

BASIC PROGRAMMING SYSTEM AND MOD 1 (TR)

1/2-INCH TAPE INPUT/OUTPUT B AND C

GENERAL SYSTEM:

SERIES 200/PROGRAMMING SYSTEMS

SUBJECT:

1/2-Inch Tape I/O B and C, the Input/Output Package for Processing Fixed- or Variable-Length Records on 1/2-Inch Magnetic Tapes.

SPECIAL
INSTRUCTIONS:

This software manual does not supersede the manual of the same name, dated February 28, 1966, which is current. A list of new and changed information appears on page iv.

INCLUDES UPDATING MATERIALS PUBLISHED AS ADDENDUM #1 ON SEPTEMBER 23, 1966 AS ADDENDUM #2 ON APRIL 20, 1967, AND AS ADDENDUM #3 ON JUNE 30, 1967.

DATE: August 29, 1966

FILE NO.: 123.4105.020B.1-010*

9881
3867

Printed in U. S. A.

*Underscoring denotes Order Number.

PREFACE

Input/output control for the Series 200 may be efficiently performed through the use of Honeywell-supplied macro packages. The standard package designed to process 1/2-inch magnetic tape files containing either fixed- or variable-length records is 1/2-Inch Tape I/O B and C. This package is operable in two memory ranges; it may be implemented in the range of 8,192 characters of memory (I/O B) as the tape I/O portion of an independent program, or it may be implemented in the range of 12,288 characters of memory in conjunction with the Series 200/Operating System - Mod 1 (I/O C). When the package is implemented in the larger configuration as I/O C, several features are modified to accompany the functional capabilities of the larger system; e. g., console messages are optionally a part of I/O C but are not implemented in I/O B. Several parameters used in I/O B macro statements are modified for I/O C, specifically in the DCA (Define Communication Area) statement and the TIOC (Tape Input/Output Configuration) statement. The I/O C addressing modes permit the flexibility of either three- or four-character addressing while I/O B is limited to the three-character mode. Wherever differences in implementation exist for one of the specified memory ranges, such differences are noted.

The 1/2-Inch Tape I/O B and C package is functionally compatible with 1/2-Inch Tape I/O A, the basic version of the input/output control program which accepts only fixed-length records. The macro instructions are the same in all versions, although the parameters for each differ slightly. As a result, programs written for the 4K memory range can be easily modified for use in the 8K and 12K memory ranges. The 1/2-Inch Tape I/O B and C package is used in conjunction with user programs assembled by any version of the EasyCoder Assembly System. For programs to be assembled by EasyCoder Assembler A or B, Library Processor B is used prior to assembly to specialize the 1/2-Inch Tape I/O B and C macro routines. For programs assembled by EasyCoder Assembler C or D, this function is performed by Library Processor C or D.

It is necessary to note that the 1/2-Inch Tape I/O B and C package is not used with Standard I/O Calls C, although the two may reside on the same symbolic library file (SPT).

It is assumed that the reader is familiar with the Honeywell Series 200 Programmers' Reference Manual (Models 200/1200/2200/4200), Order No. 139, or (Model 120), Order No. 141, and the following software publications:

Library Processors C and D, Order No. 051

EasyCoder Assemblers C and D, Order No. 041

EasyCoder B Assembly System, Order No. 011 (includes Library Processor B).

TABLE OF CONTENTS

		Page
Section I	Basic Concepts	1-1
	Introduction	1-1
	Equipment Needed	1-1
	Functional Description	1-2
Section II	Preparing to Process a Tape file	2-1
	File Formats	2-1
	Unbannered Records	2-1
	Bannered Records	2-1
	Labels	2-1
	Header Labels	2-2
	Header Labels for New Tapes	2-3
	Trailer Labels	2-3
	Tape Marks	2-4
	File Macros	2-4
	DCA Macro	2-5
	Sample DCA Macro Statements	2-9
	TIOC Macro	2-10
	Action Macros	2-12
	@ OPEN	2-12
	@ GET	2-13
	@ PUT	2-14
	@ PUT (LOCATE Mode)	2-14
	@ PUT (MOVE Mode)	2-15
	@ CLOSE	2-15
@ FEOR	2-16	
Section III	Detailed Functional Description	3-1
	Beginning-of-File Subroutine (@ OPEN)	3-1
	Input Files	3-1
	Output Files	3-1
	Read/Write Subroutine (@ GET and @ PUT)	3-2
	End-of-Reel-or-File Subroutine (@ FEOR and @ CLOSE)	3-2
	Input Files	3-2
	Output Files (End of Reel)	3-3
Output Files (End of File)	3-4	
Description of Exits	3-4	
Section IV	Operating Procedures	4-1
	Program Halts	4-1
	Program Timeouts (I/O C Only)	4-1
Appendix A	Sample 1/2-Inch Tape I/O B and C Source Program	A-1

LIST OF ILLUSTRATIONS

		Page
Figure 2-1.	Record Formats Acceptable to 1/2-Inch Tape I/O B and C	2-2
Figure 2-2.	Standard Header Label	2-3
Figure 2-3.	Standard Trailer Labels	2-4
Figure 2-4.	DCA Macro Statement (Full Form)	2-10
Figure 2-5.	DCA Macro Statement (Condensed Form)	2-10
Figure 2-6.	Sample @OPEN Macro and Resulting Generated Coding	2-13
Figure 2-7.	Sample @GET Macro and Resulting Generated Coding	2-14
Figure 2-8.	Sample @PUT Macro and Resulting Generated Coding	2-15
Figure 2-9.	Sample @CLOSE Macro and Resulting Generated Coding	2-16
Figure 2-10.	Sample @FEOR Macro and Resulting Generated Coding	2-16
Figure A-1.	Sample Program Using 1/2-Inch Tape I/O B and C	A-1

LIST OF TABLES

Table 2-1.	Parameters Used With DCA Macro Statement	2-5
Table 2-2.	Parameters Used With TIOC Macro Statement	2-11
Table 4-1.	Halt Codes and Operator Action for 1/2-Inch Tape I/O B and C	4-1
Table 4-2.	Typeout Messages and Operator Action	4-2

NEW AND CHANGED INFORMATION

<u>Page</u>	<u>Figure(s)</u>
2-12	Coding Example
A-1	Page 1 of A-1
A-2	Page 2 of A-2

SECTION I
BASIC CONCEPTS

INTRODUCTION

The 1/2-Inch Tape I/O B and C package processes 1/2-inch magnetic tape files containing either fixed- or variable-length records on Honeywell Series 200 computer systems. Using five macro instructions (@GET, @PUT, @OPEN, @CLOSE, and @FEOR), programmers can call macro routines to:

1. Open and close tape files,
2. Read and write records within files,
3. Unblock and block items within records, and
4. Detect read/write errors and correct when possible.

EQUIPMENT NEEDED

The 1/2-Inch Tape I/O B and C package is used with the following minimum hardware configuration:

1. Series 200 processor with Advanced Programming Instructions (Feature 010,011, or 1011).
2. Approximately 2,200 to 3,500 core memory locations plus up to 73 locations for each file to be processed by I/O B. Approximately 2,200 to 4,500 core memory locations plus up to 75 locations for each file processed by I/O C.
3. One or more Type 204B Magnetic Tape Units.

For specialization by Library Processor B and assembly by EasyCoder Assembler A or B, the following hardware is needed:

1. Series 200 processor with Advanced Programming Instructions (Feature 010,011, or 1011).
2. 8,192 locations in core memory.
3. Card Reader or one Type 204B Tape Unit.
4. Card Punch or one Type 204B Tape Unit.
5. Printer, if EasyCoder assembly listing is desired.

NOTE: A Type 224-1, 224-2, or 214-2 cannot be used as both a card reader and a punch by this program.

For specialization by Library Processor C or D and assembly by EasyCoder Assembler C or D, the following hardware is needed:

1. Series 200 processor with Advanced Programming Instructions (Feature 010,011, or 1011).

2. 12,288 locations of core memory.
3. Card Reader or Type 204B Tape Unit.
4. Three Type 204B Tape Units.
5. Printer, if EasyCoder assembly listing is desired.

Optional equipment may include:

1. IBM Code Compatibility (Feature 051).
2. IBM Format Compatibility (Feature 050) - I/O B only.
3. Auxiliary Read/Write Channel (Feature 016) - I/O B only.
4. Type 220-1, 220-2, or 220-3 Console - I/O C only.
5. Second Input/Output Sector for the Type 2201 Central Processor (Feature 1115) - I/O C only.

FUNCTIONAL DESCRIPTION

The main control routine of the package, called the Tape Input/Output Configuration (TIOC) routine, is used in processing all files to perform such common functions as reading, writing, blocking, unblocking, opening and closing files, and error checking.

For each file, the programmer must furnish an associated communication area by writing a DCA (Define Communication Area) macro statement. By entering descriptive parameters in the DCA macro statement, the programmer supplies all the information needed by the TIOC macro routine for controlling and processing the file.

