Lynx Chipper
Coded by Sage

Document Date : April 27th 2011
VER: 0.1

(rough translation and additional guide by cTrix^disasterarea)

Forward

Please note this is written for an **early beta build** of the software and not yet a complete tool for writing and compiling music. At current build stage (and check for updates) it can only generate sounds – not sequence them - but think of this as a good opportunity to learn the art of sound generation from this tool while the rest is under dev. :-)

A lot of the seemingly weird and wonderful looking features of this program will become apparent as you learn the limitations and work arounds required to make music for the Lynx platform. While this may look like a complex program, it is essentially easy to use once you learn to work with it.

If you keep things simple – then this tool is straight forward.

What is Lynx Chipper?

This is a tool for creating sounds / music for the Atari Lynx portable platform.

The editor consists of:

- Instrument Editor
- Tremolo, Vibrato and Waveform Editor
- Fourier analysis window (black window)
- List of "preset" waveforms (Well, sort of. It's the LenSort part)

And in the future:
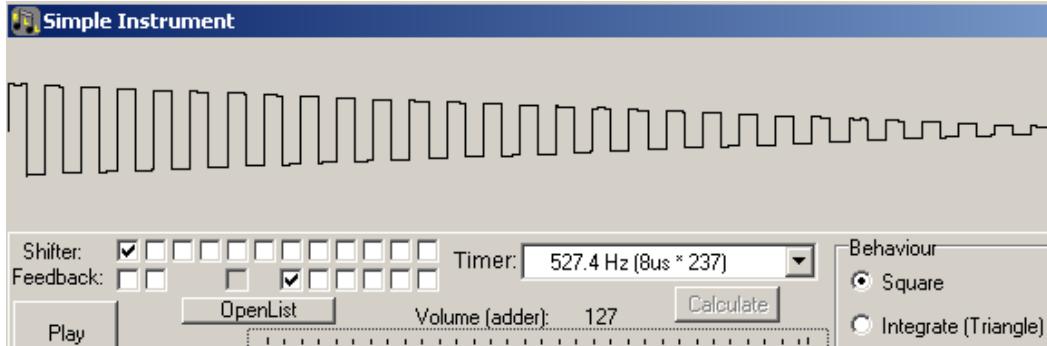
- The Tracker (currently under dev)
- Import tool (future)

The Waveform Editor (with PC generated preview)

This concept is based on the hardware design of the Lynx. In contrast to ST, where you have waveforms such as Square or Noise, one has to generate the sound for a Lynx by making a "wave form" which then gets converted to audio by a DAC. Note, the Lynx has a filter cutoff at 4 kHz, which is not simulated by the editor. The sound on the Lynx hardware will be smoother and more rounded than what you hear in the preview.

Chipper generates waveforms that are drawn to look rectangular or triangular. Triangular waveforms can be clipped if you push them too loud essentially making them sound slightly more square – a combination of waveform (but very loud!).

More complex forms specify harmonics and noise. You can do this with the "shifter" and "feedback" tick boxes.

How are the waveforms are generated?

