



# Introduction

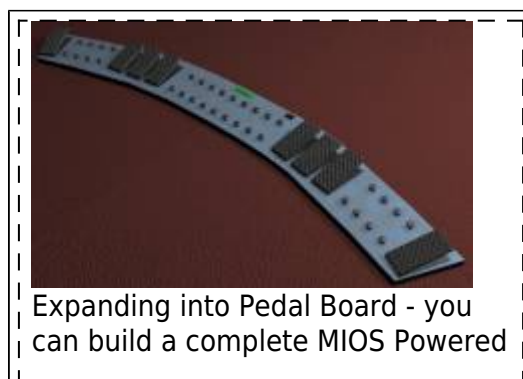
The Pedal Box was built to be a very basic extension to a floorboard. Giving it an extra 8 Expression Pedals, and a display to see patch change / CC names, as well as their values. It has grown far beyond it's original intention.

## Pedal Box



Is an expression pedal controller to expand an existing floorboard. In addition to adding up to 8 expression pedals, up to 8 relays can be triggered and the LCD display reflects the actions of your existing floorboard as well.

## Pedal Board





- Up to 128 patches can be listed in any order
- Cue Next / Cue Previous can be done from any button or a MIDI IN event
- PC Editor files allows easy...ish customization for different MIDI devices.
- Settings configurable from box. (this is a very slow process but can you help you out in pinch, it's far quicker to use the PC Editor).

## What's The Difference?

Basically Pedal Box is a MIDI Monitor / Processor. Pedal Board is a... well... floorboard.

	PbX	PbD
Display MIDI Input	Y	Y
Push Buttons	N	Y
Bank functions	N	Y
Rig Control Via MIDI	Y	Y
Rig Control Via DIN	N	Y
Relay Control Via MIDI	Y	Y
Relay Control Via DIN	N	Y
Gig Control Via MIDI	Y	Y
Gig Control Via DIN	N	Y
Setup Via MIDI	Y	Y
Setup Via DIN	N	Y
External Pedal Input	Y	Y
MIDI Merger	Y	Y

