNST	1UA								FLOATING \$ INDI
0400	X*FILLCNST	1UA								
0401										
0402										MOVE CHARACTERS AND EDIT SUBROUTINE

0403						
0404	X+E	MVN	1212BASE	+203UNBASE	+323UN	CREATE TWO ADDR
0405		INC	2 5BASE	+163UNBASE	+333UN	IN THE STACK,
0406		INC	2 5BASE	+163UNBASE	+333UN	POINTING TO THE
0407		DEC	1 52	NLBASE	+333UN	THE A FIELD, TH
0408		INC	2 5BASE	+183UNBASE	+393UN	POINTING TO THE
0409		INC	2 5BASE	+183UNBASE	+393UN	THE B FIELD.
0410		DEC	1 52	NLBASE	+393UN	
0411		CPA	1 1-	ALBASE	+383IA	DETERMINE IF ED
0412		NEQ	+A			CONTAINS TRAILI
0413		MVN	1 21	NLBASE	+36 UN	OR CR SYMBOLS.
0414		BUN	+B			IF A-FIELD IS N
0415	.A	DEC	1 52	NLBASE	+393UN	NEGATIVE, BLANK
0416		CPA	2 2CR	ALBASE	+383IA	SYMBOLS.
0417		NEQ	+D			
0418		MVN	1 22	NLBASE	+36 UN	
0419	.B	SDE	16 1X*Z=B	BASE	+323IA	
0420		LEQ	+C			
0421		INFL		BASE	+36	
0422		MVA	1	ALBASE	+383IA	
0423	.C	DEC	1 52	NLBASE	+393UN	
0424		BUN	+H			
0425	.D	INC	1 52	NLBASE	+393UN	
0426		CPA	1 1-	ALBASE	+263IA	DETERMINE IF ED
0427		NEQ	+E			CONTAINS LEADIN
0428		MVN	1 21	NLBASE	+36 UN	OR CR SYMBOLS.
0429		BUN	+F			IF A-FIELD IS N
0430	.E	CPA	2 2CR	ALBASE	+263IA	NEGATIVE, BLANK
0431		NEQ	+H			SYMBOLS.
0432		MVN	1 22	NLBASE	+36 UN	
0433	.F	SDE	16 1X*Z=B	BASE	+323IA	
0434		LEQ	+G			
0435		INFL		BASE	+36	
0436		MVA	1	ALBASE	+263IA	
0437	.G	INC	2 5BASE	+36 UNBASE	+273UN	ADJUST ADDRESS
0438		INC	2 5BASE	+36 UNBASE	+273UN	SYMBOL LOCATION
0439	.H	CPA	1 1&	ALBASE	+383IA	RIGHT=MOST CHAR



0440		NEQ	+I			= &
0441		MVA	1 1	ALBASE	+383IA	IF SO, BLANK IT
0442		DEC	1 52	NLBASE	+393UN	ADJUST ENDING A
0443		BUN	-H			
0444	.I	CPA	1 10	ALBASE	+383IA	DETERMINE IF RI
0445		NEQ	+J			CHARACTER OF ED
0446		MVN	2 2BASE	+163UNBASE	+36 UN	IS ZERO. IF SO
0447		INFL		BASE	+36	A-FIELD = ZERO.
0448		CPN	1 0	ALBASE	+203IA	IF A-FIELD = ZE
0449		NEQ	+J			SPACES TO B-FIE
0450		MVN	2 2BASE	+183UNBASE	+36 UN	GO TO EXIT.
0451		INFL		BASE	+36	
0452		MVA	1	ALBASE	+263IA	
0453		BUN	+X			
0454	.J	MVN	6 6200000	NLBASE	+443UN	SET UP ZERO ADD
0455	.K	CPA	1 10	ALBASE	+383IA	TEST- MASK CHAR
0456		NEQ	+L			= ZERO.
0457		CPN	1 50	NLBASE	+453UN	IS THIS RIGHT=M
0458		NEQ	+L			ZERO. IF SO, S
0459		MVN	6 6BASE	+383UNBASE	+443UN	END SUPR ADDRES
0460		DEC	1 52	NLBASE	+393UN	LOOK AT PREVIQU
0461		CPA	1 1*	ALBASE	+383IA	IS IT = *
0462		NEQ	* +26			IF SO,
0463		MVA	1 1*	ALX*FILL		SET FILL TO *
0464		CPA	1 1\$	ALBASE	+383IA	IS PREV CHAR \$.
0465		NEQ	* +26			IF SO,
0466		MVA	1 11	ALX*FLOT		SET \$ FLOAT IND
0467		INC	1 52	NLBASE	+393UN	ADDR TO CURRENT
0468	.L	SDE	3 10 *	ALBASE	+383IA	MASK CHAR = 0,*
0469		LEQ	+N			
0470		CPA	1 1&	ALBASE	+383IA	
0471		NEQ	+M			
0472		MVA	1 1	ALBASE	+383IA	
0473		BUN	+M			
0474	.N	MVN	1 1BASE	+323IABASE	+383IA	
0475		DEC	1 52	NLBASE	+333UN	DECREASE A-ADDR
0476	.M	DEC	1 52	NLBASE	+393UN	DECREASE B-ADDR

0477	CPN	5	5BASE	+393UNBASE	+273UN	IS B-FIELD EXHA
0478	LSS		+P			
0479	CPN	5	5BASE	+333UNBASE	+213UN	IS A-FIELD EXHA
0480	GEQ		-K			
0481	.P	SDE	4	1X*REPL	BASE +263IA	B-CHAR = 0 , *
0482		GTR		+Q		
0483		MVA	1	1X*FILL	BASE +263IA	REPLACE WITH FI
0484		INC	1	52	NLBASE +273UN	INCREMENT B-ADD
0485		CPA	1	1.	ALBASE +263IA	MASK CHAR = DEC
0486		EQL		+Q		IF SO, END.
0487		CPN	5	5BASE	+273UNBASE +453UN	END OF ZERO SUP
0488		NEQ		-P		
0489	.Q	CPA	1	11	ALX*FLOT	FLOATING DOLLAR
0490		NEQ		+X		
0491		DEC	1	52	NLBASE +273UN	REDUCE B-ADDRES
0492		MVA	1	15	ALBASE +263IA	INSERT DOLLAR S
0493	.X	MVA	2	20	ALX*FLOT	RESET INDICATOR
0494		EXT				
0495						
0496*						
0497*						
0498*						
0499*						
0500*						
0501*						
0502*						
0503*						
0504*						
0505*						
0506						
0507						PRINT ROUTINE FOR OTHER THAN SINGLE-SPACING
0508						
0509	X*PRT	EQIV	4X*P	26		PAPER MOTIION VA
0510*						****FOR BASIC PROGRAMS, CHANGE ABOVE INCREMENT TO 24
0511	X*EOP	MVN	1	11	NLX*CH12 1	SET CHAN=12 BOO
0512	X*P	WRIT	1	F*PRN1	X*EOP	WRITE UN LINE P
0513		MVN	4	40100	NLX*PRT	RESTORE TO SING

