

TRS-80 BASIC PROGRAM UTILITY
DISK OPERATING SYSTEM SORT MERGE
"DOSORT"
USERS MANUAL

Written for RACET computes by

T.S. Johnston
and
R.D. Johnston

For Use On The Radio Shack® TRS-80™
32K-48K Disk Operating System (Two to Four Drives)

© COPYRIGHT 1979, RACET computes, Orange, California

IMPORTANT NOTICE

All RACET computes programs are distributed on an "AS IS" basis without warranty. Neither RACET computes nor the contributor makes any express or implied warranty of any kind with regard to this program material, including, but not limited to the implied warranties of merchantability and fitness for a particular purpose. Neither RACET computes nor the contributor shall be liable for incidental or consequential damages in connection with or arising out of the furnishing, use or performance of this program material.

© 1979, RACET computes, Orange, California

The Government Law (Title 17 United States Code) has been amended by a recent Act of Congress, Public Law 92-140, protecting certain sound recordings against unauthorized duplication. It is an infringement of this law to copy any properly registered cassette designated with the copyright notice (e.g. © 1979, RACET computes, Orange, California).

TRS-80 BASIC PROGRAM UTILITY DISK OPERATING SYSTEM SORT MERGE "DOSORT"

INTRODUCTION

DOSORT is a system operating under the TRS-80 Disk Operating System (DOS) for sorting and merging files. This generalized system has the following key attributes;

- Data files that can be read and written by standard TRS-80 input/output commands can be processed by DOSORT.
- DOSORT will sort input files according to user specified sequence utilizing multiple ascending or descending sort keys.
- Two modes of sort variable specification may be used - multi-variable mode and character string mode. The multi-variable mode allows handling of mixed singly-dimensioned string, floating point (single- or double-precision), and integer arrays. The character string mode sorts on multiple fields within a record.
- Data files may be spread across several diskettes.
- Presorted files may be included in a given sort/merge application for merging.
- User exits are provided during input and output processing allowing specialized processing.
- Specification of sort parameters has been designed to be easy - even for the inexperienced user.
- Multi-volume (diskette) files may be sorted or merged on a minimum 32K two-drive system. System performance is improved on a 48K system with a three- or four-drive configuration.

DOSORT utilizes very efficient machine language routines for the critical tasks of sorting and comparing. GSF (Generalized Subroutine Facility) is used as an interface to these subroutines. The resulting system is both a versatile and effective sort/merge package for the TRS-80 system.

DESIGN CHARACTERISTICS

Data sorting remains one of the primary tasks performed by computers today. Many sorting programs have been implemented for almost every computer system marketed. These systems have ranged from the very simple tape sort to the very elaborate multi-purpose generalized sort merge programs.

Through the history of computing, no single accepted method of sorting has been devised. The basic reason for this is that large amounts of data must be manipulated efficiently. The efficiency of sorting is very sensitive to the hardware architecture, system configuration, and the application involved. Each computer system must be evaluated independently to select the most appropriate sorting technique and method of implementation.

This is further complicated by the requirement of a single

general purpose system to cover a wide range of application and hardware configurations. From the previous discussion it can be seen that these two requirements are, in many respects, mutually exclusive. Selection of the appropriate sorting method and implementation must be done with a careful analysis of the tradeoffs involved.

The primary objectives and design constraints considered for the development of DOSORT for the TRS-80 system included the following:

- Data to be sorted may be distributed on several volumes.
- Sorting must be accomplished with a minimum disk configuration of two disk drives - but must be able to take advantage of three or four disk drives, if available.
- Sort data input and output must be compatible with existing TRS-80 BASIC file structure.
- User input and output processing of records must be provided (user exits).
- Specification of the data to be sorted and the sort keys to be used must be easy - even for the inexperienced user.

The first two characteristics impose significant restrictions on the sort system. Diskette switching is required for large sorts, especially for the minimum configuration system.

The third characteristic of file compatibility results in a decreased performance of DOSORT. A more efficient system could be implemented by restricting the types of data to be sorted and by incorporating special purpose input/output routines.

The final two characteristics imply than an interface to BASIC must be provided. This is accomplished by implementing the control program directly in BASIC, while utilizing machine language routines for the time-critical tasks of sorting and data comparison.

SYSTEM CONFIGURATION CONSIDERATIONS

Any size sort may be performed on a minimum two drive TRS-80 system. Performance will be improved for three and four drive systems.

Drive #0 is used to store the Disk Operating System, DOSORT routines, GSF load module, and checkpoint tables. This drive is, therefore, not available for user input, output, or DOSORT work files. For a minimum two disk system, all user and work files will be mounted on Drive #1.

In most cases, the user will be required to switch volumes at various stages during the sort/merge process. Data input buffering is performed in an attempt to minimize the switching necessary. It is not recommended, however, that the user attempt to perform very large sorts using a minimum system configuration because of the excessive volume switching required.

