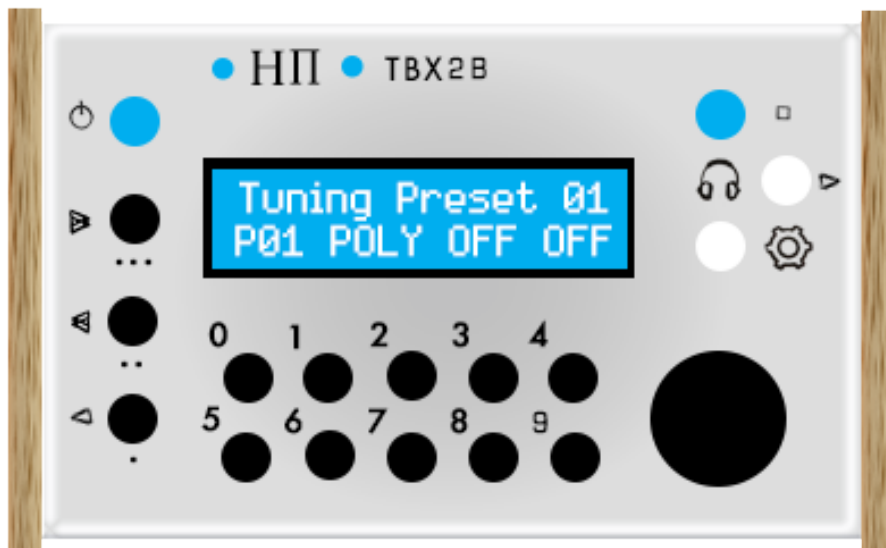


TBX2b



HΠ INSTRUMENTS

Aaron Andrew Hunt

Changes from Previous Documentation	4
<i>v1.0 - October 20, 2020</i>	4
Precautions	5
<i>Location</i>	5
<i>Power</i>	5
<i>Handling</i>	5
<i>Cleaning</i>	5
<i>FCC REGULATION</i>	5
<i>DISPOSAL</i>	5
Introduction	6
<i>Improvements</i>	6
<i>Tuning</i>	6
<i>MIDI I/O</i>	6
<i>Manufacture</i>	6
<i>Using This Documentation</i>	7
Startup Options	8
<i>Firmware Update Startup</i>	8
<i>Factory Reset Startup</i>	8
1. User Interface	9
<i>ON / OFF Switch</i>	9
<i>Display</i>	9
<i>Jog Dial</i>	9
<i>Function Buttons</i>	10
<i>Preset Buttons</i>	11
<i>Settings (Set & Exit) Button</i>	11
<i>Browse (Enter) Button</i>	11
<i>THRU (Cancel) Button</i>	12
<i>MIDI Input Switch (Bottom of Unit)</i>	13
<i>USB Status LEDs (Bottom of Unit)</i>	13
2. Inputs & Outputs	14
<i>Dual Pedals Input</i>	14

<i>MIDI IN & OUT</i>	15
<i>USB Host Input</i>	15
<i>USB-MIDI Interface</i>	16
<i>MIDI THRU</i>	16
<i>Power Input</i>	17
3. Setups	18
<i>Basic MIDI Signal Path</i>	18
<i>USB Signal Path</i>	18
<i>MIDI DIN Signal Path</i>	19
<i>Converting between USB & MIDI</i>	19
<i>Keyboard Controllers vs. Keyboard Synthesizers</i>	19
<i>MIDI DIN Keyboard Synthesizers</i>	19
<i>USB Keyboard Synthesizers</i>	20
<i>Separate Keyboard & Synthesizer</i>	20
<i>Keyboard Controller & Computer</i>	20
<i>Using TBX2b with U-PLEX</i>	21
4. Global Settings	22
<i>Target Device Pitch Bend Range</i>	22
<i>Transposition</i>	22
<i>Local Control OFF</i>	23
<i>How to turn Local Control back ON</i>	23
<i>CC Retransmission</i>	24
<i>MTS Device ID</i>	24
<i>Display Brightness</i>	24
<i>Bank Select Format</i>	24
<i>Pitch Bend Timing</i>	25
<i>Sysex Retransmission</i>	25
<i>USR Data</i>	26
5. Preset MODES	27
<i>Pitch Bend 2048 vs. Sysex 128</i>	27
<i>POLY</i>	27
<i>MONO</i>	28
<i>MTS</i>	28

<i>USR</i>	29
<i>A Note About MPE</i>	29
6. Presets	31
<i>Preset Programming</i>	31
<i>Recalling a Preset</i>	35
7. Using TBX2b with Universal Tuning Editor (UTE)	36
<i>Download, Install & Authorise UTE</i>	36
<i>Connecting TBX2b</i>	36
<i>Programming a Tuning</i>	36
<i>Batch Programming Tunings</i>	37
<i>Programming a Preset</i>	38
<i>Setting Global Parameters</i>	39
<i>Setting USR Data</i>	40
<i>USR Data Example 1</i>	41
<i>USR Data Example 2</i>	41
<i>Backing up & Restoring Data (.udevicebackup)</i>	42
<i>Updating Firmware (.ubootsyx)</i>	43
<i>UTE + TBX2b Troubleshooting</i>	43
Troubleshooting	45
Appendix – UPLEX Modification	47

Changes from Previous Documentation

Here are lists of changes for each version of this documentation.

Please report typos or problems with this text via email to hpiinstruments@zentral.zone

v1.0 - October 20, 2020

- **First release:** This is the documentation for TBX2b and is not to be confused with documentation for its predecessor: TBX2. (The devices are not identical).

Precautions

Location

Using the unit in the following locations can result in malfunction and damage to the unit.

- In prolonged direct sunlight
- In extreme temperature or humidity
- In excessively dusty or dirty locations
- In excessive vibration
- In close proximity to magnetic fields

Power

A 12-Volt 0.4-Amp international switching DC power supply has been shipped with this unit. A DC power supply of more than 12 Volts will destroy the unit. A DC power supply of less than 9 Volts will not cause damage, but the unit will not function. The possible operating range of DC power supplies for this unit are: between 9 and 12 Volts, 0.4 Amps to any higher number of Amps, barrel negative, pin positive. An AC power supply will destroy the unit.

Handling

Do not physically abuse the unit. Do not use excessive force to plug or unplug any cable to or from any port of this unit, especially the power port. Do not apply excessive force to the buttons, switches, knob or display.

Cleaning

Clean the housing of the unit only with a clean, dry cloth. Never use liquid cleaners or flammable polishes.

FCC REGULATION



This device complies with Part 15 of FCC Rules (USA). Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

DISPOSAL



Disposal of this product, manual, package or battery must be done in an approved manner (EU). Do not discard this product, manual, package or battery along with ordinary household waste. Disposing in the correct manner will prevent harm to human health and potential damage to the environment. The correct method of disposal depends on the applicable laws and regulations in your locality.

Introduction

TBX2b is an improved second-generation microtonal MIDI retuning device which receives standard (non-microtonal) input from a MIDI controller and sends microtonal output to a MIDI synthesizer according to stored tuning tables. Tunings are uploaded to the device using software called *Universal Tuning Editor* (license included with purchase of TBX2b).

Improvements

Here are the ways in which TBX2b improves upon its predecessor, TBX2.

1. **Built-in USB-MIDI Interface** – avoids external MIDI Interface incompatibility issues.
2. **MIDI Input Switch** – secures the integrity of MIDI Input data.
3. **No Battery** – avoids problems with data integrity due to low power.
4. **Barrel Power Connector** – avoids confusion with other USB connectors.
5. **More Memory Headroom** – allows for easier firmware changes and expansion.
6. **Updated Firmware** – new features and improved functionality.

Tuning

TBX2b features MIDI tuning precision better than 0.006 cents. Four preset modes allow retuning GM (multichannel) synths, monophonic synths, MTS synths, and even proprietary sysex synths. It offers storage space for 8,192 tuning tables of 128 notes each, for a total of over a million pitches. 40 tuning presets can be navigated hands-free using dual pedal input.

MIDI I/O

The unit features MIDI DIN, built-in USB-MIDI Interface, and USB-MIDI Host input and output. Each of the 16 MIDI input channels can be assigned to a unique tuning table in POLY and MONO modes, for up to 2048 microtonal pitches output, providing full compatibility with microtonal controllers like the Tonal Plexus, microzone, and Terpstra keyboards, as well as more conventional multichannel controllers like MIDI guitars and MIDI organs. MTS and USR modes allow 128 pitches output.

Manufacture

TBX1, the predecessor to TBX2 and TBX2b, was in manual production for ten years. For the latter half of that time, TBX2 was in development. TBX2b is released two years after TBX2. This device is not mass produced. TBX2b begins its life with a run of 25 units produced in Bulgaria using a combination of state-of-the-art machine and hand assembly. Each unit is individually “brought to life” and given a unique and slightly imperfect hand-made aluminium and bamboo enclosure ca. 142 x 84 x 42 mm.

Using This Documentation

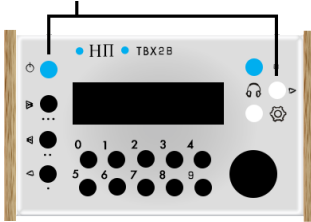
Please read through this text completely. It is written to be read from beginning to end, and also contains cross referencing for normal use. Detailed information concerning MIDI implementation is found in the separate *TBX2b MIDI Reference* document.

Please report typos or problems with this text via email to hpiinstruments@zentral.zone

Startup Options

The following options are available when powering up the unit.

Firmware Update Startup

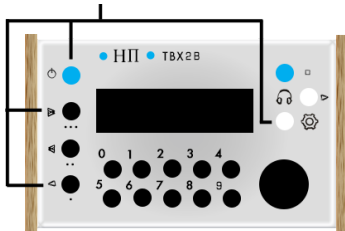


Press the rightmost white button and hold it down while you turn on the unit with the blue power switch on the left. This tells the unit “we are ready to update the firmware”. The display will read as shown below.



