By
Peter Bennett

A TOUCHSCREEN CAR ALTIMETER

This modified version of Jim Rowe's Touchscreen Altimeter is optimised for use in a car, truck or other land-based vehicle, rather than a glider or ultralight aeroplane. The hardware has been simplified and adapted to be powered from the vehicle, while the software has been updated to make its readings more accurate on a typical driving trip.

This is a modified version of the Touchscreen Altimeter and Weather Station project from December 2017 (siliconchip.com.au/Article/10898), to better suit car usage.

You might be wondering why I want an altimeter in my car. I find it interesting to know how high I am when driving in the mountains, especially when stopping at lookouts (some have their altitude posted, but not all).

Also, engine performance is reduced at altitude, so the information may do more for you than just satisfy your curiosity.

The power output of naturally aspirated petrol engines drops by about 3-4% per 300m (1000ft); turbocharged engines are less affected, but can still lose some power due to the thinner air at higher altitudes, depending on their particular design.

In a motor vehicle, the Altimeter can be powered from the vehicle's accessory socket, so there is no need for the internal battery used in the original design. This means that we can fit all the hardware in a single UB3 Jiffy box, with an exhaust fan to remove the heat generated by the display, avoiding the need the mount the sensors in a separate box.

In this design, power is supplied via a USB cable. Many modern cars have USB charging sockets. If yours doesn't, you can use a USB charger plugged into the accessory socket.

You can buy low-cost pre-built altimeters but they are not very accurate. That's because they typically convert the air pressure reading to altitude with reference to "Mean Sea Level" (MSL), a pressure of 1013.25hPa. But sea level pressure can vary (in extreme weather) from 870hPa to 1084.8hPa, an error range of 1770m/5800ft.

Of course, we seldom see the extremes, but you can see that basing an altitude reading on MSL will often lead to significant altitude errors.

To solve this, I have modified the Altimeter software so that you can set the local altitude, such as the altitude of your driveway or a lookout (it's usually given), to give a very accurate reference pressure, your local QNH.

The original Altimeter software stored the QNH setting when you turned it off, and loaded it again at startup.

If you drive to a pretty spot for a picnic and shut the Altimeter down, it will restore with the same QNH

Here's the altimeter built into the standard (DIN) dash cutout in my car. Being such a large screen, it's very easy to read. As the screen says, you can change both the mode and units (eg, feet above sea level, as seen here [which is used in aviation] to metres above sea level, which we're all familiar with). Incidentally, QNH means the atmospheric pressure adjusted to mean sea level. It is neither constant nor the same for various locations – you can get the QNH from weather services.

and preferences when you power up to depart.

But if you stayed overnight, the QNH will probably be significantly different when you set off in the morning, leading to errors that accumulate with each stop.

To solve this, the Vehicle Altimeter software records the ground altitude when you power down and uses this value to compute the new QNH on power-up.

The assumption is that the vehicle does not change altitude while you are not driving it (hopefully, a safe assumption!). So the unit should remain accurate for a whole trip, as long as you set its altitude correctly at the start.

This saves you from having to frequently check the current QNH at your location (via the internet, for example) and update the unit to maintain accuracy.

The Car Altimeter is sized to fit into a typical car console pocket (eg, it fits nicely in the console of a Mazda 6). The pocket has an accessories outlet which is hidden, along with the USB adaptor, to the left of the Altimeter.

Circuit changes

The modified Altimeter circuit is shown in Fig.1.

In addition to the Micromite LCD BackPack, DHT22 temperature/humidity sensor and BMP180 temperature/pressure sensor retained from the previous design, the following elements have been added: a fan with PWM speed control, a small Li-ion battery and a relay driven by a Mosfet plus several diodes.

The PWM control circuity for the cooling fan is provided to keep its noise to a minimum, as small cooling fans are notoriously noisy. This is based on a standard NPN transistor, Q1, driven from Micromite pin 24 via a 2.7kΩ resistor. Schottky diode D5 prevents back-EMF spikes from the fan damaging Q1.

The software uses a PWM frequency of 20Hz with a 50% duty cycle. This gives adequate airflow with minimal noise.

So that the unit can save the altitude at power down, we need to monitor the 5V supply and detect when it starts to drop. Since it drops too fast to give the software enough time to save its settings, rechargeable lithium-ion button cell BAT1 powers the circuit while the 5V rail collapses.