The user can usually avoid large sorts by merging a large sorted file with a small file to be sorted by DOSORT. Provisions have been made to allow the user to specify files which are already in the correct sort sequence (User Merge Files). The User Merge Files can then be merged with sorted input files (normally smaller) at the appropriate stage during the sort/

merge.

The user will be requested to supply work volumes during the course of the sort/merge. It is recommended that:

- The work volumes contain the maximum amount of space available (i.e., empty volumes - formatted, non-DOS diskettes).
- A scheme is maintained to identify disk volumes. This is required since several volume switches may occur during the sort. Inserting an incorrect diskette at any time will cause DISASTEROUS results!

DOSORT requires a minimum 32K TRS-80 system. Performance will be greatly improved with 48K systems. The number of files open will also be reduced in 48K systems which provides additional performance improvement.

SYSTEM DEFINITIONS

Several terms are used throughout this document that have specific definitions in relationship to DOSORT. These are summarized below:

Term	Definition
VOLUME	The data contained on a single diskette (one side only). The volume name is an arbitrary identification assigned by the user. It is <u>not</u> necessarily identical to the name assigned when formatting a disk.
DRIVE	The physical disk drive attached to the TRS-80 system. Disk drives are specified by their standard TRS-80 numbers (0,1,2, and 3).
DATASET	Data contained on a single volume under one TRS-80 file name. The dataset name is specified directly by the TRS-80 file name maintained in the volume directory. A dataset cannot be larger than a volume.
FILE	The data contained in one or more datasets considered to be a single stream of input or output. The datasets may be located on one or more volumes. A file is specified by listing the dataset name and volume for each component of the file. The order in which they are specified defines the sequence in which they will be processed.
SORT VARIABLES	Data to be sorted is read into one or more singly dimensioned arrays specified by the user. The sort variable name is the array name, and must conform to standard TRS-80 BASIC naming conventions.
SORT RECORD	The collection of data consisting of corresponding elements of all sort variables carried as a single entity during sorting or merging.

SORT KEY

The sort variables within a sort record to be used during comparison operations while sorting or merging. Sort keys can be specified to be in ascending or descending mode. DOSORT utilizes variable names starting with T-Z. The user may utilize variable names starting with A-S.

SYSTEM FILES

A number of different files are created or used by DOSORT. These are summarized below:

File Name	Description
SORT INPUT	This file contains the data to be sorted by DOSORT. The data can be in any format that can be read using BASIC statements.
USER MERGE	A file which is already in the correct sort sequence which is to be merged with other files to produce a final sorted output file.
SORT BLOCK	This is an intermediate file produced by DOSORT containing one or more sorted datasets on a single volume. Dataset names SRTBLKA, SRTBLKB, ... etc. are used to identify the datasets.
SYSTEM MERGE	This file is either an intermediate file or the final file produced by merging sorted blocks and/or user/system merge files.
SORTPARM	A file maintained by DOSORT to save the status of all files, disk assignments, work volumes, and sort parameters required. This file is maintained on the operating system disk (Drive 0).
WORK	A volume that can be used by DOSORT to hold intermediate or final output files. The user may specify the names of the work volumes, or use the defaults (WORKA, WORKB, ... etc.) provided by DOSORT.

Each file is assigned a number by DOSORT during processing. This file number is displayed in the summary phase of the sort sequence.

SORT KEY SPECIFICATION

The GSF sort subroutines are utilized by DOSORT for the internal sorts and data comparisons. The user should refer to the GSF documentation for GSF #17 & 18 for details on the form of sort keys.

Two modes of sorting are provided by GSF and DOSORT. DOSORT solicits, during the assignment phase (SORTPA), the mode to be used. This is followed by a request for the appropriate sort variables and keys to be used during the sort/merge application. In addition, certain statistics are requested regarding the number, type, and character string sizes. This information is used to determine memory allocation parameters.

STRUCTURE OF DOSORT

DOSORT consists of five separate phases. Not all phases are used during a given sort/merge application. Each phase is implemented as a BASIC program as summarized below and on the facing figure.

