Screenshake Plugin for RPG Maker [MV / MZ]

LUX ScreenShake Documentation

1. License

This plugin is provided under a **commercial license**:

1. You MAY use this plugin in both personal and commercial game projects
2. You MAY use this plugin in multiple games created by you
3. You MAY make compatibility patches for personal use only
4. You MAY NOT unminify, reverse engineer, or attempt to reconstruct the original source code
5. You MAY NOT redistribute, share, resell, or include this plugin in any product without explicit written permission
6. You MAY NOT copy, adapt, or incorporate any portion of this plugin (except of the exposed public API) into other plugins, scripts, or software, whether for personal use, distribution, or sale, without explicit written permission
7. Credit to "Lux Ferra" must be included in your game credits
8. All future updates, bug fixes, and compatibility patches for this plugin will be provided free of charge to legitimate purchasers

9. Technical support is available to legitimate purchasers through available communication channels as specified

License violations may result in termination of your license without refund.

For license inquiries, contact Lux Ferra.

2. Contacts

- **Discord**: luxferra
- **Discord Server**: LUX Codes & Crafts
- **Email**: luxferra22@gmail.com

3. Overview

Default RPG Maker screen shake is limited to random x (horizontal) movement with basic power and speed controls. LUX ScreenShake expands this with:

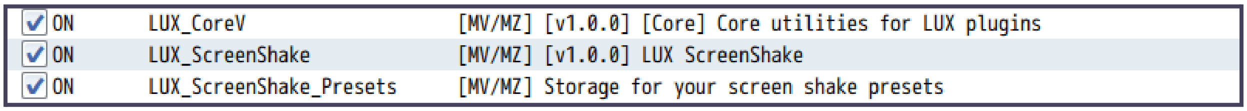
- Four shakeable screen properties (x, y, scale, and rotation)
- Multiple shake patterns (sine, noise, pulse, random, etc.)
- Customizable shake fade-in and fade-out using attack, sustain, and decay phase
- Triggering shakes through notetags, script calls, or animation timing
- Savable and reusable shake via presets

This plugin gives you precise control over all aspects of screen shake to create effects that match your game's needs.

4. Requirements

Hard Requirement: LUX_CoreV.js is required for this plugin to work. Make sure it's placed above LUX_ScreenShake.js in your plugin list.

Soft Requirement: LUX_ScreenShake_Presets.js is required if you want to use the shake preset feature. Place the plugin below LUX_ScreenShake.js in your plugin list.



☑ ON	LUX_CoreV	[MV/MZ] [v1.0.0] [Core] Core utilities for LUX plugins
☑ ON	LUX_ScreenShake	[MV/MZ] [v1.0.0] LUX ScreenShake
☑ ON	LUX_ScreenShake_Presets	[MV/MZ] Storage for your screen shake presets

5. Important Notes

- Screen shake effects are only available in Scene_Map and Scene_Battle. They will not work in other scenes like menus, title screens, etc.
- Map shakes are automatically persistent and will survive menu/battle transitions. It means when there are shakes happening on map and you go to pause menu or battle, then go back to the map, the shakes will be continued.
- The screen shakes are cumulative, which means if you have multiple screen shakes, they will be combined.
- This plugin is designed to be used with the web tool.
- **For maximum compatibility with other plugins, place all required plugins at the bottom or near the bottom of your plugin list.**

6. Plugin Parameters

The screen shakes are cumulative, which means if you have multiple screen shakes, they will be combined.
Therefore it is recommended to limit the screen shake effect on each property.

You can set the limit via plugin parameters:

- **Max X Offset**: Maximum pixel offset for horizontal shake (default: 48)
- **Max Y Offset**: Maximum pixel offset for vertical shake (default: 48)
- **Max Rotation**: Maximum rotation in degrees (default: 10)
- **Min Scale**: Minimum scale factor (default: 0.8)
- **Max Scale**: Maximum scale factor (default: 1.2)

7. Core Concepts

7.1 Shake Duration

Shake duration is always expressed in **game frames**.

- 60 game frames = 1 second
- Example:
  - A shake with duration 30 will last 0.5 seconds
  - A shake with duration 120 wil last 2 seconds

7.1.1 Game Frames vs Animation Frames

RPG Maker MV and MZ uses two different types of frames that often cause confusion:

Frame Type	Duration	Used For
Game Frame	1 game frame = 1/60 second	Wait, tint screen, flash screen, and **Effekseer in MZ**.
Animation Frame	1 animation frame = 1/15 second (default, if animation speed is 4)	**MV animation editor and MV-style animation editor on MZ**.

This difference is important when timing a screen shake for an animation. For example:

- A shake designed with 60 game frames lasts 1 second
- But an animation that runs for 60 frames lasts 4 seconds!
- Animation frame is 4x longer than a game frame at default speed (set as 4). Several plugins such as Yanfly and VisuMZ allow you to change animation speed.

Remember:

- Effekseer animations in RPG Maker MZ use game frames directly, not animation frames.
- Outside of MV-style animation timing, ALL shake effects in LUX ScreenShake use game frames. When using the web tool, only enable "Animation Frame Mode" if you're designing a shake specifically for MV-style animation timing.

This distinction is crucial when designing screen shakes for different animation types (MV-style or Effekseer).
See section “8.3 Using Animation Timing to Trigger Shake Effects” for how to handle this difference.

7.2 Shake Amplitude

Shake amplitude controls how far each property can move from its normal position during a shake. Think of it as the "intensity" of your shake effect.

For each shakeable property, amplitude means something slightly different:

- **X and Y Position**: Measured in **pixels**, this is how far the screen can move left/right or up/down from its center point. An amplitude of 20 means the screen can shift up to 20 pixels in either direction.
- **Rotation**: Measured in **degrees**, this is how much the screen can tilt clockwise or counterclockwise. An amplitude of 5 degrees means the screen can rotate up to 5 degrees in either direction.
- **Scale**: Measured as a **multiplier**, this controls how much the screen can zoom in or out. An amplitude of 0.1 means the screen can grow up to 1.1x its normal size or shrink to 0.9x its normal size.

