

Hard Drive Specialist

A Guide to Disk Basic for the Color Computer

Hard Drive Specialist 1984

Credit is due to Bob Rosen in the preparation of this document from his donation of "AN OUTLINE OF DISK BASICS (or what can I do after I put it together)"

For Further Reference See

Radio Shack, TRS-80, Color Computer Disk System, Owners Manual & Programming Guide, a Publication of Radio Shack, a division of Tandy Corporation, Fort Worth, Texas, National Parts # MU2603022

Radio Shack, Color Computer Technical Reference Manual, a Publication of Radio Shack, a division of Tandy Corporation, Fort Worth, Texas, Part #26-3193

Hard Drive Specialist Floppy Disk Manual, a manual on Tandon disk drives from Hard Drive Specialist, Houston, Texas

Installation -

NOTE...To avoid any damage to the Computer System, ALL power is turned OFF while connections are being made.

1. Connect the Disk Interface to the plug in the opening on the side of your computer.
2. Connect the Disk Cable to the Interface and the Drive.
3. Plug in the Drive power cord to a standard electrical outlet.
4. Power up the TV/Monitor, Computer, and Disk Drive.
5. The message "Disk Extended Color Basic" should appear on the screen. If not, turn OFF the power, check your connections and power it up again.
6. To start processing, insert your "applications program" disk or a blank (see DSKINI, all blank diskettes must be formatted before use).

How data is stored ..

A disk is divided up into 35 tracks with 18 sectors in each. You can store 256 bytes of information per sector or 4608 bytes per track.

The unit for space allocation that the Computer uses is a granule. One unit (a granule) or a 2304 byte cluster is the minimum size of a disk file. Therefore, a disk will hold a maximum of 68 disk files or have a total of 68 granules of space available.

The Computer's disk directory is stored on track 17. A file allocation table and file information directory entries are contained in its sectors. If data is written incorrectly to the disk, or the disk drive itself or the Computer is powered OFF and ON with a disk in the drive, the directory track can be damaged and for all practical purposes, the data on the disk lost. In this case, the Computer has lost its pointers to where the information was stored and effectively it's gone.

Common sense disk handling guidelines -

- Store a disk upright in its storage envelope in a vertical file.
- Do not power the System ON or OFF with a disk in the drive.
- Avoid magnetic fields, direct sunlight or excessive heat, and contamination from dust or greasy fingerprints.
- Write on the disk label with a soft felt-tip pen only.
- Always insert a disk with the notch on top.

With prudent care and handling, your Disk System and its diskettes will provide a entire new dimension to your Computer's capabilities and greatly expand its potential applications.

A write protect notch is located on the outside of the diskette, when this notch is covered by a piece of gummed tape then no writing of data can be made to the diskette, only reading.

DISK BASIC COMMAND SUMMARY

Some command parameter definitions :

filename

Mandatory for all information stored on disk. General format :

name/extension : drive number

Where :

name is required. It must have 1 to 8 characters.

extension is optional. It can have 1 to 3 characters.

drive number is optional. Defaults to 0 if not specified (or the drive number setup by the DRIVE command).

number

Can be any number, numeric variable, numeric operation, or a numeric function.

string

Can be characters bounded by double quotes, a string function, string variable, or a string operation.

data

Can be a number or a string.

BACKUP source drive **TO** destination drive

The contents of source drive are duplicated to the destination drive. For a single drive **BACKUP**, only the source drive is specified, and the Computer will issue disk switching prompts.

CLOSE # buffer,...

Terminate communication to the buffers specified. If buffer is omitted, all open files will be closed.

COPY filename 1 **TO** filename 2

Copies the contents of file 1 to filename 2. Each filename must include an extension.

CVN (string variable)

Converts a five byte coded string back to the number it represents. (see MKN&)

DIR drive number

displays directory information of the disk specified by drive number (default 0) in a five column format :

1. name
2. extension
3. file type (0=BASIC program, 1=BASIC data file, 2=machine language file, 3=editor source file)
4. storage format (A=ASCII, B=Binary)
5. file length in granules.

DRIVE drive number

Changes to default (0) drive specification to where the Computer will automatically store or look for files.

DSKINI drive number

Formats a disk in the drive number you specify. All blank disks must be formatted before they can be used to store information, including **BACKUP**.

DSKI\$ drive number, track, sector, string variable 1, string variable 2

Inputs the first 128 bytes into string variable 1, and the second 128 bytes into string variable 2 from a certain sector within a certain track on the disk in drive number.

DSKO\$ drive number, track, sector, string variable 1, string variable 2

Opposite of **DSKI\$**. Data is written to the disk. These two commands bypass the entire disk's filing system and will output data without opening a file or modifying locations in the disk directory. Unless you have researched the technical information concerning track and sector formats, disk directory entries and the granule allocation table, **NO NOT** mess with these commands.

EOF(buffer)

Returns a 0 if more data is to be read in the buffer and a -1 if there is no more data left.

