Get instructions, example code, and more: https://github.com/tadr/WYOEM-velocity-installation

Write Your Own Eurorack Module

Get started with a simple approach to building the missing part of your eurorack system using existing hardware. Learn to write alternative software (called firmware) for a Teensy-powered module.

Step 1 – Get a Radio Music module by Music Thing Modular.

(or something similar).

This modular has a simple interface and powerful digital Arduino-based controller called Teensy that was built with audio programming in mind. Thanks to designer Tom Whitwell, Radio Music is 100% Open Source, which means the possibilities for hacking the software and hardware are endless!

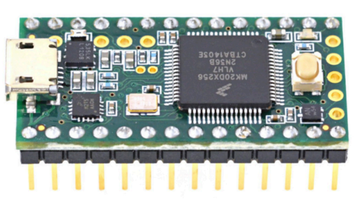
Note! Without modification, Radio Music will only output audio or gates. If you need CV output, consider choosing from other options....
See Module Options below.

Radio Music and variant Prok BD

Music Thing Modular

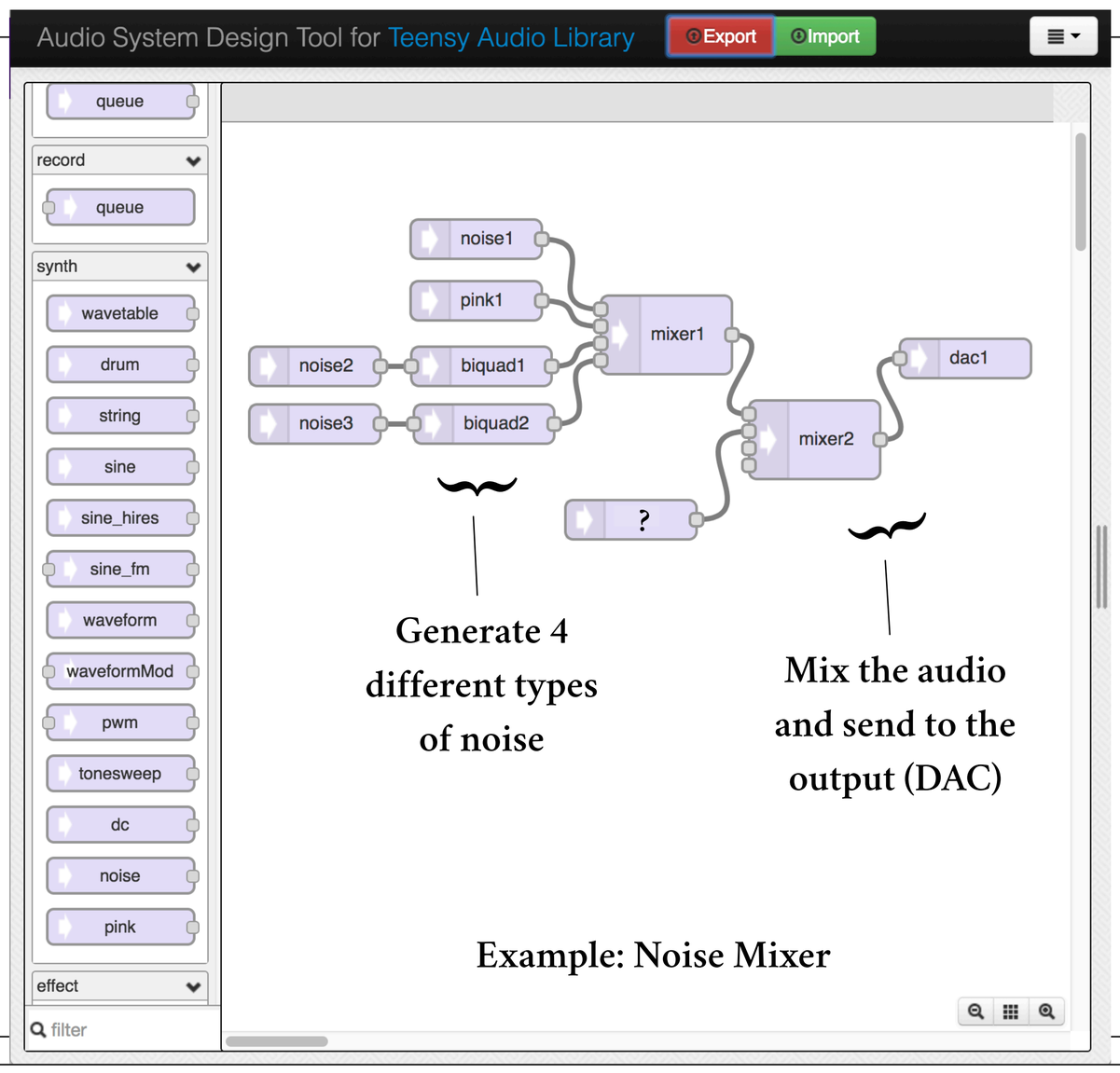
prOk

PJRC



Teensy

A powerful micro controller



Generate 4 different types of noise

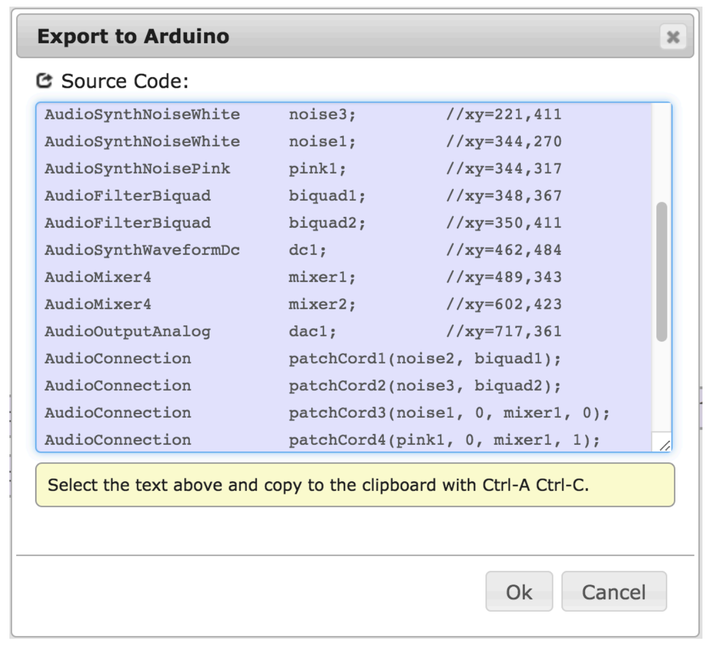
Mix the audio and send to the output (DAC)

Example: Noise Mixer

Step 2 – Create your Audio System.

Teensy provides an intuitive graphical tool for composing an audio system that generates sound. It contains the fundamental components for building many patches, leaving the details of writing the control and configuration up to you.

The tool generates the code that starts your firmware.



Step 3 – Code your firmware in the Arduino IDE.

/* A – First paste generated code. */

```
…
AudioSynthNoiseWhite     noise2;
AudioSynthNoiseWhite     noise3;
AudioSynthNoiseWhite     noise1;
AudioSynthNoisePink      pink1;
…
```

/* B – Add component configuration. */

