Hex Edit Help Index

How To ...

Commands

TODO

- add help for Message Boxes
- add SF1 help for all keys
- Review keywords
- After changes search for “implemented” and “TBD”

Quick Start

HexEdit is a binary file editor that ostensibly looks like many other such programs. However, it has many useful features not found in many, or even any, other editors. This is a brief introduction to some of its more interesting and useful facilities.

If you are familiar with binary files and at least one Windows word-processor or editor then you can probably use most of HexEdit's features straight away, but you may still like to read this section to get the most out of the program. If you can't even be bothered to read this then just leave the "Tip of the Day" feature enabled until you start seeing the tips repeated.

If you are **not** familiar with running Windows software and/or this type of program you should read the help topics covered in the Overview of HexEdit rather than this concise guide. If you are just unfamiliar with binary files then you may just need to read the next section (Overview of Binary Files) as well as this section.

If you use HexEdit only occasionally you will find its extensive on-line help very useful. You can usually find out about something you don't understand by pressing F1. Pressing Shift+F1 enters help mode whence you may select a menu item or click on a button or other object you want help on. Dialogs also provide popup help accessed through the F1 key or the "?" button on their title bar.

Tool Bars

The Toolbar (as opposed to the other "toolbar" called the "Edit Bar") has several buttons for changing how the display looks. A useful one is the Autofit button, which allows you to easily adjust the number of columns by dragging the sides of the window. The EBCDIC button toggles between display of ASCII and EBCDIC characters – this also affects how string searches are done, and how characters are copied to the clipboard. The other buttons are also useful – try them. Other options, such as the number of columns, and column grouping are found in the "Window Display" page of the Options Dialog.

The Edit Bar is the other toolbar. It has two similar tools (the Hex and Decimal Address Tools) that allow you to quickly jump to an address. The Find Tool allows you to search for a sequence of bytes. You can enter the bytes in hex or as a string. To do a string search type "=" at the start of the string. For a case-insensitive string search use "~" instead. When EBCDIC mode is on the string searches use EBCDIC characters. More options (including Unicode searches) are available with the Find Dialog (Ctrl+F).

Status Bar

The left side of the Status Bar is the message area. It displays a short description of menu items and toolbar buttons as you move the mouse over them. The message area also sometimes displays status messages while a process is executing or indicates an unusual situation that is not serious enough to warrant displaying a message dialog.

To the right of the message area, the status bar is divided into several segments or *indicators*. The first indicator displays information about the current byte at the cursor (the hex, decimal, octal, and binary values and the ASCII or EBCDIC character). To get even more information double-click this indicator and the Properties Dialog will appear. (The Properties Dialog is a "modeless" Dialog, so you can leave it open to view the properties of the data as you move the cursor within a file.)

There are two "Distance to Mark" indicators showing the number of bytes between the mark and the cursor, in hex and decimal. The mark can be positioned anywhere in a file by double clicking on a byte.

The RO/RW indicator shows if the window is read-only. You can toggle its state by double-clicking it (unless the *file* attached to the window is opened read-only). The OVR/INS indicator indicates if the window is in overtype or Insert mode. The REC indicator shows when macro recording is on – double-

click this indicator to start/stop recording.

Editing

HexEdit supports the typical ways of selecting bytes using the mouse or the shifted cursor keys plus cutting, copying, pasting, deleting etc. Once selected you can perform many operations such as writing the bytes to a file etc. A useful option available from the Print Dialog box is to print the selection (particularly as Windows 95 does not seem to allow page numbers greater than about 30000).

You can also copy to the clipboard as hex text or even as C/C++ source code, which is useful for initialising an array of bytes in your source code. The PasteAs ASCII/Unicode/EBCDIC commands allow you to perform such tasks as converting a block of text from EBCDIC to ASCII, removing invalid text bytes, or taking (ASCII) text from another program and pasting it as EBCDIC etc.

HexEdit has a two-level unlimited undo. Undo information is stored for cursor movement, changes to the display etc for each window. At the file level changes made to the file are also tracked. Changes to the file made in one window affect all windows open on that file. If you undo a change to the file all window level changes made in other windows of the file are undone back to when the file was changed – this is necessary to keep things consistent.

HexEdit does not store the bytes of the file in memory, it only keeps track of the changes made to the file (which are also used to undo changes). This means you can easily edit very large files. A consequence is that when you write a file to disk all the undo information for the file is deleted. Moreover, if you insert or delete bytes in a large file HexEdit has to copy the file to a temporary file, before deleting the original. This can be slow and perhaps cause you to run out of disk space. One way to avoid this is not to insert or delete bytes in a large file (and not to create a backup copy) – the easiest way to make sure of this is always to work in overtype, not insert, mode.

Tools

Quick searches can be performed with the Find Tool on the Edit Bar, although the (modeless) Find Dialog provides more options. Searches are done with the Boyer-Moore algorithm which, apart from being very fast, has the peculiarity that the longer the string of (unique) characters you are searching for the faster the search, although the limiting factor is likely to be the speed of your disk or network.

You can compare the data in any two (maximised) windows using Alt+C, even two parts of the same file. If a difference is found the offending byte is selected in both windows – pressing Alt+C again will continue the comparison at the following byte.

Useful Keys

There are many keyboard shortcuts for menu commands. You can see what these are in the menus themselves. But there are a few keys that do not correspond with any menu commands.

Ctrl+L	Centre cursor within the window, and redraw the window
Ctrl+U	Scroll the window up
Ctrl+D	Scroll the window down
Tab	Swap between hex and character areas
Enter	Change the offset so the current column is at the left of display.
Ctrl+F3	Search forward, using the currently selected bytes
F4	Start case-insensitive string search in the Find Tool
F5	Start case-sensitive string search in the Find Tool
F6	Start hex data search in the Find Tool
Alt+G	Jump to Hex Address Tool
Ctrl+Shift+F9	Select from the cursor to the mark

Overview of Binary Files

All information in a computer is stored as bits, which can have a value of on or off (true or false, 0 or 1). Nowadays, almost all computers and peripherals (for many reasons) are based on storing bits in "bytes" or chunks of 8 bits. Admittedly many processors transfer data to and from memory in chunks of 16, 32 and even 64 bits, but they invariably have instructions that operate on a byte even if they have to take an extra 8, 24 or 56 bits along for the ride.

Further, storage and transmission of data (except at the lowest levels of software and hardware) always deals in bytes. Files on disk or tape are outwardly stored as zero or more bytes (although at the physical level they may be stored as very large chunks of bits or bytes).

Although some (peculiar) people have been known to refer to anything from a chunk of 6 bits to a word of 16 bits as a byte, I will only refer to the more conventional 8-bit byte. Here a few more terms that I will use:

Nybble	4 bits, or half a byte
Word	16 bits, or 2 bytes
Double-word	32 bits, 4 bytes, or 2 words
Quad-word	64 bits, 8 bytes, or 4 words

A group of X bits can store 2^X different values. So a single bit can store 2^1 or just 2 values (which we will call 0 and 1). A byte has 8 bits so it can store 2^8 or 256 different values. Similarly a word can store 2^16 values or 65,536, a double word 2^32 or 4,294,967,296, and a quad word 2^64 or a lot.

Hexadecimal Notation

You can display numbers in different bases. For example, decimal or base 10 means that you use 10 different digits (0, 1, … 9), binary or base 2 means you use 2 digits (0 and 1). Binary data stored in bytes is often displayed in hexadecimal (base 16) rather than decimal (base 10), because it is easier to convert hexadecimal to binary. For hexadecimal (or hex for short) we need 16 digits. To the normal 10 decimal digits (0 to 9) we add the 6 alphabetic characters A to F to store the values 10 to 15.

It is much easier to convert between hex and binary since one hex digit converts to 4 binary digits (one nybble). This is useful because sometimes a byte is not used to store a number, instead each of the bits are used as (true/false) flags – being able to easily see which bits are on is very useful. Another advantage to hex is that there are 2 hex digits per 8-bit byte. (Some people prefer octal or base 8, but you need 3 octal digits for a byte with one bit left over, so it's not quite as neat as hex.)

The point of all this is that files are stores as a sequence of bytes, and bytes are most easily shown in hex (because, as we saw above, its too hard to convert decimal to binary, octal is ugly and binary would just take up too much screen space). So HexEdit's (main) display of binary files is a sequence of pairs of hex digits representing the bytes of the file.

You may remember from school about how numbers are stored in different bases. In general a number, say ABCD in base X is A*X*X*X + B*X*X + C*X + D. That is, the number is the sum of the values for each digit. The value for digit N is the digit times X^N (where N = 0, 1, 2, …).

For example, if you convert the hex number 468A to decimal you get 4*16^3 + 6*16^2 + 8*16^1 + 10*16^0 = 4*4096 + 6*256 + 8*16 + 10 = 18,058 (base 10). Converting the same number to binary is easier since each hex digit converts to 4 binary digits. So 468A = 0100 0110 1000 1010 (base 2).

Numbers

Binary files, unlike text files, contain at least some information that is not text. Often this takes the form

of numbers. Numbers are usually stored as fixed-point values in a format that the processor can use, or as floating-point values that the processor (or a numeric coprocessor) can use. Another form of storing numbers is in packed binary-coded decimal (packed BCD), but is becoming less fashionable these days, so I won't cover it here.

Fixed-point values are usually just stored in 1, 2, 4, or 8 bytes. As we saw above these sizes can store 2^8, 2^16, 2^32, or 2^64 different values respectively. For example, a byte can take all the binary values 00000000, 00000001, …, 11111111. These values are by convention taken to mean the decimal values 0 to 255.

Bits are usually numbered starting at zero for the least significant bit. That is, bit X stores the binary digit for 2^X. So the least significant bit is 2^0 or 1, the next is 2^1 or 2, then 2^2 or 4, and so on. Since there are 8 bits in a byte these are numbered from 0 to 7.

Signed Numbers

You can also think of a byte as storing numbers that can be either positive or negative, usually called **signed** numbers. Generally, computers nowadays use **Two's complement** notation when performing signed operations. In this notation zero has all bits off, whereas -1 has all bits on. For a byte, two's complement uses the values 00000001 to 01111111 to store the values 1 to 127, and 10000000 to 11111111 to store the values -128 to -1. Note that to negate a value you flip all the bits and add one.

There are other schemes for representing signed numbers, but all use the top bit to indicate the sign. **One's complement** just flips all the bits to negate a value – so –1 stored in a byte is 11111110, and 11111111 is negative zero. **Sign and magnitude** format just uses the top bit as the sign and the rest of the bits as the magnitude so –1 is 10000001 and 10000000 is negative zero. Note that for positive numbers (the top bit off) all schemes (even unsigned) represent numbers in the same way.

Two's complement notation is favoured over other schemes because the results for signed and unsigned arithmetic are the same – it's just a matter of interpreting the bits and overflow conditions differently. For example, adding the binary numbers 00000001 and 11111100 gives the result 11111101. If you consider this to be a signed operation this is 1 + -4 = -3. Whereas, if you consider it to be unsigned then it is 1 + 252 = 253. However, two's complement notation does have the oddity that there is one more negative number than there are positive numbers, but at least there is no negative zero.

Byte Order

For numbers that are stored in several bytes the order of the bytes is also important. There are two formats: ones that store the most significant digits first and those that store them last. These are colloquially known as big-endian and little-endian formats respectively. For example, if we stored the number 258 (2^16 + 2^1) in a four byte number we could store it as the hex bytes 00 00 01 02 or as 02 01 00 00. (Other systems, luckily out of favour, might store it as 01 02 00 00.) Here the least significant byte is 02, the most significant is 00.

The Intel-based system on which you are probably reading this uses little-endian numbers so 258 is stored as 02 01 00 00. But if you work with binary files from other systems such as Mac, Amiga and most UNIX systems you may see numbers that are big-endian.

Floating Point Numbers

Floating-point formats have a mantissa and an exponent, both of which are signed numbers. The value of the number is quite simply the mantissa x 2^exponent, although some systems may use a hexadecimal or even a decimal (rather than base 2) exponent. Most computers nowadays use IEEE floating-point formats, which have a 32 bit (4 byte), a 64 bit (8 byte) and a (less common) 80-bit (10 byte) format using a binary exponent.

Unlike fixed-point numbers, not all patterns of bits in the bytes of a floating-point number are valid. Even zero cannot be represented by a normal pattern. Zero is often represented (as it is in the IEEE formats) by a pattern with all bits 0 -- although a bit pattern for negative zero (zero with the sign bit on) may be treated the same.

Other bit patterns can be used for +/- infinity, or may just be invalid floating-point values. In IEEE formats invalid bit patterns are called NaNs (for Not a Number). Seeing the right value again also depends on the order of the bytes (ie, big- or little-endian byte orders).

Character Sets

One reason that an 8-bit byte is popular is that it stores 256 values, which is about the right number for storing a set of characters such as the English alphabet, digits, punctuation plus a few other miscellaneous things. (Actually the ASCII character set fits into 7 bits, but 7 is not as nice a number as 8 since 8 = 2^3).

The most common encoding of a byte into a character set is ASCII. This reserves 32 values (0 to 31) for control characters, 94 characters for upper and lower case letters, digits, punctuation etc, plus character 32 is a SPACE, and character 127 is DEL (sort of another control character). Notice that ASCII only uses 128 values and can fit into 7 bits – the top bit of the byte is usually left as zero.

Most ASCII control characters are not important in text files, but a few are very important such as carriage return, line feed, and tab. For example, in MSDOS and MSWindows a carriage return followed by a line feed indicates the end of a line of text. (But UNIX uses a line feed only.)

Another character set is EBCDIC. IBM invented it for its larger computers. It has 64 control characters, a SPACE (character 64) and upper and lower case letters, digits, punctuation etc. Since EBCDIC uses 8 bits it fits exactly in a byte, although some of the 256 values are not used.

One problem with ASCII (and EBCDIC) is that it does not encode all characters from all languages around the world. The ANSI character set extends ASCII to use the extra 128 values of a byte for characters used in western European languages. However, to store all the characters of the world a 16 bit value must be used. This gives 65536 (2^16) different characters. This is what the Unicode character set is for. The first 128 values in Unicode are the same as ASCII.

Binary Files

We often make the distinction between text and binary files. What is the difference? A text file just consists of normal text characters (alphabetic, punctuation etc) plus a few special control characters. In most operating systems a text file is a file containing ASCII display characters plus a few control-characters (line feed, tab etc). (In MSDOS, Ctrl+Z may also be used in text files to signal end of file, for historical reasons).

Binary files on the other hand can contain any of the 256-byte values in each byte. A common use is to store the executable image of a program in a .EXE, or .DLL file under MSWindows. Even a word processor may stores its documents in binary files, since the document is more than just the text, it also includes formatting information etc.

Also it is important that bytes in binary files (unlike in text files) are not inserted or deleted, for various reasons. This is why HexEdit defaults to overtype mode (OVR) rather than insert mode (INS). This is certainly the case with executable files – for example you may be able to change a string in a program, but you cannot insert bytes in order to increase the length of the string. If you do this you will wreck the program so that it cannot run.

Conclusion

I trust the above brief introduction to binary files will be useful in helping you to get the most out of HexEdit. You should now understand what a binary file is, why they exist, and have an inkling of why you may want to use HexEdit to look at and modify them.

Installing and Removing HexEdit

To run HexEdit you need an IBM compatible PC running 32 bit Windows (Windows 95, 98 or NT).

Installation

1. Before you install HexEdit you need to obtain the installation files. These files are probably distributed in a zip file called HexEdit1.ZIP. This file can be downloaded from the HexEdit web site at **http://members.tripod.com/AndrewPhillips**.
1. Place the installation files in a temporary directory so that they can be removed again after you have completed the installation.
1. Exit any applications that you are running. This allows updated versions of shared DLLs (MSVCRT.DLL and MFC42.DLL) to be more easily installed.
1. Run setup.exe and follow the instructions.
1. You may install the following components:

   Program Files – You need these files if you want to run HexEdit!
   Help Files – These should be installed if you want to use the on-line help.
   Shared Files – These are Microsoft shared DLLs that are copied to the Windows SYSTEM directory. They are only copied if they are more recent than the versions currently on the system.
   INI File – Copy this file if you want to keep stored settings in a filed called HexEdit.INI. This has the advantage that you can easily view and change your settings manually. If you do not install HexEdit.INI then all settings are stored in the registry. This allows different users to have different settings, but may clutter the registry.
1. When the installation is complete remove the installation files from the temporary directory. You may want to keep HexEdit1.ZIP in case you have to reinstall HexEdit or to pass along to friends and colleagues.
1. If you intend to run HexEdit from a console command line then it needs to be installed in your PATH. You can do this by adding the directory where you installed HexEdit to your PATH. Alternatively you can move HexEdit.EXE to a directory that is already in your path, although this may confuse any attempt to remove HexEdit from the system.

Note: Always keep the help files (HexEdit.HLP and HexEdit.CNT) and the tips file (HexEdit.TIP) in the same directory as the executable (HexEdit.EXE).

Removing HexEdit

1. In Control Panel double click the "Add/Remove Programs" icon.
1. Click HexEdit in the list of files that may be removed and Click the Add/Remove button.
1. Click "Yes" if you really want to remove HexEdit. The removal procedure is displayed.
1. Once HexEdit has been removed click the OK button.

