

MOD 1 (TR)

**TAPE SORT C(V) AND  
COLLATE C(V)  
ADDENDUM NO. 1**

GENERAL SYSTEM:

SERIES 200/OPERATING SYSTEM - MOD 1  
(TAPE RESIDENT)

SUBJECT:

Additions to the Tape Sort C(V) and Collate C(V)  
Manual to Include the Utilization of the Second  
Input/Output Sector for the Type 2201 Central  
Processor (Feature 1115).

SPECIAL  
INSTRUCTIONS:

The attached pages should be used to replace  
the corresponding pages in the manual. They  
will be incorporated into the next edition of the  
manual.

NOTE: The user can signal the updating of the  
manual with Addendum No. 1 by insert-  
ing this cover immediately behind the  
manual cover.

DATE: April 28, 1967

FILE NO.: 123.6005.021C.1-207

9629  
4467

Printed in U.S.A.

\*Underscoring denotes Order Number.

## FOREWORD

Tape Sort C(V) and Collate C(V) are programs which function in the Series 200/Operating System - Mod 1. Both programs are described in this manual because of the likelihood that they will be run consecutively. For instance, when two or more sorted output files have been produced, it may be desirable to collate them into a single ordered file.

Tape Sort C(V) and Collate C(V) can be stored on a binary run tape (BRT) and run under control of Tape Loader-Monitor C or Floating Tape Loader-Monitor C, or the specific program could be punched on cards for a binary run deck (BRD) and loaded via Card Loader-Monitor B or the card loading option of Floating Tape Loader-Monitor C. In any case, the sort or collate program is loaded and used independently. For more extensive information on the Loader-Monitor programs, refer to the manuals entitled Tape Loader-Monitor C, Order Number 221, Floating Tape Loader-Monitor C and Interrupt Control D, Order Number 005, or Card Loader-Monitor B, Order Number 154, as applicable.

This manual has been arranged in sections for direct reference according to the reader's familiarity with Honeywell sorting and collating programs. An explanation of the Tape Sort C(V) is contained in Sections I through V: (I) general sorting procedures and sort program characteristics, (II) parameter information for specialization of the sort, (III) descriptions of optional own-coding routines, (IV) considerations and methods for running the sort, and (V) detailed operating procedures. Collate C(V) program information is contained in Sections VI through IX: (VI) collate program characteristics and general collating procedures, (VII) parameter information for specializing Collate C(V), (VIII) considerations and methods for running the collate, and (IX) detailed operating procedures. Information concerning optional user-coded routines is contained in Appendix A.

Copyright 1967  
Honeywell Inc.  
Electronic Data Processing Division  
Wellesley Hills, Massachusetts 01281

SECTION I  
BASIC CONCEPTS OF TAPE SORT C(V)

INTRODUCTION

Sorting is an integral function of most data processing applications, in that sorting processes rearrange randomly ordered input data to produce a sequentially ordered output data file. For many applications, it is necessary to sort files consisting of variable-length units of information. Tape Sort C(V) is the general-purpose Series 200 program for performing variable-length sorting functions using 1/2-inch magnetic tape.

As a general-purpose program, Tape Sort C(V) can be adapted to different data formats and equipment configurations, enabling it to handle a wide variety of sorting applications. The information used to specialize Tape Sort C(V) for a particular application is entered at the beginning of sort processing by means of parameters specified by the user.

Input to Tape Sort C(V) is processed through the program's many operational segments which may be grouped into three logical segments: presort, merge, and last pass. The presort segment accepts the input data in the form of variable-length units of information called "items" and distributes ordered groups of items called "strings" on from two to five tape reels. The merge segment combines the presorted strings into fewer and longer strings during a series of merge phases, resulting in only one long string on each work tape except one. The last-pass segment further combines these single strings into a contiguous sequence — the sorted file.

The activities performed during the presort and last-pass segments can be augmented by routines written by the user. These routines, which can be easily written using EasyCoder symbolic language, are referred to as "own-coding." Own-coding can be used to provide all the input to be processed, or to supplement sort processing by modifying header and trailer labels and inspecting, modifying, adding or deleting items.

CHARACTERISTICS OF TAPE SORT C(V)

Tape Sort C(V) is characterized by the following activities:

1. Sorts variable-length items blocked to a variable number per record;
2. Allows a maximum record size dependent upon the main memory capacity and the number of tapes used in the merge;
3. Performs read-backward, polyphase merging using from three to six tapes;
4. Allows execution in three- or four-character address mode;

5. Sorts according to control information contained in up to ten sort key fields in each item;
6. Sorts at least one full reel of records (input or output data can be contained on any number of reels, but successful completion of the sort is not guaranteed if the input data exceeds one full reel);
7. Operates with either control panel or console typewriter (Type 220-1, -2, or -3);
8. Labels output tapes as specified by the user;
9. Provides remedial activities for handling unreadable records;
10. Provides for the inclusion of own-coding elements;
11. Provides the final output tape on a specified drive (work tape 1);
12. Provides a program history on the printer and/or typewriter, as designated; and
13. Offers the capability of immediate or delayed restarting.

#### Check and Checkpoint Terminology for Tape Sort C(V)

Throughout Sections I to V of this manual, the following terms should be interpreted as defined here. A "checkpoint" is a routine programmed to check or delay processing when Tape Sort C(V) requires an operator decision. A "check" is a term used to denote the occurrence of a checkpoint and is relayed to the operator through either the control panel (via a machine halt) or the console typewriter (via a typed message). All checkpoints associated with Tape Sort C(V) are listed in Table 5-1 and numbered to correspond to the appropriate operator action.

#### MACHINE REQUIREMENTS

Tape Sort C(V) requires the following equipment configuration.

1. A minimum main memory capacity of 8,192 character locations is required if the program is assembled and run in 3-character address mode, but up to 32K locations can be used to advantage; similarly, with 4-character address mode, a minimum of 12,288 locations is required, but up to 65K locations can be used to advantage.
2. A minimum of three tape drives (Type 204B) using 1/2-inch magnetic tape is required. Up to five additional drives can be used for any or all of the following purposes:
  - a. To provide an alternate reel for multireel input;
  - b. To increase the sort power to 3-, 4-, or 5-way merging; or
  - c. To eliminate tape changing for the merge segment.
3. Additional tape drive, or a card reader, for program loading.
4. The Advanced Programming Instructions feature (011).
5. Read/write channels two and three (RWC2 and RWC3) or their counterparts in the second input/output sector (if available) will be used for reading and writing, respectively. The second I/O sector counterparts of RWC2 and RWC3 are RWC5 and RWC6, respectively. RWC5 and RWC6 are available if the machine configuration includes Feature 1115 (second input/output sector for the Type 2201 Central Processor).
6. Index registers one through four.

