Sintefex Audio

Technical Documentation

Dynamic Convolution

Previously published under the title

Analysis and Simulation of Non-Linear Audio Processes using Finite Impulse Responses Derived at Multiple Impulse Amplitudes

*at the AES 106th Convention, 1999,Munich Germany, preprint # 4919(J5)**

Michael J Kemp

Sintefex Audio Lda, S. Marcos da Serra, P-8375 Portugal
Tel: (+351) 282 361748, Fax: (+351) 282 361749
E-mail: mikekemp@sintefex.com
http://www.sintefex.com

Abstract

A method of analysing non-linear audio effects is described in which a sequence of differing level impulses is fed to a device under test. The resulting impulse responses are recorded and applied by dynamic convolution to input signals to simulate the sampled effect. In this way effective simulation of effects such as valve amplification and analogue recording may be achieved.

**With minor textual amendments 19/1/2000*

0 Introduction

It is well known that various items of audio equipment are preferred for processing audio not for their perfection but for their imperfections. This has lead (to give only two examples) to the re-emergence of valve amplifiers and the continued preference amongst some engineers for the use of analogue tape recording. We have even seen the emergence of new analogue formats such as 1" stereo recorders which attempt to exploit the desirable characteristics of analogue recording and ameliorate the less desirable aspects such as noise and drop-out. [1]

So what are the desirable characteristics of these devices that lead them to be preferred even thought they are operationally frequently inferior to modern digital alternatives both in pure accuracy and reliability and repeatability? It seems clear that there is often an element of subtle frequency and phase response variation that may not easily be created in a typical equaliser (consider for example the extraordinary low frequency response of an analogue tape recorder). In addition these items of equipment exhibit noticeably non-linear behaviour resulting in distortions to the signal which can rise to several percent but still result in a tonally acceptable or even preferred overall behaviour. Some of these distortions may mimic the psycho-acoustic performance of the ear at high levels resulting in distortions that 'sound loud'.

Attempts have been made to model the behaviour of valves derived from the physical functionality of the device [3,4,5,6] and conceivably these could be developed into real time simulations exhibiting characteristic behaviour of these devices. We have chosen on the other hand, rather than to model the functionality of each such desirable device, to adopt an impulse response based system for analysis and simulation.

1 Impulse Response Assessment at Multiple Levels

It is clear that to simulate one of these non-linear audio processes we need to simulate (i) the frequency and phase response (including absolute gain) at any particular signal level and (ii) to track the change in these characteristics with variation in signal level to characterise the non-linearity of the process.

It is well known that a linear time invariant signal process may be totally characterised by its impulse response and that a convolution of its impulse response with an input stream provides a perfect simulation. When transferred from the analogue to the digital domain and where the duration of the impulse response is limited the simulation is as close as the sample rate, precision and duration of the response allow. (See for example [2]). At any particular signal level an impulse may be applied to the system to be analysed and the resulting impulse response recorded. By convolution of this impulse response with a new signal stream the linear characteristics may be reproduced to arbitrary accuracy limited only by the length of the convolution and the accuracy to which the impulse response could be determined given the presence of noise in the signal process under analysis. In many cases it is possible to achieve a sufficiently accurate simulation meeting requirement (i) above.

In order to determine the non linear characteristic (ii) of a signal process we now apply a sequence of impulses at different levels to determine the way the characteristic of the system under analysis changes with amplitude. We choose to perform a number of impulse tests at linearly spaced impulse amplitudes between the maximum permissible and (non-inclusively) zero. The resulting impulse responses are in all cases normalised in level by simple multiplication in inverse proportion to the amplitude of the test impulse.

In fact we have chosen to use a step impulse signal to determine impulse response as the step has significantly more energy in the lower frequency than the ‘unit’ impulse. Figure 1 shows an illustration of a typical analysis signal to be used. Here ‘fs’ refers to full scale digital value (in fact the digital values range from –fs to fs-1 in a typical implementation). This signal is appropriate to a device in which the excitation from the impulse has sufficiently decayed after 4000 sample periods that it is acceptable to terminate recording of the impulse response at this time and move onto the next impulse response sample. In situations where this is not the case it will be necessary to extend the horizontal time scale of the test signal (we have adopted a principle that the interval between impulses should always be a multiple of 1000 in order to simplify data extraction and to recognise when any sample rate error has occurred). We used a sample rate of 44.1kHz in our experiments but sample rate can best be selected according to the effect to be sampled and the required rate at the time of simulation.

