The HUGETracker  
manual

introduction

Hi, this is the manual to hUGETracker. I wrote this program because there wasn’t a music editing tool for the Gameboy which fulfilled the following requirements:

* Produces small output
* Tracker interface
* Usable for homebrew titles
* Open source

But now there is!

I’d like to acknowledge

1. Christian Hackbart for creating UGE, which serves as hUGETracker’s emulation core
2. Rusty Wagner for writing the sound code which was adapted for UGE
3. Lior “LIJI32” Halphion for SameBoy, a super-accurate emulator which I used for debugging and copied a bit of code from
4. Declan “Dooskington” Hopkins for GameLad, which I yanked a bit of code from
5. Eldred “ISSOtm” Habert, who helped me navigate the Gameboy’s peculiarities and for writing an alternative sound driver for the tracker
6. The folks who created RGBDS, the assembler used for building ROMs from songs.

I hope you enjoy composing in hUGETracker, and if you make any cool songs, I’d love to hear from you and potentially include them as demo tunes that come with the tracker.

E-mail me at [yux50000@hotmail.com](mailto:yux50000@hotmail.com) and get in touch!

-Nick “[SuperDisk](http://nickfa.ro)” Faro

Contents

[Concepts 4](#_Toc28880421)

[Interface 5](#_Toc28880422)

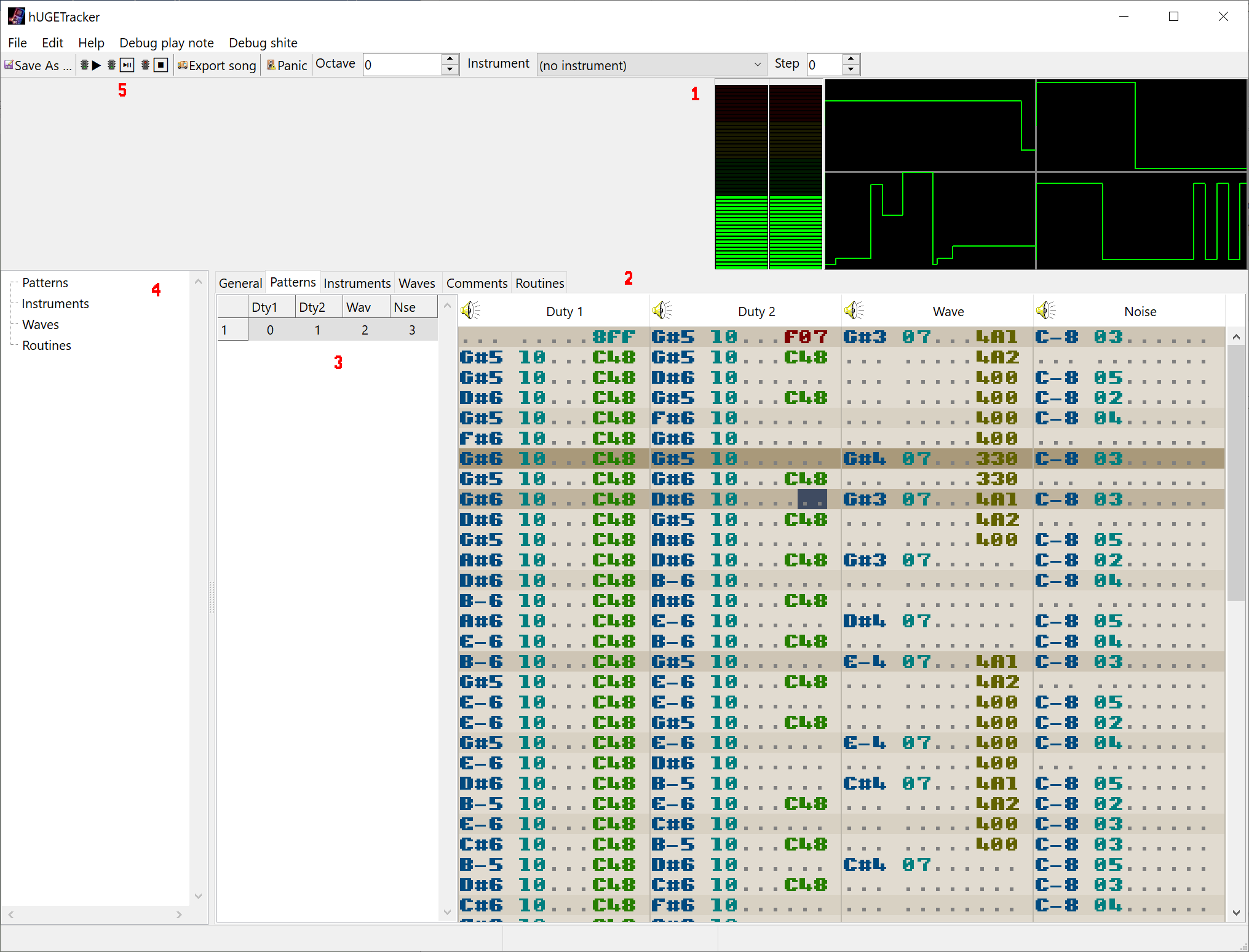
[Effect reference 7](#_Toc28880423)

