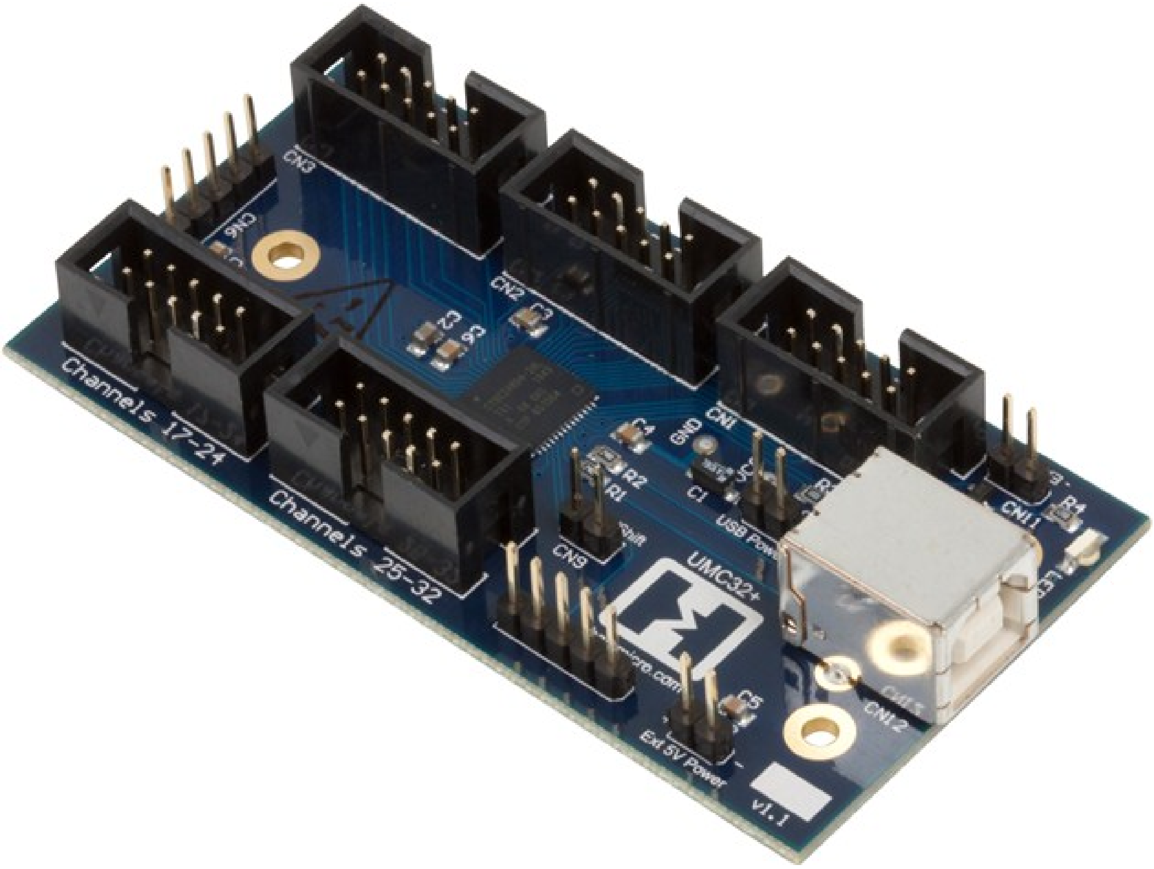


UMC32+ DIY USB-MIDI Controller

USER MANUAL

Firmware v3.1.0 / Hardware v1.1

Product Description

The UMC32+ is an OEM/DIY product that allows Electronic Musicians, Multimedia Artists and Experimenters the ability to create custom user interfaces to control any software application that supports the MIDI protocol. The UMC32+ provides the essential core (microcontroller, power supply, USB functionality) to allow the user to easily implement their choice of control elements (example: switches, pots and or faders). The UMC32+ will also translate data from your MIDI host application into logic levels for driving LEDs, Relays and more.

- Easy to implement with little or no electronics experience.
- 32 individually configurable inputs or outputs that send or receive user defined MIDI data with any MIDI host application.
- 64 I/O's can be achieved by linking two UMC32+'s together using the UMC-Linker board (sold separately).
- Multiple units can co-exist on the USB bus.
- Small printed circuit board (1.7" x 3.3") allows creating portable end devices
- True plug-and-play (USB-MIDI class compliant drivers are provided by the operating system)
- Compatible with all versions of Mac OS X, Windows 7 / Vista / XP / 2000
- USB bus powered. A single USB cable handles power and data.
- No programming is necessary (No code to write).
- Upgradeable firmware via USB using Windows XP bootloader utility (32 bit only).
- Using extended MIDI commands, complex MIDI mappings can be realized including:
  - Individual selection of each I/O's corresponding MIDI channel (1 to 16)
  - Individual selection of each I/O's corresponding MIDI message value
  - Individual selection of each I/O's corresponding MIDI message type(s):
    - Note On
    - Poly Pressure
    - Controller

- Individual selection of each I/O's corresponding hardware type including:
  - Digital Input with pull-up (active low) - For use with SPST tactile switches
  - Digital Input (active low) - For receiving messages from sensors with active low TTL outputs
  - Digital Input (active hi) - For receiving messages from sensors with active high TTL outputs
  - Analog Input - For connection to potentiometers, joysticks, faders etc.
  - Analog Input (Inverted) - Same as above, except that MIDI data is inverted.
  - Digital Output (Active High) – For driving LED's, Relays, or high power FETs etc.
  - LED Output with Blink –For blinking of status LED's (see manual for details).

- The new "plus" version incorporates the following enhancements over original UMC32.
  - A redesigned PCB with dedicated GND and VCC planes for enhanced signal integrity.
  - I/O Configuration is now handled entirely by the configuration utility.
  - Connector for external status LED, on board status LED is now blue.
  - USB connector is now the standard TYPE B and is much more secure.
  - Mounting holes (#4 screw) close to USB connector make for a near bulletproof installation.
  - AUX I/O block for future expansion (TBD)
  - Improved Firmware Boot loader
  - Linkable with the original UMC32
  - RoHS Compliant (lead Free), manufactured to IPC-A-610 Rev. B, Class II.

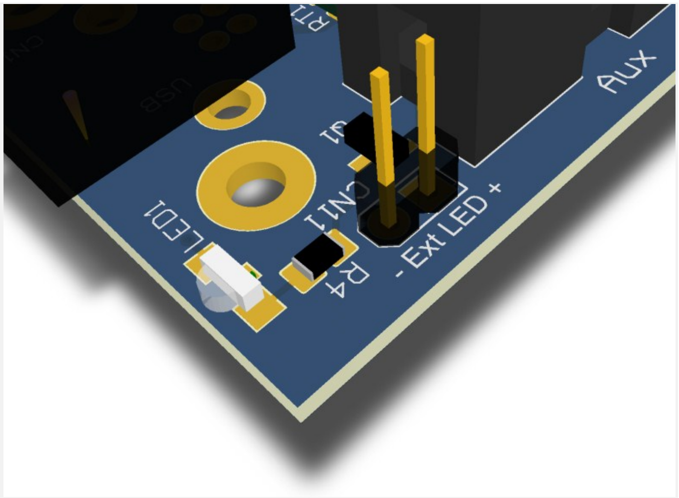
* Please note that that the new UMC32+ firmware is not compatible with the original UMC32.