First it is decided how many impulse levels we require to take. For ease of arithmetic and to reflect that by the time an input signal is more than 40dB below the peak level most devices have entered a linear region of operation, 128 linearly spaced impulse levels were chosen. Accordingly the step change in impulse response d is set to fs/128.

From the rest state the signal is taken negative to –1/2 fs and held for 8000 samples. A positive transition of fs then takes place which is the first excitation pulse of a device or process to be analysed. The signal is held for 4000 samples during which the impulse response may be recorded. This signal is then taken negative again to –(fs – d) / 2. After a further 4000 samples to allow the system under test to stabilise, a further positive transition of (fs – d) is performed. The process is repeated with positive steps of (fs –2d), (fs –3d) etc until the test impulse signal reaches zero.

The above signal is generated (or may be recorded and applied remotely) to an audio system under test and the resultant output of the system is recorded for analysis. If it is desired to later simulate the audio system at signal levels that drive it into significant distortion then the level applied to the device under test must be sufficient to stress the device at least as much as any desired simulation. If it is considered that the low frequency content of this signal is excessive for a particular device then the test signal may first be high pass filtered but of course the resultant simulations will always include the effect of this high pass filter.

From the above process a set of 128 step impulse responses is obtained. These are converted to 'unit' impulse response by first order differing of the derived signal,

$y_n = x_n - x_{n-1}$.

Care must be taken to ensure that all arithmetic performed now preserves the full range of digital values and precision. All arithmetic in our experiments was performed in double precision floating point.

In addition it is necessary to scale up impulse responses in a process we term normalisation, thus the m^{th} response taken at a step amplitude of

$(fs - m.d)$

must be scaled by the factor

$fs / (fs - m.d)$.

At this stage it is possible to do further processing as will be described later. Once all processing is complete a final window function is applied to remove smoothly any tail of impulse response which may still exist at the end of the sampled period. We have used a linear ramp in the last quarter of the sampling period but there is ample room for additional experimentation in this area.

For the first experiments a digital audio workstation was employed and the test signal computed and loaded as a wave file. The available equipment was not accurately calibrated as to absolute level but a maximum drive estimated at +18dBu for a full scale sine wave was delivered to a valve equaliser. Several measurements were taken with different control settings, and figures 2 and 3 show the first results of this impulse testing procedure, with 80 samples displayed horizontally. The graphs show 16 of the 128 impulses taken spread linearly across the range. The vertical axis is linear and the number of samples of the response is shown against each figure. Each graph is presented with the impulse response from the lowest level impulse at the back of the graph and that from the highest level impulse in the foreground.

Figure 2 shows the derived impulse responses when the controls were set to a nominal flat position on the equaliser. Clearly it is not possible to determine visually any variation as no gross distortion was taking place at the available drive level.

Figure 3 shows the derived impulse responses when maximum 1kHz boost was set on the controls. There is now some visual evidence of a variation in response as the frequency boosted signal would be at a high level within the unit under test.

For comparison we also sampled a guitar amplifier and speaker setup and the result is shown in figure 4. The horizontal scale is now set to 400 samples as there is significantly more energy stored in the system. There is evidence that this amplification set-up is only moving into linearity in the back two or three impulse curves, i.e. at perhaps 20dB below the peak level we applied. We also note a characteristic we have observed with other devices that run into gross distortion and that is an apparent simplification of the impulse response at the higher levels.

It was now possible to move onto the simulation phase of the experiment.

2 The Dynamic Convolution Algorithm

The standard convolution operation of an input stream x with an impulse response h of length L to give output stream y from a linear system is defined by

$$y(n) = \sum_{k=0}^{L-1} x(n-k).h(k). \qquad (1)$$

This convolution provides a complete simulation of a linear time invariant system. In a non-linear case we make an assumption that the non-linear process can be modelled by a linear summation of responses to impulses that are appropriate to the magnitude of each individual input sample. Clearly this assumption will be violated where a system markedly changes its response to subsequent input after any particular input. However experiments indicate that in the typical situations this assumption holds sufficiently to achieve a useful result. A special case exists for example in the case of a dynamics compressor circuit which is designed to alter its response in this way and this will be discussed later.

We have discussed how to obtain a set of impulse responses at different impulse amplitudes for the case where 128 equally spaced impulse amplitudes are used. In general we may have a set of M equally spaced impulse responses $h_1, \ldots h_M$ where h_M is the response to the peak impulse and h_1 is the response to the lowest impulse. Note that these are all normalised to the peak value as discussed in section 1.