7.3 Shakeable Properties

The plugin can shake four different properties of the screen, all using the center of the screen as the pivot point:

- **X Position** (Horizontal movement)
    - **What it does**: Moves the entire screen left and right
    - **Amplitude**: Measured in pixels (5-30 recommended)
    - **Effect**: Great for earthquakes, impacts, and rumbling effects
    - **Tip**: Larger values feel more violent, while smaller values feel more subtle
- **Y Position** (Vertical movement)
    - **What it does**: Moves the entire screen up and down
    - **Amplitude**: Measured in pixels (5-30 recommended)
    - **Effect**: Useful for impacts, jumps, and falls
    - **Tip**: Use shake bias to specify the direction of the movement (impacts, jumps, falls)
- **Rotation** (Screen tilting)
    - **What it does**: Rotates the entire screen clockwise and counterclockwise
    - **Amplitude**: Measured in degrees (1-5 recommended)
    - **Effect**: Creates disorienting effects or represents dizziness
    - **Tip**: Use sparingly! Even small values (1-3 degrees) can be very noticeable
- **Scale** (Size changes)
    - **What it does**: Zooms the screen in and out from the center
    - **Amplitude**: Measured as a decimal multiplier (0.05-0.2 recommended)
    - **Effect**: Creates pulsing effects or impacts that "push" the camera
    - **Tip**: Keep values small! a scale amplitude of just 0.1 is already quite noticeable

7.4 Shake Patterns

The plugin offers nine distinct shake patterns that control how the screen moves over time. These patterns are divided into two groups::

- Periodic (repeating)
- Non-periodic (non-repeating)

7.4.1 Periodic

- Sine
    - **Description**: Creates smooth, flowing oscillations that ease between extremes.
    - **Use Cases**: Natural motions such as swaying, gentle earthquakes, or character dizziness
    - **Tips**: The most natural-looking and smooth pattern for periodic pattern
- Square

    - **Description**: Creates sharp, digital-like oscillations with sudden jumps between extremes
    - **Use Cases**: Rough motions such as heavy impacts, strong explosions, abrupt camera jolts
    - **Tips**: Very noticeable, so use with lower amplitude for subtle effects
- Triangle
    - **Description**: Creates linear transitions between extremes with sharp direction changes
    - **Use Cases**: Structured impacts with clear direction changes
    - **Tips**: Good middle ground between sine (too smooth) and square (too harsh)
- Sawtooth
    - **Description**: Creates gradual increase with sudden drops
    - **Use Cases**: Tension build-up and release
    - **Tips**: Great for effects that build up then release suddenly
- Reverse Sawtooth
    - **Description**: Creates sudden jumps with gradual decrease
    - **Use Cases**: Impacts followed by settling, sudden movements that fade such as recoil.
    - **Tips**: Excellent for impacts where the initial shock fades away
- Pulse (duty cycle not implemented yet, use this on a later update!)
    - **Description**: Similar to square wave but with adjustable duty cycle
    - **Use Cases**: Rhythmic impacts with uneven timing
    - **Tips**: Can create asymmetric patterns like quick jerks with longer pauses

7.4.2 Non-Periodic

- Step Random
    - **Description**: Randomly changes position at regular intervals
    - **Use Cases**: Unstable terrain, small debris impacts
    - **Tips**: Higher frequency makes changes more frequent, not faster
- Random
    - **Description**: Changes completely randomly each frame with no pattern
    - **Use Cases**: Extreme chaos, severe damage, maximum instability
    - **Tips**: The most chaotic pattern
- Noise
    - **Description**: Creates smooth, organic random motion
    - **Use Cases**: Random motion but smoother and more organic. Such as realistic earthquakes or simulating an unstable hand holding camera.
    - **Tips**: The most realistic and natural "chaotic" pattern. The industry standard for screen shake

7.4.3 Controlling Shake Patterns

All shake patterns can be fine-tuned using these parameters:

- **Frequency**: How often the pattern completes a cycle (for periodic) or changes (for non-periodic)
    - **Higher frequency**: faster/more chaotic shake
    - **Lower frequency**: slower/calmer shake
- **Phase**: Shifts the starting point in the pattern
    - **Use Case**: When combining multiple properties with the same pattern (such as sawtooth on x and y), adding phase offsets creates less synchronized motion
- **Bias**: Shifts the center point of the movement
    - **Positive bias** = more movement in the positive direction
        - **x**: right
        - **y**: down
        - **rotation**: right skew
        - **scale**: zoom in
    - **Negative bias** = more movement in negative direction
        - **x**: left
        - **y**: up
        - **rotation**: left skew
        - **scale**: zoom out
    - **Use Case**: When you want the shake to favor one direction, like a screen that mostly shakes downward after an impact

Each pattern supports different controls:

Shake Pattern	**Frequency**	**Phase**	**Bias**
Sine	✔	✔	✔
Square	✔	✔	✔
Triangle	✔	✔	✔
Sawtooth	✔	✔	✔
Reverse Sawtooth	✔	✔	✔
Pulse	✔	✔	✔
Step Random	✔*	✔	✔
Noise	✔**	✔	✔
Random	✗	✗	✔

* For step random: higher frequency = more frequent changes

** For noise: higher frequency = larger sampling step size = more erratic or less smooth

7.4.4 Tips for Choosing Patterns

- **For natural organic movements**: Use Noise or Sine
- **For mechanical movements**: Use Square, Triangle, or Step Random
- **For impacts with aftermath**: Use Reverse Sawtooth
- **For increasing tension**: Use Sawtooth
- **For complete chaos**: Use Random
- **For most general purposes**: Sine, Random, and Noise are usually the one you needed

