Peter Cat Productions

# GC>MIDI

## User Manual

## Table of Contents

**First off, thanks for buying the GC>MIDI Interface!**

**I hope you'll like it, but first a bit of homework to learn how it all works. Depending on how familiar you are with MIDI setups, it may not take any time at all.**

**GC>MIDI is designed to be plug and play with most MIDI capable devices, but understanding how it works will help you incorporate it with your gear however you want by configuring your receiving devices accordingly.**

# So What is this thing?

**GC>MIDI is a MIDI interface. Basically it takes button presses from your controller and converts them into MIDI signals. Since MIDI is a standard that's been used for decades and is still heavily used today, this means your controller can be used to control a huge number of devices, old or new.**

**GC>MIDI is Open Source and uses a common microcontroller project board, the Arduino Pro-Micro, loaded with custom code to translate controller button presses into MIDI. This means that anyone who knows (or wants to learn) how to work with Arduinos can modify this device however they like, right down to the code.**

**But that's a big subject, so in this manual I will stick to how it works out of the box.**

# MIDI Basics

If you're already familiar with MIDI setups, you will be familiar with most of this; but there is some useful information on how the GC>MIDI communicates with other devices in this section, so take a moment to review.

## So what is MIDI?

To keep it simple, MIDI is a data standard that communicates musical information. So instead of sending an actual audio signal, MIDI sends information in packages called MIDI Messages. There are several types of MIDI Messages, but I will only discuss the ones used by this interface to keep it relevant.

## Note Messages and Velocity

Note Messages are the heart of MIDI. Note On messages communicate when notes should be played. Note Off messages communicate when notes should stop playing.

Velocity is also sent as part of Note On messages. Velocity determines how "hard" a note should be played. Note that not all MIDI Devices can do anything with velocity, but typically the higher the velocity, the louder the note. Note that Velocity cannot be changed while a note is being held (just like you can't change how hard you hit a piano key after you've already hit it).

The GC>MIDI interface sends Note On messages whenever a note trigger is pressed, and sends Note Off messages when that trigger is released (the Controller Mapping section later on will detail what these triggers are and what notes they send).

## Continuous Control (CC)

**A Continuous Control message is used to control parameters outside of playing notes. There are a lot of CC messages (127 of them) but to keep it simple they control the "other" parts of a musical performance. They can control parameters like Modulation, Pan/Balance, Effect Levels, and much more.**

**They are called Continuous Controllers as they are sent very rapidly, allowing for smooth transitions across values for dynamic control.**

**The GC>MIDI Interface sends the Modulation (CC 1) and Pan (CC 10) CCs. Check your receiving device's manual to see if it can receive these messages and how it can be configured to respond to them as this will vary by device.**

## Pitch Bend

**Pitch Bend Messages seem very similar to a CC, but are sent separately as their own special type of message.**

**This is because Pitch Bend doesn't actually send any information about notes or other sound parameters. Instead, the Pitch Bend message tells the receiving device the position of the Pitch Bend Wheel. Note that many devices can respond to Pitch Bend messages even if they don't have a physical Pitch Bend Wheel. See your receiving device's manual for specifics on how it will respond.**

**Due to the very small range of the Gamecube controller joystick, the GC>MIDI interface only sends Pitch Bend messages for half the range of the Pitch Bend Wheel. That means if your synth is configured to bend notes up to two octaves up, pushing the pitch bend control on the GC>MIDI all the way up will only bend the note up 1 octave. Keep this in mind while configuring the Pitch Bend settings on your receiving device.**

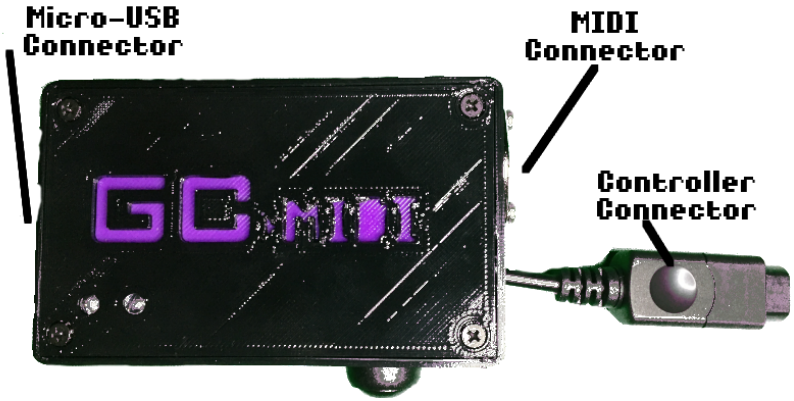
## All Notes Off

**All Notes Off is a CC that tells the receiving device to stop any notes it might currently be playing on that MIDI Channel instead of an individual note like a MIDI Off message.**

**The GC>MIDI Device sends All Notes Off on Channels 1 and 10 whenever the Mode Switch Button is pressed, and whenever the Start button is pressed in Synth Mode. If you get a "stuck note" that won't stop playing, pressing one of these controls should stop it on any device that can receive the All Notes Off (CC 123).**

# Hooking up the GC>MIDI

There are several connectors on the GC>MIDI interface:




## GC Controller Connector

The GC Controller Connector is straightforward. Plug your GC Controller into this port to use the device.

