

For Version 1.2.3

Newest documentation available from: <https://github.com/FinlayMac/Unity-Simple-Controller-Vibration>

Unity-Simple-Controller-Vibration

A package for creating custom vibrations for gamepads using Scriptable Objects.

The tools are reliant on the New Input System and Editor Coroutines (These are imported automatically when adding the package)

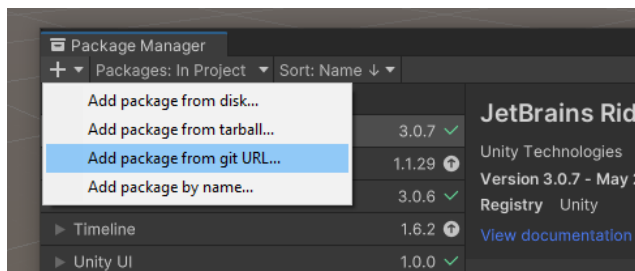
Table of Contents

1. [Getting Started](#)
2. [Samples](#)
3. [Implementation](#)
4. [Creating Vibration Parts](#)
5. [Creating Vibration Sequences](#)
6. [Testing Vibration Sequences](#)
7. [Limitations](#)

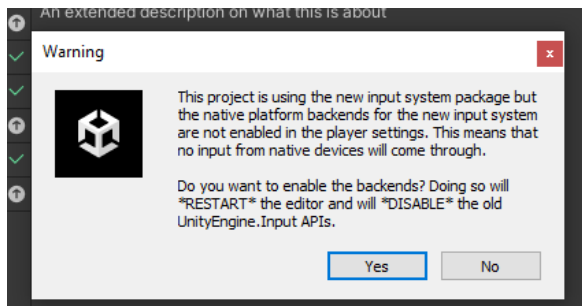
Getting Started

To use the package, add the project using Unity's package manager.

Window > Package Manager > Add Package From Git URL



If you are prompted with a warning about the new input system, select Yes

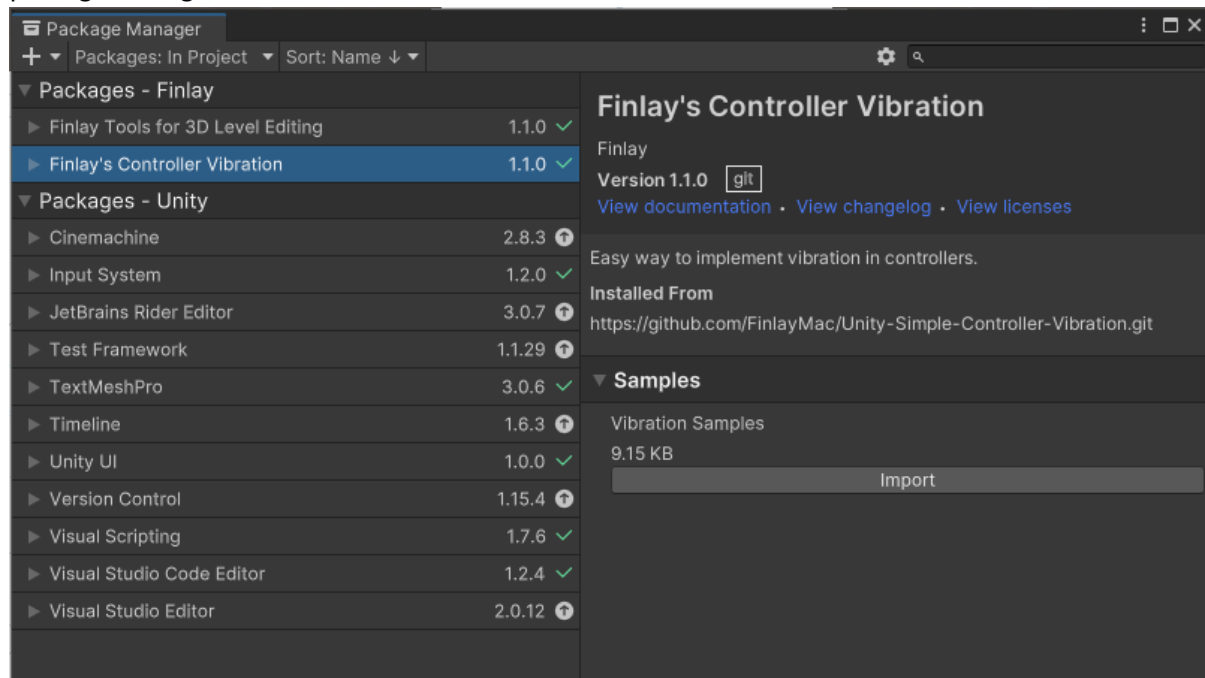


The assets are imported into the packages section.