RETURN TO PROGR

0514 EXT

0515\*

0516\*

0517\*

0518\*

0519\*

0520\*

0521\*

0522\*

0523\*

0524\*

0525\*

0526\*

0527\*

0528\*

0529\*

0530\*

0531

0532

PRINT ROUTINE TO HANDLE TRIPLE SPACING

0533

0534 X\*PM EQIV 4X\*PRNT 26 PAPER MOTION VA

0535\* \*\*\*\*FOR BASIC PROGRAMS, CHANGE ABOVE INCREMENT TO 24

0536 X\*PRT CNST 4UN 0100 P/M VARIABLE ST

0537 X\*P MVN 3 4X\*PRT 1 X\*PM SET VARIABLE IN

0538 X\*PRNTWRIT 1 F\*PRN1 X\*EOP WRITE ON LINE P

0539 BOT 0110X\*PRT TEST FOR 3RD SP

0540 NEQ +P BRANCH IF NOT N

0541 X\*SKIPPOSN 1 F\*PRN1 X\*EOS SINGLE SPACE PR

0542 .P MVN 4 40100 NLX\*PRT RESTORE TO SING

0543 EXT RETURN TO PROGR

0544

0545 X\*EOP MVN 1 11 NLX\*CH12 1 SET CHAN=12 800

0546 BUN X\*PRNT GO BACK TO PRIN

0547

0548 X\*EOS MVN 1 11 NLX\*CH12 1 SET CHAN=12 800

0549 BUN X\*SKIP GO BACK TO SPAC

0550\*

0551\*  
 0552\*  
 0553\*  
 0554\*  
 0555\*  
 0556\*  
 0557\*  
 0558\*  
 0559\*  
 0560\*

0561

0562 MULTIPLY SUBROUTINE

0563

0564	X*U	MVA	2	2BASE	83UABASE	16	UA			STORE A AND B S
0565		ADD	1	21	NLBASE	34	BASE	36		B-FIELD SIZE +
0566		INFL		BASE	32	BASE	32			
0567		MVN		BASE	243IAX*WRK1					STORE A-FIELD
0568		MVN	1	1C	NLX*WRK1		UN			SET A-FIELD PLU
0569		ADD	2	6BASE	163	BASE	243	+A	12	FIND
0570		INC	2	5BASE	163	+A	13			A-FIELD
0571		DEC	1	52	NL+A	13				LSD
0572	.A	SDE	16	1X*Z-B	BASE					IS A-FLD LSD SI
0573		GTR	*		26					BRANCH IF NOT
0574		MVN	1	1D	NLX*WRK1		UN			SET A-FLD MINUS
0575		INFL		BASE	34	BASE	36			
0576		MVN		BASE	303IAX*WRK2					STORE B-FIELD
0577		MVN	1	1C	NLX*WRK2		UN			SET B-FIELD PLU
0578		ADD	2	6BASE	183	BASE	303	+B	12	FIND
0579		INC	2	5BASE	183	+B	13			B-FIELD
0580		DEC	1	52	NL+B	13				LSD
0581	.B	SDE	16	1X*Z-B	BASE					IS B-FLD LSD SI
0582		GTR	*		26					BRANCH IF NOT
0583		MVN	1	1D	NLX*WRK2		UN			SET B-FLD MINUS
0584		INFL		BASE	32	BASE	36			
0585		MPY		X*WRK1	X*WRK2		X*WRK3			PERFORM MULTIPL
0586		ADD	4	5BASE	203	BASE	313	+D	13	FIND
0587		INC	4	5BASE	203	+D	13			RESULT-FIELD

0588	DEC	1	52	NL+D	13	LSD		
0589	MVN	1	40	NLBASE	32	SET RESULT SIZE		
0590	MVN	4	2BASE	203	BASE	32	STORE RESULT SZ	
0591	INFL		BASE	32	BASE	32		
0592	MVN		X*WRK3	1	UNBASE	303IA	MOVE 1ST 100 OR	
0593	SUB	3	4100	NLBASE	203	BASE	32	FIND EXCESS OVE
0594	LEQ		+D				BRANCH IF NOT >	
0595	INC	1	32	NLBASE	313		INC RES=FLD BY	
0596	INFL		BASE	34	BASE	34		
0597	MVN		X*WRK3101	UNBASE	303IA		MOVE 2ND PART 0	
0598	.D	MVN	1	1X*WRK3	UNBASE		STORE RESULT SI	
0599		EXT					RETURN TO PROGR	
0600*								
0601*								
0602*								
0603*								
0604*								
0605*								
0606*								
0607*								
0608*								
0609*								
0610*								
0611*								
0612*								
0613*								
0614*								
0615*								
0616*								
0617*								
0618*								
0619*								
0620*								
0621*								
0622*								
0623*								
0624*								

```

0625*
0626*
0627*
0628*
0629*
0630*
0631
0632          DIVIDE          SUBROUTINE
0633
0634 X*INVDCNST 23UA          DS-ED == INVALID DIVIDE
0635 X*D   SDE 3397X*Z-AR     BASE 263IA          FIND B-FIELD LS
0636          LEQ      +A          BRANCH IF FOUND
0637 .D   DISP23 X*INVD      DISPLAY INVALID
0638          STOP          ABORT--END OF J
0639
0640 .A   INC 1 21          NLBASE 38          SET PROPER INDI
0641          MVR 2 2BASE 38  BASE 34          FIELD LENGTHS
0642          INC 2 2BASE 183 BASE 34          DIVISOR, DIVI
0643          OFL      -D          TOTAL B-FIELD
0644          MVN 2 2BASE 163 BASE 32
0645          DEC 2 5BASE 183 BASE 273          FIND BEGINNING
0646          DEC 2 5BASE 183 BASE 273          OF TOTAL B-FI
0647          INFL  BASE 32  BASE 32
0648          MVN  BASE 203IAX*WRK1          STORE A-FIELD
0649          MVN 1 1C          NLX*WRK1  UN          SET A-FIELD SIG
0650          ADD 2 6BASE 32  BASE 203 +B 12  FIND
0651          INC 2 5BASE 32  +B 13          A-FIELD
0652          DEC 1 52          NL+B 13          LSD ADDRESS
0653 .B   SDE 16 1X*Z=B          BASE          IS A-FIELD SIGN
0654          GTR      * 26          BRANCH IF NOT
0655          MVN 1 1D          NLX*WRK1  UN          SET A-FIELD SIG
0656          INFL  BASE 34  BASE 34
0657          MVN  BASE 263IAX*WRK2          STORE B-FIELD
0658          MVN 1 1C          NLX*WRK2  UN          SET B-FIELD SIG
0659          ADD 2 6BASE 34  BASE 263 +C 12  FIND
0660          INC 2 5BASE 34  +C 13          B-FIELD
0661          DEC 1 52          NL+C 13          LSD ADDRESS