The waveforms are (like the Atari 2600) generated by polynomial counter. These use a "shift register" (top tick boxes) and a "feedback form" (below). The shifter register is how each "clock cycle" is a bit pushed up and with the feedback register logically. The number of bits is counted and the result can be a zero (0) or a one (1). Depending on the mode the following happens:
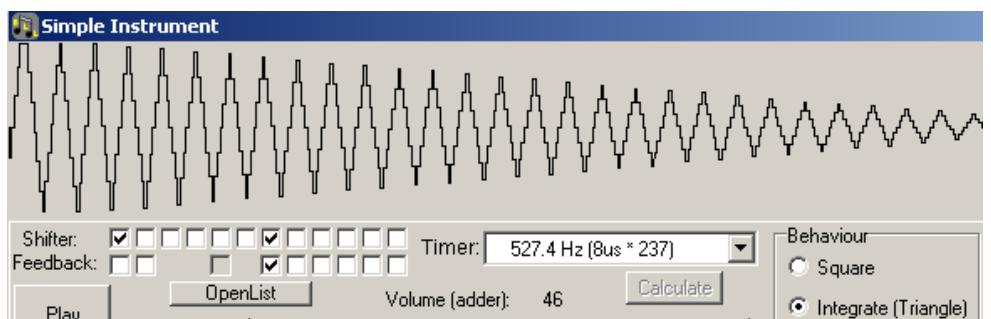
Square (Behaviour) - *Pictured Above*
With a 1, a positive "Volume" is sent to the DAC, and with a 0 a negative value is sent to "volume". This creates a square looking waveform which is effectively one bit. Think of this a little bit like the pulse wave channel of a Gameboy DMG – except you can set up a series of pulses in a row. Please note that the pitch will change depending on the length of the sound you create. (note "Länge" = length in german)
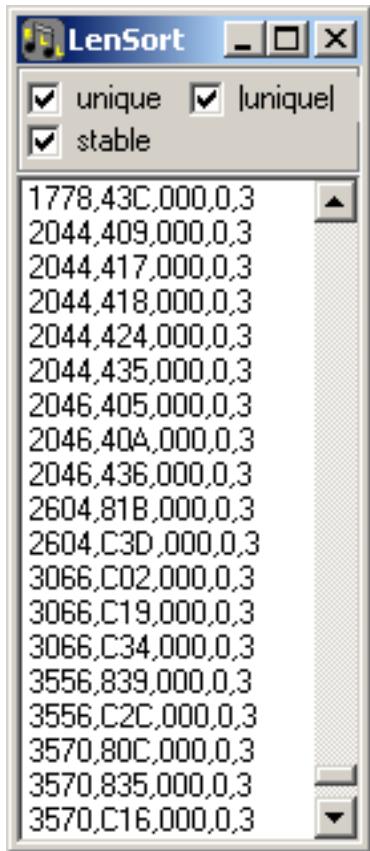
Integrate (Triangle Behaviour)
With a 1, the "volume" of the DAC adds, and with a "0" it subtracts. So rather than a pulse, the waveform diagonally rises and falls within a short period of time sounding smoother.

In the Integrate (Triangle) mode, the tones appear much louder. You can use the Volume slider to change the perceived loudness of a wave form and also clip it into a square. The waveform is displayed 3 or four times in succession to make the waveform clearer to look at. The complete generated waveform (also showing the envelope) is displayed at the top of the screen.

LenSort List

The big list of numbers (titled LenSort) is a list of all possible waveforms / sounds than can possibly be generated sorted by length.

As a general rule:

The sounds at the top of this list are high pitched and "simple" waveforms. The sounds will get lower and more complex as you go down the list. At the bottom of the list, the waveform is so complex it becomes noise

Tick "unique" to get a list of sounds which do not repeat.

Tick "stable". Waveforms generally go up and back down again – aka they cycle. This is what you see in a wave editor. There is a push and a pull. If you have an ODD length you'll find the generated waveform DC biases to either positive or negative. Trust me – it's better to have stable DC biased waveform!! So I'd recommended ticking this.

Important: The note pitch will be dictated by the speed the timer is running at. The timer can be changed to different intervals but please note that *the lower the timer, the more likely that your notes will be out of tune on playback in the lower notes.* So if you are making a low octave bass note, you will want to set the timer to a higher value and choose a more complex sound.

You can check the timer / tuning correlation by clicking "check note freq". (genius!)