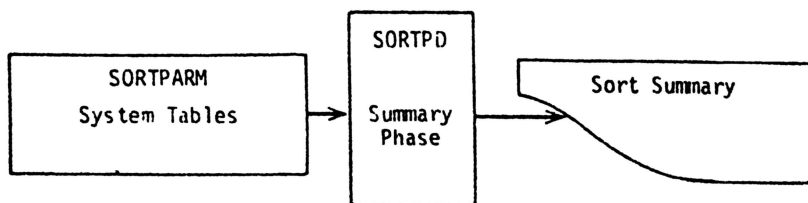
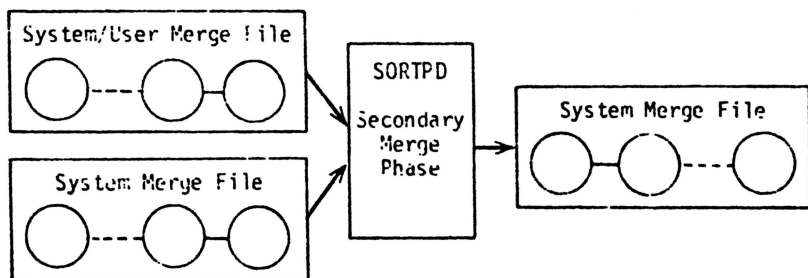
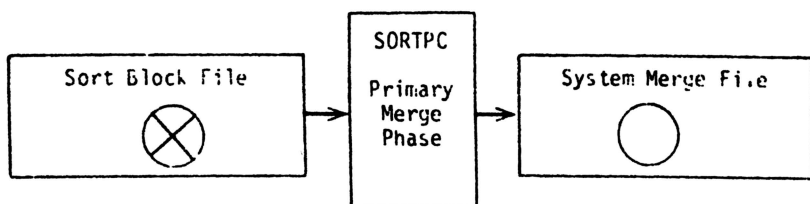
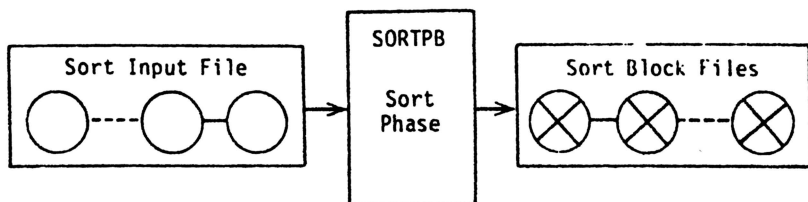
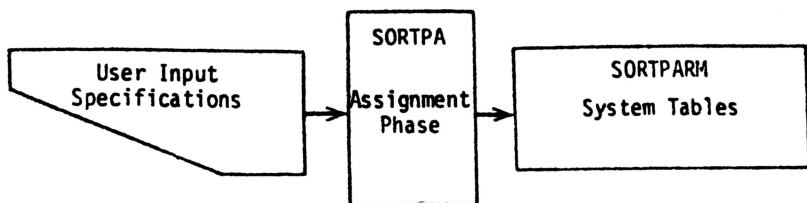
Phase	Description
SORTPA	<u>Assignment Phase</u> - This phase solicits user input/output and sort specification data, builds tables, and passes control to the phases required. The file "SORTPARM" is created during this step and <u>written to Drive 0</u> . This file is used by all other phases for receiving sorting instructions, creating checkpoints of sort status, and passing control information to succeeding phases.
SORTPB	<u>Sort Phase</u> - This phase is responsible for reading the user sort input file and performing the initial sort of the data. Memory is filled, sorted, and then written to sort block files. Each volume may contain more than one sort block file, but no sort block file is split between two volumes.
SORTPC	<u>Primary Merge Phase</u> - This phase merges the sort block files on one volume to produce a single system merge file. If the work volumes provided by the user are empty then the merge of sort block files will produce a system merge file contained on one volume.
SORTPD	<u>Secondary Merge Phase</u> - This phase performs two-way merges of system or user merge files. The result is a larger system merge file. When only one merge file remains, the sort/merge is complete.
SORTPE	<u>Summary Phase</u> - This phase is executed upon completion of the sort or if an error is encountered during execution of the previous phases. Error messages and sort status is provided. Checkpoints taken at various stages during the sort/merge process, and the information provided by this phase allows the user to restart the sort at intermediate points under many circumstances. This phase also lists the dataset names, volumes, and sizes of the final sorted datasets.

USER DATA SPECIFICATION

Data input/output is performed by standard TRS-80 BASIC instructions. This provides compatibility between other BASIC user programs and DOSORT. In order to use this facility the user must insert a read and write statement in SORTPB, SORTPC, and SORTPD program segments. In addition, a DIM statement describing the sort variables must also be included. This is best illustrated by two examples as follows:

Example #1 - Consider a multiple variable mode sort file with each sort record containing the following data:

Variable	Contents	Variable Type
NAME	Name	Character String
IC	Age	Integer
PB	Weight	Single Precision
SEX	Sex	Character String



In the multi-volume sort mode, the variables IS, IG, PB, and SX\$ are treated as singly dimensioned arrays. The following DIM statement must be inserted at line #100 in SORTPB, SORTPC, and SORTPD:

```
100 DIM NM$(YN),IG(YN),PB(YN),SX$(YN)
```

where YN is a variable set by DOSORT. The user must not specify variable names used by DOSORT. All DOSORT variable names begin with T,U,V,W,X,Y, or Z.