7.4.5 Creating Circular Shake

You can create circular or orbital motion by using the same pattern and frequency for both X and Y properties, but only for some of the periodic shake patterns. However, different patterns create different orbital paths:

- **Sine Wave**
  - Creates true circular motion (or elliptical if X and Y amplitudes differ)
  - The smoothest and most natural circular effect
- **Square Wave**
  - Creates square path motion, jumping from corner to corner
  - Movement follows a perfect square rather than a circle
- **Triangle Wave**
  - Creates diamond-shaped path motion
  - Movement is linear between corners but traces a diamond overall

If you want X and Y to move in perfect sync instead (like both sides moving right/left at the same time), set the Y wave to match the X wave shape by adjusting the phase slider.

7.5 Shake Envelope

A good screenshake should ramp up and diminish over time for natural feeling and to prevent abrupt screen changes. This is achieved by using an envelope, a concept borrowed from signal processing.

Envelope controls how shake amplitude changes over time, similar to fade-in and fade-out.

- **Attack**: How many frames to reach full intensity (fade in)
- **Sustain**: How many frames to stay at full intensity (normal)
- **Decay**: How many frames to reach zero (fade out)
- **Degree**: Rate of change of the shake fade in and fade out, ordered from slowest to fastest

    - Linear: Constant rate of change
    - Quadratic: Accelerating and decelerating curve
    - Cubic: More pronounced curve
    - Quartic: Steepest curve

8. Using LUX ScreenShake in Your Game

The easiest way to create a screen shake is using the web tool. Just follow these steps:

8.1 Creating Your Shake Effects

1. **Visit** the LUX ScreenShake Itch.io Page
2. **Click** "Run Tool"
3. **Design** your shake effect:
    - Adjust sliders for shake duration, shake amplitudes, shake type, and envelope
    - Preview the screen shake
    - Fine-tune until you get exactly what you want
    - **Important Note**:
        - Unless you're specifically designing a shake to be triggered during an MV-style animation, keep "Animation Frame Mode" OFF. All script calls, event triggers, and notetags (except for animation timing) expect durations in game frames (60 frames = 1 second).
        - If you're designing a shake to be triggered in MV-style animation editor:
            1. Go to "Global" tab, tick the "Animation Frame Mode"
            2. Read "8.3 Using Animation Timing to Trigger Shake Effects" for more details
4. **Export** your shake effect by clicking the Export button
5. **Choose the export format** that matches how you want to use it:
    - Script call: For on map shakes, eval in action sequence, custom code, or console testing
    - Notetags: For shake on skill/item use or skill/item animation start
    - Preset Registration: For reusable effects

8.2 Using Your Shake Effects

8.2.1 Script Calls

Paste the exported script call code in any of these places:

- Event script command (this is the main way to trigger on map screen shake)
- Action sequences eval
- Plugin that accept script eval

- Developer console (F12) for quick testing

An example of a basic script call:

```
LUX.shake({ duration: 60, x: { type: "noise", amplitude: 15,
frequency: 8 } });
```

8.2.2 Notetags

Paste the exported notetag code in the note box of any skill or item.

There are two type of notetags:

- **(On Hit)** Triggers when damage/effect is applied: `<shakeOnHit:{"duration":60,"x":{"type":"noise","amplitude":15}}>`
- **(On Animation Start)** Triggers when the animation begins:

  ```
  <shakeOnAnimStart:{"duration":60,"x":{"type":"noise","amplitude":
  15}}>
  ```

8.2.3 Presets

If you find yourself reusing the same effect, then you might want to register the effect as a named preset.
With preset, you can call a shake data by its associated name.

First, you need to export your shake configuration from the web tool by:

1. Click Export.
2. Choose Register Preset.
3. Copy the code.

Now, you need to paste the code from the web tool into LUX_ScreenShake_Presets.js:

1. Download LUX_ScreenShake_Presets.js.
2. Place it below LUX_ScreenShake.js.
3. Open LUX_ScreenShake_Presets.js with a text editor of your choice.
4. Paste the code after “Add your shake presets below”.

With preset, you can use the same shake but with different duration, amplitude, and frequency by using multipliers:

- **Script call**: `LUX.shake("explosion", duration multiplier, amplitude multiplier, frequency multiplier);`

- **Notetags**:
    - `<shakeOnHit:preset:explosion, duration multiplier, amplitude multiplier, frequency multiplier>`
    - `<shakeOnAnimStart:preset:explosion, duration multiplier, amplitude multiplier, frequency multiplier>`

Note: The multipliers will affect all properties (x, y, rotation, and scale)

If you prefer to not use multipliers, you can simply call the preset as it is without any numbers:

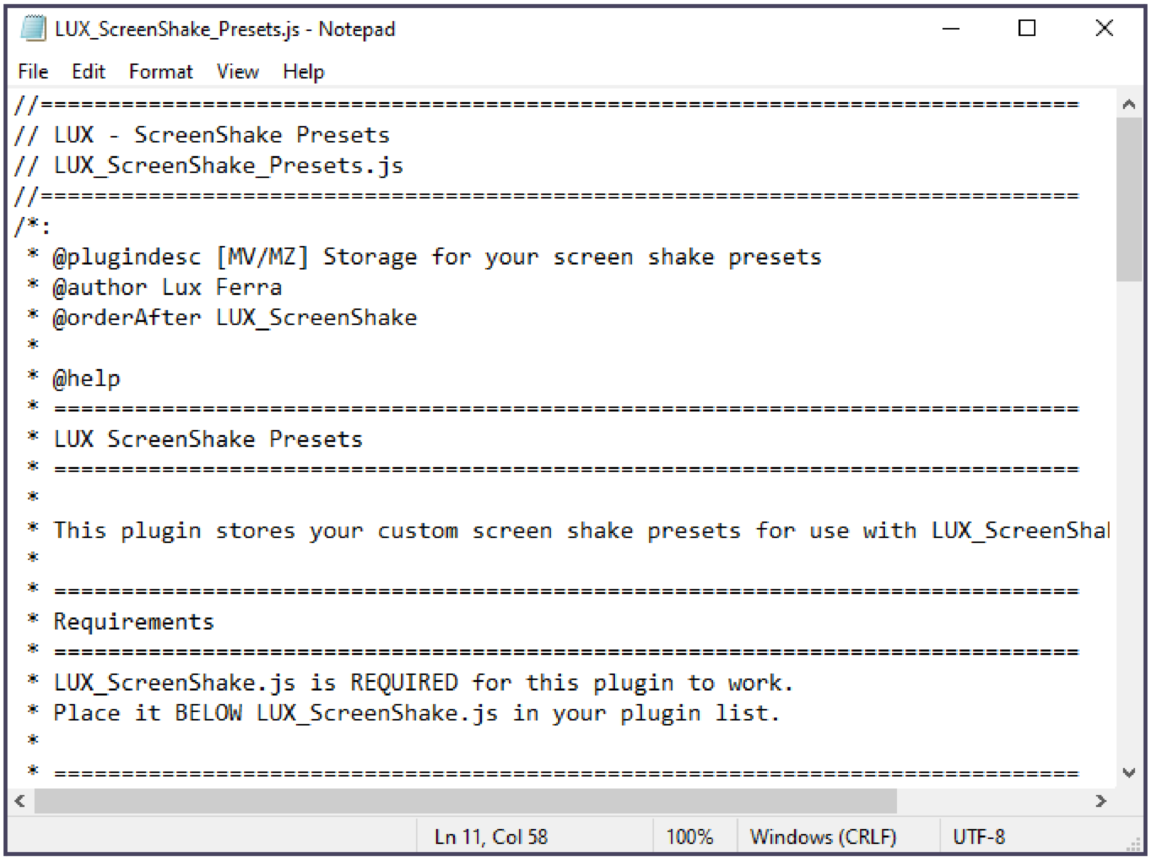
- **Script call**: `LUX.shake("explosion");`
- **Notetags**: `<shakeOnHit:preset:explosion>`

8.2.3.1 Presets Tutorial

Let's do a tutorial!
Here is a juicy explosion shake that you can save in your presets:

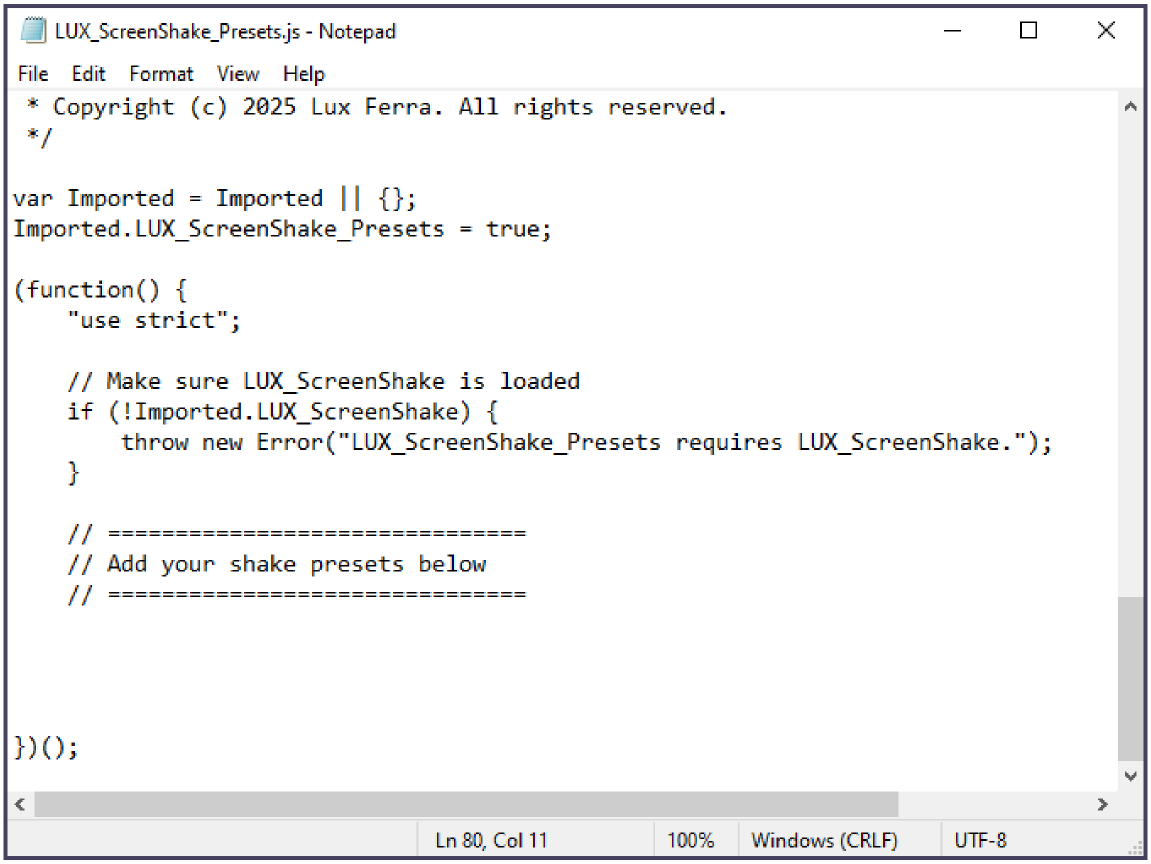
```
LUX.registerShake("explosion", {
         "duration": 60,
         "envelope": {
             "attack": 6,
             "sustain": 12,
             "decay": 42,
             "degree": "cubic"
         },
         "x": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 20
         },
         "y": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 15
         }
     });
```