```



0699\*  
 0700\*  
 0701  
 0702  
 0703  
 0704  
 0705  
 0706  
 0707  
 0708  
 0709  
 0710  
 0711  
 0712  
 0713  
 0714  
 0715  
 0716  
 0717  
 0718  
 0719  
 0720  
 0721  
 0722  
 0723  
 0724  
 0725  
 0726  
 0727  
 0728  
 0729\*  
 0730\*  
 0731\*  
 0732\*  
 0733\*  
 0734\*  
 0735\*

CONVERT 3-CHARACTER ADDRESS TO 5-DIGIT ADDRESS

X*CEPTCNST	6UN		616D4E			1401 EXCEPTION
X*RPL CNST	6UN		E1E0C0			3500 REPLACEMEN
X*ADR CNST	5UN					RESULTING 5-DIG
X*NOT1CNST	1UN					ZONE W/A 1
X*NOT2CNST	1UN					ZONE W/A 2
X*NOT3CNST	1UN					ZONE W/A 3
X*CHARCNST	3UA					1401 3-CHAR ADD
X*CNV MVA	3	3BASE	163IAX*CHAR			STORE 3-CHAR AD
MVN	3	5X*CHAR	X*ADR			STORE PORTION <
.A SDE	1	3X*CHAR	X*CEPT	UA		TEST FOR EXCEPT
GTR		+B				BRANCH IF NONE
MVN	6	5X*RPL	LL+A	7		SET ADR OF REPL
INC	1	5BASE	39 +A	7		INCREMENT INTO
INC	1	5BASE	39 +A	7		REPLACEMENT T
.A MVA	1	1X*RPL	UAX*CHAR			REPLACE EXCEPTI
.B NOT	1	1F27	NLX*CHAR	UNX*NOT3		CONVERT ZONE MO
MVN	2	2X*NOT2	X*NOT1			STORE MODIFIED
MVA	1	1X*CHAR	2 X*CHAR			SET UP LSD CHAR
MVL	2	-B	7 +C	+C		SET SWITCH TO R
.C NOP		-A				REPEAT FOR LSD
INC	1	2X*NOT1	X*ADR			ADD MODIFIER <
MPY	1	14	NLX*NOT2	-B	9	FIND 4000 MOD
INC	2	2-B	9 X*ADR			ADD 4000 MOD
EXT						RETURN EXIT



0736\*  
 0737\*  
 0738\*  
 0739\*  
 0740\*  
 0741  
 0742  
 0743  
 0744  
 0745  
 0746  
 0747  
 0748  
 0749  
 0750  
 0751  
 0752  
 0753  
 0754  
 0755\*  
 0756\*  
 0757\*  
 0758\*  
 0759\*  
 0760\*  
 0761\*  
 0762\*  
 0763\*  
 0764\*  
 0765\*  
 0766\*  
 0767\*  
 0768\*  
 0769\*  
 0770\*  
 0771  
 0772

## DETERMINE CORRECT B3500 INDEX REGISTER VALUE

X*J	MVN	2	2BASE	183	BASE	32	STORE IND A-FIE
	INFL		BASE	32			
	MVN		5BASE	203IAX*ADR			STORE A-FIELD
	BZT	0130	BASE	193			IS A-FIELD 3 CH
	EQL		+B				BRANCH IF NOT
X*I	MVA	3	3BASE	203IAX*CHAR			STORE 3-CHAR AD
	NTR		X*CNV	18			CONVERT TO 5-DI
.B	INC	5	5X*ADR		X*ADR		DOUBLE IT
	MVN	2	2BASE	163	+C		OP IS INC, DEC,
.C	INC	5	7X*ADR		BASE	263IA	PUT VALUE INTO
	EXT						RETURN EXIT

POSITION PRINTER IF NO TRIPLE SPACING

0773							
0774	X*POSNMVN	3	4BASE	173	+P	27	SET PRINTER VAR
0775	.P	POSN	1	F*PRN1		X*EOS1	SKIP OR SPACE P
0776		EXT					RETURN EXIT
0777							
0778	X*EOS1MVN	1	11		NLX*CH12	1	SET CHANNEL 12
0779		BUN		-P			GO BACK TO POSN
0780*							
0781*							
0782*							
0783*							
0784*							
0785*							
0786*							
0787*							
0788*							
0789*							
0790*							
0791							
0792							POSITION PRINTER IF TRIPLE SPACING IS REQUIRED
0793							
0794	X*POSNMVN	3	4BASE	173	+P	27	SET PRINTER VAR
0795	.P	POSN	1	F*PRN1		X*EOS1	SINGLE OR DOUBL
0796		BOT	0110BASE	163			IS 3RD SPACE NE
0797		NEQ		+T			BRANCH IF NOT
0798	.Q	POSN	1	F*PRN1		X*EOS2	SINGLE SPACE PR
0799	.T	EXT					RETURN EXIT
0800							
0801	X*EOS1MVN	1	11		NLX*CH12	1	SET CHANNEL 12
0802		BUN		-P			GO BACK TO 1ST
0803							
0804	X*EOS2MVN	1	11		NLX*CH12	1	SET CHANNEL 12
0805		BUN		-Q			BO BACK TO 2ND
0806*							
0807*							
0808*							
0809*							

