

# UUCODE Version 3.1

Allows Transfer of Binary or Data Files via E-mail  
or Internet Connected Network or On-Line Service









































For Microsoft Windows™ 3.1 and Windows for Workgroups 3.1/3.11











































*Sabasoft*


























UUCODE Copyright © 1993-1994 by Sabasoft, Inc.  
All Rights Reserved.

- [Table of Contents](#)
- [Introduction](#)
- [Shareware Concept](#)
- [Documentation Conventions](#)
- [Version Information](#)
- [Installation](#)
- [Program Operation](#)
- [Procedures](#)
- [Commands](#)
- [Keyboard](#)
- [System Requirements and Usage](#)
- [General Operating Information](#)
- [Encoding Header Formats](#)
- [If You Encounter Problems](#)
- [Technical Support](#)
- [Contacting Sabasoft](#)
- [LICENSE](#)
- [ORDER FORM](#)
- [PAYMENT TERMS](#)
- [DISTRIBUTION SURVEY](#)



## Table of Contents

-  [Introduction](#)
-  [When to Use UUCODE](#)
-  [UU and XX Encoding & Decoding](#)
-  [Shareware Concept](#)
-  [Documentation Conventions](#)
-  [Version Information](#)
-  [Version 3.1](#)
-  [Version 3.0](#)
-  [Version 2.1](#)
-  [Version 2.0](#)
-  [Version 1.7](#)
-  [Version 1.6](#)
-  [Version 1.5](#)
-  [Installation](#)
-  [Files in This Package](#)
-  [Installing and Running on Various Drive Types](#)
-  [New Installation](#)
-  [Upgrading from Any Previous Version](#)
-  [Optional File Type Association](#)
-  [Program Operation](#)
-  [Command Line Arguments](#)
-  [-J Option](#)
-  [-N Option](#)
-  [Anatomy of the Main Window](#)
-  [Interactive Mode](#)
-  [Menu Operation](#)
-  [Drag-and-Drop Operation](#)
-  [Command Line Operation](#)
-  [Quiet Mode](#)
-  [Operation Based on Input File Name](#)
-  [Procedures](#)
-  [Encoding a File](#)
-  [Decoding a File](#)
-  [What if Decoding Fails](#)
-  [Sending Binary Files via E-mail](#)
-  [Receiving Binary Files via E-mail](#)
-  [Commands](#)
-  [File Menu](#)
-  [Encode Menu Item](#)
-  [Decode Menu Item](#)

-  [Exit Menu Item](#)
-  [Configure Menu](#)
-  [General Configuration](#)
-  [Become ICON When En/Decoding](#)
-  [Stay ICON When En/Decoding](#)
-  [Exit When Done En/Decoding](#)
-  [Overwrite Existing Files without Asking](#)
-  [Use Default File Names](#)
-  [No Status Messages](#)
-  [Encoding Configuration](#)
-  [Multi Part Encoding](#)
-  [Encode Method](#)
-  [End of Line String](#)
-  [Max Encoded File Size](#)
-  [Generate Checksums on Encoded Output](#)
-  [Decoding Configuration](#)
-  [Automatic Detection of Decoding Type](#)
-  [Decoding Type](#)
-  [Decode Method](#)
-  [Decode Multiple Input Files as 1](#)
-  [Test Checksum Data if Present](#)
-  [Help Menu](#)
-  [Index Menu Item](#)
-  [Commands Menu Item](#)
-  [Procedures Menu Item](#)
-  [Keyboard Menu Item](#)
-  [Using help Menu Item](#)
-  [About Menu Item](#)
-  [Register Menu Item](#)
-  [On-Line ORDER FORM Menu Item](#)
-  [Hand-written ORDER FORM Menu Item](#)
-  [Keyboard](#)
-  [System Requirements and Usage](#)
-  [CPU Instruction Set](#)
-  [Windows Operating Systems and Modes](#)
-  [Disk Usage](#)
-  [Memory Usage](#)
-  [General Operating Information](#)
-  [DOS and UNIX End Of Line \(EOL\) Characters](#)
-  [Cooperative Multitasking](#)
-  [Temporary File Usage](#)
-  [Multiple Instances and File Sharing](#)

-  [File Permissions](#)
-  [Encoding Character for Binary Zero](#)
-  [Multipart File Limitations](#)
-  [Encoding Header Formats](#)
-  [Single File Format](#)
-  [Internet's SIMTEL20 Archive Site Format](#)
-  [Usenet Newsgroup comp.binaries Format](#)
-  [Usenet Newsgroup alt.binaries Format 1](#)
-  [Usenet Newsgroup alt.binaries Format 2](#)
-  [Usenet Newsgroup alt.binaries Format 3](#)
-  [Usenet Newsgroup alt.binaries Format 4](#)
-  [UNIX Shell Archive Format](#)
-  [DOS uuencode/uudecode Program Format](#)
-  [X-File Format](#)
-  [UUXFER Program Format](#)
-  [WINCODE Program Format](#)
-  [If You Encounter Problems](#)
-  [What if the Multiple Part UU Format I Use Isn't Supported](#)
-  [What if the Output File Seems Corrupt](#)
-  [Technical Support](#)
-  [Contacting Sabasoft](#)
-  [LICENSE](#)
-  [ORDER FORM](#)
-  [PAYMENT TERMS](#)
-  [DISTRIBUTION SURVEY](#)

## Introduction

-  When to Use UUCODE
-  UU and XX Encoding & Decoding

## **When to Use UUCODE**

Whenever you need to send binary data over communication links or networks, such as the Internet, which cannot handle 8-bit data, then UUCODE provides you a solution. Binary data is used in file types with extensions of .EXE, .COM, .ZIP, .ARC, .LZH, .ZOO, and many others.

UUCODE will also allow sending of binary data files between popular on-line services such as AT&T Mail, America On-Line, CompuServe, Delphi, GENie, MCI mail, and educational and other sites on the Internet.

Another popular use for UUCODE is on Usenet, the "Bulletin Board System (BBS) of the Internet." Again, the reason is to allow binary data to be posted as part of news articles.

UUCODE has won a CompuServe Shareware Picks of the Week award given by the WUGNET FORUM (GO WINUSER).

## **UU and XX Encoding & Decoding**

UUCODE uses a method called UU or XX encoding and decoding to convert the 8-bit binary data into a form usable by communications links or networks which cannot transport such data. This method generates (encodes) four ASCII characters from every three binary characters, which is basically converting 6 bits of binary information into the appropriate 6 bits within a 7-bit ASCII character. This guarantees that only ASCII text is sent when the encoded output is sent over computer communication links or networks. Similarly, the decoding process converts four ASCII characters into three binary characters. The penalty of this approach is that the encoded file is 33% larger than the original file, but that's better than not being able to transfer binary data at all!

XX encoding is similar, but uses a different character set than UU encoding so that character set translations will work better across multiple types of systems, i.e. between EBCDIC (IBM mainframe) and ASCII on UNIX.

Now you can handle these encoded files on a Windows 3.1/Workgroups equipped PC such that UU or XX encoding and decoding is possible between two PCs or between your PC and any other computer system, such as a UNIX system or Macintosh. UUCODE can do both encoding and decoding, and is fully compatible with the standard "uuencode" and "uudecode" programs found on UNIX or other computer systems.

Now a very brief bit about history. Those familiar with UNIX systems are probably aware of the need for encoding binary files to send via electronic mailing (e-mail) programs. If not, the "UU" stands for UNIX-to-UNIX, where the program originated.

## Shareware Concept

UUCODE is a shareware program. The shareware concept allows you to try out the software before you buy it. *It is not free software and is copyrighted by the author.* Unlike demonstration versions of software, UUCODE is fully functional and not limited or crippled in any way.

In addition, you may freely distribute the unmodified version of this software to allow others to try it before they buy. If you continue to use this software, then you are obligated to purchase it via registration. An order form to purchase UUCODE is contained in the **ORDER FORM** section. The shareware version pops up a reminder notice for registration, however, as previously indicated, no features are limited or missing. Registered users receive a password which disables these reminders.

For full details of your rights for using and distributing UUCODE, please refer to the **LICENSE** section.



## Documentation Conventions

Programs are indicated using small uppercase letters in the following font (9 point Times New Roman):  
UUCODE OR FILE MANAGER.

File names are indicated using uppercase letters in the following font (12 point Courier New) :

this is a FILENAME.

Menu commands are indicated by **Menu|Item**, where **Menu** is the top level menu for the program, and **Item** is the particular choice within the menu.


When you see bold italic phrases within paragraph text, such as ***A Bold Phrase***, this indicates a pointer to the so named heading elsewhere in this document where more detailed information can be found.

The bitmaps in this document were created in standard VGA mode: 640 x 480 x 16 colors.

The Windows 3.1 TrueType ® fonts were used.

## Version Information

This package contains Version 3.1 of UUCODE, which is a copyrighted shareware program.

-  Version 3.1
-  Version 3.0
-  Version 2.1
-  Version 2.0
-  Version 1.7
-  Version 1.6
-  Version 1.5

## Version 3.1

UUCODE version 3.1 was released in February 1994.