Samples (optional)

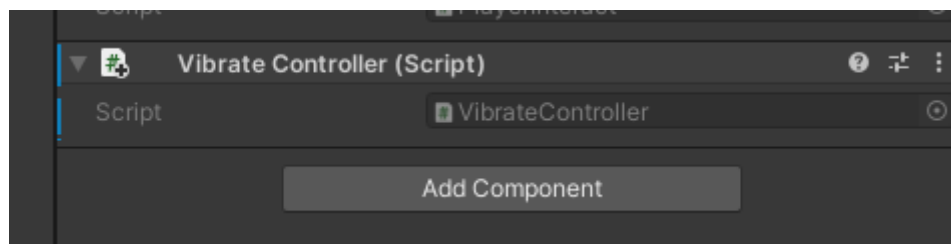
To help get you started, samples are provided.

These samples can be added into your assets folder by clicking import after adding the package in the package manager.



Implementation

To use the package, add the vibration controller Component to a Game Object.



- Adding it to the player would be a good idea but not required
- The vibration controller asset has the method Vibrate:
- Supplying a vibration sequence to this method will play the vibration

```
public void Vibrate(List<IVibrationPart> newList)
```

Example Implementation

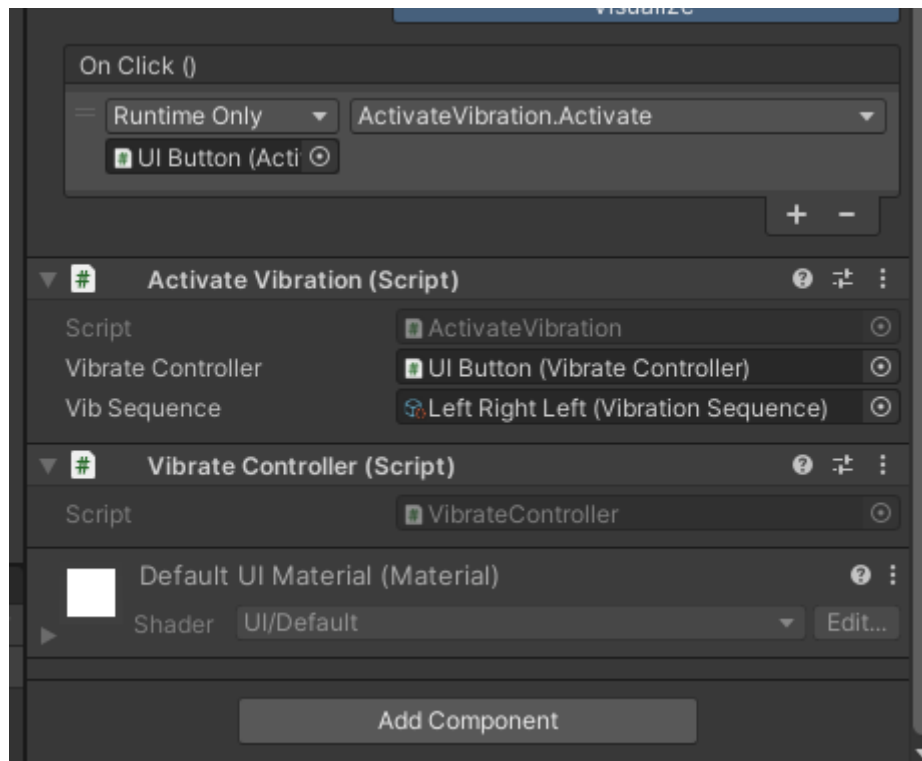
This example is only to show how to get started and I would recommend something cleaner.