Note: Some things may not be completely removed until you shutdown and restart the system. For example, you may still see HexEdit shown in the Start Menu until you reboot.

Support

I'd like to hear from you if you want to report a bug, request an enhancement or just tell me you like the program.

If you have a question about how to use HexEdit please check the on-line help first. If you still can't work it out you might try the FAQ (frequently asked questions) at the HexEdit web site (see below). As a last resort send me email.

Email

It is very easy to get in contact with me if you are connected to the Internet for email. While you are running HexEdit you can send email using the "Email Support" command on the Help menu. If you want to send email from a machine where HexEdit is not running or you can't get HexEdit to run at all send email to (in order of preference): **HexEditSup@hotmail.com**, aphillip@geocities.com or aphillips@one.net.au.

If you have found a bug include in your email:

1. A brief description of the problem in the subject prefixed by "BUG:".
2. Describe the problem as fully as possible.
3. Try to remember exactly what you did and what happened.
4. List the steps required to reproduce the problem if you can reproduce it.
5. Include the exact text of any error messages from HexEdit or the system.
6. Include the operating system version and the version of HexEdit you are using.
7. Try to think of anything peculiar to you configuration that may be relevant to the problem.
8. Include your email address(es) so I can get back to you for more information.

Web Site

If you have questions on how to use HexEdit, want to participate in a discussion group, want to download the latest version or find out about new features and add-ons, then check out the HexEdit web site at **http://members.tripod.com/AndrewPhillips**. If you can't find the web site there, email me to find out its new location.

Note that this web site is provided at no cost to my grateful self by Tripod. The disadvantage is that you see advertisements popping up onto your screen when you connect to the web site and change pages. If you are not interested in these ads, please close or ignore it.

Acknowledgments

I'd like to thank Ian Grierson and Steve Mann of Encom Technology for their ideas for enhancements to HexEdit.

I'd especially like to thank my wife Annette and son Daniel for their patience during the many months it took to add the "finishing touches".

If you would like to see your name here in the next version of HexEdit please report any bugs you find, or any ideas you have for enhancements, to HexEditSup@hotmail.com.

Technical Details

Bugs

There are no outstanding known bugs at this time.

Limitations

Due to its use of signed 32 bit numbers to store addresses HexEdit can only be used on files up to 2 Gbytes (2^31 bytes) in size.

There is also a similar limitation on the display 2^31 pixels in the vertical (and horizontal) direction. By default, HexEdit uses a font with a height of 16 pixels and a row size of 16 bytes which allows a file up to 2 Gbytes to be edited. If you use shorter rows and/or a larger display font the maximum file size will be commensurately reduced.

There is no limit on the number of undo operations or "keys" in a keystroke macro except for the limitation of available virtual memory. Memory size would not normally be a problem as storage for these is very efficient. Storage for most undoable operations is 5 bytes; most macro operations use 8 bytes.

The maximum page number you can enter in the page ranges of the print dialog is about 30,000. This is a limitation of Windows 95 and NT (or perhaps MFC). To print parts of a file past page 30,000 use the "print selection" option.

International Support

Unfortunately HexEdit only currently supports English in its menus, messages, help etc. The documentation is in Australian English, which is very similar to English English and not entirely dissimilar to American English.

Luckily HexEdit does support international numeric and date formats when these are correctly configured in the Windows control panel.

Development System

HexEdit was developed using Microsoft Visual C++ 6.0 and MFC. I use NT 4.0 on a Pentium 100 system with 80 Mbytes of RAM.

The Display

The HexEdit display is divided into two or three areas. The main area in the middle of the window is the "hex area" which contains pairs of hex digits representing the bytes of the file. To the left is the address of the first byte of the row (displayed in hex or decimal). Optionally to the right of the hex area is the "character area" which shows the corresponding characters (in ASCII or EBCDIC).

Adjusting the Window Display

HexEdit allows you to edit multiple files simultaneously, and you can open more than one window on the same file. For example, you might want to make changes to the end of a file, while viewing the start of it. To do this, open two windows for the same file and scroll one to the end of the file and start making the changes. You can even make the other window "read only" so that you don't accidentally make changes in it.

Every window in HexEdit has a set of attributes associated with it that can be set individually for each of them. This includes the display font, the number of columns (default 16) and the column grouping (default 4), whether the character area is displayed, and whether it shows ASCII or EBCDIC, etc. Another nice feature is the ability to use "AutoFit" where the number of display columns is not fixed but is adjusted to fit the width of the window when it is resized.

You can configure a window to use overtype or insert mode, and toggle between read only and modifiable modes. Each window also has an associated "mark", which allows you to mark a position, return to it, and measure distances from it.

EBCDIC and Unicode Support

The EBCDIC character set is fully supported in the character display and editing, in searches (case-sensitive and case-insensitive), and in the use of the clipboard.

Unicode is supported in searches and pasting from the clipboard but not in the display. The reason is that since Unicode characters are two bytes it is not possible to say which pair of bytes to use. Also Windows 95 does not support the display of Unicode characters. However, under Windows NT you can view the Unicode character at the cursor position using the Character Properties Page.

Global Options

Most display options are controlled separately for each window – you can use a different font etc in each window. There are, however, several global options that change the display in all windows. For example, there is a global option to determine whether hex numbers are displayed in upper- or lower-case. These are found on the "General" page of the Options Dialog.

The "Defaults" page of the Options Dialog determines the default states of the toolbar buttons when a new window is created – when a new file is opened or created. You can set the default value of the display options that appear in the View menu and on the Toolbar, such as Autofit, EBCDIC mode etc. You can also choose whether windows should open in read-only or read-write mode and overtype or insert mode.

The "Window Display" page of the Options Dialog offers several options not available from the View menu or the Toolbar. You can set the number of columns (if Autofit mode is off), how columns are grouped together, and the "offset" of the first byte of the file (the column it appears in).

For further information see:

Window menu	for options to create windows and change window arrangement
View menu	for display options of a window

Toolbar	for display options of a window
Window Display Options	for other window display options
Colour Options Page	for options to change the colour of hex values
General Options	for global system and display options
Defaults Options	for how to change how new windows are displayed

Moving Around

The normal cursor movement keys (arrow keys, **Pg Up** etc) are supported for moving the cursor. As with most MSDOS/Windows software, the **Home** key moves to the first byte on the line, **End** moves to the last. **Ctrl+Home** and **Ctrl+End** move to the start and end of the file. **Ctrl+Left** and **Ctrl+Right** move left and right by a whole group of columns. You can edit the file either in the hex or the character area – to quickly flip between the areas use the **Tab** key.

The scroll bars allow you to move the current display without changing the cursor position. For a large file the vertical scroll bar (on the right side of the window) is particularly useful. Clicking the arrows at the end of the scroll bar (or using the Ctrl+U or Ctrl+D keyboard shortcuts) moves by one line. Clicking above or below the scroll box moves by a page. You can also drag the box to a position anywhere within the file. (Note that the size of the box, in proportion to the size of the entire scroll bar, reflects the amount of the file that is visible within the window – if the entire file is visible then the scroll bar will disappear.) The Ctrl+L keyboard shortcut scrolls the display so that the cursor is positioned in the middle of the window.

Jumping to an Address

A simple way to move to a distant address, is to use a removalist, or better still drag the scroll box until the address you want is visible and click within the display to set the cursor location. However, for very large files the scroll bar resolution may mean that you can't drag to the exact position but may have to use the Page Up or Page Down keys as well.

Alternatively, the Decimal and Hex Address Tools in the Edit Bar allow you to jump quickly to a known address within the file. (Note that, as is sensible and conventional in binary file editors, the "first" byte of the file has an address of zero.) As you type an address in one tool the corresponding address is updated in the other – as a useful side effect, this provides a simple way to convert between hex and decimal numbers. These tools also display the current address as you move the cursor around the screen and during searches and file comparisons.

Both of these tools also store separate histories of the last addresses jumped to, which can be selected from the drop down list. Note that you have to press the Enter key to actually jump to the address and return to the edit window. You can use the Tab and Shift+Tab keys to quickly flip between these two controls without returning to the edit window.

The Mark

A way to return to a previous address is to mark that address with the Mark Position command and later return to it using the Go To Mark command. The mark is also useful for measuring distances between points in the file. There are decimal and hex "Distance to Mark" indicators in the status bar that show the distance from the mark to the current cursor position.

For further information see:

Edit Bar	to see options for jumping and setting the mark
Hex Address Tool	to see how to jump to a hex address
Decimal Address Tool	to see how to jump to a decimal address
Scrollbars	to see how to use scrollbars
Status Bar	for information on the "Distance to Mark" indicators

Editing

When editing a file HexEdit behaves much like a word-processor or editor. You can move the cursor around using the keyboard or the mouse (see Moving Around). You can select text with the keyboard or mouse and cut, paste etc (see Selecting and the Clipboard).

One difference is that there are two separate areas for editing – the hex area and the character area (although the character area can be disabled). You can enter any byte value in the hex area but only by typing two hex digits – HexEdit beeps if you type any other characters. On the other hand you can only enter ASCII (or EBCDIC) characters in the character area. To quickly flip between the areas use the Tab key.

There are two modes that affect editing behaviour. If you are in read-only mode then you cannot modify the file at all. This mode is separately controllable for each window so you can have two windows on the same file in different modes. But if the *file* itself is opened read-only then all windows for the file must remain read-only. The RO/RW indicator on the status bar indicates the current mode – you can double-click this indicator to toggle the state.

The other mode is the Insert/Overtype mode. Similarly to the read-only mode, double clicking the INS/OVR status bar indicator toggles the state. The difference between insert and overtype is explained below..

In insert mode the length of the file may change. As you enter bytes the following ones are shifted forward. Similarly deleting, cutting etc removes data and shifts the following bytes backward. But in overtype mode existing bytes are never moved, instead changing bytes destroys their previous value. Further you cannot delete or cut (but you can copy) to the clipboard in overtype mode.

In many types of binary files it is essential that any changes do not move bytes around and the length of the file does not change – for safety always edit these types of files in overtype mode. On the other hand it is often very useful to be able to insert bytes within a binary file. Also, to add to the end of a file you must be in insert mode.

Undo

When you make changes to a file, HexEdit keeps track of these changes and allows you to undo them. In fact HexEdit uses this undo information plus the original file on disk to show you what the file is currently like. This means that you can quickly edit very large files since, unlike many editors, the whole file does not have to be read into memory (or some sort of virtual memory). But this system does have the disadvantage that you cannot undo changes that were made before you last saved the file to disk. Also if you insert or delete bytes in the file, then HexEdit has to make a copy of the file while saving it which can be very slow for a large file – another good reason to work in Overtype mode.

As well as a file-level undo, which keeps track of changes to the file, HexEdit also has a window-level undo that keeps track of cursor movement, changes made to the display etc. However, since changes made to the file can affect the display in another window for the same file HexEdit must take special precautions to keep its internal structures consistent. If you undo a change to the file in one window, all display changes in other windows on that file are undone back to the point in time when the original file change was made. Moreover, if you try to undo back past a change made in another window HexEdit will ask you if you really want to undo the change.

File Handling

HexEdit behaves like many Windows programs, such as word-processors, when it comes to the opening, closing, saving etc of files. The only difference is that instead of HexEdit being associated with a particular type of "document", you can work with any type of file. In the File Open Dialog you will see all

files, since the default file type is "All Files (*.*)".

The **Read only** check box in the File Open dialog allows you to open a file for read only. You may want to do this so that you don't accidentally modify the file or to allow other software to simultaneously read the file while you have it open. If the file *must* be opened read-only because it is a hidden or system file or because something else is reading it then HexEdit will warn you. Note that even if a file is not opened read-only a window on that file may be read-only (see above).

For further information see:

Edit menu	for information on the Undo, Allow Changes and Insert commands
Status Bar	for information on RO/RW and INS/OVR indicators
File menu	to see options for opening, creating and saving files
File Open Dialog	to see options available when opening an existing file
File Save Dialog	to see options available when saving a file under a new name

Help

HexEdit provides extensive help. As well as the standard help topics available from the Help menu there are status bar messages, tooltips, tips of the day, context-sensitive help and popup help for dialog controls.

The status bar displays a message when you move the mouse over a menu item or a toolbar button, which describes what it does. A short "tip" is also displayed if you hold the mouse over a toolbar button or an indicator on the status bar, which describes what it is for.

To get more information on a menu item, toolbar button, or any part of the HexEdit display you can press Shift+F1. This invokes a help mode where the context-sensitive help is shown for the next thing you click on. You can also get context-sensitive help by clicking a toolbar button or menu item and pressing F1 before you release the button.

When you start HexEdit a new Tip of the Day is displayed. This can be disabled using the "Show tips on Startup" check box. You can also see the latest tip using the Tip of the Day command on the Help menu. The tips are stored in a text file called HexEdit.tip, which is normally found in the same place as HexEdit.exe.

Dialog Help

Often the place you really need help is in some obscure dialog, which you rarely use. HexEdit provides extensive help in dialogs.

Often the simplest way to find out about a dialog is to click its Help button. The help topic displayed show general information about the dialog as well as specific information about the controls in the dialog.

To get quick help on a control in the dialog click the "?" in the dialog's title bar and click on the control. Alternatively, select the control with the mouse or arrows keys and press F1. A small description is displayed in a "tip" window.

For further information see:

Help menu	for the Help Topics and Tip of the Day commands
Status bar	for information on help messages

Selecting and the Clipboard

To select part of a file you can use the mouse or the keyboard. To use the mouse, click the first byte you want to select and drag the mouse to the last byte. Alternatively, you can click (and release) on the first byte then shift-click (hold down the Shift key while you click) the last byte. Shift-clicking is very useful if you need to select a large area since you can use the scroll bar to quickly move to the other end of the desired range.

To use the keyboard you can use the Shift key with the normal cursor movement keys. For example, to select everything from the current cursor position to the end of the file press Shift+Ctrl+End.

Once selected you can cut or copy the bytes to the clipboard in the normal manner. Note that you cannot cut if the window is in either of the read-only or overtype modes since this will delete bytes from the file. You can also perform other operations with selected text such as write it to a file or print it.

Clipboard Formats

When HexEdit places data on the clipboard it does so in two formats. One format is ASCII text which allows other programs (such as word-processors and editors) to read it. But in text format some control characters, such as the character with ASCII value zero, cannot be stored. For this reason HexEdit also stores the data in a binary format, allowing you to cut and paste binary data within HexEdit without losing information. (Note that this binary format is the same as is used in the Microsoft Developer Studio binary file editor.)

Note that if you are in EBCDIC mode the text format data placed on the clipboard is converted from EBCDIC to ASCII. (The binary format data is still the actual binary data values.) This allows you to copy EBCDIC text to any Windows editor or word-processor. As for ASCII control characters, any invalid EBCDIC characters are silently ignored. Also, if there is no binary data format on the clipboard (ie, the data on the clipboard was not placed there by HexEdit) and you're in EBCDIC mode then the text is converted to EBCDIC when pasted.

Sometimes you may actually want the hex data as the hex digits so you can paste it into a text file. To do this, copy it to the clipboard as hex text. The text is copied with spaces between the pairs of hex digits and with the same line length as shown on screen. This is also useful if the some bytes of the file contain an address of data elsewhere in the same (or another) file. Copy the address as hex bytes, and paste it into the Hex Address Tool and press Enter to jump to the address.

A similar option to copying as hex text is to copy as C/C++ source code. This text is suitable for initialising an array of bytes in a C or C++ program. That is, it consists of a comma-separated list of byte values. In the future other integer and floating-point formats will be supported.

The "Paste As" menu items ensure that you paste you paste text (and not binary data) and it is pasted in the desired character set – ASCII, Unicode or EBCDIC. These commands can be used to convert a large amount of text from one format to another. For example, to convert text from EBCDIC to ASCII you need to cut it to the clipboard while in EBCDIC mode and then use the "Paste As ASCII" command. You **cannot** simply change to ASCII mode and paste, as the binary format data placed on the clipboard by HexEdit takes precedence, and the data will be pasted without conversion.

For further information see:

Keyboard shortcuts	for how to select, cut and paste using the keyboard
Edit menu	for clipboard and selection commands

Properties Dialog

The Properties dialog lets you easily view the properties of the file and the properties of the byte(s) at the cursor in the active window. You can view the name, type, size, date modified and attributes of the file. You can also view the character at the cursor position in hex, decimal, octal and binary, as well as its ASCII, EBCDIC and Unicode representation. You can also see the decimal value (signed or unsigned, big-endian or little-endian) of the byte, word, double word or quad word after the cursor, as well as the 32 or 64 bit IEEE floating-point value.

As you move the cursor around the file or swap between windows the Properties page, if open, is automatically updated to display the current values. In the future you will also be able to use the Properties dialog to modify some of these values, such as decimal and floating-point values.

For further information see:

Properties command	to find out how to invoke the Properties dialog
File Properties Page	to view information about the disk file
Character Values Page	to view information about the byte at the cursor
Decimal Values Page	to view the decimal values of the byte, word etc following the cursor
Floating-Point Page	to view IEEE floating-point values of the bytes following the cursor
IBM Floating-Point Page	to view IBM floating-point values of the bytes following the cursor
Status bar	to view byte values and invoke the Properties dialog

Tools

Quick searches can be performed with the Find Tool on the Edit Bar. The Find Dialog provides similar facilities but more options. Searches are done with the Boyer-Moore algorithm which, apart from being very fast, has the peculiarity that the longer the string of (unique) characters you are searching for the faster the search, although the limiting factor is likely to be the speed of your disk or network.