### New Features:

- Another command line option, `-N` or `/N`, is provided to run UUCODE in **Quiet Mode** which minimizes progress display and never prompts the user for an action; default actions are always taken. This is useful to launch UUCODE from another program which is used to automate a task without requiring user input.
- UUCODE now has two modes instead of three; **Quiet Mode** and **Interactive Mode**.
- Selection of multiple input files is now possible from the input file open dialog box. Pressing the Control key while clicking on a file will add it to the list to be processed. Pressing the Shift key while clicking on a file will add a range of files to the list to be processed. For more information on how to use the standard Shift or Control Click feature of Windows, please read your Windows manuals.
- Upgraded support of DOS uuencode/uudecode programs to version 5.24.
- When invalid UU or XX encoded data is encountered, the user is prompted to continue. UUCODE will, if instructed, attempt to decode the data. The reason for this feature is that on some occasions, the encoded files ending with a certain character pattern (spaces) sometimes get truncated by email systems. UUCODE will assume that spaces are required to fill out the encoded data and attempt decoding anyway.
- UUCODE now places its initialization file `UUCODE.INI` in the home directory of UUCODE rather than in the Windows directory. This makes un-installation easier.
- Additional logic has been added to improve detection of multiple part types.
- Both UNIX and DOS path names (versus just the filename part) are accepted in the encoded data file. Path separators for DOS `\` (backslash) and UNIX `/` (slash) are recognized. The path component is stripped off when creating the original input file contained in the encoded file.
- An electronic on-line order form is provided which allows the user to type the registration information into a dialog box and print the order instead of requiring the user to first print an order form and then hand writing the order. In addition, the on-line order form automatically calculates the order price. This limits math mistakes and makes it much easier to read the orders. The printable order form which is completed via hand-writing is still provided, however.
- The installation program will ask to create a Program Manager group instead of automatically creating one. The group name may also be changed from the default.



## Version 3.0

UUCODE version 3.0 was released in September 1993 with *substantial* new capabilities. Of course, all features of past versions are also provided.

### New Features:

- Support of both UU and XX encoding and decoding methods.
- Support of 12 common e-mail/Usenet header formats, for both encoding and decoding! Instead of requiring others compatible with your encoding header format, you can be compatible with them and avoid proliferating yet another header format. This will make automated decoding available to all. The formats supported are (see Encoding Header Formats for details):
  - \* Single file.
  - \* SIMTEL-20 internet archive site.
  - \* Usenet newsgroups comp.binaries.ibm.pc & comp.binaries.ms-windows
  - \* The most common 4 formats used in the Usenet newsgroups alt.binaries (unfortunately, this group has the most prolific number of unique formats and it is difficult to keep up with them all) :-)
  - \* UNIX shell archive.
  - \* The format used by the DOS programs UUENCODE and UUDECODE v5.22.
  - \* The format used by the program UUXFER v2.0.
  - \* Format using "X-File" markers as multipart headers.
  - \* The format used by the program WINCODE v1.5.
- Capability of decoding with any combination of files and multiple parts within those files. Don't forget the *exclusive* ability UUCODE has to accept multiple file parts in *any order* so that you don't have to put multiple parts in order, which saves a tremendous amount of time. That's what tools are for, to make your life easier! The combinations of files and parts supported are:
  - \* Single part in a single file.
  - \* Multiple parts in a single file, with the parts in *any order*.
  - \* Combinations of the above; any mixture of single or multiple parts, each in their own file, with the parts in *any order*.
- Normally, automatic detection of decoding format is used. However, now a particular decoding format to look for can be specified.
- Configurable retry of decoding in case the header format can't be automatically detected.
- Configurable error detection capability to ensure the integrity of the UU/XX encoded data. For encoding, file checksum data can be optionally output for testing by the receiver's decoding program. For decoding, automatic detection of file checksum data is provided and the checksum data is optionally tested with output of error warnings if so configured.
- Capability to remember your favorite configuration settings. A separate UUCODE . INI file is used to avoid cluttering up your WIN . INI file.
- Support for both DOS (CR/LF) and UNIX (LF) end-of-line terminators on encoded output. (UUCODE has always supported decoding with either DOS or UNIX end-of-line terminators)
- Automated installation utility to save time and make installation easier.



## Version 2.1

UUCODE version 2.1 was released in May 1993.

### **New Features:**

- Multiple file names are allowed on the command line instead of only one file name. The filenames must be separated by "white space", e.g. one or more space or tab characters. Note that wild card matching (i.e. \*.UUE or MYFILE?.ZIP) is not provided.
- Extra checking is done to ensure the file name to be decoded as indicated in the UU encoded file meets DOS file name specifications; up to 8 characters, a separator '.', and up to three characters after the separator as the extension. Duplicate '.' characters will also be removed. UUCODE will automatically modify the name, if necessary, to allow successful decoding.

## Version 2.0

UUCODE version 2.0 was released in January 1993. UUCODE has been significantly enhanced from versions 1.5 through 1.7. Version 2.0 contains all the features of version 1.X plus the following new features:

### General Features:

- Use of common dialog box functions present in Windows 3.1 (Windows 3.0 users can still take advantage of this feature with the `COMMdlg.DLL` file supplied).
- Full drag and drop support with multiple file capability (Windows 3.1 only).
- New user interface providing more interactive information.
- Configuration capabilities allowing running as an ICON and suppressing status messages (good for background operation).
- On-Line help facility.
- On-Line order form and shareware license agreement.

### Decoding Features:

- *Automatic detection* of three common input formats; single file format (as in version 1), the format used by Internet's *comp.binaries.ibm.pc* network news group, or the format used by the popular SIMTEL20's Internet file archive site.
- Automatic multiple part joining, with *no restrictions* on the ordering of the parts. The ordering of the parts need not be in 1 to N order, but may be in any order. This provides a large time and effort savings.
- Automatic filtering of non UU encoded data, such as E-mail headers. There is no need to manually strip these from the input, thus saving time and effort.

### Encoding Features:

- Support of optional multiple part output using either Internet's *comp.binaries.ibm.pc* network news group format or the Internet's SIMTEL20 archive site format.
- Configurable encoded file maximum output size for multiple part output to allow for various maximum file sizes supported by your E-mail system(s). The number of output parts are automatically determined by the input file size and the maximum output file size.



## Version 1.7

UUCODE version 1.7 was released in May 1992. A new feature was added on Version 1.6 to allow a *single* file argument on the command line so that UUCODE can be launched via the file manager (or launched via any other Windows replacement shell). A filename with an extension ".UUE" will automatically run the decoding portion of UUCODE, while any other extension will run the UU encoding part of UUCODE.

UUCODE is also compatible with Windows 3.1. Note the drag and drop capability is not supported.

## **Version 1.6**






UUCODE version 1.6 was released in October 1991. Two bugs were fixed from Version 1.5. First, UUCODE will allow multiple instances of itself. Of course, when two or more instances are running with file I/O, system performance will be slower. Second, Version 1.5 might possibly insert an additional byte or two when decoding. This caused no problems with archived (.ZIP, .ARC, .LZH, .ZOO, etc.) or executable (.EXE or .COM) files, but could interfere with straight text files.

## **Version 1.5**

Version 1.5 of UUCODE for Windows was released in August 1991 as the original shareware release of the program.

## Installation

To save time and make installation easier, UUCODE now has an automated setup utility called `SETUP.EXE`, which copies the files to their appropriate location on your destination disk.

-  **Files in This Package**
-  **Installing and Running on Various Drive Types**
-  **New Installation**
-  **Upgrading from Any Previous Version**
-  **Optional File Type Association**

## Files in This Package

This UUCODE package contains the following files:

READ.ME	Important information and updates about UUCODE.
WHATS.NEW	New features and bug fixes in this version of UUCODE.
SETUP.EXE	Automated installation utility.
SETUP.INF	Installation file required by SETUP.EXE.
LZEXPAND.DLL	Required by the setup program.
UUCODE.EXE	The UUCODE executable program.
UUCODE.HLP	The on-line help file for UUCODE. This includes the shareware license information as well as an order form.
WHIO3.DLL	A support dynamic link library for file I/O.
BARCTRL.DLL	A support dynamic link library for the progress indicator.
FILE_ID.DIZ	Assists in BBS distribution of this program.

Registered users who have ordered a diskette will also find the following files on their distribution diskette.

UUCODE.RT_	Compressed version of this users manual in Rich Text Format (RTF). This file can optionally be uncompressed and installed on your system with the automated setup utility provided.
UUCODE.INI	Contains registration username and password so entry of registration information data is not required. This file is automatically installed by the setup program.
REGDATA.TXT	Registration username and password information in printable form.

## **Installing and Running on Various Drive Types**

UUCODE may be installed on and run from either a floppy disk, hard disk, or networked drive as long as the WHIO3.DLL and BARCTRL.DLL files are in the same directory as UUCODE.EXE or somewhere in the PATH (see your DOS manual for setting the PATH environment variable in the AUTOEXEC.BAT file).

## **New Installation**

Run `SETUP.EXE` from the drive/directory containing the UUCODE files.

