Mini MIDI Controller Manual

Introduction

The Mini MIDI Controller is a device used for converting control voltages to definable MIDI messages and visa versa. The device has sixteen input/output ports: eight 10-bit 0 to +5 volt scalable control voltage input; and eight ports that can be configured as digital inputs or control voltage outputs. Additionally, the unit can be upgraded for wireless performance.

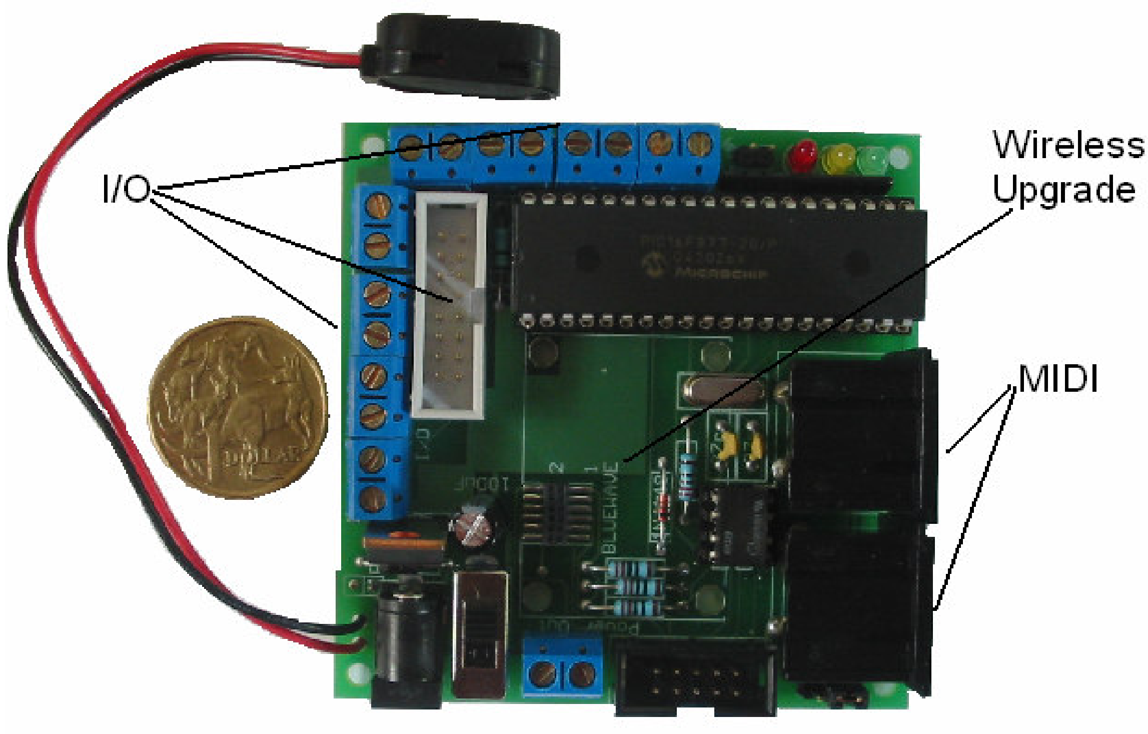


Figure 1 Mini Midi Controller

When a port is configured as an input, a change in the input voltage will cause the device to generate a MIDI messages to the MIDI output port. When a port is configured as an output, inputting a MIDI message—for example, from a MIDI keyboard, sequencer, or other MIDI device—and if the controller is configured to respond to that message, the controller will alter the output voltage depending upon the message. This makes it possible to control up to eight analogue or digital devices—such as relays or robotic motors—using a MIDI keyboard, MIDI sequencer, or any other MIDI generating device. Additionally, the device has a merge feature that will merge incoming MIDI messages with those generated by the controller on MIDI output port 1. The MIDI controller is powered by a 9V DC battery or 9V power supply.

Control Voltage to MIDI conversion.

The control voltage to MIDI conversion is separated into two sections: scalable voltage and digital switch, of which there is eight of each. Warning: Voltage inputs greater than 5V may cause damage to the unit.