Firmware updates are sent using [Universal Tuning Editor](#) (see Chapter 7).

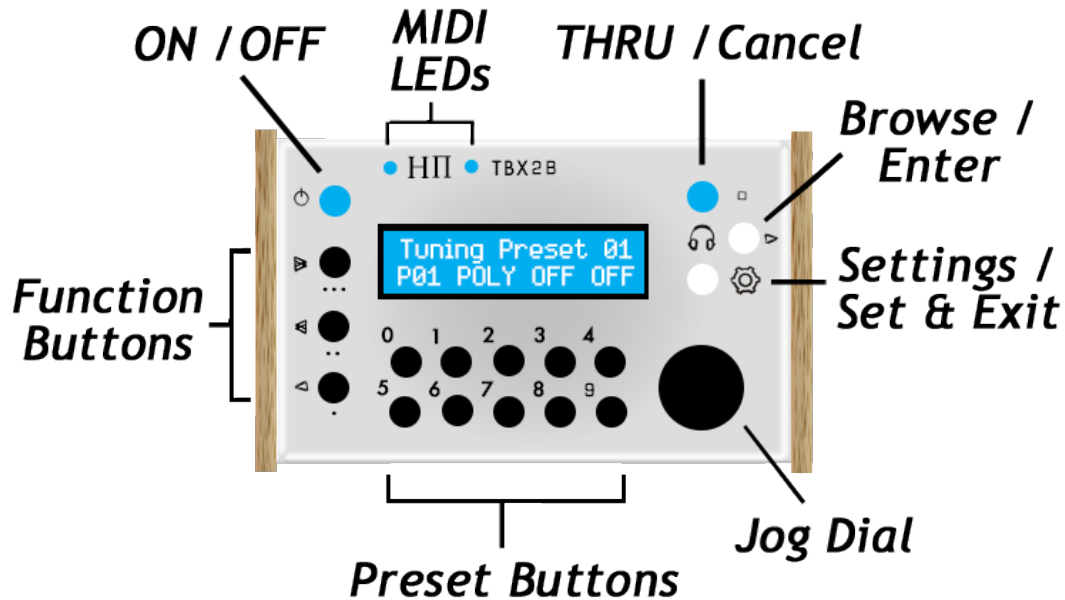
Factory Reset Startup



This 4-button startup combination tells the the unit “reset to initial factory state”. All tuning data will be lost and all settings will be overwritten with factory defaults. It is made intentionally awkward so that the function is not accidentally used.

1. User Interface

TBX2b has a simple interface consisting of a display, two LEDs, two switches, fifteen buttons and a jog dial.



ON / OFF Switch



When the unit is plugged into external power, latching the power switch into the ON position will power the unit.

Display



The top line of the display shows the 16-character name of the currently selected preset. The bottom line shows the number of the preset (here, P01), the mode (here, POLY), and the bank and patch numbers (here both set to OFF). Display backlight brightness defaults to 50% and can be changed using the settings button.

Jog Dial









Value selection is done using the jog dial, which turns continuously in both directions, so that values “start over” when reaching maximum or minimum. The knob also acts as a button. Pushing the knob as a button during normal operation will begin programming the currently selected preset, a process involving many steps outlined in [Chapter 6](#). During programming, pushing the knob enters a selected value. See also [Chapter 4](#) which outlines the process of programming Global Settings using the jog dial.

Function Buttons









There are 3 function buttons on the left side of the unit, labeled up from the bottom with arrows and dots respectively. The dots denote the function number counting from the bottom up as Function 1, 2, and 3, which we will refer to as **F1**, **F2**, and **F3**. These buttons are modifiers which work in combination with other buttons (similar to SHIFT, ALT, and CTRL on a computer keyboard). There are three modifying functions for each button: 1. **PRESET**, 2. **TURN**, 3. **ENTER**. Also, when two function buttons are pressed together, some special **INFORMATION** is shown on the display.

1. **PRESET** - when pressed in combination with a preset button, the function buttons add 10, 20, or 30 to the preset number. The number of dots under each function button shows how many times ten will be added.







 **F3 + PRESET** = select preset number + 30 [ = $+(10 \times 3)$]
 **F2 + PRESET** = select preset number + 20 [ = $+(10 \times 2)$]
 **F1 + PRESET** = select preset number + 10 [ = $+(10 \times 1)$]

Examples: holding down **F3** and pressing Preset 9 activates preset 39
 holding down **F2** and pressing Preset 2 activates preset 22
 holding down **F1** and pressing Preset 3 activates preset 13

2. **TURN** - when held down and the jog dial is turned during programming or browsing, value changes are altered by powers of ten. The number of dots under each function button shows which column in the number (in other words, which power of ten) will change.

 **F3 + jog dial TURN** = +/- 1000 [ = $\pm 10^3$]
 **F2 + jog dial TURN** = +/- 100 [ = $\pm 10^2$]
 **F1 + jog dial TURN** = +/- 10 [ = $\pm 10^1$]

3. **ENTER** - when function buttons are held down and the jog dial (or alternate enter button) is pressed during programming, the functions have to do with navigating forward and back, as shown by the arrow icons to the left of each button. Double-arrows denote moving forward or back between programming pages, and the single arrow denotes moving back a step on the current page.

 **F3 + ENTER** = GO FORWARD to the NEXT PAGE 
 **F2 + ENTER** = GO BACK to the PREVIOUS PAGE 
 **F1 + ENTER** = STEP BACK on THIS PAGE 

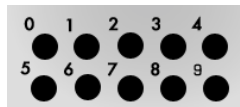
Note that the missing function to STEP FORWARD on THIS PAGE is handled by the **Browsing & Enter Button** (see below).

4. INFORMATION - when two function buttons are pressed together, special information is shown on the display.

1. Pressing **F2** & **F3** together shows the firmware version.



Preset Buttons



TBX2b stores 40 presets, numbered 0-39. Pressing a preset button 0-9 recalls the preset of that number. Holding down one of the function buttons at the same time as pressing a preset button adds 10, 20 or 30 to each number to recall presets between 10 and 39 (see **Function Buttons** above). Presets and preset programming are the subjects of **Chapter 6**.

Settings (Set & Exit) Button



Pressing this button begins programming global system settings, a process consisting of several steps outlined in **Chapter 4**. As of firmware 1.9.9, during programming of any kind, pressing this button will save any changes made and exit the programming process before it is finished. The connection between **SETTINGS** and *saving (setting)* & exiting should make this easy to remember.

Browse (Enter) Button



The two symbols to the left and the right of this button denote its two different functions. The headphones symbol to the left denotes a listening / browsing function, which is a simple way to try out any of the different tunings in memory. When pressed during normal operation, a special *browsing behaviour* begins and the display changes to show temporarily selected tunings for quick auditioning.




When entering browsing behaviour, the *MODE*, *Bank*, and *Patch* of the currently selected preset are respected, and the tuning table assigned to MIDI Input Channel 1 of the current preset is applied to all input channels (see **Chapter 6** for an explanation of presets).


During browsing behaviour, all input channels are assigned to the same tuning for easy auditioning. Turning the jog dial selects different tunings. Pressing any of the function buttons while turning the knob navigates tunings by tens, hundreds, or thousands (see **2. TURN** under *Function Buttons* above).

Pressing the browse button or any preset button exits the browsing state without altering any preset tunings, and returns the unit to normal functioning. *Pressing the jog dial applies the tuning to all MIDI Input Channels of the current preset*, then exits browsing and returns the unit to normal functioning.

Pressing a preset button during browsing assigns the currently auditioned tuning to all MIDI Input Channels of that preset and then exits browsing.

 The right-facing arrow to the right of this button denotes its behaviour during programming, belonging to the set of arrow symbols used for the function buttons navigation behaviours (see **3. ENTER** under *Function Buttons* above). During programming, the behaviour of this button is the same as the jog dial button, to **ENTER** a value or STEP FORWARD on THIS PAGE. In this sense the button acts as a fourth navigation button, only during programming (see *Chapter 4 Global Settings* and *5 Presets*).

THRU (Cancel) Button

 You may notice that the small box next to this switch is the same symbol used to mark MIDI THRU on the port side of the unit. This is because when engaged, all MIDI OUT ports essentially become MIDI THRU ports, and TBX2b passes all incoming MIDI data direct to MIDI OUT without processing the data in any way.

When the THRU button is pressed during normal functioning, the display changes to:



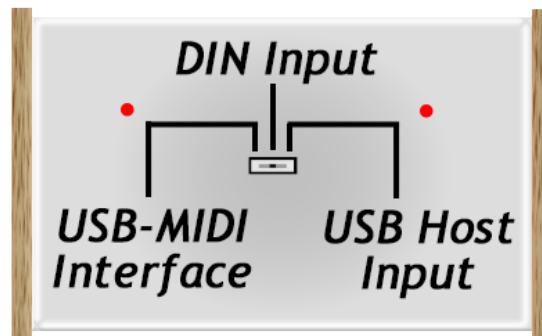
```
BYPASSING:
ALL
```

During programming of any kind, this button acts as a *Cancel* button. Any changes made during programming will be cancelled and the unit will return to normal functioning. The similarity between *BYPASS* and *exit/Cancel* concepts should make this easy to remember.