To use this I/O package the programmer must issue one TIOC macro call with parameters and define a communication area for each specific file by means of descriptive parameters. He may then perform file handling operations by means of five action macro instructions: @OPEN, @GET, @PUT, @CLOSE and @FEOR (Force End of Reel).

The @GET and @PUT macro instructions can access data items in either of two distinct modes, known as the "locate" mode and the "move" mode. If the programmer chooses to operate in the locate mode, he specifies an index register as the linkage between the tape I/O package and the main program. The I/O package uses the specified register to provide the address of the next input item or the address of the next available output buffer location. In the locate mode, the programmer moves data items into and out of his buffers. If the programmer chooses to operate in the move mode, he specifies as a linkage the address of the work area. In this mode, the tape I/O package handles the moving of items between the work area and the input and output buffers.

One of the main advantages accruing to the user of the 1/2-Inch Tape I/O B and C package is the fact that all blocking and unblocking of data records is handled automatically and he is

presented with the individual items for processing. It should be noted that fixed-length records are always composed of a fixed number of fixed-length items, whereas variable-length records are always composed of a variable number of variable-length items. Any single file to be processed by the tape I/O package may contain either fixed- or variable-length records, but not both; although a file of fixed-length records and a file of variable-length records can be processed during the same program run.

Buffers and move-mode work areas must be reserved and cleared of extraneous punctuation by the programmer, and their high-order addresses must be specified in the appropriate parameters of the associated DCA macro statements. A record mark must appear in the location immediately following the buffer. For fixed-length items, a buffer must be equal in size to the length of one item times the number of items per record (plus one location if each data record is preceded by a banner character). For variable-length items, an output buffer must be equal to or greater than the length of the maximum size item plus five locations, and an input buffer must be equal to or greater than the length of the maximum size input record.

When the same buffers are used concurrently for input and output, the result is unspecified. Buffers must also be cleared and repunctuated before using them for another file (even if not used concurrently).

I/O B macro routines are assembled by Easycoder in the three-character addressing mode; therefore, all entries to and exits from the I/O B package must be in the three-character mode. I/O C macro routines may be assembled in either three- or four-character addressing modes as specified by the ADMODE statement in the user's program. I/O C macros must be assembled in the same address mode as the highest memory address used in the program; i. e., if the program uses memory addresses extending beyond the range 0 to 32,768 characters of memory, four-character addressing must be used in order to permit references to the higher memory locations specified.

SECTION II
PREPARING TO PROCESS A TAPE FILE

FILE FORMATS

The 1/2-Inch Tape I/O B and C package processes files containing either fixed- or variable-length items in the following formats (see Figure 2-1).

Unbannered Records

1. Fixed-length, Unblocked: All data records within one file are of a fixed and constant length. All characters of each record are the data of one item.
2. Fixed-length, Blocked: All data records within one file are of a fixed and constant length. Within each physical tape record is a fixed and constant number of logical items, each of which is of a fixed and constant length. All characters of each item are the data of that item.
3. Variable-length: All data records within one file contain a variable number of variable-length items. Each record has a signed decimal count of the characters within that record (including the count itself); these form the first four characters of the record. Each item within a record contains an unsigned decimal count of the characters within that item (including the count itself and the character record mark, octal 72, if present); these form the first three characters of the item. This type of file is ordinarily read and written in even parity using Code Compatibility (Feature 051).

Bannered Records

1. Fixed-length, Unblocked: All data records within one file are of a fixed and constant length. The first character of each record is the identifying banner character; otherwise, it is the same as an unbannered record of this type.
2. Fixed-length, Blocked: All data records within one file are of a fixed and constant length. The first character of each record is the identifying banner character; otherwise, it is the same as the unbannered record of this type.
3. Variable-length: All data records within one file contain a variable number of variable-length items. Each record has the identifying banner character as its first record; the next three characters contain a binary count of the number of characters within the record (including the banner character and the count itself). Each item within the record has a binary count of the number of characters within the item (including the count itself); these form the first two characters of the item. This type of file is ordinarily read and written in odd parity.

LABELS

The tape I/O package processes files having three different types of header and trailer label records. If the user specifies standard Series 200 labels, the package will read and write

the labels and may perform label checking, if specified. If the user specifies non-standard labels for files containing labels which do not conform to Series 200 standards, the I/O package provides exits from which the user may do his own reading, writing, and checking of labels. Or the user may specify no labels at all, and the I/O package bypasses all label processing.

NOTE: In the special case where non-standard output labels are exactly as long as the label area specified, the tape I/O package allows them to be described as standard (so that non-standard labels can be written by the I/O package), but the user must specify that no label checking is to be performed on such a file.

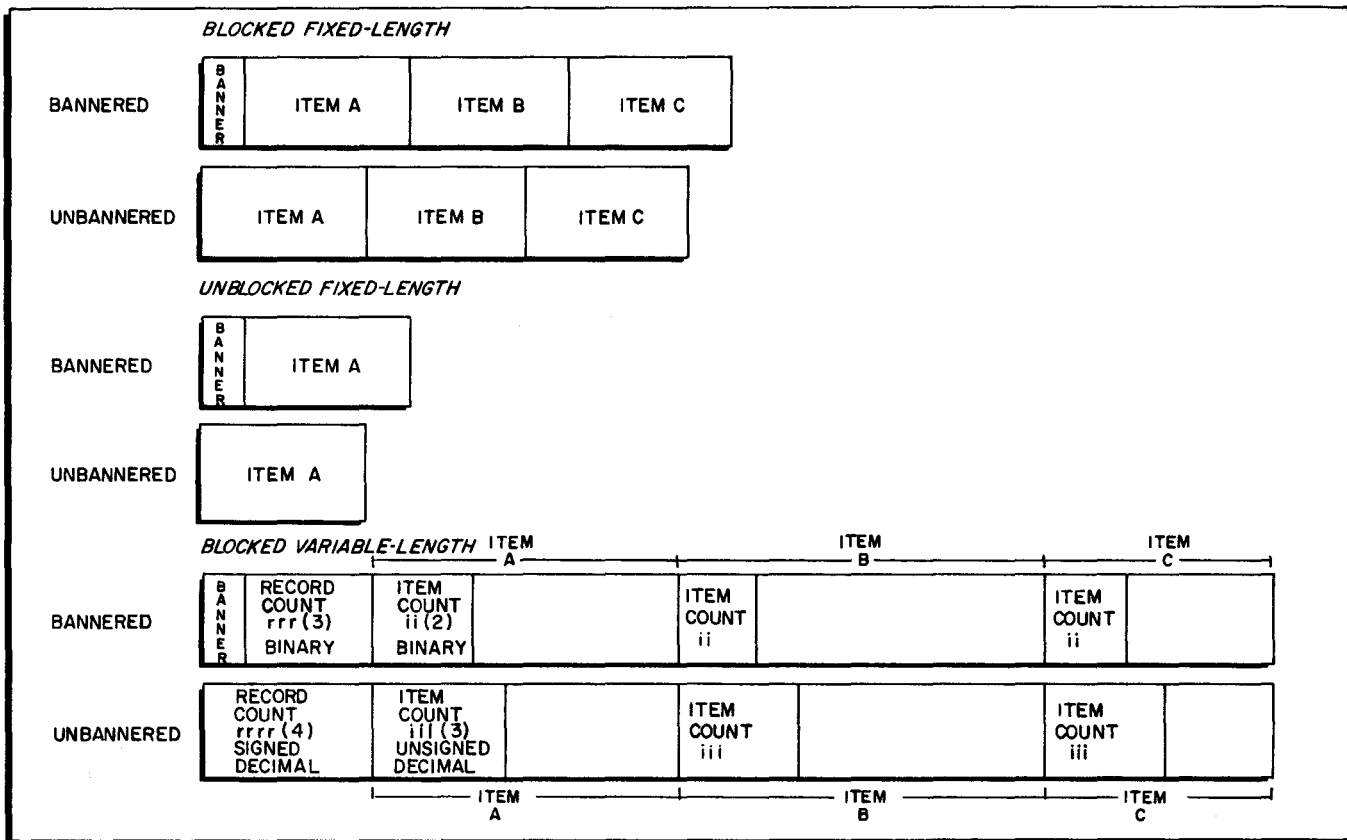


Figure 2-1. Record Formats Acceptable to 1/2-Inch Tape I/O B and C

Header Labels

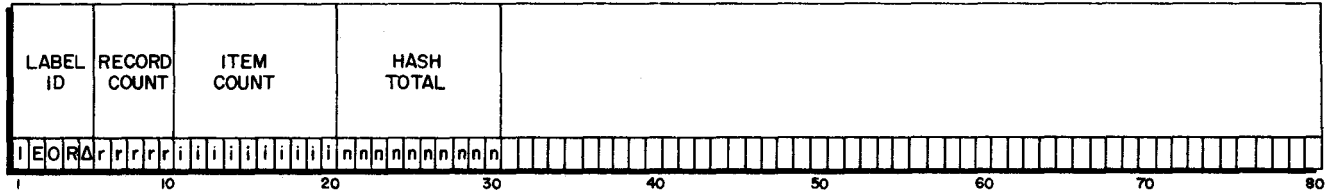
The header label is the identification record at the beginning of each file. The tape I/O package accepts, and creates, standard header labels having the following eight fields (see Figure 2-2):

<u>Character Position</u>	<u>Contents</u>	<u>Description</u>
1-5	1HDRΔ	Header flag (identifies the header label).
6-10	ttttt	Tape serial number (permanent on this reel). ¹
11-15	nnnnn	File serial number.

