PEEK A BYTE 64 Tip Sheet #1

Thank You for purchasing **Peek A Byte** 64. We believe that it is the finest disk and memory utility available for the Commodore 64 computer. With this utility you can read or write data from any sector on a disk, edit the data, change a program name, disk ID or name, correct a program or directory, read the 64 memory, modify values in the BASIC memory, etc. The data can be displayed as HEX values, screen or PETASCII characters, or disassembled to machine code.

Peek A Byte also will do a fast compare of disk sectors, read the disk drive memory, convert between HEX and decimal, and recover or un-new Basic programs. A monitor program or user supplied machine language routine may be exectuted from **Peek A Byte**. This feature is unique and allows for additional versatility and future expansion capability. Compatability with the DOS Wedge 5.1 is maintained, allowing it to be used after exiting **Peek A Byte**. Many Basic programs can also be run while **Peek A Byte** is resident in memory. This Tip Sheet suggests additional uses for **Peek A Byte**.

If you have comments, suggested improvements, or routines to use with **Peek A Byte** 64, please mail them to us. We appreciate hearing from our customers.

1) Make a whole disk backup of the **Peek A Byte** 64 disk and put the original in a safe place.

2) The repeat key flag can be set by editing location $28A:

```
HEX:        BASIC:
#$80        POKE 650,128 : REM All keys to repeat
#$40        POKE 650,64  : REM No keys to repeat
#$00        POKE 650,0   : REM Default keys to repeat
```

3) Changing the drive number will initialize the drive to the BAM of the disk in that drive of a dual disk drive. If the drive or disk is not present, an error will result. Changing the drive to 1 and back to 0 will initialize a disk in the 1541 drive.

4) The 1526 printer can be used with the normal printer output if the following patches are made to **Peek A Byte** 64 after loading the program:

```
POKE 50604,32 : POKE 50605,0
POKE 50620,32 : POKE 50621,0
```

Or pick a character other than a space, which is CHR$(32).

5) Use **Peek A Byte** 64 to examine BASIC programs. Look for the byte CC just before the byte 00. It is often used at the end of a REM statement to produce a SYNTAX ERROR when listing a program.

QUANTUM SOFTWARE, P.O. Box 12716, Lake Park, FL 33403

6) The DOS 5.1 wedge is now included on the Peek A Byte disk, thanks to permission from Commodore Electronics, Ltd. The program "DOS LOADER" is a BASIC program that may be run after loading Peek A Byte. DOS LOADER will load DOS 5.1, activate the wedge, and store it in the RAM memory under the BASIC ROM. It is not necessary to use the procedure in the manual on page 6.

7) A single drive whole disk copy program, "DISK COPIER", is now included for making backups of normal DOS disks. If any read errors occur it will BEEP, report the error, and procede after the "C" key is pressed. DISK COPIER will not reproduce the error, however. The built in format (or NEW) command uses an ID of "QS". Format the disk separately if a different ID is desired.

8) Use the following BASIC program to stop the head banging resulting from a drive read error:

```
10 open15,8,15
20 print#15,"m-w"chr$(106)chr$(0)chr$(1)chr$(133)
30 close15
```

To reset the drive, use the wedge command "@ui" or replace 133 by 5 (the number of read attempts) in line 20 of the BASIC program and then run it.

9) Occasionally one of two errors occur when attempting to read a disk sector using Peek A Byte. These errors typically occur if the drive has been used extensively for file manipulation, such as scratching a file or validating a disk. The drive will need to be reset from BASIC - the DOS wedge commands are easiest:

a) 70, no channel, 00, 00

This error results if the BAM (or other channel) has been assigned to drive buffer #0. Use the wedge command "@ui" to reset the drive channels.

b) 29, disk id mismatch, 00, 00

This error results if the disk has been changed and the BAM has not been updated in the disk drive. Use the wedge command "@i0" to initialize or read in the BAM. This error may also be due to a copy protected track on the disk.

10) Remember that Peek A Byte can be exited to BASIC using "Q" <RETURN>. To restart Peek A Byte use either:

```
BASIC:      SYS 49152
Monitor:    .G C000
```

11) Peek A Byte can be used with a monitor program without losing disk sector data stored in the computer memory. Use your favorite monitor to assemble programs, then restart Peek A Byte and write the data to the disk.

Peek A Byte 64 V2.0 Manual

I. EQUIPMENT

Peek A Byte, V2.0 with the Disk Mechanic is designed to work with one or two 1541 disk drives. If the Disk Mechanic detects a non-1541 drive, it will return the user to BASIC. A 1541 "clone" may work, but none have been tested. Peek A Byte, V1.3 will work with any compatible serial bus disk drive, but only works with normal 1541 format type disks.

