

MAKEUEF V1.7

Welcome to MakeUEF, an accurate data format computer cassette preservation program.

A second programmer has extended MakeUEF from V0.3b. MakeUEF now comprises of an accurate version for professional cassettes and a simpler version for amateur cassettes with the file name MakeUEFam.exe. MakeUEF V1.7 was released in February 2005.

This program attempts to convert genuine BBC/Electron tapes to tape images via sound sample files (CSW, WAV or RAW format) or via your sound card. The professional tape version uses only CSW files. All data waves, baud rates and gap lengths are now recorded with great accuracy.

This manual mainly addresses the professional cassette version of MakeUEF. The amateur cassette version supports more input formats and has a wider wavelength toleration range to allow easier reading of cassettes. Also the UEF file output is simpler with standard baud rate and pilot tone lengths.

IMPORTANT NOTICES

- Before November 2004, all UEF tape images available on the Internet will have been created with earlier, 0.x releases of MakeUEF; hence they only contain program data. The gap lengths, baud rates etc. will have not been retained.
- As cassettes can now be preserved to a highly accurate level, please create/recreate UEF (and CSW) files from your original cassettes and send them to the Acorn Preservation project at <http://www.acornpreservation.org>. Files created by the new, 1.x versions of MakeUEF will have a secondary extension of HQ (meaning High Quality), to distinguish them from the older files.

Try to also send the LOG files created while making UEF files and small sized scans of the cassettes of both sides in order to ascertain what cassette was used.

N.B. the Stairway to Hell web site – <http://www.stairwaytohell.com> – will continue to collect and archive low-quality BBC/Electron tape images that have been created with MakeUEF v0.3b (or MakeUEFam).

- Do not remove or edit any blocks in UEF files produced by MakeUEF.
- Do not convert sound sample files to a different sampling rate. This would cause a loss of accuracy. 44100 samples per second is the minimum sampling rate required by MakeUEF. Higher sampling rates are ok and will produce CSW files no larger than lower rates.
- Do not use a stereo cassette player or a Hi-Fi cassette deck unless you change the stereo read head to a mono one.

HOW TO USE

THE RECORDING PHASE:

Refer to the document 'How to record WAV files for conversion to CSW.PDF' that is supplied with CSW Viewer.

THE CSW FILE CREATION PHASE:

Refer to the document 'How to record WAV files for conversion to CSW.PDF' that is supplied with CSW Viewer.

THE UEF FILE CREATION PHASE:

Start a command prompt window by using the Run... option on the start menu and typing CMD. MakeUEF only runs on Windows at present.

Run the program with no switches or the -h switch to get a list of options.

Run MakeUEF from a command prompt box in Windows with, for example the instruction:
makeuef -i 747 (Doctorsoft) (B) (Tape) (Early) [side-lab].csw

If you haven't ran MakeUEF before use the -g switch explained in the detailed descriptions of command line options section.

Type the correct cases of letters. MakeUEF should speed through the CSW file creating a UEF file. Listen for a beep, which indicates an error and use the pause key and another such as the space bar to observe the output. Also a LOG file will be produced that the output can be observed in.

You should rerun MakeUEF with the -s switch and a mean baud rate reported by a previous run of MakeUEF. You don't have to be particularly accurate choosing the mean value. Even if all the data blocks were read this switch will help MakeUEF to get waves at the starts and ends of files.

If you see no data blocks, consider whether you used the correct phase shift settings. A few tapes require a change of phase shift between files at a gap, particularly compilation tapes. MakeUEF will expect the initial phase shift to be 0 degrees although the initial phase shift with the UEF format is 180. The phase shift can be set with the -w switch.

If you see any bad checksums or misread blocks you can try running MakeUEF again with a different baud rate with the -s switch. Some tapes have a non-standard data format, which would require switches explained later. In the worst case you may have to record again with different tone/volume settings and make a new CSW file. Control-C or Control-Break can be used to terminate MakeUEF when you know it isn't working as expected. The -r switch can be used to identify a point where reading is failing.