If no controller is detected when the device is powered on, it will check for a controller connection every few seconds until a controller is found. You can tell the device is searching for a controller when the Drum Mode LED is lit, but the Mode Switch Button doesn't change the mode when pressed (more to come on what those are). For now, just know the device won't do anything until a controller is plugged in, but it doesn't really matter what order you connect everything up.

# Micro-USB Connector

**The Micro-USB Connection powers the device and allows for USB MIDI Communication.**

**If you are just using the USB Connection for power, you can connect it with any cable to any USB port to power the device, including portable USB chargers. But long story short, the USB Connector should always be plugged into something or the device won't have any power.**

**If you are using the USB Connection for USB MIDI, first make sure your cable supports Data. Some cheaper cables are just charging cables and won't allow for a data connection, but most USB cables do both. You also need the device you connect to to be a USB Host.**

**A USB Host is usually a computer of some kind. So if you connect the USB Port of your interface to the USB Port of a USB MIDI Keyboard, odds are nothing is going to happen (unless that keyboard happens to be a USB Host). This connection is most useful for connecting to things like DAWs or special devices intended to serve as USB Hosts and route USB MIDI signals.**

# MIDI Connector

**The MIDI Connector on the GC>MIDI interface is a MIDI Out, meaning it only sends MIDI signals out.**

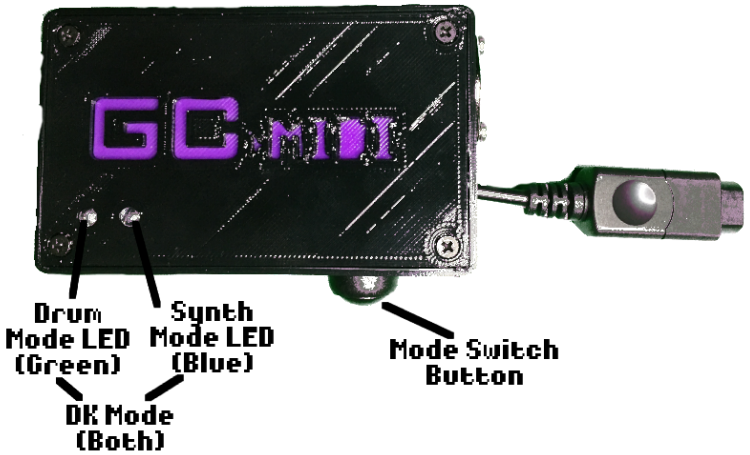
**That means you can connect this port to the MIDI In port on any device that has one, and you should be able to control it.**

**At this point you should be good to go! If the device you are controlling isn't responding the way you'd like, check the manual to see what MIDI signals it's able to receive and any settings you may be able to change to make it respond more the way you'd like.**

**The Controller Maps sections shows what MIDI Notes and CCs the GC>MIDI Interface is sending. Comparing these to your device's manual should help you determine what your device is configured to do when it receives those messages and tell you what customization options are available.**

# Modes

**The GC>MIDI Interface has three modes. Pressing the Mode Switch Button on the side of the device toggles between them.**




## Drum Mode

**Drum Mode is the default mode the device will start up in. You can tell the device is in Drum Mode when the Green LED on the device is lit.**

**Drum Mode sends on Channel 10 and uses the General MIDI Drum Standard to send a selection of MIDI notes that most drum machines, plugins, and synths with percussion functions are set up to respond to out of the box. Most devices will receive percussion messages on Channel 10 by default, but refer to your device's manual to see how that needs to be configured.**

**See the Drum Mode Controller Mapping to see exactly what notes and CCs the various controls send so you can configure your receiving device to respond exactly how you want.**

## Synth Mode

**Synth Mode is designed with a note and control layout intended for use with synthesizers, plugins, or any device that you want to control using the Chromatic Scale. You can tell the device is in Synth Mode when the Blue LED on the device is lit.**

**Synth Mode sends on Channel 1 so that it can be used with most devices, even those that only receive on a few channels. Note that there are some Drum Triggers even in Synth Mode and these Drum Triggers will send on Channel 10.**

**See the Synth Mode Controller Mapping to see exactly what notes and CCs the various controls send so you can configure your receiving device to respond exactly how you want.**

## DK Mode

**DK Mode is a special mode designed for use with the Donkey Conga Bongo Controller. You can tell the device is in DK Mode when both the Green and Blue LEDs on the device are lit. Like Drum Mode, DK mode uses the General MIDI Drum Standard and sends notes on Channel 10.**