\* - Setup can still be done via the PC Editor and uploaded via MIDI

## LCD Display

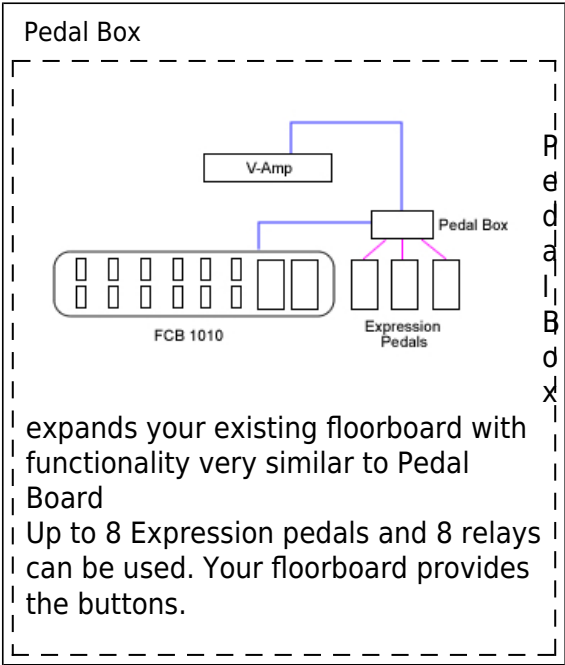




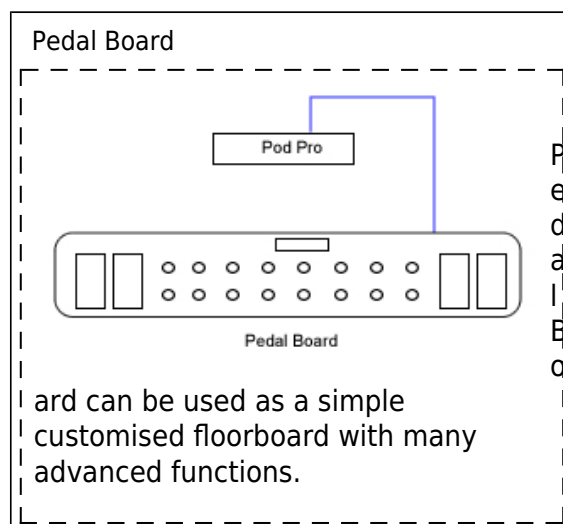
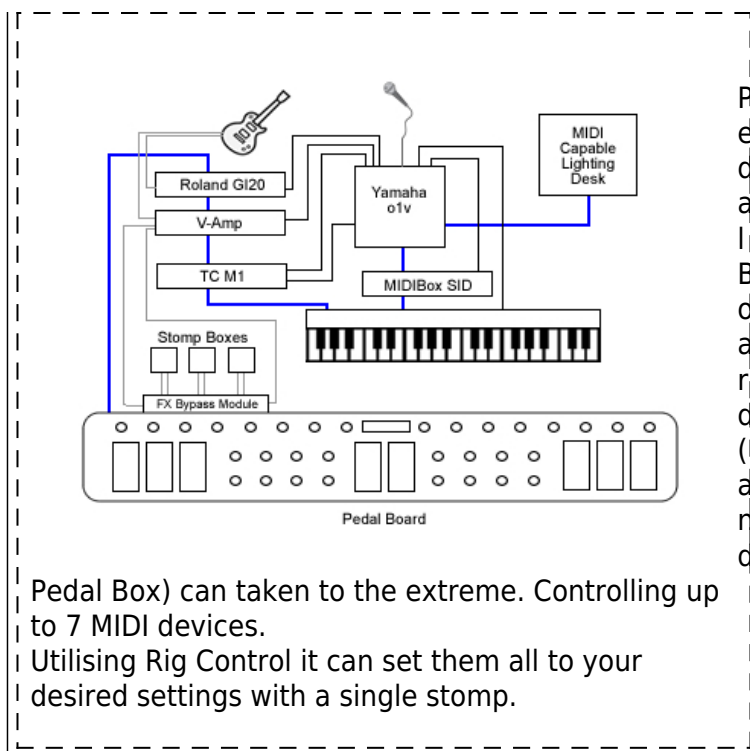
Here are some examples of what is displayed on the LCD.

	Program changes
	Controllers
	Controllers
	Controllers
	Controllers
	On/Off controls
	Tap Tempo Display

## Setup Examples



Extreme



## Manual

Pedal Box / Pedal Board [manual](#)

PC Editor Manual [manual](#)

Help can be found in the [Pedal Box forum thread](#)

## Building Pedal Box / Pedal Board

# Software

Pedal Box / Pedal Board has been written in C for PIC18F4620.  
The legacy version for the PIC18F452 is no longer available.

## Current Limitations

- No Sysex Support

## Installing

Firstly your PIC must be setup with bootloader and MIOS 1.9 or higher.  
You should be familiar with compiling 'C' files for MIOS.  
As well as uploading files to the Core module. See <http://www.ucapps.de/>

You need to upload the PB software, as well as fill the banksticks with the required info.

The first bankstick is used by PB with info about itself (Master Bankstick), the others are used to tell PB info about the devices it is controlling (Device Banksticks).

USE THE [PC EDITOR](#) to setup and upload the Master Bankstick data.

You can also see the [manual](#) for more information.

## Uploading device information

Device Banksticks tell PB about the MIDI devices it is controlling

In these banksticks:

- The device name
- The default MIDI channel
- The Event Map
  - MIDI event name and event type
  - 128 Entries of CC's, 128 of Program Changes
- 10 x Parameter 2 value maps
  - Value name and associated value
  - Up to 128 entries each (1 for each value, does not need all entries)

USE THE [PC EDITOR](#) to setup and upload Device Bankstick data.

## Uploading Pedal Box / Pedal Board

Now we are ready to upload Pedal Box. Browse to your Pedal Box directory and open the pbx\_config.h file.

You need to customise the application here. Comments inside in main.h will tell you what each setting

does.

Once you've customised the application it needs to be recompiled. Simply run re-make.bat.