To determine which impulse response to use for a given input sample we define a selector function S(x(n)) which selects an impulse response on the basis of the instantaneous input sample value x(n). The S(x(n)) is defined as

$$S(x(n)) = 1 + \{|x(n)| / (fs / M)\},$$

where the braces {} are used to indicate the integral part of the value. There is in fact a slight modification to this in that if the maximum negative sample value –fs is encountered, the absolute value |x(n)| is limited to fs –1 to match the maximum positive value. The assumption is also made that fs is exactly divisible by M for simplicity.

We now define the dynamic convolution algorithm as follows:

$$y(n) = \sum_{k=0}^{L-1} x(n-k).h_{S(x(n-k))}(k). \qquad (2)$$

In the case where M = fs we have a separate impulse response for every possible sample magnitude and we can stop here in our simulation. However, where M < fs it can be seen that as input sample magnitudes |x(n)| increase steadily from zero, response h1 is first used, then when the sample value reaches fs / M we have a discontinuous switch to response h2. This occurs repeatedly as |x(n)| grows.

This is handled gracefully by determining an interpolation between adjacent impulse responses. Clearly an arbitrary complex interpolator could be designed but we have chosen a simple linear interpolation between the two impulse responses that represent the responses to impulses located nearest in value to the input sample.

Thus an interpolator fraction p(x(n)) is defined as follows

p(x(n)) = (|x(n)| modulo (fs / M)) / (fs / M).

Now we can define our desired simulation equation as

$$y(n) = \sum_{k=0}^{L-1} x(n-k).(p(x(n)).h_{S(x(n-k))}(k) + (1 - p(x(n))).h_{(S(x(n-k))-1)}(k)). \qquad (3)$$

We only need to define a new response h_0 as equal to h_1 to provide for the selection of h_0 by the modified selector term (S(x(n))-1). This means we do not have to make a special case of low level input samples and is also convenient for the practical implementation

On first impression it would appear that implementation of this algorithm would be a lot slower than a simple convolution but as will be described later this can be implemented to run at half the speed of a simple multiply accumulate convolution operation (with a small overhead for selection), which since it needs two multiplies and two accumulates per step means that we can achieve close to 100% efficiency from a DSP chip.

Two areas are the subject of continued development work. The first regards the selection function S(x(n)). This can easily be generalised as S(x,n). In other words instead of being solely dependent on the value of an individual sample, the selector function can consider the whole input stream x (or at least its history) for each n. A typical example would be that the selection procedure can be generalised to depend on a neighbourhood of sample values. We have done some tests using an envelope of the input signal as the selection criterion and this represents a field for further experiment

The second area of further development recognises that impulse response selection described so far is dependent solely on the magnitude of incoming audio samples. It is however possible to use impulse response samples taken with positive going and negative going impulses and to use the sign of the incoming audio samples to select the appropriate response as well as the amplitude.

3 Test Implementation and Auditioning

Initial implementation of the desired simulation algorithm took place on a Pentium PC at 90MHz and it was not unusual for a 30 seconds sound sample to take several hours to process using impulse lengths L of up to 1000 samples. A sample screen from this program, FIRST (for Finite Impulse Response Synthesis Technology), is shown in figure 5. At the top left is an entire impulse train, and at the bottom left is a derived set of impulses. At the right are some sample audio signals to be processed.

With the arrival of faster PCs it was decided to try a real-time simulation on a 300MHz Pentium-II. A sample program SFXDEM was devised which would replay audio pre-recorded on the hard disk through the algorithm in real time. It was found that with a sample rate of 44.1kHz, impulse lengths of about L=150 were possible in mono; half this was possible in stereo. We suspect that without limitations of the operating system this processor should achieve several times this. A sample screen is shown in figure 6 which shows that the controls are intended to represent the eventual hardware implementation.

Optimisation of the algorithm (equation 3) was achieved by realising that for each input sample it is only necessary to determine once the appropriate pair of impulse responses required. A literal implementation of equation 3 would be to run the main convolution loop once per output sample, working through the history of input samples and selecting appropriate impulse responses for each input sample. However it is possible to do all the processing for a given input sample in one go after determining the selector values to use.

Figure 7 shows a suitable re-arrangement of the simple convolution operation to illustrate the process. The output stream is shown at the bottom as it is generated. The memory buffer is filled with zeroes at the start (to the right) and final output values are left in the buffer on the left. An 8 tap impulse response is shown h(0) to h(7).