Find Tool

The Find Tool on the Edit Bar allows you to search for bytes by either entering them as hex digits, or as a string. String searches can be case-sensitive or case-insensitive, and use EBCDIC rather than ASCII characters if EBCDIC mode is on. You can also quickly start a case-insensitive, case-sensitive or hex search using the F4, F5 or F6 key.

The Find Tool also retains a history of the most recent searches performed. This history list is retained between invocations of HexEdit and is shared with the Find dialog.

Find Dialog

The Find dialog provides a few more options than the Find Tool. In particular, you can perform Unicode searches. Also the active options are more obvious. The only disadvantage of the Find Dialog is that it takes up more screen space than the Find Tool, but it does automatically get out of the way of the cursor if it can.

Other "find" commands allow you to repeat a search, using the current search bytes, in either direction. The "Find Selection" command allows you to search for the next occurrence of the currently selected bytes.

Comparing Bytes

You can compare the data in any two (maximised) windows using Alt+C, even two parts of the same file. If a difference is found the offending bytes are selected in both windows – pressing Alt+C again will continue the comparison at the following byte. It's often useful to tile (vertically) the two windows being compared before starting the comparison to easily see the results and turning on Autofit mode.

While a long comparison (or search) is in progress the Address Tools in the Edit Bar are updated with the current address to show progress. If no differences are found (ie, the files are the same in both windows up to the end of file) then a message to that effect is displayed in the status bar and the cursor is not moved. If a difference is found (ie, a different byte was encountered or end of file was encountered in but one of the files) then a message is displayed in the status bar indicating the number of bytes that were the same before the difference was found.

For further information see:

Find Tool	to find out how to do quick searches
Find command	to invoke the Find dialog
Find dialog	to see the full range of searching options
Find backward command	to find out how to repeat a search towards the start of file
Find forward command	to find out how to repeat a search towards the end of file
Find selection command	to search for the next occurrence of the selected bytes
Compare command	to find out how to compare files or parts of the same file

Macros

Keystroke macros provides a quick and simple way to repeat a sequence of actions. (Actually, this name is a bit of a misnomer since more than just "keystrokes" can be recorded.) Macros are not limited to one window or file. You can include commands that open or create files, switch between windows etc.

Almost all commands invoked with menus, the keyboard or toolbars can be recorded. Changes to settings in the Options Dialog are **not** recorded except for Window Display page. Any settings that are changed in this page are recorded, but unchanged settings are not.

Useful Keys

Some keys are particularly useful in macros. Of course, the cursor keys and the keys for cutting and pasting are the chief tools. Keys dealing with the mark are also often very useful. It is common to set the position of the mark and later return to it (with Ctrl+F9), or swap the cursor and the mark (Shift+F9), or select the text between the cursor and the mark (Shift+Ctrl+F9). I also find the command to increment the current byte (the keyboard shortcut is "+" on the numeric keypad) useful in many ways in macros.

You will often want to observe the progress of the macro but as the macro runs it may creep towards the bottom or top of the display. For a better view of what's happening you may want to include the scrolling commands (Ctrl+U or Ctrl+D) in the macro to reposition the display. If you intend to replay a macro many times you can have the display refreshed after every replay (see below) and record Ctrl+L as the last key of the macro to centre the cursor and redraw the window.

As an example of a keystroke macro, suppose you wanted to create a file with all 256 byte values. To do this open a new file and insert a null byte (ie. a byte with value zero) and select it. Next, start macro recording and copy the byte to the clipboard (Ctrl+C), move the cursor forward and paste it (Ctrl+V), then invoke the Increment Byte command ("+"), and stop recording. You now have two bytes with the values zero and one. You need 254 more bytes so invoke the Multi-Play command from the Tools menu type 254 and press **Enter**. The rest of the file will be filled with bytes ranging from 2 to 255.

Display Refresh

If you need to run a macro hundreds of time on a very large file it may take a very long time to run. A large part of that time (up to 90% or more) may be due to redrawing the screen every time the file is changed or the window is scrolled. To speed things up HexEdit allows you to turn the screen refresh off when macros are running.

On the other hand, you may want to see that the macro is behaving as you expected so HexEdit has several options to regularly refresh the screen while the macro is running. You can refresh based on the passage of time (eg. every 5 seconds) the number of keystrokes replayed, or the number of times the whole macro has been played (for multiple replays). The Macro page of the Options dialog controls these options.

Error Handling

If an error occurs during macro playback the macro will stop playing immediately. This helps avoid having runaway macros completely destroy or mangle your file.

Of course, you can always use the Undo command to reverse any changes that are made if the macro goes haywire. But if you are running a macro hundreds of times it can be tedious to hold down the undo button that many times (although you could record undo in a macro and play it multiple times). Also remember that if the macro includes saving the file (using the Save or Save As commands on the File menu) then any undo information is lost.

Having macros stop on error is also useful if you don't know how many times you want to run the macro. For example, you must use a macro to do a global search and replace, since HexEdit does not provide such a command. If you don't know how many times the bytes you want to replace occur just play the macro a large number of times, say 1000, and it will stop the first time the search fails, that is, after all the replacements have been made.

There are numerous error conditions that can halt a macro. Some of these are described in the help for the Macro page of the Options dialog. Although, user interaction when a macro is playing is minimal sometimes it is necessary to display a dialog to ask a question. If the user cancels the dialog or says not to continue the macro will stop.

Whether something is an error or not often depends on what you are doing. For example, if part of the macro is pressing the down arrow key, it is normally an error if you are at the end of the file and the macro will stop. But sometimes this may not be an error and you will want the macro to continue running. To allow for this you can set the error severity level that will stop a macro from running in the Macro page of the Options dialog.

For further information see:

Record command	to find out how to record a macro
Play command	to find out how to quickly replay a macro
Multi-Play command	to find out how to replay a macro more than once
Macro Options Page	to see macro play options such as how the display is refreshed
Window Display page	to see settings that can be recorded in macros
Keyboard Shortcuts	to see a list of keys, which might be useful in macros

Printing

If you are acquainted with printing in other Windows programs printing is fairly simple and straightforward. What you see in the window is almost exactly what is printed, but to see how the file would be spread across pages you can use the Print Preview command. You can select the printer to use any options such as landscape or portrait printing with the Page Setup command.

It is not unusual to want to print a small part of a binary file. You can do this in HexEdit by printing a range of pages, but you must first use Print Preview to find out the numbers of the pages you want to print.

One problem with printing a range of pages is that the standard Print Dialog in Windows 95 does not support page numbers greater than about 30,000. If you want to print part of a very large file such as a database, this is a problem. A better way to print part of a file is to select the bytes you want to print and use the "Print Selection" option of the Print Dialog.

If you are lucky enough to have a high-resolution screen, or unlucky to have a narrow printer then you may need to adjust your display before printing. HexEdit prints the file as it is shown on the screen and approximately the same size. If the printer page is not wide enough the right side of the page will be lost. You can check this before printing use Print Preview. To avoid this problem, you can use less display columns (or reduce the window width if using Autofit mode) or print in landscape mode.

For further information see:

File menu	for printing commands
Print dialog	to see options for printing the file or selection
Print Progress dialog	to see progress while printing is in progress
Print Setup Dialog	to find out how to select and configure the printer
Display Options Page	to change the number of displayed columns

Options Dialog

The General page of the Options Dialog allows you to make global changes to the display. For example, I prefer to use the uppercase letters A-F for hex numbers since there is less chance of confusing the number six (6) and lowercase B (b). Some people prefer lowercase letters, so you can change this. You can also choose whether you want nice looking addresses displayed – that is, thousands separators in decimal addresses or four digit grouping in hex addresses.

Other options on the General page are global options that determine whether backup files are created and whether the options you have changed are saved when you exit HexEdit. You can also set the options how you want them and save them immediately with the "Save Now" button.

The Window Defaults page of the Options dialog allows you to determine the behaviour of new windows. That is, you can set the default look and modes of windows that are created when a new file is opened or created.

The Colour Options and Window Display Options pages allows you to change options relating to the display of the active window. If there are no files open then these pages do not appear. The Colour Options page allows you to control the colour any and all values in the hex area. Many display options are available on the Toolbar and/or on the View menu, but a few options such as the number of columns and column grouping are only available in the Window Display Options page.

For further information see:

Options command	to find out how to invoke the Options dialog
General Options Page	to see global system and display options
Macro Options Page	to see options that affect how macros are played
Colour Options Page	to see options controlling colours in the active window
Window Display Options	to see options that affect the display in the active window
Window Defaults Options	to see options to configure the default window display

File menu commands

The File menu offers the following commands:

New	Creates a new file.
Open	Opens an existing file.
Close	Closes an opened file.
Save	Saves an opened file using the same file name.
Save As	Saves an opened file to a different file name.
Print	Prints a file, or part thereof.
Print Preview	Displays the file on the screen as it would appear printed.
Print Setup	Selects a printer, connection etc.
Recent Files	Re-opens a recently opened file.
Exit	Exits HexEdit.

Edit menu commands

The Edit menu offers the following commands:

Undo	Reverses the last operation.
Cut	Deletes data from the file and moves it to the clipboard.
Copy	Copies data from the file to the clipboard.
Paste	Pastes data from the clipboard into the file.
CopyAs Hex Text	Copies bytes to the clipboard converting to hex digits.
CopyAs C Source	Copies bytes to the clipboard converting to C/C++ source code.
PasteAs ASCII	Pastes from the clipboard ignoring non-text ASCII bytes and EBCDIC mode.
PasteAs Unicode	Pastes from the clipboard as Unicode characters.
PasteAs EBCDIC	Pastes from the clipboard as EBCDIC characters.
Select All	Selects the entire file.
Write Selection	Writes the current selection to a file.
Read File	Inserts a file at the current cursor position.
Find	Searches for hex digits, or a string.
Find Backward	Repeats the last search, towards the start of file.
Find Forward	Repeats the last search, towards the end of file.
Find Selection	Searches for the next occurrence of the current selection.
Mark Position	Sets the mark to the current cursor location.
Go To Mark	Returns the cursor to the last marked location.
Extend to Markt	Extends the selection to the current position of the mark.
Swap Cursor and Markt	Swaps the positions of the cursor and the mark.
Allow Changes	Toggles between allowing changes and read-only mode.
Insert	Toggles between Insert and overtype mode.

View menu commands

The View menu offers the following commands:

Toolbar	Shows or hides the Toolbar.
Edit Bar	Shows or hides the Edit Bar.
Status Bar	Shows or hides the Status Bar.
Autofit	Toggles the mode in which the columns are adjusted to fit the window.
Font	Invokes the standard Font Selection dialog.
Decimal Addresses	Toggles between hex and decimal addresses.
Character Display	Toggles the display of the character area.
EBCDIC Characters	Toggles between display of ASCII or EBCDIC characters.
Control Characters	Toggles the display of ASCII control characters.
Graphic Characters	Toggles the display of “graphic” characters.
Options	Invokes the Options dialog.
Properties	Invokes the Properties dialog.

Tools menu commands

The Tools menu offers the following commands:

Find	Searches for hex digits, or a string.
Compare Windows	Compares the values in two windows.
Increment Byte	Increments the byte after the cursor.
Increment Word	Increments the word following the cursor.
Increment Double Word	Increments the double-word following the cursor.
Increment Quad Word	Increments the quad-word following the cursor.
Decrement Byte	Decrements the byte after the cursor.
Decrement Word	Decrements the word following the cursor.
Decrement Double Word	Decrements the double-word following the cursor.
Decrement Quad Word	Decrements the quad-word following the cursor.
Flip Word	Flips the byte order of the word following the cursor.
Flip Double Word	Flips the byte order of the double-word following the cursor.
Flip Quad Word	Flips the byte order of the quad-word following the cursor.
Record/StopRecording	Starts or stops recording of the keystroke macro.
Play	Plays the last recorded keystroke macro.
Multi-Play	Plays a macro the specified number of times.
Options	Invokes the Options dialog.

Window menu commands

The Window menu offers the following commands:

New Window	Creates a new window that views the same file.
Next Window	Activates the next non-minimised window in a cyclic list.
Cascade	Arranges windows in an overlapped fashion.
Tile Horizontally	Arranges windows as non-overlapped, stacked tiles.
Tile Vertically	Arranges windows as non-overlapped, side by side tiles.
Arrange Icons	Arranges icons of closed windows.
Window 1, 2, ...	Goes to the specified window.

Help menu commands

The Help menu offers the following commands:

Help Topics	Offers you a Contents page and an Index to the help topics.
Email Support	Sends email to HexEdit's author for tech support.
Tip of the Day	Displays the latest "Tip of the Day".
About	Displays the version number of this application.

New command (File menu)

Use this command to create a new empty file. You can open an existing file with the Open command. The file created is given a generic default name, but when you attempt to save it with the File Save command the Save As dialog is automatically brought up.

If your options are set such that when you open a file it is opened in overtype mode then you will not be able to add to the empty file until you turn off overtype mode (see Insert Mode Toggle command).

Shortcuts

Keys: Ctrl+N

Open command (File menu)

Use this command to open an existing file in a new window. This command brings up the standard File Open dialog, which allows you to select any file to open. However, if the file you select is already open for exclusive use by another application or the operating system then you will not be able to open it. If the file is only open for read access by the other application (or the file is marked as a read only or system file) then you can still open it in read only mode.

You can also force a file to be opened for read-only access by selecting the Read Only check box near the bottom of the dialog. You might do this if you want to make sure you don't accidentally make changes to the file or if you want to allow other applications to have simultaneous read access. When the file is opened read-only you can never make changes to it unless you close and reopen it. This is not be confused with the facility to make a particular window on a file read only (see Allow Changes Toggle).

If you want to modify a file but retain the original version you can open the file as normal and save it under a different name using the File Save As command. Alternatively, you can create a new file (File New) and read the file to be modified into this file using the Read File command option.

Shortcuts

Toolbar: 
Keys: Ctrl+O

Close command (File menu)

Use this command to close the active window. If this is the last window open for a file and you have made unsaved changes to the file then HexEdit will prompt you to save those changes. If you close a file without saving, you lose all changes made since you opened it or the last time you saved it.

Before closing an untitled file, HexEdit displays the Save As dialog box and suggests that you name and save the file.

Save command (File menu)

Use this command to save the active file using its current name and directory. When you attempt to save a newly created file that has not yet been saved, HexEdit displays the Save As dialog box so you can name the file. If you want to change the name and/or directory of an existing file, before you save it, choose the Save As command.

HexEdit allows you to create backup copies of files. Every time the file is saved the previous version of the file, if any, is saved with the same name except that the file extension is changed to ".BAK". (If the file being saved has an extension of ".BAK" then the previous version is saved with an extension of ".BAC".) The second most recent version, if any, is lost. Whether or not backup files are created is determined by setting options in the General Options page.

If you have inserted or deleted bytes in a very large file HexEdit must copy the file to a new file when you save the file and then delete the old file (unless it is to become a backup file) and rename the new one. This means that the save operation may take a long time and that you may run out of disk space if you do not have more free space on the disk than the size of the file. To avoid this problem do not insert or delete bytes in very large files (or create backup copies) – this is most simply achieved by always operating in overtype (not insert) mode.

Shortcuts

Toolbar: 
Keys: Ctrl+S

Save As command (File menu)

Use this command to save the currently active file under a different name. You might do this if you want to modify a file but retain the original version. HexEdit displays the Save As dialog box so you can save the file under a different name.

To save a file with its existing name and directory, use the Save command. The information and warnings about backup files and large files mentioned under the Save command also apply here.

Print command (File menu)

Use this command to print a file, or part of it. This command presents the Print dialog box, where you may specify to print a range of pages or the currently selected bytes, the number of copies, the destination printer and its properties, and other printer setup options.

To connect to a network printer, or change the paper size, source, or orientation use the Print Setup command first.

To print a small part of a file you can specify a page or a range of pages by first using the Print Preview command to find out which pages you want to print. However, for a large binary file this has the limitation that the largest page number you can use is 30,000. An easier way to print part of a file, which also side-steps this problem, is to select the bytes you want to print and print the selection rather than a range of pages.

Shortcuts

Toolbar: 
Keys: CTRL+P

Print Preview command (File menu)

Use this command to display the active file as it would appear when printed. When you choose this command, the main window will be replaced with a print preview window in which one or two pages will be displayed in their printed format. The print preview toolbar offers you options to view either one or two pages at a time; move back and forth through the pages; zoom in and out on a page; and start printing.

Print Setup command (File menu)

Use this command to change where and how printing is done. This command brings up the Print Setup dialog box, where you can specify the printer to use and its connection, including network printers. You can also change the paper orientation, paper source and other printer settings.

Recently Opened Files (File menu)

HexEdit keeps track of the 10 most recent files that you have opened. These are added near the bottom of the File menu. You can use the numbers or file names listed there to reopen one of these files.

If the file cannot be opened because it no longer exists HexEdit will tell you of this and remove it from the list. Similarly HexEdit will warn you if the file is in use by another program and could not be opened or could only be opened in read only mode.