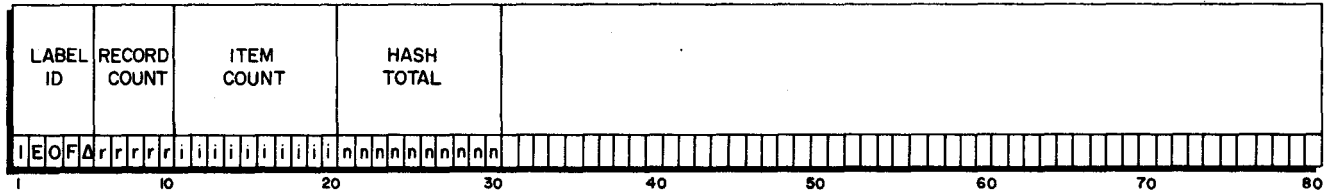
¹The I/O package neither creates nor destroys these fields.

<u>Character Position</u>	<u>Contents</u>	<u>Description</u>
	IERIA	End of reserved information.
6-10	rrrrr	Record count.
11-20	iiiiiii	Item count (optional). ¹
21-30	hhhhhhh	Hash total (optional). ¹
31-80		Available for additional information, if desired.

END-OF-REEL LABEL



END-OF-FILE LABEL



END-OF-RESERVED-INFORMATION LABEL

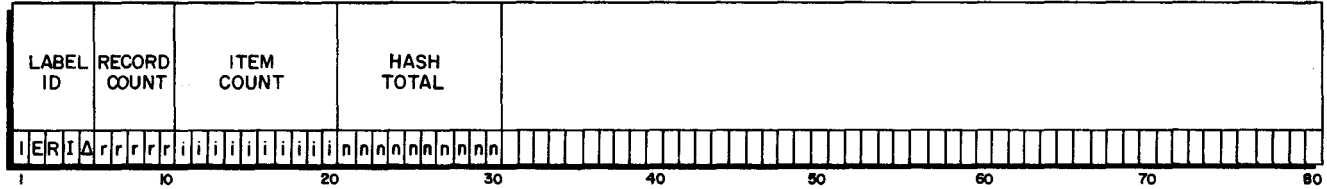


Figure 2-3. Standard Trailer Labels

TAPE MARKS

The tape I/O package reads and writes tapes which use tape mark sensing (T/M tapes), as well as standard Honeywell tapes (non-T/M tapes). To read T/M tapes with I/O B, the tape control being used must be equipped with Feature 050 (IBM format feature). A T/M tape contains a tape mark before the trailer label.² When the tape mark is sensed, the tape I/O package accepts the record which follows as a trailer label.

FILE MACROS

The programmer describes his file processing by issuing two macro statements: DCA (Define Communication Area) and TIOC (Tape Input/Output Configuration).

¹ The I/O package does not compute these values. If they are desired, they must be computed and stored in the trailer label record by the user.

² Also, a T/M tape may contain a tape mark following the header label and/or the trailer label. If present, these additional tape marks are automatically bypassed by 1/2-Inch Tape I/O B and C.

DCA Macro

The programmer furnishes a DCA macro statement for each file to be processed by his program. This macro instruction consists of the op code DCA and one or more of the 29 parameters listed in Table 2-1, which describe the file with regard to name, data format, and label format, as well as user linkages, exits, and buffer addresses. (These parameters are compatible with those specified in the 22 lines of the DCA statement of 1/2-Inch Tape I/O A, except parameter 19, which specifies buffer size for variable-length records. In order to convert the DCA statement of a program which currently uses 1/2-Inch Tape I/O A to run with 1/2-Inch Tape I/O B and C, the necessary "C" and "L" codes must be added to column 6, and a comma must terminate each line.)

The format of the DCA macro as written on the EasyCoder coding form is as follows:

<u>Column No.</u>	<u>Data</u>
1-5	Line number, if desired.
6	"L" for the last card in the macro call; "C" for all others.
7	Δ
8-14	Symbolic tag, if desired.
15-20	DCA (first card of macro call)
21-62	Parameters separated by and terminated by commas (see Table 2-1). All parameter values are alphabetic or decimal, unless specified otherwise. The first parameter (file name) is written in the same line as the op code DCA.

The following parameters are always required: 1, 2, 7, 13 and 16. The others may be omitted or left blank, in which case the tape I/O package will assume certain values, as stated in the table.

Table 2-1. Parameters Used With DCA Macro Statement

Parameter		Value	Description
Number	Name		
1	File Name	Tag	Name assigned to this tape file by the user. This tag cannot exceed six characters in length and must be unique within any one main program.
2	File Type	INPUT or OUTPUT	Indicates whether file is to be read or written.
3	Parity	EVEN or ODD or omit	If no entry, odd parity is assumed.

SECTION II. PREPARING TO PROCESS A TAPE FILE

Table 2-1 (cont). Parameters Used With DCA Macro Statement

Parameter		Value	Description
Number	Name		
4	Gap	LONG or SHORT or omit	If no entry, long gap (0.75") is assumed. Note that a long-gap tape is the only tape that can be read on <u>all</u> types of 204B tape drives.
5	RWC (This parameter is unused in the I/O C package.)	1, 2, 3, or 5 or omit	Specifies the read/write channel to be used by this file. If omitted, a free channel will be selected and used by the I/O package. The value 5 selects RWC1', the auxiliary read/write channel.
6	Tape Control I/O B	0 - 7 or omit	Specifies the tape control to be used by this file; if omitted, 0 is assumed.
	I/O C	00 - 07, 20 - 27, or omit	Specifies the tape control to be used by this file; if omitted, 00 is assumed.
7	Tape Drive Address	0 - 7	Specifies first logical tape drive address to be used by this file.
8	Alternate Tape Drive Address	0 - 7 or omit	Indicates alternate drive to be used if this is a multireel file. If omitted, a single-reel file, or a multireel file which continues on the same logical drive as that specified by parameter 7, is assumed. The alternate tape drive parameter can specify the same logical tape drive as parameter 7, in which case the I/O package comes to a tape swap halt and the user can mount a new reel on the same tape drive.
9	File Mode	LOCATE or MOVE or omit	Specifies whether "locate" or "move" mode is desired. If omitted, locate mode is assumed. (See action macros @GET and @PUT.)
10	Record Form	FIXED or VARIABLE or omit	Indicates whether records in this file are of fixed or variable length. If omitted, fixed-length records are assumed.

SECTION II. PREPARING TO PROCESS A TAPE FILE

Table 2-1 (cont). Parameters Used With DCA Macro Statement

Parameter		Value	Description
Number	Name		
11	Banner	A 2-digit octal number greater than 40, BAN, NOBAN, or omit	Specifies that data records are to be preceded by this banner character. If omitted or BAN, octal 56 will be used as the banner character. If NOBAN, records do not include banners. (See also parameter 3.)
12	Blocking Factor	A one- to four-digit number from 1 - 4,095	Specifies the number of fixed-length items in a tape record. 1 indicates unblocked records. If items are of variable length, this parameter must be omitted.

Table 2-1 (cont). Parameters Used With DCA Macro Statement

Parameter		Value	Description
Number	Name		
13	Item Size	A one- to four-digit number from 1 - 4,095	Specifies the number of characters in each fixed-length item or the maximum number of characters in any variable-length item. This parameter must be specified.
14	Number of Buffers	1 or 2 or omit	Specifies the number of buffers to be used for this file. If no entry, 1 is assumed.
15	Padding	One character or omit	<p>Character to be used to pad incomplete fixed-length output records, or to bypass those same records on input. This parameter is not entered for variable-length items.</p> <p>The padding character should differ from any leading character in the input file. Note, however, that a blank is a valid padding character.</p> <p>In I/O B, one character at the beginning of each item is checked on input; while in I/O C, three characters are checked. Any item whose checked number of characters correspond to the specified padding character number is considered to be a padded item and is bypassed.</p>
16	First Buffer Address (BUFF1)	Tag or absolute address	Specifies the high-order address (leftmost character) of the first buffer to be used for this file. The area must be reserved and cleared of previous punctuation by the programmer.
17	Second Buffer Address (BUFF2)	Tag or absolute address or omit	Specifies the high-order address of the second buffer to be used by this file. Must be omitted if a single buffer is to be used.
18	Linkage Address	Tag or absolute address of a three-character index register, or tag or absolute address of a work area, or omit	If the "locate" mode is specified for this file, this parameter must contain the low-order address (absolute or symbolic) of an index register to be used as the linkage register. If this parameter is omitted, index register 1 is assumed and assigned the tag \$X1. The tape I/O package updates the linkage register with the high-order address of the next available buffer item in

