CSW VIEWER V44

CSW Viewer is used for getting statistics about pulse lengths and wavelengths of computer cassettes by viewing them graphically, to use the mean wave statistics functions. The pulse lengths are read from CSW files, and wavelengths are read from Commodore TAP files that are converted into pulse lengths.

There are standard cassette file formats and many non-standard file formats that have been used with commercial cassette software. To help inspect data that was encoded on a cassette, pulses and waves and certain types of sequences of pulses or waves can be given colour coding.

CSW Viewer is useful for determining why CSW files won't load, often because the waves are not regular enough. Commodore 64 standard files have waves with 3 wavelengths close together. For the Commodore 64 it is necessary to ensure that the waves in a CSW file have particularly regular wavelengths.

Data from Spectrum files, Amstrad files, Commodore non-standard files and any other single wave encoding file can be written on a TZX file, which is a Spectrum orientated data format.

A user of CSW Viewer can indirectly call the CSW.exe program to convert one or more WAV files to CSW. That will require 32-bit Windows 7, 8, 8.1 or 10. There are programs available that create TAP files.

IMPORTANT POINTS

A selected range is inclusive of the last point. The left and right mouse buttons can set the values in the range edit boxes by clicking on a point. The left button sets the start box value and the right button sets the end box value.

The time elapsed before the leftmost point displayed on the pulses chart is displayed below the chart.

All whole number values calculated are rounded.

The time span on the 'File summary' page and the 'Pulses chart' page has 8 decimal places rounded.

The time span of a range of points is given in milliseconds for use with TZX files. Mean wavelengths are given in 48K Spectrum T States that the TZX format mainly uses for pulse lengths.

The 'CSW file creator' page will create files with the CSW format V2.1, which complies with the format update specifications of V2.0.

Wavelengths from TAP files are split into 2 equal length pulses. The overflow pulse length for TAP V0 files is set at 1500.

Wavelengths must be within a tolerated range for colouring data bytes. Short waves have a larger tolerated range.

If a COL file with saved point colours exists then it will be automatically loaded from when a CSW or TAP file is opened.

The colours used are listed here:

Gap	Yellow
Row of pulses	Maroon
Carrier tone	Grey
Data start mark	Teal
Data end mark	Sky blue
Bit 0	Blue
Bit 1	Aqua
Bit 2	Navy
Bit 3	Olive
Bit 4	Lime
Bit 5	Fuchsia
Bit 6	Purple
Bit 7	Red
Start bit	White
Stop bit	Black
Parity mark	Money green
Short wave frequency	White
Medium wave frequency	Medium grey
Long wave frequency	Black

INSTRUCTIONS FOR COMMON USES

Opening a CSW or TAP file:
The 'Open file...' button can be used. Alternatively it is possible to assign a default application to open the files by right clicking on a file, selecting 'Open with' and then 'More apps' followed by 'Look for another app on this PC' and selecting CSW_Viewer.exe and checking 'Always use this app to open .csw files.'

Converting WAV files to CSW files:
Select the 'CSW file creator' page. See the additional document, "How to record WAV files for conversion to CSW.pdf" with CSW Viewer for guidance. The CSW.exe program must be in the same directory as CSW Viewer. You may be prompted to install NTVDM to be able to use CSW.exe. Do not use excessively long directory paths or file names. The CSW file creator will only work with 32-bit Windows. It will not work with any edition of Windows 11. The converted files can be loaded into CSW Viewer with the 'Open file...' button.

Finding the quantity of carrier tone waves, mean carrier tone wavelength and baud rate:
Look for the start of the carrier tone, which you recognise by many pulses of a similar length. Enter the X-axis value for the first carrier tone pulse in the range start edit box. Find the end of the carrier tone and enter the X-axis value of the last pulse in the range end edit box. Press the '1' button on the 'Mean wave statistics' page to see the mean wave frequency, and the mean wavelength in samples and Spectrum T states. The baud rate will be shown and the quantity of waves in the range is also shown. Press the 'Colour pulses in range as carrier tone' button to colour the pulses.

Finding data start mark and data end mark pulse lengths:
After a carrier tone some types of files have a data start mark that is usually 1 or 2 waves. Colour these as a data start mark. There are not many waves to use to get the lengths of data start mark pulses so the values are not very accurate. Follow the instructions above for obtaining a mean carrier tone wavelength. If a mean pulse length is required then divide by 2.