Before proceeding it is strongly recommended that you read and understand the precautions section of this manual. Please pay attention to the section discussing the safe handling and how to avoid electrostatic discharge (ESD). Failure to observe the recommendations may result in damage to your UMC32+ and void your warranty.

Status Indicator LED (CN11)

The indicator LED (LED1) can be found next to the USB connector. This LED provides information about the state of the UMC. When connected, your computer will recognize the UMC and load the class compliant USB-MIDI class device driver that is provided by your operating system. The LED will stay on until communication with your host application begins. At this point the LED will flicker when MIDI traffic between the host application and the UMC occurs.

The UMC32+ also provides a 2 pin header + (Anode) and – (Cathode) for connecting to an external LED. It is important to remember that in an addition to the LED itself, you will need to provide a current limiting resistor in series with your LED (Less than 20mA is recommended at 5V dc).

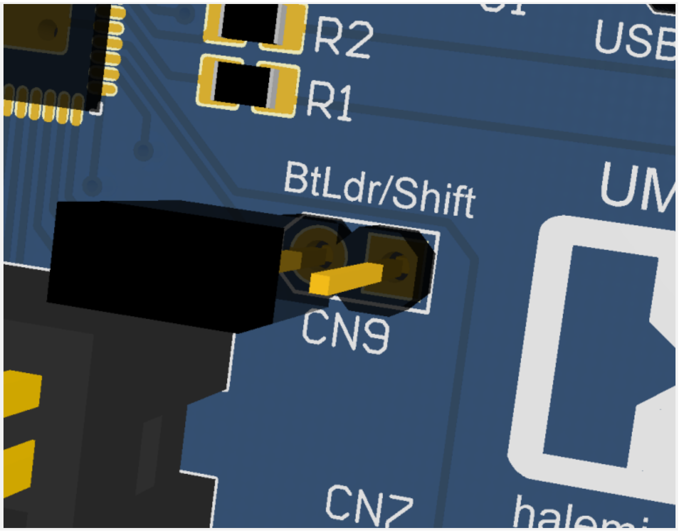


Boot Loader / Shift Pin (CN9)

When the UMC is plugged into the USB port it first checks the state of the shift pin, if it is pulled low (your switch is being held) then the UMC will enter bootloader mode and prepare to accept a firmware update. Information regarding firmware updates will be made available on our website at http://www.halemicro.com

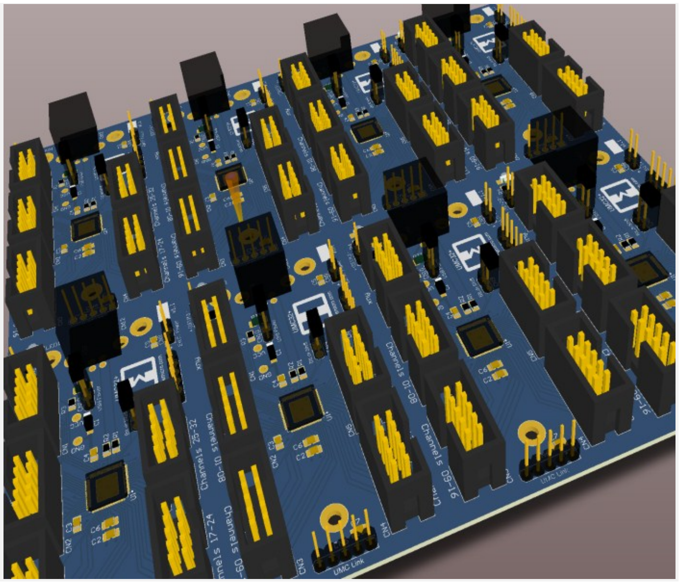
In response to customer requests there have been many revisions to the UMC firmware. It is important that you ensure that you are reviewing the correct manual that corresponds with the firmware version of your device.

During normal MIDI operation this pin can double the number of software parameters you can control. See Hardware I/O Types For more information



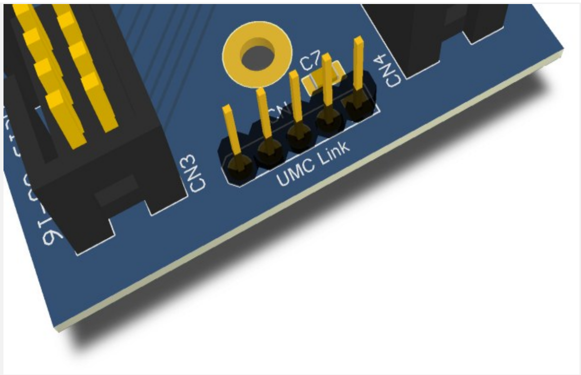
Multiple UMC's using a USB Hub

For connecting more than 1 UMC32+ to the same computer it is recommended that you install one of 8 available firmware binaries on your UMC32+. First download the latest firmware set from the Hale website. The zip file will contain multiple firmware revisions with ".DLD" extensions. Next download and install the UMC32+'s firmware update utility, follow the directions provided by the utility and select the appropriate firmware. Each firmware binary is slightly different in that they contain a unique USB product ID as well as a unique driver name that allows the operating system to identify it on the USB bus. As a rule of thumb it is recommended that when you are configuring your device using the Configuration Utility you dedicate one MIDI channel to each device (2 MIDI channels if you are using the shift feature). As MIDI host applications and operating system do not always load and assign UMCs in the same order this method has proven to be a consistent way of identifying each UMC that is connected to your system.



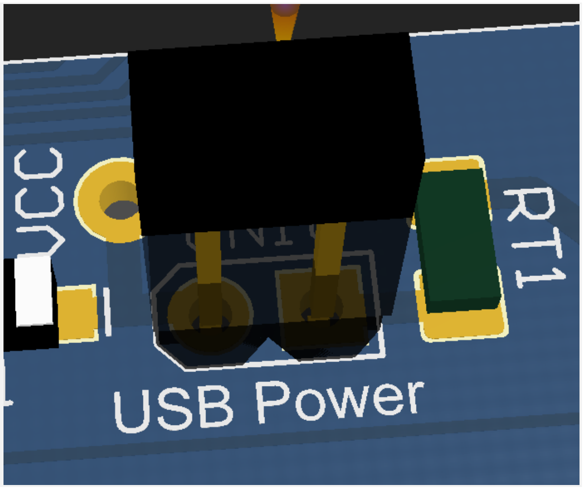
Linking of two UMC's for 64 IO's (CN6)

You have the ability to link up to 2 (two) UMC32+s together using the 5 pin aux port located at the far end of the board between Bank2 (09-16) and Bank3 (17-24). The linking is achieved by connecting the corresponding pins with jumpers or the "UMC linker" board. Master / Slave operation is auto-detected making the link hassle free. Please keep in mind that slaved units are not accessible to the Configuration utility. Therefore each UMC must be configured prior to being linked. The total current for both devices must not exceed 500mA when utilizing USB bus power. When using a UMC as a slave device the slave must have the CN10 Jumper removed or the slave will not operate.



