

Data File Manager (DATFILE) Manual
(Scanned from a photocopy by Terry Stewart, June 2019)

Contents

<u>Section</u>		<u>Page.</u>
1.0	Introduction.....	1
1.1	Program features.....	1
1.2	Terms you should know.....	2
1.3	Minimum equipment.....	2
2.0	INITIAL STARTUP.....	3
2.1	Composing your Data Form.....	6
3.0	DATFILE ACTION TABLE.....	9
3.1	Adding Data.....	11
3.2	Editing/Viewing Data.....	11
3.3	Sorting Data.....	15
3.4	Printing Data.....	16
3.5	Command Mode.....	18
4.0	DATFILE COMMANDS.....	18
5.0	FILES PRODUCED BY DATFILE.....	19
6.0	STARTING OVER AGAIN.....	20
7.0	FORMATTING DATA DISKS.....	21
8.0	PERFORMING BACKUP.....	22
APPENDIX A:	EXAMPLE DATA FORMS.....	23
APPENDIX B:	EXAMPLE PRINT FORMATS.....	25
APPENDIX C:	USING DATFILE FROM DISK BASIC.....	27

1.0 INTRODUCTION

DATFILE is a general purpose data filing system for the Dick Smith SYSTEM-80 computer. The versatility of DATFILE permits use in any area where information is to be stored for later display, sorting, printing or modification. These uses are endless and only limited by the imagination, in both the home and business environment. Examples of these uses are:-

A) In the home

- 1) Recipe file.
- 2) Cheque book file.
- 3) Grocery list.
- 4) Address book.
- 5) Amateur radio log book.

B) In the business environment

- 1) Stock list.
- 2) Personnel file.
- 3) List of Customer details
- 4) Club membership records
- 5) Mailing lists

C) Rural Applications.

- 1) Stock pedigree records.
- 2) Calving and performance.
- 3) Plant register and specifications.
- 4) Paddock records.

1.1 PROGRAM FEATURES

1. Up to 4 Disk drives may be used to store information.
2. The data format displayed on the screen and the printed format may be individually defined to suit any requirement.
3. On-the-screen editing facilities are available to make addition or modification of any existing file quick and simple.
4. DATFILE is able to display/edit/sort or print individual records or selective groups of records.
5. Disk information produced may be later manipulated by Disk Basic.

1.2 TERMS YOU SHOULD KNOW

Here is an explanation of the terms used by DATFILE and other computerised data management systems.

FIELD: an item of information entered into DATFILE. An example may be:-

COMPANY: DICK SMITH ELECTRONICS

A Field is composed of a field name and then data.

FIELD NAME: the name given to a particular item of information. In the example above, the field name is 'COMPANY'.

FIELD DATA: the data portion of the field. In the example above, the field data is 'DICK SMITH ELECTRONICS'.

RECORD: the accumulation of a number of fields to form a page of information e.g. a record may be all the information concerning a particular employee.

FORM: simply the visual representation of all the information that can be entered into each record, it is in fact the 'worksheet' of DATFILE.

FILE: the concatenation or 'adding together' of all information entered into DATFILE i.e. all records entered. The number of these records and consequently the size of the file will depend on the number of Disk drives you are using.

1.3 MINIMUM EQUIPMENT REQUIREMENTS

- a) One 16K SYSTEM-80 computer.
- b) One S-100 expansion interface.
- c) One 16k dynamic RAM card.
- d) One or more 40 track Disk drives (more can be added for greater capacity to a total of four Disk drives).
- e) One printer (This is optional).
- f) One printer cable (only if printer is used !!!!).

1.4 INITIAL SETUP

Before using DATFILE for the first time it is wise to perform a backup of the supplied progRAM diskette as a precaution against accidental erasure or corruption. This backup is performed by the DATFILE command BACKUP. Refer to section 7.0 for a detailed backup procedure.

This BACKUP should be the diskette that you use in your every day operations and the original diskette should be kept in a safe place. Remember that diskettes can be damaged very easily by magnetic fields or by mishandling. Since diskette life contributes significantly to system failures it is wise business practice to backup your diskettes at regular intervals during use.

If you plan to store data on one or more diskettes then they must be formatted. Refer to section 7.0 for a detailed formatting procedure.

