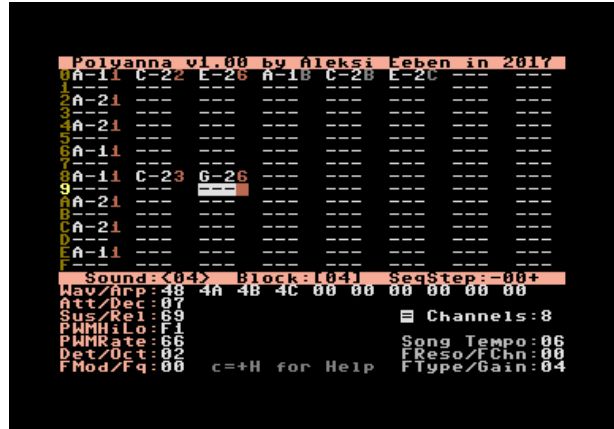


Polyanna v1.00

(27 January 2017)

Introduction

Polyanna is a 8-channel music editor for Commodore 64. It plays 3 regular SID channels together with 5 emulated SID channels on a stock machine. Old SID chip (6581) is required.



The mixer is based on Sid Vicious, a SID emulator I wrote for VIC 20 in 2006. For optimal performance the emulated channels in Polyanna are optimized for pulse waveform only. Pulse width modulation and envelopes are of course supported, although they're rough. Don't expect too much! :)

There is also an optional 6-channel mode with improved sound quality:

- 8 channels: 3 regular SID + 5 emulated channels @4.4kHz
- 6 channels: 3 regular SID + 3 emulated channels @7.7kHz

You can easily switch between modes in the editor. Don't worry, your note data in the hidden channels is never lost :)

Crash Course

The interface should be quite easy if you've tried any SID or 8-bit music software before. There's a few things to note:

- Regular SID channels use sounds labeled 1-7. Emulated channels use sounds 9-F (in hex). The correct bank is automatically selected when moving cursor in the block editor. First 3 channels to the left are SID channels
- Lowest sustain values may become silent, depending on your mixer gain. Use sustain 3-4 to play quiet, sustained notes on the emulated channels or crank up the gain

- Emulated channels are good for fast transients, repeated sounds and low PWM basses. They are very accurate. With high-pitched notes, long decayed sounds or other sounds relying on smooth envelopes they're not that good. Consider which sounds to place on regular SID channels instead
- When playing chords on the emulated channels you can make it a little smoother if you employ multiple sounds and slightly change the PWM and envelope values for each note
- In sound edit, most of the parameters are single nybble, even though the values are whole bytes. For example Det/Oct means that the high nybble is detune and low nybble is octave
- Press c= + H for a summary of keyboard commands any time
- There's some rough edges, but I hope you enjoy the ride!

Key Commands

- | | |
|----------------|---|
| Return | Focus to block edit |
| Shift + Return | Focus to sound edit |
| Space | Play song
Press Run/Stop to stop playing |
| Shift + Space | Play current block |
-
- | | |
|---------|---|
| < and > | Go to previous/next sound
SID channels: 1-7 / Emulated channels: 8-F |
| [and] | Go to previous/next block
There's 63 blocks numbered 01 to 3F (hex). Blocks are 32 steps in length, but you can cut a block short with Brk command (c= + B) |
| - and + | Go to previous/next sequencer step
Sequencer is a list of blocks to play in song. Changing block with [and] edits the current sequencer step. Block 00 marks end of song |

Inst/Del Delete note at cursor in block edit mode

. Set sound. Changes the sound number at cursor position to current sound

/ Play note at cursor position

c= + X Cut current block or sound to buffer

c= + C Copy current block or sound to buffer

c= + V Paste to current block or sound

c= + U Transpose block up by a semitone

c= + D Transpose block down by a semitone

c= + B Insert block break command Brk at cursor

c= + E Erase track at cursor column

c= + T Copy track at cursor column to track buffer

c= + P Paste to current track at cursor column

* Switch between 1-7 and 9-F banks in sound edit
Handy if you want to copy sounds between banks. You can also switch sound bank by moving cursor in the block editor to the emulated channels

c= + L Load song

c= + S Save song. A saved song is an uncompressed executable file and includes the replayer code

