Dragon Disk Commands

When writing commands the parameters describing the file are often the same. In this case the most frequently used command format is:

```
COMMAND"D:FILENAME.TYP"
```

where D selects a drive (1-4) and overrides the default drive; FILENAME is up to eight characters that specify the file; and .TYP is the file identifying code, which may be specified by the user, but if not then the DOS will assume the default value: .BAS. This format is represented by FSP.

DRIVE N
In this case N can be 1 to 4. This selects the default drive.

DSKINIT
This command formats the specified disk with:

```
DSKINIT D,S,T
```

in this case D selects the drive; S selects the side of the disk (either 1 or 2) to be formatted, default being 1; T selects the number of tracks to format (40 or 80), default being 40. With a standard single-drive system it is necessary to type only DSKINIT (ENTER) and the disk inserted will be wiped and formatted correctly.

DIR D
This command displays the directory of the specified drive, D, as follows:

```
DIR
PROGNAME .BAS 1654
M/C BINARY .BIN 1389
PROGDATA .DAT 2581
PROGNAME .BAK 1654
167322 FREE BYTES
```

PROGRAM FILES

SAVE
This will store program or binary files to disk. SAVE FSP stores a program file. It is unnecessary to specify .BAS as this is the default value. The command:

```
SAVE FSP,SSSS,EEEE,XXXX
```

stores a binary file. In this case, SSSS is the start address in decimal of the code to be stored; EEEE is the end address, and XXXX is the address the program is executed from.

LOAD
This will read program or binary files from disk with the command:

```
LOAD FSP
```

If the file specified is a binary file, FSP can be supplemented by ,SSSS. This will indicate the new start location in memory of the binary file.

RUN FSP
This will immediately load and run the specified BASIC program.

CHAIN FSP
This will load and run a program without changing previously stored variables. This is especially useful for programs that share data. Adding ,SSSS will have the same effect as for the LOAD command previously described.

FREE D
This displays the number of bytes free on a specified drive

COPY
This will duplicate the file OLDFSB as NEWFSB, as follows:

```
COPY OLDFSB TO NEWFSB
```

If drive numbers are unspecified, the copy is made on the same disk on the default drive.

RENAME
This changes the name, but not the type, of a file. Both FSPs must refer to the same drive or default.

MERGE FSP
This command will superimpose a specified BASIC program file over another held in memory, with the effect that the programs are merged. The program previously held on disk will take precedence if line numbers are duplicated.

KILL FSP
This will delete a specified file from disk.

PROTECT
This command 'software-protects' a file from erasure or overwriting by any command, except DSKINIT, with:

```
PROTECT ON FSP
```

This will also cause a reverse field P to be displayed with the file name on the disk directory. The command PROTECT OFF FSP will remove protection.

BACKUP
This will copy the entire contents of a disk in one drive (DA), onto a disk in another (DB) with the command:

```
BACKUP DA TO DB,S,T
```

S and T take values exactly as for the command DSKINIT, enabling copies to be made on different format disks and drives. The user of single disks can simply type BACKUP (ENTER), whereupon instructions are displayed for alternating source and destination disks.

VERIFY
ON and OFF control this command, which automatically checks that the contents of a stored file are the same as the original.

DATA FILES

FWRITE
This is used to create and write a data file containing variable lists. For each FWRITE, a channel is opened to the specified file, allowing further data to be appended to the data already stored. It is written as:

```
FWRITE"FILENAME";VAR
```

In this case FILENAME is the file to be written, or created and written to, and VAR is a variable list containing the data to be stored. Commas and colons terminate variable strings in data files, unless they are intended to be read by FLREAD as strings. The command:

```
FWRITE"FILENAME",FROM SB,FOR
TB;VAR
```

writes VAR to FILENAME starting at byte SB, extending the length of the list to a total of TB bytes. Only 10 files accessed by FWRITE can be open at one time.

CLOSE D
This closes channels to files opened by FWRITE, FREAD and FLREAD on a specified drive.

CREATE"FILENAME",FL
Creates a data file, FILENAME, of FL bytes in length.

FREAD
This command is constructed as FWRITE. VAR is read into memory or, taking the second example given, VAR is read starting from byte SB. The read pointer is then advanced by SB+TB bytes.

FLREAD
This is constructed in the same way as FREAD, except that commas and colons are not read as terminators.

EOF
This is used to indicate the final valid entry in a file being read. For example:

```
EP=EOF("FILENAME")
```

where EP is zero, until the read pointer reads the final record and changes EP to one.

LOC"FILENAME"
This will display the position of the read pointer as the number of the next byte to be read from a specified file.

SWRITE
This command will store data in S sector on T track as the two strings A$ and B$ to a maximum of 128 bytes each, as follows:

```
SWRITE D,T,S,A$,B$
```

SREAD
This will retrieve data stored by SREAD using the same format. A$ and B$ can be given different variable names.