The user must also insert the appropriate input/output BASIC statements at lines 1000 and 1500 as illustrated below:

```
1000 INPUT#YL, NM$(YM),SX$(YM),PB(YM),IG(YM)
```

```
1500 PRINT#YL, CHR$(34);NM$(YM);CHR$(34);SX$;",";PB(YM),IG(YM)
```

The above example assumes that NM\$ contains embedded commas which require the string to be enclosed in quotes (CHR\$(34)). It is assumed that SX\$ only needs a trailing comma. Refer to the TRS-80 DOS Users Manual for additional information on reading and writing character strings.

Example #2 - Consider a character string mode user input sort file. Each sort consists of a single character string SS. The following DIM statement must be placed at line #100 in SORTPB, SORTPC, and SORTPD.

```
100 DIM SS(YN)
```

where YN is a variable set by DOSORT. The user must also write appropriate input/output BASIC statements at line 1000 and 1500 as illustrated below:

```
1000 INPUT#YL, SS(YM)
```

```
1500 PRINT#YL, SS(YM)
```

This example assumes there are no embedded commas which would otherwise require the string to be enclosed in quotes.

Multi-volume files are allowed in DOSORT. The user must ensure that each dataset to be read by DOSORT contains an integral number of records, i.e., no record is split between two datasets. DOSORT automatically creates such files, performing volume switching when required.

Two special restrictions MUST BE OBSERVED with sequential files:

- No single PRINT# statement is allowed to produce a line longer than 255 bytes. Two or more PRINT# statements may be included at line 1500 if large sort records are to be written.
- No character string is allowed to have embedded carriage return characters, e.g. CHR\$(13).

An ON ERROR routine can be included in a user program to switch input volumes when reading multi-volume files produced by DOSORT. Special steps, however, must be taken in switching volumes on a "DISK FULL" error condition. Appendix A summarizes the techniques required for both of the above conditions.

All user sort input is performed by SORTPB at line #1000. The user may insert special selection code or variable redefinition

steps prior to passing control back to DOSORT. This allows sorting only the data required providing increased flexibility and performance. The user may test variable "TY" in SORTPD to determine if the data being read is a user merge file (TY=2).

The user may terminate reading of input data as outlined in Appendix E. To terminate output processing the user need only close the current output file (CLOSE YL) and exit.

Final user sort output may be produced in either of the three sort phases, SORTPB, SORTPC, or SORTPD. The user may test variable "TZ" prior to writing records at line #1500 to determine if the record being written is a final output record (TZ=2). Program steps could then be executed to perform tasks such as elimination of duplicate records or production of summary totals. It is not possible to predict which phase will produce the final output, thus requiring identical code in all three phases.

SPECIAL NOTE

If the user inserts additional code ANYWHERE within SORTPC of SORTPD it will be ABSOLUTELY NECESSARY to do the following:

- Place all new DIM statements required between lines #101 - 198 of SORTPC and SORTPD.
- Insert dummy initialization statements for all non-dimensioned variables between lines #102 - 198 of SORTPC and SORTPD. For example, if the variable "I" was added in user code, the statement "I=0" must be included somewhere between lines 101 and 198.

SPECIAL NOTE

SPECIAL NOTE

DOSORT WILL NOT FUNCTION CORRECTLY IF THE ABOVE ACTION IS NOT TAKEN

The user may not modify or create CHARACTER STRINGS in the user input/output subroutines except under the following conditions:

1. Any "temporary" character string must be preceded by a:

GOSUB 4900

and followed by a :

GOSUB 4910

Such character strings should only be assumed to be valid only until another GOSUB 4900 is issued or control is passed back to DOSORT.

2. The temporary strings used must be 256 bytes or smaller.
3. Permanent character strings can be used only if pre-initialized (lines #101 - 198) and changed only with the MID\$ function on the left side of the assignment statement. The statement containing the MID\$ function must also be enclosed by a "GOSUB 4900" and "GOSUB 4910".

An example of the use of special character string handling follows:

Objective: Each user input record consists of a single character string. However, only the first 10 characters are to be passed to DOSORT in variable S\$.

User Code:

```
1000 GOSUB 4900
1010 INPUT#YL,R$           :REM R$ is a temporary variable
1020 R$=LEFT$(R$,10)       :REM This modifies a variable
1030 GOSUB 4910
1040 S$(YM)=R$
```

Objective: A variable (D\$) representing the largest character string found in the input stream is to be saved (for some unspecified purpose).

```
101  A$="#####"
1000 INPUT #YL,B$(YM), ...
1010 IF B$(YM) >=A$ THEN 1050
1020 GOSUB 4900
1030 MID$(A$,1) = B$(YM)
1040 GOSUB 4910
1050  :
      :
```

PROGRAM OPERATION

DOSORT consists of the five BASIC programs SORTPA, SORTPB, SORTPC, SORTPD, and SORTPE as described earlier. The system expects these programs to be located on Drive 0 along with the Disk Operating System. The content of the DOSORT distribution tape is described in Appendix B along with instructions for transferring the programs to disk. GSF is required for operation of DOSORT. The instructions given in the GSF Users Manual should be followed for loading and initializing GSF.