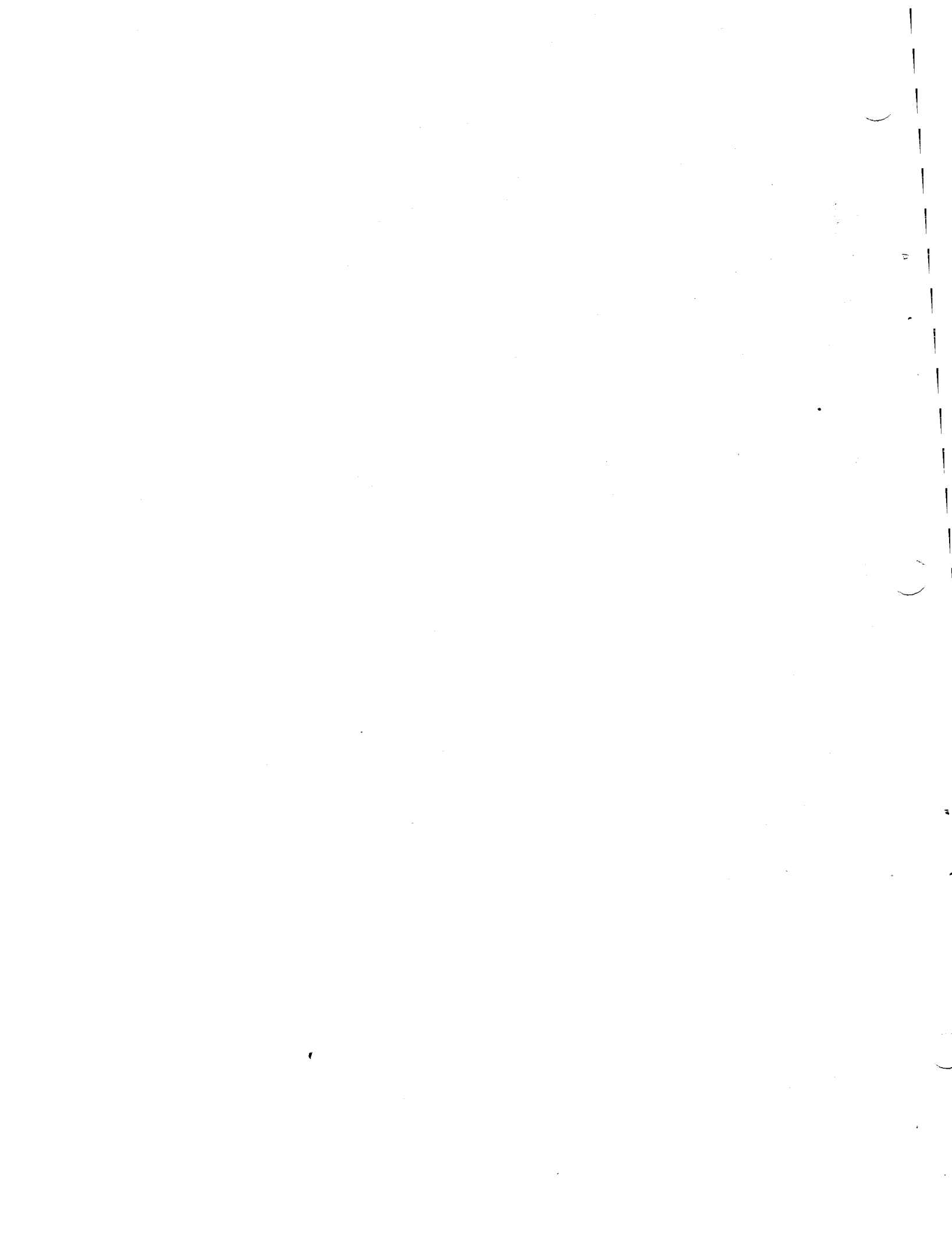


Table 2-1 (cont). Parameters Used With DCA Macro Statement

Parameter		Value	Description
Number	Name		
18 (cont)			<p>each @GET macro and with the address of the next available buffer location in each @PUT macro. NOTE: The linkage register must be punctuated by the programmer with a word mark in the leftmost position.</p> <p>If the "move" mode is specified for this file, this parameter must contain the leftmost address (absolute or symbolic) of the associated work area. The work area must be reserved and cleared of punctuation by the programmer. NOTE: This parameter must be specified for a move-mode file.</p>
19	Buffer Size	1 - 9,999 or omit	Specifies the buffer size for variable-length records. Must be blank for fixed-length records.
20	Rewind Option	REWIND or NOREWIND or omit	Indicates whether a file should be rewound when opened or closed. NOREWIND permits the use of multi-file reels. If omitted, REWIND is assumed.
21	Rewind and Lock Option	LOCK or omit	Indicates whether terminal rewind operations on this file should be done with interlock. If omitted, rewind will be without interlock. If parameter 20 is NOREWIND, this parameter is ignored.
22	Label Type	STANDARD or NON-STANDARD or omit	If standard Series 200 tape labels are to be used, STANDARD is entered. For non-standard labels, NON-STANDARD is entered. No entry for no labels. (See NOTE, page 2-2.)
23	Label Checking	CHECK or omit	<p>If standard Series 200 header and trailer labels are to be checked, CHECK is specified.</p> <p>If labels are not standard, or there are no labels, or no checking is desired, omit this parameter. (See NOTE, page 2-2.)</p>
24	Reel Sequence Number	001 - 999 or omit	If this is the first reel of a standard-label file, 001 is assumed and the user does not enter this parameter.

Table 2-1 (cont). Parameters Used With DCA Macro Statement

Parameter		Value	Description
Number	Name		
24 (cont)			If the reel sequence number is other than 001, enter the actual number. Omitted for non-standard labels, files with no labels, or if parameter 23 is omitted.
25	Label Identification	Ten characters or omit (if not omitted, ten characters are <u>required</u>)	For standard-label input files, this parameter contains the data against which characters 21-30 of the header label are to be checked. For output files, this data is written in the header labels of the newly created file. Must be omitted for non-standard or unlabelled files or unchecked files.
26	Retention Cycle	001 - 999 or omit	Must be omitted for all input files. For output files, specifies the number of days this file is to be retained after the creation date. If omitted, 000 is assumed.
27	Header Label Exit (EXIT1)	Tag or omit	The address of a user routine to be entered for header label checking. If omitted, this exit is never taken. (See "Description of Exits," page 3-4.)
28	Trailer Label Exit (EXIT2)	Tag or omit	The address of a user routine to be entered for trailer label checking. If omitted, this exit is never taken. (See "Description of Exits," page 3-4.)
29	EOF or EOR Exit (EXIT3)	Tag or omit	The address of a user routine to be entered following a normal return from EXIT2 and upon detection of end of file (input) or end of reel (output). If omitted, this exit is never taken. (See "Description of Exits," page 3-4.)

Sample DCA Macro Statements

There are two acceptable formats for writing the DCA macro statement. In Figure 2-4, a mnemonic label is written in the location field, and a line is used to enter each of the essential parameters. None of these labels are necessary; they are written for the programmer's own convenience. Figure 2-5 shows a shorter form of DCA macro. (See also the appropriate software manual for Library Processor B or C.)

which are necessary to process all the files used by the associated program. The parameters in this macro instruction pertain to all tape files in general (see Table 2-2).

Table 2-2. Parameters Used With TIOC Macro Statement

Parameter		Value	Description
Number	Name		
1	Record form	FIXED or VARIABLE or omit	<p>If all files to be processed contain fixed-length records, enter FIXED.</p> <p>If all files to be processed contain variable-length records, enter VARIABLE.</p> <p>No entry signifies that both fixed- and variable-length records may be processed by this program.</p>
2	Label size	10 - 200 or omit	<p>Indicates the length of the label buffer to be generated and cleared by the I/O package.</p> <p>If omitted, an 80-character buffer will be reserved by the I/O package.</p> <p>The tag \$LBL always refers to the leftmost character of the label buffer.</p>
3	Retention checking	CHECK or omit	<p>If this parameter is omitted, the routines for checking retention cycles will be omitted in this program.</p>
4	Read/write channels	RWC, 1, 2, 3, 4, 5, 6 or omit	<p>I/O B: If all RWC's are to be assigned by the user in his DCA macros, this parameter must be omitted. If the tape I/O package is to test for any free channels prior to performing peripheral instructions, the value RWC must be written here.</p> <p>I/O C: If this parameter is omitted, all RWC's are available to be assigned during reading or writing. The values 1, 2, 3, 4, 5, 6 specify that the RWC specified is not available to the I/O package.</p>
5 (I/O C only)	Error message option	00 - 07, 20 - 27, or omit	<p>Specifies the address assignment for the typewriter control unit if used. All error messages will be typed and there will be no halts. If omitted, halts occur as in Section IV.</p>

Column No.Data

21-26

The operands field must contain the name of the file to be opened, followed by a comma. The name must be the same as that specified in the first DCA parameter for this file.

