MIDI SWITCHER

Introduction

MIDI Switcher allows you to control electrical devices such as solenoids, motors, relays, LEDs and light bulbs using MIDI messages.

It is ideal for sequencing electromechanical musical instruments, controlling light shows from your DAW, making crazy robotic sound art installations or just about anything else you can think of!

There are some things it can't do however. Before we get into details, let's make sure these limitations are clear.

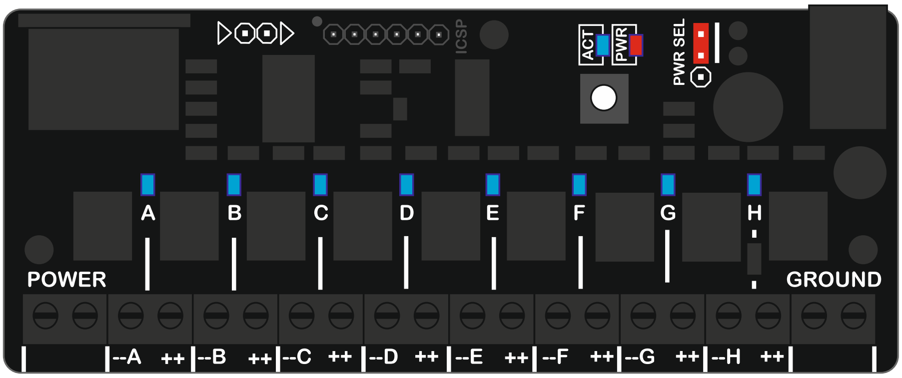
- Firstly, and most importantly, **THIS MODULE CAN NOT BE USED TO DIRECTLY SWITCH AC MAINS ELECTRICITY!!** Never even think about connecting the mains directly to this module. The module will be destroyed and so might you be.

- This module can only switch DC (direct) current. It cannot be used to directly switch signals such as audio.

- Switching is always between the "low side" (negative terminal) of a connected load (such as a solenoid or light bulb) and ground. This may make it unsuitable for directly switching currents that are not referenced to ground (an example might be a channel switching footswitch on a guitar amp)

- The above limitations can be worked around using external relays, such that the MIDI switcher turns the relay on and the relay makes the actual connection to the signal or current being switched.

Features of the Switcher



The MIDI Switcher has eight switching outputs using transistors that are rated for 8 Amps or 100 Volts DC (Power rating 24W). However, when switching large currents or voltages there are some special precautions to observe which will be explained later.

Each output can be configured to be triggered based on the following conditions

- **MIDI notes (with optional velocity range filter)**
- **MIDI continuous controller message (with selectable value range)**
- **MIDI program change (can set all outputs with a single message)**

The switch output pulse can be "shaped" in a number of ways

- **You can control the switch pulse duration based on how long the trigger condition lasts (e.g. how long a MIDI note is held) and a selectable amount of time.**
- **The "power level" of the switching pulse can be controlled using pulse width modulation (PWM). Four of the outputs (E,F,G,H) use special hardware features to support highly stable PWM at 1kHz.**
- **Both the pulse duration and power (PWM duty) can be modulated based on either the velocity of incoming MIDI note or the value of a selected continuous controller (CC)**
- **A "release" can be defined so that power (PWM duty) falls down to zero during the switch duration (good for LED fades)**
- **Gamma correction can be applied to power (PWM duty) to improve the brightness scale of LEDs and light bulbs.**

The configuration options are selected using a form on a web site, which then encodes them into a MIDI System Exclusive (Sysex) file that is sent to the switcher to reconfigure it. The settings are retained at power-off

The LEDs

- The PWR LED is on all the time the board has power. If you are powering the board and the LED is off, check the position of the PWR SEL jumper (see below)
- The ACT LED flickers when there is MIDI activity. It also blinks slowly in midi lockout mode (see below)
- The eight LEDs A-G show the status of the switcher outputs. The LED is on when the output is switched, and the LED brightness shows the PWM duty.

The Button

The button on the MIDI switcher has the following functions

- A short press will reset all the outputs to their power-on state
- A long hold will cause the switcher to go into midi-lockout mode. The activity LED will regularly switch on and off and all MIDI input will be ignored. This mode is useful if you have multiple switchers connected to a common MIDI input and only want to access one of them at a time (e.g. for sending a configuration sysex)
- Holding the button at power up causes the switcher to enter bootloader mode, which allows new firmware to be installed over MIDI. Do not do this if you simply want to load a configuration patch!

PWR SEL jumper

The MIDI switcher has two power circuits

- The microprocessor and associated electronics run from the 2.1mm socket on the back of the board. This socket can accept a 7-30V DC (positive centre) supply.
- The devices (or "loads") being switched are powered from the POWER and GROUND terminal blocks on the front of the board (each of these terminals has two identical screw connectors, either or both of which can be used).

When the PWR SEL jumper is in the top position (closest to back of board, the default position), these two power circuits are connected together. This enables both the circuits to run from the same supply, via the socket or the terminal blocks.

When PWR SEL is in the top position please note that

- The power socket should only be used if the current needed by the loads is 2 amps or less. For higher currents the terminal blocks should be used.
- Do not exceed a 30V DC supply

When the PWR SEL jumper is in the bottom position the two power circuits are disconnected from each other. This means that a separate power supply needs to be plugged into the socket to power the microprocessor.