Configuration

By default, the unit is configured for all I/O to be inputs. Any input that is not connected to a sensor will have a spurious voltage on it, known as a floating voltage. This may cause the device to generate more MIDI messages than can be handled by the computer receiving the MIDI. This can be overcome by configuring the device to muting the I/O by selecting the "M" check box for each channel to be muted, and clicking the "Write Config Button."

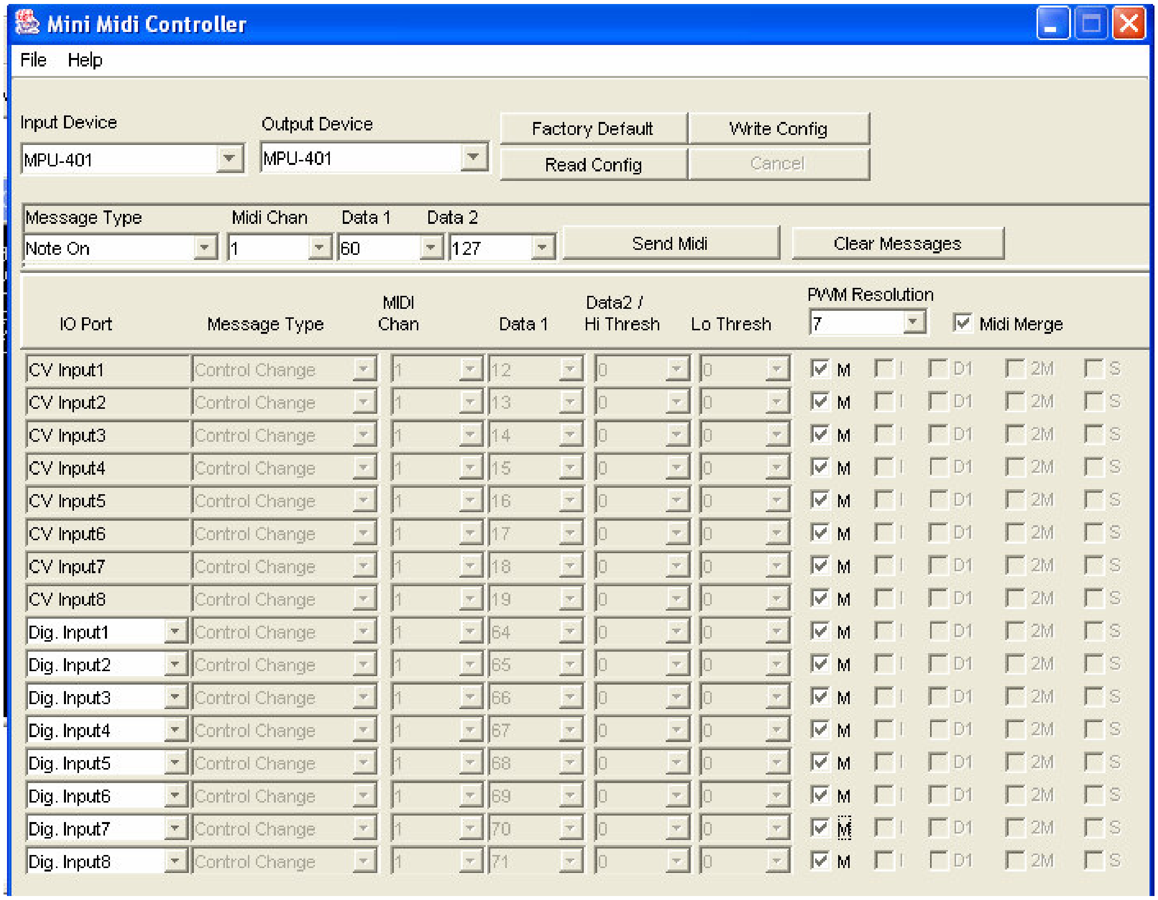


Figure 2 Configuring Device to Mute I/O

If too much MIDI is being transmitted, making it difficult to change the configuration, the device can be made to momentarily stop generating MIDI by shorting the two pins next to the red LED.