1. **Open** LUX_ScreenShake_Presets.js with your favorite text editor.



```
//=============================================================================
// LUX - ScreenShake Presets
// LUX_ScreenShake_Presets.js
//=============================================================================
/*:
 * @plugindesc [MV/MZ] Storage for your screen shake presets
 * @author Lux Ferra
 * @orderAfter LUX_ScreenShake
 *
 * @help
 * ============================================================================
 * LUX ScreenShake Presets
 * ============================================================================
 *
 * This plugin stores your custom screen shake presets for use with LUX_ScreenShal
 *
 * ============================================================================
 * Requirements
 * ============================================================================
 * LUX_ScreenShake.js is REQUIRED for this plugin to work.
 * Place it BELOW LUX_ScreenShake.js in your plugin list.
 *
 * ============================================================================
```

2. **Scroll down** until you find “Add your shake presets below”

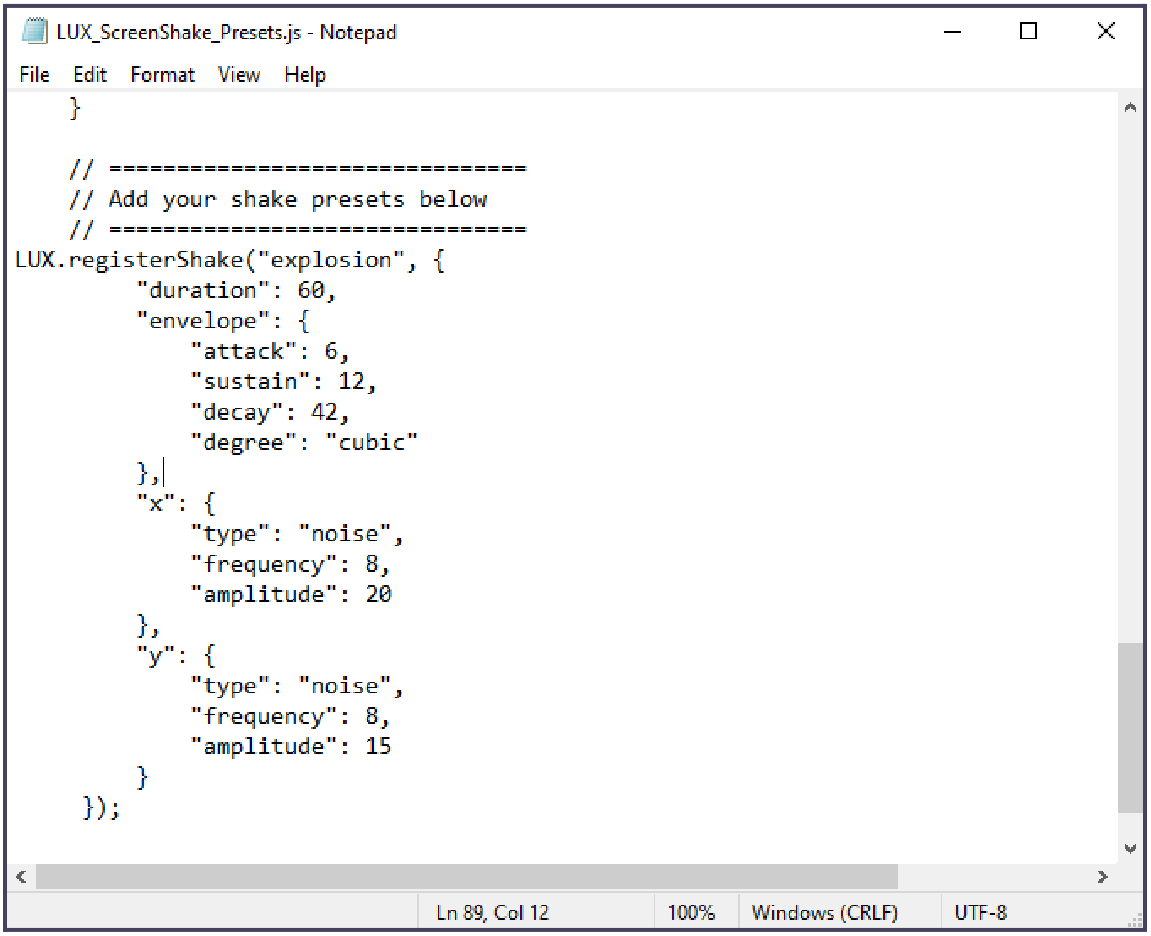


LUX_ScreenShake_Presets.js - Notepad
File Edit Format View Help

```
 * Copyright (c) 2025 Lux Ferra. All rights reserved.
 */

var Imported = Imported || {};
Imported.LUX_ScreenShake_Presets = true;

(function() {
    "use strict";

    // Make sure LUX_ScreenShake is loaded
    if (!Imported.LUX_ScreenShake) {
        throw new Error("LUX_ScreenShake_Presets requires LUX_ScreenShake.");
    }

    // ===============================
    // Add your shake presets below
    // ===============================



})();
```

Ln 80, Col 11
100%
Windows (CRLF)
UTF-8

3. **Paste** the Explosion shake code



```
    }

    // ===============================
    // Add your shake presets below
    // ===============================
LUX.registerShake("explosion", {
         "duration": 60,
         "envelope": {
             "attack": 6,
             "sustain": 12,
             "decay": 42,
             "degree": "cubic"
         },
         "x": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 20
         },
         "y": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 15
         }
     });
```