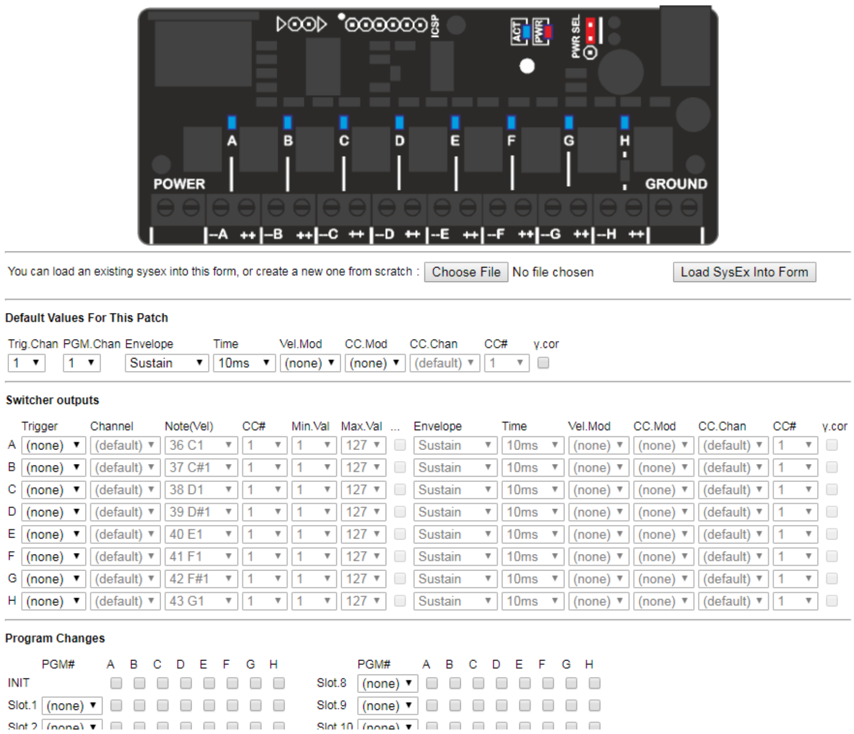
This scenario allows voltages higher than 30V to be switched. The transistors are rated up to 100V DC.

The Configuration Web Form

The MIDI switcher is configured using the configuration form at

http://six4pix.com/switcher/patch.asp

MIDI SWITCHER SysEx Builder



The page has three sections

- **The Default Values section allows common settings to be defined just once and referenced from multiple other places, saving time and repetitive entries**
- **The Switcher Outputs section allows the trigger condition and pulse shaping for each of the eight outputs to be defined**
- **The Program Changes section allows combinations of outputs to be switched on in response to specific MIDI program change messages**

At the bottom of the page is a button that submits the form and downloads a sysex file that can be sent to the switcher using a DAW or Sysex manager program such as MIDI-OX (Windows) or Sysex Librarian (Mac)

The form also lets you upload an existing Sysex file and display or edit its content.

MIDI Note Trigger

When an output is set up to switch based on a MIDI note, it switches on when the MIDI note starts playing. Depending on settings, the output switches off when the note stops playing, or after a period of time. This will be explained later

Here is an example of how a note trigger is set up using the configuration web page

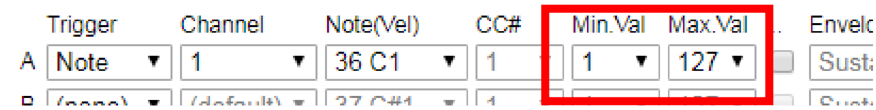
In this case a C1 note (C note, octave 1) on MIDI channel 1 will trigger output A. This particular note is MIDI note number 36.

When this note is played, output A will be activated (will turn on). The output will switch off at a later time defined by the "pulse shaping" options, which we'll look at later.

Velocity Filtering

As well as pitch information, MIDI note messages have a "velocity" (i.e. how hard a key was hit), which is a number from 0 (note off) to 127 (full velocity)

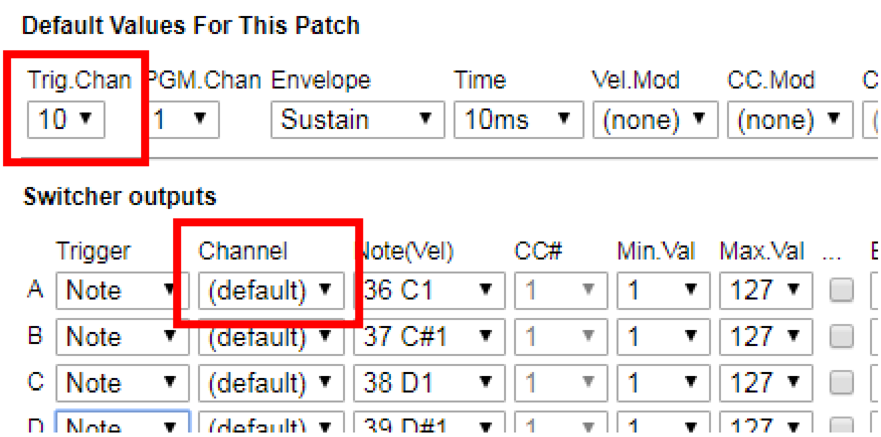
The MIDI switcher lets you check the velocity so you can (for example) only trigger the output if the velocity is above a certain level or between certain bounds.



For a note trigger the "Min.Val" and "Max.Val" refer to the note velocity. By default this is set to 1-127 so that any velocity higher than 0 (note off) will trigger the output (i.e. note on)

Default Trigger Channel

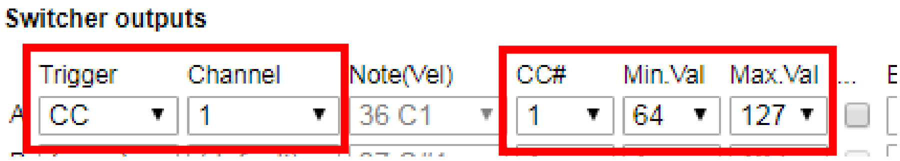
A common scenario is that each output will be triggered by different notes on the same MIDI channel, but you might want to change that MIDI channel easily. For this reason the MIDI Switcher supports a default MIDI channel for Triggers. Select (default) as the channel for each output that uses the default trigger channel, e.g.