## **Upgrading from Any Previous Version**

Run the `SETUP.EXE` from the drive/directory containing the `UUCODE` files as for new installation. This will automatically overwrite any previous versions. You can also recover some disk space by following the steps below:

Step 1. Delete the version 2.X file called `WHIO2.DLL`.



## Optional File Type Association

If you want to run UUCODE from FILE MANAGER and take advantage of the **Command Line Operation**, you can cause UUCODE to automatically be invoked when you want to decode a file. Files to be decoded are identified by either a ".UUE" or ".XXE" extension.

Encoding a file is also possible if you want to associate other file types having extensions different than ".UUE" or ".XXE". Usually this is not the case because you may want to run another program on such files, such as a word processor or archive program. The better way to do this is to use **Drag-and-Drop Operation**. If you do want to associate files to encode with UUCODE from FILE MANAGER, then follow steps 3 and 4 below using the appropriate file extension.

If you are running Windows 3.1, you need do nothing extra to take advantage of **Drag-and-Drop Operation**.

To associate files for decoding with UUCODE, follow the steps shown below.

- Step 1. Run the FILE MANAGER.
- Step 2. FILE MANAGER requires a file with the appropriate extension to exist so that associations can be made. Copy your AUTOEXEC .BAT or any other file to a file called Z .UUE.
- Step 3. Highlight the Z .UUE file. (do not double click or press Enter to run it).
- Step 4. From the top menu choose, **File|Associate**. When the Associate dialog box pops up, type the full path name of where UUCODE .EXE is located. Then press Enter or click OK. This completes the association. You may now delete the Z .UUE file.
- Step 5. Repeat steps 2 through 4 with a file called Z .XXE.






FILE MANAGER will now automatically run the decoding or encoding portion of UUCODE when the files with that extension are double-clicked from FILE MANAGER.

## Program Operation

UUCODE has two modes of operation; interactive mode and quiet mode. **Interactive Mode** is used when the command line argument of *-N is not specified* on the command line when starting UUCODE. This mode, which is the default, is useful whenever the most user interaction and progress display is desired.

**Quiet Mode** is used whenever the command line option of *-N is specified* on the command line. This mode is useful for launching UUCODE from a macro or other program in which user input and query is not desired.

Since each mode operates differently with respect to user query and display of progress, each mode is described separately.

-  **Command Line Arguments**
-  **Anatomy of the Main Window**
-  **Interactive Mode**
-  **Quiet Mode**
-  **Operation Based on Input File Name**

## Command Line Arguments

UUCODE supports two command line arguments. The options can be declared using UNIX or DOS option style indicators, / (slash) or - (dash), respectively, and may be in either uppercase or lowercase letters.



**-J Option**



**-N Option**

## **-J Option**

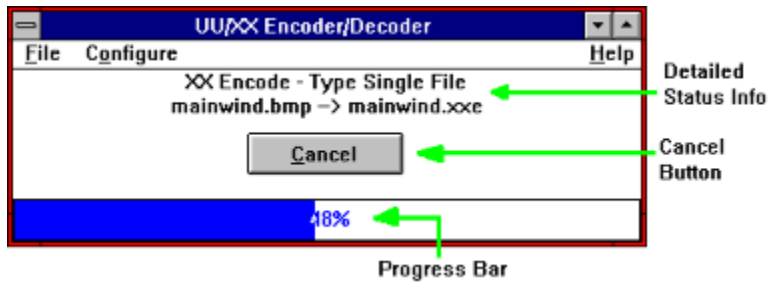
**/J or -J** Permits the user to **Decode Multiple Input Files as 1**. When this option is used, all input files specified on the command line are treated as one large input file.

## **-N Option**

`/N` or `-N` Permits the user to run in **Quiet Mode**. When this option is used, user input is never required and all actions are based on the configured options.

## Anatomy of the Main Window

The main window provides detailed interactive status information while UUCODE is running as illustrated below. This status is provided for all operating modes.



The status information lines indicate the method of encoding or decoding selected, the input and output file names, the encoding or decoding format being processed, and the number of parts.

The cancel button allows the user to cancel the operation while it's running. This button only appears when an operation is in progress.

The progress bar indicates how much of the input file has been processed.

## Interactive Mode

The interactive mode of UUCODE provides detailed status information during and after the encoding or decoding process. Interactive mode is used whenever Quiet Mode is *not selected*.

Interactive mode is most useful when several files will be encoded or decoded. In addition, while in interactive mode, unlike Quiet Mode, UUCODE will prompt the user before the following actions are done:

- A. Prompt for a file to be encoded or decoded. Prompts are not required if file(s) are dropped on the main window (**Drag-and-Drop Operation**) or file(s) are specified on the command line (**Command Line Operation**).
- B. After choosing the encoding or decoding operation, a suggested file name for output is shown. The user is allowed to modify the output file name, unless **Use Default File Names** is enabled.
- C. If a file to be output already exists, UUCODE will ask permission before overwriting it, unless **Overwrite Existing Files without Asking** is enabled.

Interactive mode can be entered in three ways as described next.



**Menu Operation**



**Drag-and-Drop Operation**



**Command Line Operation**

## **Menu Operation**

Menu operation is started by running UUCODE without any files specified on the command line. All subsequent activities are carried out via menu choices and dialog box entries. For instructions on how to use UUCODE via the menus, see either **Encoding a File** or **Decoding a File**.



## **Drag-and-Drop Operation**

Drag-and-Drop operation is activated when files are dropped on UUCODE via FILE MANAGER. Drag-and-Drop operation can also be started by dropping files on the UUCODE . EXE file in a FILE MANAGER window when UUCODE is not yet running. In that case, FILE MANAGER will automatically start running UUCODE and drop the selected files on it.

UUCODE supports dropping of one or many files. If multiple files are dropped, they will be operated on one at a time until all files have been processed. Each file is treated independently, unless **Decode Multiple Input Files as 1** is enabled. Therefore, you may encode or decode files as determined by their file name. To determine how the dropped file name affects the operation selected by UUCODE, see the **Operation Based on Input File Name** section.

## Command Line Operation

Command line operation is used via the **File|Run** command of either PROGRAM MANAGER or FILE MANAGER or when you double-click a file in FILE MANAGER and you have previously set up your **Optional File Type Association**. Command line operation is most useful when one file or set of files will be encoded or decoded.

UUCODE will allow multiple files on the command line. Each file name must be separated by one or more spaces from the other names on the command line.

An example of running UUCODE in command line operation using FILE MANAGER is shown below.



## Quiet Mode

The quiet mode of UUCODE is intended for running by macros or other programs which do not require user input. All operations are done via the programmed defaults (see **Configure Menu**). Quiet Mode is entered whenever the **-N Option** is specified on the command line when starting UUCODE .

## Operation Based on Input File Name

The choice of performing an encode or decode process can be determined by the extension of the file name supplied on the command line, if the **Decode Method** is optioned to do so. Files with an extension of ".UUE" will be run through the UU decoding process. Files with an extension of ".XXE" will be run through the XX decoding process. You can also force UU or XX decoding, regardless of the input file name, as described in **Decode Method**.

Files with any extension *other than* ".UUE" or ".XXE" will be run through the encoding process. The choice of encoding method, either UU or XX, is determined by the previously configured **Encode Method**.

## Procedures

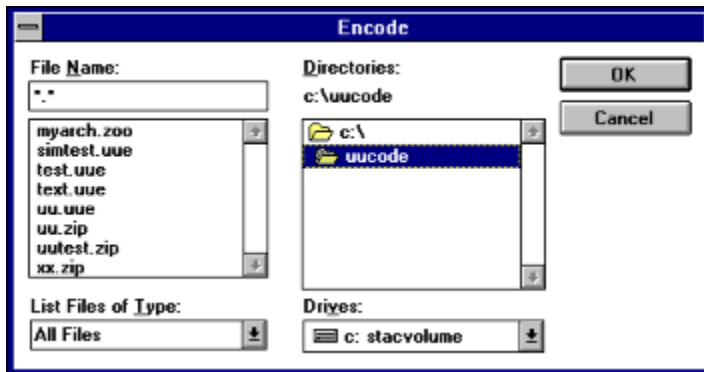
-  Encoding a File
-  Decoding a File
-  Sending Binary Files via E-mail
-  Receiving Binary Files via E-mail

## Encoding a File

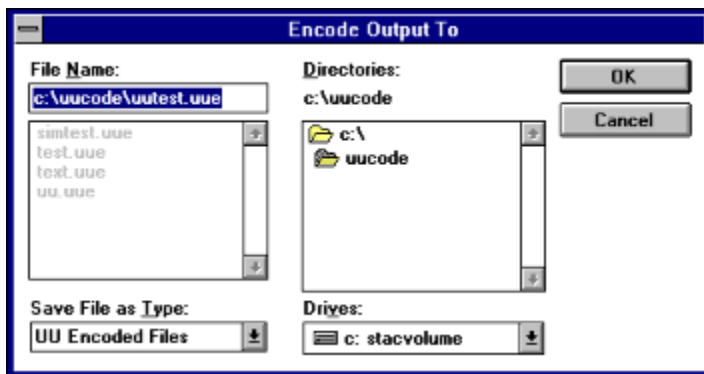
Encoding a file allows you to convert a file containing 8-bit binary data into a 7-bit ASCII encoded readable form. The following steps apply to **Interactive Mode**. In **Quiet Mode**, files with extensions other than ".UUE" or ".XXE" are automatically encoded without additional user prompting.