SECTION II  
TAPE SORT C(V) PROGRAM SPECIALIZATION

PARAMETER DESIGNATION

The information which specializes Tape Sort C(V) for a particular application is made available to the program by means of parameters designated by the user. Parameters may be supplied to the sort program in a variety of ways, depending upon the method by which the program is being run. For instance, if Tape Sort C(V) is to be executed as a single run, the parameter characters can be punched on three parameter cards, as described in Section IV, and read in from the card reader under control of Tape Sort C(V) segment P0. Another method of parameter entry, which is convenient when Tape Sort C(V) is to be run as one of a series of programs, is for the user to include coding in the program executed immediately before the sort which will set up in memory the specific parameters to be used during sort processing and call in segment P1 of Tape Sort C(V).

If Tape Sort C(V) parameters are entered in memory by the preceding program, the need for parameter card loading is eliminated, as the parameters will be in memory when the preceding program calls the Loader-Monitor to load the sort program. If the parameters are set up by the preceding program (and are therefore essentially fixed), any minor modifications needed for a particular run can be entered manually from the control panel or console when Tape Sort C(V) is loaded.<sup>1</sup>

Parameter information is summarized in Table 4-2, page 4-6, and the parameter characters are loaded into memory starting at location 2,477 (octal). The following paragraphs describe each parameter field.

NOTE: When numeric parameters are specified, leading zeros must be designated; for example, the field specifying the maximum number of characters per item (parameters 20-23) is designated 0080 if items do not consist of more than 80 characters.

Tape Unit Specifications (Parameter Characters 1-19)

Tape Sort C(V) requires a minimum of three tape drives. Up to five additional drives may be used for multi-reel input and for work tapes to reduce the total sort time. The user specifies the address and function of each tape drive with parameter characters 1 through 16. Two

---

<sup>1</sup>Parameters may be loaded into memory by any method available to the user at any time after the Loader-Monitor is loaded, but the parameters must be in correct format in the specified memory locations before the sort program is loaded.

characters are required to address a tape drive; the first character specifies the tape control and the input/output sector to which it is connected; the second character designates a particular tape drive attached to the control. The six bits of the first character ( $V_6 V_5 V_4 V_3 V_2 V_1$ ) are used as follows:

1. Bit  $V_6$  is the I/O bit and is set by Tape Sort C(V);
2. Bits  $V_5 V_4$  specify the input/output sector (00 indicates the first I/O sector; 10 indicates the second I/O sector); and
3. Bits  $V_3 V_2 V_1$  specify the tape control.

The six bits of the second character ( $V_6 V_5 V_4 V_3 V_2 V_1$ ) are used in the following manner:

1. Bits  $V_6 V_5 V_4$  are not used and
2. Bits  $V_3 V_2 V_1$  specify the logical tape drive attached to the tape control.

Characters 1 and 2 specify the address (I/O sector and tape control designation followed by drive number) of the primary input tape. Characters 1 and 2 are blank if all of the input data are to be provided through own-coding. Characters 3 and 4 designate the address of the alternate input reel if multireel input is used. Characters 3 and 4 are blank if multireel input is not used.

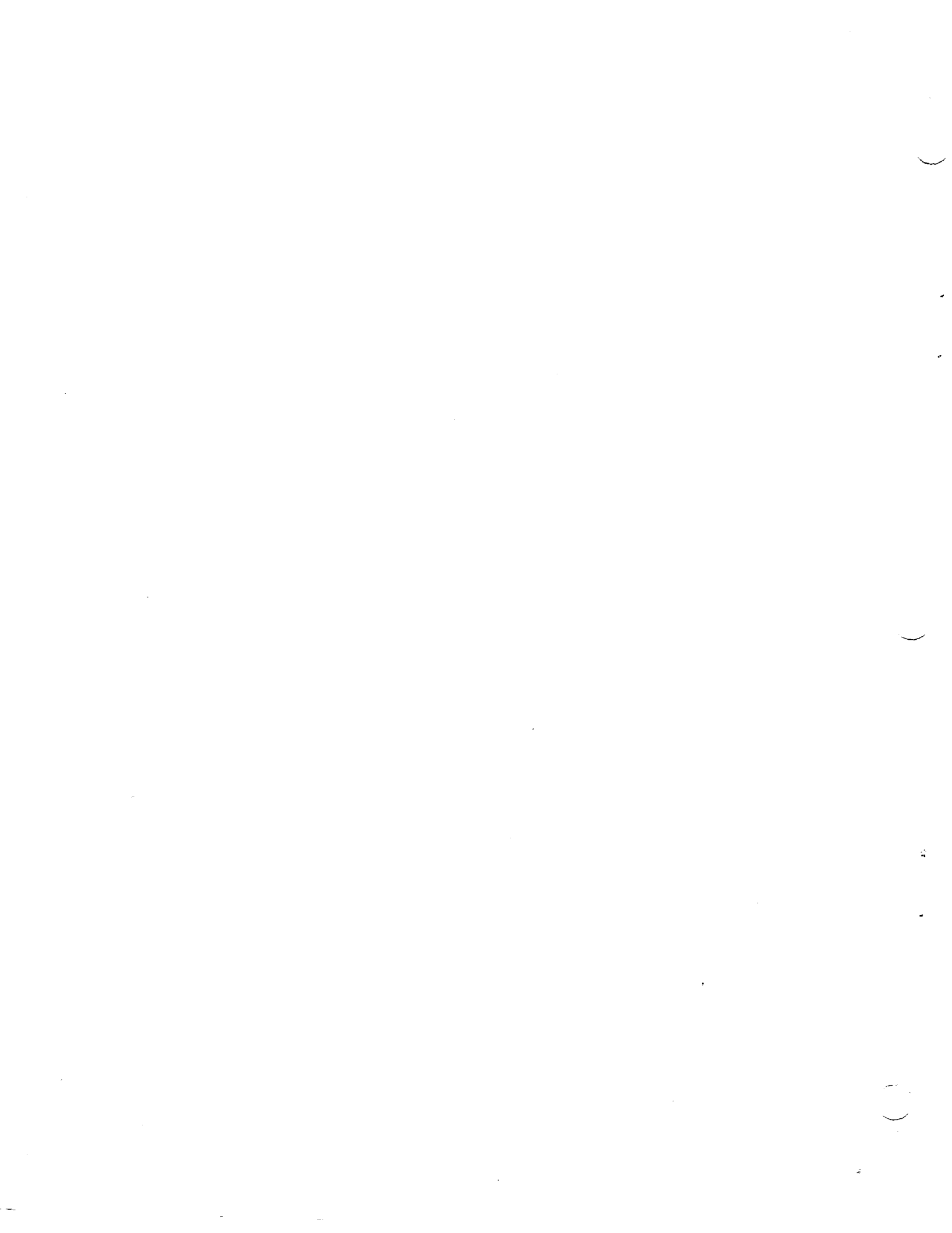
Characters 5 and 6 designate the address of the first work tape. The tape drive at this address will contain the final sorted output file. Characters 7 and 8 designate the address of the second work tape. Characters 9 and 10 specify the address of an optional third work tape; these characters are blank if work tapes 3, 4, and 5 are not used. Characters 11 and 12 specify the address of an optional fourth work tape; these characters are blank if work tapes 4 and 5 are not used. Characters 13 and 14 designate the address of an optional fifth work tape; these characters are blank if no fifth work tape is used.