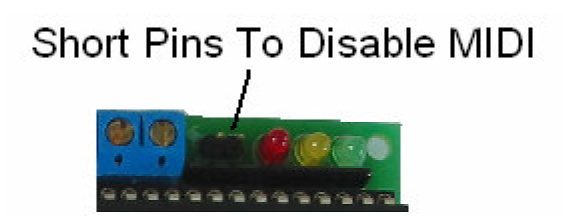


Figure 3 Momentarily Disabling MIDI

Scalable Voltage Input

Each scalable voltage input operates from a voltage ranging from 0 to +5V and generates MIDI messages as configured. For example, consider the following configuration.

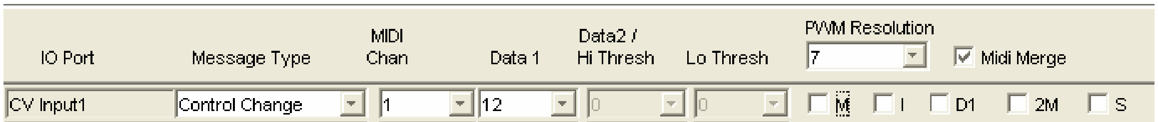


Figure 4 Control Voltage to MIDI controller

Control voltage input one is configured to generate MIDI controller 12 messages on MIDI channel 1. Increasing the voltage from zero to five volts on voltage input 1 would cause the controller to generate MIDI controller 12 messages on MIDI channel 1 with a controller value ranging from 0 to 127. The value of the data 2 byte in the message is proportional to the voltage at the input.

```
// Change voltage at input from 0 to +5V
Control Change 1 12 0
.....
Control Change 1 12 71
.....
Control Change 1 12 127
```

Alternatively, consider the following configuration

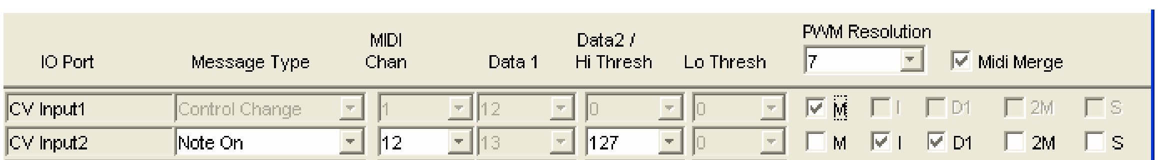


Figure 5 Inverting and Configuring Data Byte 1

Control voltage input 2 is configured to generate MIDI note on messages on MIDI channel 12 with a velocity value of 127. Additionally, the channel is configured to invert the value and to vary data byte 1. Increasing the voltage from 0 to +5V on control voltage input 2 would cause the controller to generate MIDI note on messages on MIDI channel 12 with a note number ranging from 127 to 0, while the velocity remains at 127. In this case, the value of data byte 1 is inversely proportional to the voltage at the input.

```
// Change voltage from 0 to + 5 VDC
Note On 12 127 127
....
Note On 12 65 127
....
Note On 12 0 127
```

The unit can be configured to use ten-bit resolution in three different ways: using pitch bend, using two MIDI messages, or scaling the input voltage

Pitch Bend

If the channel is set up to use a Pitch Bend message, the MIDI message will produce a 10-bit value. The seven most significant bits will be transmitted in the second data byte, while the three least significant bits will be transmitted in the first data byte. Consider the following configuration.

Figure 6 Analogue In configured for Pitch Bend

As the input voltage is moved from 0 to +5 VDC, the device will generate MIDI pitch bend messages, with the first data byte counting up from 0 to seven. When the value rolls over from 7 back to 1, data byte 2 reflects the most significant bits.

```
// Increase voltage from 0 to +5 VDC
Pitch Bend 1 0 0
Pitch Bend 1 1 0
Pitch Bend 1 2 0
Pitch Bend 1 3 0
Pitch Bend 1 4 0
Pitch Bend 1 5 0
Pitch Bend 1 6 0
Pitch Bend 1 7 0
Pitch Bend 1 0 1
Pitch Bend 1 1 1
Pitch Bend 1 2 1
.........
.........
Pitch Bend 1 5 126
Pitch Bend 1 6 126
Pitch Bend 1 7 126
Pitch Bend 1 0 127
Pitch Bend 1 1 127
Pitch Bend 1 2 127
Pitch Bend 1 3 127
Pitch Bend 1 4 127
Pitch Bend 1 5 127
Pitch Bend 1 6 127
Pitch Bend 1 7 127
```

