

NAME

axon2run, atf2run – convert Axon Instruments binary or text files to run files

SYNOPSIS

axon2run [-t] [-a] [-I] ABFfile [newrun]

axon2run [-d] [-h] ABFfile ...

atf2run [-t] ATFfile newrun

DESCRIPTION

Axon2run allows you to convert ABF format data files, created by *Axon Instruments* software on MS-DOS (e.g. Axotape or pClamp), or Windows (e.g. Axoscope or pClamp), to the run file format, used by *analysis*(1).

The specified *ABFfile* must be a valid Axon binary format (ABF) data file. (Previous versions of *axon2run* would only work with the older ABF 1.8 or older standard, as used by pClamp/Axoscope version 9 or older, and not the new ABF2 standard supported by version 10. This limitation has now been removed. See the *BUGS* section below for details.) This file must have a **.dat** or **.abf** file name suffix, and one will be added to the specified name if you omit it. The specified *newrun* is the name of the run that will be created to store the converted data. The file name suffix **.frm** is appended to *newrun*, if not already present. If *newrun* is not specified, the name is taken from *ABFfile*, with the file name suffix **.dat** or **.abf** removed, and replaced by **.frm**. Note that the program does not check to see if a run of that name already exists – it will be silently overwritten.

The conversion involves writing a new run header, based on the data in the ABF file header, and writing out one waveform file for each of the channels stored in the ABF file. The calibration information obtained from the ABF file is converted as well, and stored in the run header.

The "File Comment" field, as well as the "Long Description" field, are taken from the *ABFfile*'s header, and are written out as the run description for the *newrun* (with the **.txt** suffix). If the ABF file has any time tags, these will be added as extra lines in the run description file as well.

If the **-t** option is given, *axon2run* will convert the ABF file data into triggered traces, rather than waveforms. For this to work, the ABF file should be one that was captured and stored in one of the "oscilloscope" operation modes, either *LOSSFREEOSC* or *HIGHSPEEDOSC*, and all episodes must be of the same length. If your ABF data file is not in an oscilloscope mode, it will give you a warning. It will still attempt to make triggered traces out of episodes from such a file, provided that the episodes are all of the same length, but the results may not be what you expect – only in the oscilloscope modes are the episodes guaranteed to represent triggered sweeps. This may be appropriate for files of the operation mode *WAVEFORM*, which may have multiple episodes of the same length, with gaps in between. Without the **-t** option, oscilloscope mode data files will be converted into waveforms by concatenating all episodes, without attempting to fill in the gaps between sweeps.

If the **-a** option is given, the ABF file will be converted automatically to triggered traces only if it is in one of the oscilloscope operation modes, and into waveforms for any other operation mode.

If the **-I** option is given, the data samples will have their resolution lowered to 12 bits if the ABF header indicates that the samples are over 12 bits of resolution. This was the default mode in older version of *axon2run*, because older versions of the analysis software couldn't handle samples over 12 bits. Use this option only if you need to maintain backward compatibility, e.g. to reproduce exactly a data analysis procedure that had been done previously.

If the **-d** option is given, *axon2run* will show the start date and time from header of each of the specified ABF data file names. This tells you when the files were captured, even if they've been modified since then. No file conversion is done with this option.

If the **-h** option is given, *axon2run* will show a brief summary of the header information, including a listing of all channels, for each of the specified ABF data file names. No file conversion is done with this option.

Atf2run allows you to convert ATF format data files, created by *Axon Instruments* software to the run file format used by *analysis*(1). It can be helpful in situations where *axon2run* fails to properly convert an ABF file. You can then open that file in Axon's software, and then save as an ATF file. The specified *ATFfile* must be a valid Axon Text Format data file. The *newrun*, which must be specified for *atf2run*, is the name of the run that will be created to store the converted data. If the `-t` option is given, *atf2run* will convert the ATF file data into triggered traces, rather than waveforms.