0810*			
0811*			
0812*			
0813*			
0814*			
0815*			
0816*			
0817*			
0818*			
0819*			
0820*			
0821	STORE1CNST	2UN	SUM OF A & B DE
0822	STORE2CNST	2UN	C-FIELD DECIMAL
0823	STORE3CNST	7UN	C-FIELD LENGTH
0824	STORE4CNST	2SN	PRODUCT SCALING
0825	SAVX*1CNST	16UN	
0826*			
0827*			
0828*			
0829*			
0830*			
0831*			
0832*			
0833*			
0834*			
0835*			
0836*			
0837*			
0838*			
0839*			
0840*			
0841	MACRADDR	\$A, \$B, \$C, \$D, \$E, \$F, \$G, \$H,	
0842		\$I, \$J, \$K, \$L, \$M	
0843	MEQLTYPEOMIT	\$K	MSTART
0844	MEQLTYPENULL	\$K	MSTART
0845	MBUN	MERMES	
0847	MEQLTYPENULL	\$M	MNOP1

0848	MEQLTYPEOMIT		\$M		MNOP1	
0849	MVN 2 203		NLAS(IX)		UN	
0850	MNOP1 MNOP					
0851	MVN 1616BASE	8	SAVX*1			SAVE IX1 & IX2
0852	MVN 1 0		NLX*WRK1			CLEAR WORK1 TO
0853	MVN 1 0		NLX*WRK2			CLEAR WORK2 TO
0854	MVN 2\$H		BASE	16		
0855	MVN 2 2BASE	16	BASE	18		
0856	MVN 2\$B		BASE		UN	STORE A OPND SI
0857	INFL		BASE		UN	INFL
0858	MVA \$A		X*WRK3			STORE A-OPND IN
0859	MVN 2\$E		BASE	2	UN	STORE B-OPND SI
0860	INFL		BASE	2	UN	INFL
0861	MVA \$D		X*WRK2			STORE B-OPND IN
0862	DEC 2\$C		BASE		UN	GET A INTEGER S
0863	DEC 2\$F		BASE	2	UN	GET B INTEGER S
0864	INC 2\$I		BASE		UN	INC C-DECIM TO
0865	INC 2\$I		BASE	2	UN	INC C-DECIM TO
0866	MEQLTYPELABEL		\$J		MLAB1	
0867	INC 1 21		NLBASE		UN	ADD 1 DIGIT PRE
0868	INC 1 21		NLBASE	2	UN	ADD 1 DIGIT PRE
0869	MLAB1 MNOP					
0870	CPN 2 2BASE		BASE	2		CPN A-SIZ TO B-
0871	LSS *	34				BUN ON A < B
0872	MVN 2 2BASE		UNBASE	6	UN	MOVE A-SIZ TO B
0873	BUN *	26				ELSE
0874	MVN 2 2BASE	2	UNBASE	6	UN	MOVE B-SIZ TO B
0875	INC 1 21		NLBASE	6	UN	ADD 1 DIGIT,OFL
0876	INFL BASE		UNBASE	6	UN	
0877	MVN X*WRK3		X*WRK1			RIGHT JUSTIFY A
0878	INFL BASE	2	UNBASE	6	UN	
0879	MVN X*WRK2		X*WRK3			RIGHT JUSTIFY B
0880	MEQLTYPENULL		\$L		MLAB2	
0881	MEQLTYPEOMIT		\$L		MLAB2	
0882	MVN 1 1C		NLX*WRK1		UN	SET PLUS SIGN
0883	MVN \$B		IX1			INDEX TO A FIEL
0884	DEC 1 1		NLIX1			

0885	INC	IX1	IX1			
0886	SDE	16 1X*Z=B	\$A	1		SCAN FOR MINUS
0887	GTR	*	26			BUN IF PLUS
0888	MVN	1 1D	NLX*WRK1	UN		SET MINUS SIGN
0889	MVN	1 1C	NLX*WRK3	UN		REPEAT SIGN HAN
0890	MVN	\$E	IX1			
0891	DEC	1 1	NLIX1			
0892	INC	IX1	IX1			
0893	SDE	16 1X*Z=B	\$D			
0894	GTR	*	26			
0895	MVN	1 1D	NLX*WRK3	UN		
0896	MLAB2	MNOP				
0897	INFL	BASE	6 UNBASE	6 UN		
0898	AS(IX)INC	X*WRK3	X*WRK1			INC OR DEC DEPE
0899	MEQLTYPELABEL		\$J		MLAB3	BUN IF NOT ROUN
0900	INFL		BASE	6 UN		
0901	INC	1 5	NLX*WRK1	1 UN		ROUND RESULT
0902	MLAB3	MNOP				
0903	DEC	1 21	NLBASE	6 UN		DEC 1 DIGIT:OFL
0904	MVN	2 7BASE	6 UNIX1			STORE RESULT SI
0905	DEC	2 7BASE	16 IX1			DEC IX1 BY C=F
0906	GEQ	AD(IX)				BUN ON RSLT ≥ C
0907	MVN	210BASE	6 BASE	8 UN		ZERO X1,STORE R
0908	INFL	BASE	16 BASE	18		INFL
0909	AD(IX)MVN	X*WRK1	11UN\$G			MOVE RESULT TO
0910	MEQLTYPEOMIT		\$L		MLAB4	BUN IF SIGN HAN
0911	MEQLTYPENULL		\$L		MLAB4	NOT REQUIRED
0912	SUB	1 21	NLBASE	18 UNBASE	14	INDEX TO C=FLD
0913	INC	IX1	IX1			ADJUST FOR UA D
0914	MVN	1 1X*WRK1	UN\$G	1UN		TRANSFER SIGN
0915	MLAB4	MNOP				
0916	MVN	1616SAVX*1	BASE	8 UN		RESET IX1 & IX2
0917	MEXT					
0918	MERMESMERR	ILLEGAL EDIT PARAMETER:ADD-SUB	MACRO			
0919	MEND					
0920*						
0921*						

