University of Reading

Make your own Met measurements: build a digital barometer for about £10

Article

Accepted Version

Harrison, G. ORCID: https://orcid.org/0000-0003-0693-347X (2021) Make your own Met measurements: build a digital barometer for about £10. Weather, 76 (2). pp. 45-47. ISSN 0043-1656 doi: 10.1002/wea.3857 Available at https://centaur.reading.ac.uk/92788/

It is advisable to refer to the publisher's version if you intend to cite from the work. See Guidance on citing.

To link to this article DOI: http://dx.doi.org/10.1002/wea.3857

Publisher: Wiley

All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the End User Agreement.

www.reading.ac.uk/centaur

CentAUR

Central Archive at the University of Reading

University of Reading

Reading's research outputs online

Make your own Met measurements: build a digital barometer for about £10

R. Giles Harrison

Department of Meteorology, University of Reading

Abstract

Meteorological measurements need not always require expensive instrumentation. Modern electronic sensors are accurate and readily made to work with well-supported programmable electronic interfacing systems, such as the Arduino series of devices. Construction of an atmospheric pressure display is outlined here, which also provides temperature and relative humidity measurements.

1. Introduction

Major developments in miniature and inexpensive sensors have occurred in the past decade or so, often motivated by applications in mobile phones. Some of these devices are sufficiently accurate and stable that they are suitable for use in meteorology. For example, extending the range of sensors carried on conventional meteorological radiosondes has been a recent focus of work in the Meteorology Department at Reading, for measuring turbulence (Marlton et al, 2015), volcanic ash or liquid water (Airey et al, 2017). Further, as part of Physics of the Environment BSc degree, Reading students are taught how to use modern electronic sensors, and make their own data logging system for Pressure (*P*), Temperature (*T*) and Relative Humidity (*U*) which they evaluate against standard measurement methods (Harrison and Westbrook, 2016). As an encouragement to look at these new and inexpensive technologies, a brief description is given here of how to use an integrated digital PTU sensor, without requiring any soldering skills. Indoors, it can operate as a simple digital barometer.

2. Principles

Measurements made by electronic sensors provide outputs which are analogue (i.e. continuously varying voltage or currents), or digital (interrogated as a data stream by a computer in some way), which need to be captured for recording or display. Because of the many possible ways of connecting a sensor, programmable devices have been developed which can be configured to read a variety of sensors and provide the data in a form suitable for further processing. A *microcontroller* is an example of such a device; microcontrollers are often provided on a small circuit board - called a development board – providing the different connections needed. There are many development boards available but the Arduino series of such boards (or *shields*) has proved popular as it is intended for beginners, supported by a readily downloaded programming environment and many items of open source software altruistically contributed. To demonstrate a meteorological application for an Arduino system in a simple way, its use with an integrated sensor, the Bosch BME280, is described. This sensor is the latest in a series of sensors from Bosch which can resolve atmospheric pressure changes to 0.01 hPa, as, unlike previous generations of pressure sensors (e.g. Harrison, 2000), it implements accurate internal polynomial-derived individual temperature

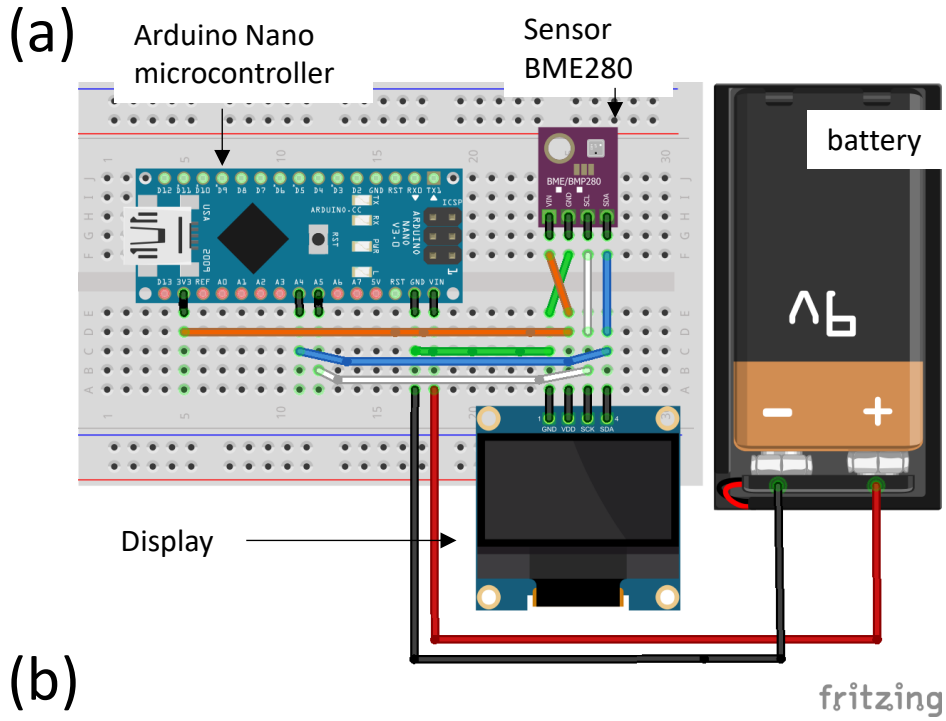
compensation (Bosch, 2020). The BME280 also provides measurements of temperature and relative humidity.