MIDI Continuous Controller (CC) Trigger

When an output is set up to switch based on a MIDI CC, it switches on when a CC message is received that matches the controller number and where the controller value is between specified limits.

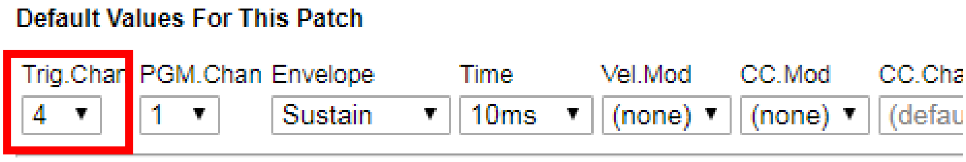
Here is an example of how a CC trigger is set up using the configuration web page

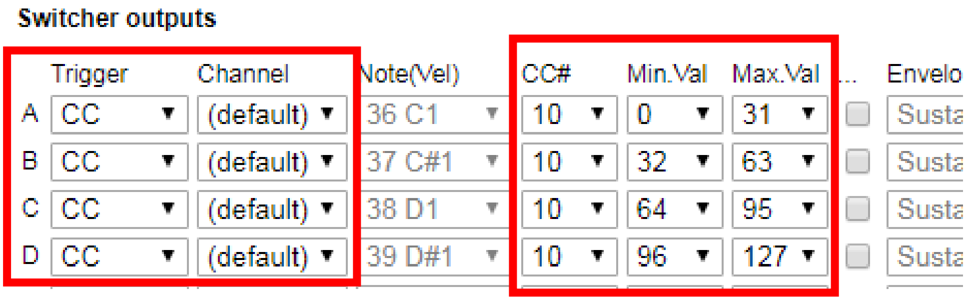


In this example, the switcher output is triggered on when CC#1 (modulation wheel) on MIDI channel 1 is between 64 and 127 (i.e. is greater than or equal to 64).

A CC trigger condition starts when the CC first has a value within the specified range and lasts until the CC has a value outside the range.

Multiple outputs can reference different ranges for the same CC and the default channel works for CC based triggers in the same way as for notes. For example





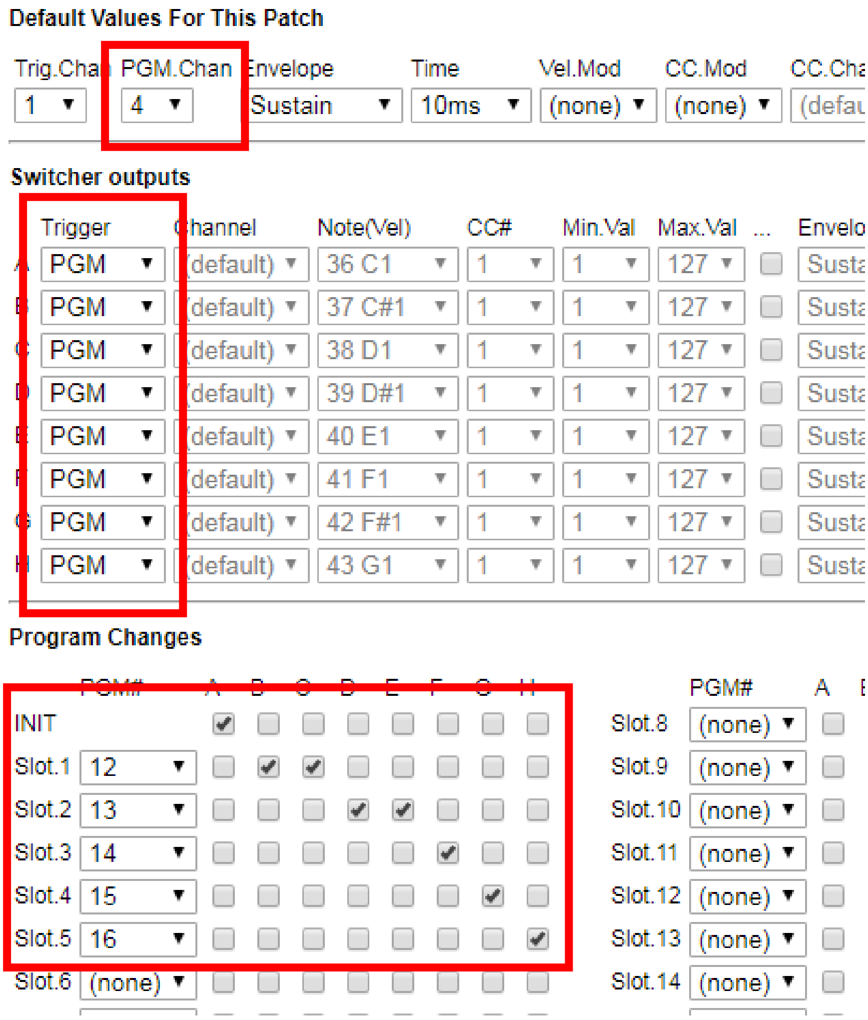
CC #10 on MIDI channel 4 will trigger outputs A, B, C and D depending on the value of the controller.

Program Change Trigger

The MIDI Switcher is able to listen for Program Change messages (these messages are usually used to select a patch on a synth). A program change message has a MIDI channel and a program number 0-127.

To trigger on program change messages you first need to configure one or more outputs in PGM trigger mode and select the PGM MIDI channel in the default section. Only outputs selected for PGM mode will respond to program change messages.

