

# Final Project Notes

Conor Holden

October 20, 2020

## Contents

<b>1</b>	<b>General Information</b>	<b>1</b>
1.1	Description . . . . .	1
1.2	Tools . . . . .	1
1.3	Videos about Iplug2 . . . . .	1
1.4	Tutorials about Juce . . . . .	2
<b>2</b>	<b>Digital Signal Processing</b>	<b>2</b>
2.1	Waves . . . . .	2
2.2	Fast Fourier Transform Algorithm . . . . .	2

## 1 General Information

[Project Link](#)

### 1.1 Description

Fiddle about with trying to develop a realistic simulation for uilleann pipes or violins, or some unique instrumental sound of your own imagination.

### 1.2 Tools

- [iPlug2 GitHub](#)  $\Rightarrow$  For Creating both plug-ins and stand-alone
- [iPlug2 Wiki](#)
- [Juce](#)  $\Rightarrow$  More mature and has more tutorials
- [ACM Digital Library](#)
- [Physical Modelling](#)

### 1.3 Videos about Iplug2

Abandoned

- [Oliver Larkin: Faust in iPlug 2](#)
- [iPlug2: Desktop Plug-in Framework Meets Web Audio Modules by Oliver Larkin](#)

## 1.4 Tutorials about Juce

- [Juce String Model](#)

## 2 Digital Signal Processing

- [Juce DSP](#)
- [Digital Signal Processing \(DSP\) Tutorial](#)

### 2.1 Waves

- Sin Wave  $\Rightarrow \text{std::sin}(x)$
- Saw Tooth  $\Rightarrow \text{map } -\pi - \pi \text{ to } -1 - 1$  (`juce::MathConstants<double>::pi`)
- Triangle  $\Rightarrow \text{map } -\pi - 0 \text{ to } -1 - 1 \text{ and } 0 - \pi \text{ to } 1 - -1$

### 2.2 Fast Fourier Transform Algorithm

Faster version of the Discrete Fourier transform.

- Transforms waves into its components or formula
- The inverse can be used to create sound waves from