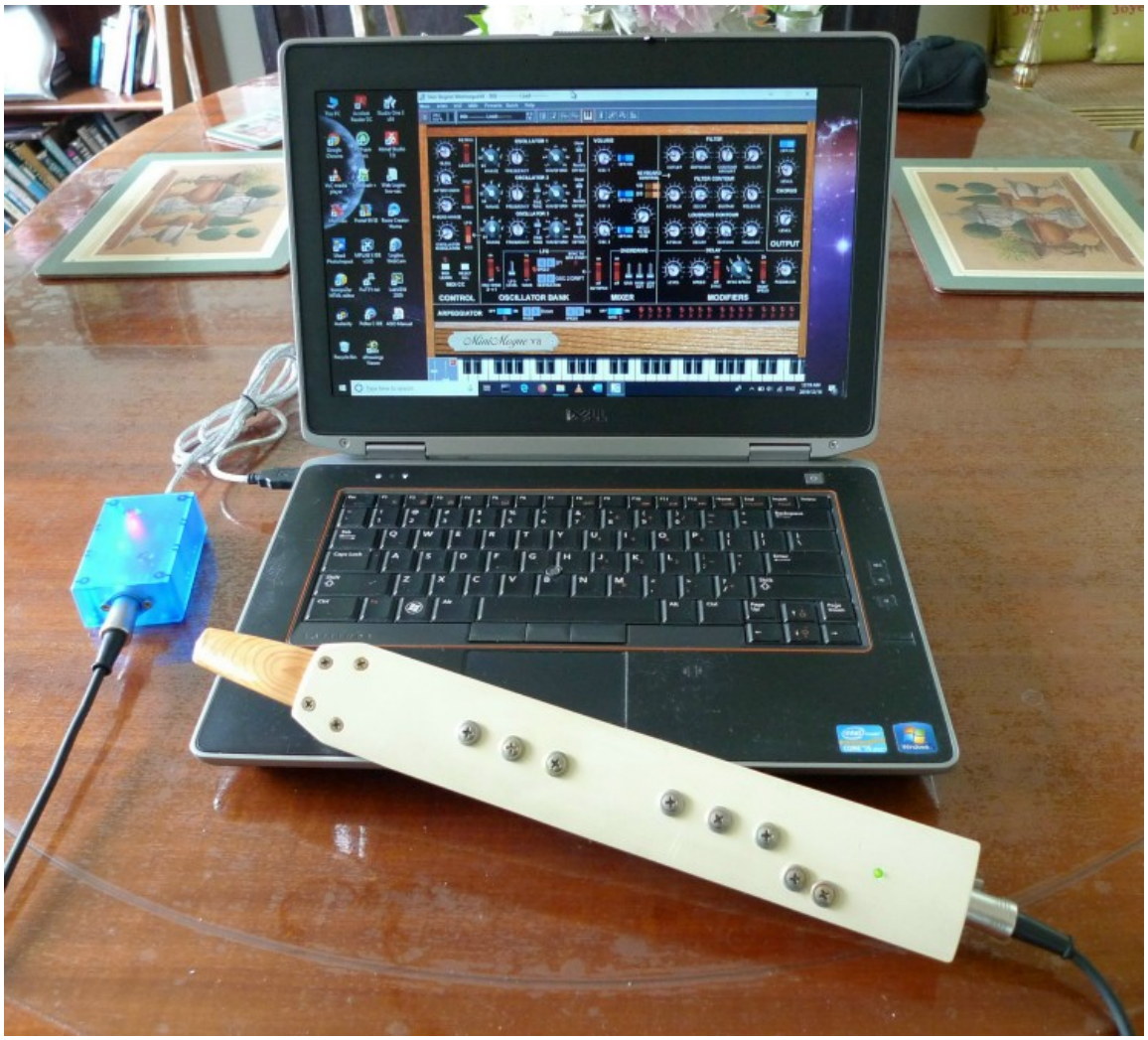
Build the "REMI" mk2

Recorder-like Electronic Musical Instrument

(EWI MIDI Controller)

A DIY Project by M.J. Bauer

This web page describes the design and construction of the "second generation" (mk2) REMI handset.
For a general introduction to the project, see page: "Introducing the REMI".



REMI 'mk2' Handset with MIDI-USB adapter box

Overview

REMI mk2 is an "electronic wind instrument" (EWI) using a fingering scheme based on the traditional recorder, simplified a little to make it easier to play. The mk2 "handset" is simply a MIDI controller with standard MIDI OUT connection (5-pin DIN socket). It does not have a built-in sound synthesizer.

A companion "REMI Synth Module" allows the REMI 2 to be played "stand-alone", without needing a computer. However, a low-cost MIDI-USB adapter (under $10) allows the REMI to be used with a "virtual synthesizer" (software application) running on a computer, as shown in the photo above. There are many freeware "virtual synthesizer" app's available. (There are much better ones available at a price!)

A button on the under-side of the handset, operated in conjunction with the touch-pads, selects one of eight instrument "Presets". The selected Preset determines various instrument and MIDI configuration parameters. For example, a Preset sound is chosen from a collection of pre-defined synth patches programmed into the REMI Synth Module. The Preset also selects one of a set of MIDI "programs" (instrument voices) when operating with a 3rd-party MIDI sound module or computer synth application.

Note: The REMI handset and sound synthesizer are not commercially available products. The hardware designs and software source code are in the public domain. The designs may be replicated by suitably skilled electronics

professionals or hobbyists. Detailed step-by-step instructions and parts lists are not provided. The information published here may be used as a basis on which to design your own customized EWI.