Using Two MIDI Messages

The device can be configured to send two MIDI messages when the input voltage changes. The seven most significant bytes are transmitted in the first MIDI message in the second data byte, while the three least significant bits are transmitted in the second MIDI message. Consider the following configuration.

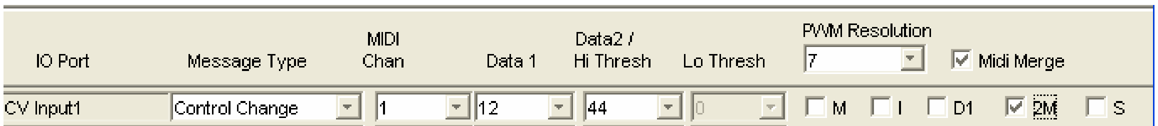


Figure 7 Sending CV input value over two MIDI messages

Changing the voltage at CV input 1 will cause a two MIDI controller messages to be generated on MIDI channel 1. The first message will be controller number 12 with the seven most significant bits, while the second message will be controller 44 with the three least significant bits.

```
// Move voltage from 0 to +5VDC
Control Change 1 12 0
Control Change 1 44 1
Control Change 1 12 0
Control Change 1 44 2
Control Change 1 12 0
Control Change 1 44 3
Control Change 1 12 0
Control Change 1 44 4
Control Change 1 12 0
Control Change 1 44 5
Control Change 1 12 0
Control Change 1 44 6
Control Change 1 12 0
Control Change 1 44 7
Control Change 1 12 1
Control Change 1 44 0
Control Change 1 12 1
Control Change 1 44 2
Control Change 1 12 1
Control Change 1 44 3
Control Change 1 12 1
Control Change 1 44 4
......
......
Control Change 1 12 127
Control Change 1 44 3
Control Change 1 12 127
Control Change 1 44 4
Control Change 1 12 127
Control Change 1 44 5
```

```
Control Change 1 12 127
Control Change 1 44 6
Control Change 1 12 127
Control Change 1 44 7
```

Scaling Analogue Input

The final method of using 10-bit resolution is through the use of the scaling mechanism. This can be particularly useful when the user is only interested in values that range between certain voltages. If a simple passive sensor—for example a force sensitive resistor—only produced voltages ranging from 1.17V to 3.51V, using a standard non-scaled value would produce MIDI messages with the CV representation ranging from 30 to 90. Additionally, the resolution of voltage changes would only be 7-bit, which means that a change would only be recognised for a change of 39mV. This means that there would be only 50 different possible values generated using that sensor. By using the scaling function, the device can be set to output values only between certain values, outputting a MIDI value ranging from 0 to 127. In this instance, the device can be set to monitor between 30 and 90. Given that the MIDI data byte only has seven bit resolution, The new resolution for this sensor would be approximately 18mV, which is twice what it was without scaling. The sensor would be configured as shown in Figure 8.

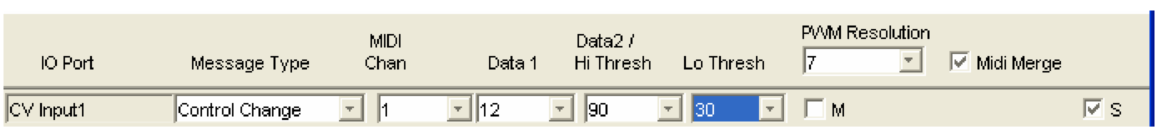


Figure 8 Scaling CV Input

Moving the sensor from minimum to maximum will produce a MIDI Controller message on channel 1 ranging from 0 to 127.