There are a total of 15 user-defined slots available to match program change messages. When a MIDI program change message matches the channel and program number for a slot, outputs be “triggered” or “untriggered” based on the selections in that PGM slot. Only outputs in PGM trigger mode will be affected, even if you check the boxes for other outputs.



Program change messages which do not match any slot are ignored. If multiple slots match the same program number only the first will be actioned.

Program change triggers are still subject to pulse shaping. To have the outputs remain on continuously until another program change switches them off, they should be in the “Sustain” envelope mode.

The INIT settings are triggered at power on or when the outputs are reset.

No Trigger Required!

The "Always" trigger mode means that a given output is triggered at power on and when the reset button is pressed. By setting the envelope type to Sustain, this means the output is on all the time.

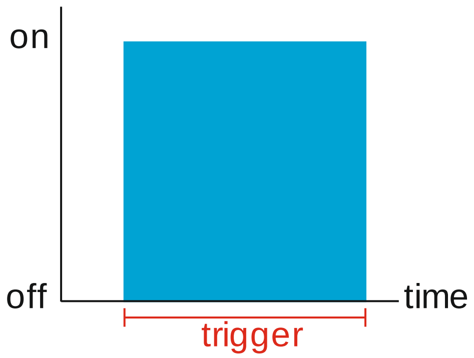
This can be useful if you simply want to control the power of an output using a CC and without having to send a note message to trigger it.

Pulse shaping

The outputs of the MIDI switcher turn “on” as soon as their selected trigger condition is met.

However, there are several settings that control the time it takes for the output to turn off again, and the power level delivered by the switcher output during the time it is switched “on” (i.e. the “shape” of the power pulse delivered to the connected load)

In the simplest scenario, the output is switched on for the duration of the trigger condition. For example a key on a MIDI keyboard might be pressed then released and the MIDI Switcher responds as follows



In some situations this might be all we need, but in others we might have specific requirements about pulse length – for example a solenoid controlled beater might hit a drum “softly” if triggered for 50ms and “hard” if triggered for 200ms.

From a sequencer, we could send our MIDI notes with exactly the correct “note on” duration, but from a MIDI keyboard it would be much more useful to use the MIDI velocity to change the duration so that if we press the key harder it will activate the output for longer.

In that case the actual length of the note as played is not important. Instead we want the output to be switched for up to 200ms depending on the note velocity. The MIDI switcher enables use to do this, for example:

Here the “Hold” setting for “envelope” means that the output is switched based on the selected Time only, rather than the time the MIDI note is played for. There are several envelope options that will be explained below.

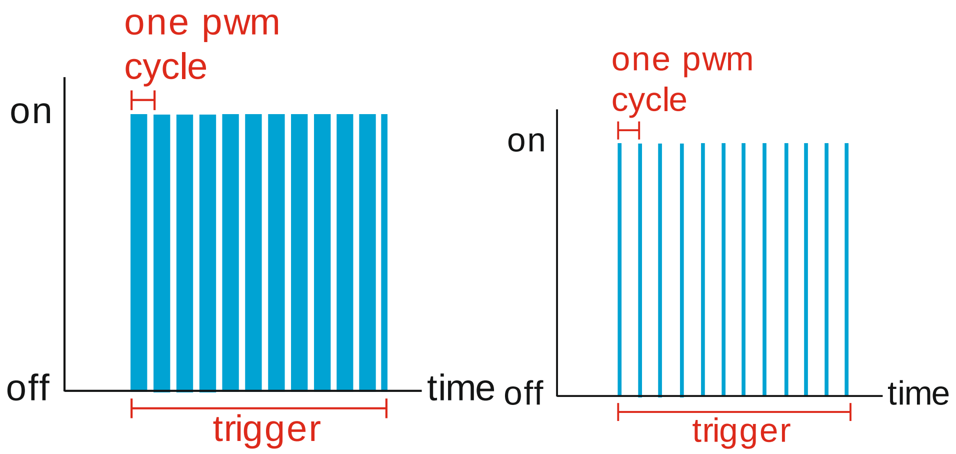
Pulse Width Modulation

As well as controlling the duration of the output pulse, we can control the effective "power" of the pulse via something called "pulse width modulation" (PWM).

When PWM is used, the MIDI Switcher turns the output on and off very quickly (up to 1000 times a second) while it is triggered. This means that the effective power delivered to the load is less than it would be if the output was solid on. By changing the ratio of the ON and OFF times (called the "duty") we can deliver a whole range of powers to the load.

Here is a diagram showing PWM in action over the duration of the trigger pulse. The shaded part is when the output is "on". By looking at one "cycle" we can see its about 75% of the time in the first picture and about 15% in the second.

The amount of power delivered to the load is reduced in the same proportion (as compared to output being on solidly)



Since the output is being turned on and off, it is generally desirable for this to happen quickly so that there is no visible effect in the load (e.g. visible flickering of a light bulb)

Switcher outputs E/F/G/H use a special feature of the microprocessor to achieve stable PWM signals at a little under 1 kHz.

On outputs A/B/C/D the PWM is emulated in hardware and runs at around 160Hz and can be less stable when the switcher is busy (e.g. processing a lot of MIDI).

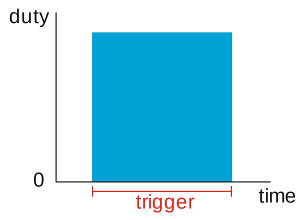
Therefore, outputs E-H should be used if possible where PWM stability is important (such as when driving LEDs, where instability can result in flickering of the LEDs)

Just like the hold time, the PWM duty can be "modulated" by both the velocity of received MIDI notes and/or by a selected MIDI continuous controller (CC)

PWM duty can also be automatically "faded" during the hold time by selecting "release" envelope types. The following sections describe each of the envelope options.