Entry. Selecting the **File|Encode** menu item enters the encoding process.

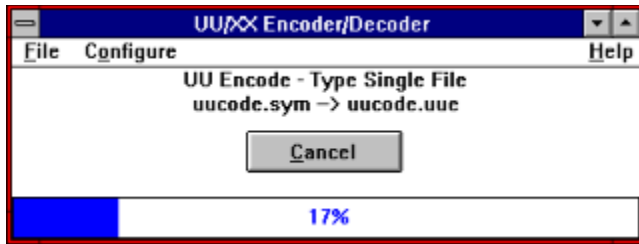
Step 1. The following dialog box will appear showing all files in the directory.



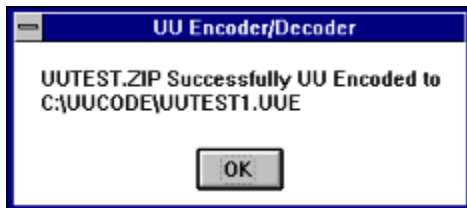
Step 2. You will first be prompted to select a file to encode. Pick the file name and then click the OK button or press Enter. An output file name will be suggested with the default file extension of ".UUE" for UU encoding or ".XXE" for XX encoding. You may change it if you wish, unless **Use Default File Names** is enabled. This is illustrated below.



Step 3. The encoding process is started by clicking the "OK" button (or pressing Enter). If the output file already exists, UUCODE will ask if you want to overwrite it, unless **Overwrite Existing Files without Asking** is enabled. You will notice a % complete bar being updated as UUCODE encodes the input file (as shown below).



- Step 4. At any time during the encoding process, you may click the "Cancel" button with the mouse or ALT-C and be returned to the main menu, ready for another command. If you cancel the encoding operation, no encoded output file will be generated. When encoding is complete, you will see a dialog box indicating success (or failure) as shown below.

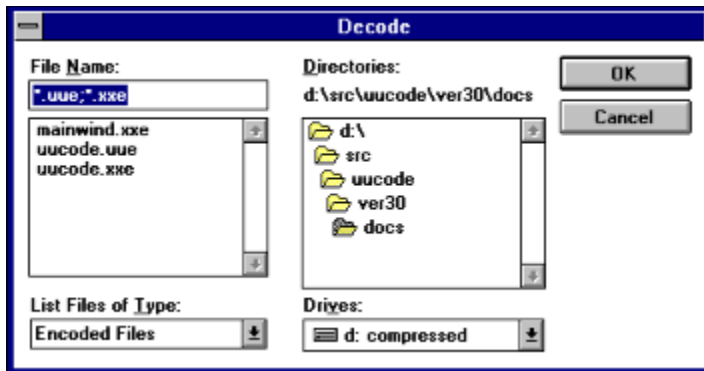


## Decoding a File

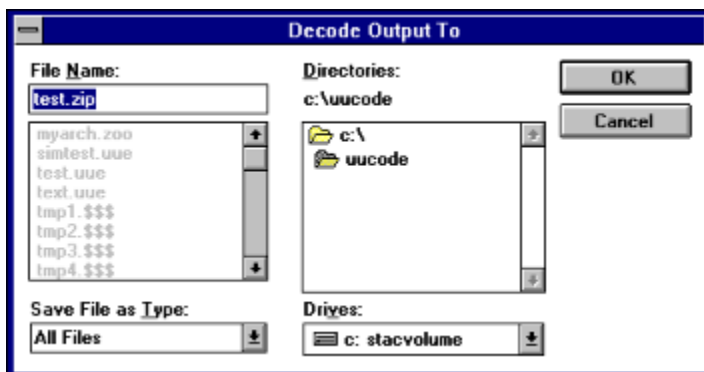
Decoding a file allows you to convert a file containing a 7-bit ASCII encoded data into its original 8-bit binary data form. The following steps apply to **Interactive Mode**. In **Quiet Mode**, files with the extension of ".UUE" or ".XXE" are automatically decoded without additional user prompting.

Entry. Selecting the **File|Decode** menu item enters the decoding process.

Step 1. The following dialog box will appear showing all ".UUE" and ".XXE" files in the directory.

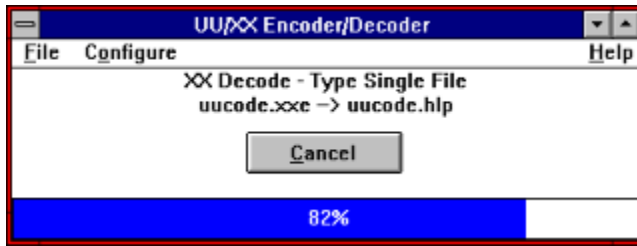


Step 2. You will first be prompted to select a file to decode, using the default extensions of ".UUE" and ".XXE". Pick the file name and then click the OK button or hit Enter. UUCODE will read the encoded file to determine the name of the output file. You will be prompted with the output file name, which you can change if you wish, unless **Use Default File Names** is enabled. This is illustrated below.

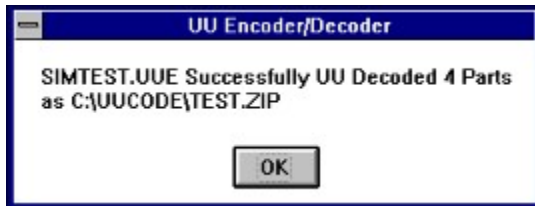


Step 3. The decoding process is started by clicking the "OK" button. UUCODE will check if the output file already exists. If it does, you will be asked if you want to overwrite it, unless **Overwrite Existing Files without Asking** is enabled. You will then notice a % complete bar being updated as UUCODE decodes the input file (as shown below).





- Step 4. At any time during this process, you may click the "Cancel" button using the mouse or ALT-C and be returned to the main menu, ready for another command. If you cancel the decoding operation, no decoded output file will be generated. When decoding is complete, you will see a dialog box indicating success (or failure) as shown below.



#### What if Decoding Fails

### **What if Decoding Fails**

When UUCODE cannot complete decoding successfully due to its inability to properly determine a multipart format, it will display a dialog box explaining the problem. The user may then retry decoding with either a specific format or try to decode using all formats supported. If decoding still fails after trying all supported formats, see **What if the Multiple Part UU Format I Use Isn't Supported** for suggestions in dealing with this situation.

## **Sending Binary Files via E-mail**

In order to send binary files via electronic mail "E-mail," one must have both an electronic mail program as well as UUCODE. Normally, sending e-mail is a one step process; just use your e-mail program. To send binary files, an additional step must be taken using UUCODE. Once the binary data is sent, the process must be reversed as explained in **Receiving Binary Files via E-mail**.

Step 1. UU or XX encode the binary file as explained in **Encoding a File**. For illustrative purposes, let's call the binary file `BINARY.EXE`. The default encoded output file would be called `BINARY.UUE`. If multiple parts are required due to file size limitations, you could have several encoded output files. In this example, let's assume three parts are required; called `BINARY1.UUE`, `BINARY2.UUE`, and `BINARY3.UUE`.

Step 2. Start or activate your e-mail program. Now attach or insert the encoded file, called `BINARY.UUE`, as the e-mail data. The recipient of this e-mail message must run the message through the decoding process, which is described in **Decoding a File**. This completes the sending process. The recipient will reverse this process as explained in **Receiving Binary Files via E-mail**.

If you must send multiple encoded parts due to e-mail file size limitations, repeat this step for each part of the encoded file set. Send one encoded file per e-mail message. In this example, each of three e-mail messages would contain one file; `BINARY1.UUE`, `BINARY2.UUE`, and `BINARY3.UUE`.

## Receiving Binary Files via E-mail

In order to receive binary files via electronic mail "E-mail," one must have both an electronic mail program as well as UUCODE. Normally, receiving e-mail is a one step process; just use your e-mail program. To receive binary files, an additional step must be taken using UUCODE. These two steps explain how to reverse the process of encoding as explained in [Sending Binary Files via E-mail](#).

Note: If you must receive multiple encoded parts due to e-mail file size limitations, first enable [Decode Multiple Input Files as 1](#) so that all input (encoded) file parts can be decoded in one step.

Step 1. Start or activate your e-mail program. Save the e-mail message to a file. For illustrative purposes, call the saved e-mail message file `BINARY.UUE`. You can save the message with all header information. There is no need to strip off the e-mail header data because UUCODE will not try to decode it.

If there are multiple parts to decode, repeat this step as many times as required, giving each part a different file name. If there were three parts, save each part with a number, such as `BINARY1.UUE`, `BINARY2.UUE`, and `BINARY3.UUE`.

