A Comparative Study of Analogue and Digital Mixing Techniques

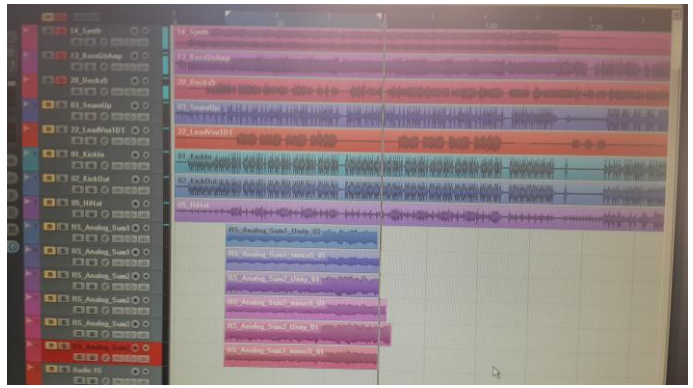
JESSICA KENT

SUPERVISED BY RICHARD FOSS WITH CONSULTATION FROM CORINNE COOPER

Presentation Overview

- Problem Statement
- Literature Review
- DAW Code investigation
  - Summing
- Different Summing Algorithms
- Track Recordings
  - Sine Waves
  - Audio
- Audio Sample Tester
- Listening Tests
  - Configuration
  - Results
- Timing Tests
- Conclusion
- Questions

VS



What's the problem?

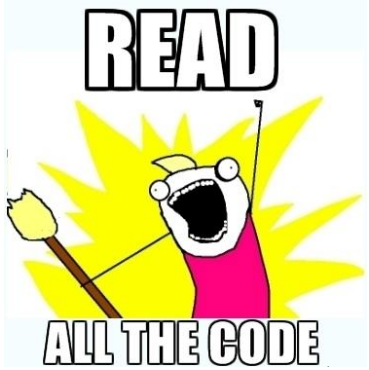
- Song is made up of multiple tracks that have been summed
- Difference between analogue and digital summing is widely debated
- Three objectives:
  1. Does a visual or audible difference exist?
     - Listening and Visual Testing
  2. Can digital mixing algorithm be created to emulate analogue summing?
  3. Create program to facilitate comparison of audio samples
     - Audio Sample Tester

Literature Review

- Investigated different aspects that could affect the digital summing process
  - Sampling Rates, Fletcher-Munson curves and RMS levels
  - How analogue equipment sums the tracks
  - Panning Laws
  - Summing Algorithms
- Previous testing method
  - Summing of five different DAWs tested (Leonard and Buttner-Schnirer, 2012)
    - Differences heard when panning included
- Source code of three open source DAWs

Comparing the code

- Testing from 2012 investigated three specific aspects of each DAW
- Differences in representation and implementation of
  - Gain
  - Panning
  - Summing
- Ardour, Audacity and Rosegarden dealt with all three in similar ways



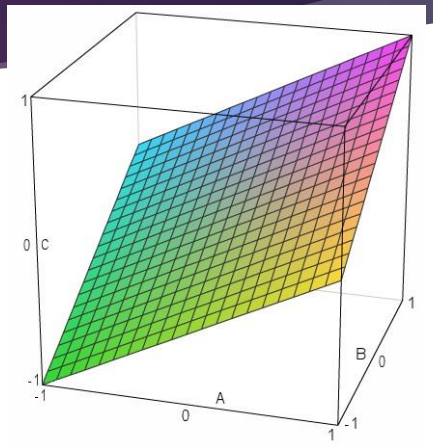
Comparing the Summing

```
void MixBuffers(int numChannels, int *channelFlags, float *gains,
                samplePtr src, samplePtr *dests,
                int len, bool interleaved)
{
        ...
      float gain = gains[c];
      float *dest = (float *)destPtr;
      float *temp = (float *)src;
      for (int j = 0; j < len; j++) {
        *dest += temp[j] * gain;    // the actual mixing process
        dest += skip;
      }
   }
```