0922\*  
0923\*  
0924\*  
0925\*  
0926\*  
0927\*  
0928\*  
0929\*  
0930\*  
0931\*  
0932\*  
0933\*  
0934\*  
0935       MACRSUBR\$A, \$B, \$C, \$D, \$E, \$F, \$G, \$H, \$I,  
0936               \$J, \$K, \$L  
0937       ADDR     \$A,\$B,\$C,\$D,\$E,\$F,\$G,\$H,\$I,  
0938               \$J, \$K, \$L, 1  
0939       MEXT  
0940       MEND  
0941\*  
0942\*  
0943\*  
0944\*  
0945\*  
0946\*  
0947\*  
0948\*  
0949\*  
0950\*  
0951\*  
0952\*  
0953\*  
0954\*  
0955\*  
0956       MACRMLTP \$A, \$B, \$C, \$D, \$E, \$F, \$G, \$H,  
0957               \$I, \$J, \$K, \$L  
0958       MEQLTYPEOMIT       \$K               MSTART

0959		MEQLTYPENULL		\$K		MSTART	
0960		MBUN		MERMES			
0961	MSTART	MNOP					
0962		MVN	1616BASE	8 UNSAVX*1			SAVE IX1 & IX2
0963		MVN	1 0	NLX*WRK3			CLEAR WORK3 TO
0964		MVN	2\$B	BASE			
0965		MVN	2\$E	BASE	2		STORE PARAMETER
0966		MVN	2\$H	BASE	16		SET UP INFLS
0967		MVN	2 2BASE	16 BASE	18		STORE C SIZE TW
0968		MEQLTYPENULL		\$C		MLAB1	BUN IF NO SCALI
0969		MEQLTYPEOMIT		\$C		MLAB1	REQUIRED
0970		MVN	2\$C	STORE1			STORE A OPND SC
0971		INC	2\$F	STORE1			INC BY B OPND S
0972		MVN	2\$I	STORE2			STORE C OPND SC
0973	MLAB1	MNOP					
0974		INFL	BASE	BASE			
0975		MVN	\$A	X*WRK1			STORE A OPND
0976		INFL	BASE	2 BASE	2		
0977		MVN	\$D	X*WRK2			STORE B OPND
0978		MEQLTYPEOMIT		\$L		MLAB2	BUN IF NO SIGN
0979		MEQLTYPENULL		\$L		MLAB2	HANDLING REQUIR
0980		MVN	1 1C	NLX*WRK1		UN	SET PLUS SIGN
0981		MVN	2 BASE	IX1			INDEX TO A FIEL
0982		DEC	1 1	NLIX1			
0983		INC	IX1	IX1			
0984		SDE	16 1X*Z=B	\$A		1	IS A=FLD LSD SI
0985		GTR	*	26			BRANCH IF NOT M
0986		MVN	1 1D	NLX*WRK1		UN	SET MINUS SIGN
0987		MVN	1 1C	NLX*WRK2		UN	REPEAT SIGN HAN
0988		MVN	2 BASE	2 IX1			
0989		DEC	1 1	NLIX1			
0990		INC	IX1	IX1			
0991		SDE	16 1X*Z=B	\$D		1	
0992		GTR	*	26			
0993		MVN	1 1D	NLX*WRK2		UN	
0994	MLAB2	MNOP					
0995		INFL	BASE	BASE	2		A & B FIELD LEN

0996	MPY	X*WRK1	X*WRK2	X*WRK3	MULTIPLY
0997	MVN	2 7BASE	STORE3		A LENGTH TO STO
0998	INC	2 7BASE	2 STORE3		ADD B LNTH TO
0999	MEQLTYPENULL		\$C	MLAB3	BUN IF SCALING
1000	MEQLTYPEOMIT		\$C	MLAB3	REQUIRED
1001	SUB	2 2STORE1	STORE2	STORE4	DECIMAL DIFFERE
1002	INC	2 7STORE4	STORE3		SCAL=ADJ PRODUC
1003	MEQLTYPELABEL		\$J	MLAB3	BUN IF NOT ROUN
1004	ADD	1 21	NLSTORE3	5 BASE	6 SET UP INFL FOR
1005	INFL		BASE	6	INFL
1007	MLAB3	MNOP			
1008	SUB	2 7BASE	16 STORE3	IX1	ADJUST IX1,RSLT
1009	GEQ	MP(IX)			BUN ON RESULT=S
1010	MVN	210STORE3	5 BASE	8 UN	ZERO IX1,SET IN
1011	INFL	BASE	16 BASE	18	INFL
1012	MP(IX)MVN	X*WRK3	11UN\$G		MOVE PRODUCT TO
1013	MEQLTYPEOMIT		\$L	MLAB4	BUN IF NO SIGN
1014	MEQLTYPEOMIT		\$L	MLAB4	HANDLING REQUIR
1015	SUB	1 21	NLBASE	18 BASE	14 UNSET IX1 TO C=FL
1016	INC	IX1	IX1		ADJUST IX1 FOR
1017	MVN	1 1X*WRK3	UN\$G	1UN	SET C FIELD SIG
1018	MLAB4	MNOP			
1019	MVN	1616SAVX*1	BASE	8 UN	RESTORE IX1 & I
1020	MEXT				
1021	MERMESMERR	EDIT PARAMETERS ILLEGAL=MTPLY MACRO			
1022	MEND				
1023*					
1024*					
1025*					
1026*					
1027*					
1028*					
1029*					
1030*					
1031*					
1032*					
1033*					