# Concepts

To be written

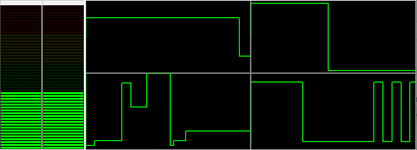
# Interface

The hUGETracker interface is styled similarly to conventional trackers such as ModPlug Tracker or ProTracker. If you’re comfortable composing in a tracker interface, then you’ll feel right at home.



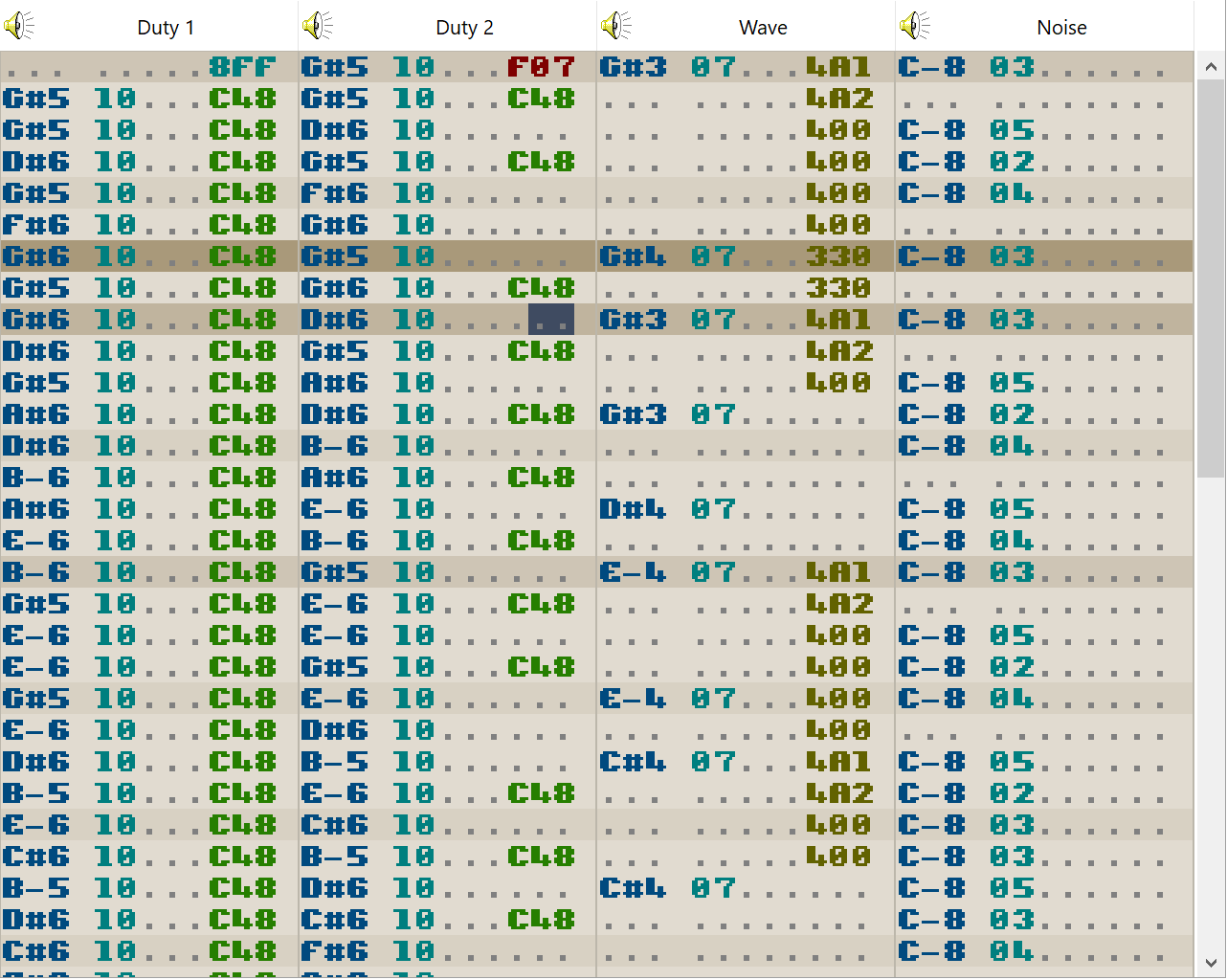
1. VU Meters and Oscilloscopes
2. Tracker Grid
3. Order Editor
4. Song components
5. Toolbar