To perform a sort/merge task the user should follow the following general steps:

1. Load GSF as described in the GSF Users Manual. The memory size should be set as directed. The number of files should be at least four for a 48K system and eight for a 32K system
2. Load the five BASIC programs from the distribution tape as described in Appendix B if they are not already loaded.
3. The user should verify that at least four granuals of free space exist on the operating system disk (Drive 0) for the SORTPARM data set.
4. The HIM, INPUT, and PRINT commands, along with any other user code required for the specific application should be inserted as described in the User Data Specification Section. The modified SORTPB, SORTPC, and SORTPD should then replace the corresponding programs on Drive 0 (Modify, then SAVE the modified SORTPB, SORTPC, and SORTPD programs).
5. SORTPA should then be run to initiate DOSORT as follows:
 RUN "SORTPA"
6. The user should enter the appropriate sort specification information when requested. A discussion of the data items required is given later in this section.
7. DOSORT will then perform the steps necessary to complete the sort/merge request. The user may be directed to change diskettes (volumes) at the appropriate steps in

this process.

8. DOSORT enters SORTPE upon completion of the sort/merge or upon detection of an error or user intervention request. SORTPE may be rerun to reevaluate the status of the sort/merge application by entering:

RUN "SORTPE"

9. The user should record the volumes and datasets containing the desired sort output data. The user could then rename or copy the datasets as desired.
10. If an error condition is detected, refer to Appendix C for restart steps to be performed (if possible).
11. If SORTPE was entered on detection of a user intervention request, the sort/merge application may be restarted by running the program segment last in control as displayed by SORTPE."

Given below is a summary of the information request messages issued by SORTPA during the assignment phase. A discussion of possible user responses is also given.

***** SORT PARAMETER SPECIFICATION *****

This section requests the user to specify certain information on the type of sort to be performed. These include:

IS THIS A SMALL SORT (Y/N)? (See Appendix D)
IS THIS A MULTI-VAR OR CHAR-STRING SORT (V/S)?
ARE FILES SEQUENTIAL OR RANDOM (S/R)?

***** WORK VOLUME SPECIFICATION *****

This section will request the names of the work volumes (empty diskettes) to be used by DOSORT. Default names of WORKA, WORKB, ... etc. are assigned by DOSORT if not otherwise directed. The following message will be issued:

VOLUME NAME?

The user should respond with an arbitrary name for each work volume desired. Approximately the same number of work volumes as input volumes will be required. The user should label the work volumes since DOSORT may require several volume switches to be performed during a given sort application. Mounting the wrong disk causes disastrous results.

Work volume specification entry can be terminated by pressing the enter key with no volume.

***** DRIVE AVAILABILITY SPECIFICATION *****

This section indicates to DOSORT which drives are available for general use. Respond "Y" or "N" to each of the following:

DRIVE 1 AVAILABLE (Y/N)?
DRIVE 2 AVAILABLE (Y/N)?
DRIVE 3 AVAILABLE (Y/N)?

***** SORT INPUT FILES SPECIFICATION *****

This section requests the user to describe the sort input datasets and the volumes to be used during the sort/merge application. The following pairs of requests will be generated for each sort input file required:

DATASET NAME?
VOLUME NAME?
ESTIMATED NO. OF RECORDS?

The user should then enter the dataset name as maintained in the directory (without the drive No.) followed by an arbitrary volume name. The volume name identifies the volume on which the corresponding dataset is located.

The above message will be repeated for additional dataset/volumes to be included in the sort. Pressing the enter key with no dataset name will terminate the sort input file specification. If no files are to be sorted (a merge only application) this section can be immediately terminated.

***** MERGE FILE SPECIFICATION *****

This section is used to specify user merge files to be included in the sort/merge application. More than one user merge file may be specified, each consisting of one or more datasets contained on different volumes. DOSORT assumes all user merge files are already in the correct sort sequence and need only be merged with other system or user merge files. The following message is displayed:

ADD MERGE FILE NO. 1 (Y/N)?

If user merge files are to be added respond "Y", otherwise "N" will terminate this section. The following sets of messages, similar to sort input file specification solicitations, will be issued:

DATASET NAME?
VOLUME NAME?
ESTIMATED NO. OF RECORDS?

The user should enter the corresponding dataset/volume for each dataset in the user merge file. These should be specified in correct sort sequence. The end of a user merge file is signified by pressing the enter key with no dataset name. The "ADD MERGE FILES" message will be reissued to allow additional user merge files to be entered into the system.

***** SORT KEY SPECIFICATION *****

This section solicits information on the variables or fields to be used for sorting. The response required depends on the mode selected in Sort Parameter Specification section.

1. Multiple Variable Mode

ENTER SORT KEY?

The user should respond with a list of variables preceeded by a + or - as described in the GSF Users Manual. The response must be enclosed within quotes since the variable list has embedded commas (e.g. "+SXS,-IG,PB,NMS"). These names must match those used on the DIM statement at line #100.