- Create a new C# Script ActivateVibration and attach to a UI button.

```
using UnityEngine;
using ControllerVibration;
public class ActivateVibration : MonoBehaviour
{
    public VibrateController vibrateController;
    public VibrationSequence vibSequence;

    public void Activate()
    { vibrateController.Vibrate(vibSequence.sequence); }
}
```

- Set the On Click event to call the method Activate()
- Assign the public fields of ActivateVibration in the inspector
- If you have not downloaded the [Samples](#), you will need to [create vibration parts](#) and then [the sequence](#)



- With the game running, pressing the UI Button with a controller plugged will play the vibration sequence.
- If you have problems, try [testing your Vibration Sequences](#) using the sample assets

Creating Vibration Parts

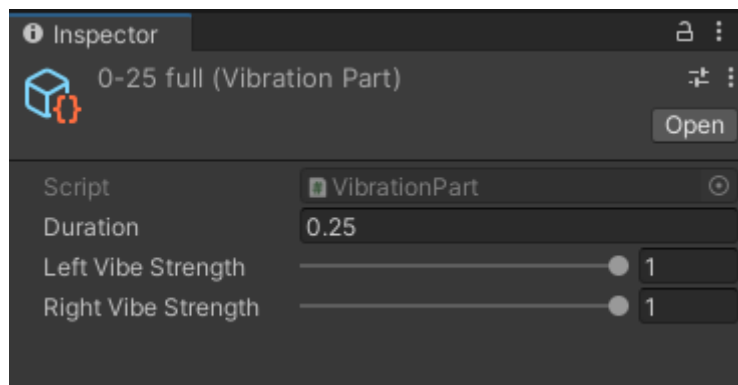
Vibration parts are the building blocks for vibration sequences.

Simple and Curve vibration assets can be created by:

- Assets > Create > Scriptable Objects > Vibration
- (either through the main menu or the Assets folder right click menu)

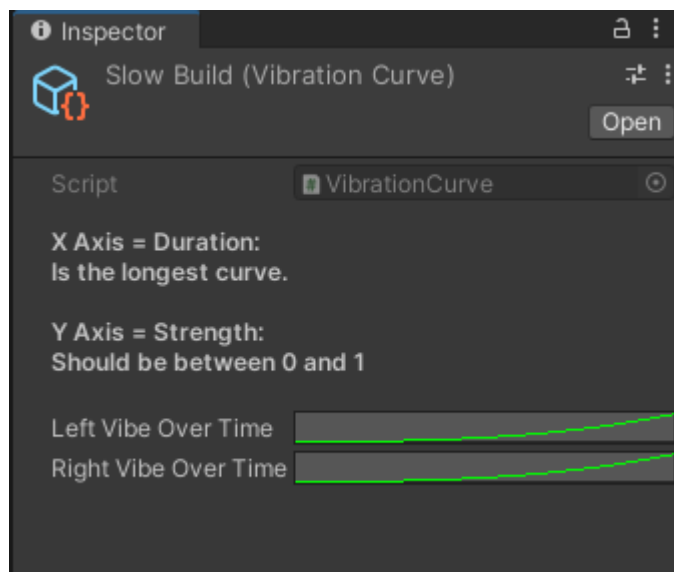
Simple Parts

Simple parts have 2 settings.