A printer is optional for making printed copies of the Peek A Byte, V2.0 screen displays. Any serial bus printer which is Commodore compatible should work with Peek A Byte, V2.0. See the original manual for details.

II. LOADING PEEK A BYTE, V2.0

Load Peek A Byte, V1.3 first. (It MUST be loaded first or the program will not work). It will load with a light blue border.

```
LOAD "PEEK A BYTE",8,1      <RETURN>
```

After loading the original program, then load the Disk Mechanic. It will load with a red border.

```
LOAD "MECHANIC V2",8,1      <RETURN>
```

This program links with Peek A Byte V1.3 to create Peek A Byte, V2.0. The program will only load correctly on disk drive 8, but will operate after loading on 8 or 9.

III. CHANGES TO PEEK A BYTE, V1.3

The operation of Peek A Byte, V2.0 is essentially unchanged from V1.3, but has been enhanced by some additions and improvements. These are listed below:

Additions

1) The Disk Mechanic's Toolkit - Press <f1>.
2) The Nibbler! reads raw GCR disk bytes including sync and gap bytes - press <N> <RETURN>.
3) A Sector GCR editor allows you to read, edit, and write the 325 GCR disk bytes (without decoding first).
4) Fast disk Format or Erase. Format with illegal disk headers or half tracks. The format is automatically verified.
5) Track Analyzer checks and lists legal and illegal track headers and stores them for duplicating a track format.

Improvements:

1) Read and write upto track 40 (\$28 in HEX) - also half tracks. (The heads in some 1541 drives can't go out to track 40). Data on sectors with illegal track headers can also be read and sometimes rewritten.
2) Head doesn't bang for most read/write functions. However, the drive routines cannot check if the correct track is being read since illegal headers could be used.
3) The <f7> key now aborts all commands as does the <STOP> key.
4) If a memory buffer is illegal for reading or editing, a printed message will be displayed.
5) Most parameters (ie. unit or device numbers, buffer, track, and sector) are saved when temporarily leaving the program. This feature can be turned off.
6) The screen at \$400 can be printed from BASIC using SYS 49158. A Device Not Present error will not hang the computer.

Changes

These additions and improvements come with a few changes which should be noted.

1) V2.0 will only work with a 1541 or identical "clone". Use V1.3 with other drives.
2) The drive number is always 0 - it can't be changed for dual drives (since the 1541 is not a dual drive.)
3) Track header errors such as 27, header checksum error, and 29, track ID mismatch, are always reported as 20, header not found. The details of the header error can be found using the track header analyzer.
4) The Disk Mechanic routines use the area of memory between \$6E00 and \$7FFF. This steals memory from BASIC, but allows the use of monitor programs which load between \$8000 and \$9FFF. Some cartridges may work if they DO NOT affect memory from \$A000 to \$FFFF. We do not have a list of cartridges which will work with Peek A Byte - you will have to try it first. Please write us with your results.

IV. DISK MECHANIC'S TOOLKIT

Most of the additional features of V2.0 are accessed from the Toolkit. The menu for the Toolkit appears when <f1> is pressed from the command mode, and appears as shown in Figure 1. To select the desired function, press the key shown and <RETURN>. Press any other key to correct the entry. <STOP> or <f7> will return you to the command mode. Several of the functions toggle (switch on or off) program modes. We'll go through each function in order.

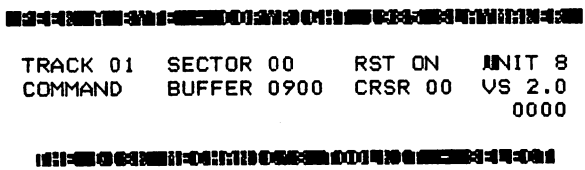

Figure 1. Disk Mechanic Toolkit Menu

I) Initialize Drive

This function reads the BAM (block allocation map) of the disk into the disk drive memory and sets the track and read/write head position. Press <I> and <RETURN>. The screen will clear and prompt for another <RETURN>. Each time a disk is changed, the drive should be initialized to be sure that the track position is set in the drive. This is important because different disks may have been formatted either 1/2 or 1 track off the correct position. This function is not done automatically because it can cause head banging if the the disk has many read errors or if the head is left in the wrong position.

Occasionally, if the drive has been turned off and then on, or if the drive has been reset, a DRIVE NOT READY error message may occur. This happens because the drive program doesn't know where the head is positioned. Try initialzing the drive to clear this error.