When we have finished storing the data, we switch off the battery supply.

There is another benefit of this battery. The effect of the starter motor on the electrical system of a vehicle can be severe, and the 5V supply can fluctuate enough to upset the Altimeter. By diode isolating the 5V rail from the USB input, and using the lithium-ion battery to provide a stable 3.3V supply, we get a reliable boot-up.

Jumper JP1 is used as a connector to access the 5V supply from the USB socket and to feed 5V back into the BackPack, which flows between these

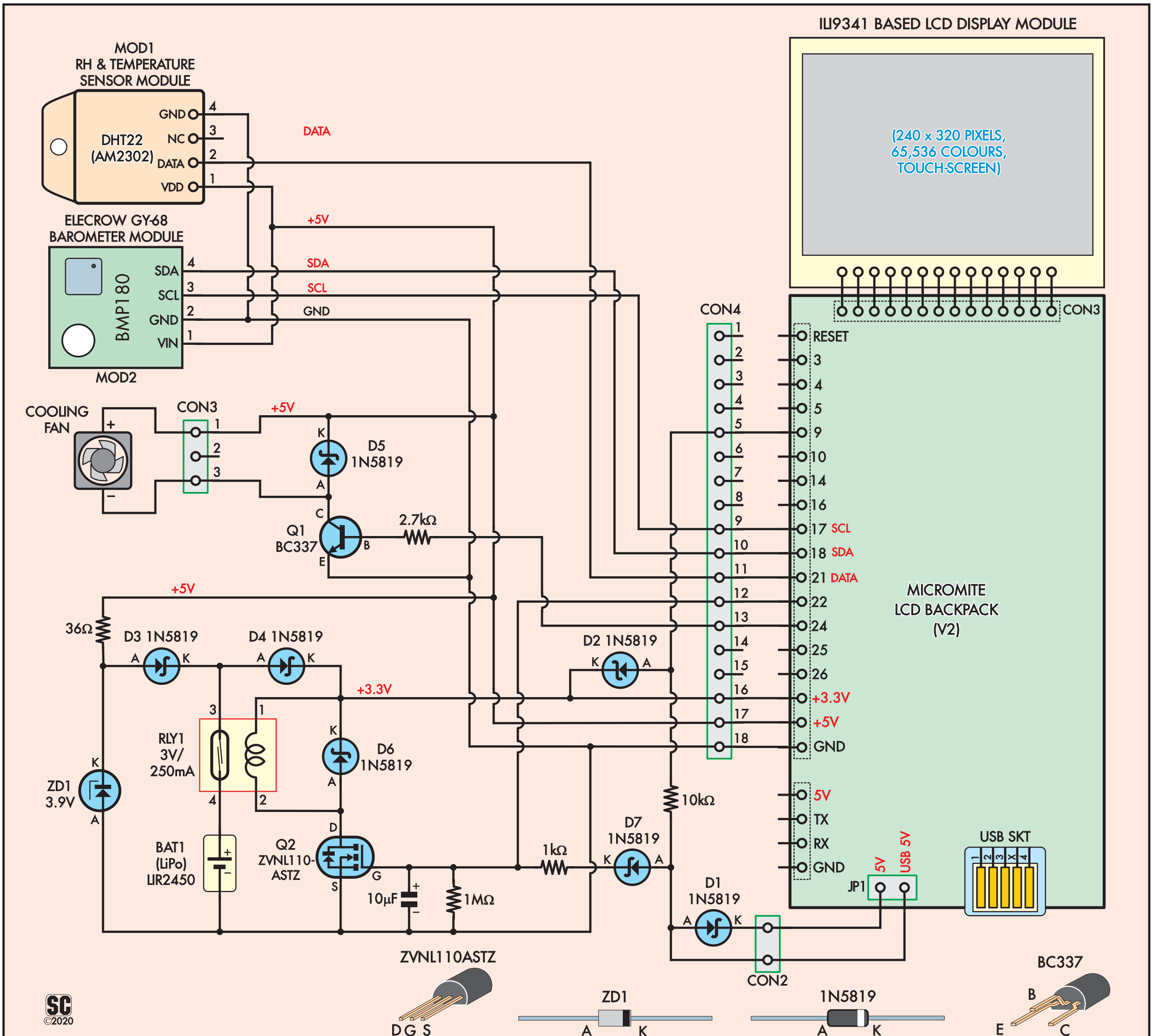


Fig.1: the Car Altimeter circuit is based on that of the Touchscreen Altimeter for Ultralights, but it has been optimised for use in land-based vehicles. This includes the addition of a small PWM-controlled fan to ensure the sensors see fresh air, and a backup battery (BAT1) switched by RLY1 to provide power for a brief time after switch-off, so that the current altitude can be saved into flash memory.