MIDI Input Switch (Bottom of Unit)

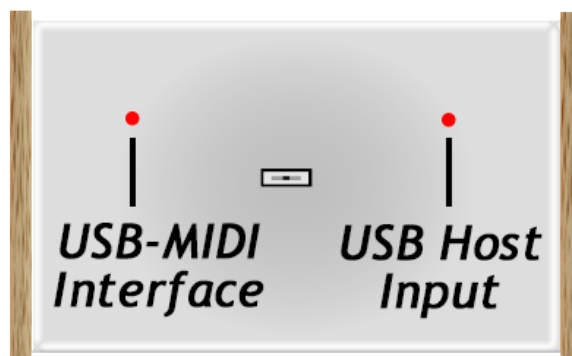
TBX2b provides several different MIDI Inputs and outputs. The unit sends output data to all MIDI outputs simultaneously, but it cannot listen to all MIDI Inputs simultaneously. A switch on the bottom of the unit allows you to select which MIDI Input you want to use. The position of the switch corresponds to the physical placement of the selected MIDI input jack. The left position corresponds to the USB-MIDI Interface Input, the center position to the MIDI-DIN Input, and the right position to the USB-Host Input. The switch can be repositioned while the unit is on, but this should not be done while MIDI data is being received.

3-Position Input Switch



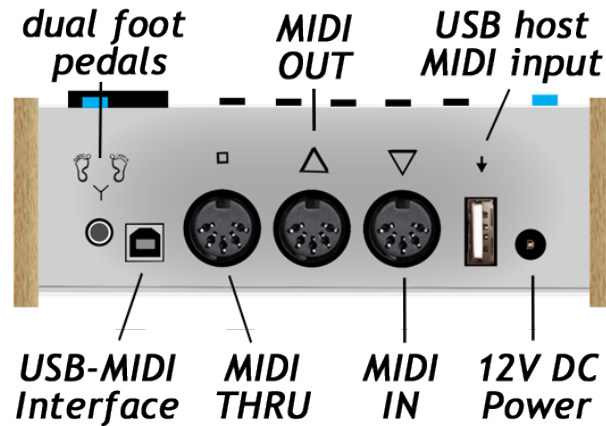
USB Status LEDs (Bottom of Unit)

When the unit is plugged in and turned on, the two LEDs on the bottom show the connection status of the two USB Jacks. When an LED is on, the corresponding USB connection is receiving power. An illuminated LED does not however guarantee USB communication. The LED will flash on and off when MIDI is transferred over the connection.



2. Inputs & Outputs

The port side of the unit has seven sockets for input and output.



Dual Pedals Input



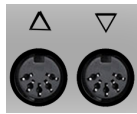
This is a *split stereo input* for two pedals (the two footprints above this jack are meant to remind you of this), although you may also connect only one pedal. The pedals are used to navigate forward and backward through 40 presets (see [Chapter 6](#)). Values “start over” when reaching maximum or minimum. When only one pedal is connected, the navigation will only step forward from one preset to the next.

The type of pedal expected is the piano sustain variety, which is a simple switch. Such pedals normally have a *mono* 1/4" plug. To connect two separate pedals, adapters are needed to combine the two mono 1/4" outputs into one 1/4" stereo signal, then to convert the 1/4" stereo output to the TBX2b 1/8" input. Some dual pedals are available which have a stereo 1/4" plug, such as *Studiologic (Fatar) VFP-2-15* or *Kawai F-20* shown below. In this case, only the 1/4" stereo to 1/8" stereo adapter is needed.



Other dual pedals have two plugs, which may be easier to work with because having separate plugs allows you to assign left and right pedals to your preferred behaviour by switching the left or right channels. In any case, TBX2b senses the polarity of each switch at startup and considers that state as open. Some pedals have normally closed switches, so if a pedal seems to behave “backwards” and there is no polarity switch on the pedal, start up TBX2b with the pedal held down.

MIDI IN & OUT



The two DIN5 sockets marked with arrowhead triangles pointing IN and OUT are for connecting standard MIDI devices to TBX2b. MIDI DIN5 ports became a standard in 1983 and are still used on the majority of devices, although many manufacturers now primarily design for USB (see *USB Host Input* below), so these ports can be used with almost all older devices but not all newer devices. Examples of connecting DIN5 devices to TBX2b are in *Chapter 3*. The MIDI OUT connector will always send MIDI OUT, but the MIDI IN connector is only active when the MIDI Input Switch on the bottom of the unit is set to the middle position, so before using this input, move the MIDI Input Switch to the correct position (see *MIDI Input Switch* above).

Move the *MIDI Input Switch* on the bottom of the unit to the MIDI-DIN Input position.

USB Host Input



TBX2b includes a USB Host input for connecting to a controller that does not have MIDI DIN5 output but instead has MIDI-over-USB output, which has become more common in recent years. The arrow symbol above the USB port is there to remind you that this is a *USB Host Input* and does *not* provide a USB connection between TBX2b and your computer.

The USB-Host input port is NOT used to connect TBX2b to your computer!

Before using this input, you must move the MIDI Input Switch on the bottom of the unit to the correct position (see *MIDI Input Switch* above).

Move the *MIDI Input Switch* on the bottom of the unit to the USB-Host Input position.

The USB-Host port serves two-way traffic both as MIDI IN and MIDI OUT for compatible USB devices. USB device requirements are as follows:

1. **Must be class-compliant ... if it requires a 3rd party driver, it will not work**
2. **Must register as a single device ... if it registers as a “USB device tree”, it will not work**
3. **USB hubs not allowed ... if there is a port on it for other USB devices, it will not work**
4. **“Port-powered” devices use power from TBX2b ... not all devices will work**

To clarify point 4 above: TBX2b is built to the USB specification to provide power to connected devices having no external power input, but in some cases a device may request more power than TBX2b can supply, and in those cases the device will not work.

USB-MIDI Interface



TBX2b is programmed over MIDI using System Exclusive MIDI messages (known as “sysex”). Uploading tunings, presets and settings, backing up or restoring your data, and sending firmware updates to TBX2b requires sending sysex messages. Although this can also be done using an external MIDI Interface connected to the MIDI DIN connectors on the unit, doing so is problematic because many MIDI-Interfaces do not handle sysex messages properly. For this reason TBX2b provides a built-in USB-MIDI interface (and this is one of the major differences between TBX2 and TBX2b). Before using this connection, you must move the MIDI Input Switch on the bottom of the unit to the correct position (see *MIDI Input Switch* above).

Move the *MIDI Input Switch* on the bottom of the unit to the USB-MIDI Interface position.

When using this connection, please also be aware of the following:

1. Plugging it into a USB-Hub will not damage anything but is likely to result in failure to communicate, even if the device is recognised. To avoid potential problems, plug the USB-MIDI Interface directly into a USB jack on the host computer.
2. Regardless of whether the unit is plugged into power or turned on, the USB-MIDI Interface will be recognised as a MIDI device named “TBX2b” when plugged into a host computer. As one might guess, the unit must still be plugged into power and turned on in order to work.
3. The LED aligned with the USB-MIDI jack on the bottom of the unit will light up when the device is connected to a host computer, but this LED does not guarantee successful communication with the host (see 1-3 above). The LED will flash on and off when messages are sent or received over the USB-MIDI Interface.


Connect the USB-MIDI Interface to your Mac or Windows PC with a standard USB cable and run *UTE* (see *Chapter 7*).

MIDI THRU



Regardless of which MIDI Input port is selected via the MIDI Input Switch, all MIDI data coming into TBX2b is passed on to MIDI THRU unprocessed. This is a convenience port allowing you to send data from your controller both to TBX2b and to another destination without having to buy a separate MIDI splitter / MIDI thru box.

Power Input

 Whereas TBX2 used a micro-USB connector for power, TBX2b uses a standard barrel jack connector. This choice was made to eliminate confusion resulting from too many different USB connectors on the unit. The international switching power supply shipped with the unit includes interchangeable plugs for worldwide compatibility. Should you ever need to replace the power supply, see ***Power*** under ***Precautions*** at the start of this document.

3. Setups

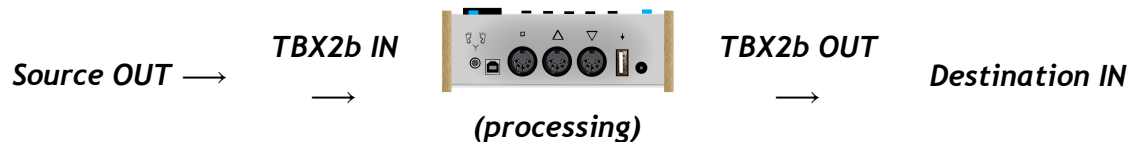
There are many possible combinations for connecting inputs and outputs when using TBX2b. In this chapter we outline the concepts that you need to figure out the best setup for your gear, and we take a look at some example setups.

Basic MIDI Signal Path

TBX2b is placed between a MIDI source (usually a keyboard) and a MIDI destination (usually a synthesizer).



A *Source* is any MIDI controller which produces MIDI Note ON and Note OFF messages (usually a keyboard, but can be some other type of device like a wind controller for example). A *Destination* is any MIDI device which receives MIDI Note ON and OFF messages and produces sound in response (usually this is called a synthesizer). Below, this model is expanded to explicitly show inputs, outputs, and processing.



The above model is referred to in this chapter as the “*MIDI signal path*”. The input and output connections vary with the type of cable used, USB or MIDI DIN. So first we take a look at how these two approaches differ.

USB Signal Path

A single USB cable provides bidirectional communication, so when the source and destination are the same, as in a keyboard synthesizer, only one USB cable is needed to complete the MIDI signal path.



If the destination is not the same as the source, another cable is needed. If a USB device is used as the source, you can use the MIDI DIN OUT port or the USB-MIDI Interface on TBX2b as the destination.

MIDI DIN Signal Path

Unlike USB cables, MIDI cables connect data flowing in one direction only (not because of the cable, but because of the ports). When traffic is needed in two directions, two cables must be used. The rule is simple: *MIDI OUT* from device 1 gets connected to *MIDI IN* of device 2.



Converting between USB & MIDI

As the name suggests, a USB-MIDI Interface converts MIDI to USB and vice versa. Since TBX2b has both MIDI DIN and USB-Host ports, an external USB-MIDI Interface can be connected to either of these ports to expand connection possibilities. For example, connecting a USB-MIDI interface to the USB Host port gives you a second set of MIDI DIN ports for input and output. Doing so provides a way to connect sources and destinations having MIDI DIN ports using only one port (the USB Host port) on TBX2b, which may be more convenient.