DIAGNOSTICS

If any error occurs, *axon2run* prints a message, and quits. If a file was being written at the time, it may be incomplete. The exit status is 0 if no errors were detected. Otherwise, the exit status code indicates the error that occurred – 1 for an unknown file type, 5 for bad parameters in the ABF file, 9 if it can't create an output file, etc. A complete list of error codes is in the file **abfiles.h**, in the program's source directory.

ACKNOWLEDGEMENTS

Axon2run was developed using Axon Instruments' File Support Package for Developers, version 1.3.2. Many thanks to Axon Instruments/Molecular Devices for making this library freely available. Support for ABF2 was added in by the SCRC, using the ABF2 file support in the open source **Neo** analysis software's file input routines as a guideline, as Molecular Devices does not provide source code for more recent versions of their File Support Package which include ABF2 support.

BUGS

The version of the File Support Package used to develop *axon2run* is Axon's latest for its DOS-based programs. Its newer Windows-based programs support some extensions to the ABF format, which their DOS FSP can't handle, and the newer FSP is too Windows-centric to port to Linux/UNIX systems. Support for ABF2 was added in by the SCRC in the November 2017 release, based on incomplete documentation of this newer file format. It has been tested with many ABF2 format files, but it's possible that in the future, new incompatibilities will be introduced.

If an older version of *axon2run* gets segmentation faults when converting an ABF file, or you run into other difficulties in converting a newer ABF file, try opening that file in Axoscope and saving it in binary integer format, ABF version 1.8. That should produce an ABF file that can successfully be converted. In the Program Options of Clampex 9 or Axoscope 9, make sure you set the option to 'Ensure data files and protocols are compatible with pCLAMP 6'. This option is still available in version 10, but hasn't always worked reliably. Note that Axoscope 10 and Clampex 10 use a newer ABF2 file format which is completely incompatible with their older File Support Package, upon which *axon2run* is based. In versions of *axon2run* prior to November 2017, you needed to save your ABF files in Axoscope version 9 format before you could convert with *axon2run*. The latest version can handle ABF2, but check carefully that conversion is accurate. In particular, conversion of Annotations, which are assumed to be compatible with Long Descriptions in the older format, has not yet been tested. Conversion of waveform data, scaling and offsets, channel names and units, file comments and time tags, all seem to work correctly.

Prior to November 2017, *axon2run* reduced sample size to 12 bits, for compatibility with older versions of the analysis software. This shouldn't have posed a problem with older data, captured on a Digidata 1200 series A/D, as it was limited to 12 bit resolution. However, newer model Digidata units have 16 bit resolution, and the analysis software has been 16-bit compatible for a long time, so this restriction should have been eliminated sooner. Also, in testing the new version, it was discovered that some 12-bit resolution data was labeled in the ABF header as though it were higher resolution, causing older versions of *axon2run* to further reduce the resolution to only 8 bits! The default is now to preserve the resolution of the original ABF data, or convert floating-point data to the full 16-bit resolution at which it was captured. The `-I` option has been added in case you absolutely need to convert data to a lower resolution, preserving the older behaviour, but in most cases this option should be avoided.

As of the May 17, 2007 release, *axon2run* can now convert ABF floating-point data files, so if you get segmentation faults converting these files with an older version of *axon2run*, try upgrading this program. If all else fails, save the data as ATF and convert using *atf2run*.

Start time field handling

As of March 2015, axon2run transfers the ABF file's start time field to the new counterpart in the run header. Confusion can occur if ABF files move across time zones: Axoscope and pClamp store the start time in local time, while run files use UTC for the start time in their run header. Axon2run does the conversion in the local time zone, which will result in an inaccurate start time in the run file if the ABF file originated from a different time zone. It is best to convert ABF files in the time zone in which they were captured, which can be overridden in software when running axon2run. E.g.:

```
TZ=Europe/Copenhagen axon2run cphdopa013.abf run013
```

See the directory `/usr/share/zoneinfo` on most systems for a list of time zone names known to the system.

SEE ALSO

qm(1), analysis(1), wtsum(1), raster(1), peel(1)
Documentation for Axon Instruments software