Step 2. On the saved file called `BINARY.UUE`, run the decoding process as explained in [Decoding a File](#). If there are multiple parts, you will have to select *all* the parts for decoding to work. In this multiple part example, you would select `BINARY1.UUE`, `BINARY2.UUE`, and `BINARY3.UUE`.

When complete, the original `BINARY.EXE` file will have been re-constructed in the same directory as the `BINARY.UUE` file resides.

## Commands

The following topics explain the menu system of UUCODE.



**File Menu**



**Configure Menu**



**Help Menu**

## File Menu

The file menu allow you to encode or decode files, or exit the program.



**Encode Menu Item**



**Decode Menu Item**



**Exit Menu Item**

## **Encode Menu Item**

Selecting [Encode](#) will cause UUCODE to begin the process of encoding a binary file (any file type actually) into an encoded ASCII file. See the [\*\*Encoding a File\*\*](#) topic for details.

### **Decode Menu Item**

Selecting [Decode](#) will cause UUCODE to begin the process of decoding a previously encoded ASCII file into its binary (original) form. See the [Decoding a File](#) topic for details.



**Exit Menu Item**

Selecting [Exit](#) will quit the UUCODE program.

If UUCODE is running when exit is pressed, UUCODE will first cancel the operation (as if the Cancel button were pressed) and clean up any temporary files used during the encoding or decoding process. Therefore, it may take a few seconds before UUCODE completely exits.

## Configure Menu

You can configure UUCODE various ways to operate according to your preferences. There are configuration options for general file handling and window display, encoding, and decoding operations.

In each configuration dialog box, there are both "Configure" and "Save" buttons. If you choose "Configure," all configuration options chosen will be used for the remainder of the time UUCODE is operation (unless you re-configure UUCODE again). The next time UUCODE is run, the configuration options will be set to the previous defaults. If you choose "Save," then all configuration options will be put into effect and saved as the defaults which will be active the next time UUCODE is run. These defaults are placed in a local file called UUCODE . INI.



**General Configuration**



**Encoding Configuration**



**Decoding Configuration**

## General Configuration

- Become ICON When En/Decoding**
- Stay ICON When En/Decoding**
- Exit When Done En/Decoding**
- Overwrite Existing Files without Asking**
- Use Default File Names**
- No Status Messages**

### **Become ICON When En/Decoding**

Setting this option applies only when UUCODE is run in **Interactive Mode**.

Enable this option if you want UUCODE to run as an ICON while encoding or decoding files. UUCODE will automatically restore to its previous size when the operation is completed.

### **Stay ICON When En/Decoding**

Setting this option applies only when UUCODE is run in **Interactive Mode**.

Enable this option if you want UUCODE to remain an ICON after encoding or decoding files.

The default is that UUCODE will automatically restore to its previous size when the operation is completed.

### **Exit When Done En/Decoding**

Setting this option applies only when UUCODE is run in **Interactive Mode**. This option is automatically enabled when UUCODE is run in **Quiet Mode**.

Enable this option if you want UUCODE to automatically exit after encoding or decoding files.

The default is that UUCODE will remain an active application after the operation is completed.

### **Overwrite Existing Files without Asking**

Setting this option applies only when UUCODE is run in **Interactive Mode**. This option is automatically enabled when **Quiet Mode** is used or when **Multi Part Encoding** is selected.

Enable this option if you want UUCODE to automatically overwrite any existing output files.

In **Interactive Mode**, the default is to ask user permission before any previously existing output files are overwritten.

## Use Default File Names

Setting this option applies only when UUCODE is run in **Interactive Mode**. This option is automatically enabled when **Quiet Mode** is used or when **Multi Part Encoding** is selected.

Enable this option if you want UUCODE to automatically choose the output file name for you. The output file name is either the result of encoding a file or the file name as specified in the encoded file when decoding is used.

For encoding and a single file output, the default file name is the input file name with the extension replaced with ".UUE" for UU encoding or ".XXE" for XX encoding. For example, the default file name for encoding ANYFILE.EXE is ANYFILE.UUE.

For **Multi Part Encoding** output, the output name of each file part is determined automatically. The output file names consist of the first six (6) characters of the input file name or the entire input file name, whichever length is less, with a 1 or 2 digit part number (1 to 99) concatenated, and having an extension of ".UUE" or ".XXE". For example, if an encoded output file required four (4) parts and the input file name was MYARCHIV.ZIP, the output parts would be named MYARCH1.XXE, MYARCH2.XXE, MYARCH3.XXE, and MYARCH4.XXE. Similarly, if a file to be encoded was named ABC.EXE, then the four output parts would be ABC1.UUE, ABC2.UUE, ABC3.UUE, and ABC4.UUE.

For decoding, the default file name is that which is specified in the encoded input file. For example, if the file being decoded is named FILE.UUE and contains the file ABC.ZIP, then the default file name is ABC.ZIP.

The default is to ask user permission to use the proposed output file name. This allows the user to manually change the proposed name if desired.



### **No Status Messages**

Setting this option applies only when UUCODE is run in **Interactive Mode**. This option is automatically enabled when UUCODE is run in **Quiet Mode**.

Enable this option if you want UUCODE to run without requiring user confirmation of success or failure of operations when they are completed.

## Encoding Configuration

- Multi Part Encoding**
- Encode Method**
- End of Line String**
- Max Encoded File Size**
- Generate Checksums on Encoded Output**

## **Multi Part Encoding**

UUCODE supports 12 formats for single or multi part encoding. These are:

- Single file
- SimTel multiple part format
- Comp.binaries Usenet newsgroup multiple part format
- Four of the most common multiple part formats used in Alt.binaries Usenet newsgroups
- UNIX shell archive multiple part format
- DOS UUENCODE/UUDECODE multiple part format
- X-File header style format
- UUXFER multiple part format
- WINCODE multiple part format

Select the format required from the list shown.

All but the single file format are considered multiple part formats.

**Encode Method**

UUCODE supports encoding using either the UU or XX character set. XX encoding uses a different character set which allows for EBCDIC to ASCII conversions. The most popular method, however, is UU encoding. Select UU for UU encoding or XX for XX encoding.

**End of Line String**

Since UUCODE encoded output may be transferred to either PC or UNIX systems, an option is provided to allow an end-of-line (EOL) terminator string specific to the destination computer type. DOS uses a carriage return (CR) followed by a line feed (LF), while UNIX uses only a LF. Select DOS/Windows for the CR/LF end-of-line or UNIX for LF end-of-line terminator.

### **Max Encoded File Size**

When encoding using any of the multiple part formats, the maximum output file size parameter limits the maximum number of characters per part to the value specified. The range for this size is 4500 to 1,048,576 (1M) bytes. The default is 60,000 bytes, which is somewhat less than a typical e-mail program limit of 64K (65,536) bytes. Note that the output file(s) may be slightly larger or smaller than this number due to the variable header and trailer data required for the multiple part formats.

The number of output parts is automatically calculated from the input file size and the maximum output file size. There is a ninety-nine (99) part maximum.

When the single file option is chosen, the encoded output file is in one piece, regardless of size. Therefore the maximum output file size field has no effect with this option (and is grayed to so indicate).

See the [Use Default File Names](#) section for information on how the output files are named.

## **Generate Checksums on Encoded Output**

To assist in detecting errors introduced by the transport network carrying UU or XX encoded files, a checksum can be placed in each encoded output file. Decoding programs capable of detecting these checksums, such as UUCODE, can indicate whether the encoded input appears corrupted.

UUCODE uses the same checksum algorithm as the UNIX "sum -r" command. Each part, whether single or multiple parts are output, has appended at the end of the part a checksum line containing the checksum data for that part as well as it's encoded size in bytes. Additionally, the last part also has appended checksum data for the original input file and input file size so that the decoded output can be checked against the original input file that was encoded.

Enable this option to automatically generate the appropriate checksum data. Please note that the UNIX shell archive format does not support the checksum data option.

## Decoding Configuration

- Automatic Detection of Decoding Type
- Decoding Type
- Decode Method
- Decode Multiple Input Files as 1
- Test Checksum Data if Present



### **Automatic Detection of Decoding Type**

When decoding, the default is to automatically detect the file format used and decode as required. Due to the prolific number of formats, for which there is unfortunately no standard - defacto or otherwise, it is possible that UUCODE can be confused by an unsupported format. To assist in decoding, a specific **Decoding Type** format can be requested.

## **Decoding Type**

When Automatic Detection of Decoding Type is disabled, the specific format to look for is chosen via a list of radio buttons.

## **Decode Method**

UUCODE supports decoding using either the UU or XX character set. Normally, UUCODE determines UU or XX decoding Operation Based on Input File Name. If a specific decoding method is required, select UU for UU encoding or XX for XX encoding. This will force the particular decoding method regardless of the input file name.

### **Decode Multiple Input Files as 1**









Normally, all parts of an encoded file are present in a single file to decode. If the encoded input is spread across multiple files, enabling this option will cause all input files to be treated as one large input file which UUCODE can then decode. This option allows decoding of a single input file split across any number of encoded files with the parts in *any order*. This option is also available from the command line as explained in **Command Line Arguments**.