Keyboard Controllers vs. Keyboard Synthesizers

For those less experienced with MIDI, it is important to point out that a keyboard which does not provide any internal sounds is not called a keyboard synthesizer, it is called a *keyboard controller*. A keyboard controller can only function as a *Source* in the MIDI Signal Path (see above). A keyboard which has internal sounds is actually two devices in one, and is called a *keyboard synthesizer*, which can act as both a *Source* and a *Destination* in the MIDI Signal Path. TBX2b is useful only when you have both a *Source* and a *Destination*. If all you have is a keyboard controller (*Source*), you need to get another device to use as the *Destination* (normally a *synthesizer*).

MIDI DIN Keyboard Synthesizers

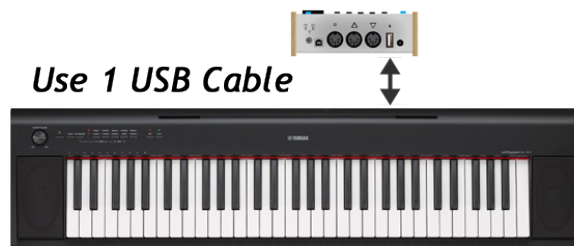
When a MIDI keyboard is also itself a synthesizer, it is two MIDI devices in one, connected internally through *Local Control*. MIDI has a message for turning *Local Control ON* or *OFF*, and TBX2b turns it *OFF* by default (the behaviour is selectable, see **Chapter 4**). When *Local Control* is *ON*, the keyboard and synthesizer are internally connected, and playing keys on the keyboard generates sound in the synthesizer. When *Local Control* is *OFF*, playing the keyboard results in no sound from the synthesizer, because the keyboard is no longer internally connected to the synthesizer. In a typical TBX2b setup involving a keyboard synth, the source and destination are the same device, with *Local Control OFF*. For example, we connect TBX2b to a Yamaha PSR-E313, an older keyboard synth having only MIDI DIN ports and no USB port.



In the above diagram, MIDI OUT from the source is connected to MIDI IN on TBX2b, and MIDI OUT from TBX2b is connected back to to MIDI IN on the source, which is also the destination. TBX2b sends *Local Control OFF* to disconnect the source from the destination. If *Local Control* is not turned OFF, two notes will sound for every key pressed.

USB Keyboard Synthesizers

A setup involving a USB keyboard synth will be the same as one involving a MIDI DIN keyboard synth above, except that only one USB cable is needed. For example, we connect TBX2b to a Yamaha NP-12, a newer keyboard having only a USB OUT (Type-B) port and no MIDI DIN ports.



TBX2b receives MIDI from the keyboard (source), turns *Local Control OFF* to disconnect the source from the destination (so that only one pitch sounds and not two), and sends microtonal output to the destination. The NP-12 must be plugged into external power.

Separate Keyboard & Synthesizer

The possibilities start to multiply when the source and the destination are not the same device and therefore may have different ports (USB or MIDI DIN). If you understand how these ports work, as described above, then you should be able to figure out how to arrange your setup. When the source and destination are not the same device, TBX2b does not need to send *Local Control OFF*, so you should disable that function in the Global Settings (see *Chapter 4*).

Keyboard Controller & Computer

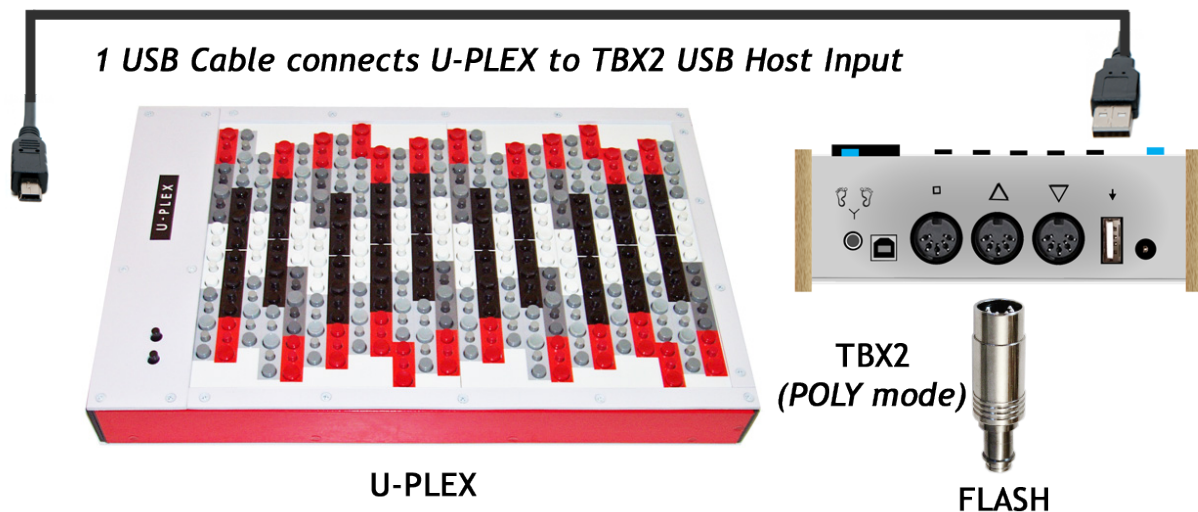
When the source is a keyboard controller and the destination is a software synthesizer running on a computer, the possibilities multiply even further.



Once the MIDI data arrives at the computer, it must be routed correctly to the final destination software, and that software must be configured properly to receive the data (see [Chapter 5](#)).


Using TBX2b with U-PLEX

U-PLEX was designed as a USB device to be used with a computer, but because TBX2b features a USB Host Input, U-PLEX can now connect to TBX2b to function without a computer. In other words, TBX2b turns U-PLEX into a standalone microtonal controller. Simply connect a destination synth to complete the MIDI signal path. Our [FLASH](#) microtonal synthesizer is an excellent choice for the final output.



Note that the octave-switching buttons on U-PLEX will have no effect when using the keyboard with TBX2b. A small modification to the U-PLEX main PCB is also needed to make the MIDI signal path work properly (see the [Appendix](#)).

4. Global Settings

 TBX2b has seven global parameters which affect output behaviour for all presets. Press the **Settings** button to enter Global Settings programming mode. Pressing this button a second time will save any changes made and exit, and pressing the **THRU/Cancel** button will cancel any changes made and exit. Pressing a preset button will also exit programming, and activate the given preset. Push the knob or press the **Browse/Enter** button to enter a value, referred to as pressing “enter”.

Target Device Pitch Bend Range

TBX2b sends pitch bend range setting messages to destination devices when pitch bend is used for microtuning. Some devices may require a fix pitch bend range, or you may want a particular range for performance purposes. The default value is ± 01 .



01 flashes, and turning the knob navigates values between 01 and 96. Press enter to select and exit.

Unlike TBX1, tuning tables stored in TBX2b remain independent of this value. TBX2b processes each tuning table data according to the value when loading a preset. When the value changes, the active preset is reloaded to reflect the change upon exiting programming.

Transposition

You may want to transpose a scale that you have uploaded to TBX2b, without having to change it at the computer and upload a new scale. TBX2b provides a MIDI Transpose function for this purpose.



+00 flashes, and turning the knob navigates values between -64 and +63. Press the knob to select and exit. These values are in MIDI Notes, so transposition is done in twelve-tone equal temperament. For example, a value of +01 will transpose all notes output up by one MIDI note, which is the same as an equal tempered halfstep. A value of -12 will transpose all notes down one octave.

TBX2b processes the tuning table notes according to this value when loading a preset. When this value changes, the active preset is reloaded to reflect the change upon exiting programming.

Local Control OFF

Local Control is the name given to the connection between a keyboard and its internal synthesizer, and this connection needs to be turned OFF when TBX2b is connected to the keyboard, or you will hear two notes when you play one key. In POLY and MONO modes, the unit sends a *Local Control OFF* message by default before sending other messages. This is a convenience to avoid having to turn local OFF manually every time you use the unit. Here you can select the behaviour for this parameter. In some cases you may not want *Local Control OFF* to be sent. You may want the unit only to send this message once, when it boots, and not send it again with any preset. You may also want to send *Local Control ON* at startup, instead of OFF (see below).



STARTUP+PRESET flashes, and turning the knob cycles between the following options:

STARTUP+PRESET

STARTUP ONLY

DO NOT SEND

StartON+PreOFF

The default is the first option, STARTUP+PRESET which sends *Local Control OFF* at startup, and when POLY and MONO presets are selected.

How to turn Local Control back ON

Most keyboard synthesizers do not remember the *Local Control* setting when you power down the keyboard, so to turn *Local Control* back ON, you can just turn the keyboard off and back on again. However, some keyboards remember this setting, in which case you may want to use the last option above StartON+PreOFF to send *Local Control ON* at startup and OFF only for POLY and MONO presets. Then when you want to turn *Local Control ON*, simply reboot TBX2b while it is still connected to the keyboard. The Local Control ON message will be sent, and then you can disconnect without having to turn Local Control back ON manually.

CC Retransmission



POLY REPEAT CCs?
YES

YES flashes, and turning the knob alternates between YES and NO. Press enter to select. In POLY mode, by default any incoming Continuous Controller messages are sent on all active channels. This can produce a lot of data at once, and in some setups can result in a “MIDI bottleneck” situation where errors and timing problems result. When you set this option to “NO”, in POLY mode any incoming CC will only get retransmitted out ONLY on the channel on which it was received. This corresponds to a more MPE-like behaviour from our POLY algorithm (see below *Chapter 5 – A Note About MPE*).