MIDI Controller Operation

The 'MIDI OUT' (transmit) command set includes: Program (voice) selection, Note-On/Velocity, Note-Off, Breath pressure, Expression or Channel Volume, Modulation (CC-01), All Sound Off and System Reset. The REMI handset can be set up to use any of 16 MIDI channels.

In normal note trigger mode, the REMI will send a Note-On/Velocity command when the breath pressure exceeds a preset threshold. A corresponding Note-Off command will be sent when the breath pressure drops below the "note-off pressure level". After a new note is initiated, a change in fingering pattern will cause another Note-On/Velocity command to be transmitted without first sending a Note-Off. If the external MIDI sound module is set to Mono mode, this should cause the module to produce a different note, i.e. to change pitch, without "re-attacking" the amplitude envelope. The musical term for this is "Legato". (Note: Legato mode can be disabled via the "service port" CLI, for synthesizers which do not support this mode of operation.)

While there is no note being played, the PRESET button may be pressed in conjunction with a touch-pad to select one of eight instrument "Presets". A MIDI "Program Change" command will be transmitted. The Program Number sent depends on the user-settable Preset configuration.

In addition, pressing the PRESET button while none of the touch-pads is touched will cause the REMI to transmit a MIDI "All Sound Off" command followed by a "Controller Reset" command. At the same time, the breath pressure sensor will be re-calibrated.

Fingering Schemes

There are two fingering schemes to choose from. The firmware has a configuration parameter allowing selection of either option. The physical layout of touch pads for each of the two schemes is shown below.

Touch-pad layout for the "standard" (recorder-like) fingering scheme

Touch-pad layout for the "alternate" (generic EWI) fingering scheme

The first option (top picture) is more reminiscent of a recorder and should appeal to existing recorder players. The second option is a tad more abstracted from the recorder. An extra pad (LH4) operated by the left-hand little finger is added, but overall, it's a simpler scheme which is compatible with some of the commercial EWIs out there, for example, the amazing Sylphyo.

Scheme #1

The "standard" REMI scheme has 8 touch-pads on the upper surface, i.e. one pad (hole) fewer than a real recorder. Three pads are operated by fingers on the left hand while five pads are operated by fingers on the right hand. The fourth and fifth RH pads (RH4 and RH5) are both operated by the little finger.

Two pads on the underside of the handset operated by the left-hand thumb select one of three ranges of notes, each range being two octaves, as shown in the chart. The octave pads are located so that one or other or both pads together can be touched. When both octave pads are touched, the "middle" note range is selected (normally C4 to C6). By moving the thumb to the upper or lower pad, the range of notes is shifted up or down by one octave, respectively. The octave pads extend the overall range to four octaves.

Referring to the Fingering Chart below, it can be seen that the fingering combinations cover two octaves,

without changing the octave selection by the LH thumb. Selection of notes in the first octave of each range follows quite closely the fingering patterns of the recorder, including C" (Alt.) above the low C'.

Contrary to the recorder, the second octave simply repeats the fingering pattern of the first octave, with the top pad (LH1) released, up to G". Above high G", the fingering gets a bit weird, but not as weird as a real recorder. Notes above G" can be avoided, if preferred, simply by changing octaves using the LH thumb (except when already in the highest octave range).

REMI Fingering Chart (std)

	B^0 (Alt)	C'	C#' Db'	D'	Eb' D#'	E'	F'	F#' Gb'	G'	Ab' G#'	A'	Bb' A#'	B'	C" (Alt)
LH1	●	●	●	●	●	●	●	●	●	●	●	●	●	○
LH2	●	●	●	●	●	●	●	●	●	●	●	○	○	●
LH3	●	●	●	●	●	●	●	●	●	○	○	◇	○	○
RH1	●	●	●	●	●	●	●	○	○	◇	○	◇	○	x
RH2	●	●	●	●	●	●	○	◇	○	◇	○	◇	○	x
RH3	●	●	●	●	○	○	x	◇	○	◇	○	◇	○	x
RH4	●	●	○	○	◇	○	x	◇	○	◇	○	◇	○	x
RH5	●	○	●	○	◇	○	x	◇	○	◇	○	◇	○	x

	B' (Alt)	C"	C#" Db"	D"	Eb" D#"	E"	F"	F#" Gb"	G"	Ab" G#"	A"	Bb" A#"	B"	C'"
LH1	○	○	○	○	○	○	○	○	○	○	○	○	○	○
LH2	●	●	●	●	●	●	●	●	●	○	○	○	○	○
LH3	●	●	●	●	●	●	●	●	●	●	●	●	●	●
RH1	●	●	●	●	●	●	●	○	○	●	●	●	●	○
RH2	●	●	●	●	●	●	○	◇	○	●	●	○	○	x
RH3	●	●	●	●	○	○	x	◇	○	◇	○	◇	○	x
RH4	●	●	○	○	◇	○	x	◇	○	◇	○	◇	○	x
RH5	●	○	●	○	◇	○	x	◇	○	◇	○	◇	○	x

Legend: ● Pad touched ○ Pad not touched ◇ Any pad(s) in group x Don't care (No effect)

If two or more pads marked with a diamond symbol (◇) are touched, the effect is the same as if any one of the pads is touched, i.e. the note is flattened by one semitone.