When input sample x(n) arrives, it is used as a multiplier for each element of h, and the multiplier is then accumulated with the partial sums being generated in the output buffer. The element below h(0) will become the output sample y(n) when the accumulation is done. The impulse response h is then moved one to the right ready for the arrival of x(n+1). Clearly once y(n) is calculated it may be removed from the buffer and output. If zero is written back the buffer can be made circular.

To implement equation 3, it is necessary to examine the input sample x(n) and determine the two impulse responses to be applied. These are selected by means of pointers into memory. The interpolation factor may then be calculated and x(n) split into two value s_1 and s_2 which sum to y(n) and are in the correct proportion for the implementation. The partial sums may then be generated with two multiply accumulates for each element of the length of the impulse responses.

By this method we achieved a usable demonstration on a PC. The demonstration program was used for simulating three principle sound processors. These were (i) the valve equaliser with controls flat as referred to in figure 1, (ii) a quarter inch tape machine running at 15ips, and (iii) a heavily distorted guitar amplifier and speaker set-up. Sound sources were a selection of pre-recorded material from complete mixes to individual instruments. Also clean electric guitar recordings were used to evaluate the guitar amplifier effects.

The listening tests were informal, and comprised the author who has considerable studio experience and a few selected recording engineers and producers. In each case there was clear indication that the expected characteristics of the simulated effects were produced in the simulation. The real-time test was found to be invaluable in that the signal to be processed could be varied in level to assess the degree of non-linear effect applied in the simulation process. For example the tape recorder simulation clearly showed an increase in expected tape compression effects as the level into the simulation was increased.

It was quickly realised that a longer impulse response was desired than could be achieved with the real time simulation. Some non real-time simulations were compared using response lengths of 400 taps and more and it was felt that these longer periods were desirable.

Detailed subjective and objective results await the completion of the first real time practical implementation discussed in section 6.

4 Further Processing of the Impulse Responses

Noise in the impulse responses results in distortion in the simulation. Since the lowest level impulse response is derived at about 40dB below peak level consideration has to be given to noise removal techniques.

The use of a step impulse and subsequent first order differencing minimises low frequency noise. but higher frequency noise can be present.

One noise reduction strategy is to determine the noise floor by analysis of a quiet section of sampled output from the device under test. It is then possible to start with a relatively high level impulse response and examine the response to determine when the impulse response data is near the noise floor. Where this is the case matching sections from the next higher level signal can be substituted (with appropriate gradual cross fading at either end of the ‘splice’). The rationale is that the significant parts of the impulse response are at higher levels and those signals near the noise floor contribute little to the characteristic of the sound. The process is repeated down to the lowest level impulse.

A further technique is to take multiple samples at the lower levels. Each doubling of the number of samples at a given level and subsequent averaging reduces random noise by 3dB. Since the sampling process is not likely to be synchronous with non-random noise the same argument may be applied to repetitious noise. In this way an arbitrary noise performance may be achieved at the expense of prolonging the sampling process.

Where the sample has already been taken and it is not possible to make multiple tests it is possible to average together a number of different level impulses with suitable scaling to improve the noise floor. This results in reducing the significance of the lower level impulses resulting in some linearising of the simulation, but the comments of the following paragraphs also apply here.

Finally in respect of noise, it is not always necessary to use the lowest level impulse responses. Many devices (particularly higher performance professional devices) are essentially linear by the time signals are reduced to say 12dB below peak. Lower level impulse data may be discarded and the impulse response for this level may be replicated for all lower signals.

Other more creative processing may also be applied to the derived impulse response set. Interpolation may be used to pick out a more linear part of a devices characteristic, and with care, extrapolation can be attempted to experiment with more dramatic effects.

A very useful effect is to sample rate convert the impulse response set. This allows characteristic resonances in the impulse response to be tuned to a particular tone that may appear in an audio stream.

5 Multidimensional simulation

We have so far concentrated on sampling a particular signal process at a fixed setting. The simulation then utilises one dimension of impulse response selection, that of input signal amplitude.

It s possible to take multiple impulse response sets at different settings of the device under test, and to use these as a second dimension of control. Bilinear interpolation is now used to select in one dimension on amplitude and on another dimension by any desirable method. One example is a user controllable parameter to vary the effect, another is an oscillator controlled parameter to synthesise rhythmic variation. A further useful parameter may be gain reduction in a compressor simulation as will be described later.