LenSort window contents:

```
☑ unique   ☑ |unique|
☑ stable

1778,43C,000,0,3
2044,409,000,0,3
2044,417,000,0,3
2044,418,000,0,3
2044,424,000,0,3
2044,435,000,0,3
2046,405,000,0,3
2046,40A,000,0,3
2046,436,000,0,3
2604,81B,000,0,3
2604,C3D,000,0,3
3066,C02,000,0,3
3066,C19,000,0,3
3066,C34,000,0,3
3556,839,000,0,3
3556,C2C,000,0,3
3570,80C,000,0,3
3570,835,000,0,3
3570,C16,000,0,3
```
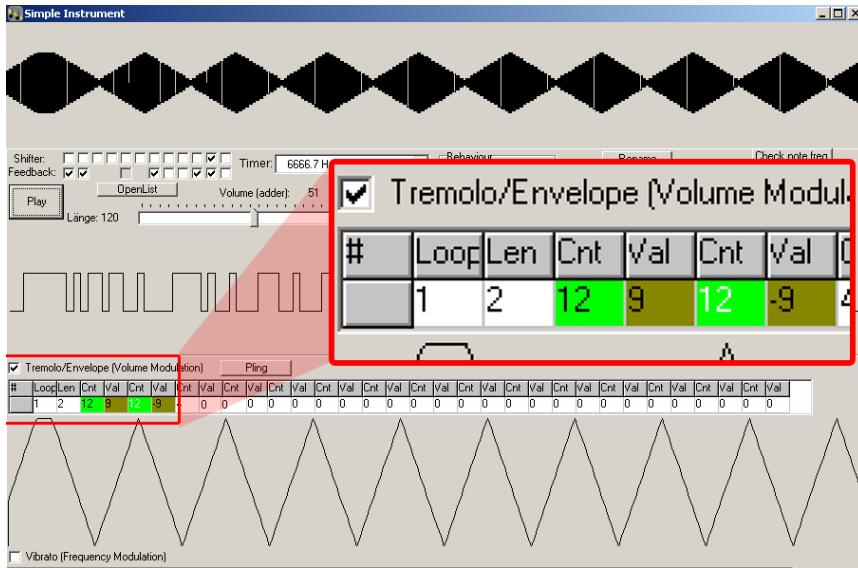
Volume Envelope / Tremolo / Pitch Vibrato

The envelope step rate is 240 times per second in the current build. This value may be able to change in future builds but this is also the playback speed of the tracker, and an ample rate for a smooth graduation (or even simple single op frequency modulation at low frequencies!)

It is a very simple modulator which pushes the amp/pitch up and down at a given rate (which is 240 times a second multiplied by the continuation value). I will cover modulation first, then envelope modes.

CONSTANT MODULATION

An exaggerated slow modulation is shown in this example. You can see the effect on the top waveform.
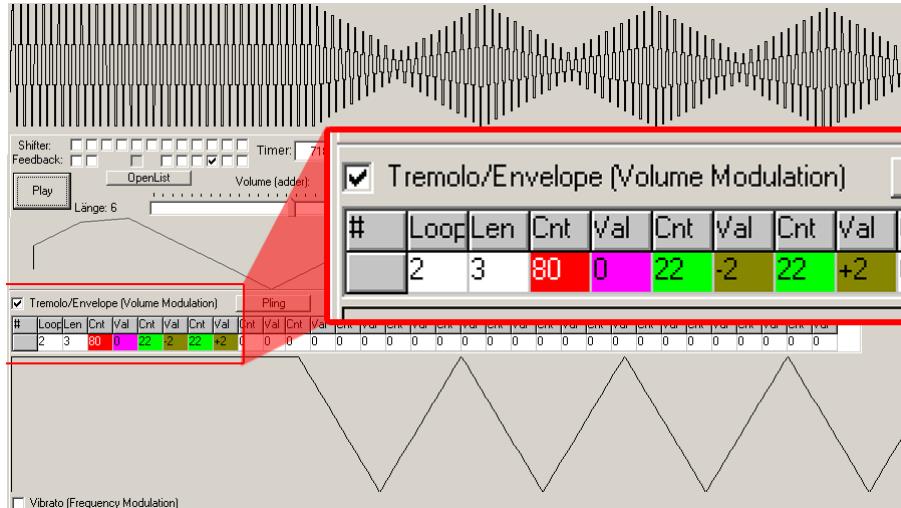
Loop: Sets the start point where to loop the modulation. If set to 1 then it modulates from the first green value.