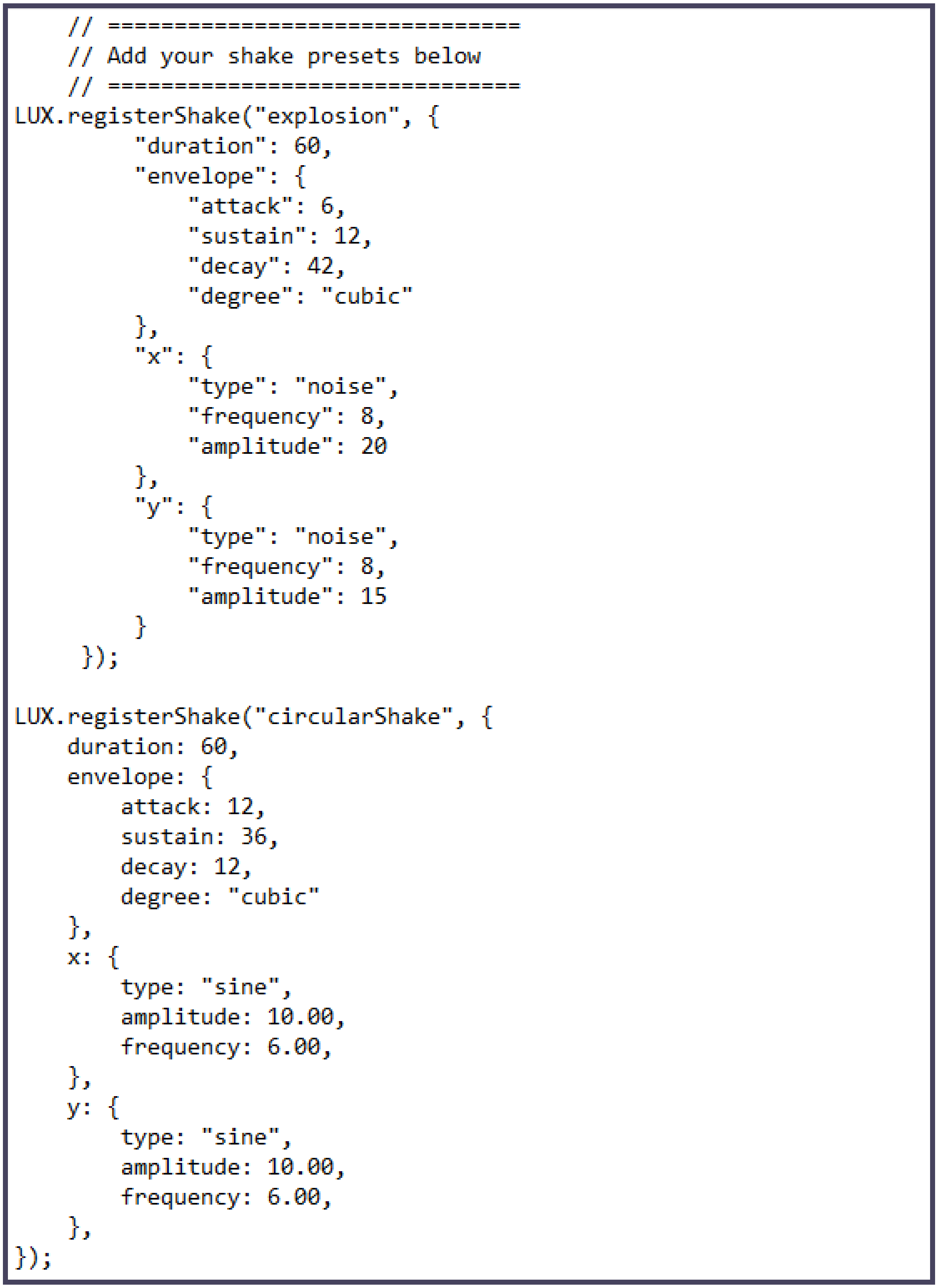
4. **Try** the Explosion shake preset! Here is a reminder on how to call it:
    a. **Script call**: `LUX.shake("explosion");`
    b. **Notetags**:
        i. `<shakeOnHit:preset:explosion>`
        ii. `<shakeOnAnimStart:preset:explosion>`

If the shake is playing, good job! Otherwise, read the step-by-step again carefully. You might have missed something!

As a reward, here is another juicy shake preset, this one will do circular shake!