6 Practical Implementation

To produce a practical embodiment of the analyser simulator, several criteria were enumerated:

1. Impulse response simulation should be capable of running 2048 taps per channel in real time at up to 50kHz sampling rate.
2. Operation at up to 100kHz sampling rate should be supported (albeit with loss of impulse duration) for demanding applications where high precision in the early impulse data can be preserved.
3. 24-bit internal precision should be maintained.
4. Analogue conversion accuracy of at least 20-bit should be built in for accurate impulse response sampling and audio processing.
5. The sample test signal should be available up to at least +30dBu to facilitate stress testing of professional audio devices.
6. The sample test signal should also be available in an electrically floating format at lower level for injection into sensitive unbalanced apparatus with minimum operational difficulty.
7. Some additional signal processing should be provided, especially easily accessible equalisation and compression.
8. Multi-channel operation should be supported to facilitate use in surround sound signal processing.
9. Support for multi-dimensional parameter control with reduced impulse length should be included.

Consideration of the algorithmic implementation of figure 7 and evaluation of several floating point DSPs lead us to choose a 60MHz SHARC device [7]. For the body of the iterative loop the following steps are required for each impulse response tap:

1. Fetch of partial sum
2. Fetch of impulse response element A
3. Fetch of impulse response element B
4. Multiply and accumulate with element A
5. Multiply and accumulate with element B
6. Write of partial sum

Since the SHARC permits 2 memory accesses and a MAC operation per clock, this iteration can be achieved in 2 clock cycles. A 60 MHz processor can thus perform 512 steps in approximately 17 microseconds, permitting operation at 50kHz with a 15% safety margin to allow for loop set-up. Since the algorithm readily lends itself to being split across multiple processors (provided the partial output sums are passed onto the next processor with each new input sample) it was decided to cascade 2 rows of 4 SHARCS to give stereo operation at 2048 impulse steps.

The two channel blocks are coordinated by a further SHARC to separate the stereo samples and pre-calculate the pointer and interpolation parameters, as well as to provide equalisation, compression and other effects.

Finally up to 4 stereo sections can be accommodated within the chassis to allow multi-channel and surround operation. The combined unit is manufactured by Sintefex Audio Lda under the name the Sintefex FX8000 Audio Effects Replicator.

7 Further applications

Mention was made in section 1 of the need for an effects process under test not to change its behaviour following the arrival of an earlier impulse. This may occur undesirably, say for example when a device latches up (even temporarily when an internal audio signal hits a power rail), or deliberately in the case of a dynamics control system (for example an audio compressor/limiter).

In the case of undesirable effects, provided the period between sample impulse is large enough for a complete recovery to be made a useful set of impulse responses may be obtained. The simulation will encapsulate the desirable elements of the device sampled but will not demonstrate the latch-up type of problem.

In the case of a compressor device it is necessary to modify the impulse response measurement to obtain an exact gain match for each impulse. Slow attacks and decay should be set and it is then possible in an interactive test set-up for a conditioning signal at a fixed level to be output between each impulse. When the gain of the compressor reaches its determined value an impulse sample can be taken.

In this way it is possible to determine the non-linear characteristics of the device at a range of attenuations. Further signal tests may be performed to determine the law of the compressor and a simulation may be achieved which, by use of the knee characteristics of the sampled compressor as well as the appropriate non-linear simulation for the degree of attenuation, closely characterises the sound of the sampled compressor.

A further application relates to the non-linearity of audio monitoring environments. It is well known that. particularly at high levels, amplifier and loudspeaker combinations exhibit growing non-linearity. In fact it is often these non-linearities that result in the sound 'sounding' loud. It has been suggested that part of the information conveyed with a master recording should relate to the monitoring conditions under which it was approved by the producer. This enables the recreation of the original experience in the ultimate listening environment. It would clearly be useful in recreating the original effect if the non-linear response of the monitoring system were also recorded, and the sequence of impulse responses described here clearly encapsulates this information if the samples are taken with microphones at the optimum monitoring position.

8 Conclusions

The continuing reduction in the price of processors clearly brings into the practical domain the routine use of substantial processing power to achieve simulation of previously common analogue processes. As the original equipment becomes scarcer and less able to respond to the faster turn-round times of modern recording, and as the library of useful sampled effects grows we feel that the use of these simulation techniques will become increasingly popular.

Finally it should be mentioned that aspects of this work are the subject of various patent applications attributed to the author. We also acknowledge any trademarks used herein.