Len: Sets the length of the loop. A value of 2 allows the modulation to drop and rise (a positive and negative).

Cnt is count. In this case we are counting 12 ticks and changing the value by +9 each tick... then counting another 12 ticks and making the value -9 each time.

The count values run in the range of 1-50, the values usually -5 to +5 but can be larger for more excessive effect.

DELAYED MODULATION
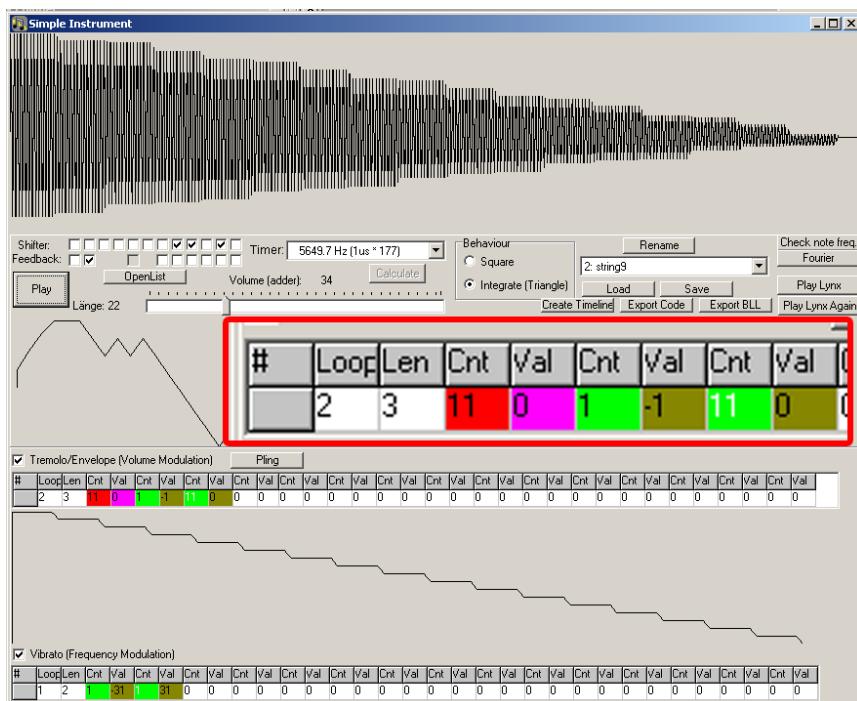
Here, our Length is set to 3 so that we have three values to work with:

1) The initial delay before mod
2) The modulation down in vol
3) The modulation up in vol

Our LOOP is set at 2 which means it will trigger the red value first (just a delay) then move onto the first green value and keep repeating them in a loop. IF we added a -1 to value then the note would fade slightly before moving onto the modulation.

