

Amigaguide

COLLABORATORS

	<i>TITLE :</i> Amigaguide		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 24, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Amigaguide	1
1.1	UUxT 3.1 Documentation	1
1.2	Using UUxT	2
1.3	Using UUxT-GUI 3.1	5
1.4	UUSplit	7
1.5	REGISTRATION -- Support Amiga Shareware!	9
1.6	Installing UUxT	9
1.7	Changes in UUxT 3.1	10
1.8	Suggestions/Bug Reports	11
1.9	DISCLAIMER (blah, blah, blah)	12
1.10	LhA	12

Chapter 1

Amigaguide

1.1 UUxT 3.1 Documentation

UUxT 3.1 -- The Ultimate "UU" Utility for The Amiga

Copyright (c)1995 Asher Feldman

The UUxT package contains three utilities designed to make uuencoding and decoding files as painless as possible, especially for beginners.

I've finally been able to put in the time to fix several problems (see

Changes

) and release an update! Thanks to everyone

who has registered, making this update possible! And if you still haven't, please do!

UUxT

- The main program

UUxT-GUI

- Graphical frontend for UUxT

UUSplit

- Easily splits uuencoded files

The UUxT package is SHAREWARE. If you use it, you've got to pay for that right. There is no longer a separate version for registered users, this is it. It's got all the features the full registered version had. Therefore all references in the docs to the registered version also applies to the public release of 3.1. This is sort of an experiment. I'm hoping people who haven't registered will see how nice it is and actually do the right thing. Who knows...

Registration

Installation

Changes

Suggestions/Bug reports

Legal Disclaimer

1.2 Using UUxT

UUxT is an all in one, full featured uuencode/decode utility. ←
 Simply,
 it allows you to convert binary files into standard ASCII (and back). This
 allows you to transmit the files over text only networks, such as UseNet
 and internet Email.

UUxT was designed to be easy to use, yet powerful at the same time. All
 traces of antiquated Unix syntax has been removed, and uuencoding and
 decoding with UUxT is no more difficult than decompressing and LhA'd file.
 If you would rather not use a CLI command, a workbench frontend is also
 included (

```
UUxT-GUI
).
```

UUxT does more than just plain old encoding/decoding and contains the
 following features:

- o Splitting the output when uuencoding. (Some networks have a limit on
 the size of email messages. UUxT allows you to split the encoded file
 for you, so it's all ready for sending.)
- o Batch decoding. (You can decode many files at once, as well as split
 files without joining them. UUxT also has wildcard support.)
- o Batch encoding. You can encode multiple files into one text file.
- o Automatically skips over mail headers and other junk that can end up in
 encoded files.
- o Ability to
 LhA
 compress files for you, and then
 uuencode the resulting compressed file, all in one step, thus
 saving you time. You can also decode and de-LhA in one step.

USAGE

When UUxT is run from the command line with no options you should see
 the following (the message at the bottom is only in the Public release):

```
UUxT Version 3.0b UNREGISTERED Release Copyright (c)1994 Asher Feldman
USAGE: UUxT [-<options>] <command> <archive name> [LhA archive name]
        [files] [dest=dir]
```

```
Commands: a - encode          l - LhAencode
```

x - decode u - LhAdecode

Options: -p - pad spaces
 -s<l> - split encoded file (in kbytes)
 (i.e. -s20 to split in 20k blocks, -s2001 for 200 line blocks)

UUxT is SHAREWARE. If you like it, please register it. Registration is US\$10 which should be sent to the address below. (Please include your email address.)

You will then receive the registered release, which contains a faster (by a factor of 10) version of UUxT, as well as an enhanced version of UUxT-GUI. Refer to the documentation for more information on registration and program usage.

Asher Feldman, 4791 Calle de Tosca, San Jose, CA 95118, USA

As you can see, there are four commands, and two optional options. The stuff enclosed in []'s are optional, or only required for certain operations. Stuff enclosed in <>'s are required for all operations. (NOTE: commands and options are case sensitive.)