NO. INTEGER SORT VARIABLES?

Enter the number of integer variables contained in the sort key/variable list entered above.

NO. SINGLE PRECISION VARIABLES?

Enter the number of single precision variables contained in the sort key/variable list entered above.

NO. DOUBLE PRECISION VARIABLES?

Enter the number of double precision variables contained in the sort key/variable list entered above.

NO. CHARACTER STRING VARIABLES?

Enter the number of character string variables contained in the sort/variable list specified above.

SUM OF CHARACTERS/RECORD IN STRING VARIABLES (MAX)?

Enter the combined length of all character string variables contained in any one sort record to be read or written. This is used to calculate the string space required.

2. Character String Mode

NAME OF CHARACTER STRING ARRAY TO BE SORTED?

The user should enter the name of the array to be used for storing the character record. This is the same name as found in the user DIM statement at line #100.

NO. SORT KEYS?

Enter the number of fields within the character record to be used as sort keys.

SORT KEY NO. ____ LOCATION(+ ASC, - DES)? ____
LENGTH? ____

The above message is issued for each sort key. This is used to specify the relative location and length of each field in the record. The location is specified positive for ascending or negative for descending sort precedence.

CHARACTERS/RECORD(MAX)?

This specifies the maximum length of records in the files to be processed. Records may be smaller than this value but all sorts by fields must be contained in the smallest record.

CAUTION

GOOD PRACTICE DICTATES THAT ALL FILES BE BACKED UP PRIOR TO PERFORMING SORTS AND MERGES TO ENSURE PRESERVATION OF DATA INTEGRITY.

APPENDIX A. USER VOLUME SWITCHING TECHNIQUES

DOSORT reads and writes multiple volume files by performing volume switching upon a "DISK FULL" or "INPUT PAST END" condition. The user may be required to perform volume switching when reading files produced by DOSORT. Automatic volume switching may be performed by the user for files to be procured by DOSORT, although such techniques are not mandatory. Given below are the techniques recommended for both of these cases.

1. USER INPUT VOLUME SWITCHING

Error code #124 is issued by the TRS-80 BASIC system upon detection of an end of dataset condition. DOSORT utilizes the "ON ERROR" technique to detect this situation. The user may use either this technique or the "EOF" function as shown in the example below:

Example #1 - EOF Technique

```
10 OPEN "I",1,"MERGE" :REM Open first
20 IF EOF(1) THEN 1000 :REM Test EOF
30 INPUT #1, .... :REM Use Input Statement
:
:
100 GOTO 30
1000 CLOSE 1
1010 PRINT "SWITCH TO NEXT VOLUME - PRESS ENTER"
1020 IF INKEY$ <> CHR$(13) THEN 1020
1030 OPEN "I",1,"MERGE"
1040 GOTO 30
```

Example #2 - ON ERROR Technique

```
10 ON ERROR GOTO 1000 :REM To Catch EOF
20 OPEN "I",1,"MERGE" :REM Open First Volume
30 INPUT #1, ... :REM User Input
:
:
100 GOTO 30
1000 IF ERR<>124 THEN ON ERROR GOTO 0 :REM Revert to BASIC
1010 CLOSE 1
1020 PRINT "SWITCH TO NEXT VOLUME - PRESS ENTER"
1030 IF INKEY$ <> CHR$(13) THEN 1030
1040 OPEN "I",1,"MERGE" :REM Open Next Volume
1050 RESUME :REM Repeat
```

2. USER OUTPUT VOLUME SWITCHING

The user can prepare output datasets contained on different volumes without performing special automated volume switching. The only requirements are that no single print record spans two volumes and no print record can be larger than 255 bytes. For special applications, the techniques used by DOSORT may be utilized in user application programs. The "ON ERROR" technique is used to intercept Error Code #122 (Disk Full) as outlined below:

```
10 DEFINT I-N:OPEN"R",1,"DUMMY" :REM Locate I/O Buffer
20 FIELD 1,2 AS U1$
30 IQ PEEK(VARPTR(U1$)+1)+256*PEEK(VARPTR(U1$)+2)
40 CLOSE 1:OPEN"O",1,"USER1" :REM Open First DSN
50 ON ERROR GOTO 2000 :REM Catch #122
60 PRINT #1, .... :REM User I/O
100 GOTO 60
```

APPENDIX B. CASSETTE TAPE

Software programs purchased from RACET computes should load at the same volume setting as your standard setting for tapes produced by your computer. It is recommended that you periodically clean the head capstan and pinch rollers on your cassette recorder using commercially available cassette cleaning and demagnetizing accessories. Dirty heads can cause substantial loss of volume and induce unwanted noise.

The General Subroutine Facility (GSF) object code is on the reverse side of your DOSORT tape. Procedures for loading GSF are included in Appendix C of your GSF manual. The DOSORT side of your tape contains the five DOSORT BASIC modules.