### **Test Checksum Data if Present**

UUCODE will automatically determine if any file checksum data is present to help detect possible errors introduced during transport across a network. If such checksum data is present, UUCODE will test that data against its own calculation and report if any mismatches occur. UUCODE does not abort decoding if mismatches are found, but rather warns that a potential problem with the decoded file exists.

## Help Menu

This documentation is in both printed and on-line electronic form.

-  [Index Menu Item](#)
-  [Commands Menu Item](#)
-  [Procedures Menu Item](#)
-  [Keyboard Menu Item](#)
-  [Using help Menu Item](#)
-  [About Menu Item](#)
-  [Register Menu Item](#)
-  [On-Line ORDER FORM Menu Item](#)
-  [Hand-written ORDER FORM Menu Item](#)

**Index Menu Item**

Selecting [Help](#) will run the on-line help mechanism of UUCODE. You will be placed in the index section which allows you to easily jump to any topic of interest.

## **Commands Menu Item**

Selecting the Commands item provides details on the menus in UUCODE.



### **Procedures Menu Item**

Selecting this item provides details on how to use UUCODE for **Encoding a File** or **Decoding a File**.

## **Keyboard Menu Item**

Selecting this item provides details on using the Keyboard short-cuts, i.e. the hot-keys.

## **Using help Menu Item**

Selecting [Using help](#) provides details on using the help system itself.

**About Menu Item**

Selecting [About](#) will cause UUCODE to display the version of this software, copyright information, and the location to send the registration fee or order.

## **Register Menu Item**

Selecting [Register](#) will cause UUCODE to display the dialog box in which you can enter your name and password to stop the registration reminder screens. UUCODE will indicate whether or not the registration information entered has passed its validity test.

### **On-Line ORDER FORM Menu Item**

Selecting **On-Line ORDER FORM** will cause UUCODE to display an on-line order form which can be filled-out on line and then printed. This order form automatically calculates the order price from the items selected. Print the form by selecting **Print** button in the dialog box. *You must fill out the **DISTRIBUTION SURVEY** by pressing the **Survey** button before the **Print** button is enabled.*

**Hand-written ORDER FORM Menu Item**

Selecting [Hand-written ORDER FORM](#) will cause UUCODE to display the order form which is ready to print and fill-in by hand. Print the form by selecting [File|Print Topic](#) from the menu.





## Keyboard

The following hot keys are defined for UUCODE.

<b>Hot Key</b>	<b>Command/Procedure</b>
F1	Access on-line help file <u><a href="#">contents</a></u>
Shift-F1	Display the About dialog box
F2	<u><a href="#">Encoding a File</a></u>
F3	<u><a href="#">Decoding a File</a></u>
F5	<u><a href="#">General Configuration</a></u> of UUCODE for its file handling and window display characteristics of operation
F6	<u><a href="#">Encoding Configuration</a></u> of UUCODE for the UU or XX encode operation
F7	<u><a href="#">Decoding Configuration</a></u> of UUCODE for the UU or XX decode operation
F8	Display the electronic on-line UUCODE order form.
Shift-F8	Display the hand-writable UUCODE order form.
F9	Display the Registration dialog box to enter your registration information.
F10	Exit UUCODE



## **System Requirements and Usage**

-  **CPU Instruction Set**
-  **Windows Operating Systems and Modes**
-  **Disk Usage**
-  **Memory Usage**

## **CPU Instruction Set**

UUCODE was written for Windows 3.1 equipped PCs using the 80386 instruction set and will only work on PCs using 80386, 80386SX, 80486, 80486DX, 80486DX2, 80486SX, and Pentium or compatible processors.

## Windows Operating Systems and Modes

UUCODE runs under Windows 3.1, and Windows for Workgroups 3.1 and 3.11.

UUCODE runs only in Windows standard or enhanced modes; *not* in real mode. This is the same as Windows 3.1.








## **Disk Usage**

UUCODE uses approximately 450K of disk space. The exact amount depends on your disk's cluster size. The RTF file for this manual takes approximately an additional 1 megabyte. If you don't know about your cluster size, don't worry about it.

## **Memory Usage**

While running, UUCODE uses about 60K of memory.

## General Operating Information

-  **DOS and UNIX End Of Line (EOL) Characters**
-  **Cooperative Multitasking**
-  **Temporary File Usage**
-  **Multiple Instances and File Sharing**
-  **File Permissions**
-  **Encoding Character for Binary Zero**
-  **Multipart File Limitations**

## **DOS and UNIX End Of Line (EOL) Characters**

For decoding, UUCODE automatically handles any combination of DOS and UNIX end-of-line terminators; those with both Carriage Return (CR) and Line Feed (LF), or just CR or just LF. There is no need to pre-process encoded input to convert the UNIX format to DOS.

When generating encoded files, UUCODE can be configured to use either a DOS or UNIX **End of Line String**.

## **Cooperative Multitasking**

UUCODE is a cooperative program with the other applications running under Windows such that it shares the CPU during the encoding and decoding processes which use extensive file I/O. This results in slightly slower performance than if no CPU sharing were done. Even so, it is not recommended that modem communications take place while UUCODE is running; file transfers might fail.



## **Temporary File Usage**

To allow multiple part decoding with the file parts in any order, UUCODE creates temporary files. This means that the disk on which UUCODE is operating must be writable and have sufficient free space of twice the size of the input file(s).

## **Multiple Instances and File Sharing**

UUCODE allows multiple instances of itself. Since Windows 3.1 is a multi-tasking system, it is possible that the same encoded or decoded file might be used by UUCODE or other applications at one time. This could happen if the user chooses the same file name twice by mistake. To solve this problem, UUCODE takes advantage of Windows file locking mechanism and does not allow any other application to write the UUCODE output file as it is being created. However, multiple read accesses to the input files are allowed.

## **File Permissions**

When decoding a file, the permissions identified in the encoded file are ignored. The output file is always created with read and write permissions for all, which is what DOS can support.

To be consistent with the UNIX versions of encoding, the file permission data are set to owner read/write, with read-only permission for group and other (file mode "644").

## **Encoding Character for Binary Zero**

Although the UU encoding specification allows either an ASCII space (hex character 20) or a ASCII back-quote (` - hex character 60), UUCODE will always UU encode its binary zero (0) output using the back-quote (`) character. This allows for easier viewing of the UU encoded file.

For decoding, UUCODE will accept input files with either the space or the back-quote character without problems.

XX encoding always uses the plus '+' character for binary 0.

## **Multipart File Limitations**













For the decoding operation, UUCODE allows the individual parts of a multiple part encoded binary file to exist in any order, e.g. not be restricted to sequential order. However, *all* the parts of one encoded binary file *must* be present in one or more input files to be decoded.

In addition, no more than one binary file's worth of encoded data can be present in any set of encoded files.

In summary, the encoded file(s) must contain all the parts of one and only one binary file for the decoding operation. In either case, UUCODE will detect and report any violations of these requirements should they occur.

## Encoding Header Formats

Ideally, only one header format would be required for UU or XX encoded files. However, the multiple part encoded formats were designed to allow a large file transfer by encoding one large binary file into multiple parts for transmission across a network. This is necessary because the E-mail on many systems cannot handle more than 64K of data (or less) for any one file. Today, large binary files are common which would prohibit sending them via E-mail unless the multiple part format is used. The multiple part formats supported by UUCODE are briefly described below. In the examples, it is assumed that three parts are used and that they are in order (not a requirement for UUCODE).

-  **Single File Format**
-  **Internet's SIMTEL20 Archive Site Format**
-  **Usenet Newsgroup comp.binaries Format**
-  **Usenet Newsgroup alt.binaries Format 1**
-  **Usenet Newsgroup alt.binaries Format 2**
-  **Usenet Newsgroup alt.binaries Format 3**
-  **Usenet Newsgroup alt.binaries Format 4**
-  **UNIX Shell Archive Format**
-  **DOS uuencode/uudecode Program Format**
-  **X-File Format**
-  **UUXFER Program Format**
-  **WINCODE Program Format**

## Single File Format

```
begin 644 archive.zip  
Encoded data goes here  
end
```

## Internet's SIMTEL20 Archive Site Format

```
----- Part 1 of 3 -----  
begin 644 archive.zip  
Encoded data goes here  
----- End of part 1 of 3 -----  
  
----- Part 2 of 3 -----  
Encoded data goes here  
----- End of part 2 of 3 -----  
  
----- Part 3 of 3 -----  
Encoded data goes here  
end  
----- End of part 3 of 3 -----
```



## Usenet Newsgroup comp.binaries Format

```
Archive-name: archive/part01  
BEGIN--cut here--cut here--  
begin 644 archive.zip  
Encoded data goes here  
END--cut here--cut here--
```

```
Archive-name: archive/part02  
BEGIN--cut here--cut here--  
Encoded data goes here  
END--cut here--cut here--
```

```
Archive-name: archive/part03  
BEGIN--cut here--cut here--  
Encoded data goes here  
end  
END--cut here--cut here--
```

## Usenet Newsgroup alt.binaries Format 1

NOTE: For the four following alt.binaries formats, UUCODE will accept decoding "Subject:" lines with any of the following formats. For encoding, however, the output is exactly as shown in the examples below. 'X' identifies the current part number and 'Y' identifies the total number of parts.

