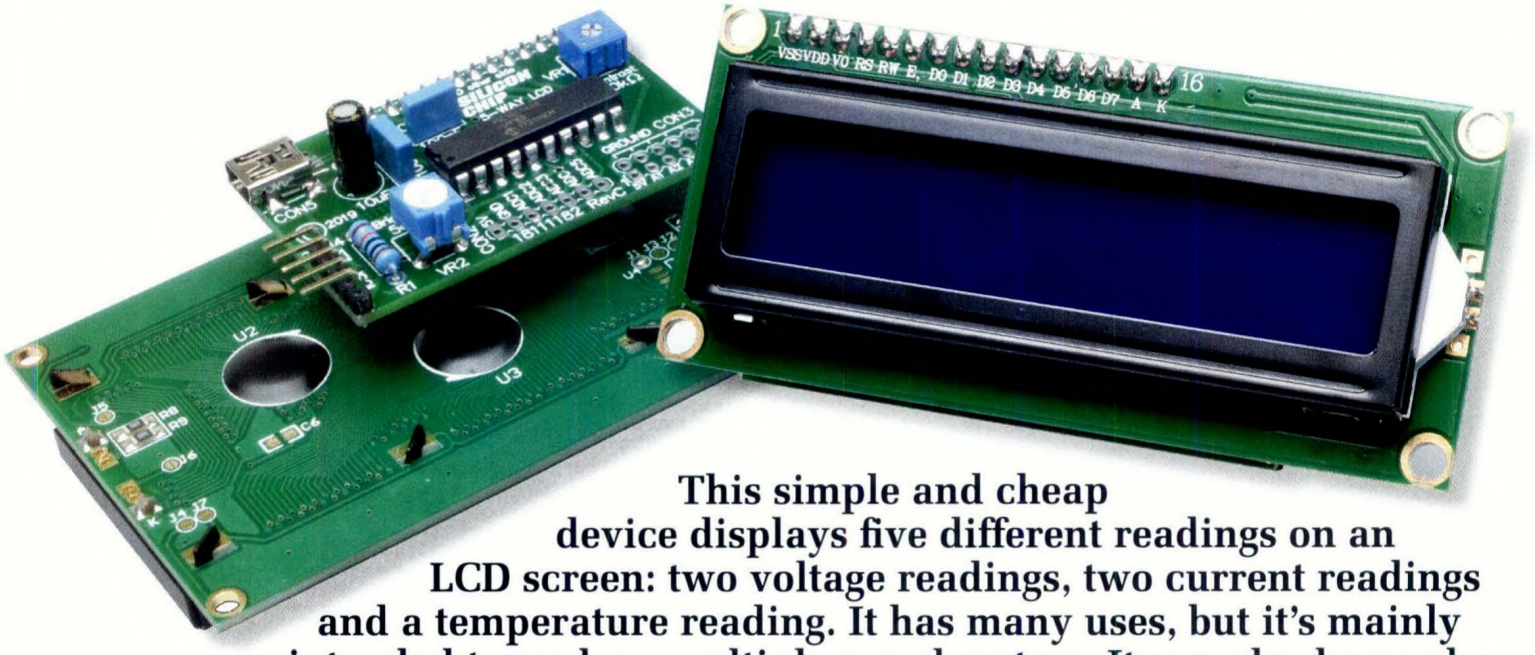
FIVE-WAY LCD PANEL METER AND USB DISPLAY

by Tim Blythman



This simple and cheap device displays five different readings on an LCD screen: two voltage readings, two current readings and a temperature reading. It has many uses, but it's mainly intended to replace multiple panel meters. It can also be used as a small additional text screen for a PC, Raspberry Pi or any other computer with USB.

We came up with this idea while working on the high-current linear power supply design that we started describing last month (part 2 starts on page 68 of this issue).

We needed a way to show several different voltage and current readings, along with heatsink temperature, and it just didn't make sense to use several panel meters for that job.