Always look for hiss or electrical hum being read as readable waves. There is more about hiss and electrical hum in the limitations section.

I have found the average tape is converted easily. Firebird tapes tend to be difficult ones and often have non-standard data blocks. Using the tapes baud rate helps particularly when it far from the ordinary value (1200). You should always use the -s switch at least once when producing a final UEF file or there will be a message in the UEF file saying it was not used.

Your motherboard speaker has to be working to hear a beep.

MakeUEF is not designed for putting together a UEF file from multiple files.

DIRECT SOUND INPUT TO MAKEUEF AMATEUR

Start MakeUEF Amateur with a command such as makeuefam -o mytape. Play the cassette and watch for all the blocks being read. When all the blocks are read, press Escape or Q. Otherwise stop the program with Control-C or Control-Break.

Refer to the document 'How to record WAV files for conversion to CSW.PDF' that is supplied with CSW Viewer for information on recognising phase shift. A phase shift of 0 requires -ls. A phase shift of +90 requires -lp. A phase shift of +180 requires -ls and -p. A phase shift of +270 requires -lp and -p. CSW files should have only a phase shift of +0 or +180 and only the -p switch is effective.

It might also help to use the `-s` switch. Particularly difficult files will be read better by following the CSW file creation process.

Turn off loudness buttons on amplifiers. Turn off Dolby noise reduction on cassette players.

FILE NAMING CONVENTIONS

The Acorn Preservation project has information on naming files.

DETAILED DESCRIPTION OF COMMAND LINE OPTIONS

- **i <file name and extension>**

Sets the input file to use. A directory may be specified.

- **o <file name>**

Sets the output file to use. A directory may be specified. The extension “hq.uef” will be appended with the professional version and “uef” is appended with the amateur version. The file name defaults to the input file name if this switch is not used.

This switch is necessary with sound card input.

- **h**

Displays the help message, just as passing no command line parameters does.

- **v**

Only included with the professional version.

No verbose messages.

- **p**

Only included with the amateur version.

Makes MakeUEF Amateur expect a phase shift of +180 or +270 degrees instead of 0 or +90 degrees.

Sound editor programs can be used to ascertain the phase shift of a file.

- **z <data block number format code>**

Only included with the professional version.

Sets the data format to use for a block. The first parameter is the number of the data block to use this data format with and all further blocks. The second parameter is the data format to use. E.g. 5 7E1 would make MakeUEF use a format of 7 data bits, even parity and 1 stop bit with the fifth block and all later blocks. 6 8N2 would make MakeUEF use a format of 8 data bits, no parity and 2 stop bits with the sixth block and later.

REG is the data format to use for standard format blocks. REG has same data format as 8N1. 8N1 can be used with non-standard block formats to suppress error messages.

Read the messages to discover the block number required and trial and error to discover the format. Alternatively you could try looking at the reported security waves to determine the format. The -r switch is useful for seeing if the correct data format has been used.

Supported formats are 7E1, 7E2, 7O1, 7O2, 8E1, 8N1, 8N2, 8O1 and REG. The blocks are not necessarily the same as audible blocks - the starts and ends of blocks may be within audible blocks.

Firebird tapes sometimes have over 40 data format changes. Most tapes with a non-standard data format have only a small number of changes.

My CSW Viewer program can be used to figure out what data format a block uses.

– **s <file number baud rate>**

With an Electron or BBC, the value of bits is determined by the quantity of wave variations in a given span of time. This switch sets how the program calibrates its interpretation of the length of that time span.

If not used MakeUEF does a calculation on the input wave frequency to determine the correct span to read data stored with the default BBC/Electron baud rate. E.g. 44100 frequency / 1200 long waves per second = 36.75 samples per long wave.