Your DATFILE program package contains one diskette, this is your MASTER DATFILE diskette. Always work from backups made from your MASTER diskette. Keep the MASTER diskette in a safe place. When using DATFILE for a new application make a backup from your MASTER diskette. In this way you will have a number of DATFILE diskettes each configured for specific uses.

Start up procedure:-

- a) Turn on your Disk drives, video monitor and S-100 expansion interface.
- b) If you are to use a printer, turn it on now.
- c) Turn on your SYSTEM-80.
- d) Insert your DATFILE diskette into Disk drive 0.
- e) If you plan to store data on more than one disk then insert formatted diskette(s) into Disk drive(s) 1-3.
- f) Press the reset button on the left rear side on the SYSTEM-80.

Your video monitor should now show:-

```
< DATFILE--VERSATILE DATA FILING SYSTEM--V1.8>
> DATFILE COMMAND :
-
```

Your DATFILE diskette contains a complete set of data management utilities to enable you to maintain your data system.

At this point you may give one of five special commands which are used to perform such functions as FORMATTING data diskettes, BACKUP or your data diskettes etc..

A description of these utilities and references to more detailed descriptions can be found in Section 4.0.

If this is the first time you have used your DATFILE diskette, the command you should now use is FILE.

[Type the command:-]

FILE <newline>

After a short pause your video monitor should show:-

```
< DATFILE-INITIALISE >

>HOW MANY DRIVES WILL YOU USE (1-4)?
```

Depending upon the number of disks you want DATFILE to be able to access at once, key in a number from 1 to 4. Then press the <NEW LINE> key.

2.1 COMPOSING YOUR DATA FORM

Your video monitor should look like this:-

```
< DATFILE-DATA FORM >

-----
<E>DIT      <F>ORWARD  <D>ELETE    <BREAK> TO EXIT
<S>EARCH    <B>ACK      <P>RINT     <+U> <+D>
* SETUP YOUR FORM --- PRESS <BREAK> WHEN COMPLETED *
```

The flashing cursor is now positioned on the top left hand corner of your screen. With this cursor you can now construct your data form.

Before starting you should become familiar with the various editing aids available to you when entering data into DATFILE.

Start by typing your name.

Press the <SHIFT> key down and while holding it, press the <R> key. You will notice that the cursor moved to the right by one character. If you keep the keys depressed for a little longer you will see the cursor keep moving to the right until the keys are released.

Now press the <BACK SPACE> key. And the cursor will now move to the left by one space. The auto key repeat feature also applies to this and every other key.

The <SHIFT> and <U> and the <SHIFT> AND <D> keys can be used in a similar fashion to move the cursor up and down respectively.

Now position the cursor at the beginning of your name and press the <SHIFT> and <I> you will see a space inserted before your name this is the INSERT function. This is used to spread the text in your data form so that additional characters may be INSERTED.

Press the <SHIFT> and the <E> key and the space you just INSERTED will be ERASED. This sequence is used to delete unwanted characters from your data text.

The following is a quick reference for the editing aids:-

<BACK SPACE>	Moves the cursor one space to the LEFT.
<SHIFT+R>	Moves the cursor one space to the RIGHT.
<SHIFT+U>	Moves the cursor UP one line.
<SHIFT+D>	Moves the cursor DOWN one line.
<SHIFT+I>	INSERTS one space before the cursor.
<SHIFT+E>	ERASES the character under the cursor.

We are now in a position to begin setting up a sample data form. Let's use a simple personal address list as an example.

DATFILE's cursor can be made to move any where on the screen from the top to the border area at the bottom indicated by '-'. It is therefore possible to overwrite the DATFILE heading line at the top of the screen and to replace it with our example:-

**** PERSONAL ADDRESS LIST ****

This topline is reserved for some sort of title text only, so that data fields should not be declared on this line or they will be ignored.

Next we enter the first field name on your form. Move the cursor to the first line following the title line. Type the field name NAME.

All field names must be followed by a colon ':' and a space so that DATFILE can distinguish the field name from the field data. The area required to be reserved for field data is now shown by a series of full stops '.' with the number of full stops being equal to the number of characters you anticipate will be required for the field data.

Your field line should look something like this:-

NAME:.....

In DATFILE you are not restricted to only one field per line. You may have more than one per line provided there is at least one space between the last full stop and the beginning of the next field name. There are however a maximum of 32 fields which may compose a complete data form.