Setup for USB Power(CN10)

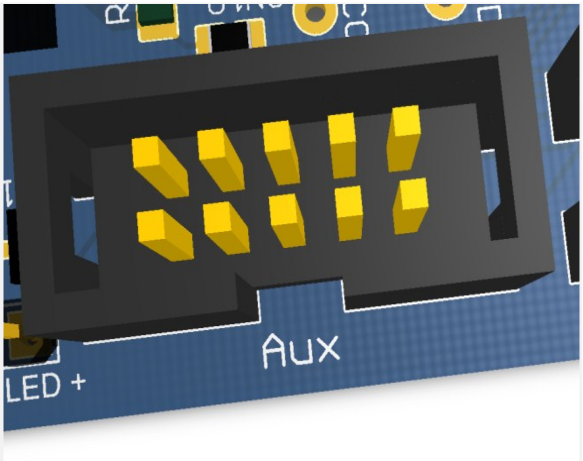
The factory default has CN10 fitted with a jumper to provide power via the USB bus. This also provides a convenients means for measuring the current consumption of your final build. The only other cases where this jumper should be removed or off is when a UMC is connected to a linker board and is being used as a slave, or when external power is being used as a Master.



Aux Connector (An extra 8 inputs!)

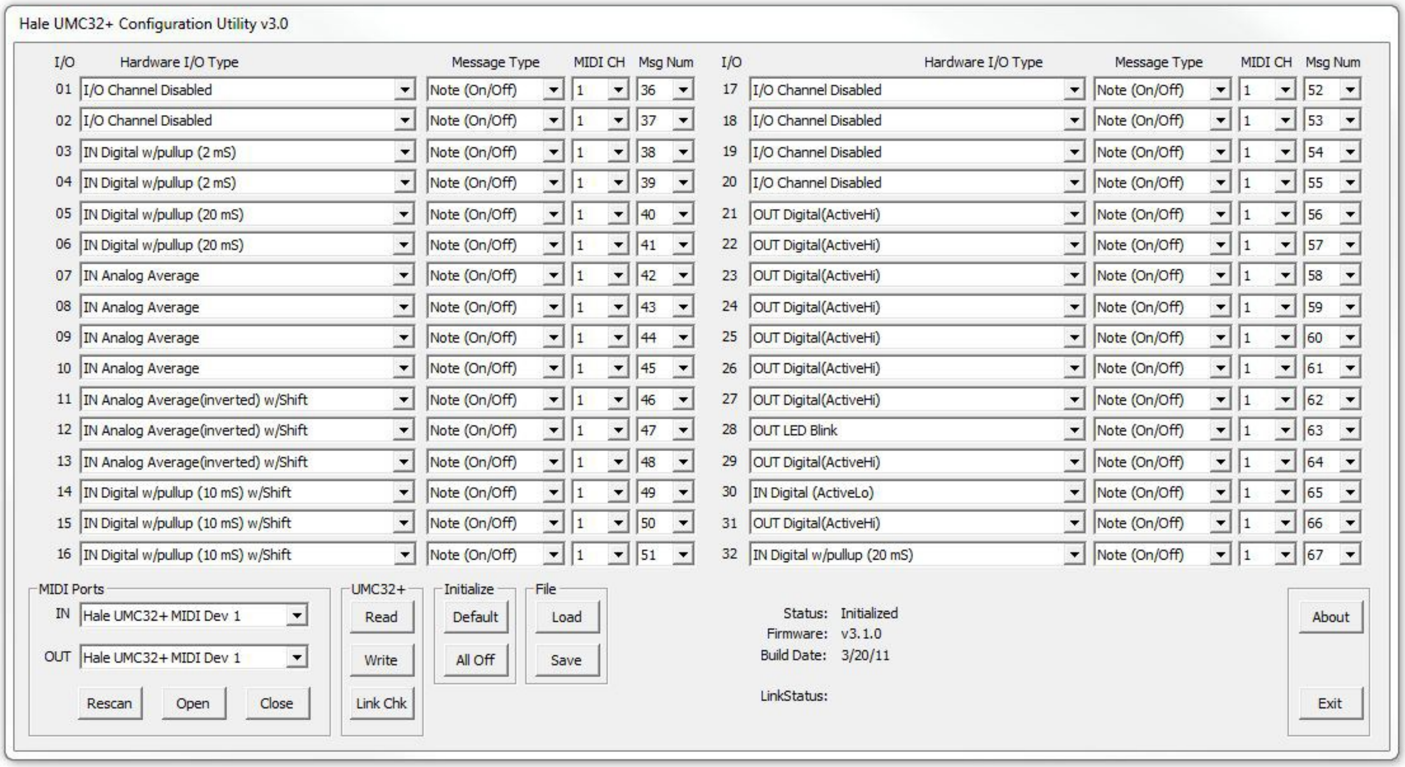
This connector will provide a means for future expansion. It is configured as a digital input with a 10mS debounce. The input Type is NOT configurable and is fixed as Note On/Off Channel 1, Note # 1 thru 8 if Master, and Notes # 9 thru 16 if a linked slave. The shift pin also works here.

If a jumper is placed across Pin1 and Pin3 of the Aux Connector ("A" and Ground - see Pin Diagram on Page 11), the UMC will configure it's 32 I/O's as Digital Inputs. This test mode is useful tor trouble shooting and diagnosis. It is also important to note that this test mode will not overwrite any of your configuration settings as only the configuration utility will do that.



I/O setup using the configuration utility

To begin, download and install the UMC32+ configuration utility. It is recommended that your video display should be set at a minimum screen resolution of 1024x768 or higher). The utility is available for download here.