It's difficult enough to cut a single neat rectangle in the front panel of the instrument case to fit one screen, let alone three or even five. And there would be a lot of extra wiring if we used separate panel meters, plus increased current draw and it could end up pretty expensive.

This one low-cost device using a micro, an LCD screen and not much else makes the whole thing so much easier.

There are two ways to use this board. In the Bench Supply, we're feeding in five analog voltages with a common ground. These voltages are a fraction of the actual measured voltages (ie, the outputs of voltage dividers). The onboard micro samples these voltages and converts the values back to the original scales, then displays on them on the screen.

In the case of the fifth input, which is used for temperature sensing via an NTC thermistor, it also performs the required calculations to deal with the non-linear behaviour of the NTC.

In the other mode, the micro detects when it is plugged into a USB interface and then behaves differently. You send it text over a virtual serial link, which is shown on the display. So you can easily show whatever you want on the 16x2 or larger 20x4 character backlit LCD screen.

More details

Our 45V 8A Linear Bench Supply, mentioned above, has five main parameters to monitor. Those are the desired voltage and current, the actual output voltage and current (which may be lower than the desired values in some cases), plus the heatsink temperature.

It will automatically switch on fans if the heatsink gets hot, and throttle back its output in the worst case if that doesn't help. But it's still handy to have a way to tell how close to the wind you're sailing!

We settled on using a PIC16F1459 microcontroller to monitor and display these voltages. It's a low-cost micro