Upload the created project.hex using MIOS Studio (or .syx using MIDI-OX or other midi software) file to the core module. MIOS will now reboot, and that's it!

### **pbx\_config.h in detail**

**PEDALBOARD** - Set to 1 if for Pedalboard mode, or 0 for Pedalbox mode

**ENABLE\_MIDI\_IN** - If you wish to utilise the MIDI IN port, set to 1. Otherwise 0

**MIDI\_MERGER** - To forward events recieved at the MIDI input to the output, set to 1. Otherwise 0.

**MIDI\_MERGER\_DISABLE\_MASTER** - If the MIDI Merger is enabled, but you do not want to forward messages targeting functions of PB (eg. A patch change) - set to 1. Otherwise set to 0 to forward all messages.

**AIN\_DEADBAND** - *(standard MIOS question)*

**AIN\_NUMBER\_INPUTS** - *(standard MIOS question)*

**DIN\_DEBOUNCE\_VALUE** - *(standard MIOS question)*

**DIN\_FIXED\_BUTTONS** - How many fixed or static footswitches are connected (0-16)

**DIN\_BANKED\_BUTTONS** - How many banked or dynamic footswitches are connected (1-16)

**DIN\_BANK\_UP** - The pin number assigned for bank up (2-33)

**DIN\_BANK\_DOWN** - The pin number assigned for bank down (2-33)

**FIRST\_PEDALSWAP\_DIN\_PIN** - The pin assigned to the first PedalSwap footswitch and LED *Be wary that placement does not interfere with Relays or LED Digit DOUT pins*

**NUM\_PEDALSWAP\_BUTTONS** - Number of PedalSwap slots available (max and default = 6)

**EVENT\_SETUP\_DIN\_PIN** - The pin number assigned for entering event setup mode (while holding bank down) (0-31) *Can not be the same as bank up or down*

**GLOBAL\_SETUP\_DIN\_PIN** - The pin number assigned for entering global setup mode (while holding bank down) (0-31) *Can not be the same as bank up or down*

**RELAY\_SHIFT\_REGISTER** - If your using relays, which SR are they connected to *if using SOME LED indicators, chosen SR must be after ALL POSSIBLE indicators*

**RELAY\_LED\_SHIFT\_REGISTER** - If your using LED indicators for relays, which SR are they connected to *SR should be AFTER the relay SR*

**USE\_LED\_INDICATORS** - Are you using LED indicators for each footswitch, set 1 for yes, 0 for no

**DIGITS\_CONNECTED** - Numver of LED digits connected (0-2). Set 0 for none.

**DOUT\_DIGITS\_SR** - Which SR is the first digit connected to.

### **Example of pbx\_config.h setup**

This is an example configured for:

- 4 expresson pedals, 8 fixed footswitches and 8 banked footswitches.

- There are 1 LED digit connected to shift register 3 (start at pin 25).
- There will also be 16 LED indicators for the footswitches .
- Bank down is at pin 16, bank up at pin 17.
- Special buttons for setup modes are the same as the first 2 fixed buttons.
- PedalSwap is activated by footswitches connected to pins 19 -24 (LED's on counterpart dout pins).
- Relays are located on Shift Register 4 (pin 32), and it's led indicators are on SR 5 (pin 40)

```

/////////////////////////////////////////////////////////////////
// Configuration Parameters
//
/////////////////////////////////////////////////////////////////

/* General Settings */
#define PEDALBOARD 1 // 1 = Pedal Board mode 0 = Pedal Box
mode
#define ENABLE_MIDI_IN 1 // 1 = yes 0 = no

#define MIDI_MERGER 0 // 0 = off 1 = on
#define MIDI_MERGER_DISABLE_MASTER 1 // 0 = disabled 1 = enabled - If
set will disable forwarding of events on the master bankstick channel

/* Pedal Settings */
#define AIN_DEADBAND 7 // 7 for 7-bit midi is best
#define AIN_NUMBER_INPUTS 4 // 1 - 8 - number of pots connected

/* Pedal Board Buttons */
#define DIN_DEBOUNCE_VALUE 20 // debounce value

// FIXED BUTTONS MUST CONNECT FIRST
#define DIN_FIXED_BUTTONS 8 // 0-16 buttons that always stay the
same, regardless of bank

// BANKED BUTTONS MUST BE CONNECTED AFTER FIXED BUTTONS
#define DIN_BANKED_BUTTONS 8 // 0-16 buttons that change with bank. 1
banked button will cause display errors when bank is over 99

//THESE SPECIAL FUNCTION BUTTONS MUST BE CONNECTED AFTER THE BANKED
BUTTONS!!!!!!
#define DIN_BANK_UP 17 // 2 - 33, pin number of button used
to bank up
#define DIN_BANK_DOWN 16 // 2 - 33, pin number of button used
to bank down

#define FIRST_PEDALSWAP_DIN_PIN 18 // 4 - 33, First PedalSwap DIN pin
(others will be next higher sequentially)
// Must be higher than Bank Up and Down
// Be careful not to set the same in the space of LED