Hale UMC32+ Configuration Utility v3.0

I/O	Hardware I/O Type	Message Type	MIDI CH	Msg Num	I/O	Hardware I/O Type	Message Type	MIDI CH	Msg Num
01	I/O Channel Disabled	Note (On/Off)	1	36	17	I/O Channel Disabled	Note (On/Off)	1	52
02	I/O Channel Disabled	Note (On/Off)	1	37	18	I/O Channel Disabled	Note (On/Off)	1	53
03	IN Digital w/pullup (2 mS)	Note (On/Off)	1	38	19	I/O Channel Disabled	Note (On/Off)	1	54
04	IN Digital w/pullup (2 mS)	Note (On/Off)	1	39	20	I/O Channel Disabled	Note (On/Off)	1	55
05	IN Digital w/pullup (20 mS)	Note (On/Off)	1	40	21	OUT Digital(ActiveHi)	Note (On/Off)	1	56
06	IN Digital w/pullup (20 mS)	Note (On/Off)	1	41	22	OUT Digital(ActiveHi)	Note (On/Off)	1	57
07	IN Analog Average	Note (On/Off)	1	42	23	OUT Digital(ActiveHi)	Note (On/Off)	1	58
08	IN Analog Average	Note (On/Off)	1	43	24	OUT Digital(ActiveHi)	Note (On/Off)	1	59
09	IN Analog Average	Note (On/Off)	1	44	25	OUT Digital(ActiveHi)	Note (On/Off)	1	60
10	IN Analog Average	Note (On/Off)	1	45	26	OUT Digital(ActiveHi)	Note (On/Off)	1	61
11	IN Analog Average(inverted) w/Shift	Note (On/Off)	1	46	27	OUT Digital(ActiveHi)	Note (On/Off)	1	62
12	IN Analog Average(inverted) w/Shift	Note (On/Off)	1	47	28	OUT LED Blink	Note (On/Off)	1	63
13	IN Analog Average(inverted) w/Shift	Note (On/Off)	1	48	29	OUT Digital(ActiveHi)	Note (On/Off)	1	64
14	IN Digital w/pullup (10 mS) w/Shift	Note (On/Off)	1	49	30	IN Digital (ActiveLo)	Note (On/Off)	1	65
15	IN Digital w/pullup (10 mS) w/Shift	Note (On/Off)	1	50	31	OUT Digital(ActiveHi)	Note (On/Off)	1	66
16	IN Digital w/pullup (10 mS) w/Shift	Note (On/Off)	1	51	32	IN Digital w/pullup (20 mS)	Note (On/Off)	1	67

MIDI Ports
IN Hale UMC32+ MIDI Dev 1
OUT Hale UMC32+ MIDI Dev 1

Status: Initialized
Firmware: v3.1.0
Build Date: 3/20/11

LinkStatus:

How to configure (quick start):

1) Disconnect any wires, ribbon or usb cables from the UMC. Close any MIDI host applications that may be running as the configuration utility needs exclusive access to the UMC's MIDI I/O.

2) Reconnect the UMC32+ to your PC with your USB cable. If this is the first time you have connected this device to your computer please wait until Windows displays a message stating that your hardware/devices are ready to use.

3) After windows has installed and loaded the drivers for the device, launch the UMC32_Plus_ConfigUtil_v1xx application.

4) In the "MIDI Ports" section, select the IN Device, and the corresponding OUT device. If you are using Windows XP the UMC32+ will show up as "USB Audio Device", under Vista and Windows 7 it will show up as "UMC32+_D0x", where x is the device number. .. The device number you see will depend on the firmware that you have bootloaded into the UMC. All devices are shipped from the factory as D01.

5) Click "Open" to open MIDI input and MIDI output ports.

6) Click "Read". This will cause the application to grab all of the current configuration data for the selected device into the application. The 'Status' window should display the following:

"Status: UMC32+ read OK"
"Firmware: v3.x.x"
"Build Date: MM//DD//YY"

7) For each I/O channel select the Hardware I/O type, MIDI Channel, Message Type and message number.

8) When you are finished, click on "Write" to update the UMC with the changes you have made.

9) Exit/Quit the application.

10) Disconnect the USB cable from the UMC, then reconnect it to resume normal MIDI operations.

Please exercise care in selecting and connecting any sensors or controls to your UMC. It is recommended that you triple check your wiring to and from the UMC. This is especially the case when using Output Mode(s). It is quite easy to set a channel for output mode and mistake it as an input when connecting your pots or switches causing a short circuit. If you are in doubt about the hardware type of your control please post your compatibility question in the UMC32+ user forums located at http://forums.halemicro.com/.

Configuration utility controls and settings

Hardware I/O Types:

I/O Disabled – Channel I/O is disabled and will not be processed. All unused channels should use this setting.

IN Digital w/pullup (2ms) – Channel is set for Digital (TTL) input with the microcontroller's internal pullup enabled. This type is the same digital input mode when using the onboard DIP pack. This type of input is typically used to connect to a single pole single throw switch. One end of the switch connects to ground the other will connect to this I/O channel. This type is also useful for sensors that pull the line low (active low) when the sensor is actuated and release the line or place it in Hi-Z state when the sensor is not actuated. The variations of this input type 2ms, 5ms, 10ms and 20ms denote the debounce delay utilized when the switch is read. Higher values are useful for switches that generate unwanted noise when actuated.

IN Digital (Active Low) - Channel is set for Digital (TTL) input. This type is used when accepting a high or low signal from the output of a powered sensor on/off type output. This is for sensors that pull the line low when the sensor is actuated and high when it is not actuated.

IN Digital (Active High) - Channel is set for Digital (TTL) input. This type is used when accepting a high or low signal from the output of a powered sensor with a on/off type output. This is for sensors that pull the line high when the sensor is actuated and low when it is not actuated.

IN Analog Average - Channel is set for Analog (0v to 5volts) inputs. This is most commonly used for connection to potentiometers, faders and joysticks. It can also accept the input from other sensors that output a voltage in the range of 0 to 5 volts. In this mode, an input of 5 volts will generate a message value of 127 (0x7F hex), whereas an input of 0 volts (ground) will generate a message value of 0 (0x00 hex).

IN Analog Average w/Shift – Same as previous Type

IN Analog Average (Inverted) – Same as above however in this mode, an input of 5 volts will generate a message value of 0 (0x00 hex), whereas an input of 0 volts (ground) will generate a message value of 127 (0x7F hex). This mode is especially useful if you happen to solder the +5V and Ground terminals on a potentiometer backwards. Instead of reopening your device to correct the problem by resolder again, simply select this mode instead.

IN Analog Average (Inverted) – Same as previous Type

IN Analog Direct – This input type reports the raw value of the ADC and does not perform any averaging on the sampled data. This mode is most useful for raw performance and is best used with so force sensing resistors or soft pots. You should expect to see erroneous data from this type of input since filtering is disabled.

IN Analog Direct (Inverted) – Same as above however a value of 0 becomes 127, and 127 becomes 0. See above for more information.

OUT Digital (Active High) – If a corresponding MIDI message is received by the UMC with a value of 0 then the output for this channel will be low or 0 volts (ground). If the corresponding MIDI message is 127 (0x7F in hexadecimal) then the output will be high or +5volts.

OUT LED Blink –This mode functions like the OUT Digital above however, values between 64 and 127 shall turn the LED on values between 1 and 63 shall blink the LED. Zero value turns the LED off.