Exit command (File menu)

Use this command to close HexEdit. You can also click on the close button in the top right corner of the HexEdit main window or use the Close command on the application control menu.

If you have any open files that have unsaved changes HexEdit will prompt you to save or discard the changes. If you elect to save any files that have not been previously saved HexEdit brings up the Save As dialog box so you can choose a name to save the file under.

Shortcuts

Mouse:	Click the main HexEdit window close button.
Keys:	Alt+F4

Undo command (Edit menu)

Use this command to reverse the last action, if possible. HexEdit supports an unlimited undo facility. Unlike many editors changes to the display not just changes that modify the file can be undone. To accomplish this HexEdit uses a two-level undo structure, since the same file can be edited in more than one window. That is, each window stores its own display changes in a "window" undo stack. Changes that actually modify the data of the file are stored in a shared "file" undo stack.

To avoid inconsistencies, if a change to a file is undone in one window all display changes back to that point are undone in all windows for that file. Further, HexEdit will warn you if you try to undo a change that was made in a different window.

Shortcuts

Toolbar: 
Keys: Ctrl+Z, Alt+U

Cut command (Edit menu)

Use this command to place the current selection on the clipboard and delete it from the file. Cutting data to the clipboard replaces the contents previously stored there.

This command is unavailable if the file is read only, or there is no data currently selected. If the window is in overtype mode and/or read only mode you cannot cut -- HexEdit prompts you to turn off these modes. Alternatively, you can copy to the clipboard using the Copy command which will not modify the file.

HexEdit stores the data to the clipboard in two formats. The first format is a binary format that allows non-text characters (such as character zero) to be cut and pasted within HexEdit. It is also the same format that the Microsoft Developer Studio uses in its binary editor.

HexEdit also cuts the bytes to the clipboard as ASCII text, which allows other text-based programs to retrieve them. However, in this case some non-text characters (in particular, character zero) are lost. If HexEdit's character display is currently EBCDIC then the characters are converted from EBCDIC to ASCII before being placed on the clipboard as text but any invalid EBCDIC characters are ignored.

See Paste for more information.

Shortcuts

Toolbar: 
Keys: Ctrl+X

Copy command (Edit menu)

Use this command to copy selected data onto the clipboard. This command is unavailable if there is no data currently selected. Copying data to the clipboard replaces the contents previously stored there.

Note that you can select and copy data even if the file is read only.

HexEdit stores the data to the clipboard in two formats. The first format is a binary format that allows non-text characters (such as character zero) to be cut and pasted within HexEdit. It is also the same format that the Microsoft Developer Studio uses in the hex editor.

HexEdit also copies the bytes to the clipboard as ASCII text that allows other text-based programs to retrieve it. However, in this case some non-text characters are lost. If HexEdit's character display is currently EBCDIC (see EBCDIC Toggle) then the characters are converted from EBCDIC to ASCII before being placed on the clipboard as text and any invalid EBCDIC characters are not copied.

Also see Paste for more information.

You can also copy binary data to the clipboard and convert it to text using the Copy As Hex Text and the Copy As C Source commands.

Shortcuts

Toolbar: 
Keys: Ctrl+C

Paste command (Edit menu)

Use this command to insert a copy of the clipboard contents at the insertion point. This command is unavailable if the clipboard is empty or the file is read only. The pasted bytes are selected so you can easily see which bytes were inserted.

The bytes pasted may have been placed on the clipboard by previously using the HexEdit Copy command or Cut command, or by cutting within another program that places bytes on the clipboard in simple text format.

When cutting and pasting HexEdit tries not to lose any bytes of data. When pasting it first searches for its own internal binary format before looking for simple ASCII text on the clipboard. This means that cutting and pasting data containing any character (including character zero) will not lose anything. Microsoft Developer Studio also uses this binary format.

If you really want to paste text from the clipboard (perhaps removing any non text characters) then you must use the Paste As ASCII (or Paste As EBCDIC) command. Of course, if the bytes are all text characters then using Paste As ASCII has the same effect as the Paste command.

Shortcuts

Toolbar: 
Keys: Ctrl+V

Copy As Hex Text command (Edit menu)

Use this command to copy the selected bytes to the clipboard. This command is unavailable if there is no data currently selected. Copying data to the clipboard replaces the contents previously stored there.

Note that you can select and copy data even if the file is read only.

Use this command to copy data to the clipboard converting each byte to ASCII text as two hex digits. The exact format of the text, such as the number of columns, the case of the hex digits A to F, etc depends on the options in effect for the window at the time the copy is done.

This command is useful if you want to paste hex values in another program such as a text editor. It can also be used to extract a value from the file which is an address elsewhere in the file – this value can then be pasted into the Hex Address Tool (using the context menu available by clicking the right mouse button) to quickly jump to that address.

See also the Copy command and Paste command.

Copy As C Source command (Edit menu)

Use this command to copy the selected bytes to the clipboard. This command is unavailable if there is no data currently selected. Copying data to the clipboard replaces the contents previously stored there.

Note that you can select and copy data even if the file is read only.

This command copies data to the clipboard converting each byte to a hex number in C source code format. The exact format of the text, such as the number of columns, case of the hex digits A to F etc, depends on the options in effect for the window at the time the copy is done.

This command can be used to extract binary data from a file for use in a C or C++ program. The text is in a format suitable to initialise an array of bytes. In the future other formats, such as words, and floating-point values will be added.

See also the Copy command and Paste command.

Paste As ASCII command (Edit menu)

Use this command to insert a copy of the clipboard contents as ASCII text. This command is unavailable if the clipboard is empty or the file is read only. The pasted bytes are selected so you can easily see which bytes were inserted.

This command can be used to paste data from the clipboard while ensuring that it is in ASCII format. One use would be to convert text from EBCDIC to ASCII by Cutting to the clipboard while in EBCDIC mode and Pasting as ASCII. Another use would be to remove invalid control characters by Cutting (in ASCII mode) before Pasting as ASCII.

Note that if there is text currently on the clipboard, but no binary data, then the Paste As ASCII command behaves exactly the same as the Paste command. But if there is binary data (presumably cut using HexEdit) it ensures that you only get ASCII text.

See also the Copy command, Paste command, Paste As EBCDIC command, and Paste as Unicode command.

Paste As Unicode command (Edit menu)

Use this command to insert a copy of the clipboard contents as Unicode text. This command is unavailable if the clipboard is empty or the file is read only. The pasted bytes are selected so you can easily see which bytes were inserted.

One use for this command would be to convert from ASCII (or EBCDIC) to Unicode. To do this, select text in HexEdit, and Cut or Copy it in ASCII (or EBCDIC) mode, then Paste as Unicode. Note that Unicode is a 2 byte character encoding which is a superset of ASCII – converting ASCII to Unicode will just add a high zero byte to each character.

See also the Copy command, Paste command, Paste As ASCII command, and Paste as EBCDIC command.

Paste As EBCDIC command (Edit menu)

Use this command to insert a copy of the clipboard contents as EBCDIC text. This command is unavailable if the clipboard is empty or the file is read only. The pasted bytes are selected so you can easily see which bytes were inserted.

One use for this command would be to convert from ASCII to EBCDIC. To do this, Cut or Copy text to the clipboard in HexEdit or any editor or word processor then Paste As EBCDIC. Note that any characters on the clipboard that are not valid EBCDIC characters (such as tilde "~" or graphic/ANSI characters that have the high bit set) are ignored.

See also the Copy command, Paste command, Paste As ASCII command, and Paste as Unicode command.

Select All command (Edit menu)

Selects the whole file. You can select all the bytes from the current cursor location to the start of file or to the end of file using Ctrl+Shift+Home or Ctrl+Shift+End.

Be careful selecting all of a very large file (many Megabytes in size) and doing some things which may need a lot of memory. Copying to the clipboard (see Copy command) may cause your machine to slow down or even crash if you don't have enough virtual memory. There may be a safer way to achieve what you want to do, such as by using the Write Selection command or even using the MSDOS "COPY /B …" command to join two or more binary files.

Shortcuts

Keys: Ctrl+A

Write Selection to File (Edit menu)

Use this command to write the selected bytes to a disk file. This command is unavailable if there is no data currently selected.

This command brings up the standard Save As dialog which allows you to save the file using any valid file name. If the file already exists you are prompted to overwrite it. Note that if you choose to overwrite an existing file no backup copy is created even if backups are on.

This command can save memory problems when working with large blocks (many Megabytes) of data. For example, rather than cutting from one file and pasting into a new file this command can be used to create a new file based on part of an old file. You can then open the new file in HexEdit and modify it. This is much more memory efficient as HexEdit works from the existing disk bytes rather than from memory except when you paste or modify a file.

This command is in some ways the inverse of the Read File command.

Read File command (Edit menu)

Use this command to insert a copy of a disk file at the insertion point. This command is unavailable if the file you want to insert into is read only.

This command is in some ways the inverse of the Write Selection to File command. It uses the standard file open dialog to allow you to select a file to insert. The inserted bytes are selected so you can easily see which bytes were added.

Find command

Use this command to search the file. This brings up a modeless dialog box that you can leave on the screen and use whenever necessary. t is sometimes quicker and simpler to use the Find Tool on the Edit Bar.

The bytes to search for can either be entered in hex or as a string (ASCII, Unicode or EBCDIC). For more information see the Find dialog.

Shortcuts

Keys: Ctrl+F

Find Backward command

This command searches backward from the current cursor position for the current search bytes. This command is useful for repeatedly searching for the same text. The current search bytes are displayed in the Find Tool in the Edit Bar.

If the search text is a string an EBCDIC search is performed if EBCDIC mode is currently on else an ASCII search is performed. To perform a Unicode search use the Find command.

The search starts at the beginning of the current selection. So for repeated backward searches just keep pressing Shift+F3.

To search forwards see the Find Forward command.

Shortcuts

Keys: Shift+F3

Toolbar: 

Find Forward command

This command searches forward from the current cursor position for the current search bytes. This command is useful for repeatedly searching for the same text. The current search bytes are displayed in the Find Tool in the Edit Bar.

If the search text is a string an EBCDIC search is performed if EBCDIC mode is currently on else an ASCII search is performed. To perform a Unicode search use the Find command.

The search starts at the end of the current selection. So for repeated searches just keep pressing F3, since the found bytes are automatically selected. This does mean that if the search patterns overlap not all will be detected. For example, searching for “BAB” in “BABABAB” will only find two occurrences.

To search backwards see the Find Backward command.

Shortcuts

Keys: F3

Toolbar: 

Find Selection command

This command searches forward from the current cursor position for the next occurrence of the currently selected bytes. This command is unavailable if there is no data currently selected.

After executing this command the selected bytes are added to the Find Tool in the Edit Bar and to the search history drop-down list. This allows you to easily repeat a search either forwards (see the Find Forward command) or backwards (see the Find Backward command).

Shortcuts

Keys: Ctrl+F3

Mark Position command (Edit menu)

This command sets the mark at the current cursor position. This position can be in the hex area or the character area.

The mark allows you to quickly jump to previous position by marking it and at a later time using the Go To Mark command to return to that position. Note that the mark is attached to a byte not an address – if bytes are inserted before the mark then the mark shifts to a new address along with the byte it is attached to.

The mark is also useful for measuring distances between two points in a file. Two indicators in the Status Bar show the hex and decimal "Distance to Mark" from the current cursor position.

Shortcuts

Mouse: Double click in the hex or character display area

Toolbar: 

Go to Mark command (Edit menu)

This command returns the cursor to the current position of the mark. If the mark was located in the character area but the character area is no longer being displayed then the cursor is positioned at the byte in the hex area with the same address.

See the Mark Position command for how to set the mark.

Shortcuts

Keys: Ctrl+F9

Toolbar: 

Extend to Mark command

This command selects all the bytes between the current cursor position and the mark, or moves the end of the selection if there already is one.

See the Mark Position command for how to set the mark.

Shortcuts

Keys: Shift+Ctrl+F9

Swap Cursor and Mark command

This command moves the cursor to the mark (like the Go to Mark command) but at the same time moves the mark to where the cursor was. This is particularly useful in keystroke macros.

See the Mark Position command for how to set the mark.

Shortcuts

Keys: Shift+F9

Allow Changes Toggle (Edit menu)

This command allows you to change between read-only mode and read-write mode. Note that this mode is applied independently to different windows, so you can have two windows on the same file in different states.

This is not to be confused with a read only file. A file may be read only if it is a system or read-only file, if modifications to the file are disallowed by another application that is also using the file, or if you opened the file using the Read Only checkbox in the File Open dialog. If the file is read-you cannot turn off read-only mode.

Shortcuts

Mouse: Double click the RO/RW indicator on the status bar

Toolbar: 

Insert Mode Toggle (Edit menu)

This command allows you to change between insert and overtype modes.

In insert mode the length of the file may change as you edit the file. That is, if you type values into the hex or character area in a window then the bytes following the cursor position are moved to higher addresses to make room for the new data. If you backspace then bytes behind the cursor are deleted and everything after the cursor is moved down. Deleting, cutting and pasting etc also shift bytes and change the length of the file.

In overtype mode existing bytes are never moved from their current address and the length of the file does not change. When you enter bytes the existing values are overwritten. When you backspace the bytes backspaced over are replaced with spaces if editing in the character area (ASCII or EBCDIC spaces depending on the display mode) or null bytes (00) if editing in the hex area.

Another consideration is that when you go to write a very large file where bytes have been inserted or deleted, HexEdit must make a copy of the file rather than changing the bytes in place. If you stay in overtype mode (and do not select the option to create backup files) then you can be sure HexEdit can save the data in place. Otherwise, HexEdit has to copy the whole file before renaming it and deleting the original, which can be slow and even impossible if your free disk space is less than the size of the file.

Shortcuts

Mouse:	Double click the OVR/INS indicator on the status bar
Keys:	Ins

Toolbar command (View menu)

Use this command to toggle on or off the display of the Toolbar. The Toolbar and the Edit Bar include buttons for many common commands in Hex Edit. A check mark appears next to the menu item when the Toolbar is displayed.

See Toolbar for help on using the toolbar.

Edit Bar command (View menu)

Use this command to toggle on or off the display of the Edit Bar. The Toolbar and the Edit Bar include buttons for many common commands in Hex Edit. A check mark appears next to the menu item when the Toolbar is displayed.

See Edit Bar for help on using the Edit Bar.

Status Bar command (View menu)

Use this command to toggle the display of the Status Bar. The Status Bar contains several *indicators* which show the status of the display and certain keyboard states (CAPS and NUM lock). The left end of the status bar is used to display status messages. In particular, it describes the action to be executed by the selected menu item or depressed toolbar button.

A check mark appears next to the menu item when the Status Bar is displayed.

See Status Bar for help on using the status bar.

Auto Fit command (View menu)

Toggles Autofit mode on and off. When autofit mode is on the columns for the window are adjusted when the window is resized so that the number of columns just fits across the window.

Shortcuts

Toolbar: 

Left Side command

Use this command to move the current column to the left side of the display. Doing this does not change the file in any way, but just changes the columns in which the bytes of the file are displayed. This is useful for analysing bytes in a file – for example, you might like to use this command when you encounter the start of a new record or structure.

This command simply changes the value of the window "offset". See the Window Display page for more information.

Shortcuts

Keys: Enter

Font command (View menu)

Use this command to change the font that is used to display the hex and character areas in the active window. This command brings up the standard font Choose Font dialog to allow you to select any available font in any size and style.

To change the size of the display font you can more easily use the Toolbar buttons for the Increase Font Size command or the Decrease Font Size command.

Increase Font Size command

Increases the font size to the next available size, without changing the font name or style. To change all aspects of the current display font use the Font command.

Shortcuts

Toolbar: 

Decrease Font Size command

Decreases the font size to the next available size, if any, without changing the font name or style. To change all aspects of the current display font use the Font command.

Shortcuts

Toolbar: 

Decimal Addresses command (View menu)

Toggles between displaying decimal addresses and hex addresses for the active window. To allow you to more easily notice the difference, hex addresses are displayed with leading zeroes but decimal addresses are displayed with leading spaces.

If the ‘‘Digit Grouping in Addresses’’ global option is on (see General Options page) then hex addresses are displayed in groups of 4 digits and decimal addresses are displayed with ‘‘thousands separators’’ (normally a comma separating groups of 3 digits in English speaking countries).

Shortcuts

Toolbar: 

Keys: Alt+D

Character Display Toggle command (View menu)

This toggles the display of the character area on the right. (The hex display area to the left cannot be turned off.) Generally, you would leave the character area displayed unless you wanted more room for the hex area.

Shortcuts

Toolbar: 

EBCDIC Characters Toggle (View menu)

This toggles the character area between displaying ASCII and EBCDIC characters. This command is not available if the character area is not displayed (see Character Display Toggle). EBCDIC control characters and invalid EBCDIC characters are displayed as red dots.

When EBCDIC mode is on the Control Characters Toggle and the Graphic Characters Toggle are not available. This command also affects how string searches are performed (see Find Tool) and how text is copied to the clipboard (see Copy command). HexEdit will tell you if you try to type an invalid EBCDIC character into the character area.