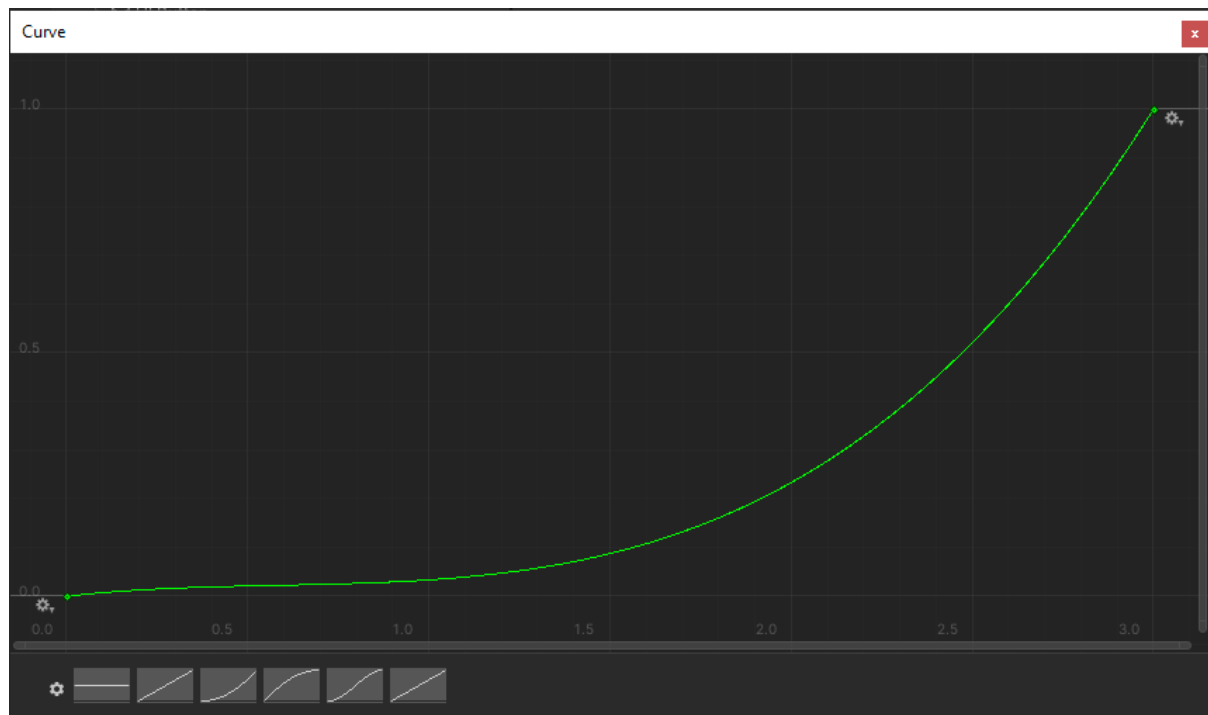


- Duration is the length of the vibration before stopping. 0 - inf
- Strength is for the strength of the left(x) / right(y) motors. 0 - 1
 - On an Xbox One controller, the left motor has the larger rumble

Curve Part



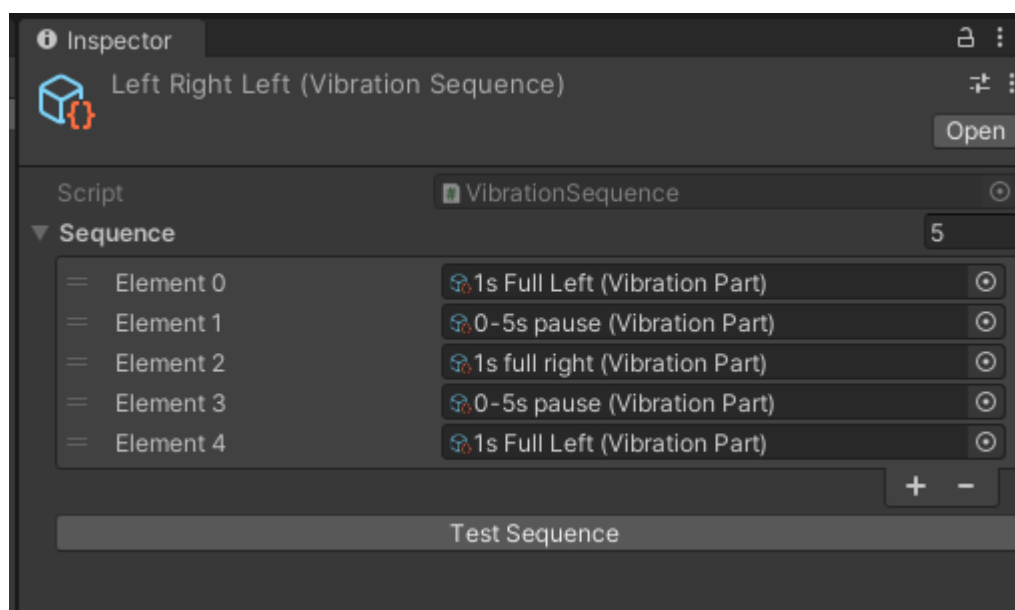
Curve parts are created from 2 graphs.



- The y axis indicates the strength of the rumble over time. (0 - 1)
- The x axis indicates the duration of the rumble (0 - inf)
- If one graph is longer than the other, the longer graph will be used as the duration of the whole vibration part

Creating Vibration Sequences

Sequences are used to play vibrations parts.



- The VibrationController.cs will expect a sequence to be passed to it
- Sequences can contain single or multiple vibration parts and are played in order.
- Vibration sequences can be created by:

- Assets > Create > Scriptable Objects > Vibration > Sequence
- (either through the main menu or the Assets folder context menu)

Testing Vibration Sequences

You can test vibration sequences in the inspector without needing to go into runtime

- Bring up the vibration sequence in the inspector
- Press the Test Sequence Button
- The Controller should perform the vibration sequence
- If your controller does not vibrate
- check it is plugged in and turned on
- it might not have vibration support
- your sequence might be empty
- your parts in the sequence might have a duration of 0

Limitations

As of version 1.1:

- It is not possible to interrupt or stop a vibration sequence
- A vibration sequence must finish before taking the next request
- Any request given whilst a vibration is in process will be ignored