```

*Digits or Relays*

```

// Best place is next to the bank up/down pins
#define NUM_PEDALSWAP_BUTTONS 6 // 0-6, Number of PedalSwap Buttons
connected

// if you wish to have access to all 6 via MIDI (with
less than 6 buttons connected
// - please set to 6 and leave the DIN pins free

// THESE SPECIAL BUTTONS CAN BE ANY BUTTON CONNECTED, THEY ARE DUAL USE.
#define EVENT_SETUP_DIN_PIN 0 // 0 - 33, pin number of button used to
enter event setup
#define GLOBAL_SETUP_DIN_PIN 1 // 0 - 33, pin number of button used to
enter AIN setup

/* Relay setup */
#define RELAY_SHIFT_REGISTER 4 // 0 - 7 Relay's Shift Register
#define RELAY_LED_SHIFT_REGISTER 5 // 0 - 7 Relay's LD indicator Shift
Register

/* LED setup */
#define USE_LED_INDICATORS 1 // Enable LED indicators connected 1 =
Enabled 0 = Disabled

/* LED Digit Setup */
#define DIGITS_CONNECTED 1 // 0-2 How many LED digits are connected
#define DOUT_DIGITS_SR 3 // What SR do the digits start from.
// 2nd digit will be on the next SR

////////////////////////////////////
// IF NO LED DIGITS ARE CONNECTED //
// A 2x20 LCD DISPLAY CAN BE USED //
// WITH THE BANK DISPLAYED ON THE //
// LAST 4 CHARACTERS //
////////////////////////////////////

```

**Download**

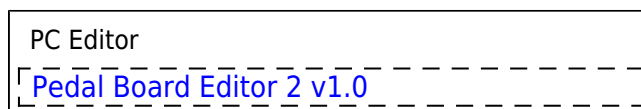
## Application

For PIC18F4620

[Pedal Box / Pedal Board v2.6beta1](#)

## Device Bankstick Downloads

[Behringer V-Amp Series](#)



## [The Complete Usage Manual](#)

---

## Known Bugs

Version 2.6:

- Possible issues with LED's not lighting up correctly
- (minor) Tap Tempo LED only functions when another led is lit up... cool huh?

Please report bugs in the [Pedal Box forum thread](#)

## History

### \* 2.6beta1

- Fixed critical rig control bug
- Fixed lagging with MIDI input while in Pedal Board mode. (May have wrecked led indicators in the process though)
- Relays can now be labeled and polarity set from the PB Editor software
- Reduced max cuelists to 20
- Added MIDI as AIN feature - allows a recieved CC to be translated as if it were an analogue input
- Added PedalSwap feature - temporarily change the settings of expression pedals

### \* 2.5beta3

- Fixed critical analogue input bug
- Fixed startup display giberish
- Added better relay support (hard code only - pbx\_relay\_config.h)
  - Virtual polarity (on when off, off when on option)
  - Relays can be named
  - When relays switch - can display something meaningful (like Input 1, Input 2 instead of just on, off)
- Probably some other stuff - I just can't remember :)

### • 2.5beta2

- Pedal Box Programming now enabled... oops
- Bug with LCD returning to show wrong current program after AIN fixed
- LCD will suspend for a short period after sending a program change