If you know the baud rate of a file, probably by a previous run of MakeUEF, you can specify the baud rate. After running MakeUEF with a sound file you can discover the baud rates within it and then rerun MakeUEF using the baud rates. Probably there is roughly one, baud rate used unless the tape is a compilation.

If you find that using an incorrect low baud rate helps you should rerecord with an adjustment to the tone.

This option will be necessary for 300 baud rate files. CSW files should be made with a low frequency filter of 200Hz and a high frequency filter of 700 Hz for 300 baud rate files.

If this switch is not used, at least, for the first file, a message will be stored in the UEF file with this fact.

– **r**

Only included with the professional version.

Reports the sizes of data blocks and the first pulse of the start bit of the currently unreadable next byte. This option is mainly for doing non-standard blocks and finding the position of waves that are not being read correctly.

– **g <signature>**

Add the given message to the UEF file. The message will be recorded in the Windows registry and always written to later files until this option is used without a message. I recommend keeping it brief with just your name and your cassette player model. This is important if you are submitting UEF files to the www.acornpreservation.org site. Some cassette players negate the polarity of the signal, which is a reason for specifying what the model is. The www.acornpreservation.org site would like to see 'Created for www.acornpreservation.org.' in the signature.

– **f <some number>**

Only included with the amateur version.

If working with RAW sound files, there is no way for the program to find out at what frequency the sound file is recorded. This is necessary information if the frequency is not the default. The number will be assumed to be in decimal unless proceeded with 0x - making it hexadecimal.

DEFAULT VALUE is 44100.

– **t <u or s>**

Only included with the amateur version.

This is only for RAW files. MakeUEF supports only mono RAW sound files with 8bit samples, but is happy to accept signed or unsigned data. A 'u' here selects unsigned reading of raw data files and a 's' selects signed.

DEFAULT VALUE is 's'.

– **l <s or p>**

Only included with the amateur version.

Only for WAV and RAW files and sound card input.

MakeUEF can use a peak method, which is suitable for waves with a phase shift of +90 or +270 from an ordinary sine wave.

The sign method is suitable for waves that start at a +0 or +180 degree phase shift from an ordinary sine wave.

The -p switch should also be used for +180 and +270 phase shifts.

DEFAULT VALUE is 's'.

– **d**

Only included with the amateur version.

Only for WAV and RAW files.

This option turns on DC offset fixing. This feature positions waves over the zero line if they are away from it, which is common at the start and end of files. The input file has its values fixed and written to a temporary file before the input file is deleted and the temporary file is renamed with the input file name. It is only necessary to fix a files DC offsets once.

The CSW.exe program performs this function better as well as other benefits so it is not recommended to fix DC offsets before using CSW.exe via CSW Viewer.

Do not use this feature with professional tapes for submission to the preservation project.

– **b**

Do not beep on errors.

– **c**

Do not compress the UEF file.

– w <sample number phase shift number>

Only included with the professional version.

Sets a change point of phase shift. The first number is a sample during a gap before the block with the changed phase shift. The second number is the phase shift, which must be 0 or 180. A sound editor can be used to find a point of change. Unfortunately Gold Wave does not have the capability of showing samples on the X-axis but other shareware programs such as Cool Edit 2000 have this capability.

The default phase shift at sample 0 is 0 degrees.

It is often necessary to use this switch: -w 0 180.

HARDWARE REQUIREMENTS

MakeUEF Amateur should work with any sound card that has a Windows driver and supports 8bit data capture at a rate of 44100 or 22050 samples per second. Some SoundBlaster cards do not have these rates. Adjust your cards mixer via Windows to select the line input socket.

Using a dial-up connection at the same time as using direct sound input may not work. Some computers may require background utilities such as internet firewalls, anti-virus and anti-spyware to be turned off while using direct sound input. Electron saved files may require more processing power if direct sound input is to be used. CSW is the recommended format to use with both versions of MakeUEF.