1034*									
1035*									
1036*									
1037*									
1038	MACRDIVI	\$A, \$B, \$C, \$D, \$E, \$F, \$G, \$H,							
1039		\$I, \$J, \$K, \$L, \$M							
1040	MEQLTYPEOMIT		\$K			MSTART			
1041	MEQLTYPENULL		\$K			MSTART			
1042	MBUN	MERMES							
1043	MSTARTMNOP								
1044	MVN	1616BASE	8 UNSAVX*1					SAVE IX1 & IX2	
1045	MVN	2\$B	BASE					STORE A LENGTH	
1046	MVN	2\$E	BASE	2				STORE B LENGTH	
1047	MVN	2\$H	BASE	16					
1048	MVN	2 2BASE	16 UNBASE	18 UN					
1049	MVN	1 0	NLX*WRK2						
1050	INFL	BASE	BASE						
1051	MVN	\$A	X*WRK1						
1052	ADD	2 2BASE	BASE	2	BASE	4		SET B/ LENGTH	
1053	INFL	BASE	2	BASE	4				
1054	MVN	\$D	X*WRK2						
1055	MEQLTYPEOMIT		\$C			MLAB1		FALL THROUGH IF	
1056	MEQLTYPENULL		\$C			MLAB1		PRESENT	
1057	INC	2\$C	BASE	4					
1058	INC	2\$I	BASE	4					
1059	DEC	2\$F	BASE	4					
1060	MEQLTYPELABEL		\$J			MLAB1		BUN ON J PRESEN	
1061	INC	1 21	NLBASE	4					
1062	MLAB1	MEQLTYPENULL	\$L			MLAB2		BUN TO MLAB2 IF	
1063		MEQLTYPEOMIT	\$L			MLAB2		PARAM NOT PRESE	
1064	MVN	1 1C	NLX*WRK1	UN					
1065	MVN	1 1C	NLX*WRK2	UN					
1066	MPY	5 200002	NLBASE			IX1			
1067	DEC	1 2	NLIX1						
1068	SDE	16 1X*Z=B	\$A	1					
1069	GTR	*	26						
1070	MVN	1 1D	NLX*WRK1	UN					

1071	MPY	5	200002	NLBASE	2	IX1	
1072	DEC	1	2	NLIX1			
1073	SDE	16	1X*Z=B	\$D	1		
1074	GTR	*	26				
1075	MVN	1	1D	NLX*WRK2		UN	
1076	MLAB2	MNOP					
1077	INFL		BASE	BASE	4		
1078	DIV		X*WRK1	X*WRK2		X*WRK3	
1079	SUB	2	2BASE	BASE	4	BASE	2 GET SIZE OF RES
1080	MEQLTYPELABEL			\$J		MLAB3	BUN IF T PARAM
1081	INFL			BASE	2		
1082	INC	1	5	NLX*WRK3	1	UN	
1083	DEC	1	21	NLBASE	2		SIZE OF RESULT
1084	MLAB3	MNOP					
1085	MVN	2	7BASE	2	IX1		SIZ RSLT TO IX
1086	DEC	2	7BASE	16	IX1		DEC IX1 BY C-SI
1087	GEQ		DV(IX)				
1088	MVN	2	10BASE	2	BASE	8	C-SIZ TO 17-18,
1089	INFL		BASE	16	BASE	18	
1090	DV(IX)MVN		X*WRK3	11UN\$G			MOVE QUOTIENT T
1091	MEQLTYPENULL			\$L		MLAB4	BUN IF SIGN HAN
1092	MEQLTYPEOMIT			\$L		MLAB4	NOT REQUIRED
1093	SUB	1	21	NLBASE	16	BASE	14 UNC-LENGTH TO IX1
1094	INC		IX1	IX1			
1095	MVN	1	1X*WRK3	UN\$G		1UN	TFR SIGN
1096	MLAB4	MEQLTYPEOMIT		\$M		MLAB5	BUN IF NO REMAI
1097	MEQLTYPENULL			\$M		MLAB5	REQUIRED
1098	INFL		BASE	4			
1099	MVN		X*WRK2	11UN\$M			RETURN REMAIND
1100	MLAB5	MEXT					
1101	MERMESMERR			EDIT PARAMETERS ILLEGAL:DIVIDE MACRO			
1102		MEND					
1103*							
1104*							
1105*							
1106*							
1107*							

```

1108*
1109*
1110*
1111*
1112*
1113*
1114*
1115*
1116*
1117*
1118      MACRLOOP$A,$B,$C
1119      MEQLTYPEOMIT      $B      MLAB1
1120      MEQLTYPENULL      $B      MLAB1
1121      MVA      $B      LP(IX)
1122  MLAB1 MEQLTYPEOMIT      $C      MLAB2
1123      MEQLTYPENULL      $C      MLAB2
1124      MVA      $C      LP(IX)
1125  MLAB2 MNOP
1126      DEC      1      1      NLLP(IX)
1127      GTR      $A
1128      BUN      *      +12
1129  LP(IX)CNST      04UN
1130      MEXT
1131      MEND
1132*
1133*
1134*
1135*
1136*
1137*
1138*
1139*
1140*
1141*
1142*
1143*
1144*

```

1145*					
1146*					
1147		MACRCOMP	\$A, \$B, \$C, \$D, \$E		
1148		CPA	\$A	\$B	
1149		MEQLTYPE	OMIT	\$C	MLAB1
1150		MEQLTYPE	NULL	\$C	MLAB1
1151		LSS	\$C		
1152	MLAB1	MEQLTYPE	OMIT	\$D	MLAB2
1153		MEQLTYPE	NULL	\$D	MLAB2
1154		EQL	\$D		
1155	MLAB2	MEQLTYPE	OMIT	\$E	MLAB3
1156		MEQLTYPE	NULL	\$E	MLAB3
1157		GTR	\$E		
1158	MLAB3	MEXT			
1159		MEND			

A-3. OPERATION CODE LISTING.

NOTE FOLLOWING IS A LISTING OF AUTOCODER OP-CODES, EQUIVALENT SPS OP-CODES, AND B3500 SYMBOLIC OP-CODES PRODUCED BY THE TRANSLATOR.