Commands

a - encode -- This is the command used for encoding one or more files. Here are several examples:

UUxT a myfile.uue myfile (Encodes myfile to myfile.uue)

UUxT a myfile.uue myfile1 myfile2 (Encodes myfile1 and myfile2 to myfile.uue)

You can also use wild cards. Instead of the above line, you may want to use:

UUxT a myfile.uue myfile?

x - decode -- This is the command used for decoding (surprised?) uuencoded files. You can decode as many files as you want at one time. Examples:

UUxT x myfile.uue (Decodes myfile.uue)

You can also use the "dest" option when decoding. This lets you choose the directory where the files will be decoded to. See below:

UUxT x myfile.uue dest=df0: (Decodes myfile.uue to df0:)

UUxT x part1 part2 part3 dest=ram: (Decodes part1, part2, and part3 to ram:. Each part may be part of a single file which has been split, or individual files.)

l - LhAencode -- This option first uses LhA (which must be installed) to compress the desired files into a single a file, and then uuencodes that file. (Yes, it's two tasks in one!)

The usage isn't as strait forward as the other commands, so check out the examples:

```
UUxT l myfile.uue myfiles.lha file1 file2 file3
```

This does the following: First, LhA is used to compress file1, file2, and file3 into myfile.lha. Myfile.lha is then uuencoded into myfile.uue. Make sure you specify the "LhA file name"! (In this example the "myfiles.lha") If it was accidentally left out of the above example, file1 would be used instead, which is not what you want! After doing its work, UUxT will delete myfiles.lha.

u - LhAdecode -- This option is used just like the 'x' option, but after each file is decoded LhA is called to decompress it. It should only be used if only LhA'd files are being decoded.

Options

The options are new to UUxT 3.0. The options are optional. (That's why they're called 'options'.) To use an option just put them before the command. The options are:

-p - Pad Spaces -- Some mail and news readers (Portal is one) do a bad thing which can cause uuencoded files to become corrupted. Some UUencoders (but not UUxT) use the space as a valid character. And sometimes, a space can end up as the last character in a line. See below:

```
M"@H*"@D*"@H*"@H*"@H* 0 (" @," P4# P4' !P8' !PH*"0D*"@H*"@H*"@H*
M"@H*"@H*"@H*"@H*"@H*"@H*"@H*"@H*"@H*"@H*"@K_P 1" , ! #
M 1$ A$! Q$!_ \0!H@      8# 0$!          ! 4&!P," 0 ("0H+ 0
                                     ^^
```

Note these two spaces

The above is part of an actual uuencoded file. Note The two spaces at the end of the last line. Some mail/news readers figure those spaces are unnecessary and just get rid of them. That line would then be two characters shorter than all the others, and UUxT's would no longer see it as a valid line. Those "evil" mail/news readers are few and far between, but they do exist, so I added this option. If you use this option, and a line looks like it could be valid except for line length, UUxT will pad spaces at the end to make it the same length as all the other lines. UUxT only does this if a line starts with an 'M' and no lower case characters are used, but it's still possible for junk to be included, corrupting the out file.

You should therefore ONLY use this option if you are stuck with a mail/news reader that has the "space problem." (PORTAL Users: This means you!) Usage:

```
UUxT -p x myfile.uue      (decodes myfile.uue with padding)
UUxT -p u myfile.uue      (same as above, then de-lha's)
```

-s - split encoded file (in kbytes or <l>ines)

This option is for use when encoding files. It is used when you want the resulting encoded file to be split into several smaller parts. Many networks have length limits, which require you to split up uuencoded files before sending. If posting a file to the Usenet, it is usually safe to split a file into approximately 50k blocks. This option requires that you have

UUSplit
installed. Usage:

```
UUxT -s50 a myfile.uue myfile
```

This encodes myfile to myfile.uue. Myfile.uue is then split into approximately 50k chunks. The resulting files will be name "myfile.uue.000", "myfile.uue.001", etc. The unsplit "myfile.uue" will still exist too.