Shortcuts

Toolbar: 

Control Characters Toggle (View menu)

This toggles the display of ASCII control characters. This command is not available in EBCDIC mode (see EBCDIC Characters Toggle). When off, control characters are displayed as red dots. When on, control characters are displayed as a red character – for example character zero is a red "@", character one (Ctrl+A) is a red "A" etc.

The Toolbar button actually toggles between *three* states. The extra state displays the control characters as the C language escaped character in red – for example, linefeed is "n", tab "t" etc including character zero as a red zero (0).

Shortcuts

Toolbar: 

Graphic Characters Toggle (View menu)

This toggles the display of *graphic* characters, ie, characters not in the ASCII character set, ie, with values above 127. This command is not available in EBCDIC mode (see EBCDIC Characters Toggle).

When the graphic characters are on, the characters displayed for byte values above 127 depends on the font being used (see the Font command to change the current font) but is typically some sort of graphic character, or characters used in Western European languages. When off, these characters are displayed as red dots. When both the graphic characters toggle and the control characters toggle are on (see Control Characters Toggle) then there are no red dots – any dots are black and represent ASCII character 46 (full stop or period).

Shortcuts

Toolbar: 

Keys: Alt+T

Options command

This displays the Options dialog, which allows you to change the behaviour of HexEdit.

This dialog has the following pages:

General Options Page	allows you to change how HexEdit behaves and looks globally
Macro Options Page	allows you to change how macros are played
Colours Options Page	allows you to change the colour in which different bytes are displayed
Window Display Page	allows you to change some settings for the active window
Window Defaults Page	determines how a new window is displayed

Properties command

This displays the Properties dialog, which shows file information and values for the byte(s) at the cursor position in the currently active window. The dialog remains active even while editing (it is "modeless" in Windows jargon) – this allows you to easily see the properties as you move the cursor within a file or when you change to another window.

This dialog has the following pages:

File Properties	the name, type, location, size etc of the file
Character Properties	value in octal, binary, etc and ASCII, EBCDIC, Unicode representation
Decimal Values	value of word, double word etc, signed or unsigned, little- or big-endian
Floating-Point Values	IEEE floating-point values, 32 bit or 64 bit, little- or big-endian
IBM Floating-Point	IBM floating-point values, 32 bit or 64 bit, little- or big-endian

Shortcuts

Mouse: Double click the "Hex, Decimal, Octal, Binary, Character" indicator of the status bar

Find Hex command

This command brings up the find dialog box in Hex mode. (It is sometimes easier to use the Find Tool on the Edit Bar.) For more information see the Find dialog.

Shortcuts

Keys: Alt+S

Compare Windows command (Tools menu)

This commands compares bytes in two windows starting at the current cursor position. The two windows can be for different files or they can be two windows on the same file.

If no difference is found between the windows (by comparing to the end of both files) then a message is displayed to that effect and the cursor remains at the same position in both windows. If a difference is found then a message is displayed in the status bar and the offending bytes are selected in both windows (or in just one window if the other encountered end of file). You can then immediately do another compare, which will start at the bytes after the selected ones.

This command requires two windows. For this reason the command is only available if there are exactly two non-minimised windows *or* there are just two windows open (whether minimised or not). If you wish to compare two currently open windows you must close or minimise all other windows.

When comparing windows it is useful to see them side-by-side. To quickly do this use the Tile Vertically command.

Shortcuts

Keys: Alt+C

Increment Byte command (Tools menu)

This command adds one to the byte at the cursor position. This command is not available if there is no active window or the cursor is positioned at the end of the file.

Repeated use of this command cycles through all byte values. When the (unsigned) value 255 is incremented it will wrap around to zero. (If you consider it to be a signed value, then it wraps from 127 to −128.)

Also see the Decrement Byte command.

Shortcuts

Keys: Num-+ ("+" key on numeric keypad)

Increment Word command (Tools menu)

This command adds one to the word (2 bytes) after the cursor. This command is not available if the cursor is too close to the end of the file.

The values are assumed to be in the normal Intel format with the least significant byte first (little-endian). However, to increment using big-endian format use the Flip Word command before and after this command. The maximum value (65535) wraps around to zero when incremented.

Also see the Decrement Word command.

Increment Double Word command (Tools menu)

This command adds one to the double-word (4 bytes) after the cursor. This command is not available if the cursor is too close to the end of the file.

The values are assumed to be in the normal Intel format with the least significant byte first (little-endian). However, to increment using big-endian format use the Flip Double Word command before and after this command. The maximum value (2^32 – 1) wraps around to zero when incremented.

Also see the Decrement Double Word command.

Increment Quad Word command (Tools menu)

This command adds one to the quad-word (8 bytes) after the cursor. This command is not available if the cursor is too close to the end of the file.

The values are assumed to be in the normal Intel format with the least significant byte first (little-endian). However, to increment using big-endian format use the Flip Quad Word command before and after this command. The maximum value (2^64 – 1) wraps around to zero when incremented.

Also see the Decrement Quad Word command.

Decrement Byte command (Tools menu)

This command subtracts one from the byte at the cursor position. This command is not available if there is no active window or the cursor is positioned at the end of the file.

Repeated use of this command cycles through all byte values. When the value zero is decremented it will wrap around to 255. (If you consider it to be a signed value, then it wraps from −128 to 127.)

Also see the Increment Byte command.

Shortcuts

Keys: Num-minus ("-" key on numeric keypad)

Decrement Word command (Tools menu)

This command subtracts one from the word (2 bytes) after the cursor. This command is not available if the cursor is too close to the end of the file.

The values are assumed to be in the normal Intel format with the least significant byte first (little-endian). However, to increment using big-endian format use the Flip Word command before and after this command. Zero wraps around to the maximum value (65535) when decremented.

Also see the Increment Word command.

Decrement Double Word command (Tools menu)

This command subtracts one from the double-word (4 bytes) after the cursor. This command is not available if the cursor is too close to the end of the file.

The values are assumed to be in the normal Intel format with the least significant byte first (little-endian). However, to increment using big-endian format use the Flip Double Word command before and after this command. Zero wraps around to the maximum value (2^32 - 1) when decremented.

Also see the Increment Double Word command.

Decrement Quad Word command (Tools menu)

This command subtracts one from the quad-word (8 bytes) after the cursor. This command is not available if the cursor is too close to the end of the file.

The values are assumed to be in the normal Intel format with the least significant byte first (little-endian). However, to increment using big-endian format use the Flip Quad Word command before and after this command. Zero wraps around to the maximum value (2^64 - 1) when decremented.

Also see the Increment Quad Word command.

Flip Word command (Tools menu)

This command reverses the position of the two bytes following the cursor. This command is not available if there is no active window or there are not two bytes before the end of file. Flipping the order of the bytes in the word allows you to switch between big-endian and little-endian number formats. The order of the bits within each byte is not affected.

Flip Double Word command (Tools menu)

This command reverses the order of the four bytes following the cursor. This command is not available of there is no active window or there are not four bytes before the end of file. Flipping the order of the bytes in the double word allows you to switch between big-endian and little-endian number formats. The order of the bits within each byte is not affected.

Flip Quad Word command (Tools menu)

This command reverses the order of the eight bytes following the cursor. This command is not available of there is no active window or there are not eight bytes before the end of file. Flipping the order of the bytes in the quad word allows you to switch between big-endian and little-endian number formats. The order of the bits within each byte is not affected.

New Window command (Window menu)

Use this command to open a new window with the same contents as the active window. You can open multiple windows to display different parts of a file (and in different ways) at the same time. Each window has its only display properties (font, number of columns, etc) as well as its own overtype/insert and read-only/read-write modes. However, if you change the bytes of the file in one window, all other windows containing the same file reflect those changes.

Record/Stop recording command (Tools menu)

Use this command to start or stop recording a keystroke macro.

As there is only one keystroke macro in HexEdit at any one time the Macro Record and Macro Play commands cannot be recorded. That is you cannot play a macro recursively, and you cannot start recording a new macro while running the current one. (Selecting the Record Macro command while recording just stops macro recording).

If you accidentally start recording you can avoid loosing the previous macro by immediately turning off recording before pressing any keys. This also means that you can't erase the current macro except by replacing it with something innocuous.

Note that settings changed while in the Options dialog are not recorded except for those in the Window Display page. (This may change in the future.) In this page only the settings that are changed are recorded as part of the macro.

To play the recorded macro see the Play command.

Shortcuts

Toolbar: 
Mouse Double click the REC indicator on the status bar
Keys: F7

Play command (Tools menu)

Use this command to replay the last macro recorded with the Record command.

If an error occurs while the macro is playing the macro stops, preventing unintentional changes to the file. You can set the severity of the error that causes the macro to stop playing in the Macro page of the Options dialog. You can also use this page to set when and how often the display is refreshed during macro playback.

Note that during playback user-interaction is not usually required but may be necessary at times. For example, if you attempt to insert or delete bytes in a window that is in overtype mode HexEdit will prompt you if you want to turn off overtype mode. (If you elect **not** to turn off overtype mode an error is generated and the macro stops playing.)

To record a macro see the Record command.

Shortcuts

Toolbar: 
Keys: F8

Multi-Play command (Tools menu)

Use this command to play the macro several times repeatedly.

This brings up a dialog that allows you to enter the number of plays. Type a number and press **Enter** to start the macro playing. This dialog also allows you to easily change playback settings using the "Play Options …" button which invokes the Macro page of the Options Dialog.

For more information on playing macros see the Play command.

To record a macro see the Record command.

Shortcuts

Keys: F9

Next Window command (Window menu)

Use this command to easily cycle through all the non-minimised windows. Unlike the Next Window command on the window's control menu, this skips minimised windows or windows in Print Preview mode.

Shortcuts

Keys: F2, Alt+N

Cascade command (Window menu)

Use this command to arrange multiple opened windows in an overlapped fashion.

Tile Horizontally command (Window menu)

Use this command to arrange the non-minimised windows in a non-overlapped manner, stacked on top of each other.

Tile Vertically command (Window menu)

Use this command to arrange the non-minimised windows in a non-overlapped manner, side by side.

Arrange Icons command (Window menu)

Use this command to arrange the icons for minimised windows at the bottom of the main window. If there is an open window at the bottom of the main HexEdit window, then some or all of the icons may not be visible because they will be underneath it.

1, 2, … command (Window menu)

HexEdit displays a list of currently open windows at the bottom of the Window menu. A check mark appears in front of the name of the active window. Choose a window from this list to make its window active. You can also flip through all the open windows with the Next Window command (Ctrl+F6).

Shortcuts

Mouse: Click on the desired window to activate it.

Help Topics command (Help menu)

Use this command to invoke the main Help window. You can then select from three pages: Contents, Index or Find.

The Contents page displays headings and subheadings, like chapters in a book, that allow you to get an overview and quickly locate information if you know where it should be. You can use this to jump to any help topic or reference information.

The Index page allows you to look up subjects based on keywords stored in the help topics. Use this if you know what you're looking for but don't know where in the Contents hierarchy it is located.

The Find page lets you search the entire help file for any word. This is a poor man's index. Only use this if you find that the keywords I put in the Index page don't cut it (and email me where I went wrong).

Help Index command (Help menu)

Use this command to display the opening help topic. From there you can jump to step-by-step instructions for using HexEdit and various types of reference information.

Shortcuts

Keys: F1

Using Help command (Help menu)

Use this command for instructions about using Help.

Email Support command (Help menu)

Use this command to send email to the author of HexEdit. This command is disabled if your system does not support MAPI. If your system supports MAPI but you never (again) connect to the Internet then there is not much point in using this command.

I greatly appreciate any feedback that can improve the program. I will do my best to support the product and respond to you within my time constraints.

For the latest information on bugs, enhancements and future directions visit the HexEdit web site at **http://members.tripod.com/AndrewPhillips**. (If you can't find the web site there, it may have moved – email me at the address below for the new location.)

To be sure of a prompt response address email to HexEditSup@hotmail.com. Should this bounce try me at aphillip@geocities.com (redirects to my current email address), aphillips@one.net.au (current home address) or andrew@encom.com.au (work).

For more information see the Email HexEdit Support dialog.

Tip of the Day command (Help menu)

Use this command to display the tip of the day, and to toggle whether tips are shown when the program starts up. This shows the same dialog that you see when you start HexEdit (and "Show Tips on Startup" is on).

By default "Show Tips on Startup" is on. This means that every time you run HexEdit a new tip is shown. These tips are taken sequentially from the (ASCII text) tip file HexEdit.tip. When all the tips have been shown the tips start from the beginning of the file again. (If you like, you can add your own tips to this file – one per line.)

About command (Help menu)

Use this command to find out what version of HexEdit you are using.

Context Help command

Use the Context Help command to obtain help on some portion of HexEdit. The mouse pointer will change to an arrow and question mark.

Select a menu item or click somewhere in the HexEdit window, such as a toolbar button. The Help topic will be shown, if available, for whatever you select.

Shortcut

Keys: Shift+F1

Restore command (Control menu)

Use this command to return the active window to its size and position before you chose the Maximize command or Minimize command.

Shortcut

Mouse: Click the restore icon  on the title bar.

Move command (Control menu)

Use this command to display a four-headed arrow so you can move the window or dialog box using the arrow keys.

After the pointer changes to the four-headed arrow:

1. Press the DIRECTION keys (left, right, up, or down arrow key) to move the window.
2. Press ENTER when the window is at the position you want.

Note: This command is unavailable if you maximise the window.

Shortcut

Mouse: Drag the window by its title bar to move it.

Size command (Control menu)

Use this command to display a four-headed arrow so you can change the size of the window using the arrow keys.

After the pointer changes to the four-headed arrow:

1. Press one of the DIRECTION keys (left, right, up, or down arrow key) to move the pointer to the border you want to move.
2. Press a DIRECTION key to move the border.
3. Press ENTER when the window is the size you want.

Note: This command is unavailable if you maximise the window.

Shortcut

Mouse: Drag the size bars at the corners or edges of the window.

Minimize command (Control menu)

Use this command to reduce the window to an icon.

Shortcut

Mouse: Click the minimize icon  on the title bar of the window.

Maximize command (Control menu)

Use this command to enlarge the active window to fill the available space.

Shortcut

Mouse: Click the maximise icon  on the title bar; or double-click the title bar.

Close command (Control menu)

Use this command to close the active window or dialog box.

Clicking the window close box is the same as choosing the Close command.

Shortcuts

Mouse: Click the close icon  on the title bar.
Keys: Ctrl+F4 closes a window
Alt+F4 closes the main HexEdit window or a dialog box

Next Window command (Window Control menu)

Use this command to switch to the next open window. Unlike the Next Window command on the Window menu this will cycle through all open windows including minimised ones.

To go back to the previous window use the Previous Window command.

Shortcut

Keys: Ctrl+F6

Previous Window command (Window Control menu)

Use this command to switch to the previous open window. HexEdit determines which window is previous according to the order in which you opened the windows.

To go to the next window use the Next Window command or the Next Window command on the Window menu.

Shortcut

Keys: Shift+Ctrl+F6

File Open dialog

This dialog allows you to select or specify an existing file or files to open.

The folders and files in the current location are displayed. Double-clicking a folder opens it – that is, your current location moves to that folder. To open the folder one level higher (the "parent" of the current folder) click the "Up One Level" button.

Only files of the type selected in the "Files of Type" list box are displayed. You can view the files sorted by name, date & time, size etc by clicking the detailed list button and clicking the appropriate column heading to sort by that column. The dialog also allows you to delete, rename, view properties etc by clicking the file with the right mouse button and selecting from the context menu that appears.

Double-clicking a file opens the file. You can open more than one file by shift-clicking to select a contiguous group of files (click the first file with the mouse and hold the Shift key down while clicking the last). You can also select individual files by Control-clicking them. After you have selected all the files you want to open click the **Open** button.

Look In

To see how the current folder fits in the hierarchy on your computer, click the down arrow. To see what's inside a folder, click it.

File Name

Type the filename you want to open. If you type the name of a folder that folder is opened and the contents are displayed. Normally you see all files matching the specification currently selected in the "List Files of Type" box – but if you type a filename with wildcards here, such as *.DOC, you will only see matching files.

List Files of Type

Select the type of file you want to open:

All files (*.*)	All files are displayed.

Open as read-only

Select this option if you do not want to modify the file. Note that even if you do not open the file in read only mode changes to the file may still be disabled for a particular window on the file.

File Save As dialog

This dialog allows you to specify the name and location of a file you're about to save.

The folders and files in the current location are displayed. Double-clicking a folder opens it – that is, your current location moves to that folder. To open the folder one level higher (the "parent" of the current folder) click the "Up One Level" button.

Only files of the type selected in the "Save as type" list box are displayed. Double-clicking a file will attempt to replace that file – you are prompted if you want to replace it.

You can view the files sorted by name, date & time, size etc by clicking the detailed list button and clicking the appropriate column heading to sort by that column. The dialog also allows you to delete, rename, view properties etc by clicking the file with the right mouse button and selecting from the context menu that appears.

Save In

To see how the current folder fits in the hierarchy on your computer, click the down arrow. To see what's inside a folder, click it.

File Name

Type the filename you want to save to. If you type the name of an existing file you are prompted if you want to replace it. If you type the name of a folder that folder is opened and the contents are displayed. Normally you see all files matching the specification currently selected in the "Save as type" box – but if you type a filename with wildcards here, such as *.DOC, you will only see matching files.