IN Analog Average (MIDI OFF) – This input type is used for internal testing and should not be used. The selected channel is sampled however no data is transmitted.

MIDI channel shifting using the BtLdr/Shift pin

The configuration utility also provides variants of the above inputs types. These are post fixed as "with Shift". When a button connected to the shift pin is pressed data is transmitted on the base of the MIDI Channel plus 8. This effectively multiplies the number of software parameters that you can manipulate with your controller. For example a button that ordinarily sends a "Note on, Note #24, on MIDI Channel 1", when Shifted instead sends "Note on, Note #24, MIDI Channel 9." For types that are assigned with a base channel of 9 thru 16, those are shifted down. For example a button that ordinarily sends a "Note on, Note #30, on MIDI Channel 10", when Shifted instead sends "Note on, Note #30, MIDI Channel 2."

MIDI CH

Here you can select the desired MIDI Input or output channel that corresponds to the UMC32+'s physical hardware I/O channel. Here You can select 1 of 16 channels. When using multiple UMC32s on different ports of a USB hub it is recommended that you assign a MIDI channel to a UMC and stick with it. As it is impossible to guarantee the order which multiple UMC32 will enumerate with Windows a full proof way to consistently identify an assigned I/O is by its MIDI channel.

Message Type

Note On (0x90) – This is the recommended message type for input or output controls that are Boolean or ON/OFF. Such controls are switches, or other sensors that output a logic level high or low. When this mode is selected for a channel there are certain conditions that will cause a note off (0x80) message to be generated. Instead of generating a Note On with a velocity of zero the UMC will generate a true note off message. This applies to IN Digital w/Pullup (active low), IN Digital (Active Low), IN Digital (Active High) hardware Input types. For the hardware Output type(s), both note on velocity of 0 as well as Note off velocity zero are interpreted correctly by the UMC.

Poly Pressure (0xA0) & Controller (0xB0): These message types are recommended for I/O's that are configured as Analog inputs.

Msg Num (Message Number)

Depending on the Message type the message number may be interpreted differently. The allowable range for this parameter is 0 to 127 (0x00 to 0x7F in hexadecimal). With regards to note on and poly pressure message types these values are indicative of note numbers or keys on a piano keyboard. For instance C-1 is 0, C#4 is 61 and
G9 is 127.

Default:

This will initialize all controls to the following: IN Digital w/pull-up (active low), MIDI Channel X (where X = the device #), Note On, message number 36 thru 67.

All Off:

This will initialize all controls to the following: I/O Disabled, MIDI Channel X (Where X = the device #), Note On, message number 36 thru 67.

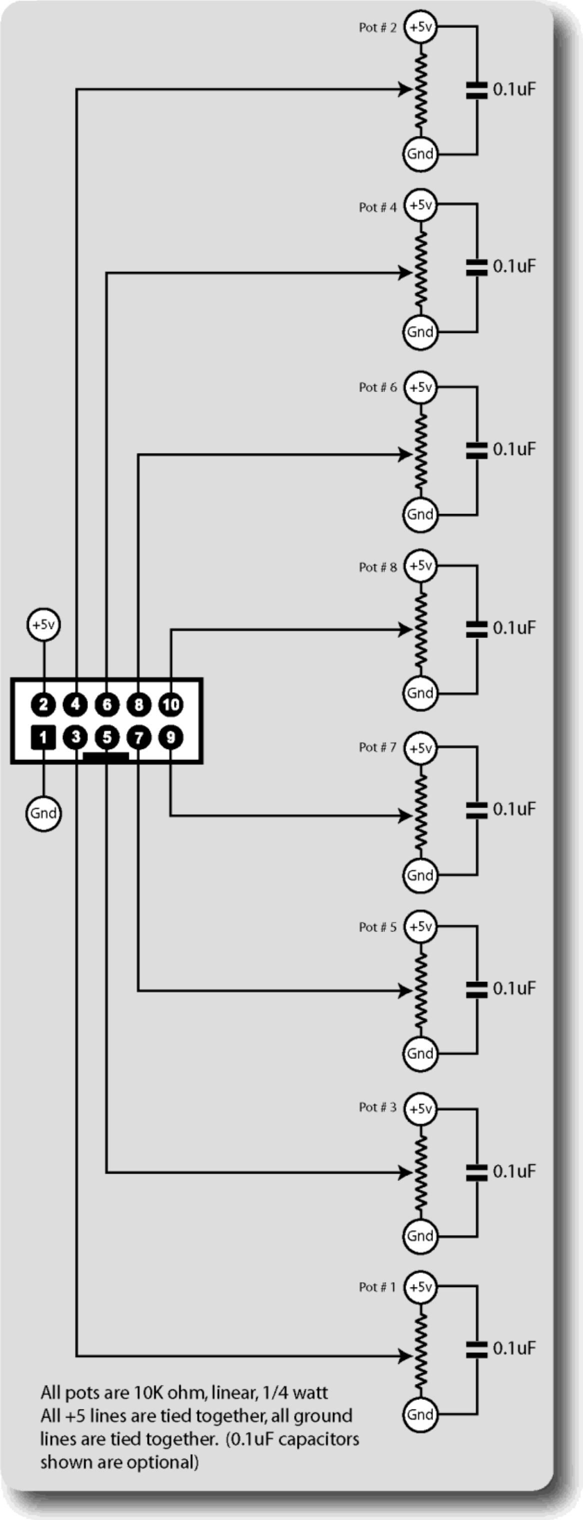
General Precautions

The UMC is designed to protect against short circuits, over current conditions, transient voltages and electrostatic discharge. However, we ask that you observe some simple guidelines when working with the UMC. Failure to do so may cause damage to your equipment and/or void your product warranty.