### • 2.5beta1

- Available only for PIC18F4620
- Re-structure of midi event and device event map handling
- Added midi event type options for DIN/AIN and Rig Control

- supports all standard midi commands without requiring them in the device event map
  - Device event map now for CC's and Program changes only, and must be listed in the appropriate slot.
  - Improved MIDI In performance 100 fold
  - Re-structure of LED handling
  - Improved LED performance dramatically
  - Lot's of bug fixes, general usage much more stable.
- 
- 2.0 (beta4)
  - Updated to new programming platform
  - Separate download for different PIC types
  - Added option to enable midi input for Pedal Board at the expense of LED Digits
  - Added bankstick magic numbers
  - Changed bankstick data format to accomadate magic numbers
  - Device Table Skeleton had incorrect bankstick address for value maps 9 and 10 - Fixed.
  - Rig Control Setup now loops back to the first message when after the last message during setup mode
  - Bankstick is correctly shown when programming during rig control setup
  - Bank now displays on the 4 characters of 2x20 LCD Display if not using LED digits
  - Revamped multiple device usage.
  - First bankstick is now for internal usage. all button / Pedal info, patches and song lists
  - Limit of devices is now 7 (not 8)
  - Bankstick usage has also been modified (v1 device files will not work without modification)
  - Added Rig Control, up to 32 events can be triggered at once across any loaded device
  - Added Gig Control. Song lists that change the patches in a given order with cue next / previous buttons
  - Added support for relays to be used for switching amp channels, etc.
  - Re-did a lot of code, trying to get code size down

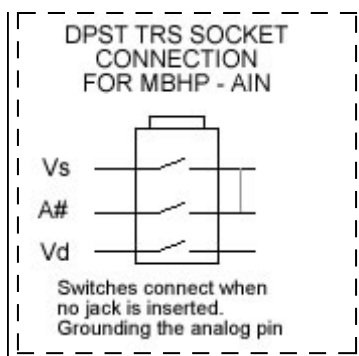
## Hardware

### Minimum Setup

- 1 x Core Module (PIC18F4620)
- 1 x 16x2 LCD Display
- 2 x 24LC256 Bankstick
- 1 x CV Pedal / Expression Pedal (or pot)
- A MIDI Floorboard / HUI (PbX) **OR** 1xDINX1 Module and 4 buttons (PbD)- *Required for editing settings after upload*

### Analog inputs





I use normalising (DPST) 6.5mm TRS Sockets.

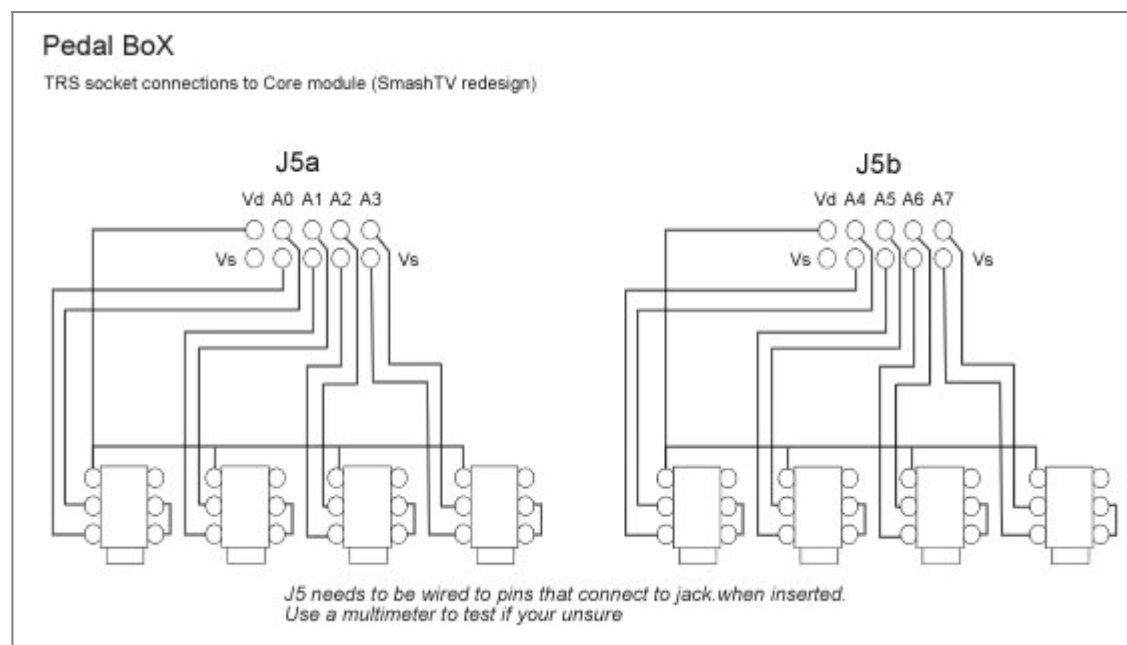
Tip = +5 (vd)