FIELD #buffer, field size **AS** field name, ...

Defines the space within a direct access buffer and organizes it into fields.

FILES buffer number, buffer size

Reserves the specified number of memory buffers of a total number of bytes (size). If **FILES** is not used, 256 bytes for buffers 1 and 2 will be reserved by default.

GET # buffer, record number

Fetches the next record or the record number you specify and places it in the buffer

INPUT # buffer, variable name, ...

Inputs data from the buffer and assigns each item in the buffer to the variable name specified.

KILL filename

Removes filename from the disk directory. An extension must be included with the filename.

LINE INPUT # buffer, data

A complete line (all data up to the -ENTER- character) is inputted, from the buffer.

LOAD filename, R

Loads the BASIC program filename, and by including R, will immediately RUN it. If a file extension is not used, BAS is assumed.

LOADM filename, offset address

Loads a machine language program and optionally adds the offset address to the load address. BIN is the default file extension.

LOC (buffer)

Returns the current record number in the buffer. Use PRINT to display the number.

LOF (buffer)

Returns the highest numbered record of the specified buffer.

LSET field name = data

Left justifies the data within the specified field name. RIGHT characters will be truncated if the data is larger than the field.

MERGE filename, R

Loads and merges the filename with an existing program in memory. With the optional R, execution takes place immediately. The MERGED file must have been SAVE d originally with the A (ASCII) option.

MKN\$ (number)

Converts a number to a five byte coded string for storage in a formatted disk file.

OPEN "mode", # buffer, filename, record length

Reserves a place in memory called a buffer that communicates data to and from a device. The buffer/device designations are

- 0 - screen (not necessary to open)
- 1 - tape recorder
- 2 - printer
- 1-15 - disk drives

The communication modes are :

- I - input from a sequential access file.
- O - output to a sequential access file.
- D - perform I/O to a direct access file.

If the mode is for direct access, a record length can be specified. If not the default size will be 256 bytes.

PRINT # buffer, data list

PRINTs the data (outputs it) to the buffer. Commas or semi-colons are used to delimit each item in the list.

PRINT # buffer, **USING** format; data list

Outputs the data to the buffer using the specified format string.

PUT # buffer, record number

Assigns a record number to the data in the buffer. If a record number is not used, it will be assigned to the current record.

RENAME old filename **TO** new filename

Changes the old filename to the new filename, Extensions must be specified for both filenames.

RSET field name = data

Right justifies the data within the field name specified. Truncation occurs under the same conditions as LSET.

RUN filename, R

Loads filename from disk and runs it. If the R option is included, all open files remain open.

SAVE filename, A

Saves filename on the disk. By using the A option, the file will be saved in ASCII format.

SAVEM filename, start address, ending address, execution address

Saves a machine language program from memory location start address through location ending address. Execution address is the entry point where processing begins. The default extension is BIN.

UNLOAD drive number

Closes any open files on the specified drive number.

VERIFY ON

VERIFY OFF

The Computer will double check or verify all disk writes.

WRITE # buffer, data list

Writes the comma delimited data list to the specified buffer.

ERROR MESSAGES

- IO** Division by zero. An impossible divide by zero was attempted or a filename was not enclosed within Quotation marks.
- AE** File Already Exists. The object of a COPY or RENAME command was to a filename which already exists.
- AO** Already Open. An attempt to OPEN a file which is already opened was made.
- BR** Bad Record Number. The GET or PUT record number is too small (less than 1) or too big (exceeds the maximum number that can fit on the disk).
- DF** Disk is Full. There is no more space to store your file. Free up disk space or use another disk.
- DN** Drive Number or Device Number error. You are trying to access a disk you don't have to use more buffers than have been reserved.
- DS** Direct Statement. An attempt to LOAD a program or data file with no line numbers.
- ER** Write or Input past End of Record. A direct access I/O operation has attempted to write more data than a record can hold or input more data than was there.
- FD** Bad File Data. Variable type conflict. eg. String data was read into a numeric variable.
- FM** Bad File Mode. An incorrect specification ("O", "I", or "D") for the operation being performed was made. eg. Input from a file that was opened for output.
- FN** Bad File Name. The name may have no more than eight characters.
- FO** Field Overflow. The field length is longer than the record length.
- FS** Bad file Structure. Data was written incorrectly to the disk or the directory track has been clobbered.
- IE** Input past End of file. An attempt to read additional data was made after reaching the end of the file.
- IO** Input/Output error. A general classification of trouble when inputting or outputting information to the disk.
- NE** The specified disk file cannot be found. Check the file mode or name spelling.
- NO** File Not Open. A file must be OPENed before you can input or output to it.
- OB** Out of Buffer space. Not enough buffer space was reserved, uses FILES.
- SE** An attempt to LSET or RSET data to a field that has not been FILEDed was made.
- VF** Verification. During a write operation to disk with VERIFY ON an error was detected.
- WP** Write Protected. An attempt to put information on a disk with the write protect notch covered was made.