MTS Device ID



MTS DEVICE ID?
000

TBX2b allows you to select the Device ID byte for MTS messages (see below *Chapter 5 – MTS*). A value of 127 (hex 7F) is defined by the *MIDI Association* to mean *All Devices*, so that any destination supporting MTS should recognise the message, but a majority of devices use a default value of 000 (zero), and will not respond to 7F. TBX2b therefore uses zero as the default value. Here you can change that value here to whatever is required by your device.

Display Brightness



BRIGHTNESS 50%
|||||

The LCD backlight brightness defaults to 50% to help prolong the life of the hardware. Turn the knob to change the brightness level in real time.

Bank Select Format

These options are available to support various synthesizers which implement the Bank Select message in different ways.



BANK FORMAT?
1 BYTE CC0

The bottom line flashes. Turn the knob to select an option.

```
1 BYTE CC0
1 BYTE CC32
2 BYTES CC0 CC32
2 BYTES CC32 CC0
```

Check the documentation for your synthesizer to choose the right value. In most cases it will be the first option, which is the default setting.

Pitch Bend Timing

TBX2b defaults to sending pitch bend messages with no time lag between the bend message and the note ON message.



The bottom line flashes. Turn the knob to select an option.

```
FAST (0ms)
MEDIUM (5ms)
SLOW (20ms)
```

The slower timing options are required by some older synthesizers which react more slowly to pitch bend messages. If you hear pitches bending at onset, a slower setting should fix the problem.

Sysex Retransmission

In some rare cases you may want to send sysex messages through TBX2b to another unit connected somewhere in the MIDI chain from MIDI OUT. Control of this behaviour is most important in POLY mode (see *Chapter 5*) where an incoming sysex message will be copied to multiple output channels when retransmission is on.



```
RETRANSMIT SYSX?  
NO
```

NO flashes, and turning the knob alternates between NO and YES. Press enter to select. You must be very careful that you know exactly what you are doing when you set this to YES, because some sysex messages may confuse TBX2b or confuse other gear in your setup and result in unpredictable behaviour.

USR Data

At the end of the Global programming process, the display shows the following message to remind you that although USR settings are global, they are too involved to program directly through the TBX2b interface, and are programmed instead via *UTE* software (see **Chapter 5**).



```
USR mode params  
are set w/SysX
```

5. Preset MODES

In this chapter we look at the four different primary functions provided by TBX2b called “modes” from which to choose for each of 40 presets. The modes primarily concern MIDI *Output*, and each mode is made for a different kind of MIDI synth (destination), with different ways of tuning. Understanding how each mode works will allow you to choose the appropriate mode for a given destination, and properly configure the destination for that mode.

Pitch Bend 2048 vs. Sysex 128

The modes are categorised into two groups: *Pitch Bend 2048*, and *Sysex 128*. In a *Pitch Bend 2048* mode, TBX2b processes all incoming MIDI data to produce microtonal output in real time using pitch bend, and each input channel can be assigned to a unique tuning table, for a total of 2048 possible pitches. In a *Sysex 128* mode, tuning data is sent via MIDI sysex to configure the destination when the preset is selected, and incoming MIDI data is then sent passively through to the destination with no realtime processing, and each input channel is routed to the same tuning, for a total of 128 possible pitches. The four modes break down as shown.

<i>Pitch Bend 2048</i>	<i>Sysex 128</i>
POLY	MTS
MONO	USR

The *USR* mode is special because it can be used not only for retuning, but also to send arbitrary sysex messages to any destination. Each mode is outlined in detail below.

POLY

This mode is intended for use with any synth that does not natively support microtuning *but supports pitch bend and is able to receive MIDI input and send MIDI output on separate multiple channels*. Please note that if a synth can receive on multiple MIDI channels, that does *not* guarantee that it can also send on multiple channels. *The synth must be able to both send and receive MIDI on multiple channels*. If the synth only sends output on one MIDI channel, it cannot be used with TBX2b POLY mode, but it can be used in MONO mode (see below). In POLY mode, each input channel on TBX2b is assigned to a 128 note tuning table, for a total of 2048 possible output pitches. Full-range microtonal output is achieved by sending each next incoming note out from TBX2b to the destination on a different MIDI channel, sending pitch bend before each Note ON event. The number of resulting retuned voices is thus equal to the number of available MIDI output channels in the destination (synth). The destination needs to be prepared to receive MIDI coming in on the channels that TBX2b uses for output, which you select. For example, if the destination supports General MIDI, then channel 10 is reserved for percussion and 15-part polyphony is available. This is the default

output channel configuration for POLY mode. All synths marked with the General MIDI symbol (GM or GM2) are compatible with TBX2b POLY mode. Other types of synths may or may not be compatible; you must consult the documentation for the synth to find out.

MONO

This mode is intended for use with monophonic synths which do not natively support microtuning, *but do support pitch bend*. This mode also may be used with synths that are otherwise polyphonic and support pitch bend, but do not support multiple MIDI output channels (see above). Like POLY mode, each input channel on TBX2b is assigned to a 128 note tuning table, for a total of 2048 possible output pitches. Full range microtonal output is achieved using pitch bend over a single MIDI channel, by tracking keys played and sending pitch bend values for each sounding note, correctly reproducing the expected behaviour for a monophonic destination.

MTS

This mode is for synths which support the MIDI Tuning Standard (MTS) Bulk Tuning Dump (non-real time) sysex message, as defined by the MIDI Manufacturer's Association (MMA). Check the documentation for your device to see if it supports this message. Sometimes this information is not in the user manual but in a separate MIDI reference document. Sysex looks cryptic, and is it not necessary to understand it. The sysex is shown below as given in the official MMA documentation so that you can compare it to what you find in the documentation for your device to be sure it supports this message.

```
[ F0 7E <device ID> 08 01 tt <tuning name> [xx yy zz] ... chksum F7 ]
```

```
F0 7E Universal Non-Real Time SysEx header
```

```
<device ID> ID of responding device
```

```
08 sub-ID#1 (MIDI Tuning)
```

```
01 sub-ID#2 (bulk dump reply)
```

```
tt tuning program number (0 – 127)
```

```
<tuning name> 16 ASCII characters
```

```
[xx yy zz] frequency data for one note (repeated 128 times)
```

```
chksum checksum (XOR of 7E <device ID> 01 tt <388 bytes>)
```

```
F7 EOF
```

The MTS message is sent to MIDI OUT immediately when an MTS preset is selected. You specify the <device ID> as a **Global Parameter** (see above). As defined by the MMA, this message supports 128-note tunings. To our knowledge there are no destination devices which support multi-channel MTS tables for more than 128 notes, so an MTS preset sends only one MTS sysex message out over one MIDI channel to retune 128 notes.

If you find a device which supports multi-channel MTS, please let us know so we may add a multi-channel MTS mode to allow MTS tunings with up to 2048 pitches.

USR

This mode is intended for synths which support sysex retuning of some kind but do not support the standard device-agnostic MTS Bulk Tuning Dump message. Required sysex messages vary, so the structure of the USR sysex message is flexible, allowing you to define the message from a set of options, including up to 32 arbitrary header bytes. This mode can be used to retune synths which support any of several other MTS messages made for 12-tones-per-octave tunings, as well as non-MTS (a.k.a. proprietary) sysex retuning found on some devices.

TBX2b stores ten banks of USR Data, numbered 0-9, so that you can define up to ten different sysex messages for various purposes. These sysex messages also do not necessarily need to send retuning data, so USR presets can also be used to send arbitrary sysex messages. USR Data is programmed using [Universal Tuning Editor](#). Further details about USR Data, including example use cases, are included in the documentation for the software.

A Note About MPE

Our POLY method of retuning has been implemented in H-Pi Instruments products since the development of our first tuning devices over twenty years ago. At the first release of the Tuning Box TBX1 in 2006, we referred to this type of tuning as *dynamic channel allocation*. A similar algorithm was being used by at least one other MIDI manufacturer around that time. In 2010 we changed the name of our algorithm to **General MIDI Microtuning**, to sound less technical and to emphasise the fact that the algorithm is compatible with any device that supports **General MIDI**. In the following years a few other products were brought to market by other companies using similar algorithms. With input from a few of the larger of these companies, in 2015 the MIDI Manufacturer's Association officially recognised this method of microtuning, calling it **Multidimensional Polyphonic Expression (MPE)**. There are however a few key differences between our POLY algorithm and the MIDI specification for MPE, namely that MPE requires a sequence of startup messages to assign channel zones, and has a dedicated common message channel. We chose not to implement MPE as such in TBX2b, because our tried-and-true POLY algorithm is more flexible than the MPE specification, and as far as we know it is fully compatible with synthesizers that support MPE. To accommodate the common message channel behaviour of MPE, we have added an option to TBX2b for the handling of incoming CC messages. For details, see **CC Retransmission** under **Global Settings** above.

If you have a MIDI synth (destination) which supports MPE but does not seem to respond correctly with TBX2b in POLY mode, please let us know.

6. Presets

To use TBX2b you will prepare at least one preset out of the 40 presets available. A preset is simply a set of instructions for TBX2b, to tell it how to behave for a given setup. All presets have a 16-character name, a bank and patch setting, and at least one assigned 128-note tuning table (some modes allow up to 16 different tuning tables). Other aspects vary according to the mode chosen for the preset (see *Chapter 5*). Preset programming can be done at the unit, and can also be done using [Universal Tuning Editor](#) (see *Chapter 7*). The process of programming a preset at the unit is described in this chapter. Pressing the *Settings/Save&Exit* button will save any changes made and exit, and pressing the *THRU/Cancel* button will cancel any changes made and exit preset programming.

Preset Programming

Programming a preset begins by pressing the knob (jog dial button). Once in programming mode, values are entered using either this knob button, or the Browse/Enter button, both of which we will refer to here simply as “enter”. Whichever preset is currently active will be programmed, so in order to program a certain preset, first select that preset. This is done by pressing a preset button, or a function button and a preset button together, or by navigating presets using pedals. Once a preset is selected, press the knob to begin programming it. For illustration, we will program preset 01.