**See the Controller Mapping for details, but one thing to mention is the use of Kits in DK mode. Pressing Start will switch between two Kits, one for standard drum sounds and another for percussion sounds that match up to the look of the bongo controller. There is also a version of each mode that disables the clap response to prevent the controller from registering a clap in louder environments or during more... energetic performances.**

# Controller Mapping

## Modulation Control

**As mentioned in the MIDI Basics section, the GC>MIDI Interface sends the Modulation CC (1).**

**In Drum Mode, moving the Joystick either left or right will incrementally raise the value of CC 1 from 0-127. The value will decrease as you bring it closer to the neutral (center) position.**

**In Synth Mode, squeezing the right trigger will incrementally increase from 0-127. There is a small buffer to prevent accidental shifts. Note that the value of 127 is reached just before the trigger "clicks" into a held position. Holding in the "clicked" position will continue to hold the value at 127.**

**The Modulation CC controls a variety of effects and what it will do really depends on the device you are controlling. In most DAWs and in more customizable synths, you can assign CCs to control whatever parameters you want. For these devices just set CC 1 on Channel 10 to control the desired parameter and you should be all set!**

## Pan Control

**In Synth Mode, GC>MIDI sends the Pan CC (10) based on the Left and Right position of the joystick. In the center it sends the center value (64).**

**Moving the stick right and left will change the value to pan right and left (value ranges from 0-127). Note there is a small buffer of movement to prevent accidental shifts.**

## Octave Control

When in Synth Mode, pressing Up or Down on the D Pad will raise or lower the octave. For each octave up or down, add or subtract 12 from the note value shown in the following controller maps.

There is a total range of 5 octaves, 2 up or 2 down from Middle C, roughly the same as a 61 Key Keyboard.

## Transpose Control

When in Synth Mode, pressing Left or Right on the D Pad will raise or lower the transpose value. For each transpose value up or down, the triggered note will be shifted up or down by that value. For example, note 60 (C) with a transpose value of -2 would be 58 (Ab).

The transpose control has a range of 11 up or down (every value possible within the same octave). The Octave control can be used in combination with the Transpose control to shift note values by more than 11.

## Reset

In Synth Mode, pressing the Start button will reset the octave back to neutral (Middle C), and also set the Transpose control to zero (no transposition). This reset also stops any currently sounding notes on both Channel 1 and 10 so it can be used as a “MIDI Panic” in the event a note gets stuck from the Synth or the Drum Triggers in Synth Mode. This is also useful to prevent getting “lost” as there is no visual indicator of the current octave and transposition values.

# Drum Mode

GC>MIDI

Standard Drum Mapping
(Channel 10)

# Synth Mode

## GC>MIDI

### Standard Synth Mapping (Channel 1)*

Up/Down–Pitch Bend
R/L–Pan (CC10)

Up–Octave Up
Down–Octave Down
R–Transpose Up
L–Transpose Down

Reset Octave, Transposition, All Notes Off

A–Kick (36)*
B–Snare (38)*
Y–Closed Hat (42)*
X–Clap (39)*

Down–C (B)
Down+L–D (Db)
L–E (Eb)
Up+L–F (E)
Up–G (Gb)
Up+R–A (Ab)
R–B (Bb)
Down+R–C+ (B+)

() Values when L is held

Flat–Note values in ()

Modulation (CC1)

Velocity

* Asterisked values on Channel 10

# GC>MIDI

## DK Drum Mode (Channel 10)

Press Start to cycle kits:
-Drum Kit
-Drum Kit (Clap Disabled)
-Bongo Kit
-Bongo Kit (Clap Disabled)

### Drum Kit


Clap-Clap(39)*

### Bongo Kit


Clap-Clap(39)*

* Volume determines velocity

# MIDI Implementation

| MIDI Message | Channel | Value | Notes |
|---|---|---|---|
| Note On/Off | 1 (Synth Mode), 10 (Drum Mode, DK Mode, Drum Triggers in Synth Mode) | 36-95 | See Controller Mapping for specific triggers |
| Velocity | 1 (Synth Mode), 10 (Drum Mode) | 64-127 (Synth), 25-127 (Drum) | Minimum Velocity due to limited control range |
| Pitch Bend | 1 | 4096-12288 | Half the full pitch bend range of the receiving device, Synth Mode only |
| CC 1 (Modulation) | 1 (Synth Mode), 10 (Drum Mode) | 1-127 | Synth Mode and Drum Mode |
| CC 10 (Pan) | 1 | 1-127 | Sends 64 (no pan) in Neutral position. |
| CC 123 (All Notes Off) | 1, 10 | | Sent on both channels whenever Mode Switch Button is pressed, or when Start is pressed in Synth Mode |

Configuration is the same for USB and Traditional 5pin MIDI. All messages are sent via USB and 5pin connection simultaneously.

This device is not a USB Host. It must be connected to a USB Host to utilize USB MIDI. Check your receiving device's documentation to determine if it can be used as a USB Host, but most DAW/PC applications will serve as a Host.