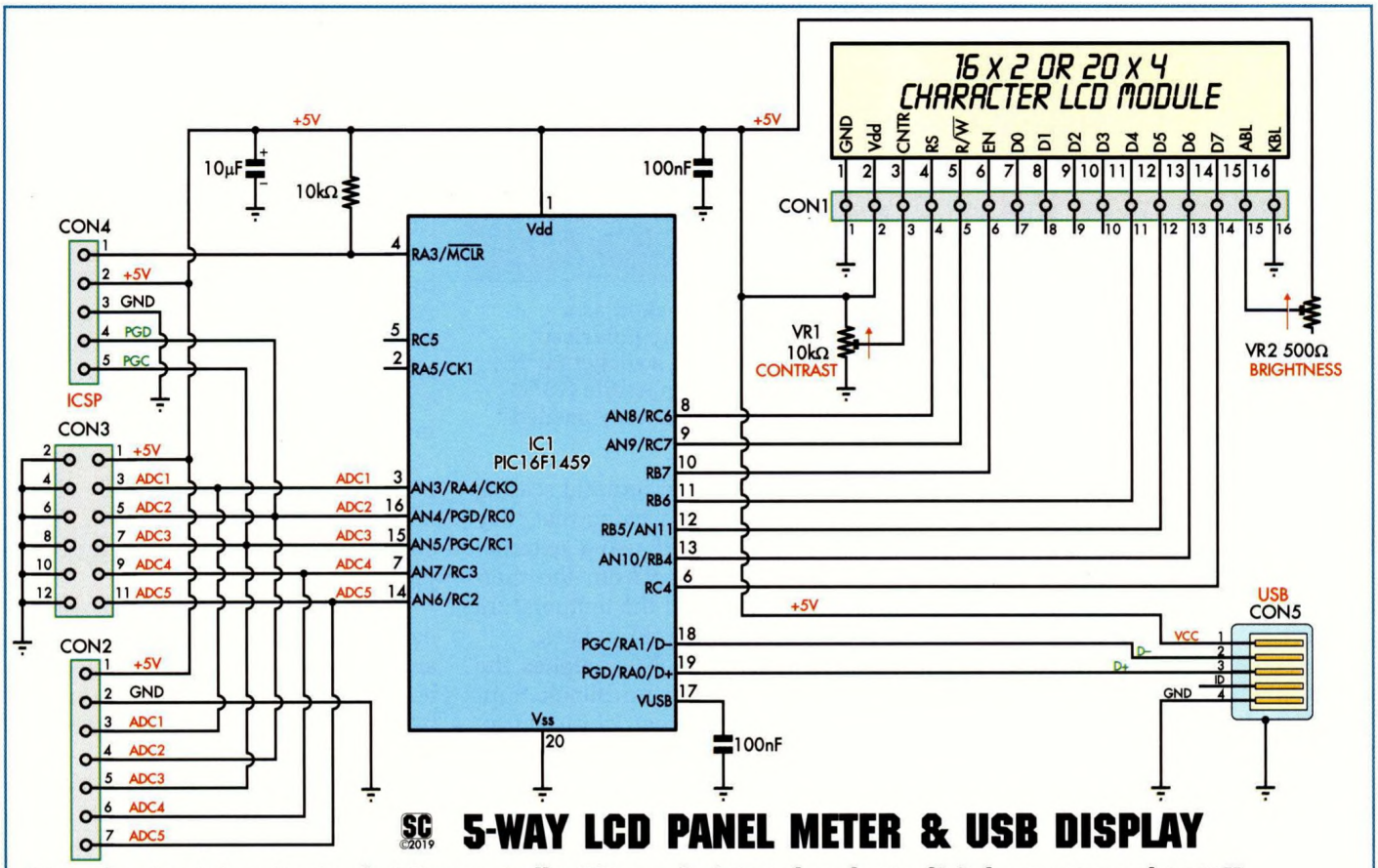


Fig.1: the circuit is quite simple. Microcontroller IC1 uses its internal analog-to-digital converter and 4.096V reference to measure the voltages at the ADC1-ADC5 inputs. It then scales the readings from ADC1-ADC4 and converts the reading from ADC5 to a temperature before updating the LCD connected via CON1. In USB Display mode, it instead receives text from a PC via CON5 and updates the display.

with some nice features. It's similar to the PIC16F1455, but it has more I/O pins, which makes it easier for us to interface with an LCD panel.

Both the 16F1455 and 16F1459 have USB interfaces, making it easy for us to implement the USB mode as a 'bonus' feature.

In this bonus mode, it is effectively a character LCD that can be controlled from your computer. If you want some extra information displayed 24/7 without needing to have a full-size monitor switched on and drawing power the whole time, it's an ideal solution.

It's even small enough to be mounted in a desktop computer's drive bay.

You could use it to display things like CPU load, memory usage, disk space usage, network activity, instant messages, unread e-mails... the list is virtually endless.

You just need to figure out how to get that information and send it to a serial port, and the display does the rest.

Circuit description

The circuit of the Display is shown in Fig.1. The aforementioned PIC16F1459 microcontroller is shown as IC1.

Its 5V power supply comes from either pin headers CON2/CON3, when used in the panel meter role, or CON5, the USB socket. The data pins from the USB socket are connected directly to pins 18 and 19, the dedicated USB data pins of IC1.

In the panel meter role, the five voltages are fed into either SIL header CON2 or DIL header CON3, whichever is more convenient. CON3 has the advantage that an IDC header on a 12-way ribbon cable can plug straight in, and each signal wire will have a ground wire on either side, minimising noise pickup

The five signal lines go straight to analog inputs AN3, AN4, AN5, AN7 and AN6 of IC1 (pins 3, 16, 15, 7 & 14). IC1's internal 10-bit analog-to-digital converter is used to read these 0-4.096V signals and convert them to digital values, with a resolution of 4mV ($4.096V \div 2^{10}$)

The 4.096V reference is within IC1, and we're using this rather than the 5V rail so that variations in the 5V supply do not affect these readings. That means we don't need to be concerned about how well regulated the 5V rail is.

These are scaled in software to the values shown in the spec panel, which are designed to suit our power supply, but these values will be useful for a range of low-voltage DC monitoring tasks.

You simply need to arrange for shunts to monitor currents, and dividers with approximately the right ratios (around 15:1 for voltages) plus trimpots for calibration, to feed the right voltage ranges to the panel.

An HD44780-based character LCD screen is connected via 16-way header CON1. It is driven in four-bit mode, with the RB6, RB5, RB4 and RC4 digital outputs of IC1 (pins 11, 12, 13 & 6) driving LCD data pins DB4-DB7.

