# Options for viewing large Recorder WAV files in ARM 2.2

## 1    Introduction

In order to improve performance when viewing large WAV files from sources such as the Microlog Recorder module, ARM has an option for buffering the WAV file for data access rather than reading the entire file into memory, as happens with traditional files containing short waveforms or spectra.

The buffering, which happens by default for any WAV file that is 2 MB or larger in size, is done by ARM only reading in 10,000 samples at a time; this keeps memory usage down while not impacting performance.

For graph plots, it was necessary to introduce some optimisation routes in order to avoid lengthy delays when plotting very large waveforms.  Without this, plotting can be very slow since large portions of the WAV file have to be read every time the plot is redrawn.

This document outlines how this process works, the three different plotting modes used in ARM, and some user settings that can be used to improve performance.

## 2    Plotting modes

ARM has three plotting modes for waveform data:

1.  Direct, all in memory – for any waveform below the buffer threshold (2 MB default).  In this mode, the entire waveform is in memory and all visible samples are plotted.
2.  Medium buffer – used when plotting buffered WAV files when there are more than 30,000 samples in the view.
3.  Cached – used when plotting buffered WAV files when there are more than 400,000 samples in the view.

Both of the buffered modes are triggered if the full waveform has 30,000 samples or more and the samples to be plotted are more than four times the number of pixels forming the width of the graph.

### 2.1    Cached plotting

When a buffered WAV file is read and plotted for the first time by ARM, the program scans the entire file and builds up a snapshot of the waveform, comprising up to 25,000 "block" or zones – each of these blocks represents $1/25,000^{th}$ of the period of the waveform and records the highest and lowest amplitude values found in the block.  The actual plot is done by drawing one vertical line per block, representing the minimum to maximum.

When the samples to be plotted are 400,000 or more, ARM uses these plot zones to display the plot.  This is very efficient.

The 25,000 blocks are stored in the data stream file (PDS) created by ARM, so there is only one major read operation carried out; all subsequent access for plotting the record uses the snapshot blocks.

Cursor and other data operations involve direct file access, using the 10,000 sample buffer method.

## 2.2 Buffered plotting

The snapshot blocks are used for plotting the waveform even if the user zooms in to a portion of the recording.  At some point, however, the number of blocks to be plotted will become less than the number of pixels across the plot.  At this point, gaps would become visible in the plot.

To get around this, ARM then implements a smaller buffered plot mode, where a similar process to the cached plotting takes place, but is only stored in temporary arrays which persist until ARM is shut down or the file is closed.

Similar to the cached plotting, ARM uses a number of cached blocks equal to the number of pixels across the plot. This is very efficient but is completely tied to the selected zoom on the waveform, so any change to this zoom entails re-calculating the blocks.

For this reason, the plots can redraw very quickly as long as the zoom is not changed.  If the zoom changes, the redraw is slower initially as the new cache is created.

## 2.3 Normal plotting

Once the visible zoom drops below the threshold below that used for buffered plotting, normal plotting occurs, where all samples in the selected range are plotted, and are read from the WAV file using buffering.

## 3 User settings to control buffering

ARM provides some user settings that you can change to alter the way it handles buffering.

On the Program options dialog you can specify the WAV file size, in kB, at which buffering begins.

Note that once a WAV file is read by ARM and a PDS is stored, the buffer size setting in Program options becomes irrelevant – once buffered for the PDS, it is always buffered.

If you wish to change that for a particular recording, simply re-open the WAV file directly using the new setup.

There are two further settings which can be set only using a registry editor, as there is no User Interface in ARM for changing them.

The two settings are related to how the plot cache works and directly affect what is stored in the PDS file:

1. MaximumBlockScaledDownPlot
2. PercentOfSamplesScaledDownPlot

MaximumBlockScaledDownPlot defines the maximum number of blocks that are used in ARM to cache the full waveform data.  This is 25,000 by default.  Increasing this number can improve plot performance a lot if you are going to zoom in to analyse a part of the waveform.  The only cost is in the amount of memory ARM is going to use and the size of the PDS file.

As a guide, the default number of blocks will add 200 kB to the file size per channel.

PercentOfSamplesScaledDownPlot is used to control how many blocks are used for the cache, up to the maximum defined by MaximumBlockScaledDownPlot.  The default for this is 5%.

For example, if your waveform has 150 million samples, and the default PercentOfSamplesScaledDownPlot is 5, that would require 7.5 million blocks, but as this is more than MaximumBlockScaledDownPlot, MaximumBlockScaledDownPlot is used instead.

Generally, if you want to zoom in more than about 20x of your recording, you should increase the maximum number of blocks that will be used.

## 3.1  How to change these settings

- Make sure that ARM is not running.

- Ensure that before you make any changes to your system, you back up your registry and set a restore point.

- Use regedit (Start > Search programs and files).

- Go to the following registry branch:

  HKEY_CURRENT_USER\Software\SKF\Analysis and Reporting Manager\Setup

- Edit the values required by finding the following keys and double-clicking on them:

  o  MaximumBlockScaledDownPlot

  o  PercentOfSamplesScaledDownPlot

- Values are saved automatically when you make changes.

- If you want to apply the new settings to any files you already opened, you need to open the original WAV file, not the PDS.

## 4  Applicability

These settings are available in version 2.2 or later of ARM.