STAKING PINS

The HDS board has three "flea clips" which fit over staking pins. These flea clips perform two functions. The first function is to choose which ROM is to be selected. The second function is to configure the 28 pin socket to accept either a 28 pin ROM or a 24 pin ROM. The A-B-C pins select the ROM's while the D-E-F-G-H pins configure the 28 pin ROM socket. To select ROM A (the 24 pin socket labeled Z6), put the flea clip over pins A-B. To select ROM B (the 28 pin socket labeled Z7), put the flea clip over pins B-C. To configure the ROM B socket to accept a 28 pin ROM (2764's), short pins E-f and G-H with the flea clips. To configure the ROM B socket to accept a 24 pin ROM (68364's), short pins D-E and F-G with the flea clips. Also note that when using a 24 pin ROM in the 28 pin socket, pin 1 of the ROM goes into pin 3 of the socket. See figure 1.2.

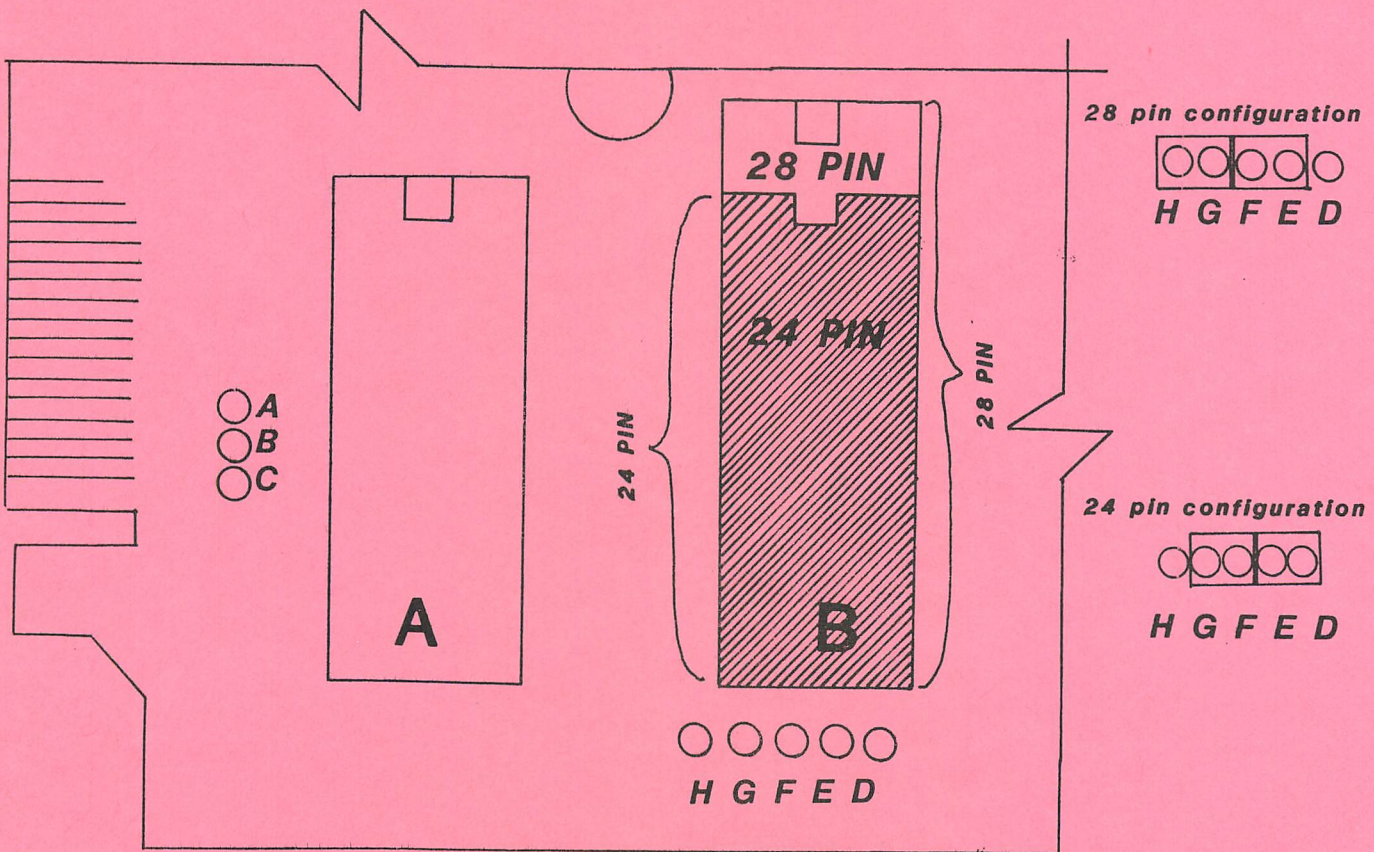


figure 1.2