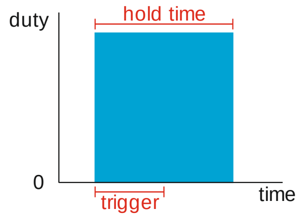
Sustain Envelope

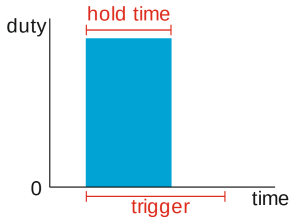
In this mode, the length of the switching pulse is controlled only by the length of the trigger condition and the hold time is not used.



Hold Envelope

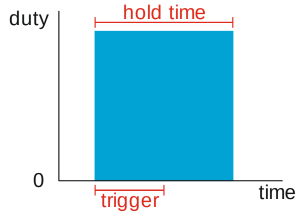
In this mode, length of the switching pulse is controlled only by the hold time and the length of the trigger condition is not used. The trigger condition might last a longer or shorter period than the hold time but the output is always switched on for the hold time

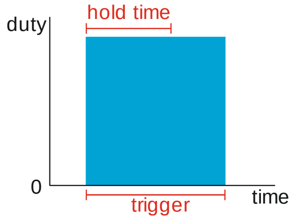




Hold->Sus(tain) Envelope

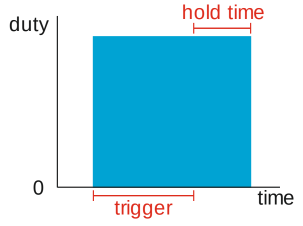
In this mode, length of the switching pulse is controlled first by the hold time and then by the remaining length of the trigger condition. The output is therefore switched for the longest of either the trigger condition or the hold time.





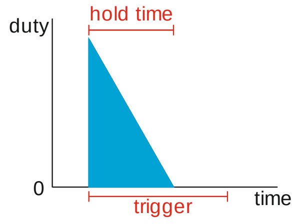
Sus(tain)->Hold Envelope

In this mode, length of the switching pulse is controlled first by the trigger condition and then by the hold time. The output is therefore switched for the total of the trigger condition time and the hold time.



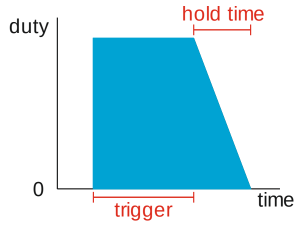
Release Envelope

In this mode, length of the switching pulse is controlled only by the hold time and the length of the trigger condition is not used, so it is similar to Hold mode. However, during the hold time, the output duty is reduced linearly down to zero.



Sus(tain)->Rel(ease) Envelope

In this mode, length of the switching pulse is controlled first by the trigger condition and then by the hold time, so it is similar to the Sustain->Hold mode. However, during the hold time, the output duty is reduced linearly down to zero.



The release modes are most useful with light bulbs and LEDs where some kind of “fade” is desirable before the output switches off.

Gamma Correction

PWM can be used effectively to control the brightness of LEDs and light bulbs, however our eyes tend to become less sensitive to changes in brightness when the bulb is bright, versus when it is dim.

The MIDI switcher has a way to scale brightness to make best use of the range, by using a technique called “gamma correction”. Since the issue is specific to our vision, this is only really useful when fading light sources such as LEDs and filament bulbs and is not intended to be used when controlling power to other types of device.

Gamma correction is enabled by checking the following box

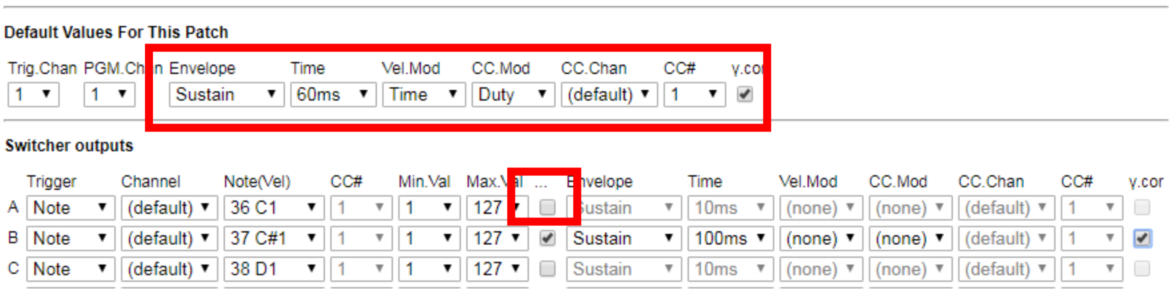
Default Pulse Shaping Settings

The MIDI switcher provides an easy way for multiple outputs to share the same pulse shaping settings, saving them from having to be selected individually for every output.

For a given output, if the checkbox to the left of the pulse shaping settings is not checked then the default settings in the top section of the form are applied to that output.

If the box is checked then the output has its own settings, which must be entered in the fields following the checkbox.

In this example outputs 1 and 3 are using the default settings but output 2 has its own defined



Connecting the Switcher

NEVER CONNECT THE SWITCHER DIRECTLY TO MAINS ELECTRICITY!!!

The MIDI switcher can control power to DC loads such as

- Solenoids
- Motors
- Relays
- Light Bulbs
- High Power LEDs (ensure correct voltage and heat-sinking)

Some of these loads might draw quite a lot of electrical current, so it is important to

- Ensure that your power supply is adequate to power the loads
- Ensure that the current being routed via the MIDI Switcher is within its capabilities
- Ensure that components (such as the transistors on the switcher) are not becoming hot during use. They might get warm, but must not get uncomfortably hot to the touch.