1. Choose a mode for the preset (modes are explained in *Chapter 5*).



POLY flashes and turning the knob scrolls through the following options:

```
POLY
MONO
MTS
USR0 ... USR9
```

Select a MODE and press enter to continue. For information on modes, see *Chapter 5*.

2. Assign a bank and patch to the preset. This allows you to change the bank and patch of the destination synth when the preset is selected. By default these values are set to OFF meaning that no bank select or patch change message will be sent.

A screenshot of a blue LCD display with a black border. The text on the screen is '01 SELECT BANK:' on the top line and 'OFF' on the bottom line.

OFF flashes, and turning the knob cycles through numbers 000 through 127. Press enter to select the desired value, and the display will change to patch selection. Note that if you choose a value other than OFF, the bank message will be sent according to the format selected in Global Parameters (see *Chapter 4*). Please note also that some synthesizers do not support the Bank Select message at all, or they use the message in a non-standard way. Check the documentation for your synthesizer so that you fully understand how it will respond to this message before using it. Whenever you are not sure, just set it to OFF.

A screenshot of a blue LCD display with a black border. The text on the screen is '01 SELECT PATCH:' on the top line and 'OFF' on the bottom line.

Again, OFF will flash, and turning the knob will cycle through numbers 000 through 127. Press enter to select the desired value for the patch and continue. Some synthesizers use the message in a non-standard way. Please check the documentation for your synthesizer so that you fully understand how it will respond to this message before using it. Whenever you are not sure, just set it to OFF.

3. Assign a tuning table to each MIDI Input Channel. Each of the 16 MIDI Channels is assigned, and the process begins with the first Input Channel, shown on the display top line as CH01.

A screenshot of a blue LCD display with a black border. The text on the screen is 'TUNING CH01 0000' on the top line and 'TableNameIsShown' on the bottom line.

The tuning table number 0000 flashes, and turning the knob changes values between 0000 and 8191. Use the function buttons in combination with turning the knob to change values \pm by 10, 100, or 1000. Press enter to select the desired table for MIDI Input Channel 1 and continue.

4. For MTS or USR modes, the table selected in step 3 will apply to all input channels, and programming skips ahead to step 5 below.

For POLY or MONO modes, you have an option to assign the selected tuning table to all input channels, or to assign each input channel to a different tuning table.

```
ALL INPUT CHANLS
SAME TUNING? YES
```

YES flashes, and turning the knob alternates between NO and YES. If you select YES, then the table selected in step 3 is assigned to all MIDI Input Channels, and programming continues with step 5 below. If you select NO, then step 3 is essentially repeated for each of the remaining MIDI Input Channels 2-16, shown on the display as CH02 ... CH16.

```
TUNING CH02 0000
TableNameIsShown
```

5. At this point a MIDI output channel is assigned. For MONO, MTS, and USR modes, only one output channel is assigned, and programming skips ahead to step 6 below.

For POLY mode, multiple output channels are assigned.

```
01 OUT CHANNELS:
0000000000X0000000
```

The bottom line of the display represents 16 MIDI Output Channels in a row from left to right. An O means that a channel is ON, and an X means that the channel is OFF. By default all channels are ON except for channel 10, as this channel is reserved for percussion in the General MIDI specification. Each output channel flashes its default value, and turning the knob toggles the value ON or OFF. Pushing enter selects the value and moves right to the next channel. The output channels you select here should match the receiving channels of your chosen destination device.

Example: if a destination receives MIDI Input on channels 1-6, then only channels 1-6 should be set to ON and the rest should be set to OFF, as shown below.

```
01 OUT CHANNELS:
000000XXXXXXXXXX
```

For POLY modes, programming continues skipping ahead to step 8 below.

6. For MONO, MTS, and USR modes, only one output channel is selected.



```
01 SELECT OUTPUT  
CHANNEL: 01
```

Channel number 01 flashes, turning the knob changes values from 01 to 16. Press enter and continue. MONO mode programming skips ahead to step 8 below.

7. For MTS, and USR modes, a program number is assigned for the sysex message. This number is optionally used by the destination to assign the received MTS tuning to a specific memory location.



```
01 PROGRAM NUM.  
000
```

000 flashes, and turning the knob selects values from 000 to 127. Press enter to select the desired value and continue.

8. Last, a 16-character preset name is assigned. To simplify this process, a few options are given.



```
01 PRESET NAME:  
EDIT STORED NAME
```

Turning the knob cycles through the following options:

```
EDIT STORED NAME ... edit the currently stored preset name  
USE TUNING NAME ... use the name of the table assigned to the first input channel  
USE GENERIC NAME ... use TUNING PRESET XX (where XX is the preset number).  
USE BLANK NAME ... input an arbitrary name starting with empty spaces
```

Press enter to make a selection.

Each character of the chosen name can now be edited. If no editing is needed, simply press enter to confirm the name.

To edit the name, turn the knob. Each character flashes, starting with the leftmost character. Turning the knob scrolls through all available ASCII characters, and pressing enter selects the character, moving to the next one through all 16 characters left to right. After editing the last character, the name is entered, and the preset programming process is over.

Recalling a Preset

Presets are recalled by pressing preset buttons 0-9. Use the function buttons together with preset buttons to select presets 10-39. Presets can also be navigated forward and backward from any given value using pedals. See ***Chapters 1 & 2*** for more information.

7. Using TBX2b with Universal Tuning Editor (UTE)



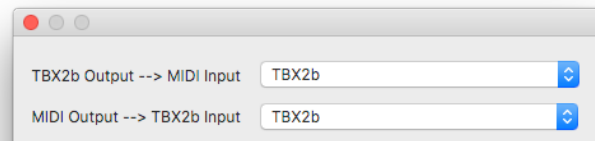
Some TBX2b functions require connecting the unit to a computer and running [Universal Tuning Editor \(UTE\)](#). The connection between the computer and TBX2b is made using the USB-MIDI Interface built into the unit.

Download, Install & Authorise UTE

With your purchase of TBX2b you received a license code via email for UTE. Go to the [H-Pi Instruments website](#) and log into your account to download the latest version of the software. The download link is found on the UTE product page. After installing the software, select the menu option Authorize (enter license) and enter the information exactly as it appears in the email you received. The computer must be connected to the internet. If registration fails with an unknown error, disable your firewall and/or virus software and try again. After authorising, you can migrate the software from one computer or hard drive to another by selecting the menu option Deauthorize (remove license) from within UTE.

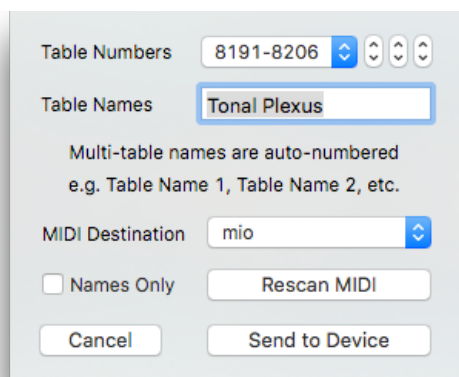
Connecting TBX2b

To work with our TBX2b, UTE needs to be able to ask TBX2b for information (referred to as a “query”), and TBX2b needs to be able to send data to UTE (called a “response”). To do this, move the MIDI Input Switch on the bottom of TBX2b to the USB-MIDI position, and connect TBX2b to the host computer using a USB cable. This will be referred to below as having TBX2b “**USB Connected**”. Do not use a USB-Hub, as TBX2b may not be recognised when connected through a hub. When TBX2b is properly connected, UTE will automatically select it as the MIDI input and output.



Programming a Tuning

To program TBX2b with your tuning, click **Devices** toolbar icon and select **Send Tuning to TBX2b**. Be sure to have TBX2b **USB Connected**.

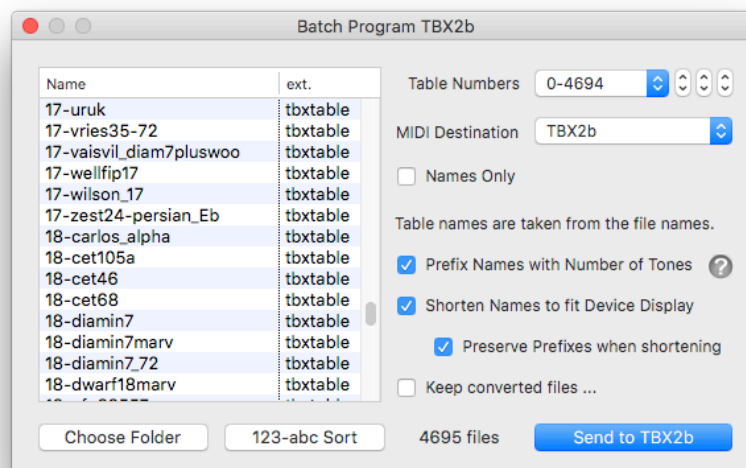


TBX2b assigns one tuning table per MIDI channel, with 128 notes per tuning table, so if your tuning contains more than 128 notes, it will be broken into multiple tables in TBX2b memory. For example, Tonal Plexus tunings contain 2048 notes ($16 \times 128 = 2048$), so the tuning is spread across 16 tuning tables in TBX2b memory. UTE figures out how many TBX2b tuning tables

are needed to store your tuning, and provides a popup menu for selecting the table numbers to program, with three controls for navigating the table numbers in increments of 10, 100, or 1000. When the tuning spans multiple TBX2b tuning tables, the names of the tables are numbered. In the example shown above, the table names will be *Tonal Plexus 1*, *Tonal Plexus 2*, *Tonal Plexus 3*, and so on, through *Tonal Plexus 16*. The checkbox *Names Only* also allows you to update only the table names without sending the tuning itself.