pins via schottky diode D1. The USB +5V also goes to the gate of Mosfet Q1 via another schottky diode (D7) and a 1kΩ resistor. This ensures that Q2 switches on as soon as USB power is available, and it powers the coil of relay RLY1.

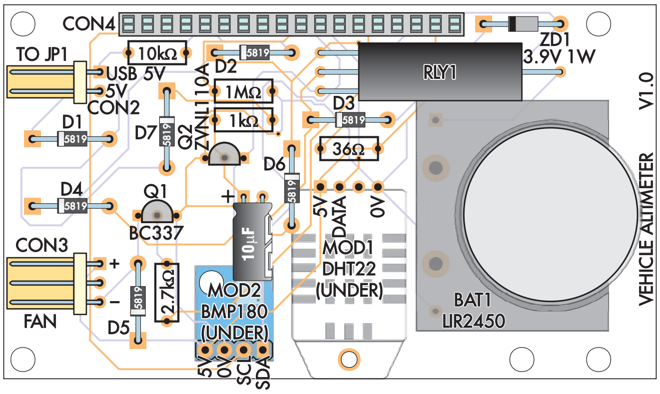
When the 3.3V rail is derived from battery BAT1, the 5V rail sits at 3.3V; it is back-fed through the 3.3V regulator on the BackPack board, from its output to its input via an internal protection diode. D1 prevents this 3.3V from being backfed into the 5V USB source.

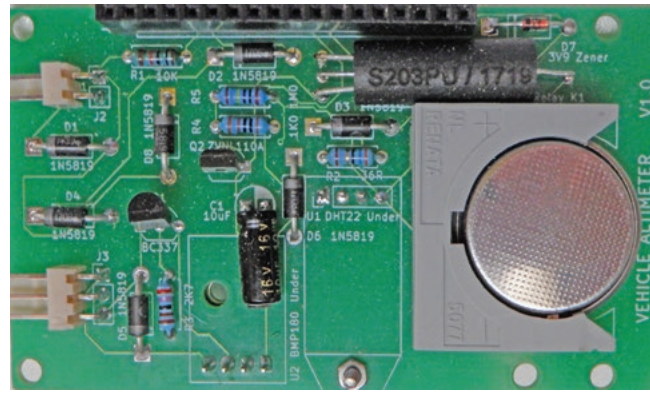
RLY1 connects BAT1 into the circuit when Mosfet Q2 is on. BAT1 is charged from the 5V rail via a 36Ω current-limiting resistor and schottky diode D3. Zener diode ZD1 limits the voltage applied to the battery to a safe level for charging (around 3.6V, taking into account the forward voltage of D3).

BAT1, in turn, powers the +3.3V rail of the BackPack via schottky diode D4. The voltage drop across D4 reduces the 3.6V from the battery to the 3.3V needed. This rail mainly runs the PIC32 micro on the BackPack, which has a recommended maximum of 3.6V and an Absolute Maximum rating of 4.0V.

Micromite pin 9 is used to sense the 5V USB voltage via a 10kΩ resistor, to determine when the external 5V supply switches off, and Micromite pin 22 is pulled low to forcibly bring the gate of Q2 low, switching RLY1 off and powering down the circuit.