Digital Switch Input

Warning: Voltages greater than 5V on the Digital inputs may cause damage to the unit.

The digital switch input generates a MIDI message when the input is shorted to ground or the +5V rail. When using Digital inputs, a pull-up or pull-down resistor is required in your sensor to prevent floating voltages from generating MIDI. Consider the configuration below.

Figure 9 Digital Input Configuration

Digital switch input one is configured to generate MIDI controller 64 on MIDI channel 1. When the input is shorted to ground, the controller will generate MIDI controller 64 with a value of 0 on MIDI channel 1. The controller will generate the same message (with the exception that the controller value will be 127 instead of 0) when the input is set to +5V. The inputs can be configured to invert the values or determine which data byte contains the state of the input, similar to the scalable control voltage inputs.

The device can be configured to generate specific values data byte 2 values by selecting the scale checkbox. As already stated, in the default configuration, 0V produces a data byte 2 value of 0, while +5V produces 127. These values can be configured to generate custom values for data byte 2. Consider the configuration shown below.

Figure 10 Customising Digital In Data Values

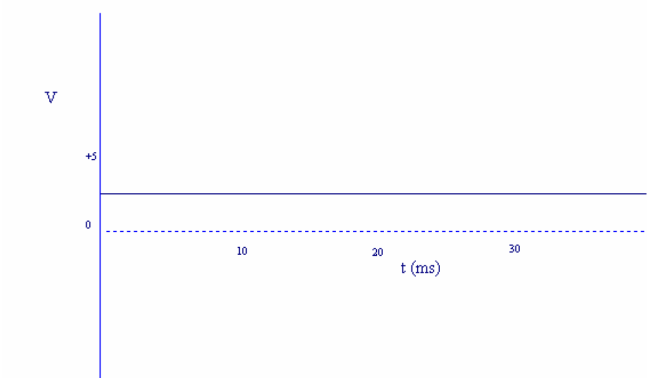
When the input becomes 0 V, the device will generate controller 64 on MIDI channel 1 with a value of 49. When the voltage becomes +5V, the controller value will be 6. This feature can be useful for configuring digital inputs to perform program changes on the change of a digital in value. In the following configuration, a momentary change on digital input one will cause program change 50 to occur on MIDI channel 1.

Control Voltage Outputs

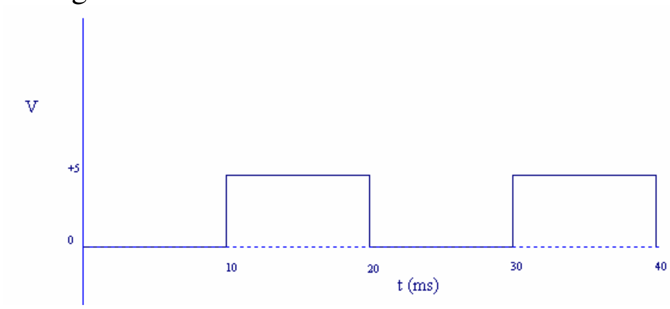
The device can generate control voltage through the use of pulse width modulation.

Pulse Width Modulation

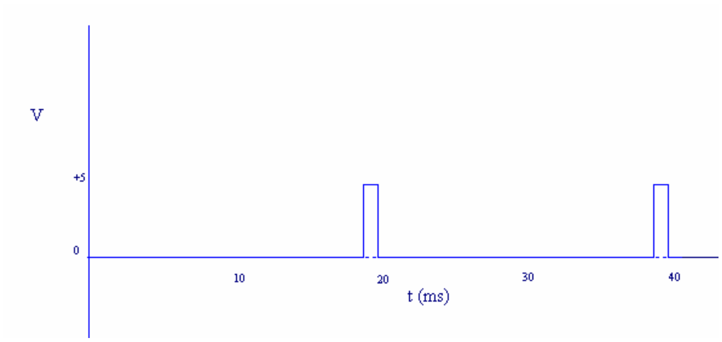
Pulse width modulation, known as PWM, is different to DC voltage in that DC voltage is a steady electrical force at any instance of time. The graph below shows 2.5V DC as a steady voltage over time.