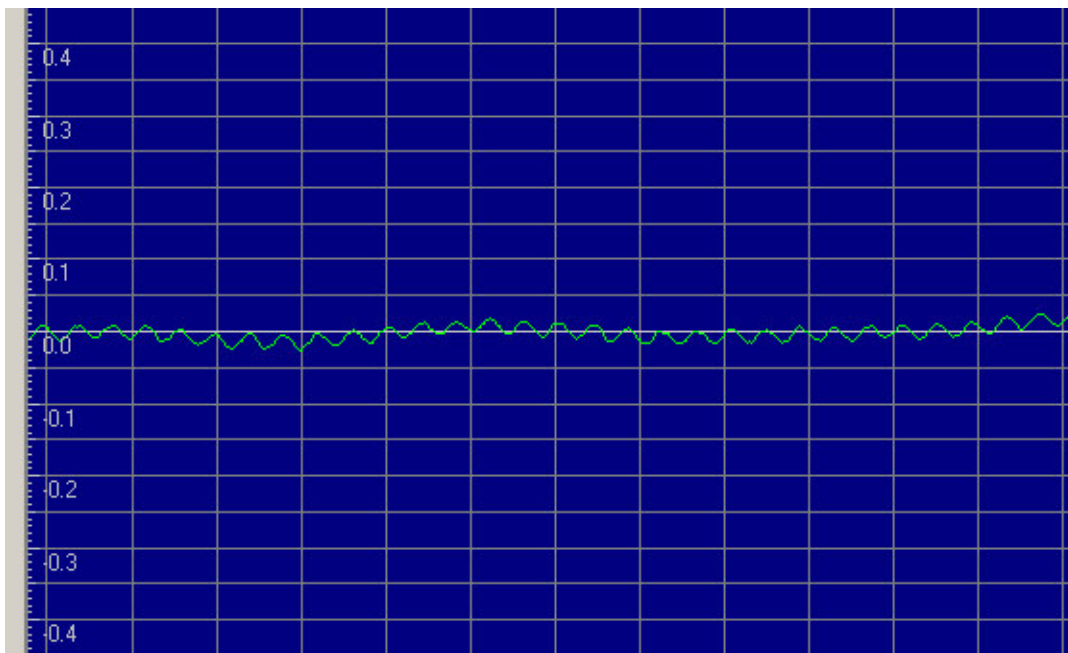
LIMITATIONS

- **Hiss or electrical hum between files can sometimes be read as security waves or, less likely for hiss, pilot waves. The baud rate will usually be noticeably different to data blocks. It can be fixed by making the waves unreadable in a gap between data with a sound editor. With Gold Wave selecting an area starting at a crossing point of the zero line and then using the filter 'low pass static' with an initial cut-off frequency of 400Hz and a steepness of 5, on the effects menu does this. DON'T entirely silence the area because this does not work well with making CSW files. Alternatively you could rerecord the tape with less volume, which should pick up fewer hisses but that does not work so well with hums. With tapes that have very small gaps you will need to carefully zoom in to the gaps with a sound editor to ensure there is no hiss or hum waves. Otherwise you would only need to filter the waves if MakeUEF reports that possibly hiss were been written, before it finishes. The security waves will have random 0's and 1's from hiss, and from hum they may be all short waves and maybe written as a pilot tone.**

Many badly recorded tapes may have a fragmented pilot tone, which IS data to be encoded. The limit of 7 waves for a group prevents hiss and hum being read most of the time.

Extremely short gaps should be examined with a sound editor for hiss and hum. Firebirds Estra would have hum turned into security waves without making the hum unreadable.