```
void setup() {
  noise1.amplitude(1.);
  noise2.amplitude(1.);
  noise3.amplitude(1.);
  pink1.amplitude(1.);
  biquad1.setLowpass(0, 100, .707);
  biquad2.setHighpass(0, 10000, .707);
  …
```

/* C – Connect the controls. */

```
void loop() {
  enum NoiseType { White=0, Pink, Brown, Violet };
  int pot = analogRead(NOISE_SELECT_POT);
  int noise = (int)pot / POT_SWEEP_SIZE;

  // For the selected noise type...
  for (int chan = 0; chan < NUM_MIX_CHANs; chan++) {
     // Turn up or down the mixer channel.
    mixer1.gain(chan, 0.9 ? chan == noise : 0.0);
     // Turn on the LED.
    analogWrite(LEDS[chan], ON ? chan == noise : OFF);
  }
```

Of course, you'll need to use a computer and to install both Arduino IDE & Teensy libraries. It's quite easy.

Can we talk? 😺 If you haven't written code yet, it's ok – it can seem really daunting at first. Just know that you can actually build interesting stuff just by doing parts A & B. *promise* 😺 🤝

Step 4 – Test your firmware, safely!

Upload your firmware to the module in the IDE over a standard USB cable. First connect the USB to the module without euro power. If the module powers on, you are risk of damaging your computer. You must cut the USB power trace on the module (see the build docs).

Or, stop power by taping over 4th pin of the USB connector.

Coding Tips

1. Work with a friend! It's less anti-social....
2. Write each step down on paper before coding.
3. Steal, borrow, modify other people's code, but give attribution if you publish.
4. Post *concise* questions to the *right* online forum with *complete* code examples.

Module Options

Arduino-based Module	Audio Output	Input Gates	Output Gates	Input CV	Output CV	$$	Pros	Cons

Thanks for coming by! Instagram: @riverbottom_