The little finger on the right hand can select either pad RH4 or RH5, which are close together.
Pads RH4 and RH5 are located so that both can be touched at once by the fourth finger, but this fingering oddity is needed only for selection of the alternate B note in each octave.

The alternate C" is designed to maintain semblance to recorder fingering in the first octave.

Scheme #2

The "alternate" scheme also has 8 touch-pads on the upper surface. Four pads are operated by fingers on the left hand and four pads are operated by the right hand. The fourth pad on the left hand (LH4), which doesn't exist on the recorder, serves to sharpen certain notes (e.g. to produce C# and F#). This arrangement, in keeping with other EWI designs, is simpler and (arguably) more logical than the fingering on acoustic wind instruments. Apart from the absence of pad RH5 and the addition of LH4, fingering is much the same as in the "standard" scheme. Octave selection is done in exactly the same manner.

REMI Fingering Chart (Alt.)

	C'	C#' Db'	D'	Eb' D#'	E'	F'	F#' Alt.	F#' Gb'	G'	Ab' G#'	A'	Bb' A#'	B'	C"
LH1	●	●	●	●	●	●	●	●	●	●	●	●	●	○
LH2	●	●	●	●	●	●	●	●	●	●	●	○	○	●
LH3	●	●	●	●	●	●	●	●	●	○	○	◇	○	○
LH4	○	●	x	x	x	○	●	x	x	x	x	x	x	○
RH1	●	●	●	●	●	●	●	○	○	◇	○	◇	○	x
RH2	●	●	●	●	●	○	○	◇	○	◇	○	◇	○	x
RH3	●	●	●	○	○	x	x	◇	○	◇	○	◇	○	x
RH4	●	●	○	◇	○	x	x	◇	○	◇	○	◇	○	x

	C"	C#" Db"	D"	Eb" D#"	E"	F"	F#" Alt.	F#" Gb"	G"	Ab" G#"	A"	Bb" A#"	B"	C'"
LH1	○	○	○	○	○	○	○	○	○	○	○	○	○	○
LH2	●	●	●	●	●	●	●	●	●	○	○	○	○	○
LH3	●	●	●	●	●	●	●	●	●	●	●	●	●	●
LH4	○	●	x	x	x	○	●	x	x	x	x	x	x	○
RH1	●	●	●	●	●	●	●	○	○	●	●	●	●	○
RH2	●	●	●	●	●	○	○	◇	○	●	●	○	○	x
RH3	●	●	●	○	○	x	x	◇	○	◇	○	◇	○	x
RH4	●	●	○	◇	○	x	x	◇	○	◇	○	◇	○	x

Legend: ● Pad touched ○ Pad not touched ◇ Any pad(s) in group x Don't care (No effect)

Different fingering schemes may be implemented by modifying the firmware, of course.

Handset Design

The REMI mk2 handset design incorporates a low-cost 8-bit microcontroller (PIC18F45K22) with direct touch-sense inputs for the touch-pads. The handset has a classic MIDI output connection (5-pin DIN socket), so it can be plugged into the 'MIDI IN' socket on the REMI mk2 Synth Module.

The MIDI-IN connector on the REMI synth module supplies 5V DC power to the handset.

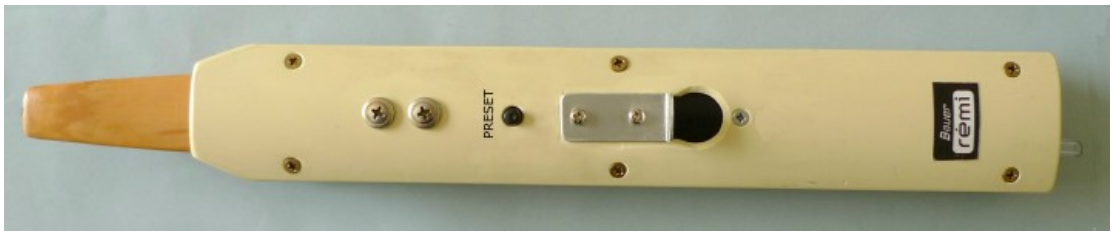
The REMI can connect to other MIDI sound modules, or to a computer via a MIDI-USB adapter module, allowing it to operate without a REMI synth module. There is a mountain of software synth app's available for Windows, Mac, iOS and Android - many free to download - just Google "VST synth".

The microcontroller board has a USB-serial bridge (FTDI FT232RL breakout board) allowing the handset to be connected to a PC running a terminal emulator (e.g. "Putty"). The USB "service port" provides a command-line user interface (CLI) to facilitate setting up and testing handset operation. (See "Firmware" section below for more information about the USB "service port" CLI.)