Save As Type

Select the type of file you want to save:

All files (*.*)	All existing files are displayed.

Print Dialog

The following options allow you to specify how the file should be printed:

Name

Select a printer from the drop down list to change where the printing is performed.

Properties

Brings up a printer specific dialog box that allows you to change additional settings that are specific to the current printer. This will allow you to set simple things like the paper size, source and orientation. (If not you can set these things using the Print Setup command on the File menu.) There may be other options specific to the selected printer such as print quality. Some printers may even allow you to perform tasks like cleaning ink jet nozzles or finding out how much ink remains in the ink cartridge currently in the printer.

Print To File

Select this to save the information (that would have been sent to the printer) to a file. Note that this is not just the text to be printed but also formatting commands specific to the currently selected printer. You can use this option if you don't currently have access to the specific printer but want to print on another machine or at another time. One way to print the file would be to use the MSDOS command: "COPY /B printfile LPT1:".

Print Range

Specify the pages you want to print:

All	Prints the entire file.
Pages	Prints the range of pages you specify in the From and To boxes.
Selection	Prints the currently selected bytes (if any).

Copies

Specify the number of copies you want to print for the above page range.

Collate Copies

Prints copies in page number order, instead of separated multiple copies of each page.

Print Progress Dialog

The Printing dialog box is shown while HexEdit is sending output to the printer. The page number indicates the progress of the printing.

To abort printing, choose Cancel.

Print Setup dialog box

The following options allow you to select the destination printer and its connection.

Printer

Select the printer you want to use. Choose the Default Printer; or choose the Specific Printer option and select one of the current installed printers shown in the box. You install printers and configure ports using the Windows Control Panel.

Properties

Displays a dialog box where you can make additional choices about printing, specific to the type of printer you have selected. There may be options such as print quality. Some printers may even allow you to perform tasks like cleaning ink jet nozzles or finding out how much ink remains in the ink cartridge currently in the printer.

Paper Size

Select the size of paper that the file is to be printed on.

Paper Source

Some printers offer multiple trays for different paper sources. Specify the tray here.

Orientation

Choose Portrait or Landscape.

Network...

Choose this button to connect to a network printer.

Find dialog box

This dialog lets you search for a sequence of bytes in the file. (Also see the Find Tool which provides a quick way to start a search.)

Note that the Find Dialog is modeless, which means that you can leave it on the screen and go and do something else. Then when you later want to do another search it is ready and waiting. While it is waiting to be used it will get out of the way if you try to move the cursor to a position where the dialog window would hide it.

Find What

Type the hex bytes or text you want to search for or select a previous search from the list.

You can search for an ASCII, Unicode or EBCDIC string or a sequence of hex values depending on the radio button currently selected (see below). You are warned of invalid characters, such as an invalid hex digit (anything apart from 0-9, a-f, A-F) for a hex search, or an invalid EBCDIC character such as a tilde (~) for an EBCDIC search.

If you enter a search string or hex bytes and then change the search type using the Hex, ASCII, Unicode or EBCDIC radio button (see below) then the search text is automatically converted to the new type of search. Any invalid values are removed – for example the hex byte 00 cannot be converted to a valid ASCII, Unicode or EBCDIC character, and the ASCII tilde (~) cannot be converted to EBCDIC.

This conversion provides a simple way to find the hex values of ASCII, Unicode or EBCDIC characters, or to convert between character sets. For example, suppose you want to find out what is the equivalent in EBCDIC of the ASCII character with value 41 (hex). To do this: select the Hex radio button and type "41" into the **Find What** box; select the ASCII radio button (the **Find What** box now contains "A"); select EBCDIC; then select Hex and the **Find What** box displays the value C1, which is the EBCDIC value for a capital A.

The drop down list provides a history of the 16 most recent searches. This is the same list as can be seen in the Find Tool on the Edit Bar. Note that, although not shown, this list stores the type of search, case-sensitivity etc, which is restored when you click on an element in the list.

The search algorithm used is faster the longer the sequence of (unique) bytes entered. So entering a search string of "ABCDEFGHIJKLMN" is better than "ABC" even if "ABC" is sufficient to find what you are looking for. But entering "ABCXDEFXGHIXJKLX" would be no faster than "ABCX", because of the repeated "X".

Hex

This option selects a hex search as opposed to a string search (ASCII, Unicode or EBCDIC).

After selecting this type of search you can enter a sequence of hex digits in the **Find What** box which represent the bytes to search for. When you select the Hex radio button the text in the **Find What** box (if any) is automatically converted from the previous string type to the corresponding hex bytes.

ASCII

The string in the **Find What** box is treated as an ASCII string (even if the active window is currently in EBCDIC mode).

Unicode

The string in the **Find What** box is treated as Unicode. Each character represents two bytes of the search sequence equivalent to the (little-endian) Unicode value.

EBCDIC

The string in the **Find What** box is treated as an EBCDIC string (even if the active window is currently

in ASCII mode).

Match Case

Select this option if the string search is to be case-sensitive. (This option is not available for a Hex search.)

It is usually best *not* to select this option. It is easy to forget the case of a word you are searching for, and sometimes a case-insensitive search can find related text that might have been overlooked. The disadvantage of case-insensitive searches is that they may be a little slower.

Direction

Specifies which direction to search starting from the current cursor position. Choose Up to search backward, toward the beginning of the file. Choose Down to search forward, toward the end of the file.

Find Next

Searches for the next instance of the text you specified in the **Find What** box. The search proceeds from the current cursor position towards the end of file if you selected the **Down** button in the Direction box, or towards the start of file if you selected the **Up** button.

As the search proceeds the current file position is updated in the Hex and Decimal Address Tools on the Edit Bar. This is useful, in a big file, to see the progress of the search and to see that HexEdit is still working.

If the search is successful the found bytes are automatically selected. If the search fails then you will be warned and the cursor position does not change.

Cancel

Closes the Find dialog box without performing a search.

Choose Font dialog box

This dialog allows you to change the font, its style, size etc for the active window.

Font

Type the name of a font or select a font from the list of available fonts. You can choose any font but for HexEdit a proportional font (where all characters are the same width) works best.

Font style

Type the name of a style or select from the list of styles supported by the current font. Most fonts support the regular style plus *bold* and *italic*. Other styles such as *light*, *heavy* etc are possible.

Size

Type a point size or select one from the list. Truetype fonts are scalable to any size; for other fonts only use values that are displayed in the list of available sizes.

A quicker way to change the font size is to use the Toolbar buttons for the Increase Font Size command and the Decrease Font Size command.

Script

Lists the available language scripts for the specified font. Pick the one appropriate for the language your computer is set up for.

Email HexEdit Support dialog

This dialog allows you to send feedback to the creator of HexEdit. You can actually use this dialog to send email to anyone if you want to.

To

The email address to send to. The default address is best to get a prompt and relevant reply. Select another address from the drop down list if your email bounces (ie. is returned unopened). You can actually type in any address and even send email to your mother, but I don't think she's interested in HexEdit bug reports.

Subject

Type a brief description of what this email is about.

Type

Selecting what type of message you are sending will help my email administration. If you are reporting a bug see the instructions below.

Description

This is where you enter the main text of the message. If you would like to make a suggestion about an enhancement to HexEdit, or just tell me you like the program, I would be very happy to hear from you.

If think you have discovered a bug please do the following:

- Select the "Report bug" radio button.
- Enter a brief description of the problem in the "Subject" box.
- In the "Description" box try to describe the problem as fully as possible.
- Try to remember exactly what you did and what happened.
- List the steps required to reproduce the problem if you can reproduce it.
- Include the exact text of any error messages from HexEdit or the system.
- Try to think of anything peculiar to you configuration that may be relevant to the problem.
- Include your email address(es) so I can get back to you for more information.

System

Select the type of system the bug was detected on. Leave the default value of "This system" to let HexEdit find out for itself.

Version

If the bug occurs in a version of HexEdit different to the one you are now using, select a different version from the list, or type in the version number if it occurs on a later version.

Your Name

Tell me your name if you want to.

Email Address

List email address(es) here, in case I need to get back to you for more info. Email addresses are of the form "name@address", eg. HexEditSup@hotmail.com or aphillips@one.net.au.

Send

Click the "Send" button to send the mail. The mail will be automatically sent if you have a MAPI session open – for example, if OutLook is currently running on your machine.

If you do not have a MAPI session running you will be prompted to select a profile to use. If you don't know what this means then you probably should just use the default profile, ie. hit the OK button. If the profile you selected is password protected then you will then be prompted for a password.

Note that if you do not have a permanent Internet connection the mail is not really "sent" until the next time you exchange mail with your mail server. (For example, when you use the Tools/Check for New

Mail menu item in Outlook.) If you never connect to the Internet then the mail will never be sent.

File Properties Page

This page shows information from the disk file associated with the active window. If the file has not yet been saved most of these fields will be blank. The exception is the **Size** field, which is the length of the file "in memory".

Name
The disk file name of the file in the active window.

Type
The type of the file according to the system registry. "None" is displayed if the filename has no extension. "Unknown" is displayed if the file type is not registered.

Location
The complete path of the folder (directory) where the file is stored.

Modified
This is the modified time of the disk file. If you have saved the file since opening it within HexEdit then this is the time that you last saved it. If you have not saved the file since opening it, this is the time that it was most recently modified before then.

Size
This is the size (in memory) of the file. This may be different to the size of the disk file if there are unsaved changes that inserted or deleted bytes from the file.

Read only
Is the file marked read only?

Hidden
Is the file marked as hidden?

System
Is the file marked as a system file?

Character Properties Page

This page shows information about the byte at the current cursor position of the active window. (Although the **Unicode** field uses two bytes starting at the cursor position.)

Hex
The hexadecimal value of the byte from 00 to FF.

Decimal
The decimal value of the byte from 0 to 255.

Octal
The octal value of the byte from 000 to 377.

Binary
The binary value from 00000000 to 11111111. This field is mainly useful for examining the individual bits of the byte. "0" means the bit is off, "1" means it's on.

ASCII
The ASCII character at the current cursor position. If the high bit is set (ie, the decimal value is greater than 127) then this is actually the ANSI character. For values less than 32, this field displays the name of the ASCII control character.

Unicode
This is the Unicode character representing the two bytes following the cursor. This field is available under NT but disabled under Windows 95 (which does not support the display of Unicode characters).

EBCDIC
The EBCDIC character at the current cursor position. The EBCDIC control character name is displayed for values less than 64, and "none" is displayed for invalid EBCDIC characters.

Decimal Values Page

This page shows the decimal values of the byte, word, double word and quad word of the data following the current cursor position in the active window. Some or all of these fields may be blank if there are not enough bytes before the end of file.

These values are useful if the file you are using is used to store binary numbers, and you need to see the actual (decimal) values of those numbers. These values can be displayed as signed or unsigned and as big-endian or little-endian depending on the check box options chosen. The big-endian option is useful for viewing files from certain "foreign" sources such as many UNIX systems.

Byte

The decimal value of the byte. This field may show values in the range 0 to 255 (unsigned) or –128 to 127 (signed).

16 bit

The decimal value of the two bytes following the cursor. This field may show values in the range 0 to 65535 (unsigned) or –32768 to 32767 (signed).

32 bit

The decimal value of the four bytes following the cursor. This field may show values in the range 0 to 2^32-1 (unsigned) or –2^31 to 2^31-1 (signed).

64 bit

The decimal value of the eight bytes following the cursor. This field may show values in the range 0 to 2^64-1 (unsigned) or –2^63 to 2^63-1 (signed).

Big endian

Select this option to view the numbers with big-endian byte order. When selected the most significant byte is considered to be the left-most byte (ie, the byte with the lowest address). This option is off by default since data created on MSDOS/Windows systems is typically little-endian.

Unsigned

Select this option to view the numbers as being unsigned.

2's complement

Select this option to view the numbers as being signed in **2's complement** notation. Negative numbers have the high bit on – to obtain the inverse value flip all the bits and add one. The high bit is the most significant bit of the most significant byte (see the **Big endian** option above).

You will typically use this option for signed numbers as the other signed formats are not used in an Intel/Windows environment.

1's complement

Select this option to view the numbers as being signed in **1's complement** notation. Negative numbers have the high bit on. Invert all the bits to find the corresponding positive number.

Sign and magnituide

Select this option to view the numbers as being signed in **sign and magnitude** notation. Negative numbers are the same as positive numbers but with the high bit on.

Floating-Point Values Page

This page is used to view the bytes following the current cursor position as IEEE floating-point numbers. You can view the value for the standard 32 bit and 64 bit IEEE formats in either big-endian or little-endian format. If viewing the 32 bit value the fields are blank unless there are at least 4 bytes before the end of file. Similarly a 64 bit value requires 8 bytes.

The exact format of a 32 bit value is a sign bit as the highest bit, followed by 8 exponent bits and 23 mantissa bits. For a 64 bit value there is a sign bit, 11 exponent bits and 52 mantissa bits. There is an implicit leading 1 bit on the mantissa so that all mantissa values are between 0.5 and 1.0. The exponent is binary using the exponent bits as an unsigned number and subtracting a bias (127 for 32 bit values or 1023 for 64 bit values). Using these numbers the value of the floating-point number = mantissa * 2^exponent (where I have included the sign bit in the mantissa and subtracted the bias from the exponent).

Note that in the range of exponents, two values are not used for "normal" numbers – those with all bits off and those with all bits on. If the exponent has all bits off then the number is zero (if the mantissa has all bits zero) or a denormalised number which is close to zero (if the mantissa does not have all bits zero). This means there are two values for zero, with the sign bit on or off, by convention zero is normally positive (giving a bit pattern with all bits off). Denormalised numbers are values closer to zero than the smallest "normal" numbers – they are used to avoid sudden underflow conditions caused by the "gap" between numbers with the smallest exponent and zero.

An exponent with all bits on and a mantissa of zero represents infinity; the sign bit determines whether it's +/- infinity. (The only other bit-patterns are: the exponent has all bits on but the mantissa does not have all bits off – the value is not a number or NaN). For example, the bytes 00 00 80 FF represent a 32 bit IEE negative infinity (or FF 80 00 00 in big-endian byte-order).

Value

This is the (approximate) value of the IEEE floating-point number. For numbers with very big or small exponents the display will contain a (signed) trailing decimal exponent preceded by the letter "e".

Note that the number may not be exact since the display shows a decimal (base 10) number whereas IEEE floating-point numbers use a binary exponent. Hence many values cannot be represented exactly as a decimal number.

Mantissa

For "normalised" numbers the mantissa is a positive or negative value between 0.5 and 1.0. As the values approach zero (from both the positive and negative direction) – that is, the value's exponent becomes more negative – "denormalised" numbers are allowed which have a smaller mantissa.

Exponent

This is the (power 2) exponent. For normal numbers it is an integer from -127 to 126 for 32 bit values or –1023 to 1022 for 64 bit values. It is –128 (or –1024) for zero and small denormalised values and 127 (or 1023) for infinity and NaNs.

Big endian

Select this option to view the numbers with big-endian byte order. When selected the sign bit is in the left-most byte (ie, the byte with the lowest address). This option is off by default since data created on MSDOS/Windows systems is typically little-endian.

32 bit

Select this option to view the value of the IEEE 32 bit (4 byte) number.

64 bit

Select this option to view the value of the IEEE 64 bit (8 byte) number.

IBM Floating-Point Values Page

This page is used to view the bytes following the current cursor position as IBM floating-point numbers. This numeric format is used in IBM mainframe and mini-computers and in many binary file formats. You can view 32 bit or 64 bit floating point numbers in big- or little-endian byte order. The byte order is typically big-endian so this is the default. If viewing the 32 bit value the fields are blank unless there are at least 4 bytes before the end of file. Similarly a 64 bit value requires 8 bytes.

Both formats have a sign bit and a seven-bit exponent in the most significant byte. The remaining bits (24 or 56) are used for the mantissa. The exponent is a hexadecimal exponent (with a bias of 64) which is equivalent to a nine bit binary exponent (since 16^128 = 2^512). The mantissa in a normalised number is between 1.0 and 1/16 (0.0625), although non-normalised values are allowed. Thus a 32 bit value is 16^(exponent-64) * mantissa/2^24 (and -ve if the sign bit is on).

There is also a 128 bit format but this is simply two consecutive 64 bit values which are simply added together. The 2nd of the pair of values has an exponent which is always 14 less than the exponent in the 1st value which corresponds to 56 mantissa bits (16^14 = 2^56). Hence this format has the same range of exponents as the 64 bit (and 32 bit) format but doubles the significant digits of the mantissa.

Value

This is the (approximate) value of the IBM floating-point number. For numbers with very big or small exponents the display will contain a (signed) trailing decimal exponent preceded by the letter "e".

Note that the number may not be exact since the display shows a decimal (base 10) number whereas IBM floating-point numbers use a hexadecimal exponent. Hence many values cannot be represented exactly as a decimal number.

Mantissa

For normalised numbers the mantissa is a positive or negative value between 0.0625 and 1.0.

Exponent

This is the exponent (power 16), which is a small, integer between –64 and 63.

Big endian

Select this option to view the numbers with big-endian byte order. The sign and exponent are in the left-most byte (ie. the byte with the lowest address). Normally you should leave this option selected since IBM floating point numbers are typically stored with big-endian byte order.

32 bit

Select this option to view the value of the IBM 32 bit (4 byte) number.