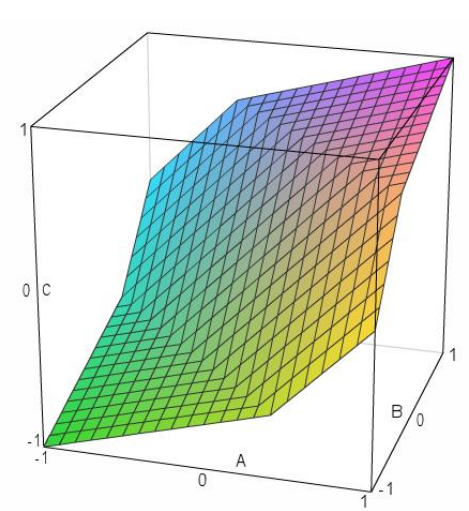
```
void default_mix_buffers_with_gain (ARDOUR::Sample * dst,
                const ARDOUR::Sample * src,
                pframes_t nframes, float gain)
{
        for (pframes_t i = 0; i < nframes; i++) {
                dst[i] += src[i] * gain;
        }
}
void default_mix_buffers_no_gain (ARDOUR::Sample * dst,
                const ARDOUR::Sample * src, pframes_t nframes)
{
        for (pframes_t i=0; i < nframes; i++) {
                dst[i] += src[i];
        }
}
```

```
for (size_t i = 0; i < n; ++i) {
        sample_t v = cached[0][scanFrame + i]
+ cached[1][scanFrame + i];
        destination[0][i + offset] += v;
}
```

Different Summing Algorithms

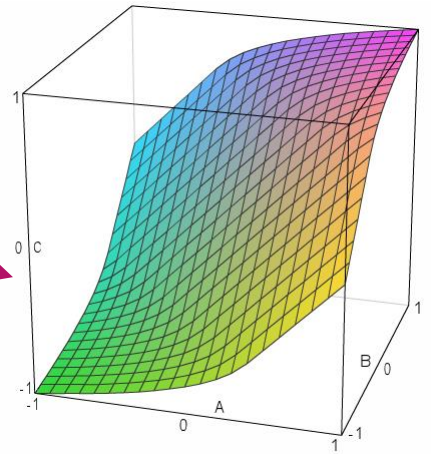


$$sum = \frac{a+b}{2}$$



$$sum = \begin{cases} x & -t \leq x \leq t \\ \frac{x}{|x|} \cdot (t + \frac{1-t}{2-t} \cdot (|x| - t) & |x| > t \end{cases}$$

$$sum = \begin{cases} x & -t \leq x \leq t \\ \frac{x}{|x|} \cdot (t + (1-t) \cdot \frac{ln(1+f_\alpha(t) \cdot \frac{|x|-t}{2-t})}{ln(1+f_\alpha(t))} \cdot & |x| > t \end{cases}$$



Equations and graphs from http://voegler.eu/pub/audio/digital-audio-mixing-and-normalization.html

Recording

- Analogue Desk (Mackie 24·8) and Digital Workstation (Cubase Artist 6.5)
  - Via RME Fireface UFX Converter
- Summed sine waves
  - Different bit depths and sampling rates
  - How high frequency waves affect sounds we can hear

