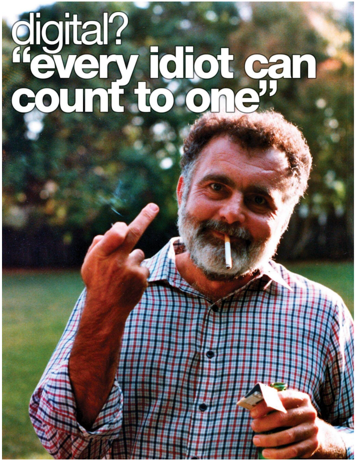
Real Time Audio Programming

Sean Enderby

Birmingham City University

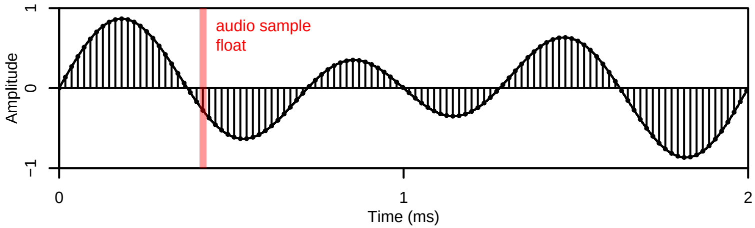
30th November 2016

Digital Audio Terminology



Audio Samples

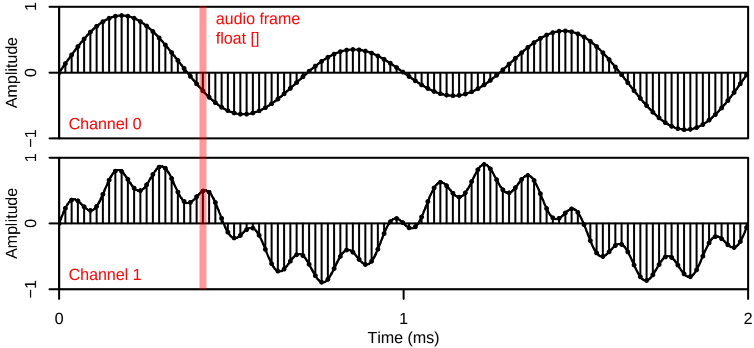
- Digital audio signals are comprised of samples.
- These a stored as floating point values between -1 and 1.



- You know this!
- If you don't know this you are in the wrong place.

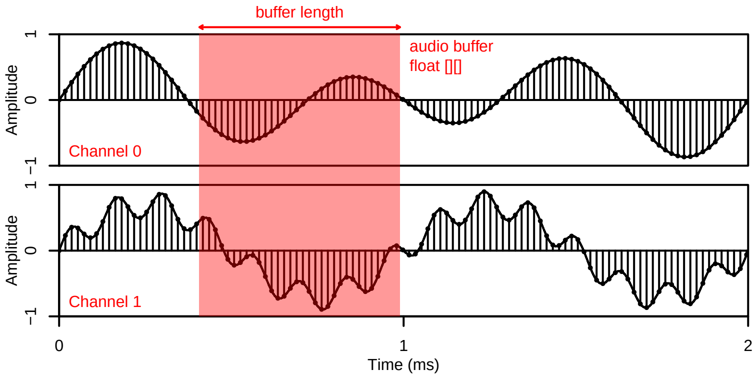
Audio Frames

- An audio frame is an array containing a single sample from each audio channel.



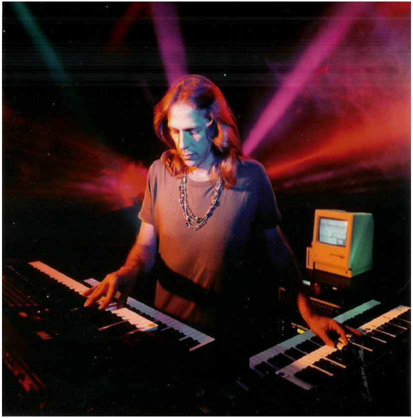
Audio Buffers

- An audio buffer is a 2D array of samples, representing multiple audio frames.



Real Time Audio

How do we get audio out of the computer?



Computer Audio

- Modern operating systems typically include an audio API.
- This defines how audio is passed between hardware and software.
- Our program will be given new audio input and asked to provide new audio output.
- This is done using an audio callback function.

The Audio Callback

- The audio callback is a function which we register with the system in order to receive and send audio information.
- The system calls the audio callback periodically with two important arguments:
  - A buffer containing the frames of the input signal.
  - A buffer to be filled with frames of the output signal.
- Inside the callback function we can do whatever audio processing we deem necessary.

Persistent Data

- It is our responsibility to manage any data we want to be persistent across calls to the audio callback.
- Say we wants to implement an FIR filter:
  - We would need to keep a record of the previous input samples.
  - If these were stored locally in the audio callback function they would get lost when the function returns.
  - We need to declare an array which is external to the audio callback function and store them in there.

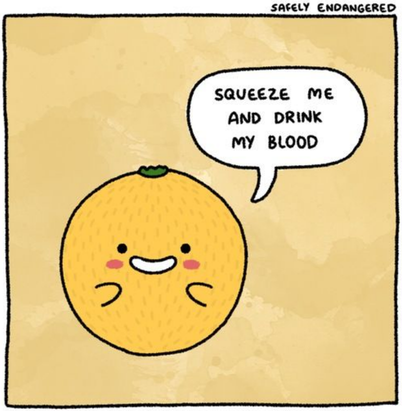
Buffer Length

- To have minimum latency the audio callback should be called with audio buffers one frame in length.
- This introduces considerable computational load, calling the audio callback at the sample rate of the system.
- Increasing the buffer length increases system latency while reducing computational load.

Buffer Underflow

- If our audio callback takes too long to execute the system is going to run out of audio frames to play.
- This is called a buffer underflow.
- The audio output is consuming audio frames faster than we are producing them.
- This leads to clicks and stutters in the output.

JUCE



What is JUCE?

- JUCE is a C++ library for developing applications.
- It is especially focused on providing cross platform audio functionality.
- The APIs of different operating systems all have different ways of registering an audio callback (among other things).
- JUCE allows us to write an audio callback in one way and have it work on a variety of systems.

JUCE Audio Buffers

- In JUCE audio is passed around in AudioBuffer objects.
- Audio data can be accessed using the getReadPointer() and getWritePointer() functions, depending on what type of access you require.
- Our audio callback function will be passed a reference to an AudioBuffer containing the input data.
- The output is then written into the same AudioBuffer.

Thanks For Listening!

Any Questions?

Let's Make Some Dank Sounds!