The filename may have an extension (filename.ext), no extension (filename or description) or be preceded by a dash '-' (- filename.ext).

The parts count identifiers can be in any of the formats (X/Y), (partX/Y), (part X/Y), (part X of Y), [X/Y], [partX/Y], [part X/Y], or [part X of Y].

```
Subject: archive.zip (1/3)
e-mail header data goes here
```

```
BEGIN --- CUT HERE --- Cut Here --- cut here --- archive.zip
begin 644 archive.zip
Encoded data goes here
--
```

```
Subject: archive.zip (2/3)
e-mail header data goes here
```

```
BEGIN --- CUT HERE --- Cut Here --- cut here --- archive.zip
Encoded data goes here
--
```

```
Subject: archive.zip (3/3)
e-mail header data goes here
```

```
BEGIN --- CUT HERE --- Cut Here --- cut here --- archive.zip
Encoded data goes here
end
--
```

## Usenet Newsgroup alt.binaries Format 2

Subject: archive.zip [part 1/3]  
*e-mail header data goes here*

```
BEGIN-----> cut here <-----  
begin 644 archive.zip  
Encoded data goes here  
END-----> cut here <-----
```

Subject: archive.zip [part 2/3]  
*e-mail header data goes here*

```
BEGIN-----> cut here <-----  
Encoded data goes here  
END-----> cut here <-----
```

Subject: archive.zip [part 3/3]  
*e-mail header data goes here*

```
BEGIN-----> cut here <-----  
Encoded data goes here  
end  
END-----> cut here <-----
```

## Usenet Newsgroup alt.binaries Format 3

Subject: archive.zip [1/3]

*NOTE: blank lines after e-mail header indicate start of encoded data*

begin 644 archive.zip  
*Encoded data goes here*

--

Subject: archive.zip [2/3]

*NOTE: blank lines after e-mail header indicate start of encoded data*

*Encoded data goes here*

--

Subject: archive.zip [3/3]

*NOTE: blank lines after e-mail header indicate start of encoded data*

*Encoded data goes here*  
end

--

## Usenet Newsgroup alt.binaries Format 4

Subject: archive.zip [1/3]  
*e-mail header data goes here*

BEGIN-----CUT HERE-----  
begin 644 archive.zip  
*Encoded data goes here*  
END-----CUT HERE-----

Subject: archive.zip [2/3]  
*e-mail header data goes here*

BEGIN-----CUT HERE-----  
*Encoded data goes here*  
END-----CUT HERE-----

Subject: archive.zip [3/3]  
*e-mail header data goes here*

BEGIN-----CUT HERE-----  
*Encoded data goes here*  
end  
END-----CUT HERE-----

## UNIX Shell Archive Format

```
#!/bin/sh
#
# This is a shell archive. Cut everything off before
# the #!/bin/sh and feed the rest to /bin/sh
#
part=1
file=archive.zip
sed -e '/^BEGIN/d' -e '/^END/d' << \End_of_Section > $file.uue.$part
BEGIN----- archive.zip ----- part 1/3 ---
begin 644 archive.zip
Encoded data goes here
END----- archive.zip ----- part 1/3 ---
End_of_Section
echo $file, part $part extracted.
if [ `echo $file.uue.[0-9]* | wc -w` = 3 ]; then
cat $file.uue.* | uudecode
if [ $? -gt 0 ]; then
    echo Error encountered when uudecoding pieces...
exit 1
fi
echo $file successfully uudecoded. Removing uencoded pieces.
rm $file.uue.[0-9]*
fi
exit
```

```
#!/bin/sh
#
# This is a shell archive. Cut everything off before
# the #!/bin/sh and feed the rest to /bin/sh
#
part=2
file=archive.zip
sed -e '/^BEGIN/d' -e '/^END/d' << \End_of_Section > $file.uue.$part
BEGIN----- archive.zip ----- part 2/3 ---
Encoded data goes here
END----- archive.zip ----- part 2/3 ---
End_of_Section
echo $file, part $part extracted.
if [ `echo $file.uue.[0-9]* | wc -w` = 3 ]; then
cat $file.uue.* | uudecode
if [ $? -gt 0 ]; then
    echo Error encountered when uudecoding pieces...
exit 1
fi
echo $file successfully uudecoded. Removing uencoded pieces.
rm $file.uue.[0-9]*
fi
exit
```

```
#!/bin/sh
#
# This is a shell archive. Cut everything off before
# the #!/bin/sh and feed the rest to /bin/sh
```

```
#
part=3
file=archive.zip
sed -e '/^BEGIN/d' -e '/^END/d' << \End_of_Section > $file.uue.$part
BEGIN----- archive.zip ----- part 3/3 ---
Encoded data goes here
end
END----- archive.zip ----- part 3/3 ---
End_of_Section
echo $file, part $part extracted.
if [ `echo $file.uue.[0-9]* | wc -w` = 3 ]; then
cat $file.uue.* | uudecode
if [ $? -gt 0 ]; then
    echo Error encountered when uudecoding pieces...
exit 1
fi
echo $file successfully uudecoded. Removing uencoded pieces.
rm $file.uue.[0-9]*
fi
exit
```

## DOS uuencode/uudecode Program Format

section 1 of uuencode 5.24 of file archive.zip  
*NOTE: blank lines indicate start of encoded data*

begin 644 archive.zip  
*Encoded data goes here*  
*NOTE: blank lines indicate end of encoded data*

section 2 of uuencode 5.24 of file archive.zip  
*NOTE: blank lines indicate start of encoded data*

*Encoded data goes here*  
*NOTE: blank lines indicate end of encoded data*

section 3 of uuencode 5.24 of file archive.zip  
*NOTE: blank lines indicate start of encoded data*

*Encoded data goes here*  
end  
*NOTE: blank lines indicate end of encoded data*



## X-File Format

X-File-Name: archive.zip  
X-Part: 1  
X-Part-Total: 3

BEGIN-----cut here-----  
begin 644 archive.zip  
*Encoded data goes here*  
END-----cut here-----

X-File-Name: archive.zip  
X-Part: 2  
X-Part-Total: 3

BEGIN-----cut here-----  
*Encoded data goes here*  
END-----cut here-----

X-File-Name: archive.zip  
X-Part: 3  
X-Part-Total: 3

BEGIN-----cut here-----  
*Encoded data goes here*  
end  
END-----cut here-----

## UUXFER Program Format

archive.zip section 1/3 UUXFER X.Y

```
BEGIN-----CUT HERE-----  
begin 644 archive.zip  
Encoded data goes here  
END-----CUT HERE-----
```

archive.zip section 2/3 UUXFER X.Y

```
BEGIN-----CUT HERE-----  
Encoded data goes here  
END-----CUT HERE-----
```

archive.zip section 3/3 UUXFER X.Y

```
BEGIN-----CUT HERE-----  
Encoded data goes here  
end  
END-----CUT HERE-----
```

## WINCODE Program Format

[ Section: 1/3 File: archive.zip Encoder: Wincodex vX.Y ]

*NOTE: blank lines indicate start of encoded data*

begin 644 archive.zip

*Encoded data goes here*

*NOTE: blank lines indicate end of encoded data*

[ Section: 1/3 File: archive.zip Encoder: Wincodex vX.Y ]

[ Section: 2/3 File: archive.zip Encoder: Wincodex vX.Y ]

*NOTE: blank lines indicate start of encoded data*

*Encoded data goes here*

*NOTE: blank lines indicate end of encoded data*

[ Section: 2/3 File: archive.zip Encoder: Wincodex vX.Y ]

[ Section: 3/3 File: archive.zip Encoder: Wincodex vX.Y ]

*NOTE: blank lines indicate start of encoded data*

*Encoded data goes here*

end

*NOTE: blank lines indicate end of encoded data*

[ Section: 3/3 File: archive.zip Encoder: Wincodex vX.Y ]

## **If You Encounter Problems**

If you encounter problems, check the following areas for more information. If those suggestions don't work, please contact Sabasoft, Inc. for technical support. See [Contacting Sabasoft](#).



**What if the Multiple Part UU Format I Use Isn't Supported**



**What if the Output File Seems Corrupt**

## **What if the Multiple Part UU Format I Use Isn't Supported**

If you *know* that the format of your encoded input file is not one of those supported by UUCODE, then you must manually put the encoded data portion only of each file part into the correct order for UUCODE to work. Any e-mail header information must also be stripped out. Keep only the portions starting with 'begin' and up to and including 'end'. Use any standard text editor, such as Windows NOTEPAD or DOS's EDIT, to do this.

If you are not sure what multiple part type is used, try running UUCODE anyway and check that UUCODE reports no errors and check the output file as per the **What if the Output File Seems Corrupt** section. The formats supported by UUCODE are described in **Encoding Header Formats**.

## **What if the Output File Seems Corrupt**

Typically UUCODE is used to send binary files containing compressed archives. Those files typically end in extensions of .ZIP, .ARC, .LZH, or .ZOO. If you find that UUCODE reports a complete and correct decoding but your archive file has problems (e.g. checksum errors), then it is likely that UUCODE cannot filter any e-mail header information contained within the file or the multiple part format used is not supported by UUCODE.