Batch Programming Tunings

To upload many tunings TBX2b at once in sequence, prepare a folder containing the tuning files, click the *Devices* toolbar icon, and select *Send Batch Tunings to TBX2b*. Click the *Choose Folder* button and navigate to the folder containing the files. The tunings may be in any format which can be imported by UTE (see *Chapter 4*). Once imported, you may change the order of the files by dragging them up or down in the list. Select the table numbers to be assigned to the files using the popup menu at the upper right.



TBX2b limits tuning names to 16 characters. Because TBX2b stores thousands of scales, it can be helpful to include the number of tones in the scale as a prefix to the tuning name in order to aid navigation. Options for converting the tuning names with prefixes and a shortened number of characters are provided as a convenience. If names are not shortened, any name longer than 16 characters will simply be cut off at the 16th character when sent to TBX2b. If you have chosen to number the files, you can use the button marked *123-abc Sort* to order the files by both number and name.

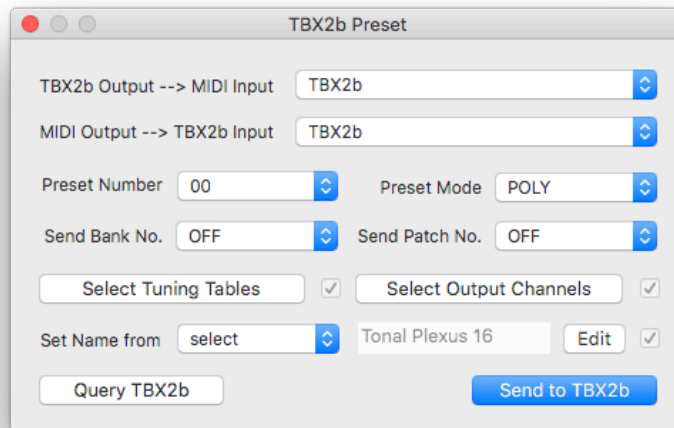
Check the option *Keep converted files ...* to archive the unique order of the imported files and save time should you want to upload the files again later. The converted files have the

extension *.tbxtuning* and are read faster by the batch window than the other file types. The names of the converted files are numbered in the order you have placed them in the listbox. Be sure to have TBX2b **USB Connected** before clicking **Send to Device**. Uploading one file takes less than one second, but if you upload thousands of files, be aware that this can take some time. For example, to upload the entire Scala archive of about 4700 scales takes about 90 minutes.

Be aware that tuning files containing more than 128 notes will be incomplete when using batch-upload, because TBX2b stores tuning tables in the MIDI standard of 128 notes. Although it is possible to import these larger scales and map them manually, this cannot be done using batch-import. Keep in mind the limitations concerning Scala files listed in **Chapter 4**.

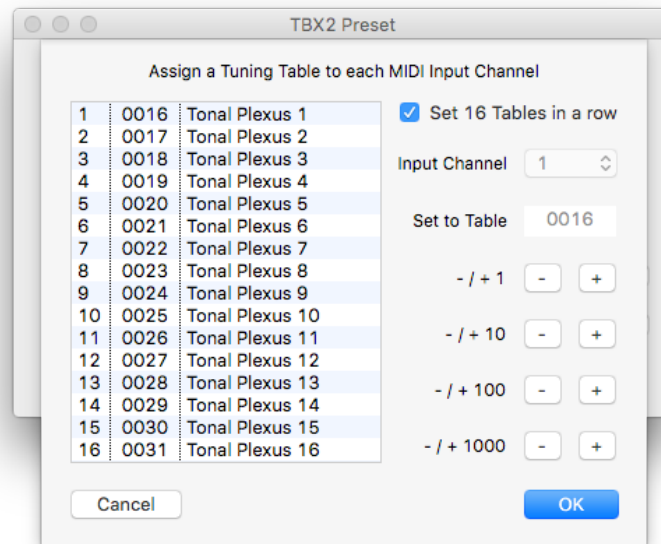
Programming a Preset

TBX2b has 40 presets (numbered 0-39) which can be programmed at the device itself, but sometimes it will be easier to set up your presets using UTE, for example particularly when you are working with multi-table presets such as those needed for Tonal Plexus keyboards. Click the **Devices** toolbar icon and select **Program a Preset** to open the window.



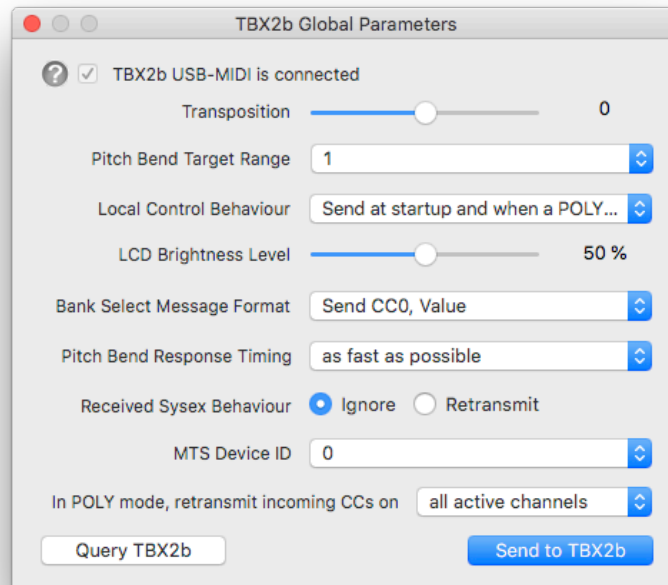
To edit a preset already programmed on your TBX2b, be sure to have TBX2b **USB Connected**, then after selecting a preset number, click the **Query TBX2b** button at the bottom left corner of the window, and the data for the selected preset will appear in the window for editing.

Clicking the **Select Tuning Tables** button opens a window allowing you to easily set 16 tables in a row to the 16 MIDI input channels, since this will be a common need for those working with Tonal Plexus keyboards (and possibly other microtonal controllers).



Setting Global Parameters

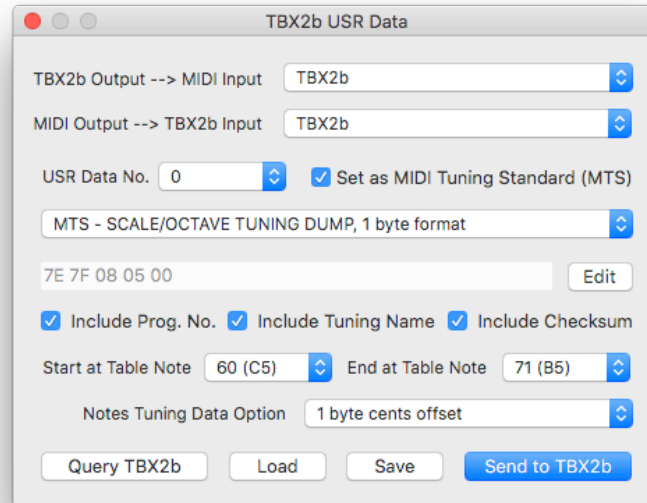
Click the *Devices* toolbar icon and select *Set Global Parameters*.



Note that TBX2b parameters can also be edited directly at the device without using UTE.

Setting USR Data

TBX2b provides 10 USR presets (numbered 0-9). USR Presets allow sending different kinds of sysex messages to retune keyboards according to manufacturers' specifications.



Unlike other features of TBX2b, USR Data functions are rather involved, so they cannot be programmed at the device, and must be programmed using UTE. Each USR Preset Data consists of two parts: *header bytes*, and *options*. The *header bytes* may vary in length anywhere from 0 to 32 bytes. The *options* are as follows:

Options	
Include program number?	Yes / No
Include tuning table name?	Yes / No
Include checksum byte?	Yes / No
Notes formula (4 options)	1 byte \pm cents offset per note (MTS 1-byte cents form) 2 bytes pitch bend MSB, LSB (MTS 2-byte cents form) 3 bytes MTS data per note (MTS frequency form) 3 bytes pitch bend data per note (Note, MSB, LSB) do not send any note data (useful for sending misc. sysex)
Tuning table start key	(0..127) [default = Note 60, middle C]
Tuning table end key	(0..127) [default = Note 72, B above middle C]

When the USR Preset is selected at TBX2b, a sysex message is sent having the following form:

Description	Bytes
sysex start	1
[header bytes]	none or 1 to 32
[program number]	none or 1
[tuning name]	16
[notes data]	none or 1 to 128
[checksum]	none or 1
sysex end	1

Refer to the documentation of your synthesizer to determine the correct header bytes and options to assign to a USB Preset. The TBX2b USB Data window has a popup menu for selecting any of the MIDI Tuning Standard (MTS) formats to simplify the process of inputting all the parameters if your keyboard requires one of the MTS message formats.

USB Data Example 1

The Roland FP-30 keyboard supports the MTS Scale / Octave Tuning 1-Byte Format. USB Data for this message can be prepared simply by selecting that option from the popup menu. The USB Data options will be automatically populated as follows.

USB Data Options for setting up Roland FP-30 MTS Sysex Retuning	
Header Bytes	7E, 7F, 08, 08, 03, 7F, 7F
Include program number?	No
Include tuning table name?	No
Include checksum byte?	No
Notes formula (4 options)	1 byte ± cents offset per note (MTS 1-byte cents form)
Tuning table start key	Note 60, middle C
Tuning table end key	Note 71, B above middle C

USB Data Example 2

The Roland FP-30 can also be retuned across its entire range to any scale (not just 12 tones) by TBX2b in POLY mode, but only when the keyboard is first put into GM1 or GM2 mode. You can push some buttons on the Roland to set that up, but there is also a sysex message which forces the keyboard into the needed mode. So you can use a TBX2b USB preset to send that message, and then press a button on TBX2b to put the synth into the correct mode, after

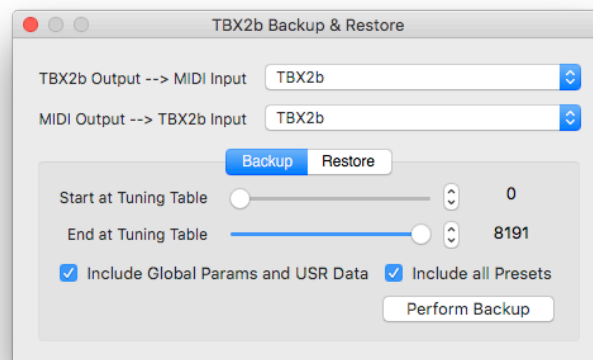
which you can press other preset buttons programmed to POLY. In the user manual we find that the sysex message required by the Roland for “GM1 System On” is: F0, 7E, 7F, 09, 01, F7. We prepare USB Data for this message as follows.