Electrical Hum:



- Minor changes in baud rate within data blocks and following pilot tones are not recorded. Nor are they recorded within leading pilot tones. CSW files do record minor baud rate changes.
- **Dropouts on the tape caused by bad cassette players will NOT make UEF files that fully represent the original cassette. Dropouts are common at the end of the last file of a program, on the trailing tone. If you have a tape with dropouts only make a UEF file for submission if no one else submits the tape and also make it known that there are dropouts. Do not be concerned about a fragmented pilot tone unless dropouts caused it.**
- A group of waves must amount to at least 7 or it will be counted as gap.
- The CSW format unifies the amplitude of all waves, so no wave is missed no matter how quiet it originally was. However with other formats used by the amateur version waves with a lower volume than 1/10 of the range will be counted as gap.
- The CSW.exe V2.0 program sometimes produces files with all lowercase names.
- The CSW.exe V2.0 program sometimes produces files with the 'number of pulses' 0.
- Fast sections of pilot tones that are sometimes at the start of a pilot tone are better recorded as a CSW file with a higher than usual high frequency filter. A suitable frequency would be 5000Hz instead of 3000Hz.
- Data blocks are limited to 128K by MakeUEF and by the Acorn tape block format to 64K.
- Phase shift changing and baud rate changing can only be done at gaps between files.
- Checksums are not calculated for non-standard blocks whether they have one or not.
- Dummy bytes in pilot tones with a different phase shift to the data cannot be recorded correctly.
- All tapes are regarded as having only mono recordings.

- Known cassettes that cannot be made into fully meaningful UEF files include: Welcome, the introductory cassette with the BBC model B which has an error sized wave as the second last wave of the pilot tone before the first data block; Dr. Who and the Mines of Terror, which has a complex encryption scheme. Haunted Abbey. Swarm. Ultron. Apple Pie. Devils Causeway. Android Attack.

MAKEUEF's DEVELOPMENT AND ITS PROCESSING

Thomas Harte made the original program and 100s of UEF files have been made with it. Thomas's program appeared to be free from bugs. One mistake was treating all tapes as inverted sine waves, which made many tapes difficult to read and this continued for a long time until I thought I noticed a cassette with normal sine waves.

After releasing V1.1 it was thought that there are 4 common sine wave types used with Acorn cassettes. I withdrew MakeUEF V1.1 from the public. This again was a mistake. I discovered it is amplifiers with digital IC components that are applying a phase shift to the waves. Fortunately a lot of code I had written was suitable for fixing files recorded with these amplifiers. The next version of MakeUEF professional could have lowered wavelength toleration and have the -p switch removed (which was for 0 degree phase shifts). MakeUEF could then record phase shift positions in the 0115 chunk of the UEF specification, which was changed. A switch similar to the old -p switch, had to be put into MakeUEF V1.2 for inverted sine waves and that is the -w switch.

I started rewriting MakeUEF at the beginning of February 2003. I worked on the new version on and off for many months. As I discovered the general format of files the program was made to record it. An excellent discovery was CSW.exe by Ramsoft, which made sound waves more uniform and greatly compacted huge sound sample files. Also, it allowed easier reading of the starts of files, which is a particularly bad part to read. CSW.exe had a major upgrade with V2.0, which does compression and records polarity better. The CSW format had to be extended again to hold a creators signature.

Support for the non-standard tapes was written in and MakeUEF had an expanded set of manual control switches. Electron saved files with occasional extra stop bits was one more peculiarity that had to be catered for. I made a program for viewing CSW files called CSW Viewer and became more aware of the varying baud rates. Towards the end of developing MakeUEF improvements were made to the accuracy of gap lengths and baud rate recording was added. The baud rates are inserted before data blocks and at the start of any other series of readable waves. A switch for calibration by baud rate was one of the last major features added. There was a calibration switch before that used wavelengths in samples.

Theoretically dummy bytes (explained later) might have a different polarity to the data. Support for this has not been written in since I haven't seen any tapes like this.