Each @OPEN macro is replaced by two or more lines of generated symbolic coding as shown in Figure 2-6.

EASYCODER

CODING FORM

PROBLEM _____										PROGRAMMER _____										DATE _____										PAGE _____ OF _____									
CARD NUMBER		TYPE	LOCATION	OPERATION CODE		OPERANDS																																	
1	2	3	4	5	6	7	8	14	15	20	21															62	63	80											
					L		ANYLBL	@OPEN				filename,														comments													
2																																							
3																																							
4					*		ANYLBL	@OPEN				filename,														comments													
5							ANYLBL	B				open routine														I/O CALL													
6					R			NOP				filename address																											

Figure 2-6. Sample @OPEN Macro and Resulting Generated Coding

@GET

The @GET macro instruction is used to obtain an item from an input buffer and to refill the buffer from tape when it becomes depleted. If the locate mode is specified by parameter 9 of the DCA macro, each execution of a @GET statement obtains the high-order address of the next sequential item in the file and stores this address in the linkage register specified in parameter 18 of the DCA macro. In move mode, a @GET instruction transfers the next sequential item to the work area associated with the file.

The @GET macro instruction must be written in the following format:

Column No.Data

1-5	Line number, if desired.
6	L
7	Δ (Blank
8-14	(Tag)
15-20	@GET
21-26	The name of the file followed by a comma. The name must be the same as specified in the first parameter of the DCA macro for the input file.

Each @GET macro instruction generates two lines of symbolic coding as shown in Figure 2-7.

EASYCODER

CODING FORM

PROBLEM		PROGRAMMER		DATE		PAGE		OF	
CARD NUMBER	T M P R	LOCATION	OPERATION CODE	OPERANDS					
				1	2	3	4	5	6
1		L ANYLBL	@GET	filename,					comments
2									
3									
4	*	ANYLBL	@GET	filename,					comments
5		ANYLBL	B	get/put routine					I/O CALL
6	R		NOP	filename address					

Figure 2-7. Sample @GET Macro and Resulting Generated Coding

@PUT

The @PUT macro is used to place an item in an output buffer and to write the contents of the buffer on tape when the buffer becomes full. The @PUT macro instruction must be written in the following format.

<u>Column No.</u>	<u>Data</u>
1-5	Line number, if desired.
6	L
7	Δ (Blank)
8-14	(Tag)
15-20	@PUT
21-26	The name of the file, followed by a comma. The name must be the same as specified in the first parameter of the DCA macro for this output file.

Each @PUT instruction generates two lines of symbolic coding as illustrated in Figure 2-8.

@PUT (LOCATE MODE)

In the locate mode, the @PUT macro is used to obtain the address of the next available buffer location for an output item. The @PUT instruction stores this address in the linkage register specified in the DCA macro.

When operating in the locate mode, each @PUT macro must be preceded by the necessary instructions to move data to the output buffer by means of the linkage address, which is set to the beginning of the output buffer in the @OPEN macro routine. In other words, no initializing @PUT is necessary.

SECTION III
DETAILED FUNCTIONAL DESCRIPTION

BEGINNING-OF-FILE SUBROUTINE (@OPEN)

When an @OPEN macro routine is executed, the Beginning-of-File-or-Reel subroutine performs the following functions.

Input Files

Standard Labelling: The header label is read into the label area (\$LBL). If label checking is specified, the label identification and reel sequence number are checked against the information stored in the DCA area.

If either check fails when using I/O B, there is a label error halt (see page 4-1). If either check fails when using I/O C with console I/O, there is an appropriate indication to the operator (halt or console typeout if specified).

In both programs, a SENSE switch option is provided to accept the tape that is already mounted or to check again after a new tape has been mounted. If the tape is accepted (or if checking was not specified), a branch is made to user EXIT1, if specified. The user may then perform additional checking on the header label, if desired.

Non-standard Labelling: The label for the first of several non-standard labels is read into the label area (\$LBL), and a branch is made to user EXIT1. This exit should always be specified for non-standard files, as it is the only opportunity available for checking non-standard header labels.

No Labelling: An immediate branch is made to user EXIT1, if specified.

After user EXIT1, if an input tape has no data records, an immediate branch is made to the End-of-Reel or End-of-File subroutine. If there is data, the input buffer(s) and linkage are initialized and control is returned to the main program.

Output Files

Standard Header Labelling: The header label is read into the label area (\$LBL), and a branch is made to user EXIT1. Upon normal return (branch to \$RE), the retention cycle is checked if specified. The new output header label is set up in positions 1 through 40 of \$LBL (leaving the remaining positions undisturbed), and the new label is written.

NOTE: When the program is loaded, today's date may be entered as a 5-digit decimal number into \$TDY via the console or the card reader. During the execution of the EXIT1 routine, the file serial number may be entered as a 5-digit decimal number into \$SER. This is the only means of entering a file serial number.

Non-standard Labelling: A single non-standard header label or the first of several non-standard header labels is read into the label area (\$LBL), and a branch is made to user EXIT1. A normal return causes an exit from the @OPEN macro routine without writing the label.

No Labelling: The tape I/O immediately branches to user EXIT1. A normal return causes exit from the @OPEN routine without writing.

READ/WRITE SUBROUTINE (@GET AND @PUT)

The Read/Write subroutine reads and writes tape records and updates the record count in the DCA area for inclusion in standard trailer labels. This routine is entered automatically by a @GET or @PUT macro when a buffer has been depleted or filled. When GETting fixed-length records, the routine also senses for the padding character, if specified. The occurrence of one padding character in programs using I/O B, and three or more consecutive padding characters in programs using I/O C causes the routine to disregard the padded item and release the next item. The Read/Write subroutine also checks for end-of-file and end-of-reel conditions.

After each tape read operation, the Read Error routine tests for an error. If a read error exists, the routine backspaces and tries to reread the record up to 63 times for I/O B; however, for I/O C, the routine rereads both forward and backward over the record up to 63 times. If the error condition persists, a halt (I/O B or C) or console typeout (I/O C only) occurs (see Section IV). Pressing the RUN button causes this routine to test SENSE switch 1. If SENSE switch 1 is ON, the Read Error routine attempts to reread 63 times. If SENSE switch 1 is OFF, the next record is read and processing continues; the error record is bypassed.

If an error occurs in writing, the routine backspaces the record in error, erases approximately three inches of tape, and rewrites. If the error condition persists after 63 attempts to erase and rewrite the same record, the machine either halts (I/O B or C) or types out an error message (I/O C only). Pressing the RUN button (or typing in the character "G" when using console I/O) causes the Write Error routine to attempt correction 63 more times.

END-OF-REEL-OR-FILE SUBROUTINE (@FEOR AND @CLOSE)

Input Files

If a file contains banner characters, any record (after the header label) with a banner

character of "1" marks the end of the file or reel. The @FEOR macro may also be used to initiate the end-of-reel procedure. The tape I/O package backspaces the tape to position it immediately before the trailer label and then performs the labelling procedure described below.

If the file does not contain banners, a tape mark or a record whose first five characters are 1EOFΔ or 1EORΔ signals the end of the file or reel.¹ The action of the I/O program depends on the labelling option.

Standard Labelling: The trailer label is read into the label area. If checking is specified, the record count is checked. If the record count does not check, a label error halt or console typeout occurs (see page 4-1). If SENSE switch 1 is ON, the operator may depress the RUN button or type "G" to proceed to EXIT2. If SENSE switch 1 is OFF, the program must be re-run. If no checking is specified, or if the record count checks, a branch is made to user EXIT2, if specified. Upon return, or if no exit is made, the label is checked for an end-of-file trailer (1EOFΔ).

If the label is an end-of-file trailer, a branch is made to user EXIT3. If it is not an end-of-file trailer, the reel sequence number is incremented, and a tape swap is executed.

Non-standard Labelling: The label is read into the label area and a branch is made to user EXIT2, if specified. If this exit is not specified, or upon return, a tape swap is executed.

No Labelling: A branch is made to user EXIT2, if specified. If this exit is not specified, or upon return, a tape swap is executed.

Output Files (End of Reel)

The end of a reel is always detected by sensing the physical end of tape. The user may also initiate the end-of-reel routine by using the @FEOR macro. If the @FEOR macro is used, and if items are of fixed length, the last record is filled with the specified padding character, if necessary, and written. If the reel is to be terminated by a tape mark, the user must write the tape mark when EXIT2 is made.

Standard Labelling: A standard EOR trailer record is constructed in the label area. A branch is made to user EXIT2, if specified. If this exit is not specified, or upon return, the contents of the label area are written, followed by three end-of-reserved-information (1ERIA) records. The reel sequence number is incremented, and a tape swap is executed.

¹ Tape marks following a header label are automatically bypassed by the tape I/O package.