In order to assist you in an understanding of the flexibility of DATFILE, a number of example data forms have been prepared and are given in appendix A. These have been chosen as representative of most variations of data forms in DATFILE.

When all fields have been declared, press the <BREAK> key at which point DATFILE will store an image of your form on disk for use later in data entry. The Disk drives you have chosen for use will be initialised at this point.

The actual number of records that may be stored is a factor of the number of fields in your form, the area you have reserved for field data and the number of disk drives you have in use.

3.0 DATFILE ACTION TABLE

It is this table that is displayed following the initialisation phase when first starting up or if **DATFILE** is entered when data has already been declared.

This is how the table looks on your screen:-

```
< DATFILE--VERSATILE DATA FILING SYSTEM--V1.8 >

RECORDS USED:      0
RECORDS LEFT:     599

<A>D D DATA
<E>D I T / V I E W D A T A
<P>R I N T D A T A
<S>O R T D A T A
<C>O M M A N D M O D E

>MAKE YOUR SELECTION: █
```

xxxx indicates the number of records which have already been filled with data.
yyyy indicates the number of free or remaining records to be filled with data.

- | | |
|---|--|
| <u><A>D D DATA</u> | Allows you to enter field data in a new record. |
| <u><E>D I T / V I E W D A T A</u> | Allows you to look at, change, or print data in a particular record or group of records. |
| <u><S>O R T D A T A</u> | Allows you to sort your records into some given order. |

<P>RINT DATA

Allows you to print individual records or groups of records or all records in an individually defined printout format.

<C>OMMAND MODE

If this command is given you will enter the DATFILE command mode. All of your data files are then saved on disk.

3.1 <A>DDING DATA

<A> is selected if you wish to add further records to the information already stored. For instance if you are keeping a filing system of all members of a club (see examples in appendix A) and a new member joins the club then his/her name and other data would have to be added to the file.

DATFILE will display a form containing field names without field data, allowing you to enter the new field data. When all data has been entered the new record is written to disk along with all your previous data.

Use the cursor movement as explained in section 2.1 to enter field data for all fields. You can, of course use all the editing facilities available in DATFILE.

3.2 <E>DITING/VIEWING DATA

This option is used if you wish to change, print, delete or simply look at data already entered into a particular record or group of records. As an example, this option would be used to remove the information concerning a club member who has resigned from the club or to change the address of a member who has moved.

A number of sub-options are available in this mode. These are displayed at the bottom of the form and will be explained individually.

a) <E>dit.

Press <E> if you wish to change the information if the record currently being displayed on the screen. The cursor will then move to the start of the data for the first field and you may then use all the editing facilities described in section 2.1 to change any field data.

b) <S>earch.

Press <S> if you wish to search for a particular record or group of records which satisfy a given search criterion.

As an example, assume you wish to find all the members in a club whose surname is SMITH. DATFILE will ask the name of the field you will use to find the record. The field name containing the name of each member may be NAME. If you have previously entered a field when searching and you wish to use this name once again simply press <NEW LINE>. The previous field name will be displayed.

You are then asked for the relationship to be used in the search. As for the field name above, if <NEW LINE> is typed in the first character position and then the last entered relationship is displayed and used. A number of relationships are possible. Here is a list of these and the operations they perform:-

- EQ - used if you wish to find all records for which field data is EQUAL to the data given.
- NE - used if you wish to find all records for which field data is NOT EQUAL to the data given.
- GT -- used to find all records for which field data is alphabetically or numerically GREATER THAN the given data.
- LT - used to find all records for which field data is LESS THAN the given data.
- GE - used to find all records for which field data is GREATER THAN OR EQUAL to the given data.
- LE - used to find all records for which field data is LESS THAN OR EQUAL to the given data.

Returning to the example given above the relationship would be EQ.

You are then asked what DATFILE should look for.
You would answer SMITH.

DATFILE will search all your records for the first record where the first four characters in the field with the name of NAME is JOHN.

Remember when searching for records that the letter 'A' is seen to be less than the letter 'B' and that numbers are less than letters.

As a further example, if you wished to find all the members who joined the club on or after the first of February 1980 and we had a field name called:-

JOINING DATE :

where each date was expressed in the form 80/02/01, then we could ask DATFILE to perform a search for the field JOINING DATE with the relationship of GE (for greater than or equal) and search for 80/02/01.