Ring = CV (A#)

Sleeve = Ground (vs)

Ring and Sleeve are connected together on the normalising side.

When a jack is not inserted, CV connects straight to ground (prevents AIN jitter).



When inserting/removing a jack, make sure the control voltage = 0 (pedal off). Or you will pass voltage straight to ground, causing mios to crash, and maybe cause damage to the circuit.

*If your using manufactured expresion pedals or other cv devices, check the pin assignment, some devices use a different setup.*

*With different setups, there is no way of stopping +5 being passed straight to ground while inserting/removing the jack (that i know of). Make sure you power off first!!!*

## Buttons / Foot Switches

Up to 34 buttons can be used with Pedal Board mode.

16 fixed buttons, 16 banked buttons, bank up and bank down

This gives a total of 144 possible midi commands.

The fixed buttons must be connected first (pin 0 up to 15), any amount 16 or less if fine (including none).

Fixed buttons are buttons numbers 1-16.

The banked buttons connect next (starting from the next available pin), again any number of banked buttons 16 or less is fine (including none).

They represent buttons 17-144.

The special function buttons pair up with bank down to access the setup menu's these can be any button aside from bank up or bank down. (pins defined in pbx\_config.h)

How many of each type of button is defined in pbx\_config.h.

The amount of banks available will vary with how many physical banked buttons you have. eg. with 2, there will be 64 banks, with 4 there will be 32 banks.

With 6 buttons there will be 22 banks. In bank 22, the first 2 buttons will represent button number 143 and 144, the 4 remaining will also represent 144.

If you don't have any banked buttons, bank display functions will also be disabled.

## Indicator LEDs

Buttons can have indicator LEDs displaying their current status.

LEDs connect to the first 1,2,3 or 4 shift registers, matching up with their DIN counterparts.

eg. SR1 DIN D0 = DOUT D7, DIN D1 = DOUT D6

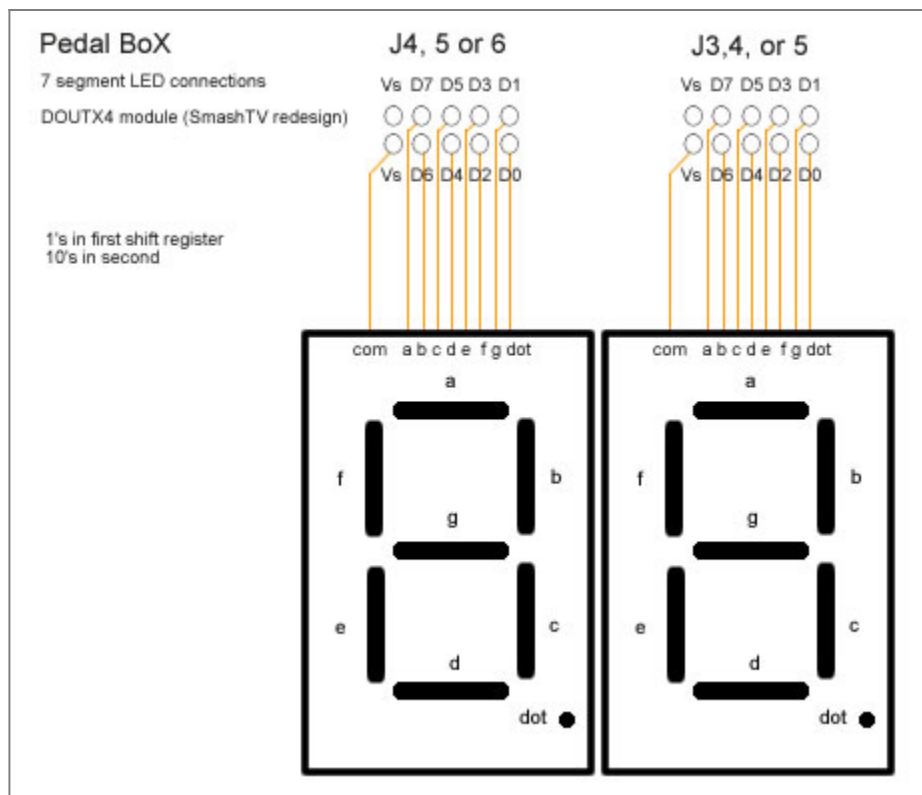
If you do not wish to use indicators for buttons, they must be disabled in pbx\_config.h