Characters 15 and 16 specify the merge work tape. For a merge work tape, the user may designate the primary input tape, the alternate input tape (assuming one is used), or any other available tape not previously specified as a work tape in characters 5 through 14 (see "Effects of Parameter Characters 15 and 16 on page 2-10).

If parameter characters 15 and 16 are " $\Delta R$ ", the primary input tape is rewound and locked at the end of the presort, and the merge uses the newly mounted reel on that drive for the merge work tape. In the event of multireel input with an alternate input drive specified and " $\Delta R$ " designated, the merge work tape will be on the drive from which the next to last input reel was removed. If a tape drive which was not previously designated is specified as the merge work tape, the merge will use that portion of the tape following the read/write head, thereby preserving the data preceding the read/write head.

Characters 17 and 18 designate the total number of input reels to be read by the sort. If the number of input reels is greater than one, each input reel is rewound and locked regardless of parameter characters 15 and 16.

Character 19 specifies record blocking for bannerless files only; for all other file types, this character is blank. If bannerless files are designated (a "1" in parameter character 36 or 37), the options given in the following table are available.





If an item is to be added after all tape input has been processed, the address of the first (leftmost) character of the own-coding item must be loaded into index register 1, and a branch must be executed to the add exit address (see note on page 3-6). This cycle continues, loading index register 1 with the address of the first character of the item to be added and branching to the add exit address, until control is returned to the own-coding routine and there are no more items to be added. Own-coding then branches to the terminate exit address, and the last-pass segment terminates.

#### Multi-Reel Output

If multi-reel output occurs, each output tape is rewound when filled and must be replaced with another tape on that drive address; the user continues to have access to header-trailer and item-by-item own-coding.

#### Hash Totals

Tape Sort C(V) does not generate hash totals. Should the user wish to sort even-parity files and desire to include hash totals on the final output, methods can be adopted to preserve or generate hash totals according to the number of input reels used and output reels anticipated. For instance, if a single reel of input is to be sorted and one reel of output is anticipated, the hash total in characters 21-30 of the standard IBM (even parity) input trailer will automatically be copied onto the final output trailer. When multi-reel input is used and a single reel of output is expected, the user can apply presort header-trailer own-coding in order to accumulate the hash totals from each input trailer and use last-pass header-trailer own-coding to include the accumulated total on the final output trailer. If the user anticipates multi-reel output and desires hash totals, he must use last-pass own-coding to generate the hash totals.

#### CONSIDERATIONS FOR PROGRAMMING OWN-CODING

The starting location of presort own-coding must be above the highest address available, as given in parameters 45 through 50. The starting location of last-pass own-coding may be specified below the highest address available at a cost of restricting the memory area to be used by the merge and last-pass segments.

If header-trailer own-coding is used, the address which references a label references the first (leftmost) character of that label. When item-by-item own-coding is being used, the address which references an item references the first (leftmost) character of that item. These rules apply to addresses supplied to own-coding, as well as to addresses supplied to the sort program by own-coding.

During the presort segment, the branch to item-by-item own-coding is executed before each item is moved from the input buffer for processing by the sort program. The branch to last-pass, item-by-item own-coding is executed after the item has been processed by the last-pass segment and moved to the output buffer.

When writing own-coding instructions, the user should be especially aware of the following:

1. When a machine configuration has only one input/output sector, Tape Sort C(V) uses RWC2 for reading and RWC3 for writing. In this case, the user must specify RWC1 and/or RWC1' in his instructions — RWC2 and RWC3 must not be used for own-coding. If a machine configuration has two input/output sectors (Feature 1115), RWC2 or its second I/O sector counterpart (RWC5) or both are used for reading, while RWC3 or its second I/O sector counterpart (RWC6) or both are used for writing. In this situation, the user must insure that no read/write channel used by Tape Sort C(V) is used for own-coding.
2. Tape Sort C(V) uses index registers 1 through 4. If these registers are used during an own-coding routine, the user must store the contents of the registers to be used each time own-coding is entered and restore the original contents of these registers before re-entering the sort program.<sup>1</sup>

---

<sup>1</sup>The one exception to this rule occurs when adding an item from own-coding, as index register 1 is not restored with its original contents (see page 3-4).

CALLING AND LOADING TAPE SORT C(V) OWN-CODING

If presort and/or last-pass own-coding are used with Tape Sort C(V) on a BRT, presort own-coding can be positioned before or after any sort segment on the BRT, but last-pass own-coding must be positioned before sort segment S3. If Tape Sort C(V) is stored on cards, presort own-coding must be positioned before segment P1 or before segment P0 if used, and last-pass own-coding must be positioned before sort segment S3.