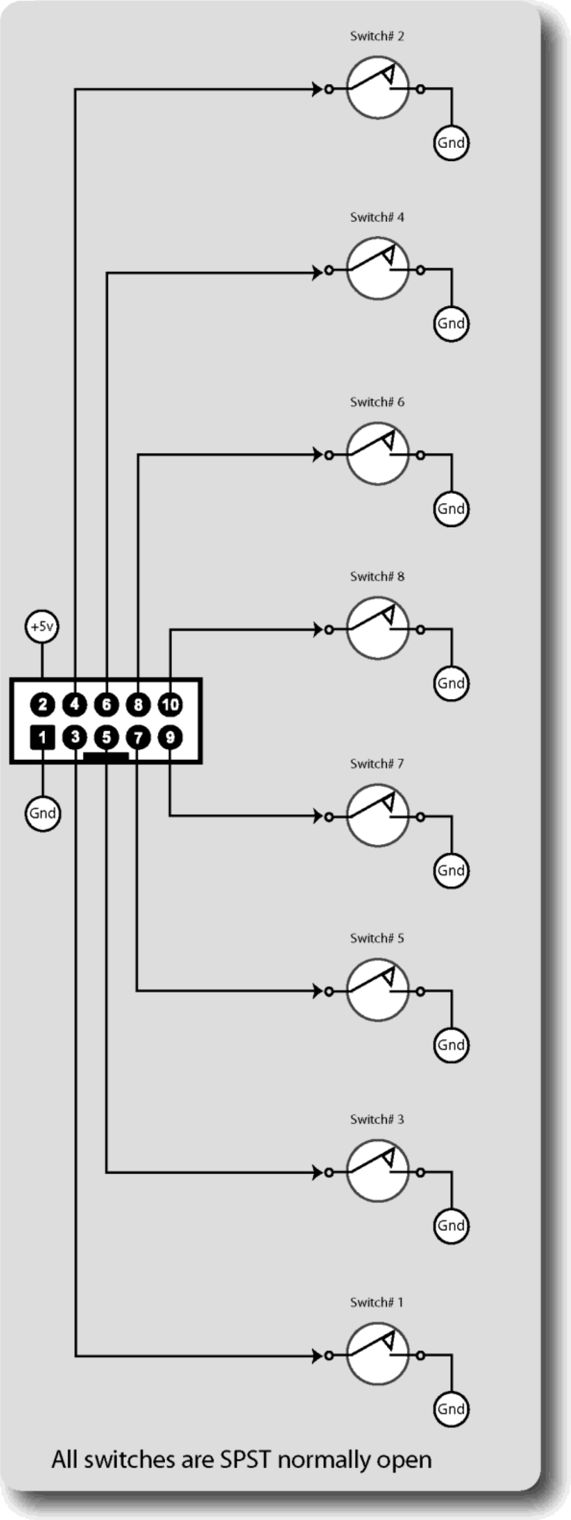
- Treat the UMC32 as you would any modern computer RAM and observe best practices for Electrostatic Discharge see: http://www.phys.cwru.edu/courses/p203/resources/ESDprocedure.pdf or http://www.esda.org/esd_fundamentals.html
- Do not place a jumper or otherwise short circuit the ExtLED pins or the Ext 5V Power pins.
- Do not solder any wires or attempt to make any modifications to the UMC directly.
- Do not connect the "+V" (+5 Volts DC) terminal to G (Ground). When this occurs, it is commonly referred to as a "short circuit".
- Always unplug the USB cable from the device before modifying connections on the UMC.
- As with all electronic devices, observe proper ESD handling practices. Always handle the printed circuit board (PCB) by the edges, never touch the top or bottom. Failure to make sure you are ground
- Never allow the UMC to touch a wet or conductive surface.
- Do not use any external voltage source to power devices that are connected to the UMC.
- The maximum recommended current per channel is 8mA..
- You may use the +5V and Ground pins off each bank to power other devices such as LED's however the maximum recommended is 80mA.
- As an additional precaution, we **strongly** recommend using a powered USB hub when experimenting with the device. Once you have finalized your project, and are sure you do not have any short circuits then you may connect the UMC directly to your computer's USB input. This is especially important when interfacing with a laptop, as damage to your laptop could be substantial.
- This product is offered to the end user in a non-enclosed OEM form, it is the responsibility of the end user of this product to ensure that they comply with local regulations pertaining to unintentional radio frequency emissions. In the United States, this pertains to FCC rules and regulations, Part 15, Section B.

Example Connections

Although there are a seemingly infinite number of types of control elements (potentiometers, switches etc.) that you can connect to the UMC, here are a few examples of how to connect the most common types.



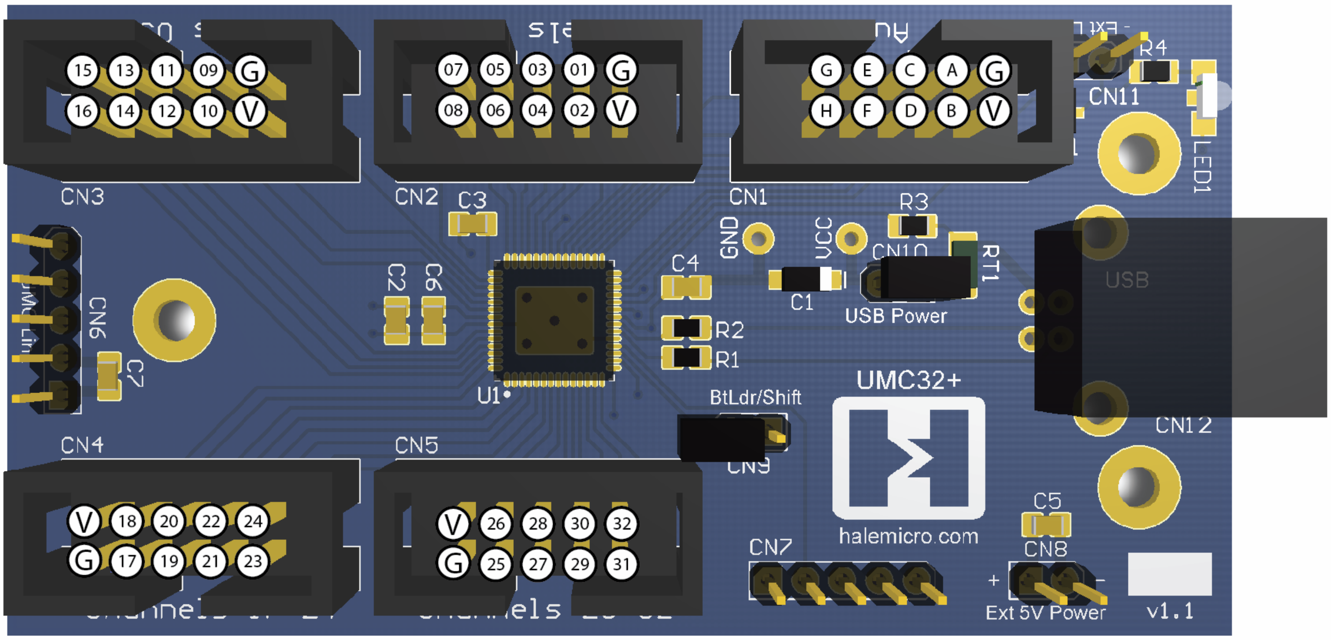
All pots are 10K ohm, linear, 1/4 watt
All +5 lines are tied together, all ground lines are tied together. (0.1uF capacitors shown are optional)