Connecting Loads

Loads (devices to be powered) are usually connected between the + and - screw terminals of each switcher output.

The MIDI Switcher works by switching the connection between the (–) terminal and power ground, so power to the load is switched on the “low” side only. The “high” (+) side of the load is permanently connected to the supply.

Voltage Rating

Each output of the MIDI switcher uses a transistor (MJD122) that is rated for 8A/100V DC with 24W power dissipation

When the PWR SEL jumper is in the top position, the maximum voltage that can be handled by the board is 30V DC, regardless of whether this is via the power socket or the screw terminals.

When the PWR SEL jumper is in the bottom position, the screw terminals for power at the front of the board can be used up to 100V DC

Current Rating

When power is supplied via the screw terminals, the board should be able to handle total continuous currents of 8A and higher currents for short periods. **This is a recommendation based on estimates and has not been destructively tested. Always ensure that the board is not overheating during use!**

When power is supplied via the socket, the total current is limited to 2A and maximum 30V DC.

Always check the PWR SEL jumper is in the bottom position before applying a supply voltage above 30V DC to the screw terminals (a separate lower voltage supply will be needed via the socket)

Always ensure that polarity is correct when powering via the screw terminals.

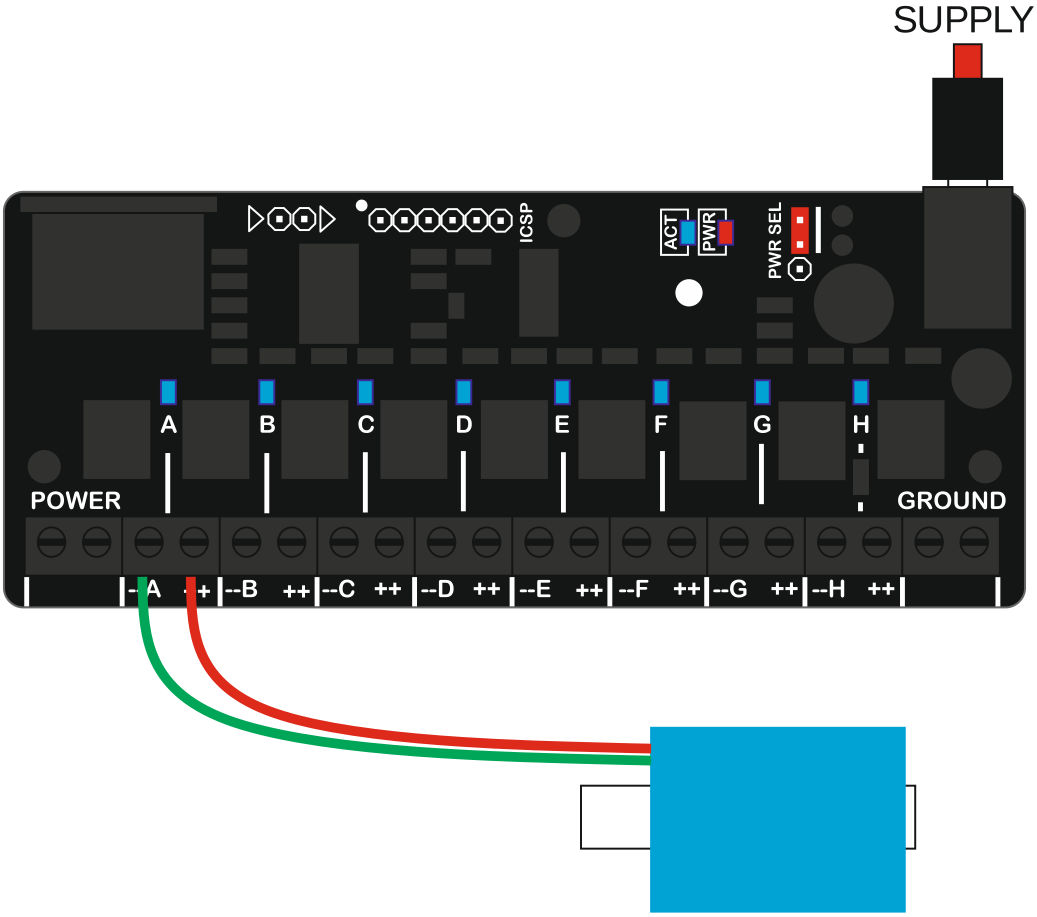
Example Connections

Low Power Scenario (Up to 2A and 30V DC)

Here all power is being received via the 2.1mm power socket. The PWR SEL jumper is in the top position so that power is shared with the switcher outputs.

Each load is connected between the (+) and (–) terminals for the appropriate switcher output (here a single solenoid on output A).