Finding data bit wavelengths/Finding baud rate of KCS type files:

Select the first and last pulses of a data block. The 'Total pulses in range' must be an even number. Press the '2' button on the 'Mean wave statistics' page. Do not select multiple data blocks because the wave coupling will be wrong if there are an odd number of carrier tone pulses between the data blocks. The data analysis chart will be filled with colour coded counts of wavelengths. The frequency of the mean wavelengths will be given and mean wavelengths in samples and 48K Spectrum T States. The baud rate will be given for KCS and Acorn/MSX files. The quantity of short waves and long waves will also be shown.

Finding wavelengths when there are 3 wavelengths used (Commodores):
Select the first and last pulses of a data block. The 'Total pulses in range' must be an even number. Press the '3' button on the 'Mean wave statistics' page. The data analysis chart will be filled with colour coded counts of wavelengths. The frequency of the mean wavelengths will be given and mean wavelengths in samples and 48K Spectrum T States. The quantity of short waves, medium waves and long waves will also be shown.

Colouring pulses of Spectrum/Amstrad/Sam Coupé type bytes:
Enter the X-axis value of the first data pulse in the range start edit box. Find the end of the file by finding the start of a gap or the carrier tone of the following file. Enter the X-axis value of the last data pulse in the end edit box. The 'Total pulses in range' value must be an even number. The "Total 'single wave encoding' bytes in range" value should be a whole number. If it is not you have got something wrong or the data does not finish with a whole byte (uncommon but some non-standard formats have this). Press the 'Colour pulses in range with big endian order – bit 7 to bit 0.' button.

There is checking of wavelengths, which therefore could mean that no pulses will be coloured. There must be a clear distinction between short and long waves.

Colouring pulses of Dragon/Enterprise/Jupiter Ace type bytes:
In general use the same procedure as for the Spectrum but use the 'Colour pulses in range with little endian order – bit 0 to bit 7.' button.

Colouring pulses of Dragon standard files:
A Dragon standard file has a leader, typically with 128 bytes of the hexadecimal value 55. This can be coloured with the 'Colour pulses in range as carrier tone' button. Following that is a very short gap that typically consists of 2 or 4 pulses. Next is a data block of up to 261 bytes that will consist of a multiple of 16 pulses. This can be coloured with the 'Colour pulses in range with little endian order – bit 0 to bit 7.' button. There may be many data blocks separated by gaps of around 0.003s.

Colouring pulses of ZX81 standard files:
There are not any carrier tone or start mark pulses with ZX81 files. '0' bits are represented by 4 waves and '1' bits by 9 waves. There are short gaps between bits with a standard length. Select a range from the first pulse of the first byte to the last pulse of the last byte. Press 'Colour pulses in range as ZX81 bits and bytes.'

There is checking of wavelengths and gap lengths, which therefore could mean that no pulses will be coloured.

Colouring pulses of KCS/Acorn/MSX bytes:
Select a range from the first pulse, of the first start bit, of the first byte, of a data block to the last pulse, of the stop bit, of the last byte, of the data block. It is difficult to know where the last byte ends but this does not matter at this time. Press the correct button for the data format from the 'Kansas City Standard derived encoding' group box. Often it is necessary to adjust the end of the range to get the last stop bit and press the button again.

There is checking of wavelengths, which therefore could mean that no pulses will be coloured. There must be a clear distinction between short and long waves. Short waves are expected to be half the length of long waves.

Colouring pulses of Commodore standard files:
Select a range starting from the first pulse of the file, which is a byte start mark. The end of the range can optionally include a data end mark for the file. Press the 'Colour pulses in range as Commodore VIC-20/C64 standard file' button or the 'Colour pulses in range as Commodore 16/Plus 4 standard file' button.

The data end mark is a long wave followed by a short wave with VIC-20 and C64 files, and a medium wave followed by a short wave with C16/Plus 4 files. This is the only difference expected by the two buttons.

There is checking of wavelengths, which therefore could mean that no pulses will be coloured. There must be a clear distinction between short, medium and long waves.

Extracting data from single wave encoding files:
Select the first pulse of the first byte to the last pulse of the last byte. Press the 'Colour pulses in range with little endian order – bit 0 to bit 7.' button. It is possible that the other endian button will be appropriate instead. Press the 'TZX file creator' tab. Press the 'Create new TZX file or open existing TZX file…' button. Enter a new file name. Press the appropriate 'Append selected pure data and optional trailing gap.' button. Press the 'Close TZX file' button.

The data will start at byte hexadecimal 15/decimal 21 on the TZX file. The bytes will be as they are on the cassette irrespective of which button was pressed. A hex editor such as Hex Editor Neo can be used to reverse the bits if they are little endian.