We only need four data pins as these LCDs can operate in a four-bit mode, with the D0-D3 I/Os left floating or tied to ground.

Digital outputs RC6, RC7 and RB7 (pins 8-10) of IC1 drive the RS, R/W and EN pins of the LCD, controlling when the data is clocked and whether the LCD should treat it as an internal

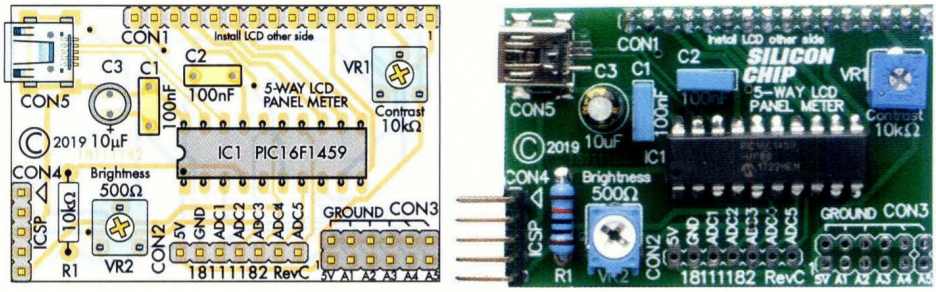


Fig.2: use this PCB overlay diagram and the same-size photo at right as a guide to help build the Panel Meter/USB Display board. The only polarised components are IC1 and the electrolytic capacitor. You can use a socket for IC1 if you want to. CON5 is not required for the panel meter version, while CON2-CON3 are not required for the USB Display version and CON4 is only needed if you plan to program IC1 in-circuit.

command or send it to the display.

Trimpot VR1 adjusts the LCD contrast voltage, while VR2 is wired as a variable resistor in series with the backlight LED, allowing its brightness to be set.

The power supply is simple. There is a 10µF bulk bypass capacitor for the 5V rail, which is the maximum value allowed to meet the USB inrush current specification. IC1 has its own high-frequency 100nF local bypass capacitor.

A 10kΩ resistor pulls up the micro's MCLR pin to 5V to prevent spurious resets, while a 100nF capacitor between pin 17 (VUSB) and ground stabilise its internal USB 3.3V regulator.

The ICSP header, CON4, is provided to allow the PIC to be programmed without having to be removed.

Using it as a panel meter

On power-up, if no USB connection is detected, it will initialise the LCD and show a splash screen. The analog pins are set as inputs, and the analog-to-digital converter (ADC) voltage reference is set to the internal 4.096V fixed reference.

After a second, it begins sampling the analog pins around five times per second. The update interval gives a quick update time, but not so fast that the numbers would blur into each other while changing.

The current and voltage values are converted using fixed internal scaling factors, with the idea being that they have been fine-tuned using external trimpots. The reading from the thermistor is used to find the temperature in a look-up table stored in flash.

The header on CON3 matches the pinout of CON6 on the Bench Supply to allow a direct connection. If the input assignments, scaling ratios etc do not suit your particular application, you can download the source code from our website and change it to better suit your needs.

It is written in the C language. Microchip's MPLAB X IDE software is a free download, and there is a free version of the XC8 compiler (plus a trial mode for the full compiler).

Once you have installed that software, you can open up the project, make some changes to the code and then 'Build' the project to produce a new .hex file for IC1.

We used MPLAB X IDE Version 5.05 and XC8 Version 2.00 and our compiled HEX file was very close to that 8kB limit. We suggest using the same version to avoid going over this limit.

Using it as a USB display

When connected as a USB display, neither CON2 or CON3 are needed as the analog pins are not sampled. On power-up, IC1 enumerates on the connected USB port as a USB-serial device and appears as a serial port to the host.

For example, this would be a COM port on Windows or a TTY device on Linux. The LCD is initialised and blanked and a default set of character graphics are loaded into code points 0-7.