## 7 Segment LED Digits

2 seven segment LED digits can be connected to a DOUT module for bank display.

Pedal BoX does not use a LED matrix, each digit requires it's own shift register

Connect digits to the shift registers after LEDs, so if you only use 4 leds, still start at SR2 (or pin 8)



0, 1 or 2 digits can be used (defined in `pbx_config.h`). If no digits are used, the current bank is displayed on the LCD (requires a 2x20 display)

The shift register of the 1st digit (the one's display) must be defined in `pbx_config.h`  
 The second digit (ten's) will be automatically assigned to the next shift register.

In place of LED digits. A 2x20 LCD Display can be used (instead of 2x16). The last 4 characters of each line are used for bank display. This will only happen when `DIGITS_CONNECTED = 0` in `pbx_config.h`.

### Buttons, LEDs and LED Digits are only available for Pedal Board mode

## Relays

Up to 8 relays can be connected to trigger external switches. Relays can be triggered with a button (PbD), a midi event (PbX) or within Rig Control (both).

Relays can also have separate indicator LEDs.

Relays connect to any free shift register (defined in `pbx_config.h`). The relay indicator LEDs are also assigned their own shift register.

NOTE: Be sure that you use free shift registers. Remember if you use LED indicators for buttons, the shift registers taken up by buttons are the same for their LED's. If not all buttons have LED's you must leave the DOUT pin's free on any pin that has a button connected on it's DIN counterpart.  
 If you don't use button indicator LED's, they can be disabled in `pbx_config.h`, freeing up the DOUT shift registers for use by relays or LED digits.

For information on how to connect relays to a DOUT module refer to the midio128 notes at ucapps:

- [relay\\_example.pdf](#)

- [mbhp\\_doutx1\\_reed\\_relays.pdf](#)

Or this forum thread about creating an FX Bypass/Looper

- <http://www.midibox.org/forum/index.php?topic=11705.msg96483#msg96483>

## Connection Overview




Pedal Board Test Bench



## Where to get Pedal Board parts

I only built Pedal Box, so I use the Behringer FCV100 Foot Pedal in CV mode. But here's some links that came up on the forum.

- [Banzai Effects](#) stock a good range of momentary footswitches - perfect for this project
- [Small Bear Electronics](#) Specializes in DIY guitar sound effects
- stryd\_one offered this google search for momentary footswitches.  
<http://www.google.com.au/search?safe=off&q=momentary+SPST+footswitch>