UUxT 3.0b now lets you split files from a specified number of lines. All you do is add an 'l' after the number. For example:

```
UUxT -s500l a myfile.uue myfile
```

This encodes myfile to myfile.uue. Myfile.uue is then split into exactly 500 line chunks. The resulting files will be named "myfile.uue.001", "myfile.uue.002", etc. The unsplit "myfile.uue" will still exist too.

That covers all of UUxT's commands and options. If that all seems a bit daunting, don't worry. I've tried to make the command line interface close to Lha in usage. If you don't want to bother with it though, you will find UUxT-GUI much easier to use, and it offers all the power and options the command line version does.

1.3 Using UUxT-GUI 3.1

UUxT-GUI is the workbench interface for UUxT. It requires that you have UUxT 3.1 installed in your command path (i.e. c:) and also reqtools.library V38, which is included in this archive. (Use the script to install.) Reqtools.library is (c) 1991/1992 Nico François. Thanks!!

Please note the following changes in 3.1: There are no longer any popup requesters in the registered version. Also after each operation, the InFiles list is automatically cleared for you now.

UUxT-GUI 3.1 is a major update to 1.0. It now supports all the features of UUxT, making the advanced options easy to use. UUxT-GUI 3.1 requires

Kickstart 1.3, but give you some extra options if you have 2.0+. (Only on the

Registered version

.) This will probably

be the last version to support 1.3. A future 2.0 only version should be much more compact and use less memory.

The window is divided into sections, each section labeled as to its function. The sections are:

- InFiles - This is a ListEditor, which lets you select as many files as you want. If you click on Add you can enter a filename into to string gadget. Or you can click on Disk which will open a file requester while allows you to select multiple files. They will then be put in the scrollable list.
- OutFile - This is used only when using Encode or LhAencode. It is the name of the file the InFile(s) will be encoded to. The Disk gadget opens a file requester. However, you can't double click to select a file here. This is so you don't accidentally choose a file you don't want to overwrite.
- Dest Dir - This is of use for decoding. If you enter a directory in the string gadget (or select on by clicking Disk and using the requester) that directory will be used for decoding files too. If nothing is specified, then files will be decoded to the directory UUxT-GUI was run from.
- LhA Name - This string gadget is used only when using LhAencode. See the documentation for UUxT to find out more about what the "LhA Name" is for.
- Options :
- Pad Spaces - This checkbox is off when UUxT-GUI is first opened. Click on it if you want to use it. (See section on UUxT for info on what this is for.)
- Decode - This is a cycle gadget for choosing the operation you want. When UUxT-GUI is first run it is set to Decode. Click on it to cycle through the choices, Decode, Encode, LhA Decode, and LhA Decode.
- Iconify - If you are using the Public Release this gadget is shaded out. If you are using the Registered Release it will also be shaded out if you don't have Workbench 2.0+. (Upgrade!) When you click on Iconify UUxT-GUI closes its window and puts an appicon on your workbench. For know it just uses the default tool icon (which is very nice if you use MagicWB) but I am planning on allowing you to choose the icon it uses in a future version. You can drop uuencoded files into the appicon and
-

they will be decoded for you. When iconified, UUxT-GUI remembers if you chose "Pad Spaces" and will use that option if you chose it. However, UUxT-GUI doesn't remember the Dest Dir. Instead, it automatically changes directory to where the file is located and decodes it there. (It changes back after it's done.) If you double click on the appicon UUxT-GUI will re-open its window and remove the appicon. This feature makes decoding lots of very easy. (All you have to do is drop them on the icon!)

About - If you haven't registered this opens up the same requester that is there when you run UUxT-GUI. My email address is in here for sending bug reports or suggestions. (Hopefully just suggestions :)

Split - If you choose split (off by default) the number gadget will be unshaded, and allow you to choose how big (in KiloBytes) you want each chunk to be. (40 is default.) To find out what split does though, you'll have to look in the

UUxT

section. (You must have UUSplit ← installed to use this option.) If you add an 'l' after the number it will split by a number of lines instead of by kbytes.

