

# SERIES 200

## CARD LOADER-MONITOR B

GENERAL SYSTEM:

SERIES 200/OPERATING SYSTEM - MOD 1

SUBJECT:

Program Operating Procedures for the Card Loader-Monitor B Program (formerly PLUS — Card Loader-Monitor).

SPECIAL INSTRUCTIONS:

This software bulletin supersedes the bulletin PLUS — Card Loader-Monitor (DSI-349) dated December 31, 1964. This bulletin is a companion document to the publication PLUS — Tape Loader-Monitor (DSI-327).

DATE: September 27, 1965

FILE NO.: 122.5105.040B.00.00

8443  
4965

Printed in U. S. A.

## FOREWORD

The Card Loader-Monitor B Program of the SERIES 200/OPERATING SYSTEM - MOD 1 (formerly the PLUS — Card Loader-Monitor) is designed to operate in both the two- and three-character addressing modes, permitting the storage of programs which require up to 32K characters of memory.

Card Loader-Monitor B is similar in operation to the Tape Loader-Monitor C Program (PLUS — Tape Loader-Monitor), which is described in DSI-327. The communication areas of the two programs are identical in both structure and absolute location.

The program load cards in Card Loader-Monitor B have the same format as binary run tape (BRT) records, except that they are limited to 80 characters.

Copyright 1965

Questions and comments regarding this manual should be addressed to:

Honeywell Electronic Data Processing  
Information Services  
60 Walnut Street  
Wellesley Hills, Massachusetts 02181

## TABLE OF CONTENTS

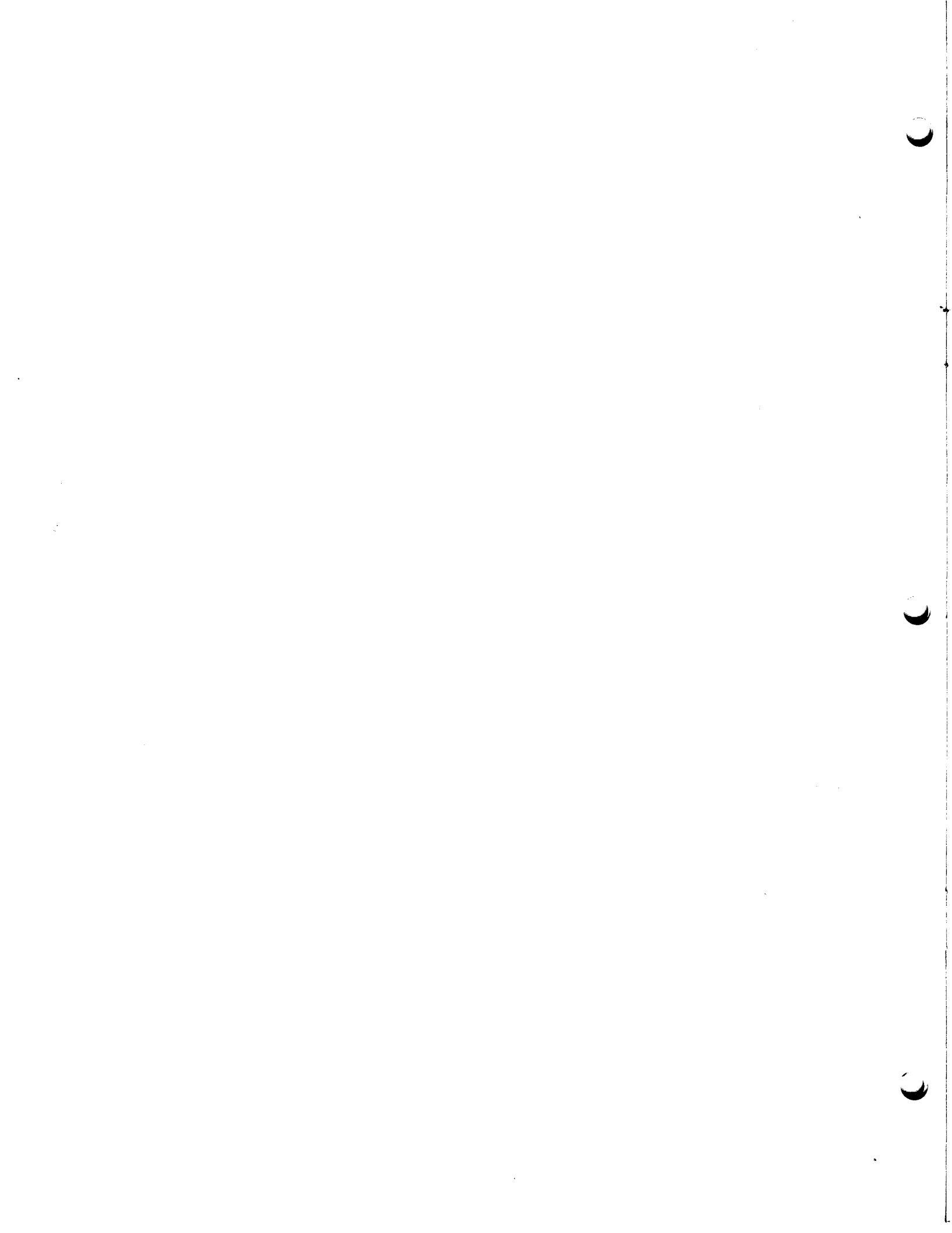
		Page
Section I	Introduction .....	1-1
	Loading Unit .....	1-1
	Binary Load Deck .....	1-1
	Equipment Requirements .....	1-2
Section II	Card Formats .....	2-1
	Format of Program Cards .....	2-1
	Segment Header Cards .....	2-1
	Non-Header Cards .....	2-1
	Loading Data .....	2-2
	Control Characters .....	2-2
	Use of Control Characters in Data Records .....	2-3
Section III	Functions and Parameters .....	3-1
	Program Calls .....	3-1
	Normal Call .....	3-1
	Special Call for EasyCoder Compatibility .....	3-1
	Searching .....	3-2
	Loading .....	3-3
	Starting .....	3-4
	Initial and Reset Values of Parameters .....	3-5
Section IV	Operating Procedures .....	4-1
	Bootstrap and Loader Operating Procedures .....	4-1
	Console Call Procedure .....	4-2
	Programmed Halts .....	4-3
Appendix A	Loader Halts .....	A-1
Appendix B	Summary of Parameters .....	B-1
Appendix C	Sequential Layout of Loader Communication Area .....	C-1