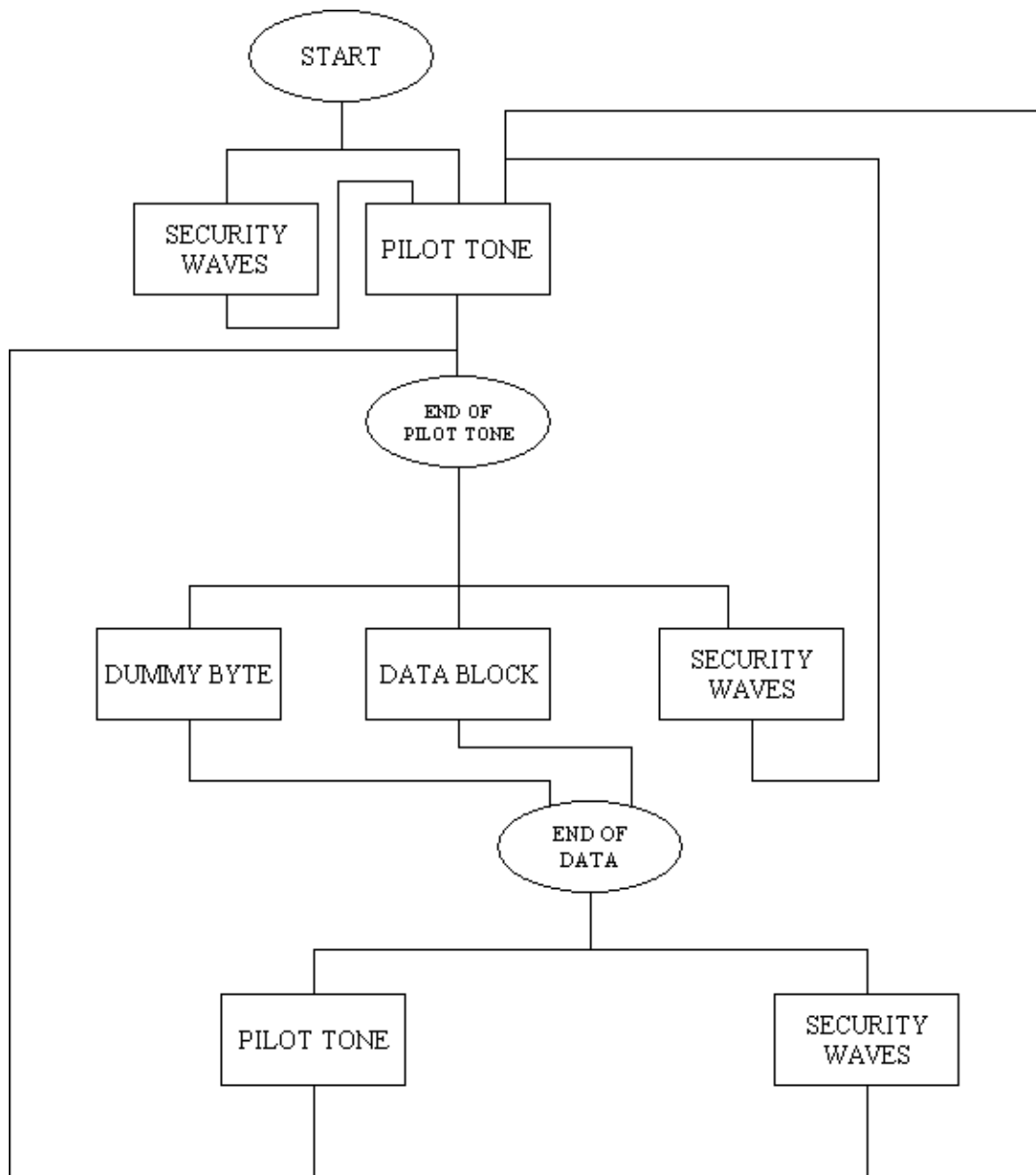
The program had to be thoroughly tested with many cassettes. It also had to be carefully examined for errors, which would make UEF files flawed.