When the particular record satisfying the given conditions is found, DATFILE gives you the opportunity to continue searching for another record using the same field name and relationship. To do this simply type the new date to be found and hit <NEW LINE>. This facility is seen to be handy for business purposes. If you wish to terminate the search or change the field name or relation simply hit <BREAK>.

We could then step through the particulars of all members fitting the given category by use of the <F>orward or ack options. These will be discussed next.

c) <F>orward.

Press the <F> key following the use of the <S> key if you wish to step forward through all your records looking for more data that satisfies your search criteria. Each time the <F> key is hit a new record will be displayed or DATFILE will indicate that there are no more records that satisfy the search.

You can at any time while stepping through your records decide to use the <E>dit, <D>elete, <P>rint or to vary the search criteria.

d) ack.

Pressing a following a <S>earch order will cause DATFILE to step back through your records, displaying all earlier records satisfying the search criteria.

e) <P>rint.

Press a <P> if you require a printout of the information in the current record, just as it is displayed on the video monitor.

f) <D>elete.

Pressing a <D> permits you to delete from your file all information about the currently displayed record. As a precaution against accidentally destroying a record DATFILE will require verification of your delete order before continuing.

g) <BREAK> to EXIT.

Pressing <BREAK> at any time will remove you from the EDIT/VIEW DATA mode and return you to the DATFILE ACTION TABLE once again.

h) <^U> and <^D>.

If at any time you wish to step to display the next record in your file ignoring any search criteria then you can use the <SHIFT+U> keys as described in section 2.1.

If you wish to step back through your file use the <SHIFT+D> keys.

3.3 <S>ORTING DATA

Use the <S> key if you wish to SORT your file data into a particular order. So that you may, for instance print or search for records in some known order.

DATFILE will ask for the name of the field you wish to use as the sorting "key". In other words, the field whose data is actually used to perform sorting. You may elect to use the last field name, if so just press <NEW LINE> and this will be done.

Your records may be sorted according to any field and in either ASCENDING or DESCENDING order. Use ASCENDING if you wish to for instance, sort all members names in alphabetic order, or DESCENDING if you wish to sort in reverse order.

Since DATFILE will sort strictly according to the field data and ignoring the fact that some fields may contain numeric data, there are some suggestions that should be made at this point. If as in the example of a club membership list, you wish to sort all members according to their joining date, then each record date should be entered in the format of YY/MM/DD or YYYY/MM/DD so that the most significant portions of the date appear first in the data field. If the data was entered in the form DD/MM/YY then DATFILE will think that the date 01/11/79 was after the date 01/03/80 since the day and month are more significant than the year.

Also remember that all field data in numeric form should be entered in the same way. For example if the figure \$123.45 is entered into one record then the sum \$94.78 must be entered in the form \$ 94.78 with the preceeding space. This will position the decimal place always the same number of characters into the field data and DATFILE will correctly find that \$123.45 is greater than \$94.78.

Sorting data may require a number of sort passes to complete the sort. DATFILE will keep you informed as to the current pass it is executing.

When sorting is complete DATFILE will return you to the ACTION TABLE once again.

3.4 <P>RINTING DATA

Use the <P> key when you wish to print a data file in table form with specific fields in certain columns.

As with your DATFILE form used for field data entry, you can specify the format that you wish data to be printed.

The printed page has been broken up into 132 columns on the page. If you use the <SHIFT+R> key to move the cursor to the right you will see that all 132 columns are actually available to you.

You will notice that there are 3 lines on the screen as well as the printer graduation line that appears on your screen in the following format:-

```
< DATFILE-PRINTING >
  10    20    30    40    50    60
-----+-----+-----+-----+-----+-----
( PAGE TITLE LINE )
( COLUMN TITLE LINE )
( FIELD SIZE LINE )

#PRESS <BREAK> WHEN COMPLETED
```

The PAGE TITLE LINE can be changed using the cursor positioning keys as described in section 2.1, to show the actual title line which will be printed on the top of each printed page (see appendix B for sample formats).

The COLUMN TITLE LINE can be changed in a similar manner to indicate the title given to each column of information to be printed. Space the column titles to make best use of the column width of the printer you are using.