DELAYED ENVOLOPE
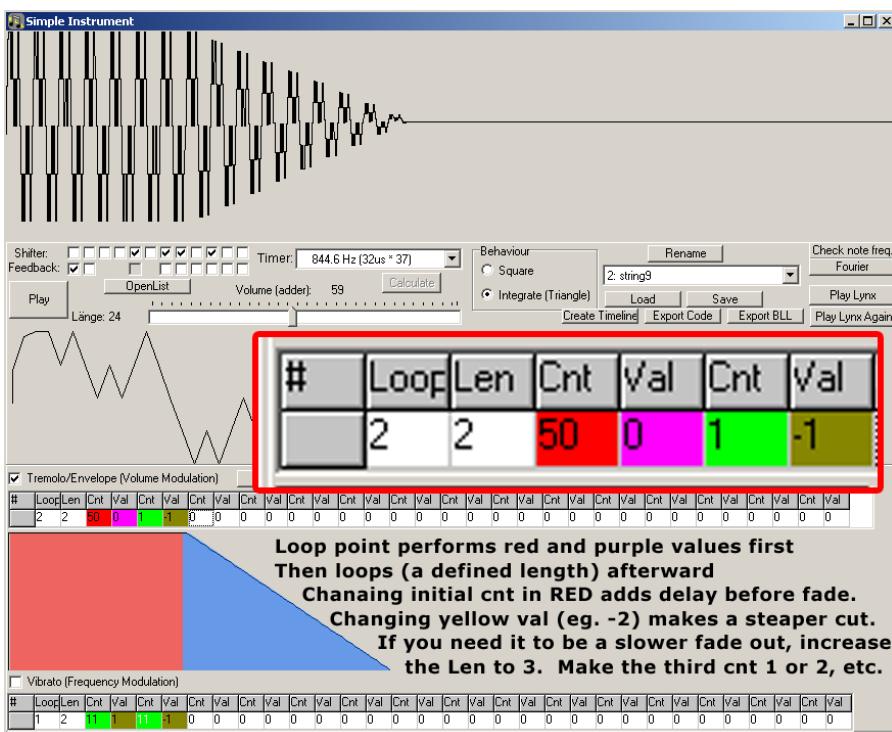
Here, our Length is set to 3 so that we have three values to work with:

1) The initial delay before fade in red

2) The fade down for 2 ticks
(cnt is effectively repeat command)

3) A pause for 11 ticks to wait before turning the volume down again.

Our LOOP is set at 2 which means it will trigger the red value first (just a delay) then move onto the first green value and keep repeating them in a loop.

I've since realised that the value in the first cnt green cell should actually be 0. This would lead to a smoother fade out. (the above demo is slightly rougher as you can see it stepping)

| #	Loop	Len	Cnt	Val	Cnt	Val	Cnt	Val
	2	3	11	0	1	-1	11	0
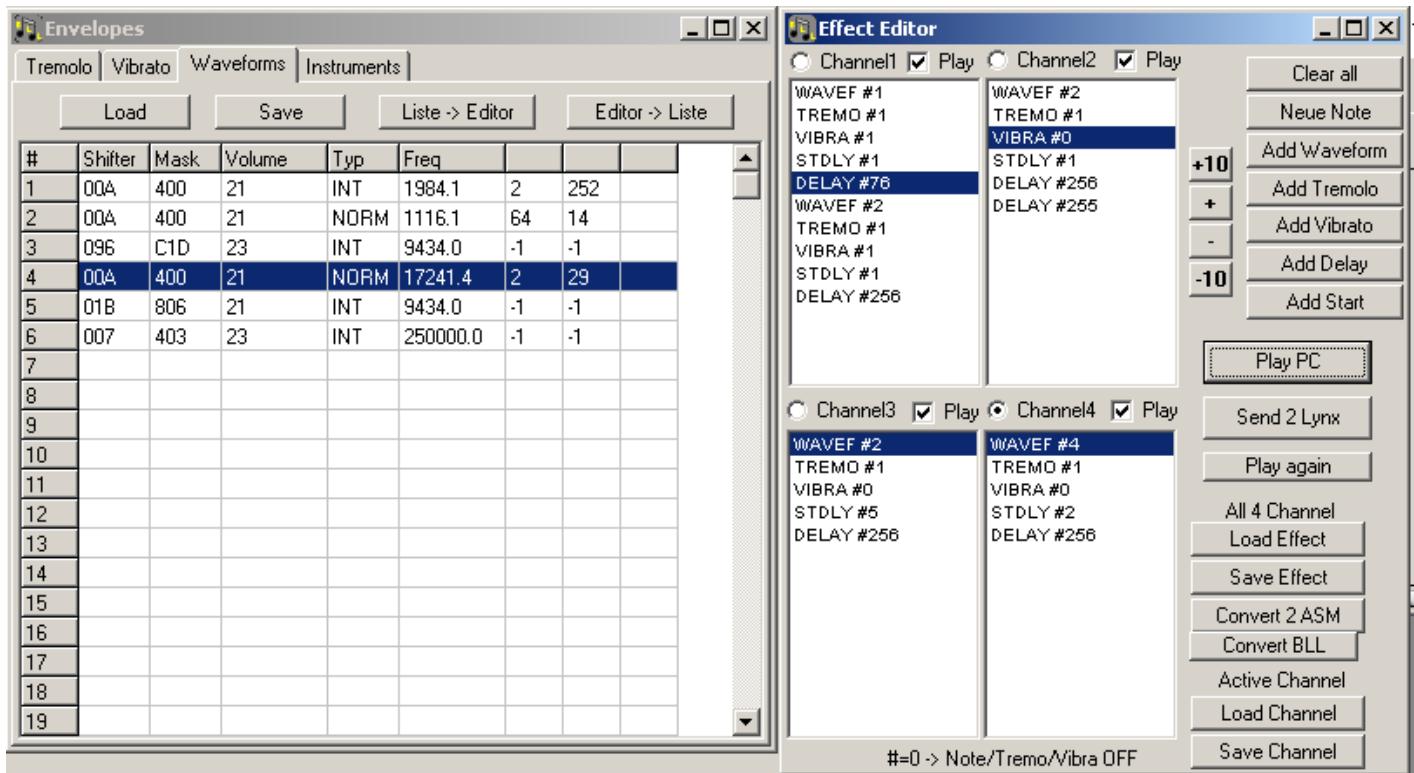
You can make a staccato note by doing a quick fade out. Here we are pausing for 50 ticks then simply repeating a value of -1 constantly until the value hits 0 at which point the note terminates.