= Switch between 6 and 8 channel modes. This information is also saved with the song

<- (arrow left) Key up / Gate off

Musical Keyboard

2 3	5 6 7	9 0	
Q W E R T Y U I O P @			C-2 to F-3
S D	G H J		
Z X C V B N M			C-1 to B-1

Hold Shift to play notes in sound edit mode. Shifted notes in block editor play as tied notes, that only change the frequency, but without retriggering the sound

Sound Parameters

- Wav/Arp Waveform/arpeggio table of 10 values, read once a frame. Enter 00 to stop, otherwise the 10 values will loop over and over.
- High nybble is waveform (1 = triangle, 2 = saw, 4 = pulse, 3 tri+saw, 5 tri+pulse or 8 = noise)
- Low nybble is arpeggio value in semitones, relative to the current note (0 = current note, 3 = minor third, 7 = perfect fifth, B = major 7th, C = octave, E = ninth, etc.)
- Noise waveform is an exception and plays absolute pitch instead of relative arpeggio notes (81 = lowest noise, 8F = highest noise)
- Emulated channels can only play pulse waveform (4), and they interpret any given waveform value as 4
- Att/Dec Attack and decay envelope times
- Sus/Rel Sustain level and release time. Key up / gate off (left arrow, shown as -/- in block data) moves to the release phase
- PWMHiLo Pulse width modulation boundaries. Setting funny values here may cause pops and clicks.
- High nybble is top limit of modulation. Low nybble is bottom limit of modulation and also the starting value
- PWMRate Pulse modulation rate. 01 is slowest, 10-40 moderate, 80-C0 (hex) fast, FF the fastest. Pulse width modulation always starts from the low limit and initially moving upwards

Det/Oct Detune and base octave

High nybble is amount of detune. This is useful when layering multiple sounds. Only detuning upwards is implemented.

Low nybble is base octave (0-7) of the sound. Keyboard and block data only cover a 2.5 octave range, so it's here how you define which sounds are basses and which are screaming high leads

Fmod/Fq Filter modulation rate and filter cutoff frequency

Filters can be only triggered by notes on the first SID channel. However, you can run any of the 3 SID channels through the filter if you like.

Before using filters, you must initialize the parameters FReso/FChan and FType (see below). These can be defined at song level, so Polyanna filter functions are quite limited.

Fmod/Fq value of 00 means no filters are used for this sound.

High nybble is modulation rate, a 4-bit signed value: 8-F is modulation down, where 8 is fastest and F is slowest. 1-7 modulates up: 1 slowest and 7 fastest. 0 is no modulation.

Low nybble is starting cutoff frequency (1-F). The actual SID cutoff value is 10-F0 (hex).

Unlike PWM, filter modulation doesn't bounce back. Upward modulation stops at maximum value and downward modulation stops at zero. You can create more complex cutoff modulations by combining sounds and triggering them appropriately.

Fmod/Fq value of 00 doesn't reset the modulation, so you can use it to create filter sweeps spanning multiple notes. Just use another sound to initialize the modulation first

Song Tempo Song tempo in half frames. Default is 0C (hex) which steps in the block every 6 frames

FReso/Fchn Filter resonance and Filter channel switch

High nybble is resonance 0-F and low nybble selects filtered SID channels in binary:
1 = channel 1, 3 = channels 1 & 2, etc.

FType/Gain Filter type and emulated voices mixer gain

High nybble is filter type: 1 = low-pass, 2 = band-pass, 4 = hi-pass, and these can be combined.

Low nybble sets mixer gain (0-7) for the emulated channels. 0 is quietest and 7 very loud (distorted). 4 is the default and 5-6 is close to the volume of regular SID channels

Known Bugs

- Shifted D#3 key is not recognized, live with it :)

Future Plans

- Multi-SID support (without emulated channels)
- Compressed song files and some basic disk options
- More sound slots and ditch separate sound banks
- Vibrato and portamento