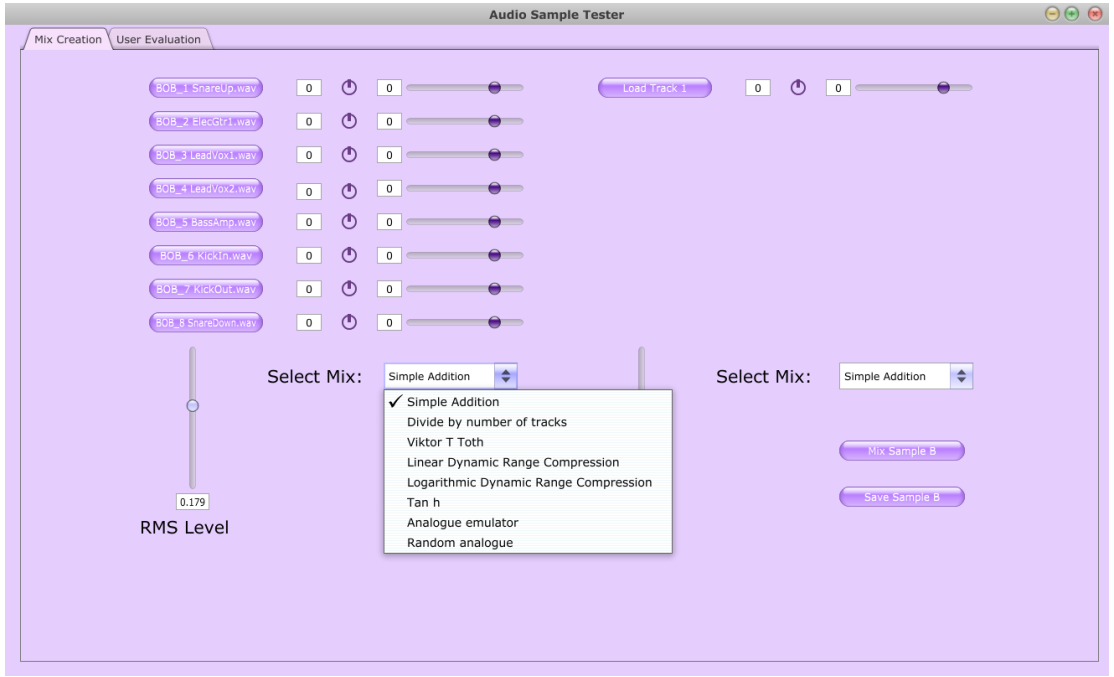
Waves	48kHz & 24-bit	48kHz & 32-bit	96kHz & 24-bit	96kHz & 32-bit
30 kHz alone				
300Hz + 30kHz				
1kHz + 30kHz				
300Hz + 1kHz + 30kHz				
1kHz sin + 20kHz square				