If you build Pedal Board please 

## To Do - Future Plans

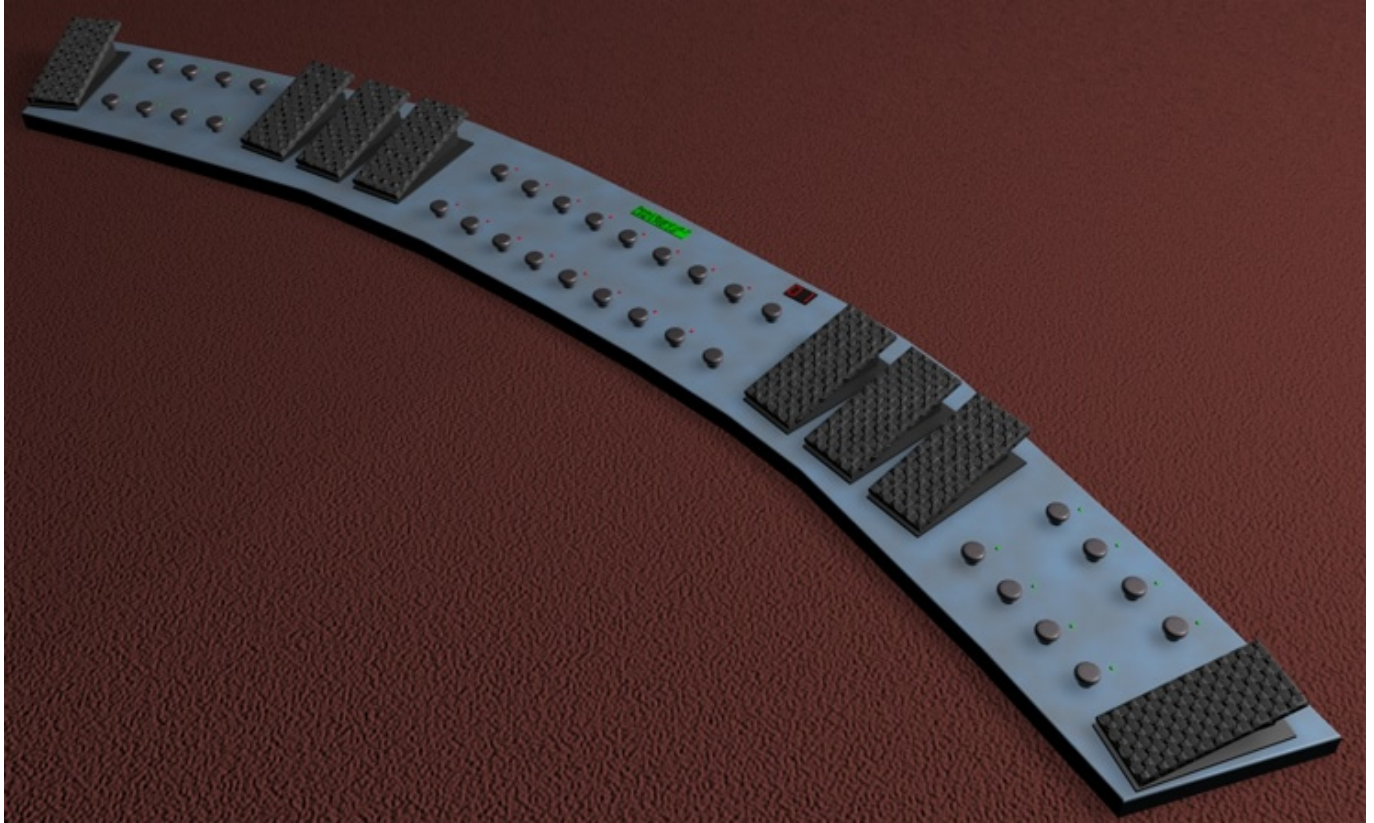
- Allow control of Midi devices without using banksticks. So default numbers only setup is possible
- Add MIDI filter
- Add IIC Midi module so feedback can be recieved from the device (if running in pedal box mode)

with separate floorboard)

- Probably a lot more. I expect this will be an ongoing project

# Inspirado

Here's a vision I had:



You know you want to build it!

From:

<http://www.midibox.org/dokuwiki/> - **MIDIbox**

Permanent link:

[http://www.midibox.org/dokuwiki/doku.php?id=pedal\\_box&rev=1271144401](http://www.midibox.org/dokuwiki/doku.php?id=pedal_box&rev=1271144401)



Last update: **2010/04/13 08:40**