The FIELD SIZE LINE should indicate by means of full stops '.' the number of characters of information that you expect to be printed from each field. The more full stops the more characters are able to be printed, so make sure that the number you supply are sufficient to cover the number of characters in the field to be printed.

When you have completed composing your printout format press the <BREAK> key. DATFILE will ask if you are happy with your format.

DATFILE will then ask for the field names of the field data you wish to be printed in the various columns from the left hand side of the page to the right. This is because it is not necessary for the column title you have given to each column to be the same as any existing field name. This choice is your own.

When all columns have been designated, DATFILE will ask if you wish to print all your data in the order in which it was last sorted (Refer to section 3.3). If the answer is no, then DATFILE will ask you what you wish to print. The response should be given in the same form as for the <S>earch function (see section 3.2 b).

Study some of the examples in Appendix B if you are uncertain as to what you can do with your printout format.

* note Sector 5 track 4 Bit 30 is the number of lines per page.

3.5 <C>OMMAND MODE

Once all work on your data files has been completed you should return to DATFILE command mode by typing a <C>. Never remove your DATFILE diskette after use without first using this command and returning to command mode.

Upon entering command mode, you can if you wish, remove your DATFILE diskette(s) and safely store them away until you next wish to use DATFILE. All data has been saved for you so that you may re-enter DATFILE at a later time with any updates intact.

4.0 DATFILE COMMANDS

When in DATFILE command mode one of five data management commands are available to you to assist you in maintaining your data system.

A brief description of these commands follows:-

a) BACKUP

This command enables you to produce copies of your DATFILE disk and other disks as security against disk faults or misuse. Refer also to section 8.0.

b) DIR

This command enables you to check on the files found on your DATFILE disk or other data disks. The format of the command is:-

DIR :dn <NEW LINE>

Where dn is the Disk drive number (0-3) NOTE: if no drive number is specified drive 0 is assumed.

c) FILE

This command permits the operator to enter a mode where the data that has been stored may be changed, added to, printed or simply looked at. No command parameters are necessary.

d) FORMAT

This command is used to initialise a diskette by clearing all data on the diskette. This is required when using more than one disk drive to carry information. Refer also to section 7.0.

e) KILL

This command is used to delete a file from a diskette. Due to the power of this command, extreme care must be exercised when using it. Refer also to section 6.0.

5.0 FILES PRODUCED BY DATFILE.

DATFILE produces a number of data files during operation which are necessary to save data so that you are able to turn your computer off and not lose your valuable data.

A brief description of these files follows:-

DATVDU This file carries a picture of your data form for later use. Created only on disk drive 0.

DATPNT This file carries a picture of your printout requirements. Created only on disk drive 0.

DATFLD This file carries data concerning the deposition of information in your form. Created only on disk drive 0.

DATINF This file carries all the data in all records. Created on all disk drives carrying data.

6.0 STARTING OVER AGAIN

If at any time you wish to clear all information you currently have stored and start all over again with more field names or a completely new data form, then this is what you have to do.

As a word of caution for those who may have misunderstood, the following procedure will **COMPLETELY DESTROY** any data already entered.

Before continuing you must have entered **DATFILE** command mode by typing <C> when in the **ACTION TABLE**. Refer section 3.5.

As indicated in section 5.0 , there are a number of files automatically produced by **DATFILE** during operation. All of these files must be deleted from your **DATFILE** disk on drive 0. Use the command **KILL** for this purpose as follows:-

Type KILL DATVDU	then press <NEW LINE>
Type KILL DATPNT	then press <NEW LINE>
Type KILL DATFLD	then press <NEW LINE>
Type KILL DATINF	then press <NEW LINE>

Give the command **DIR** then <NEW LINE> to confirm that your **DATFILE** disk is now empty. You now have a clean disk on which to start over beginning with section 2.0.

7.0 FORMATTING DATA DISKS

Diskettes must be FORMATTed before first being put into use. This procedure will only be necessary when the diskette is first being used to carry data.

Before continuing you must have entered the DATFILE command mode by typing <C> when in the ACTION TABLE. Refer to section 3.5

Following is a step by step procedure for formatting a new disk:-

a) Insert a new diskette or a diskette you wish to re-format into drive N.

b) Type:-

```
FORMAT,:N,DATFILE,mm/dd/yy,DATFILE <NEW LINE>
```

where: mm/dd/yy is the current date.

