

Teensy Guitar Effects Processor



Introduction

The Teensy Guitar Effects Processor (Teensy GEP or just GEP) is a microcontroller based sound effects processor designed for enhancing and modifying electric guitar sound. It uses a Teensy microcontroller paired with a dedicated sound processing chip to provide a multitude of effects.

The base design uses a Teensy 3.2 along with the Teensy Audio Shield from PJRC (pjrc.com) to process stereo audio at a full 16-bit 44.1kHz sample rate. The audio shield supports line-level-in, line-level out, and has an on-board headphone jack. The shield also contains a micro-SD card slot to allow recording and playback of sound. A separate battery powered pre-amp module interfaces most common electric guitar pickups to the line-level input.

The design inspiration for the Teensy GEP is based on a project that appeared in the July 2017 edition of Circuit Cellar magazine by Brian Miller. The original version of the source code (dubbed 1.0) does not seem to have any significant change since the article was published, and no longer compiles with the current version of the Teensy compiler and/or audio libraries. Significant changes to the baseline code were required to a) bring up-to-date with current library versions, 2) match the current compiler versions, and 3) to simplify the code so that my limited brain power could understand how the code worked (or at least how I thought I worked). The current version of the updated code with the original feature set is Teensy_GEP_1.34

Teensy Microcontroller

The baseline design used a Teensy 3.2 which contains a Cortex-M4 processor running at a max of 96 MHz in dual-in-line 28 pin package. It has 256 kB of flash-based memory and 64kB of RAM. It also includes 2048 bytes of on-board non-volatile EEPROM to allow storage of user configuration. See all the specs at PJRC's [Teensy 3.2 page](#).

In the last 2 years, Paul at PJRC has expanded the Teensy line with higher performance processors and increased memory, including the Teensy 3.5, Teensy 3.6, and Teensy 4.0. The Teensy 3.5 and 3.6 maintain the same functionality of the base 28 pins but add an additional 20 pins for increased I/O. The Teensy 4.0 maintains the same 28-pin footprint but alters some of the pins functionality and is therefore not compatible with this version of the Teensy GEP. A comparison of the various features are PJRC's [Teensy Tech Specs](#).

Although not specifically mentioned in the magazine article, it appears some trade-off were made to fit the code into the Teensy 3.2's memory, primarily for the reverb design. Starting with version 1.4, the reverb now utilizes the library's built-in 'freeverb' reverb and requires either a Teensy 3.5 or Teensy 3.6. Also note that the both the 3.5 & 3.6 are physically bigger than the socket and extend beyond the audio shield.

Operating instructions

Basic Operation

Operation of the Teensy GEP is fairly simple: connect power, apply and audio input, and connect the audio out to a guitar amplifier (line in) or use the headphone jack (although the volume is somewhat limited).

Power: The current PCB 1.0 version GEP gets power via Teensy's micro-USB connector. Power may be supplied by a 5V USB source: a PC USB connector, an AC adapter wall module, or even a USB battery pack. Current is typically 150 ma, so even a 1.0 ampHr battery pack should last for hours. Note: there is provision for a PC mount barrel power jack. This has not been populated since a design error has **the center pin negative**, which is opposite of most standard AC adapters. It is fully functional and a jack can be installed if used carefully.

Line-level Audio: The Teensy Audio Shield hosts a 10-pin IDE connector with stereo line-in and line-out that matches the internal PC audio connector standard. The GEP is supplied with two shielded cables that connect to the 10-pin IDE and terminate with 1/8" stereo connectors. Some cables have both cables terminated in 1/8" female connectors, while some have a male 1/8" connector for the audio in, and a 1/8" stereo female connector for audio out.

Headphone Jack: located on the Audio Shield is a 1/8" stereo headphone jack. Audio headphones can be plugged directly into this jack (if space permits).

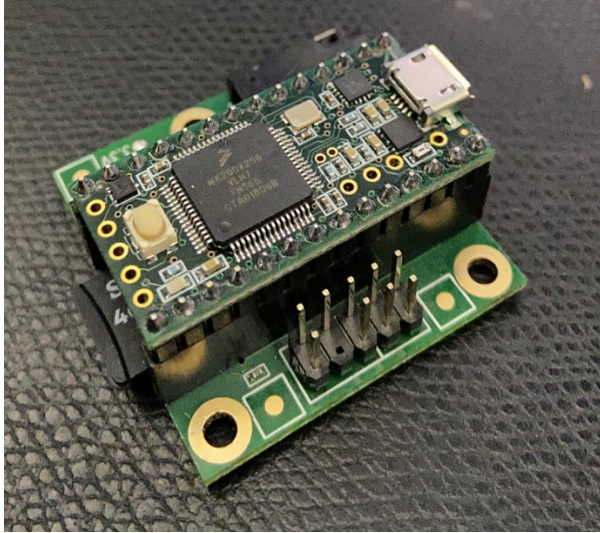


Figure 1 -Teensy 3.2 and Audio Shield line-level connector

Headphone Jack: located on the Audio Shield is a 1/8" stereo headphone jack. Audio headphones can be plugged directly into this jack (if space permits).