If presort own-coding is applied, the user must provide a method of calling and loading the own-coding. Presort own-coding can be called and loaded in the same manner as any other segment of a loading unit, but it must be loaded before the first sort segment is loaded.

NOTE: If the user has the memory locations available, he can load both presort and last-pass own-coding before the sort program is loaded; in this case, parameter characters 161 through 169 are blank.

ENTERING THE INPUT TO BE SORTED

Parameters 1 and 2 must designate the method used to enter the input to be sorted. The input must be in an acceptable format (see pages 1-7 and 1-8). The two possible methods for entering input are given below.

1. If parameters 1 and 2 specify the tape control address and drive number of a tape unit, the input is expected to be on magnetic tape, and the input tape must be mounted before the sort program is loaded (see page 2-2 for multi-reel input).
2. If parameters 1 and 2 are blank, all the input is to be provided by own-coding, and the user must include instructions in his own-coding routine to read the input from a designated Series 200 input device.

NOTE: When own-coding is used to enter all the input, the presort segment is initialized as having processed all of its normal tape input, and own-coding is used to add one item at a time for sort processing.

TAPE SORT C(V) PARAMETER SUMMARY

The table below lists and defines the parameters for Tape Sort C(V), gives the octal locations for the first character of each parameter field, and describes the specific parameter configurations (giving options where possible) to be entered in memory.

NOTE: In Table 4-2, "T" is used to designate a tape control and the I/O sector to which it is connected, "t" specifies a tape drive attached to the control unit, and "x," unless otherwise specified, represents a decimal digit.

## SECTION IV. TAPE SORT C(V) OPERATING CONSIDERATIONS

Table 4-2. Tape Sort C(V) Parameter Specification

Parameter Group	Parameter Number	Octal Location of First Character of Field	Description	Parameter Configuration										
Tape Unit Specifications	1-2	2477	Address of primary input tape	Tt or $\Delta\Delta^*$										
	3-4	2501	Address of alternate input tape (optional)	Tt or $\Delta\Delta$										
	5-6	2503	Address of first work tape (final output tape)	Tt										
	7-8	2505	Address of second work tape	Tt										
	9-10	2507	Address of third work tape (optional)	Tt or $\Delta\Delta$										
	11-12	2511	Address of fourth work tape (optional)	Tt or $\Delta\Delta$										
	13-14	2513	Address of fifth work tape (optional)	Tt or $\Delta\Delta$										
	15-16	2515	Address of merge work tape Tt - the address of any tape not previously specified in characters 5-14 $\Delta R$ - the primary input tape is rewound and demounted, and the newly mounted tape on that drive acts as the merge work tape	Tt or $\Delta R$										
	17-18	2517	Total number of input reels	xx										
Record Blocking	19	2521	Record blocking (bannerless files only):  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;"><u>input</u></td> <td style="text-align: center;"><u>output</u></td> </tr> <tr> <td><math>\Delta</math> - blocked</td> <td>blocked</td> </tr> <tr> <td>1 - unblocked</td> <td>unblocked</td> </tr> <tr> <td>2 - blocked</td> <td>unblocked</td> </tr> <tr> <td>3 - unblocked</td> <td>blocked</td> </tr> </table>	<u>input</u>	<u>output</u>	$\Delta$ - blocked	blocked	1 - unblocked	unblocked	2 - blocked	unblocked	3 - unblocked	blocked	$\Delta$ , 1, 2, or 3
<u>input</u>	<u>output</u>													
$\Delta$ - blocked	blocked													
1 - unblocked	unblocked													
2 - blocked	unblocked													
3 - unblocked	blocked													
Item and Record Sizes	20-23	2522	Maximum number of characters per item	xxxx										
	24-27	2526	Maximum number of characters per input record	xxxx										
	28-31	2532	Maximum number of characters per output record	xxxx										
	32	2536	Not used (must be blank)	$\Delta$										
Label Indicators	33-34	2537	Label indicators:  <table style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;"><u>Beginning of file</u></td> <td style="text-align: center;"><u>End of file</u></td> </tr> <tr> <td><math>\Delta\Delta</math> - no label</td> <td>tape mark</td> </tr> <tr> <td><math>\Delta T</math> - no label</td> <td>tape mark, trailer, and tape mark</td> </tr> <tr> <td>SS - std. label</td> <td>std. trailer</td> </tr> <tr> <td>SS - std. label and tape mark</td> <td>tape mark and tape mark</td> </tr> </table>	<u>Beginning of file</u>	<u>End of file</u>	$\Delta\Delta$ - no label	tape mark	$\Delta T$ - no label	tape mark, trailer, and tape mark	SS - std. label	std. trailer	SS - std. label and tape mark	tape mark and tape mark	$\Delta\Delta$ , $\Delta T$ , SS, N $\Delta$ , or NT (When specifying SS, the user should have one of the proper file label combinations given under "Label indicators"; the labels in the given combinations can not be interchanged.)
<u>Beginning of file</u>	<u>End of file</u>													
$\Delta\Delta$ - no label	tape mark													
$\Delta T$ - no label	tape mark, trailer, and tape mark													
SS - std. label	std. trailer													
SS - std. label and tape mark	tape mark and tape mark													

Table 5-2. Restart Entries

Character	Content	Explanation
1-2	pd -	I/O sector and tape configurations (p) and drive number (d) for work tape 2.
3	M or Δ	If the sort is stopped during the merge, "M" should be specified to prevent restarting at the beginning of the presort rather than the beginning of the current merge phase. "M" must not be entered if the sort was stopped during the presort or last pass.
4	P or Δ	"P" is designated if work tape 1 (the final output tape) was not rewound at the beginning of the sort; otherwise, this character is blank. When "P" is specified, the user must reposition work tape 1 as it was when the sort began.

When the input to the sort is such that no string is written on work tape one, the presort prepares to bypass merge processing, and only last-pass processing is required to complete sorting the file. In this case, a restart during the merge will produce unspecified results and should be avoided.

#### Tape Positioning for Restarting

Tape Sort C(V) will always resume execution at the beginning of the phase in which the halt or interruption occurred. When restarting during the presort, the restart point is at the beginning of the presort. Restarting in the merge causes execution to resume at the start of the current phase of the merge; if the print option was specified, a printout will appear when merge processing has begun (see page 5-10). A restart during the last pass resumes execution at the beginning of the last pass.