T) Tracks Half or Full ... Toggles

This function toggles between full and half track positions. Press <T> and <RETURN>. When half track is in effect, TRK:HALF is shown in reverse on the message line. It affects normal READ, WRITE, Sector GCR R/W, Analyze, and Nibbler! functions. It does not affect Format or Verify functions. The head is moved to the track shown, stepped an additional 1/2 track, the read or write occurs, and the head is stepped back 1/2 track.

H) Headers GCR or Normal .. Toggles

This function toggles between using the normal track headers and the GCR header values stored by A) Analyze. Press <H> and <RETURN>. HDR:GCR is shown in reverse when this mode is in effect. It affects the same functions as T) Tracks. The displayed sector number is used as a position count (under #) to the correct header in the Track Headers display.

S) Sectors GCR or Normal .. Toggles

This command turns the Sector GCR read/write mode on or off. Press <S> and <RETURN>. When it is off, the sector read/write behaves normally. When it is on, SECTOR GCR EDT appears in the message line, and the 325 disk GCR bytes recorded on the disk for the sector data are shown in two buffers without decoding or encoding. The buffer must be set on an even buffer number to READ or WRITE. The buffer editor works normally, so GCR bytes may be changed to produce or remove 22 or 23 errors, if desired. Commands which change the sector and buffer together are turned off when SECTOR GCR EDT is in effect.

A) Analyze Track Headers

This function will analyze a track for valid track headers. A) Analyze should be used to check that the drive head is on the correct track before writing to a disk. Press <A> and <RETURN>. A prompt will appear to insert a disk and a flashing cursor will appear at the track. A new track value may be entered in HEX if desired. Press <RETURN> to accept the track value shown and analyze the disk track. The header values are stored in the computer and then displayed as shown in Figure 2. See D) Display.

Headers are identified by the presence of sync bytes just before the header GCR values and sync bytes less than a sector length after the header values. Headers for some protection techniques (which do not have sector sync bytes in the normal locations) will not be identified.

D) Display Analyzed Headers

Press <D> and <RETURN>. This will display track headers which have already been stored in the computer without having to reanalyze the track. The stored headers are used in HDR:GCR mode and for duplicating a track format (see F) Format). The screen may be printed by pressing <ctrl P>.

A typical display is shown in Figure 2. The left hand side of the display shows (all values in HEX) the header position, #; the sector number, SR; the track, TK; disk ID 1, I1; disk ID 2, I2; the correct header checksum, CL; and the recorded checksum, CK. The upper right hand corner shows the physical track and ID for the disk. The right side of the display shows the first eight GCR bytes of the header. The last two bytes of the header and the eight bytes in the gap are all 55's on a normal 1541 disk and are used only as a spacer. Some protection schemes may change the value or number of bytes. The Disk Mechanic does not store or use these bytes.

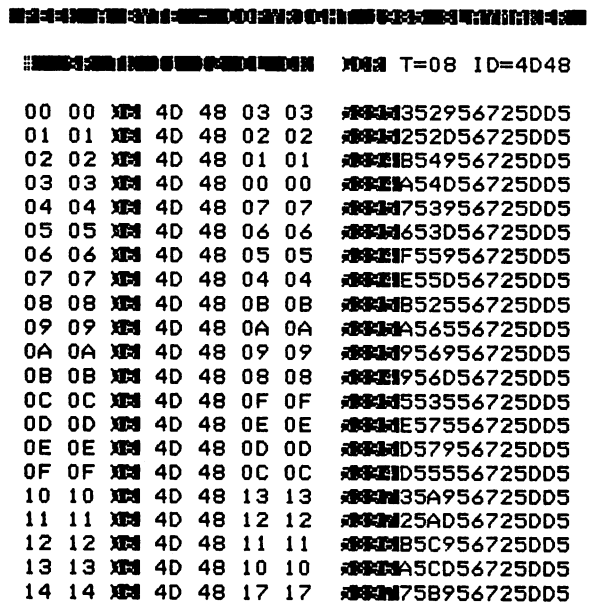

Figure 2 Track Header Display

Disk Format and Header Errors

Errors in the headers for the track are shown in reverse, instead of simply giving an error number. Many protection schemes require that the headers match, not simply have the same error number. A brief explanation of disk errors will help you interpret the displayed header information.

20: READ ERROR (header for sector not found)

The header for the specified track and sector was not found. Normal DOS could not find a $52 byte after sync bytes, or could not find the sector specified. In this program, disk errors 27 and 29 will be reported as error 20 since the entire header must match. The lead byte of $52 in GCR corresponds to a $08 in the Format routines and can be changed. If this byte is changed, either the first or second GCR header byte will be illegal and will be shown in reverse when analyzed.