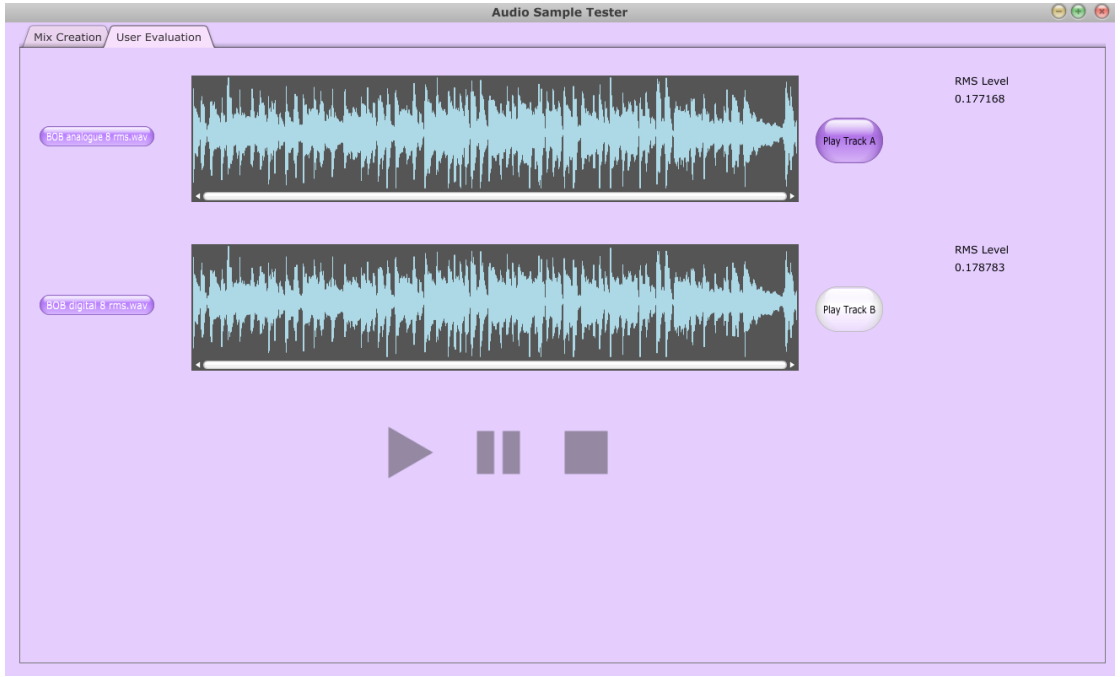
Audio Tracks

- Summed three contemporary songs of different genres
  1. Pop/Rock ("Black Out Betty")
  2. Singer-Songwriter ("Catching Up")
  3. Hip-Hop/Electronic ("Rockshow")
- Eight tracks chosen for each
- Different number of tracks summed

Song	Black Out Betty	Catching Up	Rockshow
1	Snare Up	Acoustic Guitar 2	Synth
2	Electric Guitar 1	Snare Up 1	Bass Amp
3	Lead Vocals 1	Electric Guitar 1	Decks 5
4	Lead Vocals 2	Lead Vocals 1	Snare Up 1
5	Bass Amp	Bass DI	Lead Vocals 1
6	Kick In	Kick In 1	Kick In 1
7	Kick Out	Kick Out 2	Kick Out 1
8	Snare Down	Bass Amp	Hihat