Non-standard Labelling: A standard EOR trailer record is constructed in the label area, and a branch is made to user EXIT2, if specified. If this exit is not specified, or upon return, a tape swap is executed. Control is then given to the beginning-of-reel routine.

No Labelling: A branch is immediately made to user EXIT2, if specified. If this exit is not specified, or upon return, the same procedure as for non-standard labels is followed.

Output Files (End of File)

The user signals the end of a file by issuing a @CLOSE macro. For fixed-length record files, the last record is filled with the specified padding character, if necessary, and then written. For variable-length-record files, the end-of-file routine writes the last data record without padding.

Standard Labelling: The end-of-file trailer record (1EOFΔ) is constructed in the label area, and a branch is made to user EXIT2, if specified. If this exit is not specified, or on return, the contents of the label area are written. A branch is then made to user EXIT3, if specified. If this exit is not specified, or upon return, three end-of-reserved-information records are written and the tape is rewound if specified in parameter 20 of the DCA macro. If rewinding is not specified, the tape is left positioned immediately preceding the first end-of-reserved-information (1ERIAΔ) record.

Non-standard Labelling: An end-of-file trailer record is constructed in the label area, and a branch is made to user EXIT2, if specified. If this exit is not specified, or upon return, the tape is rewound if specified in the DCA macro.

No Labelling: A branch is made to user EXIT2. Upon return, or if the exit was not specified, the tape is rewound if specified in the DCA macro, and a tape swap is executed.

DESCRIPTION OF EXITS

The following is a summary of the three optional user exits in the 1/2-Inch Tape I/O B and C package and some of their possible uses.

- EXIT1 Input: Used to perform additional checking of standard input header labels, checking of additional input header labels, or checking of non-standard input header labels.
- Output: Used for entering additional information into the standard output header label, checking output header labels, writing additional output headers, or writing non-standard output headers.
- EXIT2 Input: Used to perform additional checking of standard input trailer labels, checking of additional input trailer labels, or checking of non-standard input trailer labels.

Output: Used to enter additional information into the standard output trailer, to write labels in addition to the standard output trailer label, or to create and write non-standard output trailer labels.

For IBM-compatible tapes, this exit is used to write a tape mark preceding the trailer label.

EXIT3 Input: Used to initiate the programmer's end-of-file procedure for input files, such as @CLOSE.

NOTE: No normal return to the I/O package is possible from this exit.

Output: Used for IBM-compatible tapes to write a tape mark following the trailer label, if desired.

Following execution of the user exit routines, the user performs a normal return to the I/O package by branching to a standard location tagged \$RE. If a normal return is to be used, no action macro calls may be included in the user's routine.

SECTION IV
OPERATING PROCEDURES

PROGRAM HALTS

Table 4-1 shows the conditions which can cause 1/2-Inch Tape I/O B and C to stop a program run and the halt codes that are displayed for each such condition. The last column indicates the action to be taken by the user. The corrective actions are described in greater detail in Section III.

Table 4-1. Halt Codes and Operator Action for 1/2-Inch Tape I/O B and C

A Register	B Register ¹	Condition	Action
Buffer Address	0pp1d	Read error	If SENSE switch 1 is ON, reread the error record. If SENSE switch 1 is OFF, bypass the error record.
Buffer Address	0pp2d	Write error	Press RUN to rewrite.
\$LBL	0404d	Input Files: Label ID or reel sequence incorrect. Output Files: Work tape does not begin with a valid record.	If SS1 is ON, reopen the new reel. If SS1 is OFF, accept the reel mounted.
\$LBL + 5 (address of record count)	0406d	Reel record count incorrect on input	If SS1 is ON, accept the reel mounted. If SS1 is OFF, lock the input reel and halt.
\$LBL + 30 (address of creation date)	0405d	Retention cycle not expired	If SS1 is ON, reopen the new reel. If SS1 is OFF, accept the reel mounted.
000	0pp3d	A swap is necessary but no alternate tape unit has been specified.	After mounting the new reel, continue the run on the same drive.

¹Where "pp" is the low-order five bits of the tape control address, and "d" is a logical tape unit number. If standard Honeywell peripheral address assignments are used, pp = 00.

PROGRAM TYPEOUTS (I/O C only)

Table 4-2 shows the conditions which cause the console I/O to type a message to the operator on the console typewriter. Following each typeout, the program requires the operator to type an alphanumeric "G" in order to proceed to the option shown in the "Action" column. Note that where a SENSE switch is tested, it must be set before typing "G."

Table 4-2. Timeout Messages and Operator Action

MESSAGE ¹	CAUSE	ACTION
:pp d RDΔER□	Read Error	SENSE switch 1 ON, reopen. SENSE switch 1 OFF, bypass.
:pp d WRΔER□	Write Error	Type "G" to rewrite the record.
:pp d LBLER□	Input Files: Label ID or reel sequence incorrect. Output Files: Work Tape does not begin with a valid record.	SSW1 ON, reopen. SSW1 OFF, accept.
:pp d RETCY□	Retention cycle not expired	SSW1 ON, reopen. SSW1 OFF, accept.
:pp d RECCT□	Record count on this reel incorrect	Type "G" to accept.
:pp d ENDTP□	Tape swap necessary but no alternate tape specified.	Type "G" to continue after mounting a new reel.

¹Where "pp" is the PCU address and "d" is the drive number.

APPENDIX A
SAMPLE 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM

Figure A-1 illustrates a complete 1/2-Inch Tape I/O B and C program, showing how the various elements may be coded.

EASYCODER
CODING FORM

PROBLEM 1/2" TAPE I/O C SAMPLE PROGRAM PROGRAMMER _____ DATE _____ PAGE 1 OF 3

CARD NUMBER	TYPE	LOCATION	OPERATION CODE	OPERANDS	
				14 15 20 21	62 63 80
1			PROG	EDIT	
2			ADMODE	4	
3			ORG	36026	
4		CH1	CEQU	#1C51	RWC1 VARIANT
5		CH3	CEQU	#1C53	RWC3 VARIANT
6		PTR	CEQU	#1C02	PRINTER PCU ADDRESS
7		TPOUT	CEQU	#1C00	OUTPUT TAPE PCU
8		START	CAM	60	4-CHAR ADMODE
9			LCA	IR1,4	SET UP X1
10	L		@OPEN	INMAST, FORS2V, ORDFIL,	
11	L	ITEMS	@GET	INMAST,	GET MASTER TRANSACTION
12			BCE	OUT, MWK+6, 0	CK FOR OUT-OF-STOCK
13	L		@PUT	FORS2V,	NO, OK FOR SORT
14			B	ITEMS	NEXT ITEM
15		OUT	EXM	MWK, 0+X1, 51	DUMP OUT-OF-STOCK ONTO
16	L		@PUT	ORDFIL,	TAPE 3 AS FIXED LENGTH
17			B	ITEMS	NEXT ITEM
18			*		
19			****	EDIT ROUTINES	****
20			*		
21		OWN	PDT	\$LBL, CH3, PTR, 21	PRINT INPUT LABEL
22			PCB	*, 0, PTR, 10	WAIT
23			PDT	*, CH3, PTR, 57	SPACE TO HOF
24			B	\$RE	RETURN TO I/O
25		EOT	PDT	*, CH1, TPOUT, 02	ERASE TAPE AND WRITE
26			PDT	*, CH1, TPOUT, 02	TAPE MARK
27			PDT	*, CH1, TPOUT, 02	
28			PCB	*, 0, TPOUT, 02	
29			PDT	TPMK, CH1, TPOUT	
30			PCB	*, 0, TPOUT, 02	

Figure A-1. Sample Program Using 1/2-Inch Tape I/O B and C

EASYCODER

CODING FORM

PROBLEM 1/2" TAPE I/O C SAMPLE PROGRAM PROGRAMMER _____ DATE _____ PAGE 2 OF 3

CARD NUMBER	MARK	LOCATION	OPERATION CODE	OPERANDS	
				14 15	20 21
1			B	\$RE	RETURN TO I/O
2	L	EOJ	@CLOSE	INMAST, FORS2V, ORDFIL,	END OF RUN, CLOSE
3			LCR	IR1, 67	ALL FILES
4			ADMODE	3	
5			CAM	00	
6			B	(139)	GENERAL RETURN
7			ADMODE	4	
8	*	*****	CONSTANTS AND BUFFERS	*****	
9		IR1	DSA	0	
10		TPMK	DCW	#1C17	OCTAL FOR TAPE MARK
11	L		DCW	0	
12		RAMWK	RESV	250	WORK AREA
13		ABUFF 1	RESV	1000	
14	L		DCW	0	
15		ABUFF 2	RESV	500	
16	L		DCW	0	
17		ABUFF 3	RESV	500	
18	L		DCW	0	
19		ABUFF 4	RESV	250	
20	L		DCW	0	
21	*				
22	*	*****	I/O MACROS	*****	
23	*				
24	C		DCA	INMAST, INPUT, ODD, 1, MOVE, VARIABLE, BAN,	
25	C			250, 1, BUFF 1, MWK, 1000, REWIND, LOCK,	
26	L			NON-STANDARD, OWN, EOJ,	
27	C		DCA	FORS2V, OUTPUT, ODD, LONG,	
28	C			07, 2, MOVE, VARIABLE, 46, 250, 2,	
29	C			16, BUFF 2, BUFF 3, MWK, 500,	
30	C			22, STANDARD, CHECK,	