AUTO	SPS	B3500	REMARKS
----	---	-----	-----
A	A		ADD
		NTR	ADDITION PERFORMED BY SUBROUTINE
		CNST	6 DIGITS A-SIZE B-SIZE
		ACON	A ADDRESS
		ACON	B ADDRESS
S	S		SUBTRACT
		NTR	STRUCTURE SAME AS ADD
ZA	ZA		ZERO AND ADD
		NTR	STRUCTURE SAME AS ADD
ZS	ZS		ZERO AND SUBTRACT
		NTR	STRUCTURE SAME AS ADD
D	D		DIVIDE
		NTR	STRUCTURE SAME AS ADD
M	M		MULTIPLY
		NTR	STRUCTURE SAME AS ADD
MBC	MCW*		MOVE AND BINARY CODE
			* = D CHARACTER MUST BE CODED
			THIS INSTRUCTION NOT TRANSLATED
MBD	MCW*		SEE MBC ABOVE
			SEE MBC ABOVE

MCE	MCE		MOVE CHARACTERS AND EDIT
		NTR	EDIT COMMANDS EXECUTED BY SUBROUTINE
		CNST	4 DIGITS SIZE OF A AND B FIELDS
		ACON	A ADDRESS
		ACON	B ADDRESS
MCS	MCS		MOVE CHARACTERS AND SUPPRESS ZEROS
		NTR	STRUCTURE SAME AS MCE
MIZ	MIZ		MOVE AND INSERT ZEROS
			NOT TRANSLATED SPECIAL FEATURE USED ONLY
			IN CONNECTION WITH 7070 COMPRESSED TAPE
MLC	MCW		MOVE CHARACTERS TO A & B WORD MARK
		MVA	MOVES > 100 CHARACTERS ARE DONE BY SUBROUTINE
MCW			
MLCWA	LCA		LOAD CHARACTERS TO A WORD MARK
		MVA	SAME AS MCW
LCA			
MLNS	MN		MOVE NUMERICAL
		MVN	
MN			
MLZS	MZ		MOVE ZONE
		MVN	
MZ			
MRCM	MCM		MOVE CHARACTERS TO RECORD OR GROUP MARK
		NTR	MOVES TO RECORD MARK DONE BY SUBROUTINE
MCM		ACON	START OF A
		ACON	START OF B
MRCWG	P		SAME FORMAT AS MCM
		NTR	SAME FORMAT AS MCM

B            B            BRANCH  
                          BUN

NOTE:  
-----

IF BRANCH IS TO AN "SBR" INSTRUCTION WHICH STORES  
A RETURN POINT INTO EITHER A BRANCH TO ABSOLUTE  
ZERO OR AN INDEX REGISTER A MOVE IS GENERATED  
BEFORE THE BRANCH TO EFFECT THE LINKAGE.

BAV	B**	BRANCH ON ARIGHMETIC OVERFLOW
	MVN	** ALL BRANCH D CHARACTERS MUST BE CODED
	NEQ	THE MOVE WILL SET THE SIMULATED OVERFLOW TO ZERO(OFF) AS WELL AS SETTING COMPARISON INDICATOR TO EQL IF IT WAS PREVIOUSLY OFF.
BBE	BBE	BRANCH IF BIT EQUAL
	BOT	
	EQL	
BCE	B**	BRANCH IF CHARACTER EQUAL
	NTR	SUBROUTINE SIMULATES 1400 COMP INDICATOR
	CNST	D CHARACTER
	ACON	BRANCH TO (A ADDRESS)
	ACON	LOCATION TO BE COMPARED (B ADDRESS)
BCV	B**	BRANCH ON CARRIAGE OVERFLOW
	MVN	SAME CONVENTION AS ARITHMETIC OVERFLOW
	NEQ	
BC9	B**	BRANCH ON CARRIAGE CHANNEL 9 NOT TRANSLATED - BRANCH CHANNEL 9
BE	B**	BRANCH ON EQUAL COMARE
	BOT	
	EQL	
BEF	B**	BRANCH ON EOF OR EOR

NOT TRANSLATED - MUST BE INCLUDED AS  
ACTION LABEL IN READ OR WRITE

BER	B**		BRANCH ON TAPE TRANSMISSION ERROR NOT TRANSLATED - SAME AS BEF ABOVE
BH	B**	BOT EQL	BRANCH ON HIGH COMPARE
BIN	B**		BRANCH ON INDICATOR NOT TRANSLATED - BRANCH IF INDICATOR COVERS SEVERAL ERROR CONDITIONS. FOR THE MOST PART THESE FUNCTIONS ARE HANDLED BY THE MCP OR IOC AND ARE NOT NEEDED
BL	B**	BOT EQL	BRANCH ON LOW COMPARE
BLC	B**	BOT EQL	BRANCH ON LAST CARD THE CARD READ INSTRUCTION IS DONE BY A SUBROUTINE WHICH SETS THE INDICATOR TESTED
BLC2	B**		BRANCH ON LAST CARD 1440 NOT TRANSLATED - BRANCH LAST CARD ON READER NO. 2 - 1440 ONLY
BM	BMZ	SDE LEQ	BRANCH IF WORD WORK AND OR ZONE
BPB	B**		BRANCH ON PRINTER BUSY NOT TRANSLATED - SAME AS BIN



BSS	B**		BRANCH IF SENSE SWITCH ON
		BOT	
		EQL	
BU	B**		BRANCH ON UNEQUAL COMPARE
		BOT	
		NEQ	
BW	BWZ		BRANCH ON WORD MARK
		BOT	
		EQL	
BWZ	BWZ		BRANCH ON WORD MARK OR ZONE
		BOT	THE B3500 CODE SHOWN IS THE MAXIMUM
		EQL	FOR A BRANCH ON WORD MARK OR ZONE
		SDE	COMBINATION. IF BRANCH ON WORD MARK
		LEQ	ONLY THE BOT,EQL IS THE CODE. FOR
			BRANCH ON ZONE ONLY SDE,EQL IS PROVIDED
C	C		COMPARE
		NTR	EXECUTED BY SUBROUTINE
		CNST	4 DIGITS SIZE OF A AND B
		ACON	A ADDRESS
		ACON	B ADDRESS
CC	CC		CONTROL CARRIAGE
		MVN	IF CARRIAGE CONTROL ACTION IS IMMEDIATE
	OR		THE NTR CODE IS GENERATED TO A SUBROUTINE
		NTR	WHICH SKIPS OR SPACES ACCORDING TO THE
		CNST	PARAMETERS PASSED. IF ACTION IS DELAYED,
			SKIP AND SPACE INFORMATION IS MOVED TO THE
			PRINT SUBROUTINE TO CONTROL ACTION ON THE
			NEXT WRITE COMMAND. BOTH SUBROUTINES
			SIMULATE END OF PAGE (CHANNEL 12).
CCB	CC		CARRIAGE CONTROL AND BRANCH
		BUN	CODE AS ABOVE BEFORE BUN