<u>Program</u>	<u>Recorded on Tape As</u>
SORTPA	"A"
SORTPB	"B"
SORTPC	"C"
SORTPD	"D"
SORTPE	"E"

GSF and the five BASIC programs should be loaded from tape and transferred to a diskette containing only the DOS. The BASIC programs should be transferred using standard TRS-80 procedures (CHD"T", CLOAD"A",SAVE"SORTPA:0",etc.). It is recommended that a backup of the resultant diskette be accomplished to reduce the time required to recreate your DOSORT diskette.

APPENDIX C. RESTART PROCEDURE

During the course of the operation of DOSORT, a number of system, user, or hardware errors may occur. DOSORT writes checkpoints at intermediate stages during processing. When the sort/merge is completed, or an error occurs, SORTPE is used to summarize the states of the sort and to assist in restarting the sort.

A number of errors are beyond the control of DOSORT - such as hardware failures and user data input errors. Generally, any sort phase may be restarted by removing the current active file from the DOSORT SORTPARM table and reexecuting the phase in error. The SORTPE menu lists the specific action that can be initiated by the user.

The user can remove the current active output file by executing Option #2 and then attempt a restart by Option #3-6. Alternatively, Option #7 attempts the restart of the sort automatically.

The above technique will not be effective if the problem resides in input data, temporary write errors, or incorrect diskettes inserted at any time. The user should maintain suitable backup copies of critical datasets should such errors occur.

APPENDIX D. SMALL SORTS

Users with a minimum disk configuration may circumvent some volume switching when handling small files. A file is considered to be small when the following conditions are met:

- The input file consists of one dataset on one volume
- Sufficient space exists to allocate the required intermediate SRTBLK datasets produced by DOSORT

The amount of data stored in the SRTBLK dataset is equal to the original input dataset plus up to one granule of overhead. The number of SRTBLK datasets produced is dependent upon memory size available and amount of input data as indicated below:

File Size		No. SRTBLK Datasets		Free Space Required			
				Grans		% Available	
Grans	Bytes	32K	48K	32K	48K	32K	48K
10	12800	2	1	12	11	18%	17%
20	25600	3	1	23	21	35%	32%
30	38400	4	2	34	32	52%	49%

Note that a 30 granule file would be considered a small file for a 48K system but not for a 32K system since the work space required exceeds 50%. The above should be considered a rough approximation which may vary depending upon the characteristics of the data being sorted.

The user may indicate to DOSORT that the small file sorting technique is to be used as described earlier. This AUTOMATICALLY allows DOSORT to DELETE THE USER'S INPUT FILE. The sorted output file will replace the user input file. The user should produce a BACKUP copy of the file being sorted prior to initiating the Small Sort.

APPENDIX E. EARLY TERMINATION OF USER INPUT

DOSORT normally reads the entire input file in SORTPB until the final end-of-file of the last input dataset. This process is handled automatically by DOSORT requiring no action by the user.

The user may terminate reading of the input file in SORTPB prior to the normal end-of-file as illustrated below:

```
CLOSE #YL: Y1 = 2 : RETURN
```

An example of the use of the technique is to terminate the SORTPB input after processing a given number of records as shown below.

```
1000 IC=IC+1: IF IC > 1000 THEN CLOSE #YL: Y1=2:RETURN
1010 INPUT #YL ... :REM User Input
      :
      :
```

Note that variable IC must be explicitly initialized in lines #101-199 as described in the User Modification Section.

The above technique can also be used in SORTPD for terminating user merge input files. In this case it will be necessary to test variable TY to determine if the input file being processed is a user merge file (TY=1).

SOFTWARE AGREEMENT AND OWNER REGISTRATION

All RACET computes software is made available under a perpetual license agreement. A separate license is required for each system on which the software will be used. All copies will be procured from RACET computes or its distributors for resale.

IMPORTANT NOTE: The user, upon receipt of this product agrees to the terms and condition of the lease. The user may return the software within 30 days of receipt for a full refund if the terms of the license are not acceptable.

Product _____

Name _____

Address _____

City _____

State _____

Date Purchased _____

Where Purchased _____

System Configuration (Optional) _____

Signature Date

RACET computes will provide corrections to this program or associated documentation for those users completing this registration form, and returning it to:

RACET computes
702 Palmdale
Orange, CA 92665

NOTE: Quantity discounts are available. Contact RACET computes at the above address, or telephone (714)637-5016

```

110 '
115 ' ----- APPLICATIONS NOTE -----
120 '
130 ' Although DOSOFT can be used to sort RANDOM files it may
140 ' be more advantageous to use an indexed sort technique.
150 ' Given below is a typical example utilizing the fast GSF
160 ' internal sort subroutines which is included with DOSOFT.
170 '
180 ' Assume the user has a random file contained on one
190 ' cassette with the following attributes:
200 '
210 ' 1. File Name           EX1:1
220 ' 2. Record Size        64 bytes packed 4 rec/sector
230 ' 3. Number of Records  Variable as determined by
240 '                        L/F(1)<#
250 ' 4. Record Content     LRC NAME DECRYPTION
260 '
270 '
280 '
290 '
300 '
310 '
320 '
330 ' The purpose of this example is to illustrate how the
340 ' user can process the file in sorted order, such as by AL,
350 ' NAME, or GSF. The program used is as follows:
360 '
370 ' Line # Description
380 ' -----
390 ' 1-400 Initialize
410 ' 410-420 Open file - Determine number blocks using L/F
420 ' 420-430 Describe field for each of the four rec/sector.
430 ' 430-450 Read the entire file, saving just the keys in
440 ' arrays ZC, NC, and C, along with a record number
450 ' in array IS.
460 ' 460-470 Sort the file using GSF with NAME(NC) as the
470 ' primary key, C, ZC, and IS as secondary keys.
480 ' This is described in the GSF Users Manual on
490 ' Pages 17-19.
500 ' 500-510 Print the file in sorted order by using the
510 ' record number in IS corresponding to each
520 ' sorted entry.
530 ' 530-540 Repeat the process using a different sort
540 ' sequence, such as "-C,+ZC,+IS,+IS".

```

```

550 '
560 '  PROGRAM
570 ' -----
580 '
590 CLEAR 20000:DEFINT A-Z
600 DIM DMS(3),ZPS(3),HMS(3),CDS(3),XSS(3),ZC(1001),HC(1001),C(1001)
610 OPEN "R",1,"EX1:1":      HL=LOC(1)
620 FOR I=0 TO 3
630   FIELD 1, 1*64 AS DMS(I), 6 AS ZPS(I), 10 AS HMS(I), 2 AS CDS(I), 45 AS XSS(I)
640 NEXT
650 NR=-1
660 FOR I=1 TO NR
670   GET #1,I
680   FOR J=1 TO 4
690     NR=NR+1
700     ZC(NR)=ZPS(J)
710     HC(NR)=HMS(J)
720     C(NR)=CDS(J)
730     IS(NR)=NR
740   NEXT
750 NEXT
760 VAL="*NR,*C,*ZC,*IS"
770
780 I=NR(1)+4:NR(VALPTR(V.)) ON NR(1) GOTO INT(NR)
790 '
800 FOR I=0 TO NR
810   I=I+10000
820   GET #1,I
830   J=IS(I)-1*4
840   PRINT NR(J),NR(1),HC(J),XSS(C)
850 NEXT
860 LINE INPUT "TRY ANOTHER SIMULACE? Y/N=";CNC
870 GOTO 780

```

March 1, 1979

ERRATA No. 1 TO DOSORT - Version 1.0

1. All references to variable "S\$" on page 7-9 should be changed to "A\$".
2. The DIM statement on page 12 should be:
100 DIM A\$(YN)
3. The name of the character string array to be sorted as described on page 12. In character string mode the variable name **MUST** be A\$.
4. The user output volume switching described on page 13 is incomplete. The entire correct example is shown below:

```

10  DEFINIT I-N:OPEN"R",1,"DUMMY"           :REM  Locate I/O Buffer
20  FIELD 1,2 AS U1$
30  IQ=PEEK(VARPTR(U1$)+1)+256*PEEK(VARPTR(U1$)+2)
40  CLOSE 1:OPEN"O",1,"USER1"               :REM  Open First DSN
50  ON ERROR GOTO 2000                       :REM  Catch #122
60  PRINT #1, ....                          :REM  User I/O
:
:
100 GOTO 60
2000 JQ=IQ+255                               :REM  Truncate buffer
2010 IF PEEK(JQ)=13 THEN 2030
2020 JQ=JQ-1: IF JQ =IQ THEN 2010 ELSE STOP
2030 JQ=JQ-IQ+1
2040 POKE IQ-24,JQ: POKE IQ-27,JQ
2050 CLOSE 1                                 :REM  Close first file
2060 PRINT "SWITCH TO NEXT VOLUME - PRESS ENTER"
2070 IF INKEY$ CHR$(13) THEN 2070
2080 OPEN"O",1,"USER2"                      :REM  Open next file
2090 RESUME                                 :REM  Reexecute I/O

```
5. Users processing random files must perform their own volume switching in the current version of DOSORT if a file extends across several diskettes. The initial files will be opened as "R" files, but it is the users responsibility to pack or unpack the data (using FIELD commands) as required.
6. Memory sizes for the Extended GSF furnished with DOSORT are 45784 for the 32K version and 62168 for the 48K version. These memory sizes **MUST** be used when working with DOSORT.
7. It is highly recommended that work volumes used should have maximum empty space available (typically 67 grans) to minimize volume switching.
8. When transferring the Extended GSF for DOSORT from memory to disk, use a START=X'B2D8' for the 32K version and START=X'F2D8' for the 48K version.