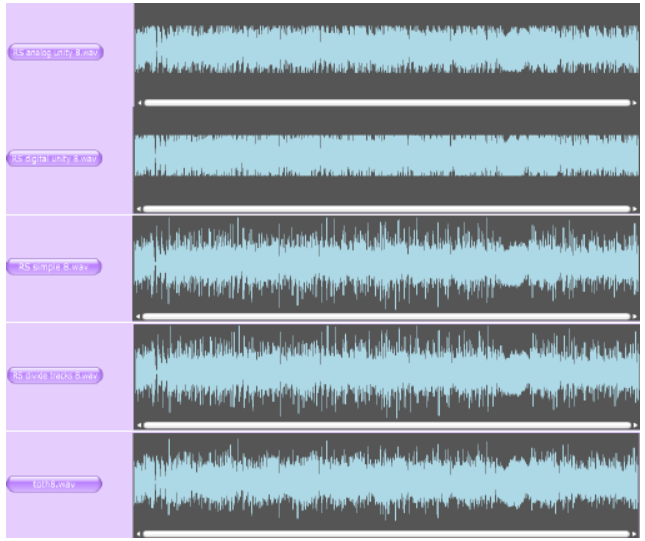
Audio Sample Tester

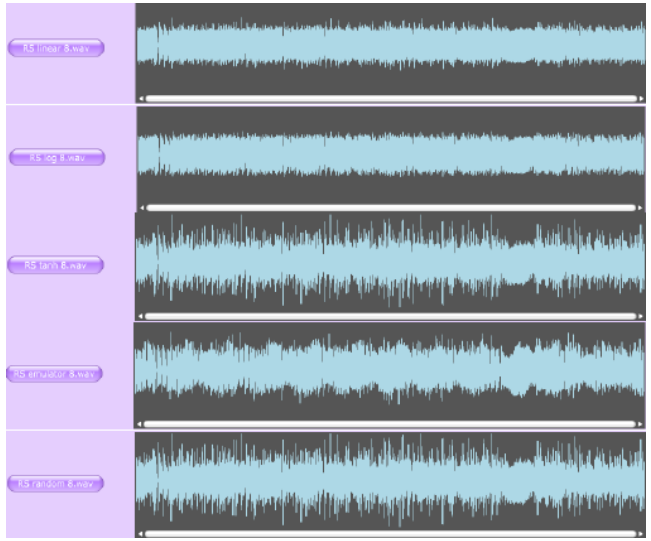




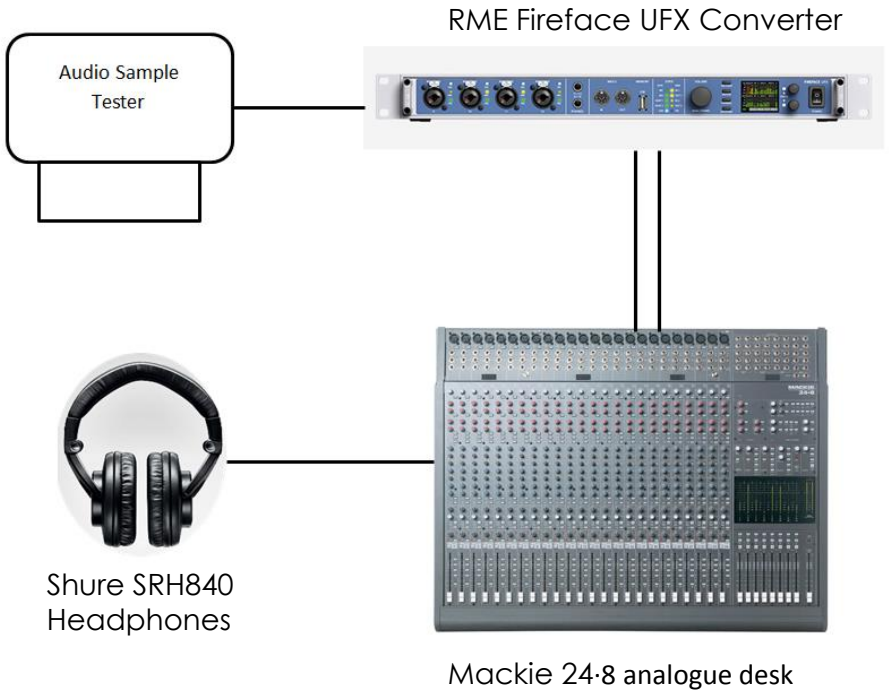
Digital Mixing Algorithms

1. Simple Addition
2. Divide by Number of Tracks
3. Viktor T Toth
4. Linear Dynamic Range Compression
5. Logarithmic Dynamic Range Compression
6. Tanh
7. Analogue Emulator
8. Random addition