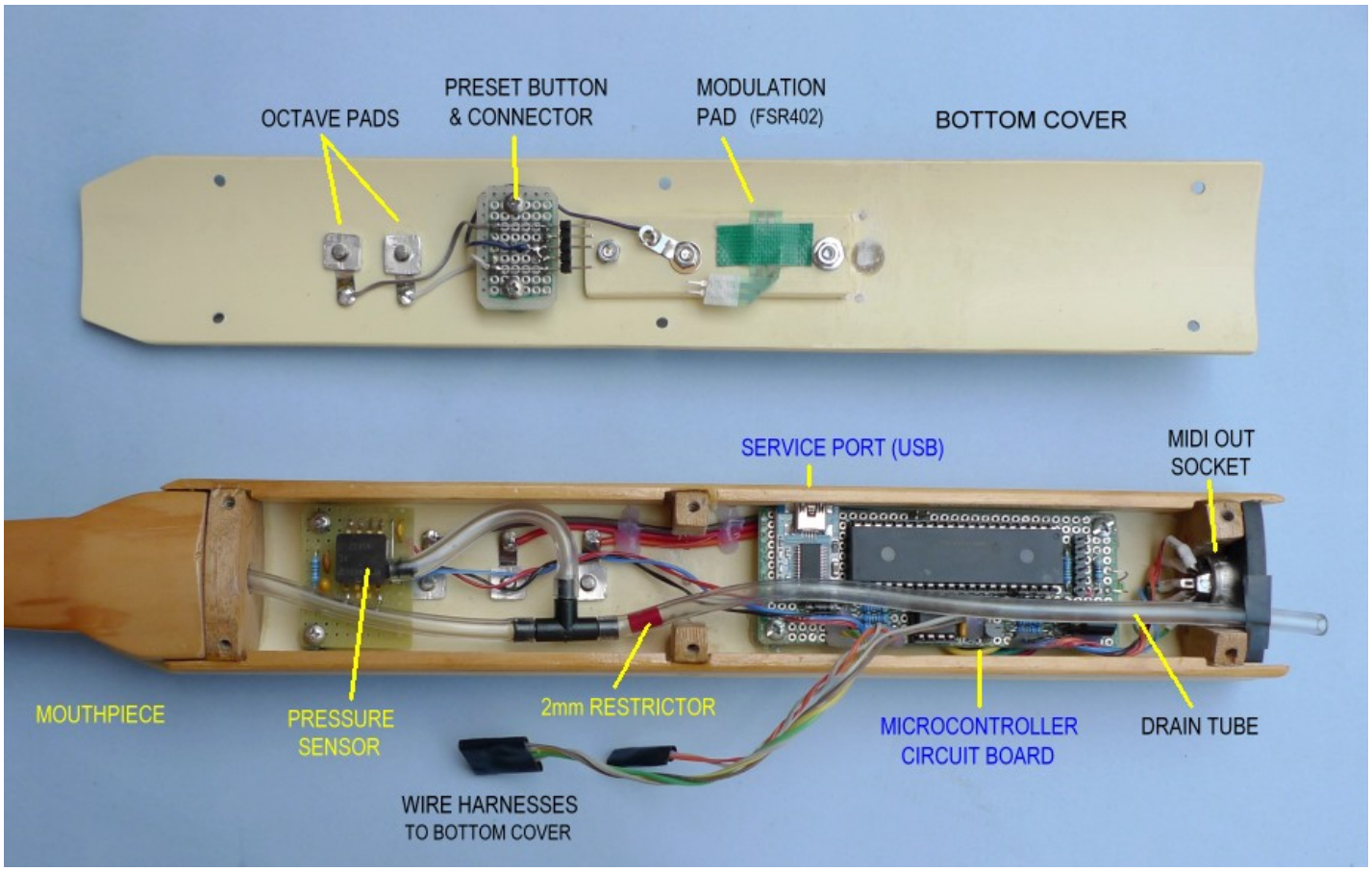
Construction

The prototype handset was made from two pieces of PVC plastic cut from a scrap of drain pipe, 110mm outside diameter, 3mm wall thickness. A suitable scrap may be found perhaps on a building site. The curved surface makes the plastic covers very rigid and strong. The upper piece has 3 wooden "ribs" (8~12mm thick) glued to it, allowing the bottom cover to be attached with self-tapping screws. Side strips of thin timber are glued to the ribs. The plastic surfaces are not painted, but rubbed with fine abrasive cloth (P400) to give a smooth matte finish. The result is a very "retro" (1950's) look and feel.

Top surface has 8 touch-pads and a status LED



Bottom cover has 2 octave pads, PRESET button and Modulation Pad
(The metal bracket near the Modulation Pad is a thumb rest.)



REMI handset mk2 -- Internal parts layout

The touch-pads are self-tapping screws with a broad flat head, cadmium-plated (I think). These are inserted through holes in the handset playing surface, fastened on the inside with aluminium retainers (~9mm square). A drill template and cross-section drawing are available for download -- see links at bottom of page.

Addendum (2022): Chicago screws, as used for leatherwork, book-binding, etc, make excellent touch-pads. These are available in a variety of metal finishes incuding brass, nickel, gun-metal, stainless-steel, etc. Size required is 4mm (shank length) x 10mm diameter head. Available online at low cost.

The mouth-piece is carved out of soft-wood with a 6mm hole drilled through the middle. It should be coated with a water-based sealer* to prevent moisture absorption in the wood. The nylon tubing is inserted into the 6mm hole, all the way through to the tip, and held in place with a water-based sealer. The mouth-piece is attached to the upper plastic cover with 4 self-tappers, so that it is easily removeable.

*Warning: Do not use a paint or laquer with oil-based solvent on the mouthpiece, for 3 reasons... (1) it tastes terrible, (2) it could cause a health hazard, and (3) the vapours given off by the solvent could damage the Fluoro-silicone membrane in the pressure sensor. Recommended coating for the mouthpiece is water-based polyurethane or some other safe product used to varnish wooden toys.

Pressure Sensor "plumbing"

Tubing and other bits and pieces needed to make the airways inside the handset may be sourced from the garden irrigation section of your local hardware store. The prototype handset (pictured) used clear nylon tubing (3mm ID, approx 5.5mm OD) for the internal airways linking the mouth-piece, pressure sensor IC and the "drain tube". The T-joiner needed the barbs cut off to fit the 3mm tubing.