PWM, on the other hand, is the average DC potential over a fixed period of time. For example, if during a 20ms space of time, a +5V pulse existed for 10ms, the average voltage would be +2.5V. If we repeat this every 20ms, the output will transmit on average 2.5V.



If we reduced the pulse width to 1ms, this would mean that a 1 ms pulse would appear every 20ms, which means that the average voltage would be 1/20th of 5V, which is 0.25V.

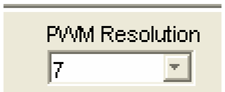


The amount of time that the signal is positive with respect to the repeat time is called the duty cycle. This is the whole concept of PWM.
The apparent smoothness of the voltage is dependant upon how often the pulse is refreshed. For example, a pulse width of 1ms every 2ms, will appear significantly smoother than a 10s pulse every 20s. The number of times the pulses are repeated is known as the frequency—higher frequency means smoother output.

Another factor is resolution, which means how many different pulse widths can be set.

For example, an 8 bit resolution means that you can 256 different PWM values.

This can configured on the device.



The resolution is inversely proportional to the frequency; i.e. a high resolution normally means a lower frequency. The frequency available for the following resolutions is listed below.

Bits	Duration ms	Frequency
8	15	67
7	7	143
6	10	100
5	2.5	400
4	2	500
3	1	1000
2	0.8	1250
1	DC	0

Figure 11 PWM resolution versus frequency

The Mini Midi controllers use PWM to generate CV output as opposed to the Dumb Controller and Smart Controllers, which use a steady DC voltage. Although PWM is better for controlling motors, steady DC voltage is better for controlling analogue synthesisers in that PWM introduces audible noise

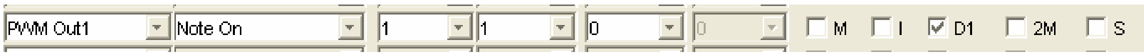
PWM Out

The control voltage output generates a PWM voltage of 0 and +5 V, with the duty cycle depending upon the incoming MIDI message and the configuration. Consider the following configuration: scalable output 1 is configured to generate a voltage when the controller receives a MIDI control 1 message on MIDI channel 1 through its MIDI input.

Figure 12 PWM Outputs

If a series of MIDI controller 1 messages on MIDI channel 1 with controller values ranging from 0 to 127 are received, the MIDI controller will generate a PWM output with the duty cycle ranging from 0 to 100%. The duty cycle is directly proportional to the MIDI data byte 2 value.

Alternatively, consider the following configuration.



PWM output 1 is configured to respond to any note on messages received on MIDI channel 1. Connecting a MIDI keyboard, generating messages on MIDI channel 1, will cause the controller to generate a PWM voltage which is proportional to the note number pressed on the key. The output can be configured to also generate a voltage that is indirectly proportional to the MIDI message data byte.

MIDI Merge

The controller will merge incoming MIDI messages with those generated by the controller. Incoming MIDI messages take a higher priority than those generated by the control voltage inputs. If an extremely intense stream of MIDI messages are input to the controller and the merge is enabled, the controller may respond sluggishly to the control voltages. This can be overcome by disabling the merge feature, using a dedicated MIDI merge box, or reducing the amount of input MIDI data.

Connectors

The connectors on the Mini Midi Controller are as follows:
Sixteen screw terminals for I/O, 2 screw terminals supplying 0 and +5 V to sensors, MIDI in, MIDI out, and Power. The screw terminals for I/O are mapped to a 16 Pin IDE connector, which enables the performer to use a single plug connection during a performance setup to connect all the sensor inputs / outputs to the MIDI controller instead of screwing sixteen individual terminals.