10 References

[1] Interview with Tim de Paravicini, George Shilling (1998) Studio Sound, Vol 40 No. 12 pp 57-58.
[2] Proakis J.G and Manolakis D.G. (1996) Digital Signal Processing, Third Edition, Simon and Schuster, New Jersey 1996.
[3] W, Marshall Lean, Jnr, (1995). SPICE Model for Vacuum-Tube Amplifiers, JAES Volume 43 No. 3, pp117 – 126.
[4] Eric Pritchard, (1997). Comments on [3] and authors reply, JAES Vol 45 No. 6, pp488,489.
[5] Frederic Broyde, (1997). Comments on [3], JAES Vol 45 No. 6, pp490,491.
[6] Charles Rydel, (1997). Comments on [3] and authors reply, JAES Vol 45 No. 6, pp491-496.
[7] Analogue Devices ADSP-21065L SHARC User's Manual (1998), Analogue Devices Inc.

Sintefex Audio

11 Figures

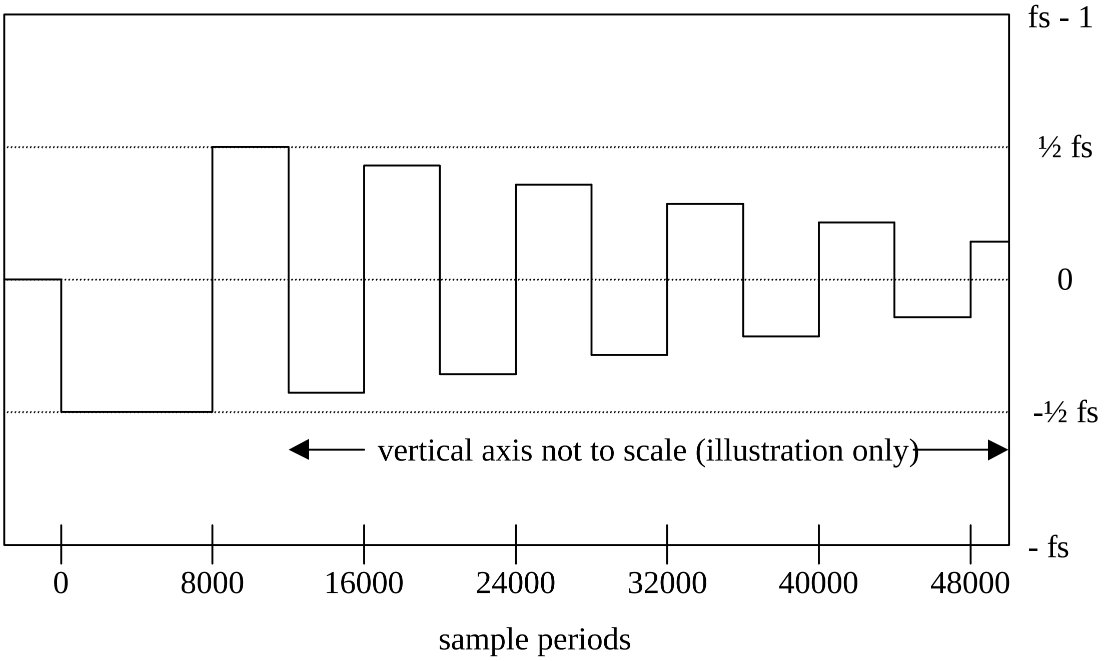

Figure 1: Schematic representation of impulse train.

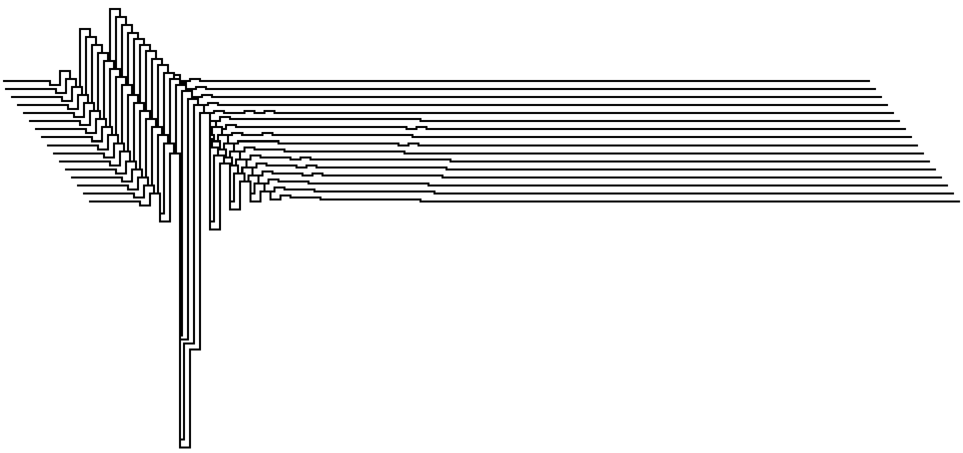

Figure 2: Typical valve equaliser, all controls flat, 80 samples. Lower level response at rear shows no significant variation to higher level at front of graph.