The sensor air inlet barb is slightly less than 3mm in diameter, which is a wee bit too small to make a good seal with the 3mm ID nylon tube. An easy solution is to fit a short bit of 2.5mm (nominal diameter) heat-shrink tubing over the barb, shrink it with a hot air blower or whatever (taking care not to melt the sensor!), then fit the 3mm nylon tube over it.

A drain tube is recommended because moisture condensation occurs inside the airways and it is probably sensible to provide an exit for the moisture. Also, the drain tube allows air-flow, which is preferable to a sealed system for playability. However, the exit air flow needs to be restricted somewhat to produce a sufficient range of pressure inside the sensor. This can be achieved by fitting into the drain-tube a small plug with a 2mm hole through it, perhaps cut from a bit of plastic insulation sleeving if you can find some of suitable size (3mm OD, 2mm ID). The best place to fit the restrictor is close to the T-joiner, to minimise moisture condensation.

It is also recommended to mount the pressure sensor above the T-joiner, i.e. close to the mouthpiece, to prevent moisture from accumulating in the sensor, as shown in the picture above.

Modulation Pad

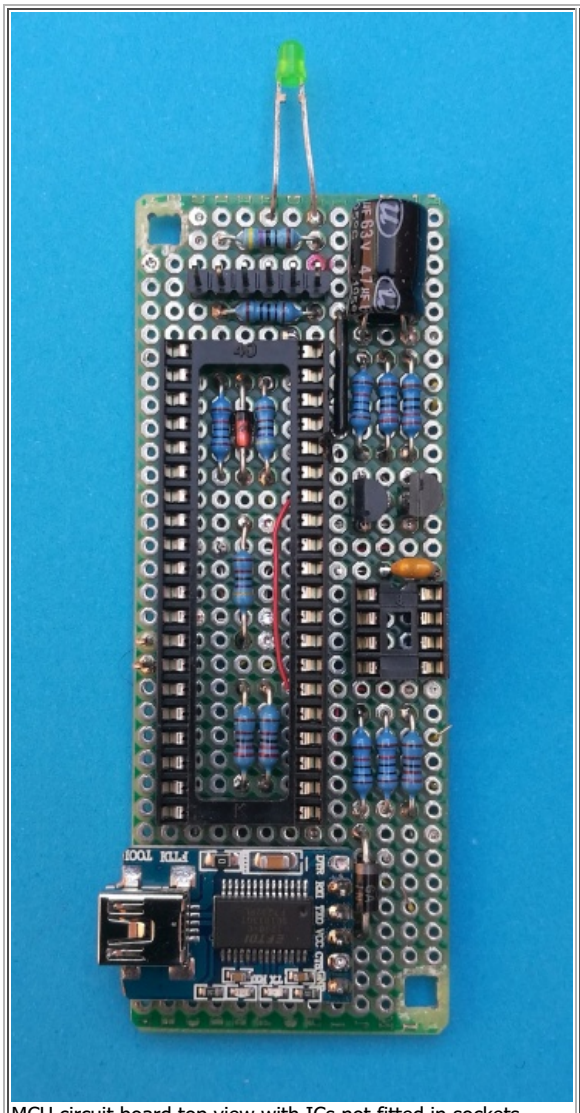
The simplest option is to use a force-sensitive resistor (FSR-402). This will give a uni-polar output, which is not ideal for a Pitch Bend controller, but quite satisfactory as an "Effect Controller" (e.g. vibrato depth, timbre modulation, etc). The FSR is mounted behind a round cutout in the bottom cover of the handset. A plastic "backing support", made from the same material as the covers, is screwed to the inside of the bottom cover. A piece of black PVC insulation tape is stuck over the top of the FSR402 to keep it in place. The right-hand thumb pushes the FSR against the backing support.

Note: The FSR402 is fragile. Take care not to damage the flex strip. Don't bend it too tightly. Insert thin washers between the handset body (bottom cover) and the FSR support piece to prevent the FSR edges from being compressed hard.

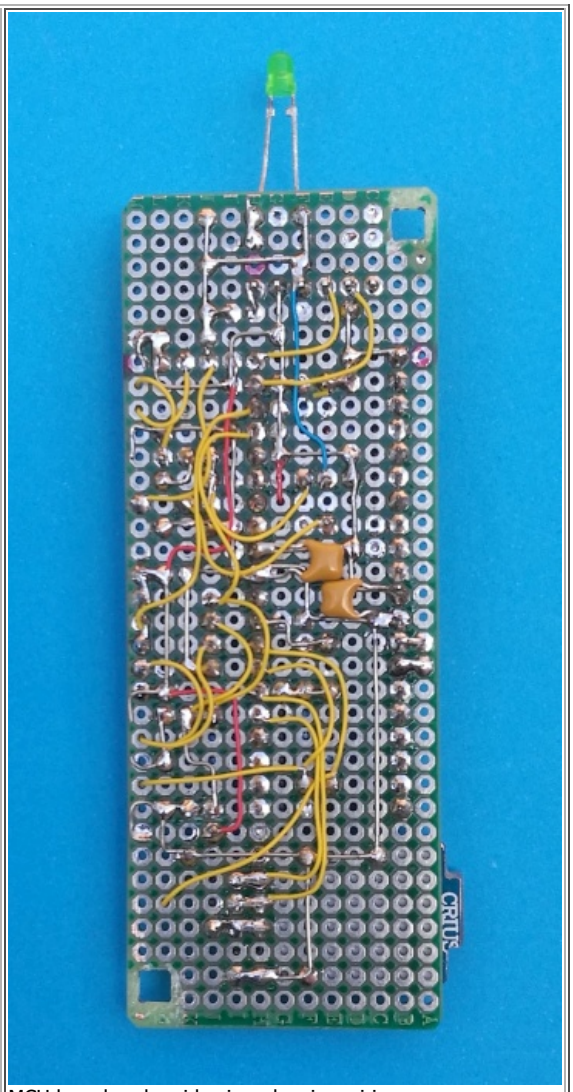
Modulation Pad (FSR402) resting on its backing support piece