## LIST OF ILLUSTRATIONS

Figure 2-1.	Segment Header Card .....	2-1
Figure 2-2.	Non-Header Card .....	2-1

## LIST OF TABLES

Table 3-1.	Searching Parameters .....	3-2
Table 3-2.	Loading Parameters .....	3-3
Table 3-3.	Starting Parameters .....	3-4
Table 3-4.	Initial and Reset Values of Parameters .....	3-5



## SECTION I INTRODUCTION

The Card Loader-Monitor B program searches for, loads, and starts programs punched on cards in a binary run tape (BRT) format. A given program is identified by various combinations of keys or search parameters. A call to search for and load a program may originate from either the operator or another program.

If the Card Loader-Monitor B program is supplied in symbolic form, it must be assembled by the Easycoder Assembly A program or the Easycoder Assembly B program, since these are the two methods of producing a self-loading, punched card object program.

Card Loader-Monitor B loads machine-language punched cards produced by a Series 200 assembly system or compiler (for example, the punched card output produced by the Easycoder Assembly C program).

### LOADING UNIT

A loading unit is a portion of code found and loaded as a result of a single call to the Loader. The loading unit appears as a deck of (one or more) punched cards, each card representing a separate BRT record. The first card is a segment header card which contains identification and control information peculiar to the loading unit it represents; it also contains loading data (i. e., Loader control characters and object coding) for the program to be loaded. Successive cards within the loading unit contain a minimum of control information followed by loading data.

Note: The term "loading data" refers to the information which the loader manipulates, although this information is not necessarily data but includes both object coding and loader control characters.

### BINARY LOAD DECK

The binary load deck consists of machine-language punched cards such as are produced by the Easycoder Assembly C program (which can optionally specify that programs are to be punched on BRT-format cards instead of being written on a BRT). The cards are punched to simulate a BRT format (a Series 200 BRT is a tape file which is the source of the programs for the Tape Loader-Monitor C program). The binary load deck, which consists of one or more loading units, has the following format:

1. A header card for each loading unit which contains the characters 1HDRΔ in columns 1-5; (This card is not necessary. It may precede the segment header card, but, if present, it is ignored by the Loader.)

2. A segment header card for each loading unit and other loading data cards in BRT format. These cards are punched in normal mode, special code, with 80 characters per record maximum;
3. An end-of-file card containing the characters 1EOFΔ in columns 1-5 and the program name in columns 21-30; and
4. Two end-of-reserved-information cards each containing the characters 1ERIΔ in columns 1-5.

#### EQUIPMENT REQUIREMENTS

1. A Series 200 central processor with the Advanced Programming Instructions feature (011);
2. 936 core memory locations (decimal locations 64 through 999);
3. Index registers five and six; and
4. One Type 227 or 223 Card Reader.

## SECTION II CARD FORMATS

### FORMAT OF PROGRAM CARDS

Each program begins with a segment header card. This card may be followed by any number of non-header cards.

#### Segment Header Cards

The format of segment header cards is illustrated in Figure 2-1 and described below.

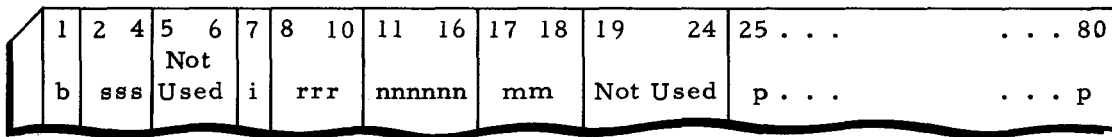


Figure 2-1. Segment Header Card

<u>Column(s)</u>	<u>Content</u>	<u>Meaning</u>
1	$b = 50_8$ or $54_8$	Constitutes banner character. An octal 50 signifies the segment header card which is <u>not</u> the last card of a loading unit. An octal 54 signifies the segment header card which is the <u>last</u> card of a loading unit.
2-4	sss	Card sequence number.
5&6		Not used.
7	$i = 30_8$	Number of locations in identification and control fields.
8-10	$000000_8 \leq rrr \leq 777777_8$	Revision number.
11-16	nnnnnn	Program name.
17 & 18	mm	Segment name.
19-24		(Visibility key not used by Card Loader).
25-80	p . . . p	Program information.