One thing not shown on the circuit is that I added a front panel LCD backlight dimming switch to the BackPack. This connects across the BackPack's onboard brightness adjustment trimpot (VR1), shorting it out when the switch is closed and thus selecting between two different





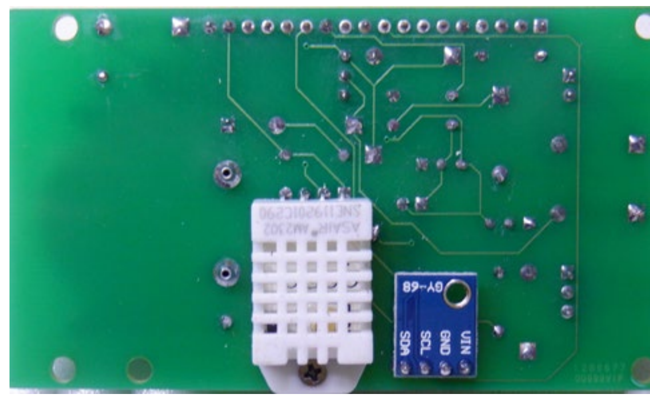


Fig.2: to make assembly easy, all the components which are not part of the Micromite LCD BackPack mount on this similarly-sized PCB, with matching front and back photos at right. Only the two sensors are mounted on the back – everything else is mounted on the front of the PCB, including the cylindrical SPST relay (black component top right of upper pic at right) and the rechargeable button cell holder.

brightness levels: that set by VR1, and full brightness.

This is important so that you can switch the backlight to low brightness at night, to avoid ruining your night vision.

Software changes

The software has been changed in a few places, and some of the changes have been described above. Some improvements have also been made to the user interface.

The weather station and altimeter screens are similar to the original. They show altitude above MSL until the QNH or exact altitude has been entered. After that, they show altitude above QNH (Screen 1 & Screen 2).

The Change Mode screen has new selections that differentiate between entry of QNH and current altitude (Screen 3). The current QNH value is also shown while you enter either current altitude or QNH (Screen 4).

If you want to change the fan PWM frequency or duty cycle, search the BASIC code for the line starting with PWM and change the values of 20 (Hz) or 50 (percent duty cycle) to suit.

Power supply

This Vehicle Altimeter draws about 90mA at 5V. It can be powered from a low-cost USB power bank (such as Jaycar Cat MB3792), providing run times in excess of 24 hours between charges, making this version practical for use outside of a motor vehicle.

Loss of USB power is detected by pin 9 of the Micromite, with a 10kΩ resistor and diode D2 clamping this

Using Weatherzone to get QNH

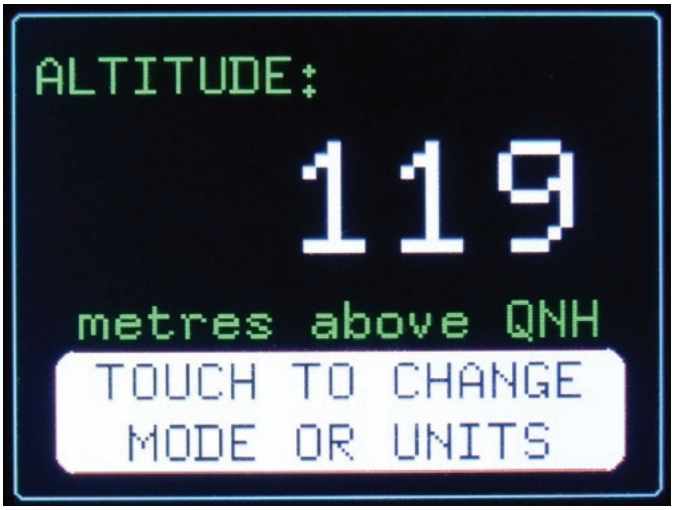
Weatherzone (weatherzone.com.au) is a free mobile app for viewing weather forecasts and related information. It also provides a simple method for getting QNH.

In this example, the screengrab on the left shows the observations at Terrey Hills; there is no QNH observation, so the Pressure field is blank.

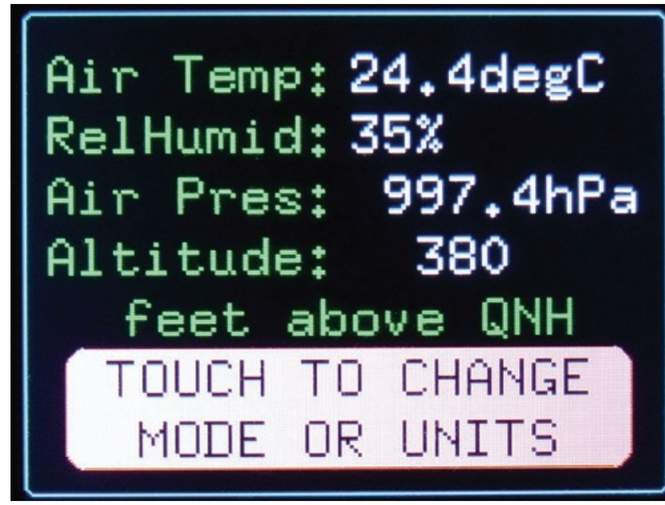
Tapping on the screen takes you to the nearest location with data, which is Sydney. The second screen grab shows that this indicates the current QNH value.