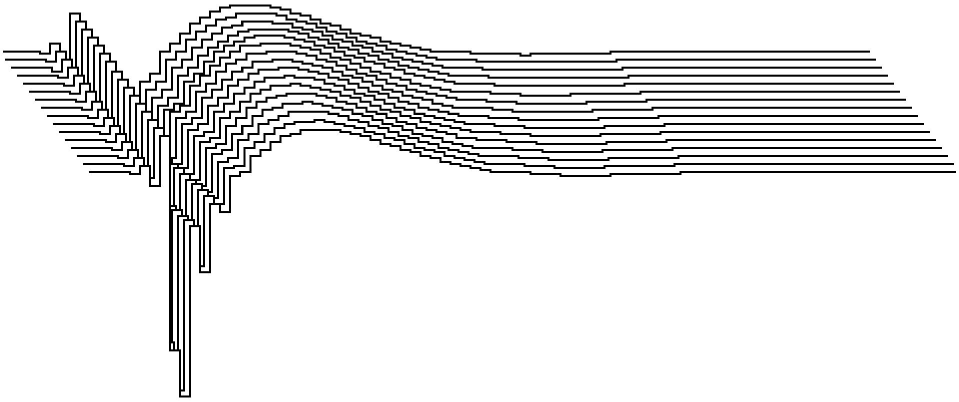

Figure 3: Same equaliser, max 1kHz boost, 80 samples. Lower level response at rear shows visible variation from higher level response at front of graph.

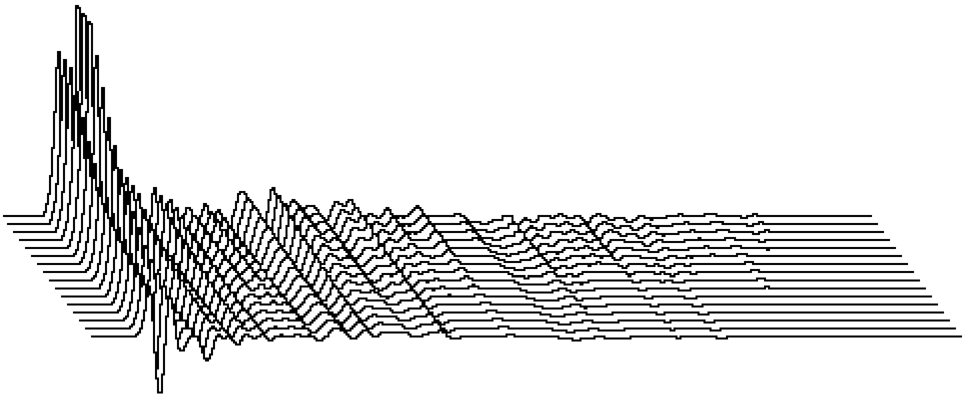

Figure 4: 400 sample plot of guitar amplifier/speaker combination driven into heavy overload. The lower level response is at the rear.