#### Non-Header Cards

The format of other binary load deck cards (which are non-header cards) is illustrated in Figure 2-2 and described below.

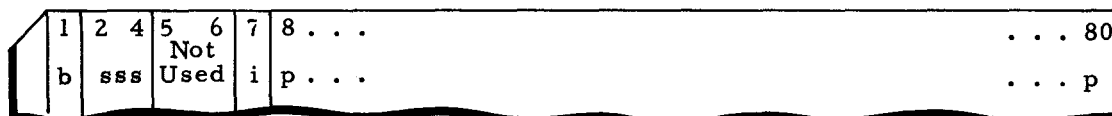


Figure 2-2. Non-Header Card

<u>Column(s)</u>	<u>Content</u>	<u>Meaning</u>
1	$b = 41_8 \text{ or } 44_8$	Constitutes banner character. An octal 41 identifies a card that is <u>not</u> the last card of a loading unit. An octal 44 identifies a card that <u>is</u> the last card of a loading unit and is not a segment header card.
2-4	sss	Card sequence number.
5&6		Not used.
7	$i = 07_8$	Number of locations in identification and control fields.
8-80	p ... p	Program information.

## LOADING DATA

### Control Characters

The data from which the loading unit is to be set up in memory follows the control information on both the header and non-header cards. The data portion of the binary load deck card is variable in length (up to the maximum record size of 80 characters). The loading data includes Loader control characters inserted by the Assembly Program that specify location and punctuation and perform various other functions. The configurations and interpretations of the control characters follow:

<u>Octal</u>	<u>Binary</u>	<u>Interpretation</u>
XX	00nnnn	Interpret nnnn as a binary number. Move the following nnnn characters to successive locations, placing the leftmost character in the location specified by the current setting of the distribution counter (X6). Clear punctuation in the locations to which characters are moved. Advance the distribution counter by the value nnnn.
XX	01nnnn	Perform the same action as for the first control character above. Then set a <u>word</u> mark in the location into which the leftmost character was moved.
XX	10nnnn	Perform the same action as for the first control character above. Then set an <u>item</u> mark in the location into which the leftmost character was moved.
60	110000	Place the following three characters into the distribution counter. The next string will be loaded with its leftmost character at this address.
61	110001	Terminate loading. Interpret the following three characters as the normal starting location of the unit just loaded.
62	110010	Clear an area of memory as specified by the following seven characters. Characters one through six are interpreted as the two three-character addresses of the lowest and highest locations to be cleared. The seventh character is transferred, with punctuation marks cleared, to every location in this area.
63	110011	Set a word mark at a location one lower than the current setting of the distribution counter.