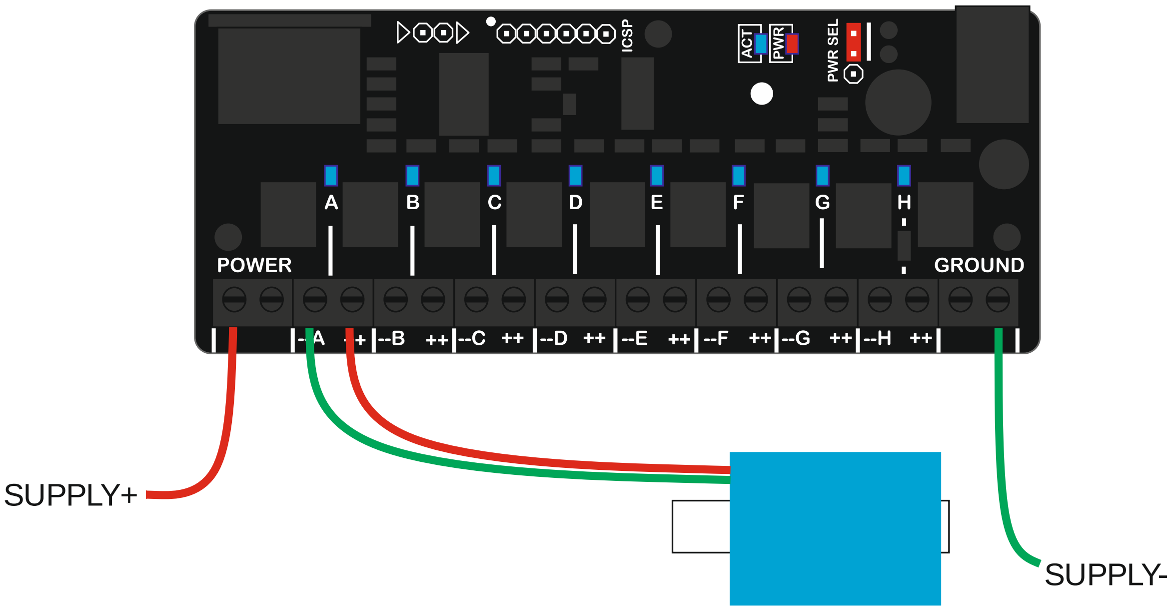
The 2.1mm socket can take a maximum supply voltage of 30V and a continuous current of 2A



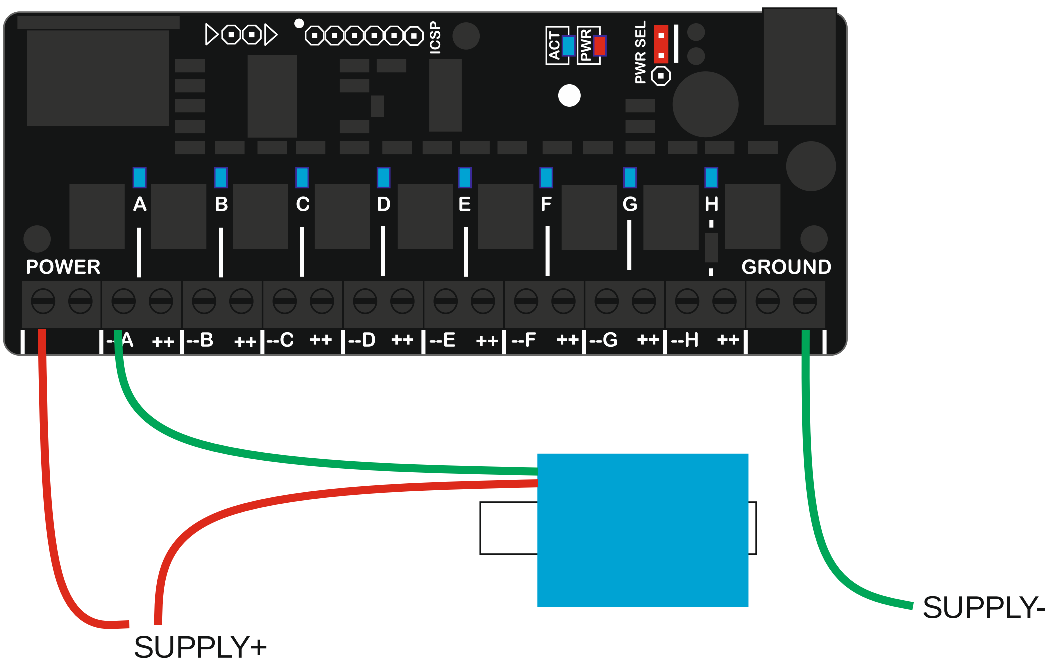
Higher Current (Up to 8A and 30V DC)

The second scenario is suitable for loads which draw higher current and where the power supply does not exceed 30V. In this scenario the PWR SEL jumper remains in the top position so that the microprocessor on the board can be powered from the front power terminals.

Note that the POWER terminal block has two identical (+) connections and the GROUND terminal block has two identical (-) connections.