## VU Meters and Oscilloscopes



The VU meters show the volume level for the left and right speakers. When volume gets too loud, they display as yellow/red. The oscilloscopes show the waveforms generated by the four Gameboy channels, duty 1, duty 2, wave, and noise.

## Tracker Grid



# Effect reference

The hUGETracker effect codes are intentionally similar to ProTracker or FastTracker’s. If you know them, then many of these effects will look familiar to you.

|  |  |  |
| --- | --- | --- |
| Effect | Name | Description |
| 0xy | Arpeggio | On every tick, switch between the playing note, note + x, and note + y, where `x` and `y` are values in semitones. Can be used to create “chords” or a strum effect. |
| 1xx | Portamento up | Slide the pitch up by `xx` units every tick. |
| 2xx | Portamento down | Slide the pitch down by `xx` units every tick. |
| 3xx | Tone Portamento | Slide the pitch towards the specified note value by `xx` units every tick. Stops when it reaches the specified note value. |
| 4xy | Vibrato | Rapidly switch between the specified note value and note + y, at the rate of `x`, where `y` is a value in units. Valid values for `x` are 0, 1, 3, 7, and F. This is similar to arpeggio, except you can control the frequency, and the amount is specified in units rather than semitones. |
| 5xx | Set Master Volume | Sets the master volume control of the Gameboy for the left and right speakers. Use the effect editor to create one of these effects. Note that a volume of zero is not completely silent. |
| 6xx | Call Routine | Call a user-defined routine. See the section Routines. |
| 7xx | Note Delay | Wait `xx` ticks before playing the note in this cell. |
| 8xx | Set Panning | Sets which channels play on which speakers. Use the effect editor to create one of these effects. Can also be used as a mute for a channel by setting it to output on neither left nor right. |
| 9xx | Set Duty Cycle | Select duty cycle for either channel 1 or channel 2. If this effect appears on the noise or wave channels, it will affect channel 2. Valid values for xx are 00, 40, 80, C0. Under the hood, the `xx` value is loaded directly into ch1 or ch2’s length register, so you could theoretically achieve other effects than just duty cycle changing. |
| Axy | Volume Slide | Slide the note’s volume up by `x` units, and then down by `y` units. This effect actually retriggers the note on each tick, which might not be noticeable for instruments without length/envelope, but could potentially sound bad if those are present. Recommended to use either instrument envelopes, or the `C` command instead if you can. |
| Bxx | Position Jump | Jump to order `xx`. |
| Cxx | Set Volume | Set the volume of the channel to `xx`. Must be accompanied by a note and instrument to work. Valid values range from 00-0F. |
| Dxx | Pattern Break | Jump to the next order, and start on row `xx`. |
| Exx | Note Cut | Cut the note short after `xx` ticks. |
| Fxx | Set Speed | Set the number of ticks per row to `xx`. Can be used in an alternating fashion to create a swing beat. |