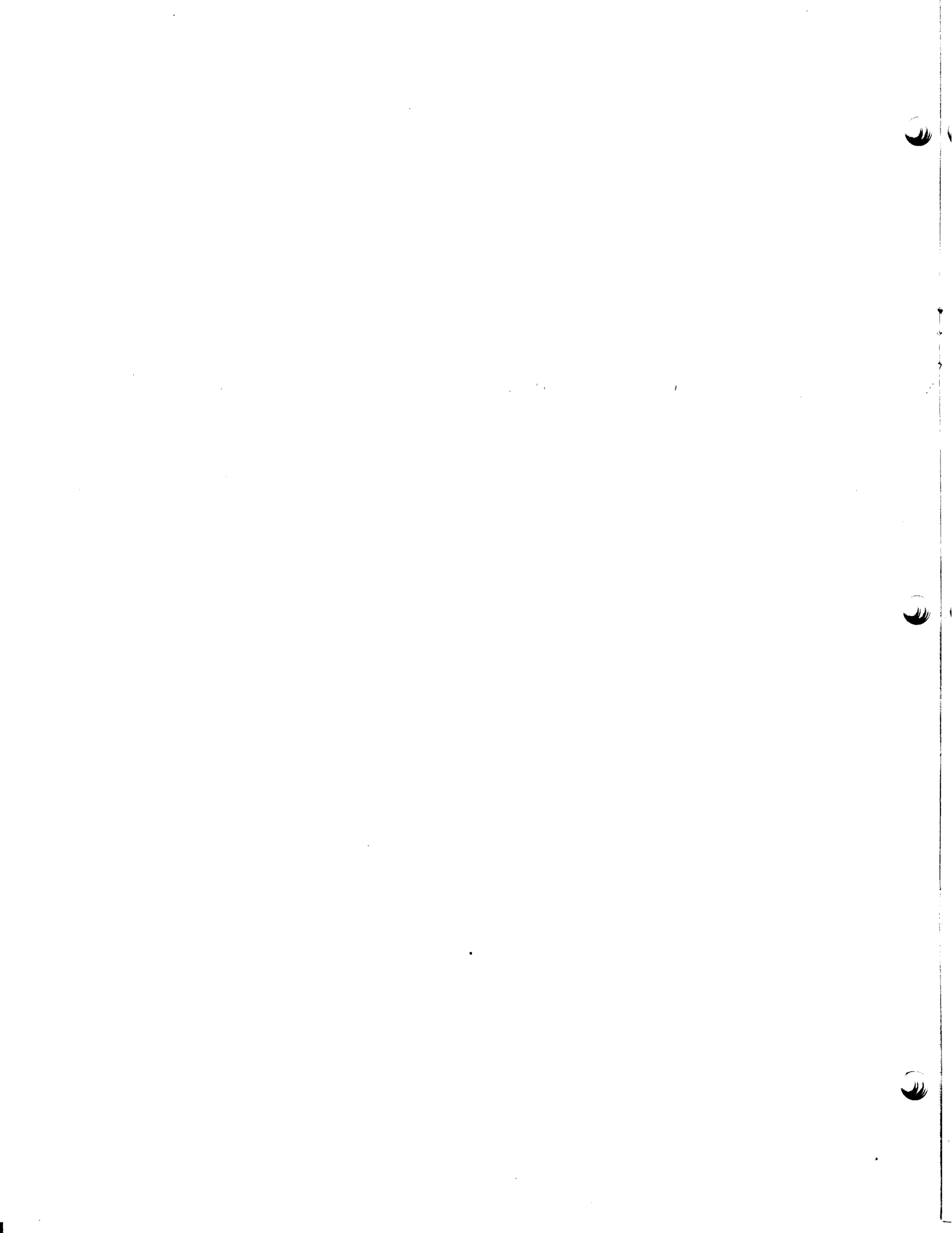


64	110100	Set an item mark at a location one lower than the current setting of the distribution counter.
77	111111	Read the next record.

#### Use of Control Characters in Data Records

The binary load deck for a unit which is to be loaded and distributed by the Loader has the following characteristics:

1. The data portion of each card begins with a control character;
2. Every card of a loading unit, except the last, terminates with an octal 77 control character;
3. The last card of a loading unit terminates with an octal 61 control character followed by a three-character address; and
4. In the control characters listed above, "nnnn" cannot be equal to 0000.



## SECTION III FUNCTIONS AND PARAMETERS

### PROGRAM CALLS

#### Normal Call

The calling unit executes a normal call to the Loader by ensuring that the parameters in the Loader communication area are properly set and then branching to decimal location 130 (see Appendix C). The calling unit must execute the branch instruction in three-character addressing mode. The Loader starts the called unit or returns to the calling unit in three-character mode.

In setting up parameters, the calling unit must not alter any punctuation marks. All communication fields are originally loaded with word marks in their leftmost locations.

The Loader uses (and does not restore) index registers five (X5) and six (X6). These registers have word marks in their leftmost locations at the completion of loading. Index register six (the distribution counter) contains an address one higher than that into which the last character of the called unit was loaded. Index register five contains the address of the control character octal 61 (in the Loader buffer) that terminated the loading operation. The three characters in the locations immediately following the address specified by index register five contain the normal starting address of the unit just loaded. The Loader does not use or disturb any locations below decimal location 64, with the exception of index registers five and six.

This section defines the Loader parameters, their locations, their initial and permissible values, the conditions under which they are reset, and the resulting Loader actions. Summaries of the searching, loading, and starting parameters are given in Tables 3-1, 3-2, and 3-3, respectively, and in Appendix B. A complete layout of the communication area is given in Appendix C.

#### Special Call for EasyCoder Compatibility

A special calling procedure is provided for compatibility with the EasyCoder Assembly A program. Under this system, an object program may call in the next unit by branching to a location obtained by storing the contents of the B-address register with the object program's first instruction (entrance point). If a unit branches to the Loader in this manner, the next unit read by the card reader is loaded and started at its normal starting location.

## SEARCHING

In searching for the required loading unit in the loading deck, the Card Loader-Monitor B recognizes a segment header card by its banner character. However, a segment header card need not be the first card read after a search is called for. Any non-header card is ignored.

In a search by program name, recognition is accomplished by comparing the program and segment name in the input buffer with the program and segment name in the Loader communication area of memory; the search is successful when a segment header card is found that contains the desired search parameter values (viz., the corresponding program and segment name). Blank cards, whether placed between programs or within a program, cause unspecified results, since they are interpreted by the Loader as segment header cards.

The Loader halts if a 1EOFΔ card is encountered in a search before the requested loading unit is found.

Table 3-1 contains the parameters that control a search (locations are given in decimal notation and values are given in octal notation).