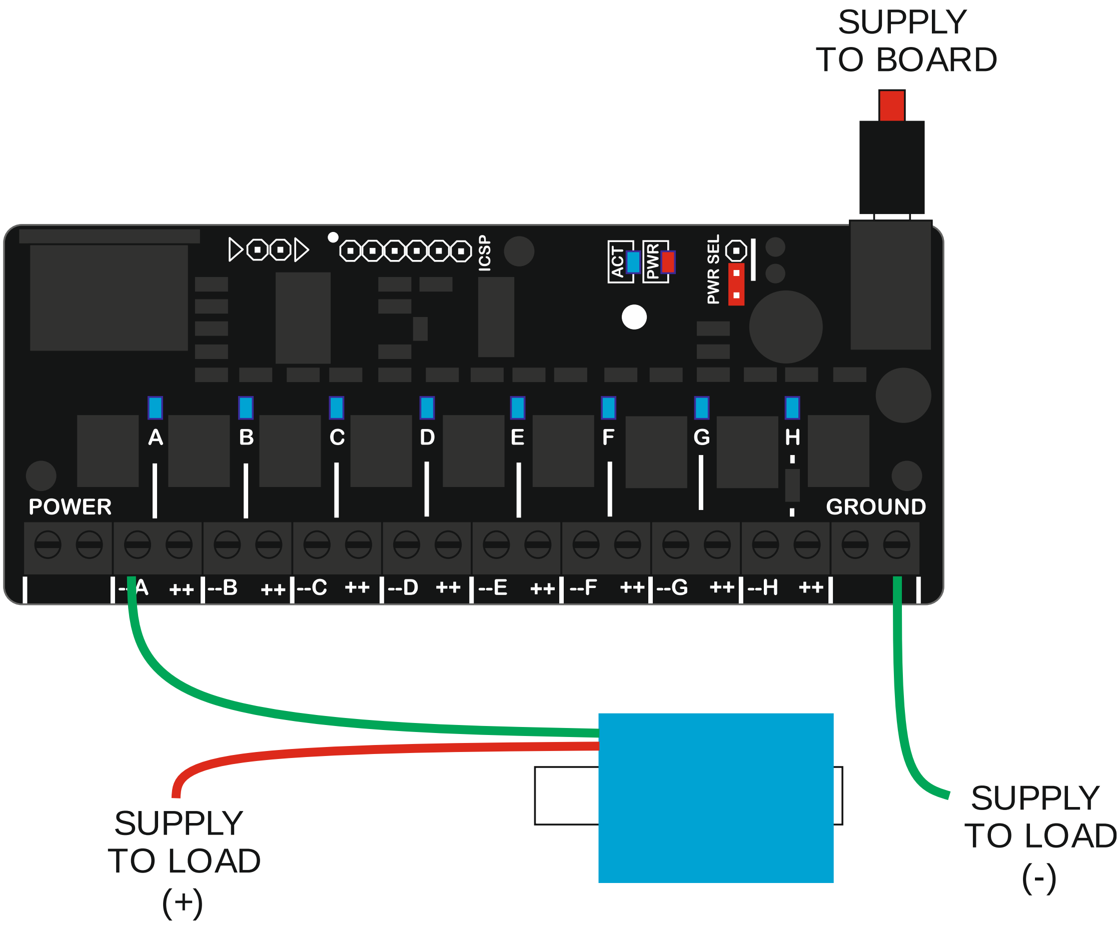


Alternatively the "high side" of each load can be powered directly without going via the board, which may simplify wiring in some situations, since only one wire needs to run from each load to the board. This might also help in high current situations since only the current to ground is flowing through the board.



Higher Voltage (Up to 100V DC)

In situations where the supply voltage to the loads exceeds 30V, the loads must be powered separately from the board. **The PWR SEL jumper is moved to the lower position** and a separate supply between 7-30V is attached to the 2.1mm power socket to power the microprocessor circuit.



Here the high side of the load is connected directly to the power supply and the ground return path is via the board. The voltage can be up to 100V and each output can handle 8A of current. However the **total continuous** current through the board should not exceed 8A.

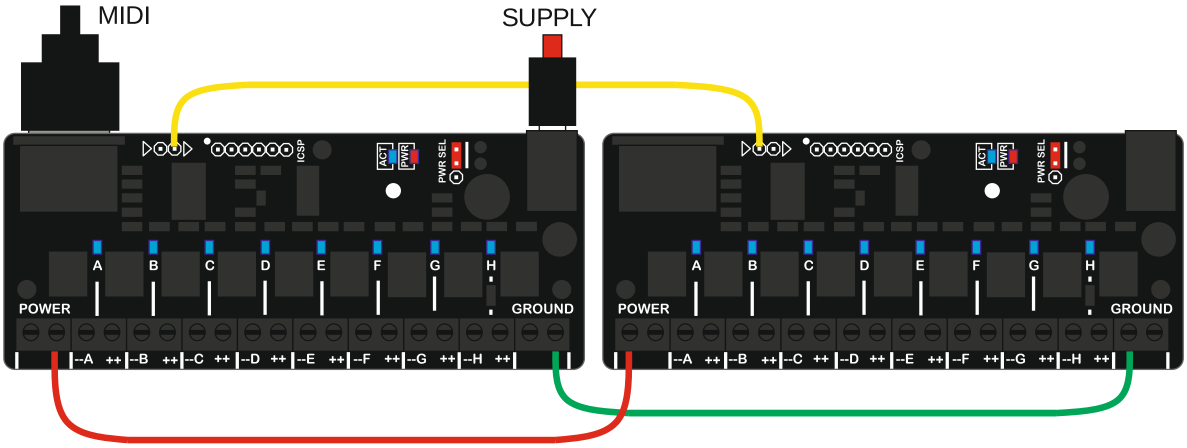
If preferred, the (+) side of the power supply can be connected to the POWER terminal at the left side of the board (**the PWR SEL jumper must be in the lower position**)

Working with Multiple Switchers

In some situations, you might need to switch more than eight outputs. In these cases you can use multiple MIDI Switcher modules, each configured to listen out for its own triggers, then you can send the same MIDI input to each of the switchers.

Sharing A Single MIDI Input

If you have only two modules you can have them share a single MIDI input and power supply as shown below.



The important point here is that both modules have a **common ground** connection (shown in green) and that there is a connection (shown in yellow) between the **right hand pin** of the MIDI bus connector of the “sending” board to the **left hand pin** on the connector of the “receiving” board. If you need to control more than two switchers you will need to use an external MIDI splitter.

Powering Multiple Splitters

When powering multiple splitters, you need to consider the total current required for all the loads that might be powered on at the same time.

In low power scenarios you might be able to “daisy chain” the power and ground connections for each splitter via the terminal blocks, and power them all via just one of the splitters (as shown above)

In higher power scenarios you might want to connect every splitter individually to the power supply via its terminal blocks, so that no single splitter needs to carry all the current.

Each splitter could also have its own individual power supply for the loads. If using a shared power supply, make sure that it has sufficient current capability to power all the loads.

Configuring Multiple Splitters

Configuring multiple splitters can be awkward when they are sharing a MIDI input, since they will all receive Sysex files

To help in this situation, there is a “MIDI Lockout” feature…if you press and hold the reset button for about 1 second the activity LED will start to flash regularly and the switcher will ignore all incoming MIDI information. By setting all but one switcher to this mode, you can program just that one switcher. Press the reset button again to return to normal.