The last work done with MakeUEF V1.1 was to create two different versions, one for professional cassettes and one for amateur cassettes. The amateur version is a minimally modified version of the professional version source code using pre-compilation directives and some new code.

Files made with all versions of MakeUEF before V1.7 have some flaws.

Following is a flowchart that shows how MakeUEF may recognise blocks to put in a UEF file.

MakeUEFs data recognition flowchart:



The leading pilot tone is often preceded by a group of watermark waves I call security waves. These waves are sometimes enough to make a BBC detect a pilot one and read a dummy byte. When this happens a dummy byte is not needed within the pilot tone. Some companies wrote a group of security waves standing alone. In the middle of the pilot tone is almost always, with BBC files, a dummy byte, which fixes a problem with the BBC hardware. BBC OS ROMs before 1.0 do not record a dummy byte.

MakeUEF must see a pilot tone with at least 7 waves before it will look for data. Data blocks are normally followed by a pilot tone but security waves will be recorded if there is any.

Gaps will be recorded wherever they are and are omitted to simplify the diagram. After a gap the current position on the diagram is START.

Data blocks comprise of bytes akin to data bytes passed on serial port connections. Ordinary files use the 8N1 data format.

Dummy bytes are sometimes not before Electron non-standard data block files. MakeUEF would record one in the current data format if possible, otherwise 8N1 and will use chunk &0111 if possible, or record it as security waves.

Publications used:

BBC Advanced User Guide: Acorn BBC/Electron tape format.

Wav.doc by Robert Shuler: WAV file format.

Csw.html by Ramsoft: CSW file format.

COMING IN A LATER VERSION

A new UEF block and support for it in MakeUEF to record a series of waves of one kind (long or short) and followed by a series of waves of the other kind, alternating with different quantities. There are only a small number of tapes like this. At present MakeUEF will record them with a mixture of UEF blocks.

Support for Atom cassettes.

CHANGE LOG

1.7

- Fixed a bug with CSW files with no data.
- Fixed a bug with security waves near the end of a file.
- Fixed a bug with changing the phase shift at the end of a file.
- Fixed a buffer under-run bug while analysing a byte at the end of a file.
- Fixed a gap length bug while finding a pilot at the end of a file.
- Fixed a gap length bug while processing security waves straight after a data block at the end of a file.
- Added the MakeUEF program name to the top of LOG files.
- Added a beep upon major errors.

1.6

- Fixed a bug with baud rate writing. This was a STL library bug.
- Added writing the date and time to LOG files.

1.5

- Fixed a bug with uncompressed CSW files.
- Added writing the command line used to the top of LOG files.
- Added writing the creator's signature to the top of LOG files.

1.4

- Fixed a bug with an error wave following a data block.
- Fixed a bug with an error wave following a dummy byte.

1.3

- Removed reporting of security waves type (pulse or wave) where it is inappropriate.
- Changed the reporting of the number of pulses, to report a start bits first pulse, of a currently unreadable byte.
- Added recording of the length of the gap at the start of a cassette and the trailing gap after the last readable waves.
- Changed recording of a leading pilot tone with MakeUEFam to record a dummy byte.
- Changed type of gap block recorded by MakeUEFam to floating point.
- Changed the UEF minor version to 10 with MakeUEFam.
- Fixed a bug with CSW files and MakeUEFam.
- Fixed a baud rate bug.
- Fixed a bug with security waves following a dummy byte.
- Fixed minor inaccuracy bug with wave calibration.
- Fixed a bug with pilot tone detecting.
- Made dummy bytes get recorded in 8N1 format if possible when the current format cannot accept the byte. If neither data formats are suitable security waves are used.
- Changed compiler to C++ Builder 6 from 4.