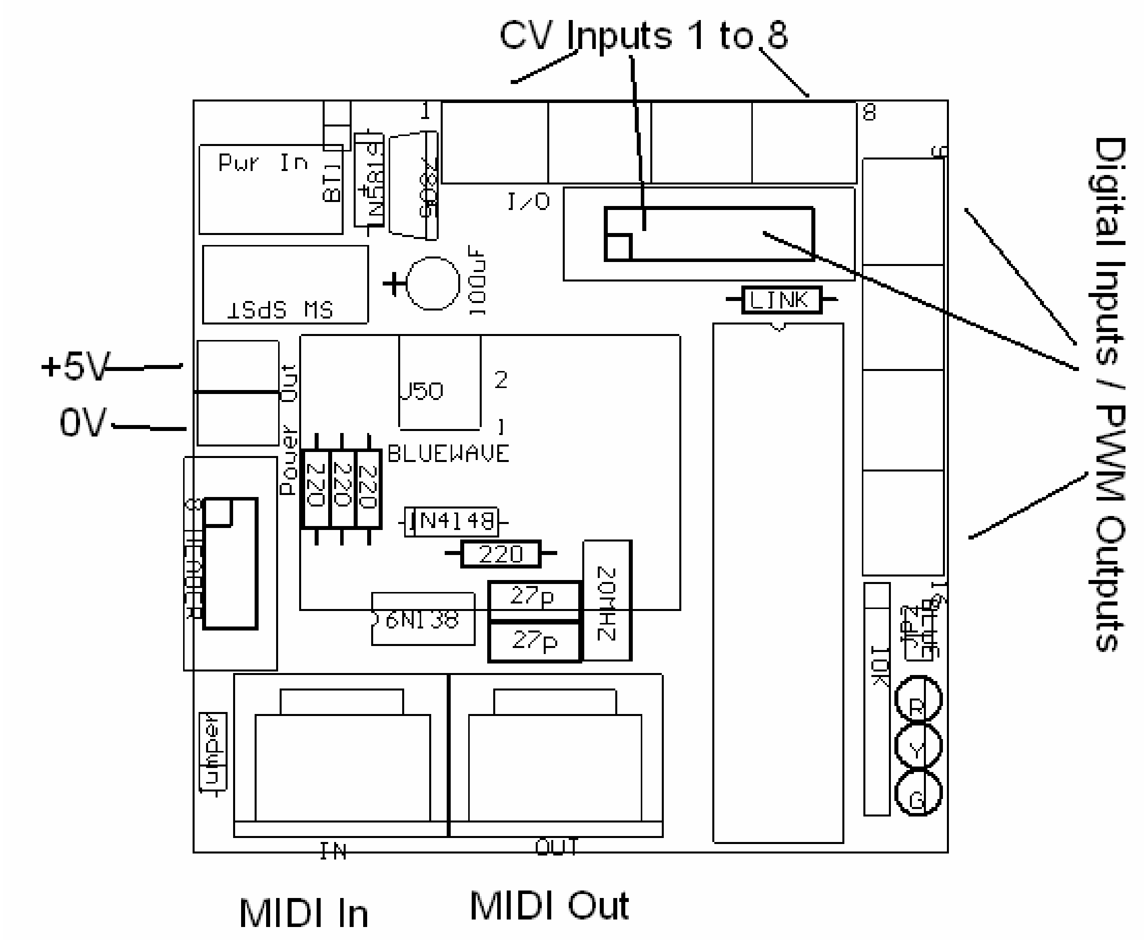


Figure 13 Connectors

Control Voltage Input

A simple sensor can be created by connecting the outer pins of a potentiometer to the sensor supply and the wiper pin to the sensor input.