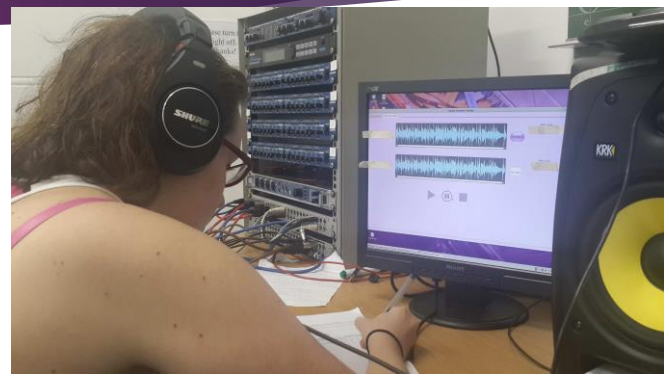




Listening Tests



Sample A



Sample B

Test Configuration

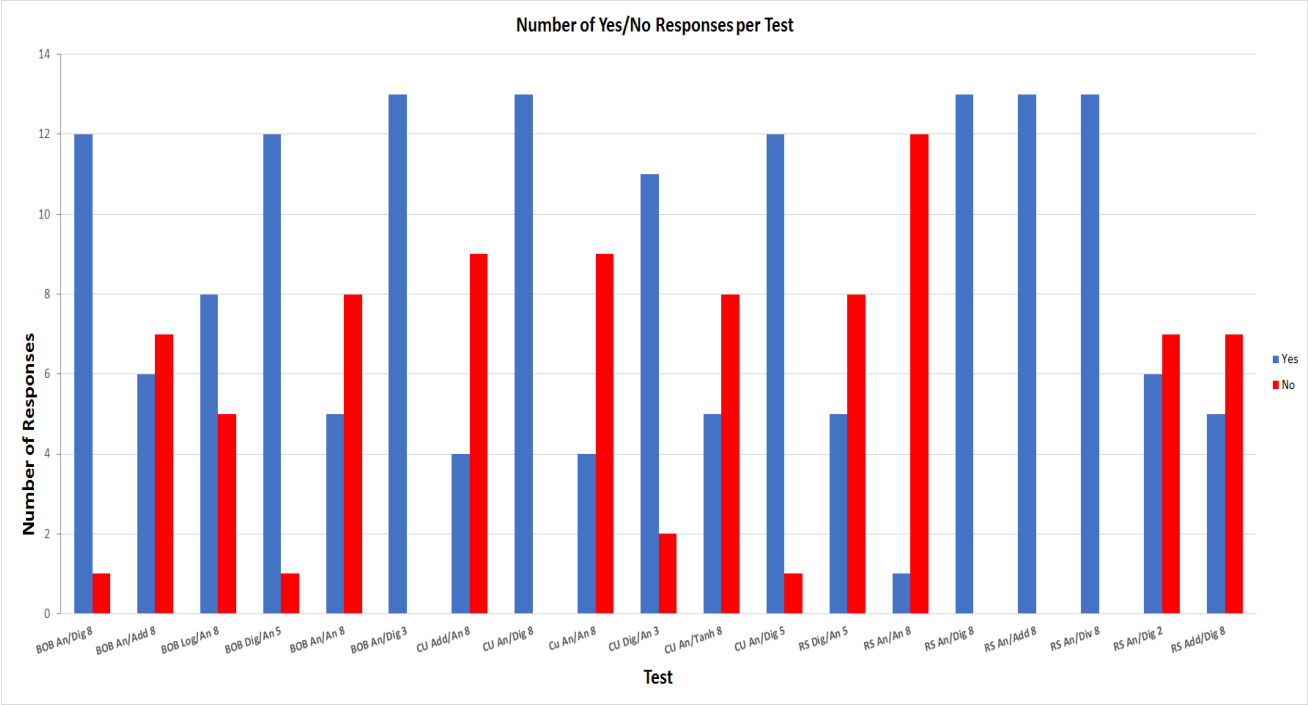
Response sheet questions

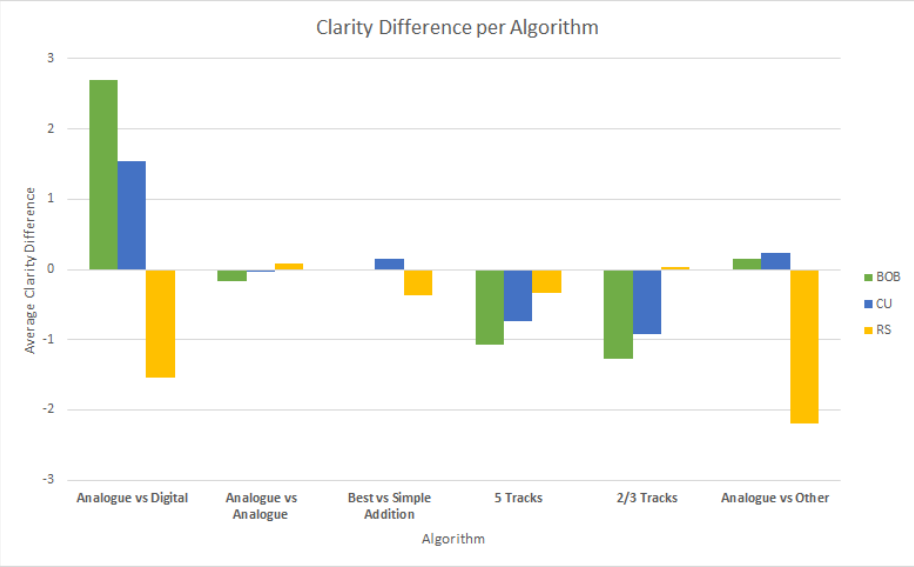
1. Do you hear a difference?
2. Describe the difference
3. Rate clarity of Sample A
4. Rate clarity of Sample B
   - From 1 (clear) to 5 (muffled)