CS	CS		CLEAR STORAGE
		MVA	THE BRANCH IS INCLUDED ONLY IF THE
		BUN	BRANCH OPTION IS SPECIFIED IN THE
			INSTRUCTION
CW	CW		CLEAR WORD MARK
		MVN	THE MOVE IS GENERATED ONLY IF THE
			CLEAR WORD MARK AFFECTS A SWITCH.
			OTHERWISE IT IS NOT NEEDED
SW	SW		SET WORD MARK
		MVN	SAME AS CLEAR WORD MARK
NOP	NOP		NO OPERATION
		NOP	
SAR	SAR		STORE A-ADDRESS REGISTER
			NOT TRANSLATED
SBR	SBR		STORE B-ADDRESS REGISTER
		****	SBR INSTRUCTIONS ARE TRANSLATED
			IN VARIOUS WAYS. IF SINGLE ADDRESS, IT IS
			NOT TRANSLATED, AND NOT NEEDED FOR LINKAGE
			SINCE LINKAGE IS ACCOMPLISHED IN ANOTHER WAY.
			IF TWO ADDRESS IT MAY BECOME EITHER A MOVE
			OF AN ADDRESS OR AN INC OR ADD TO AN INDEX
			REGISTER.
SS	SS		SELECT STACKER
			THE STACKER SELECT PORTION OF THE COMMAND
			SELECT STACKER AND BRANCH
		BUN	IS NOT TRANSLATED. THIS WILL HAVE NO
			EFFECT ON THE PROGRAM BUT ONLY IN OFF LINE
			MEDIA HANDLING
TR	T		NOT TRANSLATED

TRW	T		
MU	MU		MOVE UNIT
		****	TRANSLATION DEPENDS ON I-O TYPE
LU	LU		LOAD UNIT
		****	TRANSLATION DEPENDS ON I-O TYPE
CU	CU		CONTROL UNIT
		****	TRANSLATION DEPENDS ON I-O TYPE
R	R		READ A CARD
		NTR	IF BRANCH OPTION A BRANCH IS SUPPLIED
RCB	R		READ COLUMN BINARY
			NOT TRANSLATED - READ COLUMN BINARY
P	P		PUNCH
		WRIT	SEE READ
PCB	P		PUNCH COLUMN BINARY
			NOT TRANSLATED - PUNCH COLUMN BINARY
W	W		WRITE A LINE
		WRIT	SEE READ
WM	W		WRITE WORD MARKS
WR	WR		WRITE AND READ
		WRIT	WRITE-READ
		NTR	
RP	RP		READ AND PUNCH
		NTR	READ-PUNCH
		WRIT	

RF	P		READ PUNCH FEED
		NTR	READ PUNCH FEED
WP	WP		WRITE AND PUNCH
		WRIT	WRITE-PUNCH
		WRIT	
WRF	WP		WRITE AND READ PUNCH FEED
		WRIT	WRITE AND READ PUNCH FEED
		NTR	
WRP	WRP		WRITE, READ, AND PUNCH
		WRIT	WRITE-READ-PUNCH
		NTR	
		WRIT	
SRF	SRF		START READ FEED
			NOT TRANSLATED - NOT NEEDED
SPF	SPF		START PUNCH FEED
			THIS WILL SET A CARD IN MOTION
BSP	CU		BACKSPACE TAPE
			NOT TRANSLATED - THE PRIMARY USE OF
			THIS INSTRUCTION IS IN TAPE ERROR ROUTINES
RT	MU		READ TAPE
		MVA	THE MOVE IS TO ANOTHER BUFFER SET UP
		READ	BY THE TRANSLATOR. THIS RELIEVES THE
			MOD 4 REQUIREMENT WITHIN THE TRANSLATED AREA
			AS WELL AS PROVIDING BUFFERED I-O
RTB	MU		READ TAPE BINARY
		MVA	BINARY IS SPECIFIED IN DECLARATION
		READ	ALSO SEE RT

RTW	LU		READ TAPE WITH WORD MARKS
		MVA	LOAD MODE TAPES MUST BE CONVERTED
		READ	ALSO SEE RT
RWD	CU		CONTROL UNIT
		CLOS	
RWD	CU		REWIND TAPE
		CLOS	CLOSE WITH LOCK
SKP	CU		SKIP AND BLANK TAPE
			NOT TRANSLATED - USED FOR ERROR
WT	MU		WRITE TAPE
		MVA	SEE READ TAPE
		WRIT	
WTB	MU		WRITE TAPE BINARY
		MVA	SEE READ TAPE BINARY
		WRIT	
WTM	CU		WRITE TAPE MARK
		CLOS	CLOSE REEL
WTW	LU		WRITE TAPE WITH WORD MARKS
		MVA	SEE READ WITH WORD MARKS
		WRIT	
DISK			ALL DISK COMMANDS TO BE DEFINED

BURROUGHS CORPORATION  
DATA PROCESSING PUBLICATIONS  
REMARKS FORM

TITLE: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

FORM: \_\_\_\_\_  
DATE: \_\_\_\_\_

CHECK TYPE OF SUGGESTION:

ADDITION

DELETION

REVISION

ERROR

cut along dotted line

GENERAL COMMENTS AND/OR SUGGESTIONS FOR IMPROVEMENT OF PUBLICATION:

FROM: NAME \_\_\_\_\_  
TITLE \_\_\_\_\_  
COMPANY \_\_\_\_\_  
ADDRESS \_\_\_\_\_  
\_\_\_\_\_

DATE \_\_\_\_\_

STAPLE

FOLD DOWN

SECOND

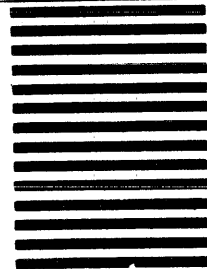
FOLD DOWN



**BUSINESS REPLY MAIL**  
First Class Permit No. 817, Detroit, Mich. 48232

Burroughs Corporation  
6071 Second Avenue  
Detroit, Michigan 48232

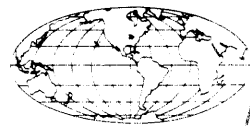
attn: Sales Technical Services  
Systems Documentation



FOLD UP

FIRST

FOLD UP



*Wherever There's  
Business There's*



**Burroughs**