Table 3-1. Searching Parameters

Information Specified by Parameter	Location	Values and Meaning
Search Mode	111	01 = Search by relative position. Load the (n)th unit where "n" is the value of the relative position parameter. 20 = Search for and load the unit with the specified program and segment name. 00 = Search for and load the unit with the specified segment name within the current program. This mode is similar to 20 above, except that the Loader halts upon reading a segment header card with a program name different from that of the unit last loaded. (The operator can call for a continued search by pressing the RUN button.) 60 = Search for and load the unit with specified program and segment name. (This mode is the same as 20 above; it is included for compatibility with Tape Loader-Monitor C.) 40 = Search for and load the unit with specified segment name within the current program. (This mode is the same as 00 above; it is included for compatibility with Tape Loader-Monitor C.) 77 = Unconditionally load the next unit.

Table 3-1. Searching Parameters (Cont.)

Information Specified by Parameter	Location	Values and Meaning
Program Name	68-73	This field contains the program name to be used as a search key in search modes 20, 00, 60, and 40. When a unit is loaded, its program name is placed in this field by the Loader.
Segment Name	74&75	This field contains the segment name to be used as a search key in search modes 20, 00, 60, and 40. When a unit is loaded, its segment name is placed in this field by the Loader.
Relative Position	110	This location contains the binary number "n" to be used with search mode 01. Relative position "0" is undefined.

### LOADING

If the search was successful, the Card Loader-Monitor B proceeds to set up the called unit in memory by reading successive cards and distributing and punctuating the data strings, as indicated by the control characters described on page 2-2. Loading is terminated by control character 61<sub>8</sub>. The deck then remains positioned in the reader with the next card to be read as the one immediately following the card on which the control character appeared. Table 3-2. contains the loading parameter values.

Table 3-2. Loading Parameters

Information Specified by Parameter	Location	Value and Meaning
Relocation Augment	107-109	This field contains the binary augment to be added to the base location of each load string. The same augment is applied to the starting location of the unit, if the starting mode parameter is set to "normal" (see "N", Table 3-3). The relocation augment is not added for any other starting mode. The augment is also added to the upper and lower addresses of an area to be cleared.
Halt Name	77-84	This field provides space for a program name (location 77-82) and segment name (locations 83-84). The Loader halts when the unit with this name has been loaded. When the RUN button is depressed, the Loader continues, as directed by the starting parameter.  The halt name is treated as one field. The only word mark is at location 77.  The halt name is checked against the name on the segment header card after a successful search has been made, but before loading begins. The resulting halt occurs after loading has been completed.

The normal operation of the Card Loader-Monitor may be modified using one or both of the two load parameters described in Table 3-2. The relocation augment parameter provides the ability to load a unit into a higher memory location than that for which it was assembled. However, since the Loader-Monitor performs no address adjustment, the unit is loaded into the new area in the same form as it was assembled. Thus, the relocation augment is normally used in conjunction with object programs that contain indexed addressing. Using the halt name parameter, the machine can be directed to halt after loading a particular unit by entering the name of that loading unit into the halt name parameter location in the communication area.

In addition, the Loader-Monitor also provides an own-coding exit after each card is read into the input buffer, thus enabling the execution of own-coding routines during loading. Also, an own-coding routine may return to the same point and use the distribution process (loading data into memory with appropriate punctuation) of the Loader-Monitor, or the data portion of the loading unit records may be distributed by an own-coding routine (bypassing the distribution process of the Loader-Monitor).

STARTING

Starting refers to the action performed after the called unit is loaded. The Loader may return to the calling unit or branch to a normal or special location within the called unit. The branch is always performed in the three-character addressing mode. The starting parameters are listed in Table 3-3.