NOTE: A restart cannot be effected if the sort is stopped between segment M3 of the merge and segment S1 of the last pass.

If the sort is to be interrupted for a considerable time and all work tapes are to be demounted, the tapes must be mounted (before the restart is initiated) on the same logical drives from which they were removed. If locations 142 through 145 of the Loader-Monitor communication area were modified during the delay, such that fixed-start 3 could not be used for restarting, a console call can be entered to initiate the restart. The console call must be entered for segment AADS2VRS.<sup>1</sup> Work tape 2 must be mounted on the proper logical drive, for work tape 2 contains the information necessary for restarting.

<sup>1</sup>See footnote on page 5-6.

### POSITIONING THE INPUT TAPE

If the restart is returning to the beginning of the presort, the input tape must be mounted and positioned as it was at the beginning of the sort. Normally, positioning requires rewinding the tape; however, if the input was positioned beyond the beginning of the tape, it is necessary for the operator to reposition it to the actual starting point.

### POSITIONING OF WORK TAPE 1

If work tape 1 (the final output tape) was positioned at the beginning of tape when the sort run began, the operator should enter a blank as the fourth control character (see Table 5-2) to cause the restart to rewind the tape. If work tape 1 was not positioned at the beginning of tape when the sort began, the operator must enter "P" as the fourth control character. The "P" entry inhibits the rewinding of work tape 1 and assumes that the operator has repositioned the tape exactly as it was at the beginning of the sort.

### Search Direction for Restart Segment

The sort program normally searches forward for its next segment; but, when the restart is called using the address in Loader-Monitor fixed-start 3, the program searches backwards for the restart segment. The restart segment, once loaded, signals the sort program to return to a forward search direction. Note that searching backward is not possible when using Tape Sort C(V) in binary run card format; therefore, fixed-start 3 can only be used when Tape Sort C(V) is on a BRT.

### Restart Message

If a typewriter or printer is available and has been specified, the following message is listed at the end of the presort:

END OF PRESORT, RESTART MSG pdM<sup>1</sup>.

After this message appears, the value "M" should always be entered as the third control character when restarting in the merge (see Table 5-2).

### Restart Failure

If the restart was attempted before the sort had established any restart information, it will fail; checkpoint 29 occurs, and the sort must be rerun.

---

<sup>1</sup>Where p = peripheral control unit address and I/O sector to which PCU belongs, d = drive number, and "M" indicates that merge processing has begun.

SECTION VI  
COLLATE C(V) BASIC CONCEPTS

INTRODUCTION

Collate C(V) is a general-purpose program which functions in the SERIES 200/OPERATING SYSTEM - MOD 1. The Collate C(V) program is designed to combine from two to five previously sequenced (sorted) files consisting of variable-length items of information in identical format, thereby producing a single sequenced output file - the collated file.

The Collate C(V) program performs a single-pass routine to compare items from each input file and to write the items in sequence (according to comparisons) on 1/2-inch magnetic tape. Each input file may be contained on one or more magnetic tape reels; however, all reels containing a given file are processed sequentially from a single tape unit. If it is desired to collate more than five input files, two or more collate runs are required.

Because Collate C(V) is a general-purpose program, it can be adapted to a variety of data formats and equipment configurations. The information used to specialize the program for a particular collating application is entered at the beginning of the collate by means of parameters specified by the user. The prescribed functions to be performed by Collate C(V) can be augmented by own-coding routines written by the user. These routines, which can be easily written using the Easycoder C symbolic language, are explained on page 7-6. Own-coding can be used to modify the header and trailer labels and to inspect, modify, add, and delete items during the collate process.

CHARACTERISTICS OF COLLATE C(V)

Collate C(V) is characterized by the following activities:

1. Combines two to five previously ordered files into a single ordered file;
2. Processes variable-length items blocked a variable number per record;
3. Collates according to control information contained in up to ten collate key fields in each item;
4. Allows manual correction or removal of unreadable records;
5. Provides for collating sequence translation;
6. Allows label record changes;
7. Collates in ascending or descending sequence;
8. Provides for the inclusion of own-coding;
9. Allows execution in three- or four-character address mode; and
10. Operates with either control panel or console typewriter (Type 220-1, -2, or -3).