Audio Test Configuration				
Test	Song	Tracks	Sample A	Sample B
1	Black Out Betty	8	Analogue	Digital
2	Black Out Betty	8	Analogue	Simple Addition
3	Black Out Betty	8	Logarithmic	Analogue
4	Black Out Betty	5	Digital	Analogue
5	Black Out Betty	8	Analogue	Analogue
6	Black Out Betty	3	Analogue	Digital
7	Catching Up	8	Simple Addition	Analogue
8	Catching Up	8	Analogue	Digital
9	Catching Up	8	Analogue	Analogue
10	Catching Up	3	Digital	Analogue
11	Catching Up	8	Analogue	Tanh
12	Catching Up	5	Analogue	Digital
13	RockShow	5	Digital	Analogue
14	RockShow	8	Analogue	Analogue
15	RockShow	8	Analogue	Digital
16	RockShow	8	Analogue	Simple Addition
17	RockShow	8	Analogue	Divide by Tracks
18	RockShow	2	Analogue	Digital
19	RockShow	8	Simple Addition	Digital

Listening Test Results

Tracks	Test	Yes
8	Analogue vs Analogue	25.64%
8	Analogue vs Digital	97.44%
5	Analogue vs Digital	74.36%
2 or 3	Analogue vs Digital	76.92%
8	Analogue vs Other	66.67%
8	Best vs Simple Addition	39.47%





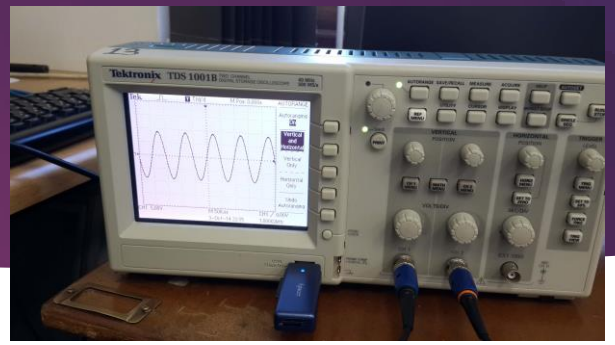
Timing Results

- Can digital algorithms be implemented in real-time?
- Time limit (based on sampling rates)