Table 3-3. Starting Parameters

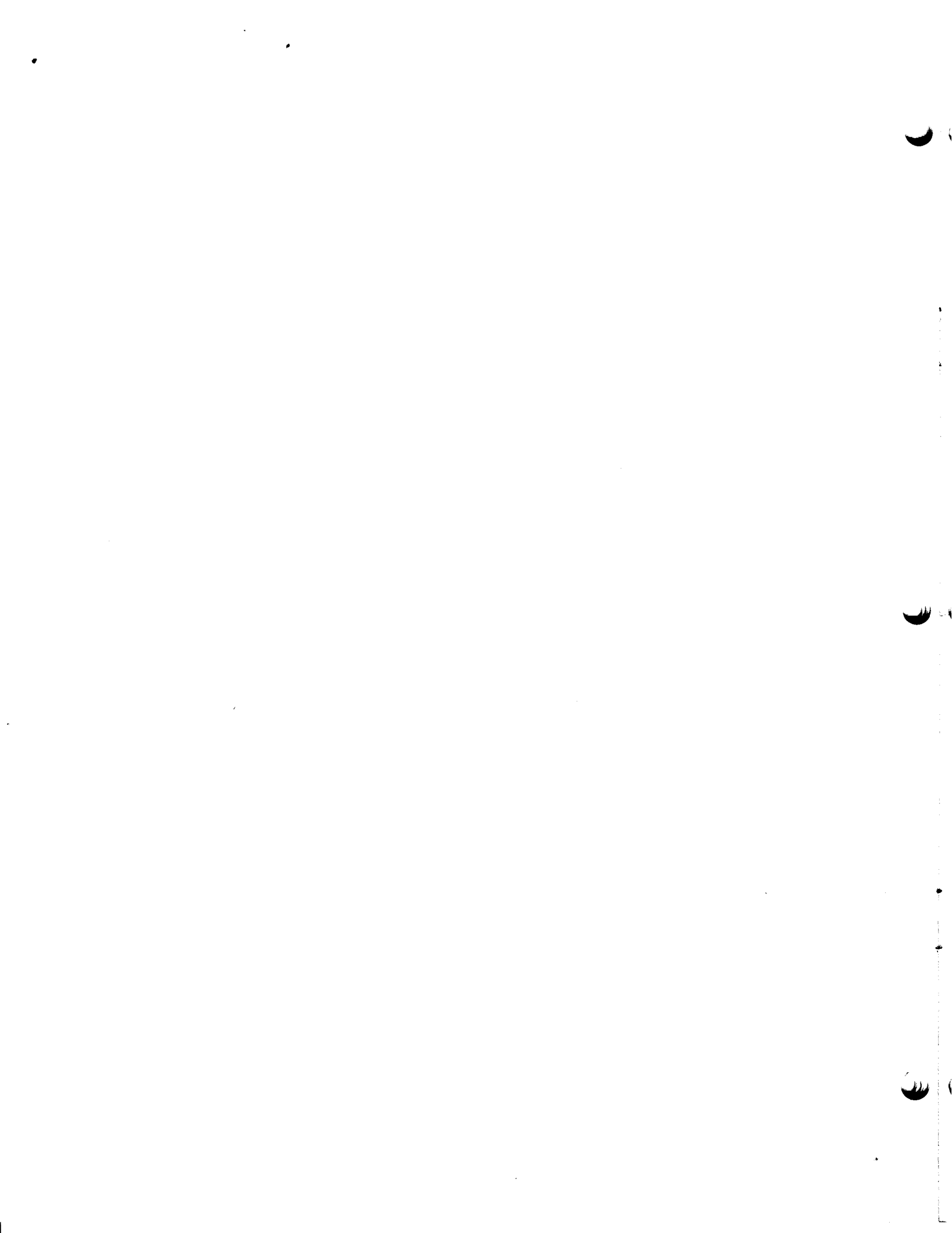
Information Specified by Parameter	Location	Value and Meaning
Starting Mode	112	N = Branch to the location specified as the normal starting location in the called program. This is the address following control character 61. The relocation augment is added before the branch is made. S = Branch to the address given by the parameter "Special Starting Location." The relocation augment is not added to this address. R = Branch to the location immediately following that from which the call to the Loader was made.
Special Starting Location	119-121	This field contains the address which the Loader branches to in order to start the called unit if the parameter "Starting Mode" has the value "S".

## INITIAL AND RESET VALUES OF PARAMETERS

The expression "initial value" refers to the contents of the corresponding field at the time the Loader is first brought into memory. "Reset" refers to a value entered into the field during execution by the Loader. The initial and reset values for the searching, loading, and starting parameters are contained in Table 3-4.

Table 3-4. Initial and Reset Values of Parameters

Parameter Type	Parameter Name	Initial Value	Reset Value
Searching	Search Mode	20	Reset to 01 by special call. Reset to 77 by console call; thus, when halt number three occurs, the "search mode" parameter has a value of 77.
	Relative Position	1	Reset to 1 after loading each unit; also reset to 1 by console call.
Loading	Relocation Augment	0	Reset to 0 after loading each unit; also reset to 0 by console call.
Starting	Starting Mode	N	Reset to N at console call and special call.
	Special Starting Location	0	Not applicable.





SECTION IV  
OPERATING PROCEDURES

BOOTSTRAP AND LOADER OPERATING PROCEDURES

The Card Loader-Monitor B program provides efficient loading of punched card programs that are placed in the card reader input hopper in the order that they are to be executed. In addition, the Loader also provides a convenient means of searching forward and selectively loading the desired program(s) according to parameter values manually entered into the communication area through the operator's control panel (or console). The Card Loader-Monitor cannot perform any action analogous to a backward search of the BRT; therefore, it does not interrogate the direction parameter in the communication area.

The Card Loader-Monitor does not accept a console call card; it is set to load unconditionally the first loading unit encountered in the card reader.

To bootstrap and load from cards, proceed as follows:

1. Place the assembled object program deck in the input hopper behind the self-loading Card Loader-Monitor deck (there may be one or more object programs, but the last object program must be followed by a 1EOFΔ card). If more than one program is to be loaded, only the last program in the binary load deck may be followed by a 1EOFΔ card;
2. Depress the INITIALIZE button;
3. Set the ADDRESS indicators to 1620;
4. Set the CONTENTS indicators to designate the address assignment of the card reader (normally 41<sub>8</sub>);
5. Cycle up the card reader;
6. Depress the BOOTSTRAP button; and
7. Depress the RUN button.
8. Check that halt number three has occurred (i. e., the A-address register contents = 00000, B-address register contents = 17002).

NOTE

When halt number three occurs, the Card Loader-Monitor is ready to accept a console call (using "search mode" 77, see Table 3-1).