When data is received from the host, (for example, if you were typing into a serial terminal program), it is processed by IC1 and used to update the display. ASCII characters are passed straight on to the LCD, while control characters such as CR (carriage return, ASCII code 13) and LF (line feed, ASCII code 10) move the printing location as expected.

TAB moves to the next screen position which is a multiple of five characters, while FF (form feed, ASCII code 12) moves the cursor to the home position.

Backspace (ASCII code 8) moves back one position, but does not erase anything. A true erasing backspace can be simulated by a backspace, space, backspace sequence consisting of ASCII codes 8, 32 and 8.

Finally, Escape (ASCII code 27) clears the screen, but does not move where the display will print next. Thus a sequence of ESC and FF returns the display to the same state as it is when it first starts up.

The entire display is held in a RAM buffer and sent to the LCD one character at a time, to ensure that the USB peripheral is not left waiting too long for the display to update. This could otherwise happen if the display needs to be cleared and many characters need to change at the same time.

While this might sound slow, the display can still fully update around 10 times per second.

The RAM buffer consists of four rows of 20 characters, as this is the largest display size that the HD44780 controller can manage. Text wraps around at the end of a line and back to the top at the end of the last line. If a smaller display is fitted, it will appear the same as the top, leftmost corner as a larger display would appear.

Thus the display operates fairly in-

Features & specifications

** Shows two voltages, two currents and one temperature reading on a 16x2 LCD*
** In alternative USB mode, ASCII text from virtual serial port is written directly to 20x4 LCD*
** Panel meter input range: 5 x 0-4.096V*
** Panel meter scaling: 2 x 0-4.096V -> 0-60V, 2 x 0-4.096V -> 0-9A, 1 x 0-3V -> 0-100°C*
** Panel meter resolution: 58.6mV for voltages, 8.79mA for currents*
** Panel meter update rate: 5Hz*

tuitively and can be easily controlled by any software that can write to a serial port. No data is sent by the USB display back to the host, so the receiving program should not expect to deal with this.

Construction

The Panel Meter/USB Display is built on a double-sided PCB coded 18111182, which measures 56 x 36mm.

The PCB overlay diagram, Fig.2, shows where to fit the components. As noted above, some parts can be left out for some applications. We will describe the installation of all parts, which will allow the unit to be used as either a panel meter or USB display.

The only surface-mounted part is the USB socket, and it should be fitted first. A soldering iron with a fine tip will make this easier. We recommend that you have flux and solder wick (braid) on hand for this step. A pair of tweezers can be helpful too.

Apply flux to the four pads on the PCB for the USB data and power signals. These are the four parallel pads to the right of the socket. Place the socket on the PCB; it should lock into place due to the two small posts on its underside.

Carefully apply solder to the pads and pins, ensuring all four are well attached. If there is any bridging, apply more flux and use the solder braid to remove it. Then apply flux paste to the four larger mechanical pads and solder them to their respective pads too. They are larger and will need more heat.

Next, mount the single resistor, followed by the two non-polarised 100nF capacitors. Follow with the electrolytic capacitor, which is polarised. It must be installed with its longer positive lead to the pad marked "+" on the PCB.

Now fit trimpots VR1 (10kΩ, "Contrast") and VR2 (500Ω, "Brightness"). Push them down, and they should both snap into place, after which you can solder their pins. If you are using a socket for IC1, install this next, ensuring the notch goes to the end closest to the USB socket.

If you have fitted the socket, gently straighten IC1's pins so that it will slot into the socket, then plug it in. If soldering IC1 directly to the PCB, start with two diagonally opposite pins. Once you are happy that the IC is flat against the PCB and oriented correctly, solder the remaining pins.

You can now mount CON2 and CON3. For CON2, you could use either a header or socket, while CON3 is designed to be fitted with a double-row male header to allow an IDC socket (plug) and cable to be attached.

The PCB sits neatly within the footprint of the 16 x 2 LCD panel, leaving the mounting holes clear. Using a female header on the PCB means it can be removed if necessary.