If you want higher accuracy, use the Weather Observations screen for your area from the Bureau of Meteorology. (www.bom.gov.au).

The BOM gives QNH to 0.1hPa resolution.



Screen 1: the main screen after setting QNH. This shows your altitude above QNH (effectively sea level) in metres or feet.



Screen 2: the extended information screen after setting QNH, showing the altitude in feet along with air temperature, relative humidity and atmospheric pressure readings.

signal to the 3V3 rail, as Micromite pin 9 is not 5V-tolerant. Power to the Micromite is held on for a short time after the loss of USB power due to the 10µF capacitor at the gate of Q2, which slowly discharges through its parallel 1MΩ resistor. During this time, the Micromite runs from BAT1.

The change in level at Micromite pin 9 triggers a software interrupt that causes the Micromite to store the current altitude data. Micromite pin 22 is then switched low, turning off Q2 and releasing the relay, shutting everything down. Diode D6 suppresses any voltage spike across the relay coil.

In practice, the Micromite runs for about 200ms after a loss of 5V power. This gives the BackPack time to send the message "Saved" to a terminal attached to the USB cable before the 3.3V supply goes away. You will notice the display dimming briefly as the display backlight runs from 3.3V rather than 5V before it switches off.

Note that the selection of Mosfet Q2 is not critical. Any N-channel enhancement mode Mosfet with a continuous drain current of at least 300mA and a maximum gate-threshold voltage up to 2.0V (typically those designed to be driven from a 3.3V logic supply) should work as well as the ZVNL110A.

However, we have not tested any substitutes.

Construction

I have designed a double-sided PCB which holds all the components of the Vehicle Altimeter, as shown in Fig.2 and the accompanying photos.

The two sensors (BMP180 & DHT22) mount on the back. This keeps the sensors away from the heat-producing components, in a dedicated cool airstream between an inlet and outlet in the case. This board plugs directly into the LCD BackPack.

Start by begging, borrowing or building the BackPack. We suggest you build V2, although the original will work. We don't recommend using V3 as the Altimeter software is not designed to accommodate the larger screen, and the inside depth of the V3 box is reduced because of its recessed front panel.

The BackPack V2 construction is fully described in SILICON CHIP, May 2017, starting on page 84 (siliconchip.com.au/Article/10652).

But given its relative simplicity and the fact that a kit is available and the PCB silkscreen shows where the components go, you don't really need to read that article. Simply fit the components where shown on the PCB, and it should work.

Once you've built and tested the BackPack, wire up a toggle switch across trimpot VR1 so that when the switch is closed, VR1 is shorted out and the LCD screen operates at maximum brightness. When it is off, the brightness is set by VR1, which you should adjust to a comfortable level for nighttime viewing.

Note that there are two otherwise identical versions of the 2.8-inch 320x240 LCD touchscreen, one of which uses backlight current control and one which uses voltage control.

If the 100Ω trimpot supplied for VR1 does not adjust the backlight brightness properly, replace it with a 100kΩ potentiometer and wire its unconnected pin to ground. That should do the trick.

Now assemble the interface board by mounting the resistors and diodes on the front side.

Next add the battery clip, connectors CON2-CON4 and relay RLY1. RLY1 is in a bit of an odd cylindrical package, with three wires at one end and one at the other. Ensure that its type number is facing up and solder it as shown in Fig.2 and the photos.

On the underside, carefully bend the pins of the DHT22 against its body so they pass through the pads.

Attach the sensor with a 2mm screw and solder the terminals, then prepare the BMP180 for mounting by soldering the supplied 4-pin header to its terminals. Secure the assembly to the PCB and solder the header to the PCB respective pads. Check that "SDA" connects to the square pad.

