GMSFXR

A GML port of SFXR

Table of Contents

Usage

Basic Flow

sfxr_init();	1) Initialize SFXR
sfxr_preset_lasershoot();	2) Set parameters to desired values.
bufferID = sfxr_create_buffer();	3) Create a buffer with the current parameters.
soundID = sfxr_create_audio(bufferID);	4) Create an audio buffer. This is the actual sound.
audio_play_sound(soundID);	5) Play the sound.
sfxr_free_audio(soundID);	6) Free the audio from memory when you don’t need it.
sfxr_free_buffer(bufferID);	7) Free the original buffer from memory.

Basic Example

Create Event:

```
sfxr_init();                        //initialize SFXR
sfxr_preset_lasershoot();       //set parameters to make a laser/shoot noise
sfxr_set_env_decay_time(1);//set the decay time to 1 so the sound fades out over a long time
laser_bufferID = sfxr_create_buffer();
laser_soundID = sfxr_create_audio(laser_bufferID);

sfxr_preset_coinpickup();      //set parameters to make a coin/pickup noise
pickup_bufferID = sfxr_create_buffer();
pickup_soundID = sfxr_create_audio(pickup_bufferID);

canShoot = true;
```

Step Event:

```
if (place_meeting(x, y, obj_pickup)) {
        audio_play_sound(pickup_soundID, 1, false);
        var inst = instance_place(x, y, obj_pickup);
        instance_destroy(inst);
}

if (mouse_check_button(mb_left) && canShoot) {
        audio_play_sound(laser_soundID, 1, false);
        canShoot = false;
}
```

Cleanup Event:

```
//It's important to free audio before buffers otherwise you'll get an error
sfxr_free_audio(laser_soundID);
sfxr_free_audio(pickup_soundID);
sfxr_free_buffer(laser_bufferID);
sfxr_free_buffer(pickup_bufferID);
```

Reference

Wave Types Enum

ESfxrWave.Square	Returns 0
ESfxrWave.Sawtooth	Returns 1
ESfxrWave.Sine	Returns 2
ESfxrWave.Noise	Returns 3

Common

sfxr_init	Starts up SFXR.
sfxr_reset	Resets all the parameters to their default values.
sfxr_create_buffer	Creates a buffer with sound data calculated using the current params.
sfxr_free_buffer	Frees a buffer from memory. *buffer_delete* is fine to use as well.
sfxr_create_audio	Creates the actual sound from the buffer created with *sfxr_create_buffer*.
sfxr_free_audio	Frees the sound from memory. *audio_free_buffer_sound* works too.
sfxr_save_settings	Saves the current settings into a file with .sfxs file extension
sfxr_load_settings	Load settings from .sfxs file. Returns true on success and false if failed.

Presets

sfxr_preset_coinpickup	
sfxr_preset_lasershoot	
sfxr_preset_explosion	
sfxr_preset_powerup	
sfxr_preset_hithurt	
sfxr_preset_jump	
sfxr_preset_blipselect	
sfxr_preset_mutate	Slightly mutates the current sound settings.
sfxr_preset_randomize	Completely randomizes everything except for the wave type.

Main Settings

sfxr_set_wave_type	Set the wave type to use with ESfxrWave enum
sfxr_get_wave_type	Returns the wave type. Returns ESfxrWave
sfxr_set_master_volume	Set the master volume between 0 and 1
sfxr_get_master_volume	Returns value between 0 and 1

Signal Envelope Settings

sfxr_set_env_attack_time	Sets the envelope attack time between 0 and 1
sfxr_get_env_attack_time	Returns value between 0 and 1
sfxr_set_env_sustain_time	Sets the envelope sustain time between 0 and 1
sfxr_get_env_sustain_time	Returns value between 0 and 1
sfxr_set_env_sustain_punch	Sets the envelope sustain punch between 0 and 1
sfxr_get_env_sustain_punch	Returns value between 0 and 1
sfxr_set_env_decay_time	Sets the envelope decay time between 0 and 1
sfxr_get_env_decay_time	Returns value between 0 and 1

Frequency Settings

sfxr_set_freq_start	Sets the base frequency between 0 and 1
sfxr_get_freq_start	Returns value between 0 and 1
sfxr_set_freq_min	Sets the minimum frequency between 0 and 1
sfxr_get_freq_min	Returns value between 0 and 1
sfxr_set_freq_slide	Sets the frequency slide amount between -1 and 1
sfxr_get_freq_slide	Returns value between -1 and 1
sfxr_set_freq_delta_slide	Sets the frequency slide speed between -1 and 1
sfxr_get_freq_delta_slide	Returns value between -1 and 1

Vibrato Settings

sfxr_set_vibrato_depth	Sets the vibrato depth between 0 and 1
sfxr_get_vibrato_depth	Returns value between 0 and 1
sfxr_set_vibrato_speed	Sets the vibrato speed between 0 and 1
sfxr_get_vibrato_speed	Returns value between 0 and 1

Note Change Settings

sfxr_set_change_amount	Sets the note change amount between -1 and 1
sfxr_get_change_amount	Returns value between -1 and 1
sfxr_set_change_speed	Sets the note change amount between 0 and 1
sfxr_get_change_speed	Returns value between 0 and 1

Square Wave Settings

sfxr_set_square_duty	Sets the square wave duty between 0 and 1
sfxr_get_square_duty	Returns value between 0 and 1
sfxr_set_square_sweep	Sets the square wave sweep between 0 and 1
sfxr_get_square_sweep	Returns value between 0 and 1

Repeat Settings

sfxr_set_repeat_speed	Sets the repeat speed between 0 and 1
sfxr_get_repeat_speed	Returns value between 0 and 1

Phaser Settings

sfxr_set_phaser_offset	Sets the phaser offset between 0 and 1
sfxr_get_phaser_offset	Returns value between 0 and 1
sfxr_set_phaser_sweep	Sets the phaser sweep between 0 and 1
sfxr_get_phaser_sweep	Returns value between 0 and 1

Low Pass Filter Settings

sfxr_set_filter_lowpass_cutoff	Sets the low pass filter cutoff between 0 and 1
sfxr_get_filter_lowpass_cutoff	Returns value between 0 and 1
sfxr_set_filter_lowpass_sweep	Sets the low pass filter sweep between 0 and 1
sfxr_get_filter_lowpass_sweep	Returns value between 0 and 1
sfxr_set_filter_lowpass_resonance	Sets the low pass filter resonance between 0 and 1
sfxr_get_filter_lowpass_resonance	Returns value between 0 and 1

High Pass Filter Settings

sfxr_set_filter_hipass_cutoff	Sets the high pass filter cutoff between 0 and 1
sfxr_get_filter_hipass_cutoff	Returns value between 0 and 1
sfxr_set_filter_hipass_sweep	Sets the high pass filter sweep between 0 and 1
sfxr_get_filter_hipass_sweep	Returns value between 0 and 1