EASYCODER

CODING FORM

PROBLEM 1/2" TAPE I/O C SAMPLE PROGRAM PROGRAMMER _____ DATE _____ PAGE 3 OF 3

CARD NUMBER	MARK	LOCATION	OPERATION CODE	OPERANDS	
				14 15	20 21
1	L		25	STKMSTRAAA,	
2	C		DCA	ORDFIL, OUTPUT, EVEN,	
3	C		07	3,	
4	C		12	1, 250, 1, BUFF 4, 4,	
5	L		28	EOT,	
6	L		TIOC	2, 7,	
7			END	START	
8					
9					
10					

COMPUTER-GENERATED INDEX

ACCEPTABLE
RECORD FORMATS ACCEPTABLE TO 1/2-INCH TAPE I/O B AND C, 2-2

ACTION
" MACROS, 2-12
OPERATOR ACTION,
HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE I/O B AND C, 4-1
TYPEOUT MESSAGES AND OPERATOR ACTION, 4-2

BANNED RECORDS, 2-1

BASIC CONCEPTS, 1-1

BEGINNING-OF-FILE SUBROUTINE (OPEN), 3-1

CLOSE, 2-15
END-OF-REEL-OR-FILE SUBROUTINE (FEOR AND CLOSE), 3-2
" MACRO,
SAMPLE CLOSE MACRO AND RESULTING GENERATED CODING, 2-16

CODES
HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE I/O B AND C, 4-1

CODING
RESULTING GENERATED CODING,
SAMPLE CLOSE MACRO AND RESULTING GENERATED CODING, 2-16
SAMPLE FEOR MACRO AND RESULTING GENERATED CODING, 2-16
SAMPLE GET MACRO AND RESULTING GENERATED CODING, 2-14
SAMPLE OPEN MACRO AND RESULTING GENERATED CODING, 2-13
SAMPLE PUT MACRO AND RESULTING GENERATED CODING, 2-15

CONCEPTS
BASIC CONCEPTS, 1-1

CONDENSED FORM
DCA MACRO STATEMENT (CONDENSED FORM), 2-10

DCA MACRO, 2-5
" STATEMENT,
DCA MACRO STATEMENT (CONDENSED FORM), 2-10
DCA MACRO STATEMENT (FULL FORM), 2-10
PARAMETERS USED WITH DCA MACRO STATEMENT, 2-5
SAMPLE DCA MACRO STATEMENTS, 2-9

DESCRIPTION
DETAILED FUNCTIONAL DESCRIPTION, 3-1
FUNCTIONAL DESCRIPTION, 1-2
" OF EXITS, 3-4
DETAILED FUNCTIONAL DESCRIPTION, 3-1

END
OUTPUT FILES (END OF FILE), 3-4
OUTPUT FILES (END OF REEL), 3-3

END-OF-REEL-OR-FILE SUBROUTINE (FEOR AND CLOSE), 3-2

EQUIPMENT NEEDED, 1-1

EXITS
DESCRIPTION OF EXITS, 3-4

FEOR, 2-16
END-OF-REEL-OR-FILE SUBROUTINE (FEOR AND CLOSE), 3-2
" MACRO,
SAMPLE FEOR MACRO AND RESULTING GENERATED CODING, 2-16

FILE
" FORMATS, 2-1
INPUT FILES, 3-1, 3-2
" MACROS, 2-4
OUTPUT FILES, 3-1
OUTPUT FILES (END OF FILE), 3-4
OUTPUT FILES (END OF REEL), 3-3
TAPE FILE,
PREPARING TO PROCESS A TAPE FILE, 2-1

FORM
DCA MACRO STATEMENT (CONDENSED FORM), 2-10
DCA MACRO STATEMENT (FULL FORM), 2-10

FORMATS
" ACCEPTABLE,
RECORD FORMATS ACCEPTABLE TO 1/2-INCH TAPE I/O B AND C, 2-2
FILE FORMATS, 2-1

FUNCTIONAL DESCRIPTION, 1-2
DETAILED FUNCTIONAL DESCRIPTION, 3-1

GENERATED CODING
SAMPLE CLOSE MACRO AND RESULTING GENERATED CODING, 2-16
SAMPLE FEOR MACRO AND RESULTING GENERATED CODING, 2-16
SAMPLE GET MACRO AND RESULTING GENERATED CODING, 2-14
(CONT.)

GENERATED CODING (CONT.)
SAMPLE OPEN MACRO AND RESULTING GENERATED CODING, 2-13
SAMPLE PUT MACRO AND RESULTING GENERATED CODING, 2-15

GET, 2-13
" MACRO,
SAMPLE GET MACRO AND RESULTING GENERATED CODING, 2-14
READ/WRITE SUBROUTINE (GET AND PUT), 3-2

HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE I/O B AND C, 4-1

HALTS
PROGRAM HALTS, 4-1

HEADER LABEL
HEADER LABELS, 2-2
HEADER LABELS FOR NEW TAPES, 2-3
STANDARD HEADER LABEL, 2-3

I/O
PROGRAM TYPEOUTS (I/O C ONLY), 4-1
SAMPLE 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM, A-1
1/2-INCH TAPE I/O,
HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE I/O B AND C, 4-1
RECORD FORMATS ACCEPTABLE TO 1/2-INCH TAPE I/O B AND C, 2-2
SAMPLE PROGRAM USING 1/2-INCH TAPE I/O B AND C, A-1

INPUT FILES, 3-1, 3-2

INTRODUCTION, 1-1

LABEL
HEADER LABELS, 2-2
HEADER LABELS FOR NEW TAPES, 2-3
LABELS, 2-1
STANDARD HEADER LABEL, 2-3
STANDARD TRAILER LABELS, 2-4
TRAILER LABELS, 2-3

LOCATE MODE
PUT (LOCATE MODE), 2-14

MACRO
ACTION MACROS, 2-12
DCA MACRO, 2-5
FILE MACROS, 2-4
SAMPLE CLOSE MACRO AND RESULTING GENERATED CODING, 2-16
SAMPLE FEOR MACRO AND RESULTING GENERATED CODING, 2-16
SAMPLE GET MACRO AND RESULTING GENERATED CODING, 2-14
SAMPLE OPEN MACRO AND RESULTING GENERATED CODING, 2-13
SAMPLE PUT MACRO AND RESULTING GENERATED CODING, 2-15
" STATEMENT,
DCA MACRO STATEMENT (CONDENSED FORM), 2-10
DCA MACRO STATEMENT (FULL FORM), 2-10
PARAMETERS USED WITH DCA MACRO STATEMENT, 2-5
PARAMETERS USED WITH TIOC MACRO STATEMENT, 2-11
SAMPLE DCA MACRO STATEMENTS, 2-9
TIOC MACRO, 2-10

MARKS
TAPE MARKS, 2-4

MESSAGES
TYPEOUT MESSAGES AND OPERATOR ACTION, 4-2

MODE
PUT (LOCATE MODE), 2-14
PUT (MOVE MODE), 2-15

MOVE MODE
PUT (MOVE MODE), 2-15

OPEN, 2-12
BEGINNING-OF-FILE SUBROUTINE (OPEN), 3-1
" MACRO,
SAMPLE OPEN MACRO AND RESULTING GENERATED CODING, 2-13

OPERATING PROCEDURES, 4-1

OPERATOR ACTION
HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE I/O B AND C, 4-1
TYPEOUT MESSAGES AND OPERATOR ACTION, 4-2

OUTPUT FILES, 3-1
" (END OF FILE), 3-4
" (END OF REEL), 3-3

PARAMETERS
" USED WITH DCA MACRO STATEMENT, 2-5
" USED WITH TIOC MACRO STATEMENT, 2-11
PREPARING TO PROCESS A TAPE FILE, 2-1

PROCEDURES
(CONT.)