USB Data Options for setting up Roland FP-30 GM System ON	
Header Bytes	7E, 7F, 09, 01
Include program number?	No
Include tuning table name?	No
Include checksum byte?	No
Notes formula (4 options)	Do Not Send
Tuning table start key	-
Tuning table end key	-

Once this USB Data is assigned to a USB number (0-9) then you can program the USB preset at TBX2b by selecting that USB number as the mode for the preset.

Backing up & Restoring Data (.udevbackup)

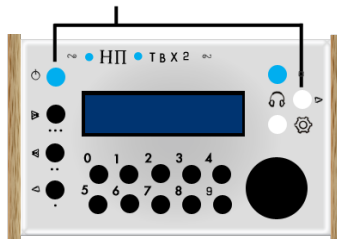
Because TBX2b is a device having a large memory for presets, USB Data, Global Parameters, and Tunings, it is useful to make backup copies of the contents of the device for example, if it is used for a series of concerts where many presets are involved. Before updating TBX2b firmware, it is also a good idea to backup data, as under some rare conditions memory corruption may occur during a firmware update. A full backup takes about half an hour to complete, and a full restore takes about an hour. Normally you will not need to do a full backup / restore, as it is unlikely that you will be using all 8192 tunings very often, so UTE gives you the option to select a range of tunings to back up. A normal partial backup / restore process can be done in a few minutes.



Updating Firmware (.ubootsyx)

TBX2b firmware can be updated at any time, to fix reported problems or add new features. When you have TBX2b **USB Connected** to UTE, the software can query the firmware version installed on your TBX2b and alert you to any available updates. Firmware updates are issued as files having a `.ubootsyx` extension, which is transferred to the device over MIDI. Click the **Devices** toolbar icon and select **Update Firmware** to select a firmware file and upload it to TBX2b. The process normally takes about two minutes to complete. Simply follow directions. Use the TBX2b Firmware Update Startup:

TBX2b Firmware Update Startup



Press the rightmost white button and hold it down while you turn on the unit with the blue power switch on the left. This tells the unit “we are ready to update the firmware”. The display will read as shown below.



- Make sure the MIDI Input Switch on the bottom of TBX2b is switched to the USB-MIDI Interface.
- Make sure the TBX2b USB-MIDI Interface is plugged into the host computer.
- While firmware is sent, UTE shows a progress bar, and TBX2b displays a percentage complete. If not, see [Troubleshooting](#) below.
- At the end, UTE shows an alert, and TBX2b shows a brief message, then restarts itself.

UTE + TBX2b Troubleshooting

TBX2b seems “frozen” when it is I/O connected and I try to send MIDI.

Your system is likely producing a MIDI feedback loop. This can happen if you have some other software running which accesses MIDI devices and routes MIDI input and output. Quit the other application(s) and try again.

Trying to send the firmware, TBX2b just shows “Firmware Updater, waiting for MIDI”.

The USB cable may not be properly connected. Do not use a USB Hub. Or you have selected the wrong MIDI destination in UTE, or you have set the MIDI Input Switch on the bottom of TBX2b to an incorrect position.

Sending the firmware, TBX2b does not display an increasing percentage, stays at 0%, or reaches some other number and stops there.

The USB cable may not be properly connected. Do not use a USB Hub.

Troubleshooting

TBX2b is on, but I get no output.			
<i>First verify that the MIDI Input Switch on the bottom of the unit is in the correct position (see Chapter 1).</i>			
<i>Does the MIDI IN LED light up when you play a keyboard key (send a Note ON message)?</i>			
YES <i>Does the MIDI OUT LED light up?</i>		NO <i>Are you connecting via USB Host Input?</i>	
YES The problem is at the destination device. Check connections and MIDI settings there.	NO Try using a different tuning table.	YES • <i>Is the source device class-compliant? (if not, it won't work)</i>	NO The problem is at the source device. Check connections and MIDI settings there.
Using TBX2b in POLY or MONO mode, when I play one key, I hear two pitches.			
<i>Is the TBX2b Global Setting for Local Control set to send Local Control OFF to the destination?</i>			
YES • The destination does not support the Local Control OFF message and you must turn off Local Control manually at the destination.		NO • Press the TBX2b Settings button and change the <i>Local Control</i> option to send this message to the destination.	
The tuning sounds wrong.			
<i>Are the pitches sounding in 12-tone equal temperament?</i>			
YES • <i>Have you connected TBX2b MIDI THRU to destination MIDI IN? Connect TBX2b MIDI OUT to the destination MIDI IN.</i> • For POLY or MONO mode, check that the Pitch Bend Range of the destination matches the Pitch Bend Range in TBX2b Global Settings. • For MTS or USR mode, check that the destination is not filtering incoming sysex messages.		NO • For POLY or MONO mode, check that the Pitch Bend Range of the destination matches the Pitch Bend Range in TBX2b Global Settings. • For USR mode, check that the tuning format you have selected in the TBX2b USR Data matches what is required by the destination.	

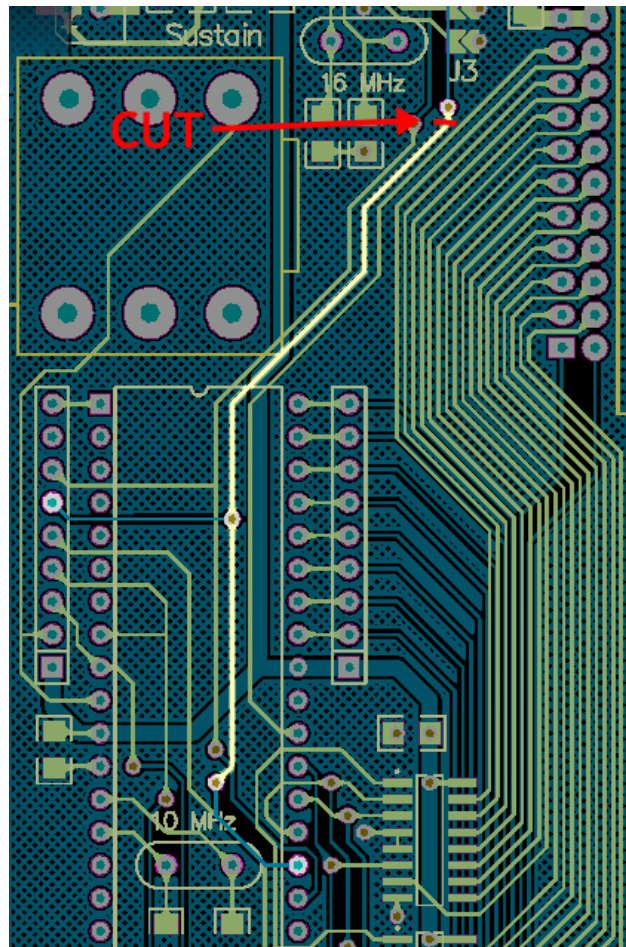
I only hear sound only every 15th (or other number) of keys I play, the rest are silent.	
Output from TBX2b using a POLY preset requires multiple channels, and your destination is receiving MIDI on only one channel. You need to match sending and receiving channels, or use a different mode if your destination is limited to input on only one MIDI channel (see Chapter 5).	
Every so often instead of microtonal pitch output, there is a percussion sound.	
The destination is General MIDI and you are sending POLY output on Channel 10. Go to the preset output channel settings and turn off channel 10.	
Pitches are bending after they sound.	
For POLY mode, this means that the destination is responding slowly to pitch bend messages. Select a slower Pitch Bend Response Timing in TBX2b Global Settings.	For MONO mode, this may be normal portamento behaviour. If the bending is not normal portamento, the destination may be responding slowly to pitch bend messages and you may need to select a slower Pitch Bend Response Timing in TBX2b Global Settings.
I'm having some other problem I don't see listed here.	
<i>Have you read through this manual?</i>	
YES	NO
Please send an email to hpiinstruments@zentral.zone	Please read the manual from beginning to end.

See also the section *UTE-TBX2b Troubleshooting* in Chapter 7

Appendix – UPLEX Modification

We found that U-PLEX Tonal Plexus keyboards cause a MIDI feedback loop when connected to the USB Host Input of TBX2b. To fix this problem, a trace on the U-PLEX main circuit board should be cut. This is a simple procedure which takes just a couple of minutes and requires only a small philips-head screwdriver and an Xacto-knife (razor blade). Cutting this trace has no negative side effects for using U-PLEX in any other way. It simply solves the problem.

1. Remove the six screws from the leftmost top panel of U-PLEX and remove the top panel.
2. Using an Xacto-knife (razor blade) carefully cut through the trace under J3 as shown in the image below.



3. Replace the top panel and screws.

The keyboard can now be connected to the USB Host Input of TBX2b.

Credits

TBX2b was invented and designed by Aaron Andrew Hunt. PCB & firmware by Jordan Dimitrov Petkov.

Special thanks to everyone who pre-ordered TBX2b.

This documentation is written by Aaron Andrew Hunt.

Thank you for supporting H-Pi Instruments and TBX2b.

©2020 H-Pi Instruments · FOR THE FUTURE OF MUSIC