64 bit

Select this option to view the value of the IBM 64 bit (8 byte) number.

General Options Page

The General options page allows you to change how HexEdit behaves and looks globally.

Restore main HexEdit window on startup

The main HexEdit window is restored to its position when you last ran the program. For this option to have any effect the settings must be saved (see the **Save settings** options below) since it is only effective when the HexEdit starts up.

Display hex digits in uppercase

When this option is selected the digits A to F in all hex numbers are displayed in uppercase. This affects all windows, dialogs, the Find Tool, the Hex Address Tool, etc. This option is recommended as there is less chance of confusing certain digits (such as "6" and lowercase "b"), depending somewhat on the font you are using.

Digit grouping in addresses

This option changes the display of addresses in all open windows. Hex addresses are displayed as space-separated groups of four digits. The effect on decimal addresses is locale-dependent but they are usually grouped in threes with a "thousands-separator" of a comma for most English-speaking locales.

Create backup files

If this option is selected then whenever you save a file the previous version of the file, if any, is retained with the file extension of ".BAK". (If you are editing a file with extension of ".BAK" the backup file has an extension of ".BAC"). The previous backup file, if any, is discarded.

If you edit large binary files then it may be better to leave this option unselected, unless you have plenty of disk space.

Save settings on exit

Selecting this option forces all the current global settings (in this page and the Macro Page and Window Defaults Page) to be saved when HexEdit is closed. Other less obvious things are also saved like the most recent positions of toolbars and modeless dialogs. A simpler way to customise your options is to set them how you want them and then use the **Save settings now** button (see below).

The storage for these options is in the file "HexEdit.ini" in the main Windows directory (usually C:\WINDOWS). A few options stored there cannot be changed from within HexEdit such as the default number of columns and column grouping in a window – you might like to modify these with a text editor.

Save settings now

Clicking this button forces all the current options to be saved immediately (even if the **Save settings on exit** option is not selected).

Macro Options Page

The Macro options page allows you to change how the keystroke macro is played. You can set how often the display is refreshed while the macro is playing and what sort of error conditions will cause it to stop prematurely.

Refresh: Never

Selecting this option causes the window display to not be refreshed while the macro is executing. This gives the fastest execution for keystroke macros since the display is not updated, but you will only see what effect your macro had after it has finished. The opposite (ie. slowest) option is to have the display updated after every recorded keystroke (see below).

Refresh: Every N seconds

Selecting this option means the display is updated after every N seconds (where N is the value in the seconds field). The refresh happens without regard to the number of keystrokes or replays of the macro.

Refresh: Every N keystrokes

Selecting this option means the display is updated after every N keystrokes (where N is the value in the keystrokes field). Using a value of one with this option means the display is updated as it would be if you were not running a macro, but it gives the slowest performance.

Refresh: Every N plays

If the macro is being played more than once this option says to update the display after every N replays (where N is the value in the replays fields).

Refresh: Include Properties Dialog

This option determines when the Properties dialog is updated. If the Properties dialog is not displayed when the macro runs then this option has no effect. (See the Properties command for how to display the Properties dialog.) When selected, the dialog is updated, as determined by the above refresh options, otherwise it is not updated till the macro finishes playing.

Refresh: Include Tools and Status Bar

Selecting this option means that status bar messages plus the Find Tool, Hex Address Tool and Decimal Address Tool are updated when the display is refreshed (as determined by the above refresh options).

Halt: All warnings and errors

This option causes the macro replay to stop prematurely if it causes any sort of error or warning condition. For example, if this option is selected and the macro includes the Read File command, but the file to be inserted has zero size, a warning is generated and the macro stops.

Other warnings include: opening a file that is read-only (the file is still opened but in read-only mode); a failed comparison (see the Compare Windows command); using invalid characters (such as non-EBCDIC characters in a search while in EBCDIC mode); etc.

Halt: Minor and major errors

This option causes the macro replay to stop prematurely if it causes a minor (or worse) error. Minor errors are situations where the command could not be completed but the problem might not be considered fatal. This is the recommended halt option.

Minor errors include such things as attempting to move past the end of file or move/backspace before the start of the file, attempting to decrease the font size below its minimum etc. Often commands cannot be completed because the window is in read-only or overtype mode – when running the macro and this happens you can elect to turn off the mode and continue, but a minor error is generated. (If instead you cancel the dialog a major error is generated.)

Halt: Major errors only

This option causes macro replay to stop if there are serious errors but to ignore all warnings and minor errors. This option is **not** recommended since many minor errors may cause your macro to

behave unexpectedly.

Major errors are any situations where a command could not complete and this could affect subsequent macro commands. Examples include: opening a non-existent file; attempting to modify a read-only file; a failed search; attempting to undo when there is nothing to undo; etc. See above for examples of minor errors.

Colour Options Page

This page allows you to display different groups of hex values in different colours. For many types of files this can be extremely useful for easily spotting patterns and irregularities. Note that any changes made only affect the currently active window. To change the colours for all windows use the **Save As Default** button (see below).

To use different colours you need to split the range of byte values (0-255) into groups. A group can consist of individual values or discontiguous ranges. For example, to use the values found in normal ASCII text files use "9,10,13,32-126", which are the values of tab, line-feed, carriage-return and the range of normal display characters. As well as the values each group has an associated name or description and, of course, a colour.

When displaying the hex byte HexEdit scans this table from the top and uses the colour of the first group the value is found in. If a value appears in more than one group the colour from the higher group is used. You can use the arrow buttons to change the order of the groups. Also if a value is not assigned to any group then that value is not displayed. (Actually it's really displayed as white text on a white background.)

Name
The description of the currently selected group. You can also type the name of a new group to be added here.

List of Names
This lists the names of all the current groups. To change the colour or values of a group first select it in this list.

Colour
Select from this list to change the colour used for the current or new group.

Values
Enter the values for the current or new group. Values are in decimal and comma-separated; a range of values may be entered by typing two values separated by a dash or a colon.

The order of the groups is significant. The down arrow allows you to move the current group down in the list.

The up arrow allows you to move the current group up in the list.

Add
This adds a group to the list with the name currently in the **Name** field (see above), and the currently selected colour and range of values. You cannot Add a group with the same name as an existing group.

Remove
This removes the current group from the list. Before removing you must select the group to remove from the list.

If you accidentally click **Remove**, simply click **Add** before you do anything else. This reinserts the same group at the top of the list. You may then need to use the down arrow to reposition the group where it was.

Reset
The Reset button sets the current colours back to the factory default settings. These settings are not necessarily the same as the current default colours set with the **Save As Default** button (see below).

Save As Default
This saves the current settings to be used as the default colours for files opened thenceforth. It does not change the colours for any windows that are already opened.

Window Display Options Page

The Window Display page allows you to change options which only affect the display in the active window. (If there is no active window then this page is not available.)

Display character area

This is the same option as is controlled by the Character Display command. It controls whether there is a character display area to the right of the hex area in the active window.

Adjust number of columns to fit the display

This is the same option as is controlled by the Autofit command. When on the number of columns changes to fit the width of the window.

Number of columns

This field is the number of display columns in the active window. This field is disabled if the **Adjust number of columns to fit the display** (Autofit) option is selected.

Starting offset

This field is the column number of the first byte of the file. The value must be between zero and one less than the **Number of columns** field.

This field is normally left at zero but it is sometimes useful to shift the columns around so that patterns in the data are more obvious. The starting offset is also changed by the Left Side command.

Column grouping

This field changes the grouping of columns in the hex display area. This also sets the number of columns by which the Ctrl+Left Arrow and Ctrl+Right Arrow keys move. The default value is 4 columns.

Window Defaults Options Page

The options on this page determine how a new window looks and behaves. Note that these values are only the default settings. They can, of course, be changed for any window once it has been opened.

These default settings are applied to a newly created file. They are also used if you open a file that you have not opened before or one that is not on the recently opened file list on the File menu. If you open a file on the recently opened list HexEdit restores the options in effect when that file was closed.

Adjust columns to fit display
When this option is selected, autofit mode is on by default for new windows. Autofit mode means the number of columns is adjusted to fit the window.

Display addresses in decimal
When this option is selected, addresses are displayed in decimal (for new windows). Otherwise addresses are displayed in hex.

Display character area
When this option is selected, the character area is displayed to the right of the hex area (for new windows). That is, characters are displayed in columns corresponding to the hex values. Whether the charcaters represent ASCII or EBCDIC is determined by the next field.

Display characters in EBCDIC
When this option is selected the character display area on the right of a new window defaults to EBCDIC rather than ASCII (for new windows). This option is not available if the **Display character area** option is off.

Display controls characters
When this option is selected, ASCII control characters (those with a value less than 32) are displayed in the character area in red (by default for new windows). Character zero is displayed as a red "@", character one as a red "A", etc. If this option is not selected all control characters are displayed as a red dot.

This option is not available if the **Display character area** option is off, or the **Display characters in EBCDIC** option is on.

Display graphic characters
When this option is selected, the so-called graphic characters (those with a value greater than 127) are displayed as black characters in the character area (by default for new windows). The exact character displayed depends on the font you're using. If this option is not selected all graphic characters are displayed as a red dot.

This option is not available if the **Display character area** option is off, or the **Display characters in EBCDIC** option is on.

Read only
New windows are by default read only. The file cannot be modified through this window unless read-only mode is turned off. Note that the window will be read only despite the setting of this option if the file was opened read only.

Modifiable
The opposite of read-only mode. The file can be modified through this window.

Overtype
New windows are, by default, created in Overtype mode. Any data added to the file overwrites existing data. Backspace does not delete bytes but changes them to null bytes (in the hex area) or space (in the character area).

For most types of binary files this is the safe option, since nothing can be inserted into or deleted from the file. Many types of binary files (such as .EXE files) rely on the position of bytes within the file not

changing. Further, if you insert or delete bytes from a large file the disk file cannot be modified in place and must be copied to a temporary file before the original is deleted and it is renamed. This can be slow and increases the chances of running out of disk space.

Insert

The opposite of overtype mode. Bytes can be inserted into or deleted from the file.

Toolbar

The toolbar by default is displayed across the top of the application window, below the menu bar. The toolbar provides quick mouse access to several common tools in HexEdit. These include facilities for file handling, using the clipboard and changing the window display.

Another toolbar is called the Edit Bar, which has commands useful during editing.

To hide or display the Toolbar, choose the Toolbar command from the View menu.

Click	To
	Open an existing file. HexEdit displays the File Open dialog box.
	Save the file. If you have not named the file, HexEdit displays the Save As dialog box.
	Print the file. HexEdit displays the Print dialog box.
	Remove the selected bytes from the file and store them on the clipboard.
	Copy the selected bytes to the clipboard.
	Insert the contents of the clipboard at the insertion point.
	Toggle Autofit mode. When Autofit is on the columns change to fit the display.
	Increase the size of the display font.
	Decrease the size of the display font.
	Toggle between decimal and hex addresses.
	Toggle the display of the character area.
	Toggle between displaying characters in ASCII or EBCDIC.
	Change the display of ASCII control characters.

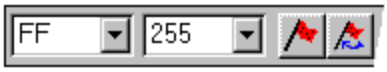
 Toggle the display of ASCII “graphic” characters (high bit on).

 Toggle between read-only and read-write mode.

 Start/stop macro recording.

 Replay recorded macro.

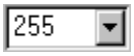
Edit Bar



The Edit Bar, by default, is displayed across the top of the application window, below the Toolbar. If your screen is wide enough you may prefer to drag it next to the Toolbar.

The Edit Bar is a toolbar which, like the Toolbar, provides quick mouse access to tools useful when editing files in HexEdit. These include facilities for searching and jumping.

To hide or display the Toolbar, choose the Edit Bar command from the View menu.

Click	To
	Enter a hex address to jump to, or select a previous address from the list.
	Enter a decimal address to jump to, or select a previous address from the list.
	Mark the current cursor position.
	Jump to the last marked position.
	Search backward using the current search string.
	Enter hex digits, or a string to search for, or select from the history list.
	Search forward using the current search string.

Status Bar

The status bar is displayed at the bottom of the HexEdit window. To display or hide the status bar, use the Status Bar command in the View menu.

The left area of the status bar describes actions of toolbar buttons or menu items as you move the mouse over them or use the arrow keys to navigate through menus. It also displays the occasional status or progress message depending on the current actions being performed.

The right area of the status bar has several *indicators* which have the following uses:

Indicator	**Description**
Byte values	The first indicator (to the right of the message area) displays the values of the byte at the current cursor position in the active window. This shows the value of the byte in hex, decimal, octal and binary. It also shows the ASCII or EBDCIC character that the byte represents, depending on EBCDIC mode. To see more information about the byte(s) after the cursor and the file in the active window double-click on this indicator to open the Properties Dialog.
Hex offset	Indicates the numbers of bytes from the mark to the cursor in hex. See the Mark Position command.
Decimal offset	Indicates the numbers of bytes from the mark to the cursor in decimal.
RO/RW	Indicates read-only or read-write mode. To toggle between read-only and read-write just double-click this indicator or use the Allow Changes command.
OVR/INS	Indicates overtype or insert mode. To toggle between overtype and insert modes double-click this indicator or use the Insert Mode Toggle command.
REC	Indicates when macro recording is active. To start or stop recording double-click this indicator or use the Record command.
CAP	Indicates the state of the Caps Lock toggle.
NUM	Indicates the state of the Num Lock toggle.

Print Preview toolbar

The print preview toolbar offers you the following options:

Print
Bring up the Print Dialog box, to start a print job.

Next Page
Preview the next printed page.

Prev Page
Preview the previous printed page.

One Page / Two Page
Preview one or two printed pages at a time.

Zoom In
Take a closer look at the printed page.

Zoom Out
Take a larger look at the printed page.

Close
Return from print preview to the editing window.

Keyboard Shortcuts

All keyboard shortcuts are currently hard-coded. In the future a facility may be provided to allow reconfiguration of the keyboard. In the meantime I chose what I thought was a sensible set of keys that would suit a lot of people likely to use the program.

The first consideration was compatibility with commands that are provided in other Windows editors and word-processors – Windows standard keys as prescribed by Microsoft as well as de facto standard keys. I also borrowed keys from MS Word, FED (a popular MSDOS binary file editor), Brief (a popular MSDOS programmers editor), and some de facto UNIX standards (like Ctrl+L to redraw the screen).

"Standard" Windows Keys

Arrows keys	Move cursor up or down a row, left or right a column
Pg Up and Pg Dn	Move cursor by one line less than a window full, up or down
Home, End	Move to start or end of line
Ctrl+Home, Ctrl+End	Move to start or end of **file**
Ctrl+Left, Ctrl+Right	Move left or right by a group of columns
Shift+cursor key	As above but also extend selection with cursor movement
Ctrl+N	Create new file
Ctrl+O	Open existing file
Ctrl+S	Save file
Ctrl+P	Print file
Ctrl+F4	Close
Ctrl+X	Cut the selection to the clipboard
Del	Delete the selection
Ctrl+C	Copy to the clipboard
Ctrl+V	Paste
Ctrl+Z	Undo
Ctrl+A	Select all
Ctrl+F6	Next window
Ctrl+Shift+F6	Previous window
Ctrl+F	Find
F3	Repeat find
F1	Help
Shift+F1	Invoke help mode
Insert	Toggle between insert and overtype mode

All Keys

Cursor Keys

Arrows keys	Move cursor up or down a row, left or right a column
Pg Up and Pg Dn	Move cursor by one line less than a window full, up or down
Home, End	Move to start or end of line
Ctrl+Home, Ctrl+End	Move to start or end of file
Ctrl+Left, Ctrl+Right	Move left or right by a group of columns
Shift+cursor key	As above but also extend selection with cursor movement
Ctrl+F9	Move cursor to mark
Shift+Ctrl+F9	Extend selection to mark
Shift+F9	Swap cursor and mark

File Commands

Ctrl+N	Create a new file
Ctrl+O	Open an existing file

Ctrl+S	Save the file
Ctrl+P, Alt-P	Print the file
Ctrl+F4	Close the file
Edit Commands	
Ctrl+X, Shift+Del	Cut to the clipboard
Del	Delete the selection
Ctrl+C, Ctrl+Insert	Copy to the clipboard
Ctrl+V, Shift+Insert	Paste
Ctrl+Z, Alt-U, Alt-BS	Undo
Ctrl+A	Select all
Insert	Toggle between insert and overtype modes
Tab	Swap between hex and character areas
View Commands	
Alt-D	Toggle between decimal and hex addresses
Alt-T	Toggle the display of graphic characters
Enter	Move the current column to the left side of display (change offset)
Ctrl+U	Scroll up one line
Ctrl+D	Scroll down one line
Ctrl+L	Scroll cursor to the middle of window and redraw the entire window
Searching	
Ctrl+F, Alt-S	Invoke the Find dialog
F3, Shift-F4	Repeat find forwards
Shift+F3	Repeat find backwards
Ctrl+F3	Find the current selection
F4	Invoke the find tool in non-case-sensitive character mode
F5	Invoke the find tool in case-sensitive character mode
F6	Invoke the find tool in hex mode
Tools	
Alt-C	Compare windows
Alt-G, Alt-J	Invoke the hex address tool
Ctrl+G	Invoke the decimal address tool
F7	Record keystroke macro
F8	Play the keystroke macro
Window Commands	
F2, Alt-N	Go to the next non-minimised window
Ctrl+F6	Next window
Ctrl+Shift+F6	Previous window
Help Commands	
F1	Help
Shift+F1	Invoke help mode

Context Menus

There are two context menus that appear when you right click on certain parts of the HexEdit display.