Electronics

The handset microcontroller circuit board is constructed on a piece of prototyping board measuring 90 x 35 mm. A 40-pin DIL socket for the PIC18F45K22 MCU chip takes up most of the board area. A 6-pin SIL header for connection of the MCU programming tool (PICkit 3) is located at the top end of the board. The USB-Serial bridge (FT232RL breakout board) is located at the bottom end of the board. The MIDI transmitter and analogue sensor signal conditioning components (LM358, etc) are placed beside the MCU socket.

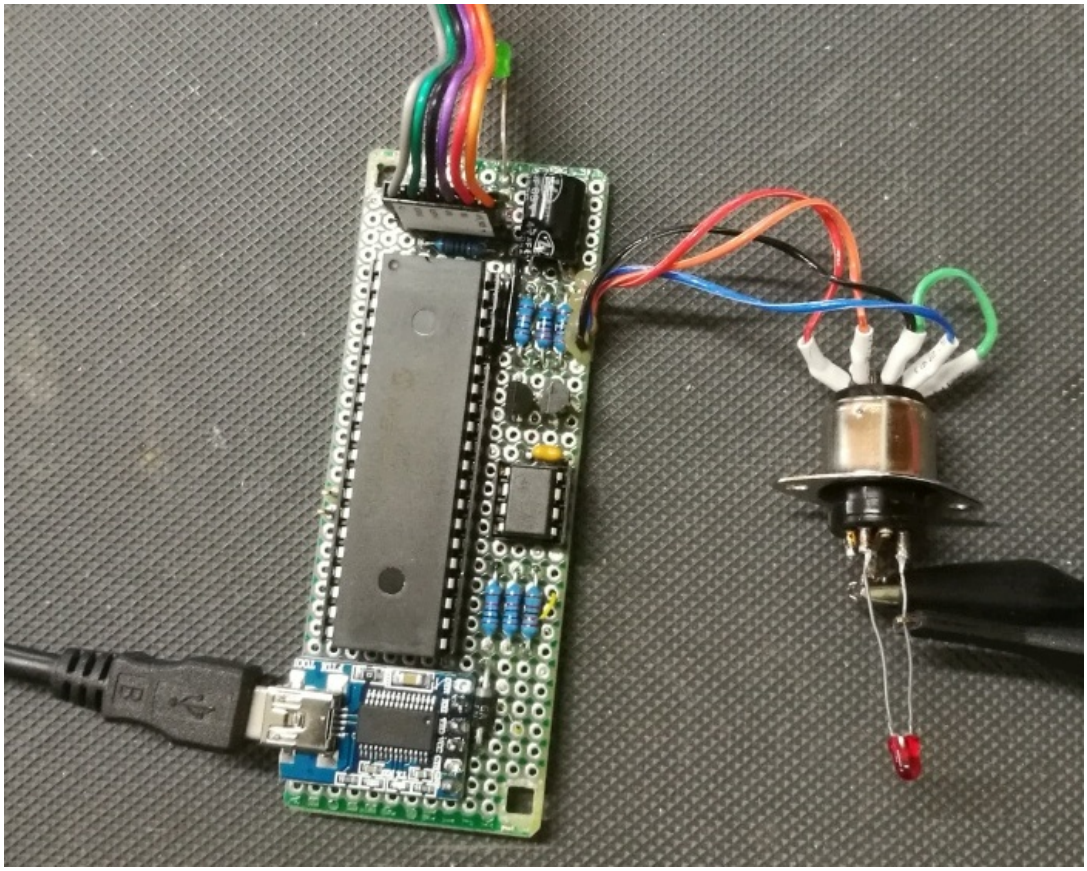


MCU circuit board top view with ICs not fitted in sockets.



MCU board under-side view showing wiring.
Leads to touch-pads, MIDI socket, sensors and other off-

The space inside the 40-pin MCU socket accommodates 5 resistors and the 8.2V zener diode.	board parts are soldered directly onto the board.



Assembled microcontroller board under test.

The picture above shows the MIDI OUT socket wired to the MCU board. The socket has a plug inserted with a LED wired between the OUT+ and OUT- pins. An oscilloscope was used to test the performance of the MIDI transmitter circuit, in particular for compliant rise and fall times in the output signal. The USB "service port" provides diagnostics for hardware testing.