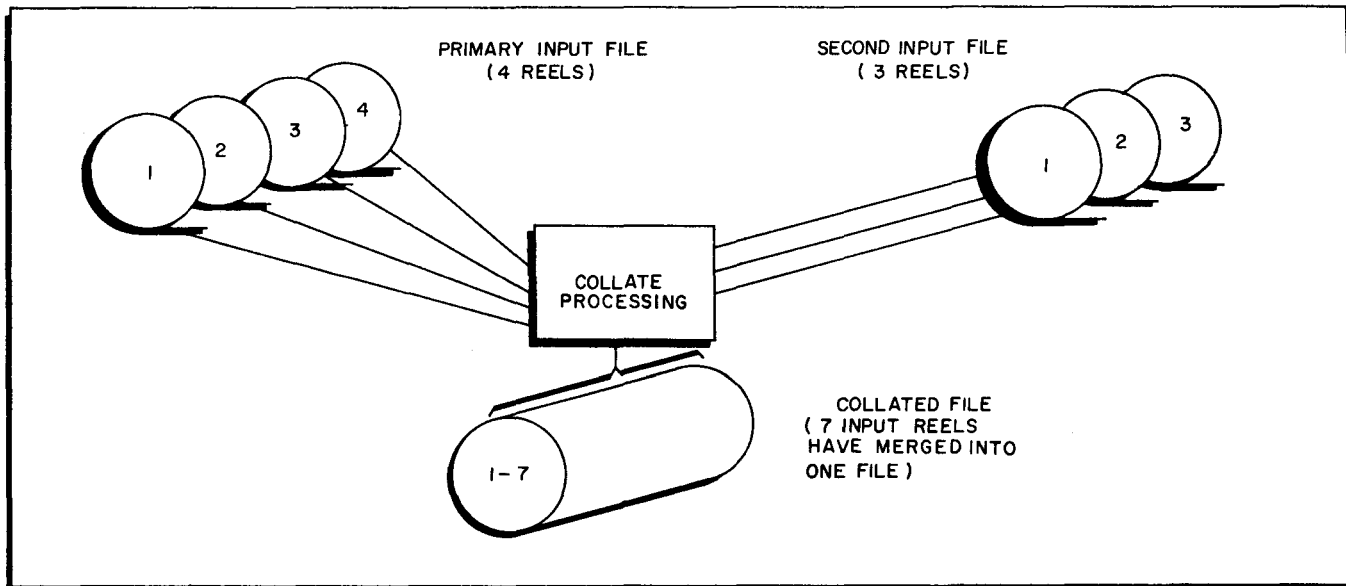


Figure 6-1. Sample Collate Process

#### Check and Checkpoint Terminology for Collate C(V)

As defined for Tape Sort C(V) (on page 5-6), a "checkpoint" in Collate C(V) is a routine programmed to check or delay processing when an operator decision is required. A "check" is a term used to denote the occurrence of a checkpoint and is relayed to the operator through either the control panel (via a machine halt) or the console typewriter (via a typed message). All checkpoints associated with Collate C(V) are listed in Table 9-1 and numbered to correspond to the appropriate operator action.

#### MACHINE REQUIREMENTS

Collate C(V) requires the following equipment configuration.

1. A minimum main memory capacity of 8,192 character locations is required if the program is assembled and run in 3-character address mode, but up to 32K locations can be used to advantage; similarly, with 4-character address mode, a minimum of 12K locations is required, but up to 65K locations can be used to advantage.
2. A minimum of three tape drives (Type 204B) using 1/2-inch magnetic tape is required. Another drive may be specified as an alternate for multireel output; and up to three additional drives may be specified to increase the number of files to be combined.
3. An additional tape drive, or a card reader, for program loading is needed.
4. The Advanced Programming Instructions feature (011) is required.
5. RWC2 and RWC3 or their counterparts in the second input/output sector (if available) will be used for reading and writing, respectively. The second I/O sector counterparts of RWC2 and RWC3 are RWC5 and RWC6, respectively. RWC5 and RWC6 are available if the machine configuration includes Feature 1115 (second input/output sector for the Type 2201 Central Processor).



6. Index registers one through four are required.

NOTE: A printer (Type 222 or 206) may be used for printouts of unreadable records, parameters, etc.

#### FILE FORMAT

Collate C(V) can process bannered or bannerless files consisting of variable-length items blocked a variable number per record. The different possible file formats require specific data item and record formats identical to those of Tape Sort C(V) as defined on page 1-7.

#### DATA REQUIREMENTS

The Collate C(V) program processes variable-length items which can be blocked a variable number per record. Note that the maximum record size designated for output records need not be the same as the maximum size specified for input records. The following formula can be used to determine the maximum record size for both input and output:

$$\text{Maximum record length} = \frac{\text{HMA} - \text{MON} - \text{PROG} - (\text{NT}) (2\text{NKC}) - \text{NKC}}{2\text{NTC}}$$

- NT — Number of input files.
- HMA — Highest memory address available to collate (parameter characters 49-54).
- TRANS — Translation requirements are 100 locations if parameter character 55 is a "T"; if character 55 is blank, no locations are required.
- MON — Monitor requirements are 1,340 locations.
- ALT — If an alternate output tape is specified, 150 locations are required; if not, the 150 locations are not needed.
- PROG — Collate program requirements are 4,200 + TRANS + ALT + ADM + TYP locations.
- NTC — Number of input files plus one.
- NKC — Number of key field characters in an item (total collate key length).
- ADM — 3-character addressing (0 locations).  
4-character addressing (1,000 locations).
- TYP — 3-character addressing (530 locations).  
4-character addressing (595 locations).

#### COLLATE KEYS AND KEY FIELDS

The files to be collated are made up of items that are identified and sequentially ordered by means of collate keys. A collate key consists of those key fields within an item which are designated as the basis for developing a sequential relationship among the items from each file to be collated. Each input item may contain up to ten key fields with up to 99 characters per field. These key fields can be located anywhere within an item, provided that they are in the same position within each item.

The location of each key field is specified by the user in parameter characters 81 through 140. If more than one key field is used, all fields must be specified in decreasing order of importance; i. e., the key field of major importance is specified first.

NOTE: If items are blocked, a character count per item occupies the first (leftmost) two (three for IBM) characters in the item; the leftmost character of the character count is considered the first character of the item - the character count field may not be used as a key field.

Collate key fields can contain any octal configuration. Standard Honeywell collating sequence is normally used in the collate, but the user may produce his output in standard IBM collating sequence by specifying the appropriate parameter.

#### MAGNETIC TAPE UNITS

Tape drives for 1/2-inch tape are used by the Collate C(V) program. In systems equipped with a recording density option, the higher density offers higher collating efficiency.

#### UNREADABLE INPUT RECORDS

In the event that an unreadable record occurs (the read-error check recurs after several attempts at rereading), the collate program will proceed according to the error option specified in the appropriate parameters.

#### OPTIONAL PRINTOUTS

If a printer is to be used and is specified in parameters, the collate program automatically prints out a list of collate parameters (if loaded by the program), the input header label (if standard), and an end-of-collate message giving a count of the items processed and items deleted.

#### COLLATE PROCESS

The Collate C(V) program compares the key fields of items from each input file and moves the item with the lowest key field configuration to an output buffer area where a variable number of collated items will be written in record form on an output tape.<sup>1</sup> The steps given below in conjunction with Figure 6-2 provide a summary of the Collate C process.

1. A record consisting of a number of items is read from each input file into an input buffer area in memory (see A of Figure 6-2).

---

<sup>1</sup>If input files are in ascending order, items will be collated according to the lowest key field configuration; input files in descending order are collated according to the highest key field configuration.

SECTION VII  
COLLATE C(V) PROGRAM SPECIALIZATION

PARAMETER PREPARATION

The information which specializes Collate C(V) for a particular application is described to the program by means of parameters designated by the user. Parameters may be entered in a variety of ways depending upon the method in which the collate program is being run. For instance, if the program is to be executed as a single run, the parameters can be punched on two parameter cards, as described on page 8-2, and read in from the card reader under control of collate segment 00. Another method of parameter entry, which is convenient when Collate C(V) is to be run as one of a series of programs, is for the user to include coding in the program executed immediately before the collate program which will set up in memory the specified parameters to be used during collate processing and call in collate segment 01.