Tool Bars Context Menu

Right clicking on a tool bar or the status bar displays a context menu that allows you to hide or show any of the tool bars or the status bar. These options are also available on the View menu.

Toolbar	Shows or hides the Toolbar.
Edit Bar	Shows or hides the Edit Bar.
Status Bar	Shows or hides the Status Bar.

Edit Window Context Menu

Right clicking within a window invokes a context menu that allows you to select various edit and display options. These options are also available on the Edit and View menus.

Cut	Deletes data from the file and moves it to the clipboard.
Copy	Copies data from the file to the clipboard.
Paste	Pastes data from the clipboard into the file.
Allow Changes	Toggles between allowing changes and read-only mode.
Insert	Toggles between Insert and Overtype mode.
Go To Mark	Returns the cursor to the last marked location.
Autofit	Toggles the mode in which the columns are adjusted to fit the window.
Font	Invokes the standard Font Selection dialog.
Decimal Addresses	Toggles between hex and decimal addresses.
Character Display	Toggles the display of the character area.
EBCDIC Characters	Toggles between display of ASCII or EBCDIC characters.
Control Characters	Toggles the display of ASCII control characters.
Graphic Characters	Toggles the display of “graphic” characters.
Options	Invokes the Window Display page of the Options dialog.
Properties	Invokes the Properties dialog.

Title Bar

The title bar is located at the top of the main HexEdit window. It contains the name of the application and the name of the currently active file, if any. You can move the window by dragging the title bar.

The title bar also contains an HexEdit icon (at the right end) and buttons (at the left end):

 Click the icon to open the Control menu.

 Click the minimize button to reduce the window to an icon.

 Click this to restore the window to its previous size and location

 Click the maximize button to enlarge the window to fill the screen.

 Click this to close HexEdit. You are prompted to save changes.

Scroll Bars

Scrollbars appear at the right and bottom edges of the window if there is more of the file than will fit within the window. If Autofit mode is on (see the Autofit command) then no horizontal scroll bar will be shown at the bottom of the window since the width of the file display is adjusted to fit the window.

The discussion below concentrates on the vertical scroll bar since it is more often utilised. Differences for the horizontal scroll bar are noted in brackets.

The arrows at either end of the scrollbars allow you to scroll a line (or column) at a time. Alternatively, you can use the Ctrl+U and Ctrl+D keyboard shortcuts to scroll up and down by a line. Clicking the scrollbar above (or to the left of) the scroll box will scroll up (or left) by just under a window full of information. Similarly clicking below (or to the right) scrolls down (or right) by the same amount.

The scroll box within the scroll bar indicates the current vertical (or horizontal) location in the file. The size of the scroll box indicates the proportion of the file that is visible within the window. You can drag the scroll box to quickly scroll to any position in the file.

Command Line Options

HexEdit supports the /P and /PT command line options. The Windows shell uses these to support drag and drop printing. The /P option takes one parameter – a filename – and prints that file to the default printer. The /PT option allows printing to any printer. It is followed by 4 parameters: filename, printer, driver, and port.

If you fire up HexEdit from a command line interpreter you can specify one or more files on the command line for HexEdit to open. The filenames specified may include wildcards. For example, "HexEdit *.BMP *.GIF" opens all the .BMP and .GIF files in the current directory.

Hex Address tool

Use this tool to move the cursor in the active window to a specific hex address in the file. Click on the tool or press Alt+J, then type the hex address and press **Enter**. Press **Esc** instead of **Enter** to return to the file without changing the current address.

While entering an address the Decimal Address Tool is simultaneously updated with the corresponding decimal address. As a side-effect this provides a quick means of converting between hex and decimal numbers. To quickly jump to the Decimal Address tool without losing the address just entered press **Tab**.

This tool contains a history of the previously jumped to hex addresses. To jump to one of these addresses select it from the drop down list and press **Enter**. Alternatively, you can use the up and down arrows to find the history entry you require then press **Enter**.

The address shown in this tool is updated as you move the cursor within the file. During long searches and compares the address in this tool is also updated to indicate the progress of the process.

You can cut and copy to, and paste from the clipboard from/into this tool by selecting the appropriate option from the context menu. (To see the context menu right click on the tool.) This might be useful if, for example, you had a hex address (as text) on the clipboard, perhaps copied there from an editor or disassembler.

To toggle between displaying hex and decimal addresses in the edit window use the Decimal Addresses command.

See also the Decimal Address Tool.

Decimal Address tool

Use this tool to move the cursor in the active window to a specific decimal address in the file. Click on the tool, then type the decimal address and press **Enter**. Press **Esc** instead of **Enter** to return to the file without changing the current address.

While entering an address the Hex Address Tool is simultaneously updated with the corresponding hex address. As a side-effect this provides a quick means of converting between hex and decimal numbers. To quickly jump to the Hex Address tool without loosing the address just entered, press **Shift+Tab**.

This tool contains a history of the previously jumped to addresses. To jump to one of these addresses select it from the drop down list and press **Enter**. Alternatively, you can use the up and down arrows to find the history entry you require then press **Enter**.

The address shown in this tool is updated as you move the cursor within the file. During long searches and compares the address in this tool is also updated to indicate the progress of the search or comparison.

You can use the clipboard with this tool (cut, paste etc) by selecting the appropriate option from the context menu. (To see the context menu right click on the tool.) This might be useful if, for example, you had an address (as text) on the clipboard, perhaps copied there from an editor or disassembler.

To toggle between displaying hex and decimal addresses in the edit window use the Decimal Addresses command.

See also the Hex Address Tool.

Find tool

Use this tool to begin searching the file from the current cursor position. Click on the tool and enter the bytes to search for and press **Enter**. To search backward click the Find Backward button instead of pressing **Enter**. Press **Esc** instead of **Enter** to return to the file without performing a search.

There is a great deal of flexibility in this simple tool. You can enter the bytes to search for in hex or as an ASCII (or EBCDIC) string. String searches can be case-sensitive or case-insensitive. The following list explains the options. To perform a:

hex search	- type the hex digits;
case-sensitive search	- type "=" (equals sign) then the characters to search for;
case-insensitive search	- type "~" (tilde) then the characters to search for.

A quick way to start a search is to press a function key. Pressing F4 jumps to the Find tool ready to start a case-insensitive search. F5 starts a case-sensitive search. F6 starts a hex search. Note that if you delete the "=" or "~" at the start of the string the characters are automatically converted to hex. Similarly, adding "=" or "~" at the start of hex digits converts them to the corresponding (ASCII or EBCDIC) string. (You can use this along with the EBCDIC Characters command to convert between EBCDIC and ASCII.)

If you're currently in EBCDIC rather than ASCII mode then string searches (both case-sensitive and case-insensitive) are performed using the EBCDIC character set. To toggle between ASCII and EBCDIC see the EBCDIC Characters command.

The Find tool contains a history of the 16 most recent searches, which is retained between invocations of HexEdit. To recall a previous search, select it from the drop down list and press **Enter**. Alternatively, you can use the up and down arrows to find the history entry you require then press **Enter**, perhaps after modifying it.

You can use the clipboard with this tool (cut, paste etc) by selecting the appropriate option from the context menu. (To see the context menu right click on the tool.) This might be useful if, for example, you have text on the clipboard that you want to search for. However, a fast way to search for the next occurrence of some bytes in a file is to select them and press Ctrl+F3 (see the Find Selection command).

After a search has completed you can repeat it using F3 (see the Find Forward command). You can also search for the same text but going backwards using Shift+F3 (see the Find Backward command).

The Find Dialog is a more civilised way to start a search. It provides the same facilities as the Find Tool but in a more obvious fashion. It also allows Unicode searches.

The following are for context sensitive popup help for controls in dialogs.

Click a radio button to select the type of search. This determines how the text in the **Find What** box is used. A hex search uses hex digits to create a set of bytes to search for. The string searches convert the characters to the corresponding ASCII, Unicode or EBCDIC bytes.

Selects a hex search. The characters in the **Find What** box must be a sequence of hex digits which are to be searched for in the file. When you select the Hex radio button the text in the **Find What** box (if any) is automatically converted from the previous string type to the corresponding hex digits.

Selects an ASCII search. That is, the string in the **Find What** box is treated as ASCII.

Selects a Unicode search. That is, the string in the **Find What** box is treated as a Unicode string, where each character is two bytes long.

Selects an EBCDIC search. That is, the string in the **Find What** box is treated as EBCDIC.

Type the hex bytes or text you want to search for or select a previous search from the list. You can search for an ASCII, Unicode or EBCDIC string or a sequence of hex values depending on the radio button currently selected in the **Type** section. You are warned of invalid characters, such as an invalid hex digit (anything apart from 0-9, a-z, A-Z) for a hex search, or a tilde (~) etc for an EBCDIC search.

OPTIONS

Restore the main HexEdit window to its position when HexEdit was last run.

Causes the hex digits A to F to be displayed in upper case globally (all windows, dialogs etc).

Displays *nice* addresses. That is, hex addresses are displayed in groups of 4 digits and decimal addresses are displayed with thousands-separators.

When a file is saved the previous version on disk is kept (using file extension ‘‘.BAK’’).

The current settings for all global options, including those under the **General** and **Defaults** tabs, are saved when you exit from HexEdit.

The current settings for all global options are saved immediately, even if **Save Settings on Exit** is not checked.

When selected, autofit mode is on by default. Autofit mode means the number of columns is adjusted to fit the width of the window when it is resized.

When selected, addresses are displayed in decimal. Otherwise addresses are displayed in hex.

When selected, the character area is displayed to the right of the hex area. That is, characters (ASCII or EBCDIC) are displayed in columns corresponding to the hex values.

When selected, the character display area on the right of a new window defaults to EBCDIC rather than ASCII.

When selected, and the character area shows ASCII, control characters (those with a value less than 32) are displayed in red. Otherwise control characters are displayed as a red dot.

When selected, and the character area shows ASCII, graphic characters (those with a value greater than 127) are displayed in the character area. Otherwise graphic characters are displayed as a red dot.

The bytes within the window can't be modified. A read only window is not to be confused with a read only file.

The file **can** be modified as long as the file is not read-only.

Overtype mode is on when a new window is opened. That is, changes to the file overwrite the data at the cursor position (which is lost).

Insert mode is on when a new window is opened. That is, changes to the file are inserted at the current cursor position, backspace deletes characters etc.

When selected, the character area is displayed to the right of the hex area. That is, characters (ASCII or EBCDIC) are displayed corresponding to the hex values in the active window.

When selected, the number of columns is adjusted to fit the window when it is resized.

The number of columns in the hex area (and the character area if it is displayed). Note that this value cannot be changed when autofit is on since the number of columns is automatically adjusted to fit the display.

The column in which the first byte of the file is displayed. Must be between zero and one less than the number of columns.

The number of columns per group in the hex display area. Note that this also sets the number of columns by which the Ctrl+Left Arrow and Ctrl+Right Arrow keys move.

The display (windows, status bar etc) is not updated until the macro finishes.

The display is updated while the macro is playing based on the passing of time (ie. after the number of seconds has passed).

The display is updated while the macro is playing based on the number of ‘‘keystrokes’’ replayed so far.

The display is updated while the macro is playing based on the number of replays of the macro performed so far during multiple plays.

The number of seconds between each (temporal) display refresh.

The number of “keystrokes” between each display refresh.

The number of macro replays between each display refresh.

Check this box if you want display refreshes to include messages displayed on the status bar and the current search text in the Find Tool and the current address in the Hex and Decimal Address Tools on the Edit Bar.

Check this box if you want display refreshes to update the Properties Dialog (if displayed).

Select this if you want the macro replay to stop if there is any sort of problem – ie. all warning and error conditions.

Select this if you want the macro replay to stop if there are minor (or major) errors, but continue on any warnings. Minor errors include trying to move the cursor past the end of file or before the beginning of file.

Select this if you want the macro replay to continue unless there are serious errors. This option is not recommended since warnings and errors (that might result in your macro doing the wrong thing) are ignored.

Enter the name of a new group here before clicking the Add button. (This also displays the name of the currently selected existing group.)

Lists the names of all the current groups. Click the name of a group in this list before changing its colour or values.

The colour of the current group or a new group to be added. Click an entry in the drop down list to choose a colour.

The values in a group. Values are comma-separated, with a colon or dash for ranges.

Moves the currently selected group down in the list.

Move the currently selected group up in the list.

Adds a new group with the name appearing in the **Name** box. You cannot add a group with the same name as an existing group.

Removes the currently selected group from the list.

Resets the current settings to the factory default colours.

Saves the current settings as the default for new windows.

PROPERTIES

The disk file name of the file in the active window. This field may be blank if the file has not yet been saved.

The type of the file according to the system registry. **None** if the file has no extension. **Unknown** if there is nothing in the registry for the extension.

The directory where the file is stored.

The date and time the file was last changed.

The current size of the file.

Checked if the file is a read only file.

Checked if the file is a hidden file.

Checked if the file is a system file.

The hexadecimal value (00 to FF) of the byte at the current cursor position in the active window.

The decimal value (0 to 255) of the byte at the current cursor position in the active window.

The octal value (000 to 377) of the byte at the current cursor position in the active window.

The binary value (00000000 to 11111111) of the byte at the current cursor position in the active window.

The *ASCII* character at the current cursor position. If the high bit is set (the value is greater than 127) then this is the ANSI character. For values less than 32 this is the name of the ASCII control character.

The Unicode character of the 2 bytes following the cursor. Note that this is only displayed under NT.

The EBCDIC character at the current cursor position. The EBCDIC control character name is displayed for values less than 64, and **none** is displayed for invalid EBCDIC characters.

The decimal value of the byte at the current cursor position in the active window.

The decimal value of the word at the current cursor position (the 2 bytes following the cursor).

The decimal value of the double-word at the current cursor position (the 4 bytes following the cursor).

The decimal value of the quad-word at the current cursor position (the 8 bytes following the cursor).

Select this option to view the decimal values as unsigned numbers.

Select this option to view the decimal values as signed (2's complement format) numbers.

Select this option to view the decimal values as signed (1's complement format) numbers.

Select this option to view the decimal values as signed (Sign and Magnitude format) numbers.

Select this option to view with big-endian byte order. Typically this option is off since Windows/Intel systems use little-endian byte order.

The value of the IEEE floating-point number.

The mantissa of the IEEE floating-point number.

The exponent of the IEEE floating-point number. Note that this is a power of 2 not 10.

Select this option to view the 32 bit (4 byte) IEEE floating-point number. This is usually a “float” in the C/C++ language.

Select this option to view the 64 bit (8 byte) IEEE floating-point number. This is usually a “double” in the C/C++ language.

The value of the IBM floating-point number.

The mantissa of the IBM floating-point number.

The exponent of the IBM floating-point number. Note that this is a power of 16 not 2 or 10.

Select this option to view the 32 bit (4 byte) IBM floating-point number.

Select this option to view the 64 bit (8 byte) IBM floating-point number.

EMAIL DLG

Select where you want to send the email. The default is by far the best choice unless mail to it bounces.

Enter a brief description of the bug or subject of the email.

Select this option to report a bug or defect of the software.

Select this option to request an enhancement.

Select this option for other matters besides bug reports and enhancements.

Enter the text of your message. For a bug report try to describe the problem as fully as possible. In particular, if the bug is reproducible list all the steps required to reproduce it, and include the exact wording of any error messages.

Normally you use the default value, ‘‘This system’’, for this field which lets HexEdit interrogate the system on which it’s running. If you are reporting a defect found on another system please enter the name and version of the operating system.

The version of HexEdit in use. You would not normally change this from the default value, which is the version you are running.

Please enter your name in case I have to get in contact with you for more information.

Please enter one or more email addresses in case I need to contact with you for more information.

Click the ‘‘Send’’ button to send the email. If you have no currently active mail session you may be prompted to select a **Profile** (choose the default profile unless you have a good reason not to). If the profile you chose requires a password you will be prompted for that too.

Enter the number of times you want the macro to play and press Enter.

Click the ‘‘Options’’ button to change the options for macro playback before running the macro.

Click OK to start playing the macro.

No Help Available

No help is available for this object.

No Help Available

No help is available for this area of the window.

No Help Available

No help is available for this message box.

<< If you wish to author help specific to each message box prompt, then remove the AFX_HIDP_xxx values from the [ALIAS] section of your .HPJ file, and author a topic for each AFX_HIDP_xxx value. For example, AFX_HIDP_INVALID_FILENAME is the help topic for the Invalid Filename message box. >>

Ruler command (View menu)

This command is not yet implemented.

Replace command (Edit menu)

This command is not yet implemented.

Replace dialog box

This command is not yet implemented.

Repeat command (Edit menu)

Use this command to repeat the last editing command carried out. The Repeat menu item changes to Can't Repeat if you cannot repeat your last action.

This command is not yet implemented.

Next Pane

This command is not yet implemented.

Prev Pane

This command is not yet implemented.

Page Setup command (File menu)

This command is not yet implemented.