9. Depress the RUN button to load the first program unconditionally; otherwise, perform steps 10 and 11 to search forward and load the desired subsequent program in the deck.
10. Enter the desired search parameter values into the appropriate fields of the communication area through the operator's control panel (or console). Do not disturb word marks in these fields.

<u>Location (Octal)</u>	<u>Contents</u>
104-111	Program name of requested loading unit.
112-113	Segment name of requested loading unit.
115-124	If a halt after loading is desired, enter the program and segment name into these locations. After the loading of the unit is completed, the machine is directed to halt. (This field is initially blank.)
157	Search mode ( $20_8$ ).

11. After performing the above entries, depress the RUN button.

If an end-of-file card is encountered before the requested loading unit is found, the Loader will halt. The operator may continue the search by placing a binary run deck in the card reader (without the Loader deck) and depressing the RUN button.

#### CONSOLE CALL PROCEDURE

A console call is a request by the operator for the loading of a unit. The console call procedure is performed after the Card Loader-Monitor is loaded and awaiting operator action (i. e., after the occurrence of halt number 3 in step "8" above).

Halt number 3 is obtained automatically at the conclusion of the loading procedure for the Card Loader-Monitor. When halt number 3 occurs, the Loader parameters have been reset to load unconditionally the next unit in the card reader and to execute a normal start. The normal start is effected when the operator depresses the RUN button (see step "9" above).

If the operator wishes to search for a specific unit by program and segment name, to start at some address other than the normal starting address, or to perform any non-standard operation, he must place the appropriate parameters in the Loader communication area (through the control panel or console) before depressing the RUN button (see steps "9", "10", and "11" above).

To obtain halt number 3 at any time other than the normal halt (following the loading of the Card Loader-Monitor), proceed as follows:

1. Set the sequence register to  $000126_8$ ;
2. Set the central processor to operate in the three-character addressing mode;

3. Depress the RUN button and wait for halt number 3 to occur.

At this point, the operator may search for a specific unit by program and segment name (as described above), or he may depress the RUN button again to load the next unit in the input hopper unconditionally.

#### PROGRAMMED HALTS

Programmed halts are listed in Appendix A. Generally, these halts result from errors. However, halts 3 and 6 are included to facilitate the operations of the Card Loader-Monitor. Halt 3 always occurs during loading to allow a console call, and halt 6 occurs when the halt name loading parameter is specified.

APPENDIX A  
LOADER HALTS

Halt Number	A Address	B Address	Meaning	Operator Action
3	000000	017002	Loader awaits console call.	If next unit in card reader is to be loaded and started under normal conditions, depress the RUN button. Otherwise, enter necessary parameters into communication area through the control panel and depress the RUN button.
5	000000	010110	Hole-count error on input card.	Depress the NON PROC RUN OUT button, correct, and refeed cards. Depress RUN button to reread.
6	000000	014000	"Halt Name" unit has been loaded.	Perform action directed by object program operating instructions. Depress the RUN button.
8	000000	014011	Called unit not found on forward search prior to reading end-of-file card.	Operator may load a deck in card reader (without Loader deck) and continue search by depressing the RUN button.
9	000000	014012	Called segment was not found within current program.	Operator may continue search by placing a deck in the card reader (without loader deck) and depressing the RUN button.

APPENDIX B  
SUMMARY OF PARAMETERS

Parameter Type	Parameter Name	Decimal Location		Octal Location		Values	Initial Value	Reset at Console Call	Reset After Loading	Reset Upon Special Call
		From	To	From	To					
Searching	Search Mode	111		157		01 = Relative Position 20 = Program and Segment 00 = Segment Within Program 60 = Program and Segment 40 = Segment Within Program 77 = Load next unit unconditionally	20	77		01
	Program Name	68	73	104	111					
	Segment Name	74	75	112	113					
	Relative Position	110		156		1 = Load next unit	1	1	1	
Loading	Relocation Augment	107	109	153	155		0	0	0	
	Halt Name	77	84	115	124					
Starting	Start Mode	112		160		R = Return N = Normal S = Special	N	N		N
	Special Starting Location	119	121	167	171		0			

APPENDIX C  
SEQUENTIAL LAYOUT OF LOADER COMMUNICATION AREA

Decimal Location		Octal Location		Function
From	To	From	To	
64		100		Not Used.
65	67	101	103	Revision number of unit last loaded.
68	73	104	111	Program name.
74	75	112	113	Segment name.
76		114		Not Used.
77	84	115	124	Halt name.
85		120		Not Used.
86	89	126	131	Fixed start 0. Start here for console call.
90	93	132	135	Fixed start 1. Not used by Loader.
94	97	136	141	Fixed start 2. Not used by Loader.
98	101	142	145	Fixed start 3. Not used by Loader.
102	105	146	151	Branch to own-coding routine.
106	152			Not used.
107	109	153	155	Relocation Augment.
110		156		Relative position.
111		157		Search mode.
112		160		Start mode.
113	118	161	166	Not used.
119	121	167	171	Special starting location.
122	125	172	175	Own-coding return 1 - before distribution.
126	129	176	201	Own-coding return 2 - after distribution.
130	138	202	212	Loader entrance for normal call.
139	141	213	215	General return address.
142	146	216	222	Current date.
147		223		Not used.
148	150	224	226	Alternate return address.
151	154	227	232	ECD field
155		233		Console typewriter.