N is the disk drive that contains the diskette you wish to format (0-3)

c) If the disk already contained data then DATFILE will ask for confirmation before formatting the diskette. If you are satisfied that this is the diskette that you wish to re-FORMAT then type 'P' else type a 'C' this will cancel the command.

WARNING

FORMATting will destroy any information currently on that diskette, so be certain.

d) If more diskettes are to be FORMATTed return to step (a).

8.0 PERFORMING A BACKUP

Good data management techniques call for regular BACKUPS of all data. This BACKUP will be necessary on your DATFILE diskette as well as on any additional data diskettes. The following is the recommended backup procedure.

Before continuing you must have entered the DATFILE command mode by typing <C> when in the ACTION TABLE. Refer to section 3.5

- a) Insert the diskette to carry the backup into the disk drive you have chosen as the destination.
- b) Type:-

BACKUP :d1 TO :d2 mm/dd/yy <NEW LINE>

Where: mm/dd/yy is the current date.
 d1 is the number of the source disk drive.
 d2 is the number of the destination disk drive.
- c) When you are asked to insert the source diskette into drive d1, insert the diskette you wish to BACKUP or copy into drive d1 and hit <NEW LINE>.
- d) When asked to insert the destination diskette into drive d2, simply hit <NEW-LINE>.
- e) Following the completion of BACKUP, when asked, insert your DATFILE diskette back into drive 0 and hit <NEW LINE>. Remove and store the backup and copied diskettes.
- f) Repeat procedure from step (a) for additional diskettes.

NOTE: The disk drive number can be from 0 to 3, depending upon the number of disk drives you are using. If you only have one disk drive you can still perform a BACKUP by alternating between the source and destination diskettes.

APPENDIX A: EXAMPLE DATA FORMS.

As previously indicated there are virtually limitless uses to which a program like DATFILE can be put. Here are but a few examples that we feel are a reasonable cross section of possible applications of DATFILE. These are in the same format as would appear on your video monitor when in <E>dit/view mode.

Example 1.

SYSTEM-80 USERS GROUP MAILING LIST

NAME:.....
ADDRESS:.....
PHONE (HOME):.....
PHONE (BUS):.....
SYSTEM PARTICULARS:.....
:.....
SOFTWARE INTERESTS:.....

Example 2.

ACCOUNT CUSTOMER LISTING

COMPANY NAME:.....
ACCOUNT NUMBER:.....
ADDRESS:.....
ACCOUNT DETAILS:.....
:.....
ACCOUNT BALANCE:.....

Example 3.

BASIC STOCK CONTROL

CATALOG NUMBER:.....
DESCRIPTION:.....
QUANTITY ON HAND:.....
COST PRICE:.....
RETAIL PRICE:.....
BULK PRICE:.....
SUPPLIER:.....
REORDER LEVEL:.....

Example 4.

AMATEUR RADIO LOG

NAME:.....
CALL SIGN:.....
DATE:.....
POWER:.....
EQUIP:.....
REL:..
MODE:.....
SIGNAL S/R:.....
FREQ MHZ:.....
TIME:.....
COMMENTS:.....

APPENDIX B: EXAMPLE PRINT FORMATS

< DATAFILE-PRINTING >



SYSTEM-80 USERS GROUP MAILING LIST

NAME ADDRESS PHONE (HOME) PHONE (BUSINESS)

< DATAFILE-PRINTING >



BASIC STOCK CONTROL

CAT NUMBER DESCRIPTION QUANTITY COST SELL SUPPLIER

< DATFILE-PRINTING >

10

20

30

40

50

60

70

AMATEUR RADIO LOG

NAME CALLSIGN DATE POWER EQUIPMENT DIST MODE FREQ MHz

< DATFILE-PRINTING >

10

20

30

40

50

60

70

ACCOUNT CUSTOMER LISTING

COMPANY NAME ACCOUNT NUMBER ADDRESS DETAILS

APPENDIX C DISK BASIC AND DATFILE

As already stated, DATFILE produces file data which can be read using DISK BASIC. Because DATFILE has been written as a general purpose filing system no arithmetic functions have been included. Using DISK BASIC however, programs could be written to produce cost analysis figures, graphs of sales figures, plots of occurrences of particular field data and many other analytical functions.