COMPUTER-GENERATED INDEX

PROCEDURES (CONT.)
 OPERATING PROCEDURES, 4-1

PROCESS
 PREPARING TO PROCESS A TAPE FILE, 2-1

PROGRAM
 " HALTS, 4-1
 SAMPLE PROGRAM USING 1/2-INCH TAPE I/O B AND C, A-1
 SOURCE PROGRAM,
 SAMPLE 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM,
 A-1
 " TYPEOUTS (I/O C ONLY), 4-1

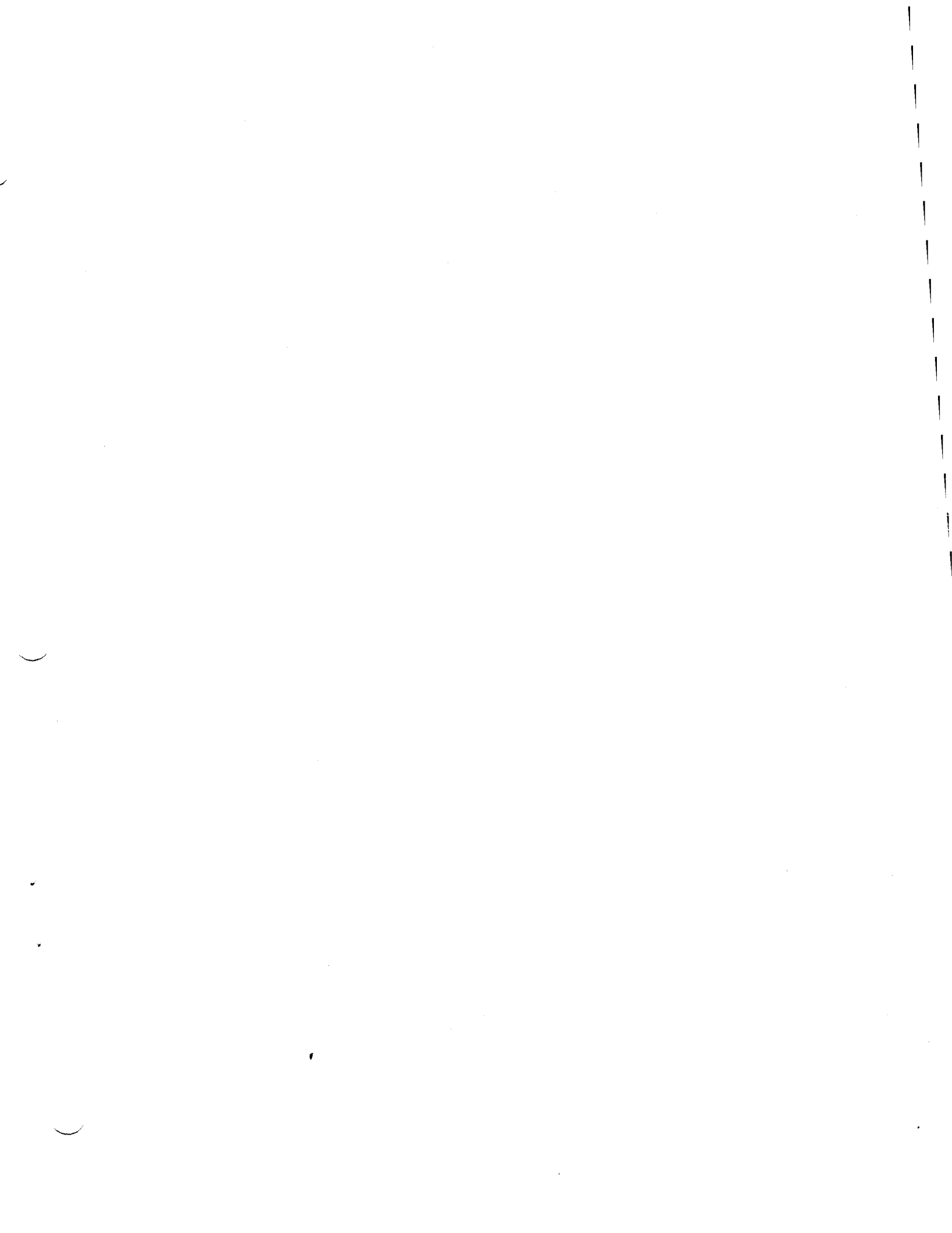
PUT, 2-14
 " (LOCATE MODE), 2-14
 " (MOVE MODE), 2-15
 " MACRO,
 SAMPLE PUT MACRO AND RESULTING GENERATED
 CODING, 2-15
 READ/WRITE SUBROUTINE (GET AND PUT), 3-2
 READ/WRITE SUBROUTINE (GET AND PUT), 3-2
 RECORD FORMATS ACCEPTABLE TO 1/2-INCH TAPE I/O B AND
 C, 2-2

RECORDS
 BANNERED RECORDS, 2-1
 UNBANNERED RECORDS, 2-1

REEL
 OUTPUT FILES (END OF REEL), 3-3

RESULTING GENERATED CODING
 SAMPLE CLOSE MACRO AND RESULTING GENERATED CODING,
 2-16
 SAMPLE FEOR MACRO AND RESULTING GENERATED CODING,
 2-16
 SAMPLE GET MACRO AND RESULTING GENERATED CODING,
 2-14
 SAMPLE OPEN MACRO AND RESULTING GENERATED CODING,
 2-13
 SAMPLE PUT MACRO AND RESULTING GENERATED CODING,
 2-15

SAMPLE
 " CLOSE MACRO,
 SAMPLE CLOSE MACRO AND RESULTING GENERATED
 CODING, 2-16
 " DCA MACRO STATEMENTS, 2-9
 " FEOR MACRO,
 SAMPLE FEOR MACRO AND RESULTING GENERATED
 CODING, 2-16
 " GET MACRO,
 SAMPLE GET MACRO AND RESULTING GENERATED
 CODING, 2-14
 " OPEN MACRO,
 SAMPLE OPEN MACRO AND RESULTING GENERATED
 CODING, 2-13
 " PROGRAM USING 1/2-INCH TAPE I/O B AND C, A-1
 " PUT MACRO,
 SAMPLE PUT MACRO AND RESULTING GENERATED
 CODING, 2-15
 " 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM, A-1
 SOURCE PROGRAM
 SAMPLE 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM, A-1
 STANDARD
 " HEADER LABEL, 2-3
 " TRAILER LABELS, 2-4
 STATEMENT
 DCA MACRO STATEMENT,
 PARAMETERS USED WITH DCA MACRO STATEMENT, 2-9
 DCA MACRO STATEMENT (CONDENSED FORM), 2-10
 DCA MACRO STATEMENT (FULL FORM), 2-10
 SAMPLE DCA MACRO STATEMENTS, 2-9
 TIOC MACRO STATEMENT,
 PARAMETERS USED WITH TIOC MACRO STATEMENT, 2-11
 SUBROUTINE
 BEGINNING-OF-FILE SUBROUTINE (OPEN), 3-1
 END-OF-REEL-OR-FILE SUBROUTINE (FEOR AND CLOSE),
 3-2
 READ/WRITE SUBROUTINE (GET AND PUT), 3-2
 TAPE
 " FILE,
 PREPARING TO PROCESS A TAPE FILE, 2-1
 " I/O,
 HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE
 I/O B AND C, 4-1
 RECORD FORMATS ACCEPTABLE TO 1/2-INCH TAPE I/O B
 AND C, 2-2
 SAMPLE PROGRAM USING 1/2-INCH TAPE I/O B AND C,
 A-1
 SAMPLE 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM,
 A-1
 " MARKS, 2-4
 TAPES
 HEADER LABELS FOR NEW TAPES, 2-3
 TIOC MACRO, 2-10
 " STATEMENT,
 PARAMETERS USED WITH TIOC MACRO STATEMENT, 2-11
 TRAILER LABELS, 2-3
 STANDARD TRAILER LABELS, 2-4
 TYPEOUT MESSAGES AND OPERATOR ACTION, 4-2
 TYPEOUTS
 PROGRAM TYPEOUTS (I/O C ONLY), 4-1
 UNBANNERED RECORDS, 2-1
 1/2-INCH TAPE I/O
 HALT CODES AND OPERATOR ACTION FOR 1/2-INCH TAPE I/O
 B AND C, 4-1
 RECORD FORMATS ACCEPTABLE TO 1/2-INCH TAPE I/O B AND
 C, 2-2
 SAMPLE PROGRAM USING 1/2-INCH TAPE I/O B AND C, A-1
 SAMPLE 1/2-INCH TAPE I/O B AND C SOURCE PROGRAM, A-1



Honeywell
ELECTRONIC DATA PROCESSING

HONEYWELL EDP TECHNICAL PUBLICATIONS
USERS' REMARKS FORM

TITLE: SERIES 200
1/2-INCH TAPE INPUT/OUTPUT B AND C
SOFTWARE MANUAL

DATED: AUGUST, 1966
FILE NO: 123.4105.020B.1-010

ERRORS NOTED:

Fold

SUGGESTIONS FOR IMPROVEMENT:

Fold

FROM: NAME _____

DATE _____

COMPANY _____

TITLE _____

ADDRESS _____

Cut Along Line

BUSINESS REPLY MAIL

No postage stamp necessary if mailed in the United States

POSTAGE WILL BE PAID BY

HONEYWELL

ELECTRONIC DATA PROCESSING DIVISION

151 NEEDHAM STREET

NEWTON HIGHLANDS, MASS. 02161

ATT'N: MARKETING INFORMATION SERVICES, MS 251

FIRST CLASS
PERMIT NO. 39531
NEWTON HIGHLANDS
MASS.



Cut Air
line

Honeywell
ELECTRONIC DATA PROCESSING