Saving and loading point colours:
After colouring data pulses press the 'Write point colours to file' button and the point colours will be saved. The file will have the same name as the CSW or TAP file loaded from and the extension COL. This file will hold the number of pulses which will be checked against the number of pulses when you try to load from a saved COL file. If a COL file exists then it will be loaded from when a CSW or TAP file is opened.

Finding a gap length:
Find the first pulse of a gap by seeing where pulses do not look a similar size to the coupled pulses at the end of the preceding data. Enter the X-axis value of the first pulse of the gap in the range start edit box. Find the end of the gap, which usually is the pulse before the first pulse of a following carrier tone and enter its X-axis value in the range end edit box. The time span is automatically shown in seconds and samples.

Finding single wave encoding data quantity in bytes:
Find the first pulse of data, which is the first pulse after the data start mark pulses and is the third pulse after the carrier tone with most Spectrum files. Enter its X-axis value in the range start edit box. Find the last pulse of the data before where the pulses are from a gap or carrier tone and enter its X-axis value in the range end edit box. The 'Total pulses in range' must be an even number. If the "Total 'single wave encoding' bytes in range" is not a whole number you have got something wrong or the data does not end with a whole byte (uncommon but possible).

CSW.EXE

This file is included with CSW Viewer. It requires the virtual DOS machine which is a part of 32-bit Windows up to 10. That is not available with 64-bit Windows and was discontinued from Windows 11.

The CSW format and the CSW.exe utility are both excellent and will accurately preserve almost any computer cassettes readable waves. Only a minority of cassettes will require a wider frequency range than the default range.

The Atari computers are one exception since they detect the general frequency of waves rather than reading individual waves. The waves at the changes of frequency can be quite misshaped.

KNOWN BUGS WITH CSW V2.0

Known bugs with CSW V2.0:
The 'number of pulses' is sometimes set to 0.
The 32-bit 'number of pulses' in the header of CSW files only has a 24-bit number written. This bug generally may appear with tapes of 45 minutes or longer.
The 'number of pulses' is short by 1 if an extremely short WAV file is converted and compression is not used.
The output file name sometimes turns to all lowercase.
Directory path lengths and file name lengths are limited in length if called by CSW Viewer because of a limitation with the Windows API function CreateProcess.
CSW files are created with the name of the WAV file up to the first dot instead of the last.
It is unreliable with very short files with less than 20 samples.

KNOWN BUGS WITH CSW V1.3 (USES FORMAT REVISION V1.1)

Files often finish with a special code for a 32-bit value without a following value.
Polarity is often recorded wrongly.

TAP FORMAT

The similar TAP format used by the Commodore community I have found is slightly less accurate due to conversion from one sampling rate to another, one wave at a time, discarding the fractional part of the converted value. MTAP does this even though rounding only requires adding 0.5 before disregarding the fractional part. It stores only wavelengths in V0 and V1 files. V2 files store pulse lengths for the shorter pulses, which most data pulses comprise of. TAP files are larger than equivalent CSW files because there is no compression. Some converters clean gaps unnecessarily, often altering the sizes of gaps. V0 files do not attempt to record gap lengths.

The best documentation can be found in the manual for the emulator Vice.

KNOW PROGRAMS SUPPORTING CSW V2.0

MakeTZX	V2.33+,
RealSpectrum	V0.97.08+,
Spectaculator	V6.0+,
Electrem Future	2nd binary release+,
MakeUEF	V1.2+.
SpecEmu	V2.3+
TZX2WAV	V0.1+
WinTZX	V0.5A+
B-Em	V1.2+
BeebEm	V3.4+
CSW.dll by Fraser Ross	V1.1+
CSW.dll by Mark Woodmass	
Tape2WAV	V1.2+
ZXSpin	V0.1d+

CSW V2.1 FORMAT EXTENTION

Flags byte:
b0: initial polarity; if set, the signal starts at logical high.

b2: creators signature in header extention data, if set.

Header extention data:
First byte uchar: Number of chunks.
Chunks[Number of chunks]:
  First byte uchar: Chunk ID,
  Second byte uchar: Chunk length,
  The rest uchar: Chunk data.

Currently used chunks:
Chunk ID 2:
Creator's signature:
  UTF8[Chunk length]. Backwardly compatible with printable 7-bit ASCII.

AUTHOR

Fraser Ross

Any bug reports, suggestions for enhancements or requests for previous versions should be sent to FRASER.ROSS8@BTINTERNET.COM