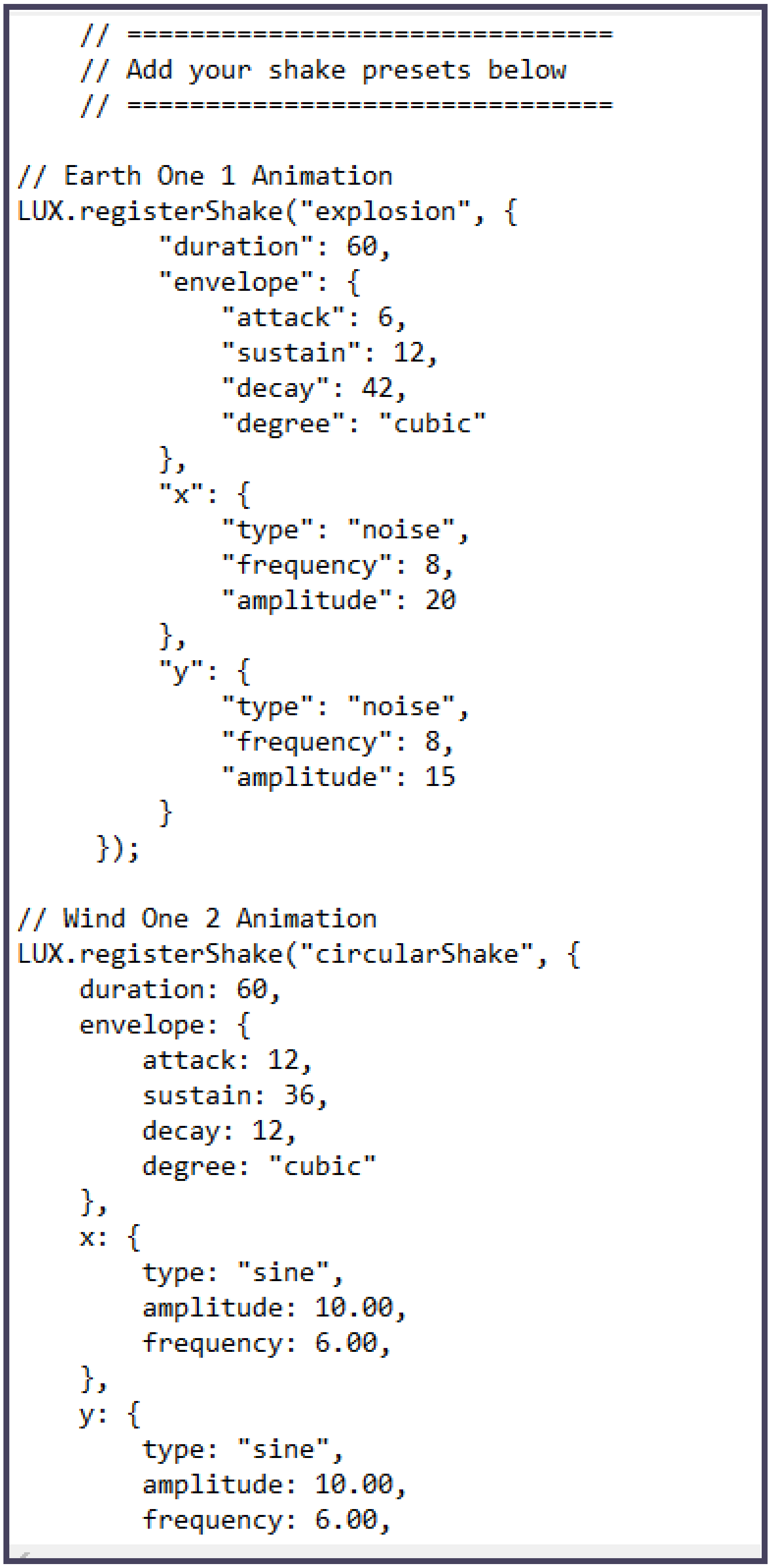
```
LUX.registerShake("circularShake", {
    duration: 60,
    envelope: {
        attack: 12,
        sustain: 36,
        decay: 12,
        degree: "cubic"
    },
    x: {
        type: "sine",
        amplitude: 10.00,
        frequency: 6.00,
    },
    y: {
        type: "sine",
        amplitude: 10.00,
        frequency: 6.00,
    },
});
```

Try adding it to the preset, just copy and paste it below the explosion preset like so:



```
    // ===============================
    // Add your shake presets below
    // ===============================
LUX.registerShake("explosion", {
         "duration": 60,
         "envelope": {
             "attack": 6,
             "sustain": 12,
             "decay": 42,
             "degree": "cubic"
         },
         "x": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 20
         },
         "y": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 15
         }
     });

LUX.registerShake("circularShake", {
    duration: 60,
    envelope: {
        attack: 12,
        sustain: 36,
        decay: 12,
        degree: "cubic"
    },
    x: {
        type: "sine",
        amplitude: 10.00,
        frequency: 6.00,
    },
    y: {
        type: "sine",
        amplitude: 10.00,
        frequency: 6.00,
    },
});
```

Keep in mind that LUX_ScreenShake_Presets.js is just a JavaScript file, so you can add JavaScript comments to help you organize it by adding double forward slash (//) and adding a line of text in front of it.



```
    // ===============================
    // Add your shake presets below
    // ===============================

// Earth One 1 Animation
LUX.registerShake("explosion", {
         "duration": 60,
         "envelope": {
             "attack": 6,
             "sustain": 12,
             "decay": 42,
             "degree": "cubic"
         },
         "x": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 20
         },
         "y": {
             "type": "noise",
             "frequency": 8,
             "amplitude": 15
         }
     });

// Wind One 2 Animation
LUX.registerShake("circularShake", {
    duration: 60,
    envelope: {
        attack: 12,
        sustain: 36,
        decay: 12,
        degree: "cubic"
    },
    x: {
        type: "sine",
        amplitude: 10.00,
        frequency: 6.00,
    },
    y: {
        type: "sine",
        amplitude: 10.00,
        frequency: 6.00,
```

Note: If LUX_ScreenShake_Preset.js feels too full for you, you can create another .js file and copy the preset registration there. Make sure the file is loaded in the plugin list after LUX_ScreenShake.js

8.3 Using Animation Timing to Trigger Shake Effects

Note: This section only applies to shakes triggered through animation timing. All other shake methods (script calls, event commands, on-hit notetags) always use game frames, regardless of where they're used.

Animation timing in RPG Maker MV/MZ is used to synchronize sound effects, flashes, and tints with specific frames of an animation.
LUX ScreenShake uses this system to trigger screen shakes at specific animation frames.

Important:
To use this feature, you must:

1. Understand the difference between game frames and animation frames.
   Go to "7.1.1 Game Frames vs Animation Frames" for an intro.
2. Know how to register shake presets in the LUX_ScreenShake_Presets.js.
   Go to “8.2.3 Presets” for more details.
3. Understand which animation editor you’re using:
   a. **MV or MZ using MV-style**: Use Animation Frame Mode in the web tool
   b. **Effekseer animations in MZ**: DO NOT use Animation Frame Mode

8.3.1 How to Create a Shake for Animation Timing

Animation frames and game frames use different timing systems.