Try manually putting together the encoded file using any ASCII text editor such as Windows NOTEPAD or DOS's EDIT. Then re-run decoding on that file.

## Technical Support

Questions about the operation of UUCODE or problems encountered can be directed to Sabasoft, Inc. as indicated in **Contacting Sabasoft**.

Technical support is provided free of charge to registered users.

## Contacting Sabasoft

You may contact Sabasoft, Inc. by two means:

1. US Mail - write to:

Sabasoft, Inc.  
924 Bayhill Avenue  
Naperville, IL 60565 USA

2. Electronic mail at America On-Line (AOL) - send a message to:

[sabasoft@aol.com](mailto:sabasoft@aol.com)

Check with your e-mail provider to learn how to send e-mail to America On-Line.

Note 1: AOL has an incoming mail limit from other networks (via the Internet) of 250,000 ( $\cong$  250K) bytes. If you are a subscriber to AOL, then you may attach a file which avoids this limitation.

Note 2: Due to Internet restrictions, you cannot send binary data from other networks to AOL. If you need to send binary data, then use UUCODE! (Remember the size limitation as stated in Note 1) If you are a subscriber to AOL, then you may attach a file which avoids this limitation.

*Telephone support is not provided.*



## **LICENSE**

**YOU SHOULD READ CAREFULLY THE FOLLOWING TERMS AND CONDITIONS BEFORE USING OR DISTRIBUTING THIS SOFTWARE. USING OR DISTRIBUTING THIS SOFTWARE INDICATES ACCEPTANCE OF THESE TERMS AND CONDITIONS.**

## **COPYRIGHT**

This software program (hereafter the "Software") and documentation (hereafter the "Documentation") is copyrighted and remains the property of Sabasoft, Inc. All rights are reserved.

## **LICENSE**

Sabasoft, Inc. hereby grants you a limited, non-exclusive license to use the Software and accompanying Documentation contained in the shareware distribution package (hereafter called the "Shareware Package") you received. The Shareware Package consists of copies of the Software and Documentation received via an electronic transfer mechanism or an electronic storage medium including, but not limited to, diskette, hard disk, removable disk, tape or optical devices such as CD-ROM or WORM.

You may use the Shareware Package on one or more computers provided there is no chance that it will be used simultaneously on more than one computer. If you need to use the Software on more than one computer simultaneously, please contact Sabasoft, Inc. for information about site licenses.

You are not obtaining title to the Software or Documentation or any copyright rights. You may not sublicense, lease, rent, sell, convey, translate, modify, decompile, incorporate into another product, convert to another programming language, reverse engineer or disassemble the Software or Documentation for any purpose.

## **TERM AND TRIAL PERIOD**

You are permitted to use the Shareware Package for a trial period (hereafter the "Trial Period") of sixty (60) days after receipt of the Shareware Package. After such time, if you continue to use the Shareware Package, you must register your copy with Sabasoft, Inc. Failure to register after the Trial Period constitutes termination of this license. Upon registration, a password (hereafter called the "Password") will be provided to eliminate the registration reminder screen.

This license is effective until terminated. You may terminate it by destroying the Software, Documentation, and all copies thereof. This license will also terminate if you fail to comply with any of the terms or conditions herein. You agree, upon such termination, to destroy all Software, Documentation, and copies thereof.

## **LIMITED WARRANTY**

THE SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. SABASOFT, INC. DISCLAIMS ALL WARRANTIES RELATING TO THIS SOFTWARE, WHETHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND ALL SUCH WARRANTIES ARE EXPRESSLY AND SPECIFICALLY DISCLAIMED.

## **EXCLUSIVE REMEDY**

YOUR EXCLUSIVE REMEDY AND SABASOFT, INC.'S ENTIRE LIABILITY ARISING FROM OR IN CONNECTIONS WITH THE SOFTWARE, DOCUMENTATION, AND/OR THIS LICENSE (INCLUDING WITHOUT LIMITATION FOR BREACH OF WARRANTY OR INFRINGEMENT) SHALL BE, AT SABASOFT, INC.'S OPTION, THE REPAIR OR REPLACEMENT OF THE SOFTWARE DISKETTES OR REFUND OF REGISTRATION FEES.

## **LIMITATIONS OF LIABILITY**

IN NO EVENT SHALL SABASOFT, INC. OR ANYONE ELSE WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION, OR DELIVERY OF THIS SOFTWARE SHALL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL OR INCIDENTAL DAMAGES, ARISING OUT OF THE USE OR INABILITY TO USE SUCH SOFTWARE EVEN IF SABASOFT, INC. HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES OR CLAIMS.

Some states do not allow the exclusion of the limit of liability for consequential or incidental damages, so the above limitation may not apply to you.

## **BACKUP**

You may make as many copies of the Shareware Package as you need for backup purposes only.

## **DISTRIBUTION**

You may distribute the Shareware Package to others via an electronic transfer mechanism or electronic storage media provided that no files are deleted from the Shareware Package or are modified in any way. Files may be added to the Shareware Package expressly for the purpose of identifying the party distributing the Shareware Package. You may also place the Shareware Package into any archive or compressed format for ease of distribution.

You may not distribute the Password to others under any circumstances.

## **GENERAL**

This agreement shall be governed by the laws of the state of Illinois and shall inure to the benefit of Sabasoft, Inc. and any successors, administrators, heirs, and assigns. Any action or proceeding brought by either party against the other arising out of or related to this agreement shall be brought only in a STATE or FEDERAL COURT of competent jurisdiction located in DuPage County, Illinois. The parties hereby consent to in personam jurisdiction of said courts.

# ORDER FORM

Note: You may print this form by selecting File|Print Topic from the File menu on the help system.

Program: **UUCODE 3.1**

Name \_\_\_\_\_

Company \_\_\_\_\_

Address 1 \_\_\_\_\_

Address 2 \_\_\_\_\_

City \_\_\_\_\_ State/Province \_\_\_\_\_

Zip/Postal Code \_\_\_\_\_ Country \_\_\_\_\_

E-mail Address \_\_\_\_\_

(important for information on updates)

<b>Choose only one of the following:</b>	<b>Price Each</b>	<b>Quan</b>	<b>Net</b>
UUCODE 3.X Registration Only for New Customers	US\$10.00	_____	_____
UUCODE 3.X Registration Upgrade from v2.X	US\$4.00	_____	_____
UUCODE 3.X Registration Upgrade from v1.X	US\$8.00	_____	_____

**Options: (add to registration price)**

Copy of 3.X on disk and printed manual ( <i>You get a disk only if you choose this option. Quantity limited to registration quantity only.</i> )	US\$6.00	_____	_____
--	----------	-------	-------

Diskette size (circle one)    5.25" 1.2 MB    3.5" 1.44 MB

Additional Shipping <i>outside</i> the continental USA (per copy on disk)	US\$5.00	_____	_____
--	----------	-------	-------

State Sales Tax (IL Residents only) [6.75% × net total of above]		_____	_____
---	--	-------	-------

Please Remit Total (US\$ Funds) (see **PAYMENT TERMS**): \_\_\_\_\_

Returning the **DISTRIBUTION SURVEY** would also be greatly appreciated.

Mail this form to: Sabasoft, Inc.  
924 Bayhill Avenue  
Naperville, IL 60565 USA

## PAYMENT TERMS

Please remit payment to:

Sabasoft, Inc.  
924 Bayhill Drive  
Naperville, IL 60565 USA

Payment must be in US\$ funds.

Personal check, bank check or money orders are accepted. Credit cards are not accepted. Purchase orders are also welcome. Terms are 30 days net. Cash is accepted, however, Sabasoft Inc. will not accept responsibility of funds lost in the mail. If cash is sent, the purchaser agrees to accept all risks of loss.

Purchasers outside the USA are also responsible for any VAT, duties, or tariffs.

If you are ordering a large quantity of this program, please see **Contacting Sabasoft** for information regarding site license agreements.

Please allow from 1 until 3 weeks for delivery.

## DISTRIBUTION SURVEY

We would appreciate if you told us where you found or heard about our software. Please check the appropriate items and return with your registration. This form may also be filled-out electronically by pressing the Survey button in the on--line order form dialog box (see **On-Line ORDER FORM Menu Item**).

Note: You may print this form by selecting File|Print Topic from the File menu on the help system.

Distributor Type:    \_\_\_\_\_ Bulletin Board System (BBS)  
                          \_\_\_\_\_ Friend  
                          \_\_\_\_\_ Archive Site (i.e. SimTel, ftp.cica.indiana.edu)  
                          \_\_\_\_\_ On-line Service (CompuServe, GENie, Prodigy, AOL, etc.)  
                          \_\_\_\_\_ Public Domain or Shareware Software distributor

Name of distributor: \_\_\_\_\_

Location of distributor: \_\_\_\_\_

Telephone of distributor: \_\_\_\_\_

E-mail/FTP address of distributor: \_\_\_\_\_