Sintefex Audio

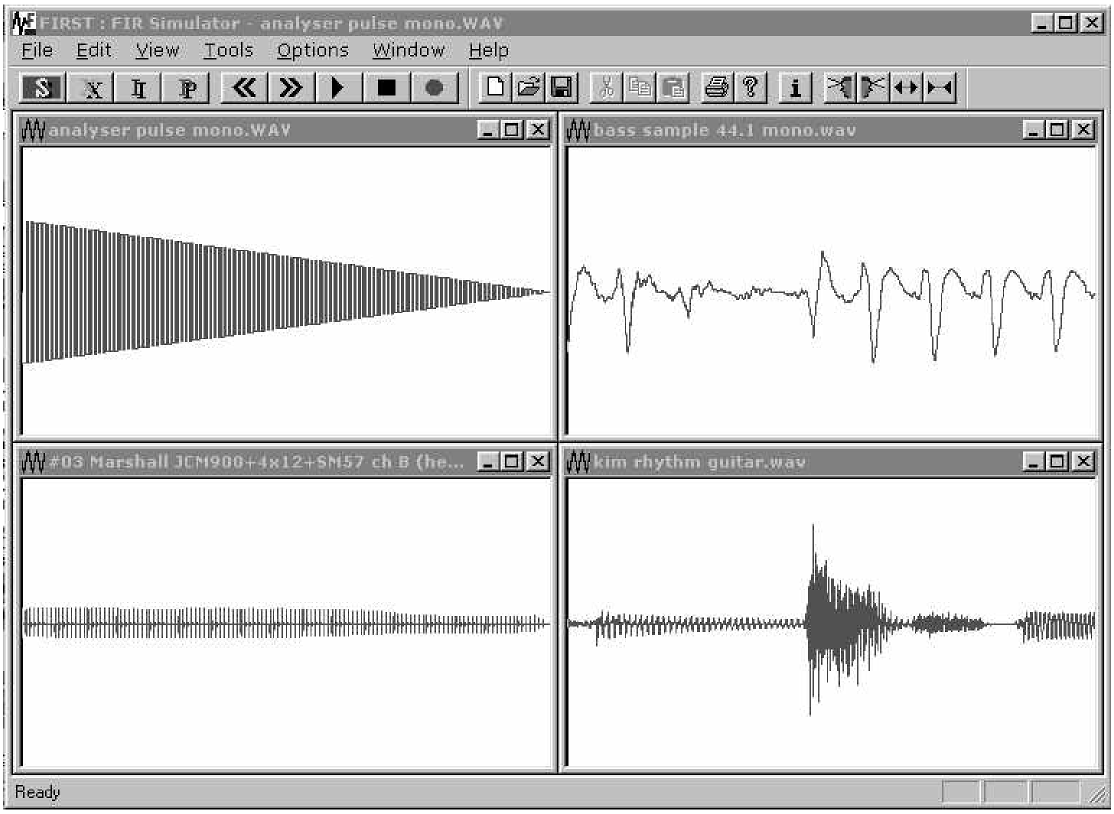

Figure 5: Example display from FIRST non-real time simulator program.

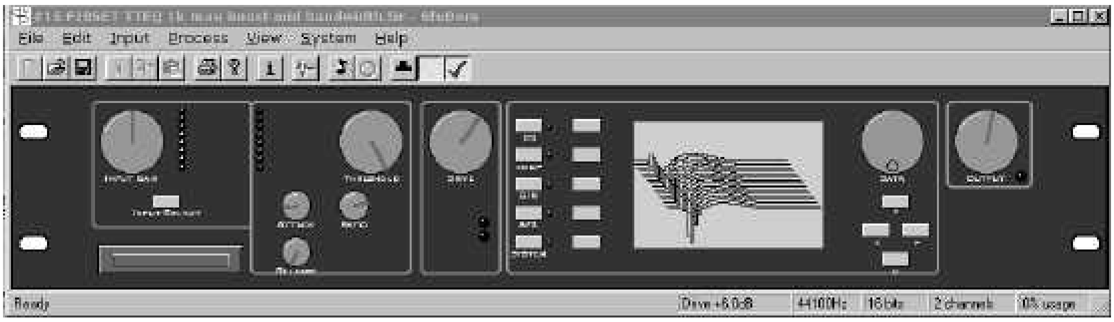

Figure 6: Example display from SFXDEM real time simulator program.

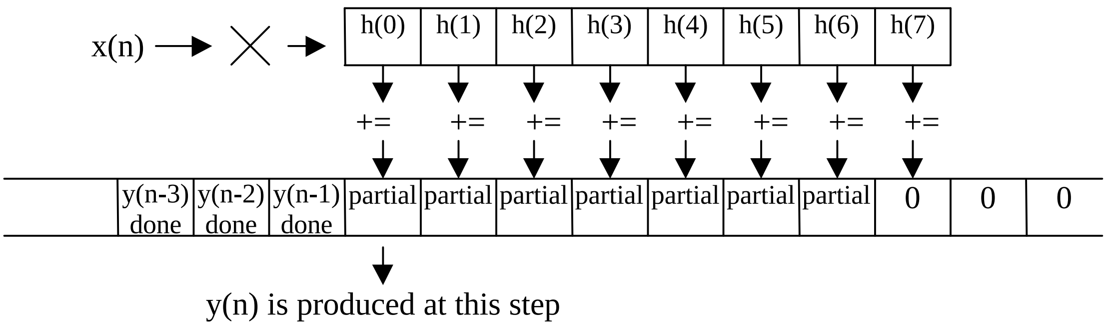

Figure 7: Implementation of convolution algorithm by generating partial sums.