If collate parameters are entered in memory by the preceding program, the need for parameter card loading is eliminated, as the parameters will be in memory when the preceding program calls the Loader-Monitor to load Collate C(V). If the parameters are set up by preceding program (and are therefore essentially fixed), any minor modifications needed for a particular run can be entered manually from the control panel when Collate C is loaded.<sup>1</sup>

Parameter information is summarized in Table 8-2, and the parameter characters are loaded into memory starting at location 2,477 (octal). The following paragraphs describe each parameter field.

NOTE: When numeric parameters are specified, leading zeros must be designated; for example, the field specifying the maximum number of characters per item (parameters 25 through 28) is designated 0100 if there is to be a maximum item size of 100 characters.

Tape Unit Specifications (Parameter Characters 1-24)

The Collate C(V) program requires a minimum of three tape drives. Up to four additional drives may be used for multireel output and to increase the number of input files being collated. The user specifies the address and function of each tape drive in characters 1 through 24. Two characters are required to address a tape drive: the first character specifies a tape control and the input/output sector to which it is connected; the second character designates a particular tape drive attached to the control. The six bits of the first character ( $V_6 V_5 V_4 V_3 V_2 V_1$ ) are used as follows:

---

<sup>1</sup>Parameters may be loaded by any method available to the user at any time after the Loader-Monitor is loaded, but the parameters must be in correct format in the specified memory locations before the collate program is loaded.

1. Bit  $V_6$  is the I/O bit and is set by Collate C(V);
2. Bits  $V_5V_4$  specify the input/output sector (00 indicates the first I/O sector; 10 indicates the second I/O sector); and
3. Bits  $V_3V_2V_1$  specify the tape control.

The six bits of the second character ( $V_6V_5V_4V_3V_2V_1$ ) are used in the following manner:

1. Bits  $V_6V_5V_4$  are not used; and
2. Bits  $V_3V_2V_1$  specify the logical tape drive attached to the tape control.

Characters 1 and 2 specify the address (I/O sector and tape control designation followed by drive number) of the primary input file. Note that header and trailer labels for the output file will be taken from this file. Characters 3 and 4 designate the number of reels in the primary input file. Characters 5 and 6 designate the address of the second input file. Characters 7 and 8 designate the number of reels in the second input file.

Characters 9 and 10 designate the address of an optional third input file, or they are blank if a third input file is not used. Characters 11 and 12 designate the number of reels in the third file, or they are blank if a third input file is not used. Characters 13 and 14 specify the address of an optional fourth input file; these characters are blank if only two or three files are to be collated. Characters 15 and 16 designate the number of reels in the fourth file; these characters are blank if there is no fourth file. Characters 17 and 18 specify the address of an optional fifth input file; these characters are blank if only two, three, or four input files are to be collated. Characters 19 and 20 designate the number of input reels in the fifth file; these characters are blank if there is no fifth input file.

Characters 21 and 22 designate the address of the output tape drive. Characters 23 and 24 designate the address of an alternate tape drive; if an alternate output tape is not used, these characters are blank.

#### Item and Record Sizes (Parameter Characters 25-36)

Characters 25 through 28 specify the maximum number of characters for any item. Note that items will vary in size, so that a maximum item size must be specified. The maximum number is specified in decimal with leading zeros (if needed).

Characters 29 through 32 designate the maximum number of characters per input record. Characters 33 through 36 designate the maximum number of characters per output record.

#### Record Blocking (Parameter Character 37)

Record blocking is specified for bannerless files only; for all other file types, this character is blank. If bannerless files are designated (a "1" in character 42 or 43), the options given in the table below are available to the user.

Character 37	Input Records	Output Records
Δ	Blocked	Blocked
1	Unblocked	Unblocked
2	Blocked	Blocked
3	Unblocked	Unblocked

)

0

+

(

Tape Labels (Parameter Characters 39-40)

Characters 39 and 40 specify tape labeling options as indicated in Table 7-1.

Table 7-1. Collate C(V) Label Options

Characters 39 and 40	Beginning of File	End of File
Δ Δ	No label	Tape mark
Δ T	No label	Tape mark, trailer, and tape mark
ST	Standard label	Tape mark, standard trailer, and tape mark
SS <sup>1</sup>	Standard label (with or without following tape mark)	Standard trailer (with or without tape marks)
NΔ	Nonstandard label	Tape mark
NT	Nonstandard label	Tape mark, trailer, and tape mark

<sup>1</sup>If a tape mark appears at the beginning of file, one must precede the trailer at the end of a file; if a tape mark precedes a trailer, the final output will always have a tape mark following the trailer.

The output file will have the same type of labeling as the primary input file, unless the user formulates a change through own-coding. If standard labels and trailers are specified, the output tape header label will be identical to the header label of the primary input tape, except that the serial number of the output tape will be retained. One exception occurs in that the Collate C(V) program references the 5-character data field as the Loader-Monitor communication area (octal locations 216 through 222) and writes the data field configuration in the output header label. The data is written in character positions 31 through 35 of an 80-character label. Note also that Collate C(V) inserts a reel sequence number in character positions 17 through 19.

The output trailer will be identical to the trailer of the last reel of the primary input file, except that record and item counts for the combined file will be established by the collate program. If nonstandard labels are specified, the header label of the primary input file and the trailer record, if any, on the last reel of this file will be copied onto the final output reel.

Output Label Verification (Parameter Character 41)

Character 41 is a "1" if multireel output is expected and it is desired to have the collate program verify the date and retention cycle on all final output reels except the first. (It is the user's responsibility to assure that the first final output tape can be written on.)

For each output reel after the first, the tape is read forward, so that one record is read into memory. If the first five characters of the record are "1HΔ", the date and retention cycle are compared against those prescribed by the Loader-Monitor, and a check occurs if the label is in error (checkpoint 24).

Character 41 is blank if the output label check is not desired.

Banner Characters (Parameter Characters 42-43)

Characters 42 and 43 are used to specify the presence or absence of banner characters in input and output records. Character 37 is a "1" if banner characters are not expected on the input. If character 37 is blank, the collate program checks input records for a banner character, regardless of its binary configuration. If character 37 is any nonblank character other than "1," the collate program will select from the input files only those records that have the specified banner character.