21: READ ERROR (no sync bytes found)

Normal DOS could not find sync bytes within the 0.02 second time limit (about two sectors on the disk). In this program this error occurs if no sync bytes were found on the track, or if in the A) Analyze routine no valid headers were found. Sync bytes are a string of $FF's which appear just before header and sector data in order to identify and synchronize the drive with the disk data.

22: READ ERROR (sector data not found)

The 325 sector GCR bytes are read in and decoded. The first byte is compared to the value in memory $47, which is normally $07. The first GCR byte is normally $55. If the comparison fails, a 22 error occurs. The Sector GCR editor can be used to either duplicate a 22 error or to produce one by changing the first byte. A 22 error can be eliminated by reading a sector in normal mode and then writing it back after checking the data for errors.

23: READ ERROR (sector data checksum incorrect)

The sector data checksum is incorrect. The correct checksum will be recorded if a sector is read, checked for data errors, and then rewritten in normal read/write mode. An incorrect checksum can be duplicated using the sector GCR editor to read the sector from one disk and to then write it to the other disk. A 23 error can usually be created by changing byte $42 of the second buffer in the GCR mode after reading the sector and and before rewriting it.

25: WRITE ERROR (sector data mismatch)

The sector GCR data written to the disk does not match the data that was written to the disk. This may occur due to hardware problems, if illegal GCR bytes were written to the disk, or if the gap between the header and the sector sync bytes was less than the normal eight bytes. A double set of sync bytes would prevent reading the data correctly. For this reason, NEVER write to an original disk - you may destroy data on it.

26: WRITE PROTECT ON

The write protect tab should not transmit light - do not use tape. Remove the write protect tab if you wish to write to the disk. All Peek A Byte disk write or format routines check for the write protect tab.

27: READ ERROR (header checksum error)

The header checksum is incorrect. This program reports a 27 error as a 20 error. The correct and recorded checksums are displayed by the A) Analyze function. An offset value or difference can be added to the correct checksum in the F) Format function.

28: WRITE ERROR (long sector)

The sector data overwrote the next sector's sync bytes. This error check is not implemented on the 1541.

29: READ ERROR (ID mismatch)

One of the IDs on the track read does not match those of the BAM track (stored at $12 and $13 in the drive). This error is reported as a 20 error in this program. The IDs are displayed by the A) Analyze function in reverse if they are incorrect. A track can be reformatted with a different ID in the F) Format function.

F) Format or Erase Tracks

The format or erase commands allow you to either format or erase a range of tracks. Press <F> for Format or <E> for Erase. <STOP> or <f7> will abort the command before formatting has begun - after that the drive must be turned off or reset to abort. If erase is selected, all information on the specified tracks will be erased. If the disk to be formatted or erased has data on it, the disk should be initialized first using I) Initialize Drive. This sets the drive head position. A previously formatted disk should be used to initialize the drive if a blank disk is to be formatted.

The starting and ending track numbers should be entered in HEX when prompted by the blinking cursor. Digits may be reentered if necessary and <RETURN> pressed when correct. Normally a track range of $1 to $23 (1 to 35, decimal) is chosen for a full disk format or erase. A maximum track of $28 (40 decimal) may be selected.

WARNING! Some drives cannot move the head out to track 40, decimal. The head may be left in the wrong position if it is attempted. Most commercial software only occasionally uses track 36 ($24) or 37 ($25). Very little software uses track 38 ($26) or higher. The normal 1541 DOS cannot access these tracks either unless the ROMs are alterred.

After the tracks have been entered, the screen will prompt for the two disk ID values if formatting. Enter these values in HEX - use the HEX conversion tables in the back of the Peek A Byte manual to find the equivalent HEX value for the normal ASCII. HEX allows any of the possible values to be entered, not just ASCII keys. Press <RETURN> to format after the prompt to insert the disk appears.

If a range of tracks is selected which does NOT include track $12 (track 18, the BAM and directory track), then you can choose half track (answer Y or N) if desired. In addition, the header start byte (normally $08), the first header track value (normally the same as the physical track), and the checksum offset (normally 0) can all be entered. Press <RETURN> when the value shown is correct.

If a single track is selected, then you have the option of formatting the track with the header values stored in the computer by the A) Analyze function. You must have analyzed a track first - use the D) Display function to be sure. Answer Y or N. Advanced users who wish to create their own formats can transfer 21 headers, each 8 bytes long, to the memory under the BASIC ROM starting at $A002. Use D) Display to be sure the GCR values are correct (the sector, track, etc. won't be).