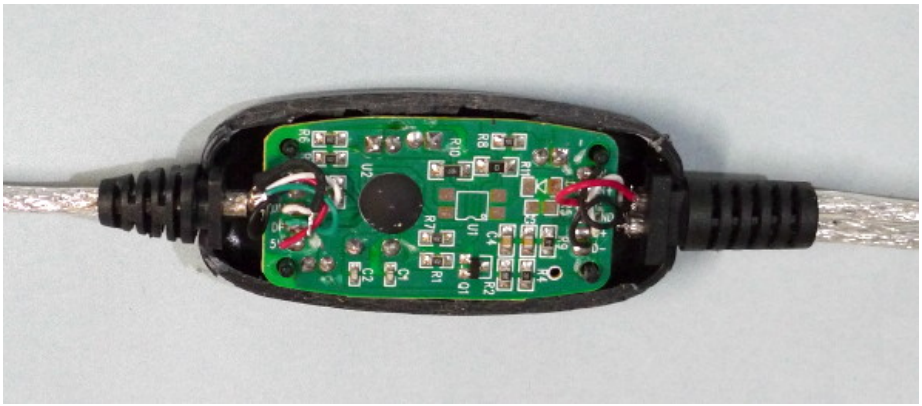
The USB/Serial bridge (FT232 breakout board) supplies 5V power to the REMI MCU (PIC18) when using the "service port" CLI, so there is no need to plug in a MIDI cable at the same time. There are many different types of USB/Serial modules available. Be sure to buy a device that outputs 5V on the Vcc pin and has 5V logic levels on the TX and RX signal lines. (Some devices are configurable for either 5V or 3.3V operation.)

Note: These pictures are provided as a rough guide to makers intending to build a REMI handset. Don't rely on the photo of the underside of the board as a guide to wiring. It's best to follow closely the schematic diagram (see download links below) and check your wiring carefully before applying power. It is also recommended to test MCU board operation before wiring it into the handset (as shown in the above photo). After soldering leads to off-board parts, e.g. MIDI socket, pressure sensor, modulation pad, PRESET button and touch-pads, apply hot-melt glue to the board around the entry points of leads to prevent them from breaking off.

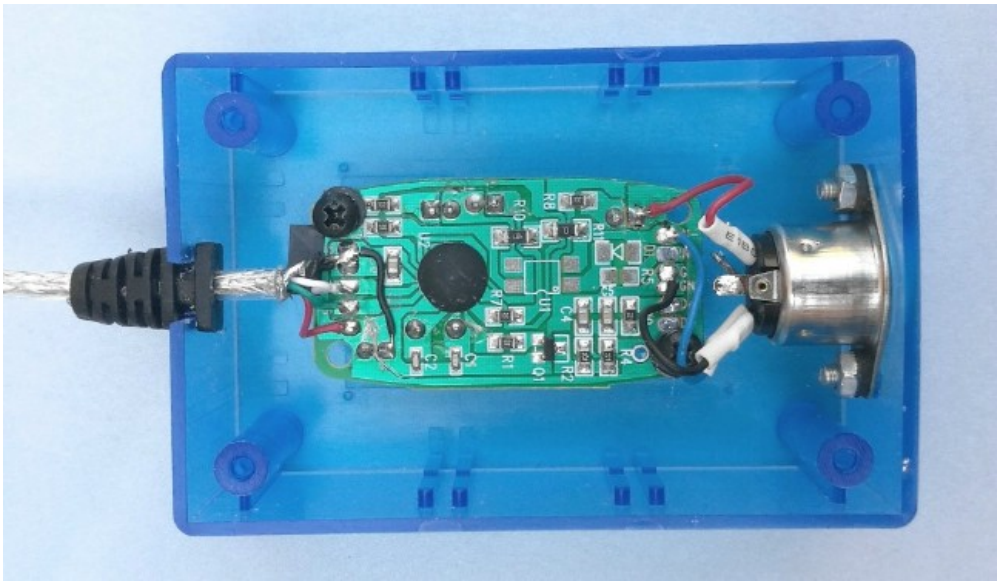
MIDI-USB Adapter box

The REMI MIDI-USB adapter is simply a commercial device re-fitted and re-wired into a new housing.
The adapter pictured below is readily available at very low cost (under A$10) from online suppliers.

The original adapter casing can be easily broken apart by cutting around the join with a hacksaw or utility knife, taking care not to damage the circuit board inside. The two MIDI cables with DIN-5 plugs on the end are unsoldered from the circuit board and discarded. The USB cable and connections are kept intact.



The circuit board is removed from its casing and mounted in a blue translucent box together with a 5-pin panel-mount DIN socket. A couple of plastic standoffs are glued into the box to hold the circuit board in place. A translucent box was chosen to allow the LED indicators to be visible through the top of the box without needing to drill holes for them. The assembled MIDI-USB adapter is pictured below.



REMI MIDI-USB adapter with bottom cover removed

REMI MIDI-USB adapter - DIN socket view

MIDI-USB adapter - USB cable entry

The table below gives wiring details for the DIN-5 socket. In the MIDI 1.0 specification, pins 1 and 3 are not allocated for any purpose. The REMI MIDI-USB adapter uses pins 1 and 3 to supply 5V DC power to the handset. Pin 4 is usually required for MIDI IN+ which connects to the anode of an opto-coupler LED. These cheap Chinese MIDI-USB adapters omit the opto-coupler, although there is provision for one on the PCB (U1). Hence, pin4 (IN+ on the PCB) is left unconnected.