When you want to synchronize a shake with an animation:

1. Visit the LUX ScreenShake Page
2. Click "Run Tool"
3. Decide which animation system you’re designing for:
   a. **For MV or MZ using MV-style**
      i. Go to "Global" tab and enable "Animation Frame Mode"
      ii. Set the animation speed that matches yours (default RPG Maker animation speed is 4)
      iii. When entering shake duration, enter in animation frames (the tool will convert to game frames when you export)
      iv. Example: If you enter 10 frames, it will create a shake lasting 40 game frames
   b. **For Effekseer animations**
      i. Keep "Animation Frame Mode" OFF
      ii. Enter durations directly in game frames (60 frames = 1 second)
      iii. Example: For a 0.5 second shake during an Effekseer animation, enter 30 frames
4. Design your shake effect

5. Export your shake effect as preset
6. Register the preset in LUX_ScreenShake_Presets.js

Technical Explanation:

1. **When Animation Frame Mode is OFF** (default)**:**
    a. Duration values are interpreted as game frames (60 frames = 1 second)
    b. A 60-frame shake lasts 1 second in game time
    c. This should be used for Effekseer animations
2. **When Animation Frame Mode is ON:**
    a. Duration values are interpreted as animation frames
    b. The tool automatically converts them to the equivalent game frames (×4 at default animation speed)
    c. Example: If you enter 10 frames, it will create a shake lasting 40 game frames
    d. This ensures your shake and MV-style animation timing match correctly

Note: If you've modified your game's animation speed using plugins like Yanfly or VisuMZ, make sure to change the **Animation Rate** on the web tool too.

8.3.2 How to Trigger a Shake with Animation Timing

1. Download the 'LUX_Shake_presetname.ogg' file from the plugin page.
2. Place it in your 'audio/se' folder.
3. Rename it by changing the 'presetname' part into the name of your shake preset that you want to use. Example: LUX_Shake_explosion (will use the "explosion" preset)
4. Go to the animation tab on your RPG Maker.
5. Create a sound effect using the file that contains the prefix of your shake preset name.
6. The frame of the sound effect means the shake will trigger on that frame.
7. Volume, pitch, and pan sliders correspond to duration, frequency, and amplitude multipliers.

8.3.3 Volume, Pitch, and Pan

Volume, pitch, and pan sliders correspond to duration, frequency, and amplitude multipliers.

Each slider can multiply your shake preset duration, frequency, and amplitude.

- **Volume**: Controls shake duration multipliers

    - 0 = 0.0x duration (No shake) - not recommended, useless
    - 25 = 0.5x duration (Half)
    - 50 = 1.0x duration (Normal)
    - 75 = 1.5x duration
    - 100 = 2.0x duration (Twice)

- **Pitch**: Controls shake frequency multipliers
    - 50 = 0.5x frequency (Half)
    - 75 = 0.75x frequency
    - 100 = 1.0x frequency (Normal)
    - 125 = 1.25x frequency
    - 150 = 1.5x frequency
- **Pan**: Controls shake amplitude multipliers
    - -100 = 0.0x amplitude (No shake) - not recommended, useless
    - -50 = 0.5x amplitude (Half)
    - 0 = 1.0x amplitude (Normal)
    - 50 = 1.5x amplitude
    - 100 = 2.0x amplitude (Twice)

9. For Developers

The following guidelines are for developers creating compatibility patches or integrating this plugin with their own work. Please adhere strictly to both these technical guidelines and the license terms.

9.1 Namespace Structure

This plugin provides two specific namespaces for developer interaction:

- LUX.ScreenShake.API: Public methods intended for external use
- LUX.ScreenShake.Alias: Aliased RPG Maker methods for compatibility

9.2 API Usage Policy

When integrating this plugin's functionality:

- EXCLUSIVELY use documented methods from LUX.ScreenShake.API
- Refer to the API documentation sections for available methods

9.3 Compatibility Guidelines

For compatibility with other plugins:

- LUX.ScreenShake.Alias contains references to all modified RPG Maker methods
- Use these references when creating compatibility patches
- Compatibility patches are permitted for personal use only

9.4 Technical Boundaries

To maintain stability and update compatibility:

- **RESPECT the API boundary**, only use published API methods
- **PRESERVE the LUX namespace integrity**, do not override or modify
- **REFERENCE Alias methods** for compatibility rather than overwriting or duplicating functionality
- **AVOID direct manipulation** of internal state or private properties
- **PREVENT conflicts by not extending core objects** with same-named methods

The plugin's architecture is intentionally structured to allow legitimate integration while protecting both its integrity and your game's stability.

9.5 API Documentation

The following methods in LUX.ScreenShake.API are available for external use:

`createShake(params)`

Creates and starts a new screen shake effect.

Parameters:

- params: Object containing shake configuration
  - duration: Total duration in frames (60 frames = 1 second)
  - x/y/rotation/scale: Property-specific settings
  - envelope: Optional envelope settings

Returns:

- The shake instance if successful, null otherwise

`usePreset(presetName, durationMultiplier, amplitudeMultiplier, frequencyMultiplier)`

Creates a screen shake using a predefined preset.

Parameters:

- presetName: String name of the preset to use
- durationMultiplier: Optional multiplier for shake duration (default: 1.0)
- amplitudeMultiplier: Optional multiplier for shake amplitude (default: 1.0)
- frequencyMultiplier: Optional multiplier for shake frequency (default: 1.0)

Returns:

- The shake instance if successful, null otherwise

`stopAll()`

Stop all active screen shakes immediately.

Returns:

- Boolean indicating success

`getCurrentManager()`

Get the current shake manager (Map or Battle depending on which scene you are at).

Returns:

- The active shake manager or null if unavailable

`registerPreset(name, params)`

Registers a new shake preset for later use.

Parameters:

- name: String identifier for the preset
- params: Object containing shake configuration