P) Parameter Restore ..... Toggles

Parameters such as the track, sector, unit, buffer, and screen colors are saved when exiting the program temporarily if RST ON is displayed. <P> and <RETURN> toggles this mode on and off. If this mode is off, all values are reset to default upon restarting Peek A Byte.

V. NIBBLER! Press <N> and <RETURN> from command mode

The Nibbler! is a new function in the command mode which reads "raw" GCR and sync bytes off the disk from the track displayed. It will read 5 memory pages of data into the disk drive memory and display it in 5 buffers in the computer starting with the buffer displayed. It will start reading at the first sync byte encountered. If none are present, an arbitrary starting point will be selected. Extra sync bytes are displayed to indicate the beginning of the data.

VI. COLORS <f2> <f4> <f6>

The screen border, background, and character colors can be changed by pressing these keys in the command mode. These functions are in Peek A Byte V1.3, but were not documented in the manual.

VII. MEMORY USAGE

The Disk Mechanic uses additional memory from $6E00 to $7FFF. All the BASIC memory between $900 and $6DFF is available as storage buffers for disk sector data. The buffer displayed is the page of memory in the computer with the same number. You are responsible for protecting your BASIC programs from overwriting by Peek A Byte by not using buffers which contain your program. You may edit your BASIC program tokens and line links using the EDIT function.

The highest memory which can be edited or used as a disk data buffer is determined by the page number stored in MEMSIZ at $37 and $38. These values and FRETOP at $33 and $34 are set by Peek A Byte after checking MEMSIZ at $283 and $284. Both values of MEMSIZ are set to the smaller of the values in $283 and $284 and the address value $6E00. You may lower the top of memory and protect it from Peek A Byte by setting $283 and $284 and restarting Peek A Byte.

If you wish to edit buffers which are normally protected, place the value $FF at memory location $2E4, the edit flag. These buffers are protected to prevent your "bombing" the program. You edit these buffers at your own risk.

VII. HALF TRACKS, GAPS, DENSITY, AND OTHER WARNINGS

Peek A Byte 64 with the Disk Mechanic will read and write half tracks. That's the good news. The bad news is that the 1541 drive writes a guard band to the side of each track. Therefore, data cannot be written on both a track and its adjacent half track using a 1541 - if you try it you will either erase some data or garbage the adjacent half track. In addition, the 1541 head alignment can easily be off by 1/4 track - this may be OK for normal tracks, but not half tracks. Some professional drives can write half tracks and are used to produce commercially protected software - but not the 1541.

The Disk Mechanic can also read data off tracks with some types of header errors (such as illegal track values) without headbanging. The penalty is that there is no way to check whether the head is correctly positioned when writing, and especially when formatting single tracks. Therefore, you should:

ALWAYS INITIALIZE THE DRIVE AND CHECK THE TRACK FIRST

Use the A) Analyze function or read a sector first to be sure you are on the correct track.

The Disk Mechanic WRITE and FORMAT routines assume normal track density and a normal gap of 8 bytes between the end of the track header and the sync bytes marking the beginning of the sector or block data. Some protected disks change the number and value of the bytes in this gap. For this reason (and others)

NEVER WRITE TO OR FORMAT AN ORIGINAL DISK !!!

If you must write to a protected disk, check the formatting first using the NIBBLER!

If you have suggestions for improving Peek A Byte and how to do it, please write. We'll work on a version 3.0 if there is enough demand to make it a commercial success. Please help us out by telling your friends and club members to buy Peek A Byte (if you like the program - tell us if you don't). We would like to keep Peek A Byte unprotected and with the links to BASIC and machine language monitors that make it one of the most useful and powerful disk and memory utilities for the Commodore 64 and disk drive.

COPYRIGHT NOTICE

LIMITATION ON WARRENTIES AND LIABILITY

Quantum Software and the program author have endeavored to ensure that this program works as specified in these manuals. Nevertheless, there may be errors in the program or its documentation. We would appreciate receiving notice of any errors you may find.

The original disk is not copy protected so that the original purchaser may make a backup copy and put the original disk in a safe place. However, if the original program diskette becomes unreadable, Quantum Software will replace it upon return of the defective diskette within 30 days of the date of delivery. After 30 days there will be a $10.00 service charge for disk replacement.

Quantum Software and the program author shall have no liability or responsibility to the purchaser or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by this software, including, but not limited to any interruption of service, loss of business or anticipatory profits or consequential damages resulting from the use or operation of this software. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you.