The single capacitor is an electrolytic type which is fitted bent over on its side. Make sure the longer (positive) lead goes to the square pad, marked

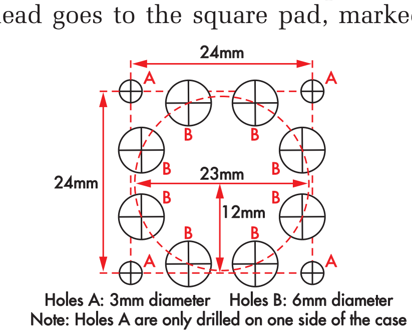
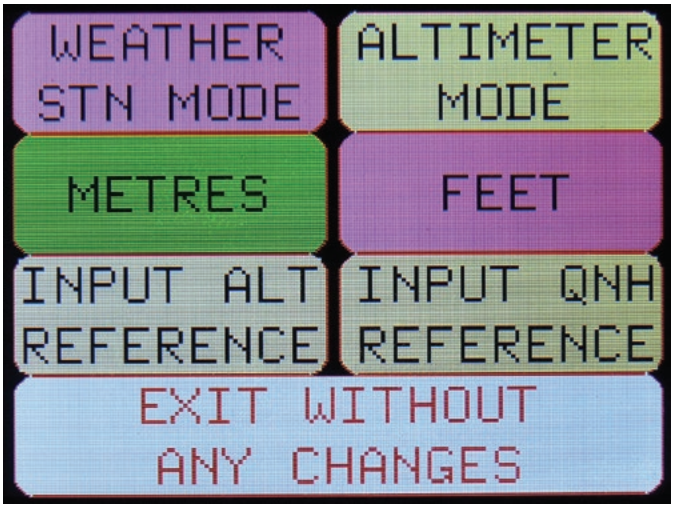


Fig.3: use this diagram as a guide or template to drill the eight airflow holes at both ends of the case, plus the four mounting holes for the fan at the right-hand end.



Screen 3: the settings screen has two buttons at the bottom for calibration; one for entering the currently known QNH value, and one for entering your current altitude in feet.

Screen 4: the current value of QNH is shown as you are typing the new one, to remind you which value you are updating.

+. Secure the body of the capacitor to the board with a dab of silicone adhesive or a piece of double-sided foam mounting tape. Add Mosfet Q2 and BC337 transistor Q1 where shown, and the board is complete.

Case preparation

Next, prepare the UB3 Jiffy box. The cooling fan mounts on the right-hand end, looking at it from the front (lid), as far towards the back as practical.

Drill four 3mm mounting holes, each at the corner of a 24x24mm square (or simply mark the positions using the fan, then drill). You then need to drill some holes inside its footprint to allow air through. I suggest eight 6mm holes arranged equally around a 23mm diameter circle. You can use Fig.3 as a template to mark these holes before drilling.

Drill the same eight air inlet holes on the left-hand end of the case, opposite the fan, but without the fan mounting holes.

Next, locate a convenient point on the back of the box for the USB cable to exit. Drill an 11.5mm diameter hole to take a cord grip clamp. We located it 20mm from the fan end (RH), 10mm from the top. This gives enough length to withdraw the electronics from the box.

Drill a hole in the front panel to mount the dimmer switch, ensuring the switch clears the fan and BackPack connectors.

Cut the cooling fan leads to about 150mm and attach the 3-pin female socket to match CON3 on the interface board. Then make up the 2-pin cable linking CON2 on the interface board to LK1 on the backpack.

To connect to LK1, cut a two-contact section from the leftover remnant of the strip used to make CON4, fold the pins against the body, solder the wires to the pins and heat shrink the wires to the body. This keeps the connector short enough to fit between the BackPack LK1 and the display.

Carefully check the connections. If you swap the wires, diode D1 on the interface board will isolate everything from the 5V input.

The USB cable is a tight fit against the end of the box. We carefully removed some of the plastic reinforcement at the mini connector, and applied gentle heat to persuade the cable to lie in our preferred direction.