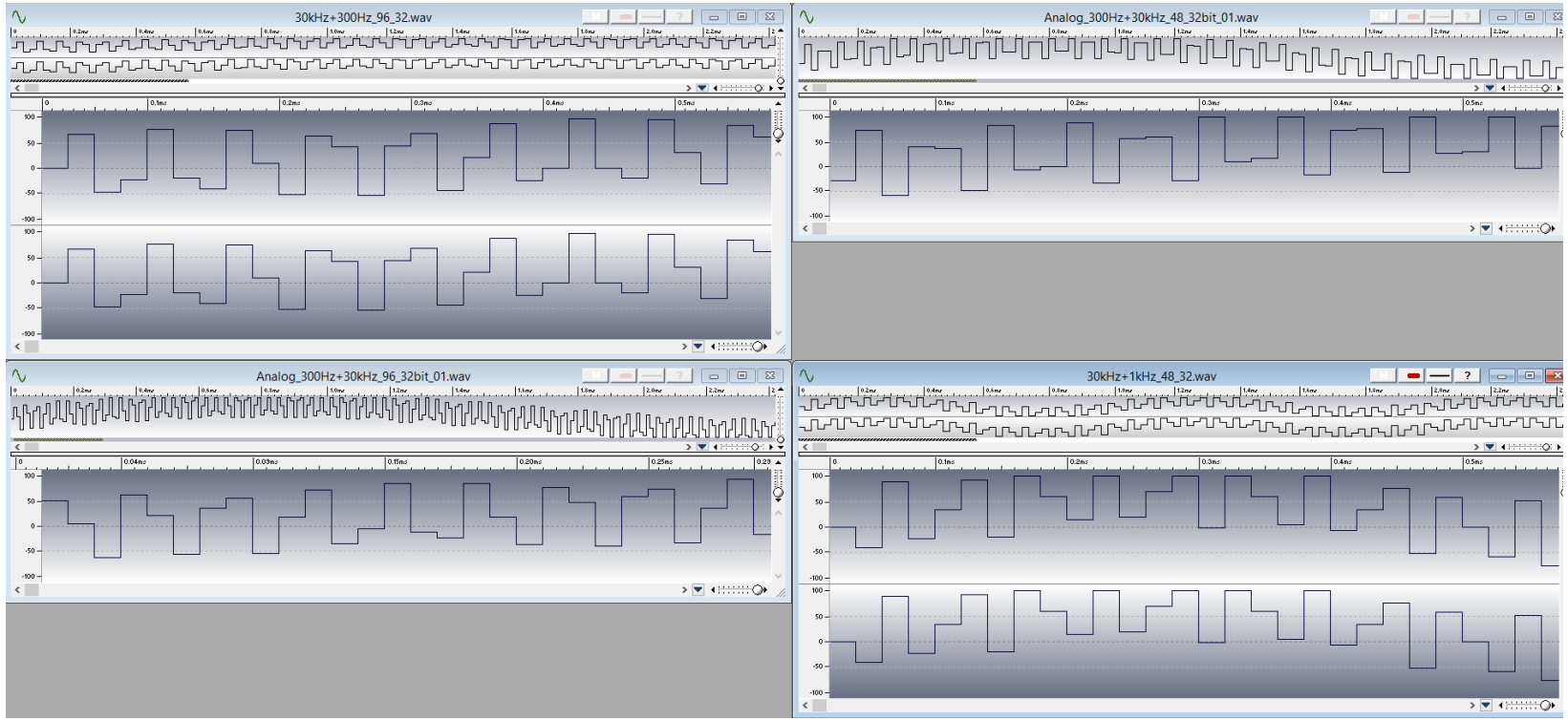
Sample Rate (kHz)	Maximum Time (μs)
44.1	22.68
48	20.83
72	13.89
96	10.42

Time Taken to Sum Eight Samples (in microseconds)				
Algorithm	**Addition**	**Divide Tracks**	**Viktor T Toth**	**Linear Compression**
BOB	0.640	0.645	1.679	0.648
CU	0.914	0.670	1.689	0.668
RS	1.966	1.405	3.622	1.407
Average	1.173	0.907	2.330	0.908
Algorithm	**Random**	**Tan h**	**Emulator**	**Log Compression**
BOB	0.903	1.154	1.037	0.665
CU	0.928	1.171	1.053	0.676
RS	1.961	2.522	2.242	1.438
Average	1.264	1.616	1.444	0.926

Not enough time



- Analogue experiments
- Visual testing of sine waves



Conclusions

1. Definite difference between analogue and digital summing
   - Depends on genre
   - Number of tracks makes a difference
2. Simple Addition algorithm closest to analogue summing

Further work needed:

- Investigate influence of panning
- Real-time digital mix creator

QUESTIONS???