Ok, now that you (hopefully) understand what everything does you're ready to get to work. Using UUxT-GUI is very simple. Just choose the options you want to use, the files and click on the big START gadget. UUxT-GUI will then do its thing. Simple as that.

1.4 UUSplit

UUSplit is a special file split utility designed specifically for use with UUencoded files. UUSplit accepts two options, the infile name and the chunksize. The chunksize is in Kilobytes and is how big each portion will be. (Approximately. See below)

UUSplit isn't you normal file splitter. You can split a file two ways, by line or by size. In the first method you would put, say, 3000 lines per file. By size you might specify 30k per file. Splitting by size is usually more convenient, but has one problem. It usually ends up breaking lines up, which means you have to join the files back together before decoding them. UUSplit is a sort of a compromise. It's like a line splitter that's posing as a size splitter.

CHANGES IN UUSplit 1.1

Version 1.1 adds several new features. Among them:

- o UUSplit adds begin and end lines to each file, showing what part the file is out of however many parts. I.E:

```
BEGIN -CUT HERE----- "myfile.uue" part 001 of 003 ----CUT HERE-
```

This way your files are perfectly ready for mail or posting.

- o You can now split files by number of lines as well as by size in kbytes.
- o UUSplit creates an environmental variable called "lastpart" which contains the number of parts in the file it just split. This can be useful for making script files, as they can know how many parts there are.
- o UUSplit now starts counting for the files from 001 (i.e. first part is file.001 instead of file.000 as in UUSplit 1.0.)

How It Works

UUSplit doesn't actually read the exact number of bytes specified by the user. Instead it reads 256 bytes less. (256 bytes is usually the maximum number of characters that can ever end up on one line. In practice, you'll probably never see that many on one line.) This amount will almost always be in the middle of some line. UUSplit then reads until it reaches the end of a line. That way each file always begins and ends with a full line. This method insures that the split files will always decode without the need to join each part back together.

Usage

When run with no options you should be greeted by:

```
UUSplit 1.1 Copyright (c)1994 Asher Feldman
Usage: split [inputFile] [Size<l>]
Size is in kbytes (30 for 30k) or lines (2001 for 200 lines)

creates [inputfile].001, [inputfile].002, etc..
```

I think that's pretty strait forward. To split "myfile" into 40k blocks use:

```
UUSplit myfile 40
```

The split parts will be called "myfile.000", "myfile.001", etc.

To split in 400 line blocks use:

```
UUSplit myfile 400l
```

However, most of the time you won't be using UUSplit, UUXt will be using it for you. UUXt will automatically call it for you when using the '-s' option. You can of course use it on it's own without UUXt if you want.

To Do (There's Always Something)

The current version of UUSplit doesn't support multiple files or wildcards, as it's mainly meant to be used by/with UUXT. I may add such features at a future date though. If I get enough registrations to make it worth my time, I would also like to write a 2.0+ only version of UUXT-GUI which would be much smaller and have a real gadtools or maybe MUI based interface.

1.5 REGISTRATION -- Support Amiga Shareware!

The entire UUXT package is Shareware. (UUXT, UUXT-GUI, and UUSplit) If you like it, and find yourself using it more than once you are obligated to register. The version you've got right now is the full registered version with nothing taken out and no annoying requesters put in. Please don't let this stop you from registering! You are still obligated to do so!

The Registration fee is only US\$10, which I consider very reasonable for shareware. To register, send US\$10 (US currency only) to

Asher Feldman
4791 Calle de Tosca
San Jose, CA 95118-1923
USA

Please enclose your name, mail address, any comments you may have and most importantly, your *EMail Address*. (Along with the money :-)

And thanks in advance for registering!