DIN-5 Skt Pin	MIDI Adapter PCB
1	+5V
2	-
3	GND
4	-
5	IN-

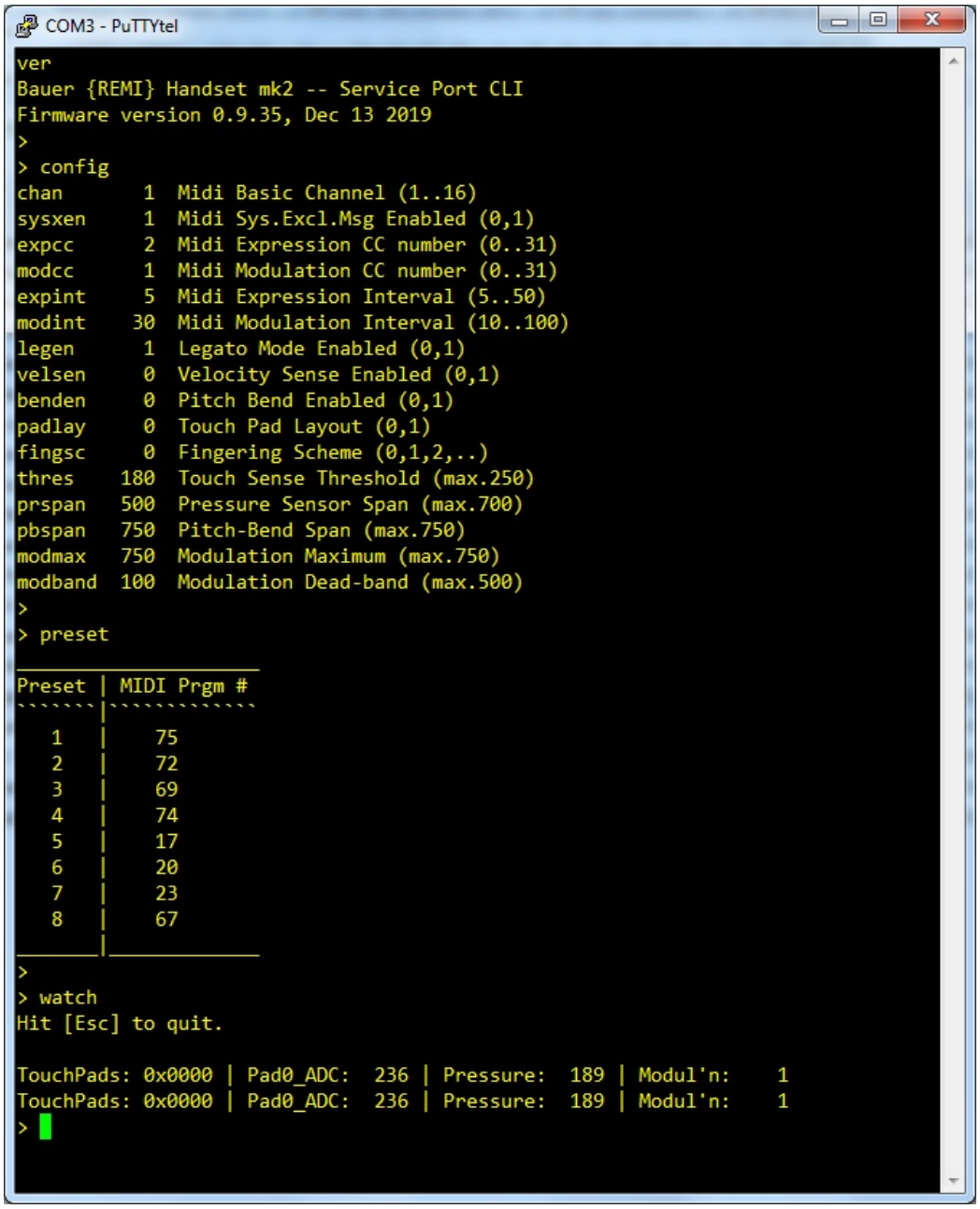
NB: Since a standard MIDI connection requires only pins 4 and 5 to be wired, some ready-made MIDI cables may have only 2 cores inside. Be sure to buy (or make) a MIDI cable with all 5 cores wired, for use with the MIDI-USB adapter and REMI handset.

Firmware

USB Service Port CLI

The screen-shot below shows CLI dialogue for a few of the commands, i.e. "ver" (version), "config" (lists user-settable configuration parameters), "preset" (lists Preset MIDI channel voice assignments) and "watch" (outputs data from touch-pads, pressure sensor and modulation pad in real time, for setup and testing).

Variants of the "config" and "preset" commands are provided to change setup parameter values and MIDI program/voice assignments (resp.). For example, to set the MIDI basic channel to 4, the user would enter the command: "config chan = 4".



To use the service port CLI, first download and install "PuTTY" (free terminal emulator app) on your computer. Plug your REMI into a USB port, then start PuTTY. If using a Windows PC, open the Device Manager utility and click "Ports - COM & LPT" to view connected devices. Note the COM port number of your REMI USB/Serial bridge (FT232) device.

In PuTTY, set up Serial port operation with the COM port number you found in Device Manager and set the baud rate to 38400. Set the Backspace key to 'Ctrl-H' (optional, but worthwhile). Open the serial port in PuTTY. Hit the Enter key on your PC and you should see a REMI startup message in the PuTTY window. Type the command "help" to get information on CLI usage.

Programming Tools

A PIC programming tool, e.g. Microchip PICkit-3, is required to install the REMI application firmware. Low-cost PICkit-X clones are available from online suppliers via AliExpress, eBay, etc.

REMI firmware is built using Microchip PIC development tools - MPLAB.X IDE with XC32 and XC8 compilers - free to download from Microchip's website. If you intend to modify or extend the firmware, you will need these tools. Otherwise, you just need to install the PIC programmer application (IPE) on your computer.

Downloads & Links