At this point a few comments could be made concerning data produced by DATFILE. DATFILE always produces data in a file DATINF where each record is of the same size independent of the amount of field data which may have been entered. The record size is the sum of the maximum field data size for each field in your form plus one since DATFILE always terminates a record with a carriage return (0DH). Deleted records are filled with 0C0H (192 decimal).

EXTRACTING FILE CONTROL DATA

In order to be able to create file data that can be read by DISK BASIC DATFILE automatically produces a file with the name DATFLD. This file contains control data necessary to decypher the disposition of data in your records.

The following is a DISK BASIC program which may be used to extract record disposition data from the file DATFLD:-

```
10 DEFINT A-Z
20 DIM ND$(32),CP(32),LD(32)
30 OPEN "R",1,"DATFLD:0"
40 FIELD 1,2 AS P1$,2 AS P2$,1 AS P3$,2 AS P4$,
   1 AS P5$,1 AS P6$,2 AS P7$(0),2 AS P7(1),
   2 AS P7$(2),2 AS P7$(3)
50 GET 1,1
60 TR=CVI(P1$):MR=CVI(P2$):ND=ASC(P3$):DR=CVI(P4$)
70 LR=ASC(P5$):NF=ASC(P6$)
80 FOR I=0 TO 3
90 NR(I)=CVI(P7$(I))
100 NEXT I
110 FOR I=0 TO NF-1
120 RS=NF-(16*INT(NF/16))
130 FIELD 1,(RS*16) AS DUMMY,13 AS P1$,2 AS P2$,1 AS P3$
140 GET 1,(2+INT(NF/16))
150 ND$(I+1)=P1$:CP(I+1)=CVI(P2$):LD(I+1)=ASC(P3$)
160 NEXT I
170 CLOSE
```

Program statements of this form should be found in any program which will access data produced by DATFILE. Following is a description of the terms and information available from this program.

TR	INTEGER. Contains the total number of records on all disk drives.
MR	INTEGER. Contains the maximum number of records allowed by DATFILE.
ND	INTEGER. Contains the highest disk drive number being used to store record data (0 to 3).
DR	INTEGER. Contains the total number of deleted records on all disk drives.

LR INTEGER. Contains the total record length of each record produced by DATFILE including the final carriage return.

NF INTEGER. Contains the total number of field names in each record.

NR(i) INTEGER ARRAY. Contains the highest record number stored on disk drive 1. Record numbers start at zero.

ND\$(i) STRING ARRAY. Contains the name of the field with data field.

CP(i) INTEGER ARRAY. Contains the PRINTQ position plus 15360, where the ith data field starts on the video monitor when viewing the data form.

LD(i) INTEGER ARRAY. Contains the number of field data characters in the ith field.

SEQUENTIAL DATA ACCESS

This is the simplest of all data access methods. Since all records by DATFILE are terminated by a carriage return, record data can easily be read by a LINEINPUT # statement.

The following program will read in the first record in DATFILE into string RD\$:—

```
10 OPEN "I",1,"DATINF:0"  
20 LINEINUT #1,RD$
```

Information available from control file can then be used to extract unique data fields from the string RD\$ for manipulation by DISK BASIC.

RANDOM FILE DATA ACCESS

While random file access is more involved than sequential file access, the advantages are well worth the effort. One of the biggest advantages with random access methods is that records may be read in any order on the diskette without the need to read through all the data leading up to a record, as is the case with sequential file handling.

When using random access methods it is advisable to create your data form so that the total number of field data characters in all your fields is one less than a number which evenly divides into 128 (the numbers 3,7,15,31,63 and 127 are possible alternatives). Since DATFILE always terminates a record with a carriage return, the actual record length will in fact divide evenly into physical sectors easily with DISK BASIC.

As an example, a program similar to the one below can be used to read a given record on drive Ø into string RD\$, where N is the record number and LR is the total record length derived from the program in the section 'EXTRACTING FILE CONTROL DATA'.

```
1Ø OPEN "R",1,"DATINF:Ø"  
2Ø RS=256/LR  
3Ø PR=INT((N-1)/4)+1  
4Ø SR=N-1(RS*INT(N/RS))  
5Ø FIELD 1,(LR*SR) AS DUMMY$,LR A RD$  
6Ø GET 1,PR
```

Once again the information extracted from the control file can then be used to manipulate individual field data within the record using DISK BASIC.