Asher Feldman (EMail: asher@shell.portal.com)

1.6 Installing UUXT

Before using, UUXT and UUSplit should be copied to a directory ← that's in your command path (The C: directory would be best.) UUSplit only needs to be around if your going to use UUXT's splitting abilities, but you should still copy it, even if you don't plan on using it.

Before UUXT-GUI can be used, you must have reqtools.library installed. It is included in this archive, and if you don't already have it, you can just run the included install script. UUXT-GUI needs to be able to find UUXT, so you should copy UUXT to an appropriate directory before use.

You also need
LhA
installed in your system in order
to use the LhA Encode and LhA Decode options.

1.7 Changes in UUxT 3.1

UUxT 3.1 Public Release

Changes since UUxT 3.0b

- o Ten times faster!

Changes since UUxT 3.0

- o Added support for new features in UUSplit 1.1
- o Added nice little header and footer line.

Changes since UUxT 2.3

- o Added option to choose destination directory.
- o Added integrated file splitting
- o Added pad space option
- o Fixed batch encode file size bug
- o Fixed miscellaneous decode bugs
- o Added AmigaGuide documentation

UUxT-GUI 3.1 Public Release

Changes since UUxT-GUI 2.0b

- o Major bug fix, should be much more stable.
- o Added Iconify option to public release.
- o The InFiles list is now cleared after every operation.

Changes since UUxT-GUI 2.0a

- o Fixed bug that didn't let you use file names with spaces (i.e. Ram Disk:file.uue)
- o Added support for UUSplit 1.1

Changes since UUxT-GUI 1.0a

- o Complete rewrite since 1.0a
-

- o Multiple file selection allowing:
- o Batch decoding
- o Batch encoding
- o Incorporated all UUxT 3.0 new features. If UUxT does it, so does UUxT-GUI.
- o Fixed nasty bug causing UUxT-GUI to lockup your machine or cause random crashes if your stack was too small.

UUxT-GUI 2.0

Registered Release

 Changes since UUxT-GUI 2.0 Public

- o Activated ICONIFY option, which lets you quickly decode files by dropping their icons on a special AppIcon. (Kickstart 2.0+ only)

UUSplit 1.1

- o Add splitting by number of lines
- o Added headers and footers that tell filename and filepart for each file.
- o Sets environmental variable "lastpart" that contains the number of the last part.
- o Now counts file from 001 instead of 000.
- o Fixed bug that caused crashing if the stack was too small.

1.8 Suggestions/Bug Reports

If you have any suggestions, find a bug, or just need some help ↵
 then
 please contact me. You can email me at one of the following addresses:

PORTAL: Wizard0

Internet: asher@shell.portal.com (Try this one first)

OR

Wizard0@cup.portal.com (If above doesn't work)

Or, of course you can write to me the old fashioned way. My address is in the

Registration Section

. And don't forget to register while you're at it. Thanks! And thanks to all who have reported bugs and made suggestions! I've tried to add everything people have suggested, and I hope it shows.

Asher Feldman

1.9 DISCLAIMER (blah, blah, blah)

WARNING: Reading this section all the way through may cause drowsiness, excitability, and/or inhibit your ability to operate heave machinery.

THIS PROGRAM IS PROVIDED ON AN "AS IS" BASIS, NO WARRANTIES ARE MADE, EITHER EXPRESSED OR IMPLIED. IN NO EVENT WILL I, ASHER FELDMAN, BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING FROM ANY USE OR MISUSE OF THIS PROGRAM. THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THIS PROGRAM IS ASSUMED BY YOU. (ZAP) THIS PROGRAM IS BEING INTERRUPTED WITH A SPECIAL NEWS BULLETIN. A MONSTER HAS BEEN SPOTTED OVER TOKYO! IT'S GAMARA! <INSERT MST3K GAMARA THEME SONG> (ZAP)

We now return to the regularly scheduled program, already in progress.

(If you're confused, don't worry.)

1.10 LhA

LhA is required in order to use certain features of UUxT. LhA is Shareware and is copyright Stefan Boberg.