1.2

- Removed the -p switch from the professional version.
- Removed creation of temporary uncompressed UEF file.
- Changed the default sine wave phase shift to 0 degrees. It was formerly +180 degrees.
- Changed chunk &0115 in the UEF format specifications to record phase shift.
- Reduced wavelength toleration to +/- 25% for short waves and +/- 15.38% for long waves from +/- 28.57% for all waves, with the professional version.
- Added support for CSW files with signatures.
- Added reporting of gap lengths written.
- Added reporting of pilot lengths written.
- Added output to LOG file as well as the screen.
- Fixed a bug with a long wave of security waves following a data block, which is never actually seen.
- Some improvements to the readability of the output log.
- Fixed the -lp switch and peak-to-peak reading, which did not work.
- Fixed a minor bug with sound card input.
- Added the -w switch.

1.1

- No functional changes.
- Rewrote the file handling code.
- Fixed bugs with DC offset fixing at the ends of files.

1.0:

- Extensive rewriting by a second programmer.
- Now MakeUEF has a professional cassette version and an amateur cassette version.
- Now MakeUEF accurately records the data.
- Now MakeUEF accurately records the gap lengths.
- Now MakeUEF accurately records the mean baud rates.
- MakeUEF is now written in C++. The sound card input code is mostly unaltered from the previous version.
- Files with changing names are now supported.
- Files with occasional extra stop bits between bytes saved by an Electron are now supported and messages are shown. Files with occasional extra stop bits between bytes that are different to the typical ones produced by the Electron may not have a message shown but they are rare. E.g. Electron Pengwyn.
- Files with negated polarity are now better supported.
- Less wavelength toleration, which forces the use of the correct polarity and better sound files.
- Out of sequence block numbers are now supported.
- Incorrect 'last block flag' blocks are now supported.
- Changes of data format within a block are now supported. E.g. Firebird tapes and The Lost Crystal.
- Signatures can now be added to UEF files.
- Changes to the calibration can now be made between files.
- A new block was added to the UEF format to support bundles of long and short waves with or without starting or finishing with a pulse.
- A new block was added to the UEF format to support files with non-standard data formats.
- A new block was added to the UEF format to support changes of polarity.
- A new block was added to the UEF format to support more accurate gap lengths.
- A change was made to the UEF format to the baud rate block for greater accuracy.
- Support for CSW files was added.
- Messages about the input file format are now recorded in UEF files.
- More extensive checking of input file headers and various error messages was added.
- The switches c, d, g, p, r, v and z are new.
- Changes to defaults.
- Removed pilot tone and sample length calibration and added baud rate calibration with the -s switch.
- Added a message recorded in UEF files if the -s switch is not used and a message on the screen.

- Added a hiss written message that appears on the screen.
- Added a 'possible dropout' message that appears on the screen for short gaps.

0.3b:

- small fixes to Windows code, hopefully will no longer crash on exit

0.3:

- iterated towards fixing windows version. Does it work for you?
- added alternate 'peaks' frequency measure for greater reliability on old tapes
- added alternate 'tones' encoding, someone may want it
- allowances for surprise non-ending of files (Elite now closer to working)
- fixed tape image encoding so that high tone hangs for a bit after files, and gaps lie between files - as with real tapes
- adding filename checking for illegal characters before printing names to the VDU - in particular trying to print the name of the '?' file on Acornsoft titles was causing the on screen prompts not to work

0.2:

- completely rewrote data parsing algorithm to sync properly with tape
- altered sound card code to select > 22050Hz frequencies where possible in DOS & Windows

0.1:

- first release

CONTACTING THE PROGRAMMERS

Don't send any large WAV files but short extracts will be fine for an assessment. CSW files are ok but are not good for assessing the phase shift used or the suitability of the tone and volume.

If you have seen what looks like a bug send the input file and the command line used and the LOG file.

New programmer:

Fraser Ross, <mailto:Fraser@members.v21.co.uk?subject=MakeUEF V1.7>

Original programmer:

Thomas Harte, <mailto:ThomasHarte@lycos.co.uk?subject=MakeUEF V1.7>