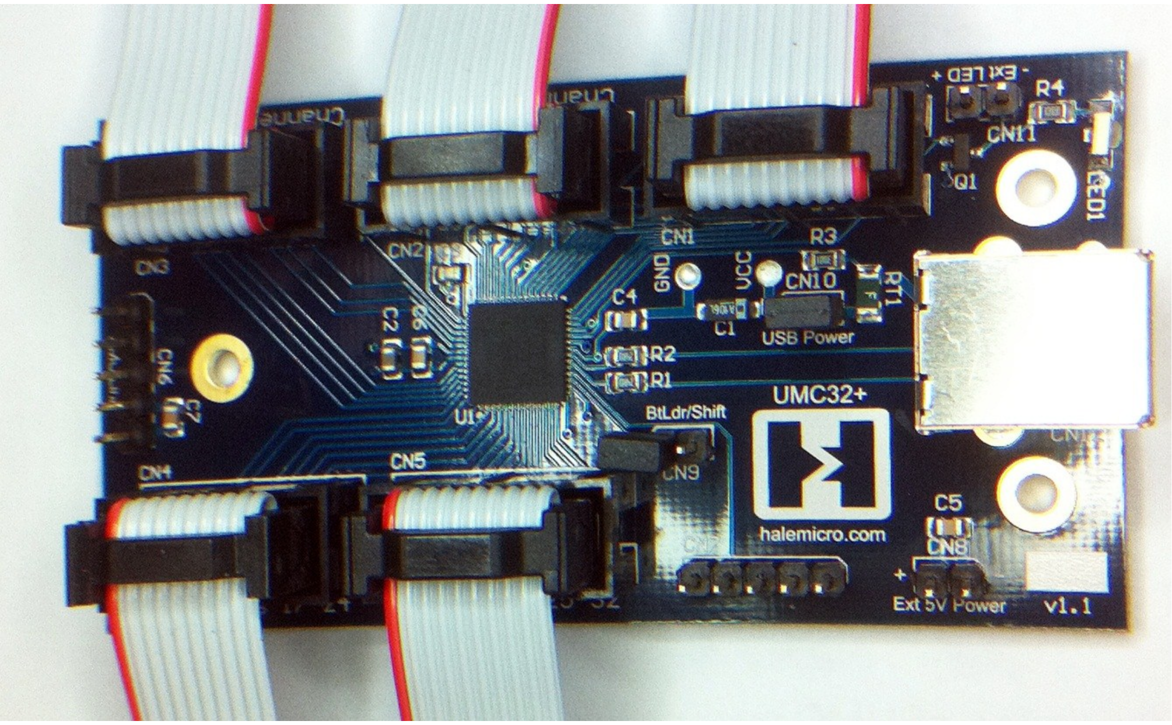


All switches are SPST normally open

** Please note that connection to +5V is not necessary for any SPST switch.*

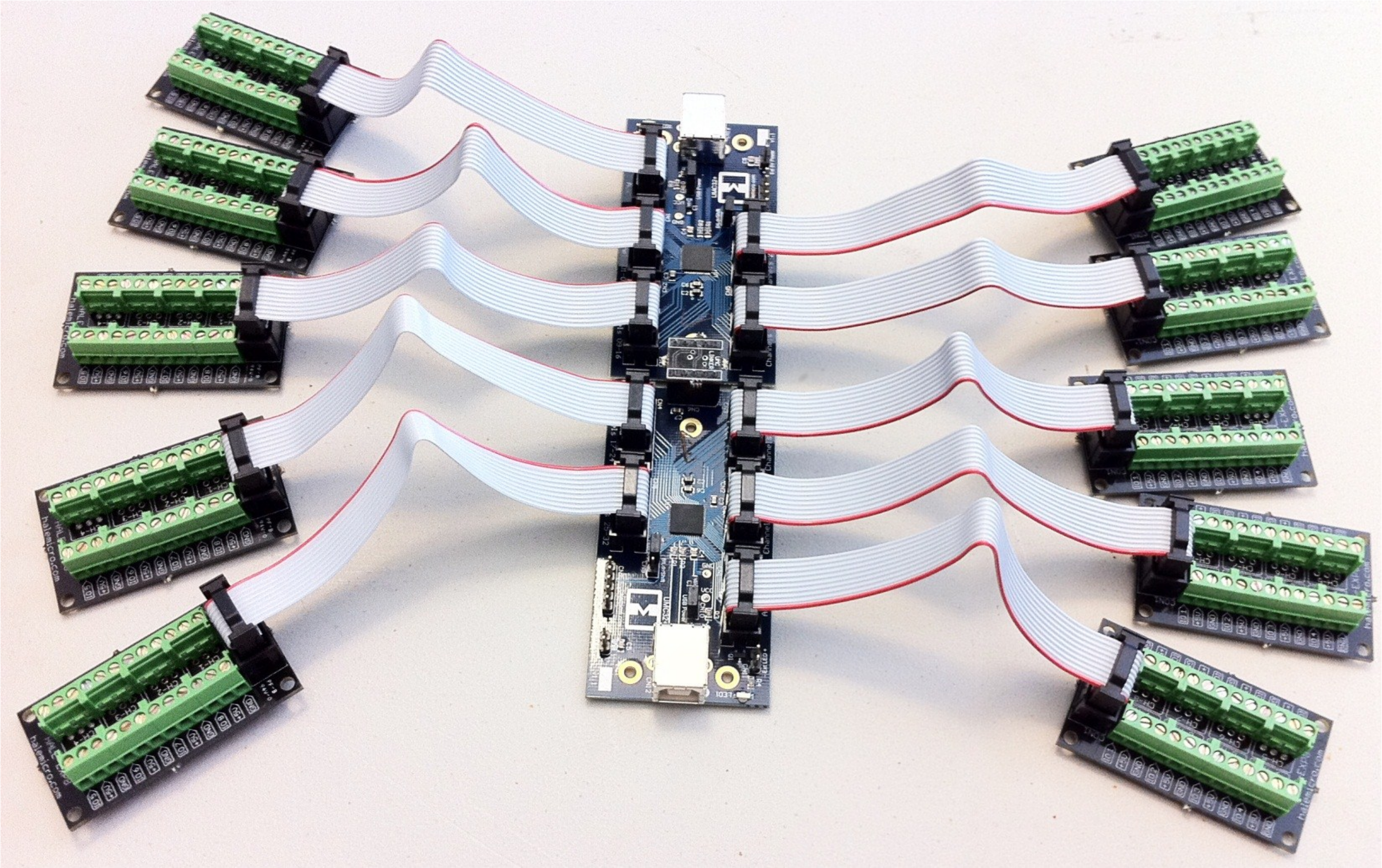
UMC32+ I/O Pin Diagram





UMC32+ in picture above with attached ribbon cable

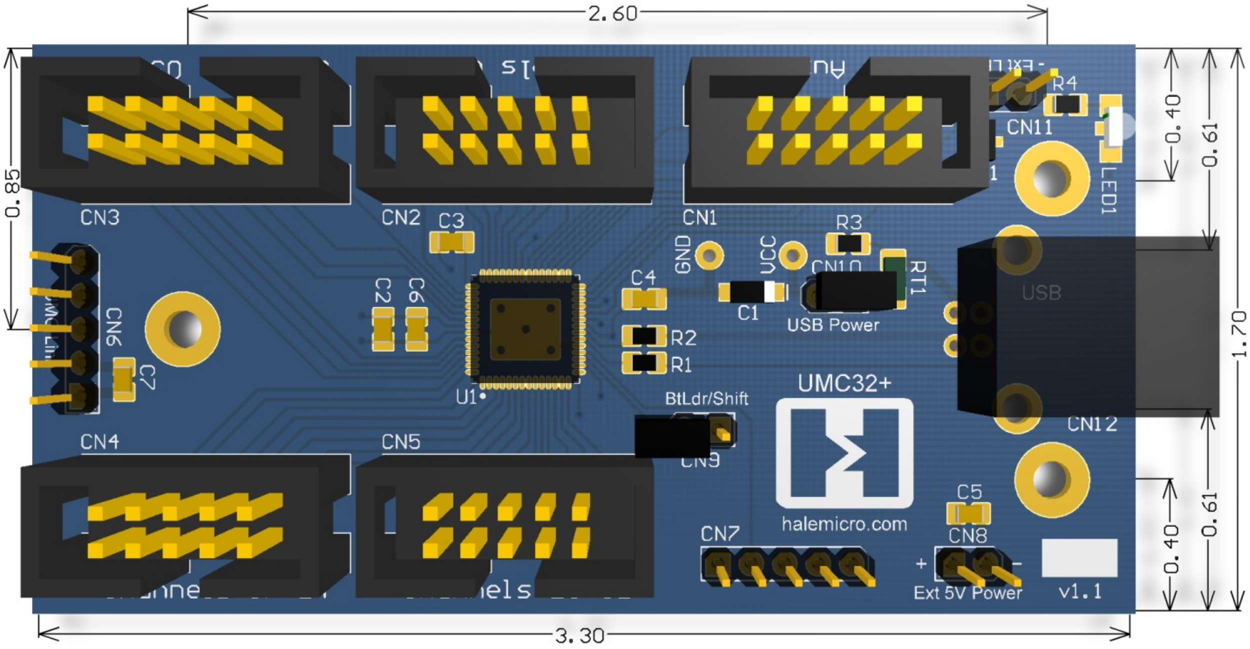
Ribbon cable kit sold separately (Part # RIB50)



Two (2) UMC32+'s with 10 EXP8SC screw terminal boards for 64 channel inputs plus an additional 16 switches on the Aux inputs (8 from master, 8 from slave). This provides a total of 80 controls using a single bus powered USB connector. Linker board (in center) connects the UMC's together. Using multiple processors (a dedicated processor for every 32 I/O's and 8 aux inputs) the UMC can achieve a scan rate that is 2X faster than a single processor. It should be noted that the UMC that is connected to the USB bus is the Master, while the linked UMC is the slave.

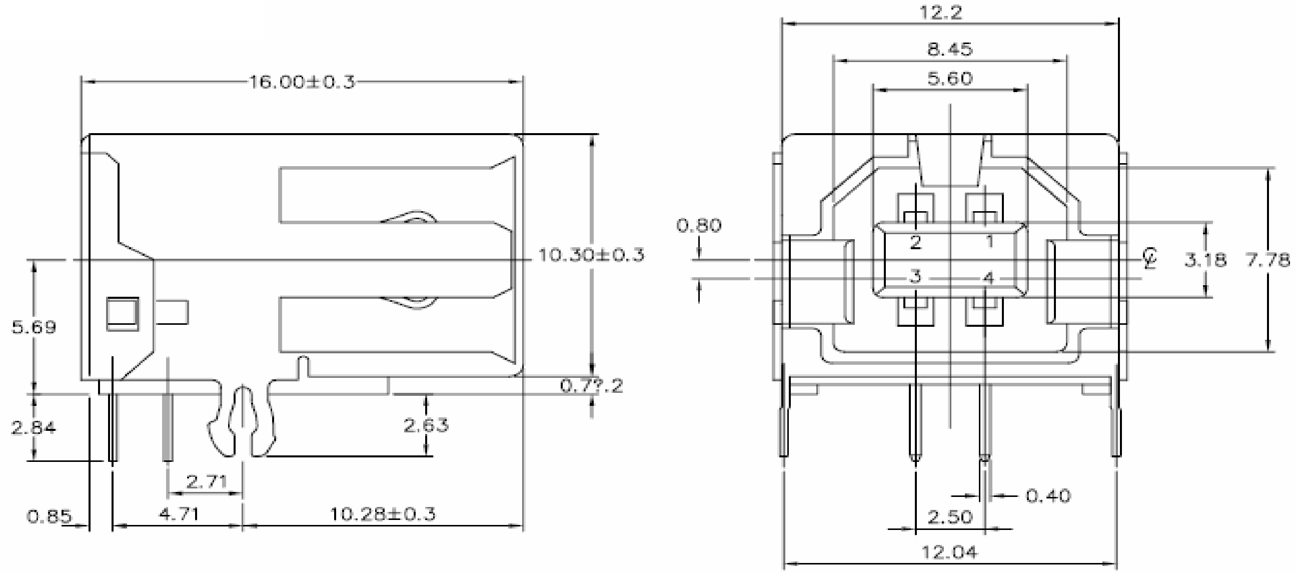
Mechanical Specifications

UMC32+ dimensions (inches):



The three 0.22” mounting holes for the UMC32+ are for #4 screws

USB Connector measurments (millimeters):



Electrical Specifications

Exceeding the absolute maximum ratings below may cause permanent damage to the device.

• Storage temperature:	-20ºC to + 70ºC
• Operating temperature:	0ºC to + 50ºC
• Operating voltage (power supplied via USB bus):	5.1 Volts DC max
• Max recommended current per I/O channel	8mA
• Maximum device current:	*500mA USB Bus Powered, 2A external.

Limited Warranty

Hale Microsystems LLC warrants this product to be free of manufacturing defects for a period of 90 days (USA only). If you feel that there is a defect with your device, please email technical support with a detailed description of the problem: support@halemicro.com. In most cases we will simply issue a replacement. It is the responsibility of the customer to cover the cost of shipping to and from Hale Micro. Devices shall only be accepted in original factory packaging. We will not accept the return of any product that has been modified beyond factory condition. Any attempt to modify your unit shall void this warranty.

Disclaimer

Neither the whole nor any part of the information contained herein or the product described in this datasheet may be adapted or reproduced in any material or electronic form without the prior written consent of Hale Microsystems. This product and its documentation are supplied on an as-is basis, and no warranty as to their suitability for any particular purpose is either made or implied. Hale Microsystems LLC will not accept any claim for damages whatsoever arising as a result of use or failure of this product. Your statutory rights are not affected. This product is not intended for use in any medical appliance, device or system in which the failure of the product might reasonably be expected to result in personal injury. This document provides preliminary information that may be subject to change without notice.

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.