If fitting CON4, do so next. You can use a straight or right-angle header; we prefer the right-angled variety in this role as it allows the programmer to sit flat when connected. CON1 should be fitted last, as it also needs to be attached to the LCD. You may choose to solder it directly, or use a female header socket on the panel meter PCB to allow the LCD to be removed.

We recommend attaching the male header to the LCD first by soldering one pin and ensuring it is straight and flush with the LCD's PCB. Then solder the remaining pins. Before soldering the LCD to the main board, check that its pinout matches that shown in our design.

Most LCDs with a SIL header should have a pinout that matches ours, but checking this now can save much troubleshooting later if you somehow have one that's different (see Fig.1). If there is a pin mismatch, you can solder only the matching pins and then use insulated wire to make the remaining connections.

You may like to slip a piece of card between the two to maintain spacing while soldering. Check that your boards are orientated the same way as in our photos. If you are using a header socket to attach the LCD, plug in the male header before soldering. This will allow you to check that all the clearances are correct.

Programming IC1

This step is not necessary if you purchased a pre-programmed PIC.

You can use a PICkit 3, PICkit 4 or SNAP programmer to flash IC1 on the board via the ICSP header (CON4).

As we wrote in our SNAP review (May 2019; siliconchip.com.au/Article/11628), the SNAP programmer cannot provide power to the micro and

Parts List – Five-way LCD Panel Meter/Display

1 double-sided PCB coded 18111182, 56mm x 36mm#
1 16x2 character LCD with backlight (for Panel Meter, eg, Jaycar QP5521) ***OR***
1 20x4 character LCD with backlight (for USB Display, eg, Jaycar QP5522)
1 16-pin male header (CON1)
1 16-pin header socket (optional, to allow LCD to be unplugged)
1 7-pin header or header socket (CON2; optional)
1 2x6-way pin header (CON3; not needed for USB Display)
1 6-way right-angle header (CON4; optional)
1 SMD mini-USB socket (CON5; not needed for Panel Meter)

Semiconductors
1 PIC16F1459-I/P microcontroller programmed with 1811118A.HEX#

Capacitors
1 10µF 10V electrolytic
2 100nF MKT or multi-layer ceramic

Resistors
1 10kΩ 1/4W 5%
1 10kΩ mini horizontal trimpot (VR1)
1 500Ω mini horizontal trimpot (VR2)

#Programmed micros and PCBs are available from the SILICON CHIP ONLINE SHOP

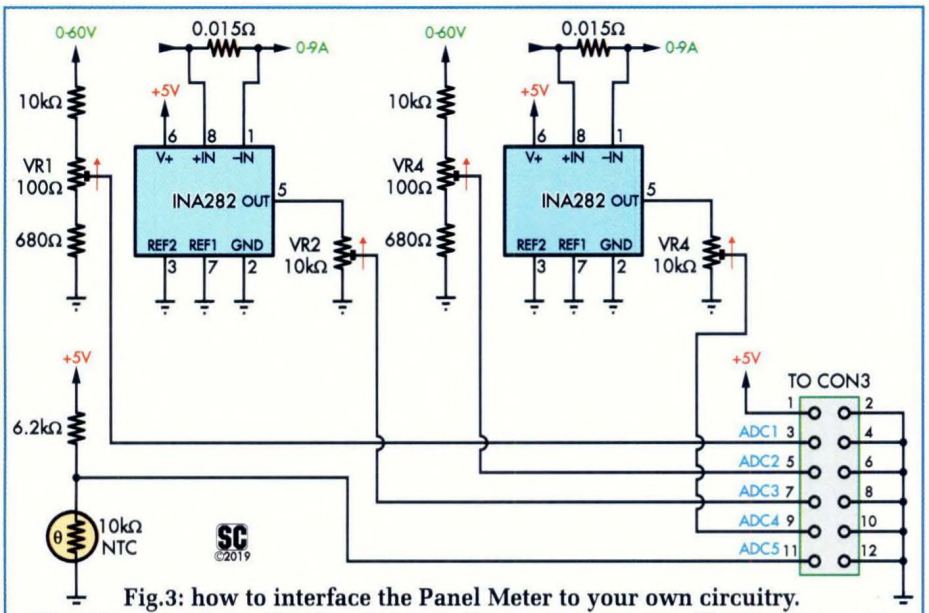


Fig.3: how to interface the Panel Meter to your own circuitry. If you're using low-side shunts to monitor current, you could use op amps to amplify the voltage across them to a suitable level for feeding to the Panel Meter. You could also use lower value shunts in combination with a higher-gain shunt monitor for less heating and power loss.