The USB mini connector can be inserted through the exit hole in the back of the box and the cable secured with the cord grip clamp. Insert the LIR2450

Parts list – Car/Truck Altimeter

1 assembled Micromite LCD BackPack (V1 or V2)
  [Silicon Chip Cat SC4024 or SC4237]
1 DHT22 temperature/humidity sensor (MOD1)
1 GY-68 BMP-180 temperature/pressure sensor module (MOD2)
1 double-sided PCB, coded 05105201, 86.5 x 49.5mm
1 black or grey UB3 Jiffy box [Jaycar HB6013/HB6023]
1 panel-mount SPST/SPDT toggle switch [eg, Jaycar ST0335]
1 thin 30mm 12V DC cooling fan [Jaycar YX2501]
1 3V DC coil, 250mA SPST reed relay (RLY1) [RS Cat 124-5129]
1 PCB-mount 2450 coin cell holder (BAT1)
  [element14 Cat 1216361]
1 LIR2450 Li-ion rechargeable cell (BAT1)
  [element14 Cat 2009025]
1 2-pin right-angle polarised header and matching plug (CON2)
1 3-pin right-angle polarised header and matching plug (CON3)
1 18-pin header socket (CON4)
1 50cm+ USB cable [eg, Jaycar WC7709]
1 6.2-7.4mm cordgrip clamp [Jaycar HP0718]
4 12mm-long M3 tapped Nylon spacers
4 M3 x 15mm machine screws

Semiconductors
1 BC337 NPN transistor, TO-92 (Q1)
1 ZVNL110ASTZ N-channel Mosfet or similar, TO-92 (Q2)
  [RS Cat 823-1833]
1 3.9V 1W zener diode (ZD1) [eg, 1N4730]
7 1N5819 1A shottky diodes (D1-D6,D8)

Capacitors
1 10µF 16V electrolytic

Resistors (all 1/4W 1% metal film)
1 1MΩ 1 10kΩ 1 2.7kΩ 1 1kΩ 1 36Ω

battery into its clip, mount the interface board on the BackPack with 12mm untapped spacers and 20mm M3 screws. Construction is now complete.

Testing

Load the revised Altimeter software named "*Altimeter with power fail 1_0.bas*" (available for download from the Silicon Chip website) into the Micromite and run it. The first time it is run, the display should initialise with the weather station screen using MSL as the reference.

Connect the Altimeter to a terminal such as Teraterm or MMEdit. The LED on the BackPack should flash twice per second as the Micromite sends the message "pass" to the terminal. If the Altimeter fails to start, check the connection from CON2 to LK1. The cooling fan should run if the software has initialised.

Check that the battery is charging. It should be approaching 3.6V. The voltage drop across the 36Ω resistor should be about 0.9-1.1V when the battery is charged. You can probe this on the reverse side of the board.

Check the touchscreen selections for correct function. To find the QNH to enter, the best method is to use an app such as Weatherzone (see panel). On Weatherzone's current forecast screen for your location is a field labelled "Pressure". If the value is blank, tap the screen to step to the nearest QNH observation.

When you make a change such as entering QNH or Alt reference (current known altitude), you may notice the altitude reading converging on the final value over five seconds.

This is because this software version averages the readings to eliminate short term fluctuations and improve the accuracy of the saved altitude at power down.

With a terminal connected and monitoring the USB signal, the terminal should show "pass" once per second. Disconnect the cable from CON2. The terminal should display the message "saved", indicating that the current altitude has been saved.

Assemble the front panel to the box. You may have to source longer self-tapping screws than those provided, or you can tap the mounting bosses and use machine screws.

The Altimeter should now be ready for use.

Precision, accuracy and errors

Remember that a pressure altimeter is not an instrument of survey accuracy. Even if it can display altitude to a precision of one foot, it is likely to be displaying the wrong altitude very precisely because it is subject to several variables.

One such variable is QNH drift. The Bureau of Meteorology is continually amending QNH, and pilots must continually correct their altimeters. Also, the QNH derived from Weatherzone is truncated to the unit of hPa. Straight away, you have a possible error of ±30ft/10m.

Another error derives from temperature differential. If you park in the sun and turn off the engine, the current altitude will be saved. However, when you return and restart the engine, the car interior temperature could be 20°C higher than ambient. The Altimeter will use this temperature to calculate the new QNH. This error can be up to 6m/20ft for a 20°C difference.

These errors would be unacceptable for night instrument landings, but are not a big deal for either road travel or recreational aviation. Don't stress. Re-enter the QNH and go and enjoy! SC