Figure 2 - - Teensy Audio Shield Headphone Jack

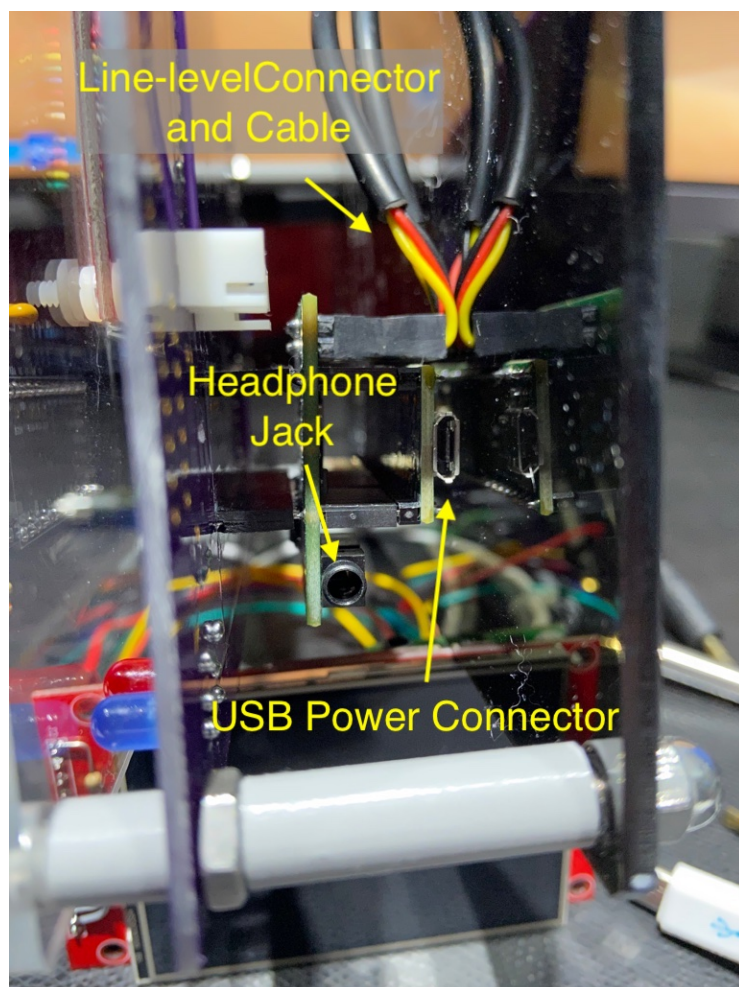


Figure 3 - GEP Side View

Guitar Pre-Amp: A separate battery-powered pre-amplifier is required to match the high-impedance output of most guitar pickups to the line-level input of the GEP. The device is battery powered to reduce and ground-loop noise. The pre-amp uses very low power and a standard alkaline (V battery should last for months). Connect the guitar to the mono 1/4" female jack and the 1/8" male or female connector the the GEP line in jack.

Note: the GEP is currently programmed to left channel of a stereo input which corresponds to the tip of a stereo 1/8" jack. This is easily modified in the software.

INSERT PRE_AMP PHOTO

Front Panel

The front panel user interface of the GEP has two potentiometers, two rotary encoders with built-in pushbuttons, 4 color pushbuttons with corresponding LEDs, and a touch-sensitive TFT LCD display. The front panel is labeled with the default configuration for the controls which can be reconfigured as the user desires.

LCD Display: At power-up, the TFT LCD defaults to a 5-band parametric equalizer. Each band is adjustable, with the selected band highlighted with a red handle. The Param encoder is used to select the desired band, and the Value encoded is used to adjust the gain (above center) or attenuation (below center) of the band. Once the EQ is adjusted, pressing the Save button (press the Value encoder knob down) will save the EQ settings to non-volatile memory (EEPROM) which will be restored at subsequent power ons.

The LCD display can show multiple configuration screens and are cycled by tapping either the left or right side of the display. The right side “increases” to the next screen or the left side “decreases” to the previous display. All screens are stored when the save button is pressed.

Versions 1.3 and 1.5 contains 5 screens:

- 1) Equalizer (EQ)
- 2) Reverb Settings
- 3) Flanger Settings
- 4) Tremolo Settings
- 5) Input Level Adjustment

Mix Pot: Adjusts between the “dry” and “wet” channels. Fully CCW sets dry to unity gain (1.0) and wet to off (1.0). As the knob is rotated CW the wet channel increases to 1.0 while the dry channel decreases proportionally to 50% (0.5). This value was set subjectively and may be adjusted in software.

Wah-Wah Pot: Adjusts the filter values to alter the “wah” effect. Ideally this input should be connected to a pedal to allow adjustment while playing.

Effect Pushbuttons: There are 4 colored pushbuttons with corresponding colored LED to control the 4 standard effects: Reverb, Flanger, Tremolo, and Wah-Wah. Pressing the button enables the effect and lights the matching LED. Pressing again disables the effect. Each effect is independent and multiple effects can be combined.

Software Programming

Introduction

The Teensy GEP is designed to be easily programmed using the Arduino Integrated Development Environment (IDE). The creator of Teensy provides a simple installation app that installs the Teensy compiler and associated libraries directly into the Arduino IDE. All that's required is for the user to select the which Teensy board they're using (3.2, 3.5, or 3.6) and the USB port that the teensy is connected to. Pressing the download button will compile the software, and if successful, load into the Teensy device and begin execution.

Teensy add-on, called Teensyduino, is available at [the PJRC website](#).

Libraries

Libraries are used to add functionality to the Arduino IDE. These can be additional software functionality or drivers to support additional hardware. Several libraries are used with the Teensy GEP to access the Teensy Audio Shield, read the encoders, interface to the LCD Display, and more.

The Teensyduino installer contains many awesome libraries, including the audio shield. If "install all" is selected during install, Teensyduino installs all necessary library files for the GEP

Configuration

The reverb used in the original GEP through version 1.3 accessed registers on the audio shield directly. This requires a minor modification to one of the library files.

- 1) Locate the Arduino Libraries directory
- 2) Copy the PFC8574 directory and files from the GEP source into the libraries directory
- 3)

IN WORK -----

Code Modules

Teensy_Guitar_Effects_150.ino

Hardware.h

Patches.h

Settings.h

Gui.h

Logo.c

Tuner.h

Utils.h

Teensy Audio Tool