only supports low-voltage programming. So if you are using the SNAP programmer, you need to provide power via another source, such as the USB socket, and ensure that the low-voltage programming option is selected in the software.

Regardless, you will need Microchip's IPE (integrated programming environment), which can be downloaded as part of the MPLAB X IDE from: www.microchip.com/mplab/mplab-x-ide

In the IPE, select "16F1459" in the device drop-down menu and your programmer from the tool menu, if it isn't already selected.

Connect the programmer to CON4 on the PCB, lining up the two pin 1 indicator triangles.

Then click the "Connect" button in the IPE and ensure that the connection is successful, according to the display in the lower output window.

Then you just need to open the HEX file and click the "Program" button to upload it to the chip.

Connecting the panel meter

Details for connecting the panel meter to the Bench Supply are included in that article.

If you wish to use it for another purpose, then connect the 5V and ground pins to a 5V supply and the five analog pins to sources of appropriate analog voltages.

The ADC1 and ADC2 inputs are scaled to display 0-60V for an input of 0-4.096V, while ADC3 and ADC4 are scaled to 0-9A for 0-4.096V.

You will need to use a 10kΩ NTC thermistor wired as a divider with a 6.2kΩ resistor across a 5V supply to feed the ADC5 input if you are to get meaningful readings.

Fig.3 shows our suggested circuitry for interfacing with the Panel Meter. If you're using a different shunt value, you will need to use a different shunt monitor IC, or provide some gain at its output, to get at least 4.096V for a current of 9A, giving the correct scaling. That's regardless of whether your circuit will reach 9A.

Once the Panel Meter is connected to such a circuit, it simply converts the analog inputs and displays the measured values, and no other action is required. You may need to adjust the contrast and brightness, as described below.

Using it as a USB display

To use the unit as a USB display, simply plug it into a computer with a mini type-B to type-A USB cable.

You may need to install a driver, in which case the same driver is used as for the Microbridge. This is because the Microbridge uses the similar 16F1455 microcontroller in a similar role.

This should not be necessary for Windows 10, Linux or Mac users.

If needed, the driver can be downloaded from: www.microchip.com/wwwproducts/en/MCP2200

Once the driver is installed and the USB device enumerated on your system, it can be tested by using a serial terminal program such as PuTTY, TeraTerm or even the Arduino Serial Monitor.

Open a connection to the appropriate port and type characters into the terminal. You should see them appear on the LCD. If not, you may need to adjust your LCD's contrast and brightness.

The baud rate is not critical as the virtual serial port enumerated by IC1 does not use this information (as it might if it were connected to a downstream hardware UART).

Contrast and brightness

No matter what the brightness setting, the backlight LED should be on. If you cannot see anything on the display, the contrast probably needs to be adjusted.

Turn VR1 until characters can be seen clearly against the background. Once the characters are clear, you can then tweak the brightness.

On the unit we have built, we had good contrast with around 1.8V on pin 3 of CON1, although this may vary depending on the specific display module used in your LCD.

If you have built the USB display and cannot see any characters, make sure you have sent some data to the terminal. If it is still not working, there may be a problem with the construction, probably to do with the LCD if the USB side is enumerating correctly.

Conclusion

While this was originally designed to replace multiple panel meters for our Bench Supply project, we've also turned it into a handy accessory for a computer.

It goes to show just how versatile the PIC16F1459 is. SC

When configured as a Panel Meter, the display should look like this, with voltage, current and temperature readings. If using it as a USB Display, the screen will be blank until it receives data from the PC via its USB serial port.