The option to select on a specific banner character allows the user to process certain types of records from the input file while bypassing all records without the designated banner. For example, if the user's input file consists of records of information from Washington, Paris, London, Los Angeles, etc., (the records of the file should have specific banner characters, e.g., "41g" for Washington, "42g" for Paris, etc.) and he wished to collate only those items with information from Washington, he could specify a "J" (equivalent to a "41g") in character 42, and the collate program would accept only the input records with banner characters of "41g" and collate the items contained therein.

NOTE: It is recommended that the special banner character designated be in the range 41 through 77 (octal).

Character 43 is a "1" if output records are not to have banner characters or a blank if the first character in each output record is to be a banner character (56 octal). As with input records, if an alphanumeric character other than "1" is specified in character 43, that specific character is written as a banner character in each output record.

The possible configurations for characters 42 and 43 are summarized in the following table, where "x" represents any desired alphanumeric character (1 excluded).

Character 42	Character 43	Input Data Record	Output Data Record
x or Δ	Δ or x	Banner character	Banner character
x or Δ	1	Banner character	No banner character
1	Δ or x	No banner character	Banner character
1	1	No banner character	No banner character

NOTE: A relationship must prevail between parameters 42 and 44 and between parameters 43 and 45, as bannered files are read and written in odd parity and bannerless files are read and written in even parity.

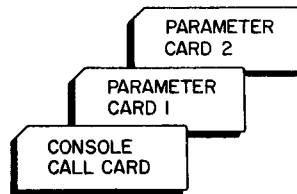
Parity and Record Gap (Parameter Characters 44-45)

Character 44 is blank for odd parity input or specifies an "E" for even parity input.

Character 45 is blank if the output is to be written with odd parity and short interrecord gaps,



a three-character mode Loader-Monitor should not be used to load a version of Collate C(V) in four-character mode that includes typewriter options.



1. The Console Call card. The Console Call card is punched as follows:
  - Columns 1-6 contain the characters AAEC2V (or AAEC2F if four-character) to indicate the program name;
  - Columns 7-8 contain the characters 00 to indicate the segment name;
  - Column 9 contains the character 0 to indicate the address of the program run tape containing the Collate program;
  - Columns 10-17 contain either blanks to specify program execution after program loading or a specific program and segment name to designate a program halt after loading the specified segment;
  - Column 18 contains an asterisk (\*) to identify the Console Call card.
2. Parameter Card 1 (parameters 1-80 in Table 8-2 on page 8-4).
3. Parameter Card 2 (parameters 81-160 in Table 8-2 on page 8-4).

Figure 8-1. Parameter Card Input for Loading Collate C(V) Parameters

#### Programmed Call for Collate C(V) Program on Cards

If Collate C(V) is loaded within a series of programs, it is recommended that the preceding program enter the collate parameters in their proper memory locations (the locations are specified in Table 8-2 on page 8-4). The program which precedes Collate C(V) is used to initiate a call by entering the Collate C(V) program name (AAEC2V) and first segment name (01) in the appropriate fields of the Loader-Monitor communication area. The program run deck should be placed in the card reader in the order shown in Figure 9-2 on page 9-3.

The Collate C(V) program deck is made up of segments, as indicated on page 8-1, each of which is identified by a segment header card, a card with an X8 (Q) or an X84 (\*) punch in column 1. After the preceding program has entered the parameters in memory and initiated a call for Collate C(V), the Loader-Monitor searches forward, reads the 01 segment header card (with the collate program name in columns 11 through 16 and the segment name in columns 17 and 18), and begins loading the Collate C(V) program.

Card Call for Collate C(V) Program and Parameters on Cards

When Collate C(V) is on cards in BRT format and is to be run as the initial program in a series or as a single program, the collate parameters are punched on parameter cards and placed in the program deck between segments 00 and 01 (the parameter cards are placed after the end-of-segment card for 00, which contains an X4 "M" punch in column 1). When typewriter segment T0 is used, the parameter cards should be placed after the end-of-segment card for T0. The program run deck should be placed in the card reader in the order shown in Figure 9-1. Once the Loader-Monitor has been loaded, it will search forward for a loading unit, read the 00 segment header card, and load the collate program.

CALLING AND LOADING OWN-CODING

The user must provide a method of loading Collate C(V) own-coding. Own-coding can be called and loaded in the same manner as any other segment of a loading unit, but it must be loaded before the first collate segment is loaded.

COLLATE C(V) PARAMETER SUMMARY

Table 8-2 lists and defines the parameters for Collate C(V), gives the octal locations for the first character of each parameter field, and describes the specific parameter configurations (giving options where possible) to be entered into memory.

NOTE: In Table 8-2, "T" is used to designate a tape control unit and the I/O sector to which it is connected; "t" specifies a tape drive attached to the control unit; and "x," unless otherwise specified, represents a decimal digit.

Table 8-2. Collate C(V) Parameter Specification

Parameter Group	Parameter Number	Octal Location of First Character of Field	Description	Parameter Configuration
Tape Unit Specifications	1-2	2477	Address of primary input file	Tt
	3-4	2501	Number of reels in primary file	xx
	5-6	2503	Address of second input file	Tt
	7-8	2505	Number of reels in second input file	xx
	9-10	2507	Address of third input file (optional)	Tt or blank
	11-12	2511	Number of reels in the third input file (optional)	xx or blank
	13-14	2513	Address of fourth input file (optional)	Tt or blank
	15-16	2511	Number of reels in fourth input file (optional)	xx or blank

TECHNICAL PUBLICATIONS REMARKS FORM

TITLE: MOD 1 (TR) TAPE SORT C(V) AND  
COLLATE C(V) ADDENDUM NO. 1  
SOFTWARE BULLETIN

DATED: APRIL, 1967

FILE NO: 123.6005.021C.1-207

ERRORS NOTED IN PUBLICATION:

Fold

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION:

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  
60 WALNUT STREET  
WELLESLEY HILLS, MASS. 02181

ATT'N: TECHNICAL COMMUNICATIONS DEPARTMENT

FIRST CLASS  
PERMIT NO. 39531  
WELLESLEY HILLS  
MASS.



Cut Along

**Honeywell**  
ELECTRONIC DATA PROCESSING