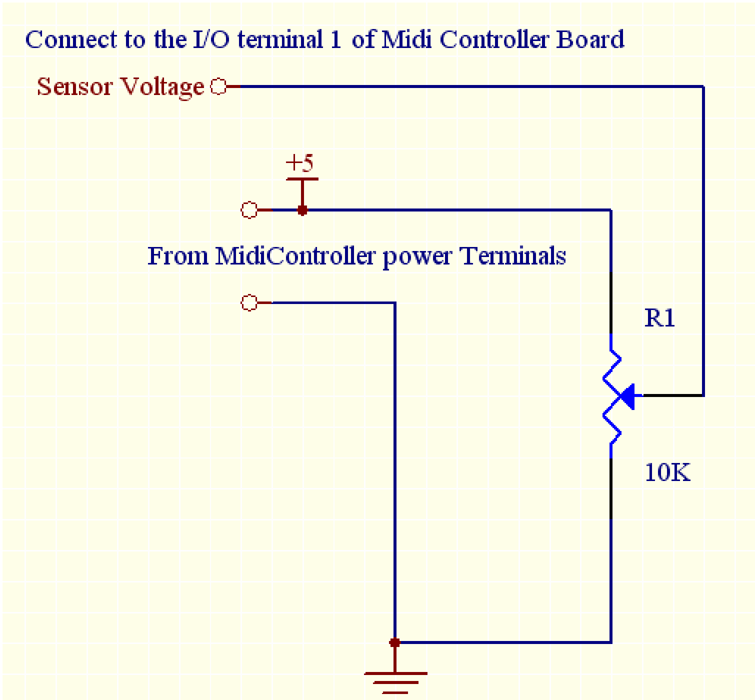


Figure 14 Simple Potentiometer Sensor

Control Voltage Output

The control voltage outputs are used to provide WM control voltages

The outputs can supply approximately 2mA per channel, and as such, must be used with a driving transistor. When driving inductive loads, such as motors, solenoids, and relays, a commutating diode must be used to prevent inductive spikes from damaging the unit. Controllers damaged from inductive spikes are considered to be abused and are not eligible for warranty repair.

The uses of a commutating diode with a motor is shown in Figure 15. Note that the diode is configured in reverse bias.

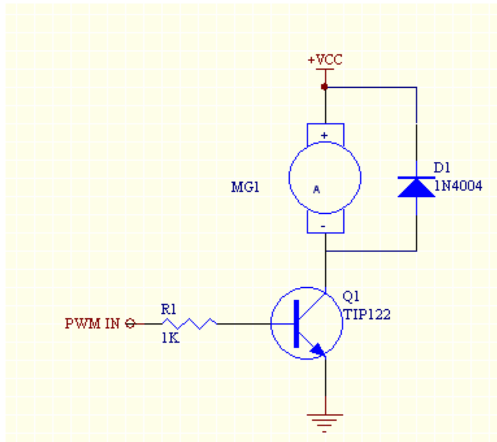


Figure 15 PWM driving a motor

Eight-bit resolution is possible for the outputs using two MIDI messages or pitch bend.

Configuration

The MIDI controller comes with the following factory default configuration:

- MIDI merge enabled
- Scalable CV inputs 1 to 8 – Controller numbers 12 to 19 on MIDI channel 1. Controller value 0 to 127 proportional to input control voltage.
- Digital switch inputs 1 to 8 – Controller numbers 64 to 71 on MIDI channel 1. Controller value equals 0 and 127 for switch open and closed respectively.
- PWM CV outputs 1 to 8 – the controller value for controller numbers 1, 2, 4, and 7 on MIDI channels 1 and 2 set the output voltage. The following table maps MIDI controller number and channel to CV output:

MIDI channel	MIDI controller number	CV output number
1	1	1
1	2	2
1	4	3
1	7	4
2	1	5
2	2	6
2	4	7
2	7	8

CV output duty cycle ranges from 0 100% with controller value 0 to 127 respectively.

The following configuration options (default off) are available for each individual channel:

- Mute – cause the input or output to have no effect.
- Invert – reverses the effect of the controlling data byte.
- Data 1 varies:

(a) Control Voltage input- causes a control voltage input to generate the data 1 value of the MIDI message proportional (or inversely proportional depending upon Invert value), with the data 2 value (f applicable to message type) at the configured value. See section on Scalable Voltage Input for an example.

(b) Control voltage output – causes the MIDI controller to generate a control voltage whose value is dependent upon the value of the data 1 byte of the MIDI message. See section on Scalable Voltage Output for an example.

- Use two messages—causes CV input and output to be transmitted over two separate MIDI messages.
- Scale Input—Scales the CV input to produce MIDI messages only between certain ranges. Available only for inputs.