If we set out loop and len both to 1 we would lose the red value all together and do the fade out instantly without the hold of 50 ticks at the front.

#	Loop	Len	Cnt	Val	Cnt	Val
	2	2	50	0	1	-1

Loop point performs red and purple values first
Then loops (a defined length) afterward
Chanaing initial cnt in RED adds delay before fade.
Changing yellow val (eg. -2) makes a steaper cut.
If you need it to be a slower fade out, increase the Len to 3. Make the third cnt 1 or 2, etc.

EVERYTHING that you "twiddle" with in the simple instrument window can be applied in the instrument section of the program. Think of the Simple Instrument window as a place you can experiment with new sounds. You can also use the QWERTY keyboard as a synth by pressing keys which link to notes (like in Protracker, Milkytracker, etc)

You can play more notes with the keyboard similar to the Protracker. The "C" is located on Q

Instrument / Instrument Table (Still in development)

(For LSDJ users: Think of this like a table... except you are not bound by a particular grid system and you can choose the delays manually. You can also switch waveforms / tremolo / vibrato tables simultaneously. By the looks of things, you will also be able to make multi channel instruments that are locked to specific channel for big chorused sounds, auto chords and complex drum programming.)

An instrument consists of:
A selected Waveform (you can move this across from simple editor by pressing "Editor > List" button
A selected Tremolo from the table (there is no editor to list button in this build... coming soon?)
A selected Vibrato from the table

With + / - you can change the value of the selected entry.
With New Note you can add a complete set of waveforms to play in order.

Example:

WAVEF # 15	Play waveform 15
DELAY # 50	Hold waveform 15 for 50 ticks
WAVEF # 16	Play waveform 16
TREMO # 1	With tremolo from table position 1
VIBRA # 1	With vibrato from table position 1
STDLY # 120	Continue above effects for 120 ticks
VIBRA # 2	Switch to vibrato pattern after the 120 ticks
DELAY # 240	Play waveform 16 for 240

TREMO / VIBRA / WAVEF only # 0 to # 31
DELAY # 1-256
STDLY # 1-256