COMPUTER-GENERATED INDEX

AREA  
 SEQUENTIAL LAYOUT OF LOADER COMMUNICATION AREA, C-1  
 BINARY LOAD DECK, 1-1  
 BOOTSTRAP AND LOADER OPERATING PROCEDURES, 4-1  
 CALL PROCEDURE  
 CONSOLE CALL PROCEDURE, 4-2  
 CALLS  
 PROGRAM CALLS, 3-1  
 CARD  
 FORMAT OF PROGRAM CARDS, 2-1  
 " FORMATS, 2-1  
 NON-HEADER CARD, 2-1  
 SEGMENT HEADER CARD, 2-1  
 CHARACTERS  
 CONTROL CHARACTERS, 2-2  
 USE OF CONTROL CHARACTERS IN DATA RECORDS, 2-3  
 COMMUNICATION AREA  
 SEQUENTIAL LAYOUT OF LOADER COMMUNICATION AREA, C-1  
 COMPATIBILITY  
 SPECIAL CALL FOR EASYCODER COMPATIBILITY, 3-1  
 CONSOLE CALL PROCEDURE, 4-2  
 CONTROL CHARACTERS, 2-2  
 USE OF CONTROL CHARACTERS IN DATA RECORDS, 2-3  
 DATA  
 LOADING DATA, 2-2  
 " RECORDS,  
 USE OF CONTROL CHARACTERS IN DATA RECORDS, 2-3  
 DECK  
 BINARY LOAD DECK, 1-1  
 EASYCODER COMPATIBILITY  
 SPECIAL CALL FOR EASYCODER COMPATIBILITY, 3-1  
 EQUIPMENT REQUIREMENTS, 1-2  
 FORMAT  
 CARD FORMATS, 2-1  
 " OF PROGRAM CARDS, 2-1  
 FUNCTIONS AND PARAMETERS, 3-1  
 HALTS  
 LOADER HALTS, A-1  
 PROGRAMMED HALTS, 4-3  
 HEADER CARD  
 SEGMENT HEADER CARD, 2-1  
 INITIAL AND RESET VALUES OF PARAMETERS, 3-5  
 INTRODUCTION, 1-1  
 LAYOUT  
 SEQUENTIAL LAYOUT OF LOADER COMMUNICATION AREA, C-1  
 LOAD DECK  
 BINARY LOAD DECK, 1-1

LOADER  
 " COMMUNICATION AREA,  
 SEQUENTIAL LAYOUT OF LOADER COMMUNICATION AREA,  
 C-1  
 " HALTS, A-1  
 " OPERATING PROCEDURES,  
 BOOTSTRAP AND LOADER OPERATING PROCEDURES, 4-1  
 LOADING, 3-3  
 " DATA, 2-2  
 " PARAMETERS, 3-3  
 " UNIT,  
 LOADING UNIT, 1-1  
 NON-HEADER CARD, 2-1  
 NORMAL CALL  
 NORMAL CALL, 3-1  
 OPERATING PROCEDURES  
 BOOTSTRAP AND LOADER OPERATING PROCEDURES, 4-1  
 PARAMETERS  
 FUNCTIONS AND PARAMETERS, 3-1  
 INITIAL AND RESET VALUES OF PARAMETERS, 3-5  
 LOADING PARAMETERS, 3-3  
 SEARCHING PARAMETERS, 3-2  
 STARTING PARAMETERS, 3-4  
 SUMMARY OF PARAMETERS, B-1  
 PROCEDURE  
 BOOTSTRAP AND LOADER OPERATING PROCEDURES, 4-1  
 CONSOLE CALL PROCEDURE, 4-2  
 PROGRAM  
 " CALLS, 3-1  
 " CARDS,  
 FORMAT OF PROGRAM CARDS, 2-1  
 PROGRAMMED HALTS, 4-3  
 RECORDS  
 USE OF CONTROL CHARACTERS IN DATA RECORDS, 2-3  
 REQUIREMENTS  
 EQUIPMENT REQUIREMENTS, 1-2  
 RESET VALUES  
 INITIAL AND RESET VALUES OF PARAMETERS, 3-5  
 SEARCHING, 3-2  
 " PARAMETERS, 3-2  
 SEGMENT HEADER CARD, 2-1  
 SEQUENTIAL LAYOUT OF LOADER COMMUNICATION AREA, C-1  
 SPECIAL CALL  
 SPECIAL CALL FOR EASYCODER COMPATIBILITY, 3-1  
 STARTING, 3-4  
 " PARAMETERS, 3-4  
 SUMMARY OF PARAMETERS, B-1

**HONEYWELL  
ELECTRONIC  
DATA  
PROCESSING**  
WELLESLEY HILLS,  
MASSACHUSETTS 02181