3. Implementation

The BME280 PTU sensor itself is tiny, 2.5 x 2.5 x 0.93mm, so it is normally obtained on a "breakout board", to provide easier access to its different terminals. Beyond some electrical connections, all that is needed to use it is a device running a program to control the data transfer (typically a cheap microcontroller), and a source of power. As a computer will be needed to program the microcontroller, this could also be used to record and display the data, but a separate display of some kind may well be preferable and allows a stand-alone pressure readout to be constructed.

(a) Hardware

Figure 1 shows a basic arrangement with a BME280 sensor, an Adafruit liquid crystal display (LCD) and an Arduino Nano, one of the more compact devices in the Arduino series which still provides many connections for both analogue and digital sensors, and 32kB of program space. Figure 1(a) shows the wiring needed. Data connections to the BME280 and LCD are paralleled, using only two pins on the Nano. This approach which works through a protocol (called I^2C, for inter-integrated circuit) which selects the device required through an identifying code. Two further connections supply power, from the Nano. Figure 1(b) shows the system implemented on a solderless "breadboard", widely used for electronic prototyping. As supplied, both the Nano and the BME280 breakouts come with a strip of pins intended for soldering. It is, however, possible to avoid soldering if the one strip of pins is placed into the Nano's connection holes on only one side and the second set of pins is not connected, but used to wedge the Nano board tightly to ensure the connections on the opposite side provide a good contact. Similarly, for the BME280 breakout, plasticine or Blu-tack can be used to keep the unsoldered pins under gentle tension. The suggested LCD is already provided with ready-soldered pins. A 9 V battery powers the system, and the usual 9 V battery connector needed has wires attached. Single core wire is suitable for the other connections.



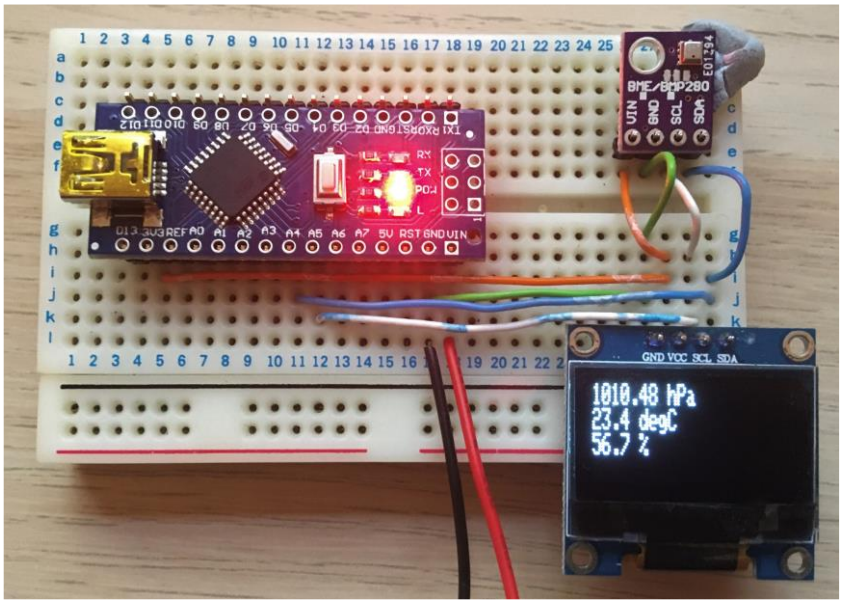


Figure 1. (a) Layout diagram showing wiring between an Arduino Nano, a Bosch BME280 on a breakout board, and a 128x64 pixel I^2C LCD display, all powered by 9V battery. (b) Physical implementation showing the parts mounted on a breadboard, and the system operating. (The 9V battery is not shown).

(b) Software

To retrieve sensor data, format the values and write the measurements to the display, the Arduino Nano needs to have a program loaded, to which a link is provided at the end of this article. Programs for Arduino devices (known as *sketches*) are readily generated and compiled in the Arduino

Integrated Development Environment (IDE), installed on a desktop or laptop, which can be obtained for a small donation. The IDE can also load software libraries to support specific devices, which simplifies the structure of the code. Arduino IDE sketches are written in C++, and, when complete, are transferred from the IDE to the Arduino though a standard USB connection. This connection can also allow data to be transferred back to the computer used for programming. For the pressure display outlined, a small sketch has been written using existing libraries for the BME280 and LCD, with integer arithmetic and string handling employed to avoid the extra code size associated with floating point calculations.

4. Evaluation

The combination of the BME280, LCD, Nano and the simple monitoring sketch allows display of data from the sensor updated every second, which can also be seen in the serial monitor window of the IDE. Although the intention here is just to illustrate a simple arrangement, further refinements can clearly be considered, such as adding a SD card to allow data storage for subsequent transfer to a computer, a Real Time Clock to timestamp the values obtained, or the "Internet of Things" variant of the Nano (Nano 33 IOT) to provide wireless communications. GPS modules for time and position can also be easily included. Such additions allow measurements to be made in a wide range of situations, and figure 2 shows examples of pressure measurements made (a) during a balloon ascent and (b) whilst passing through the tunnels leaving King's Cross station in an express train, both using the earlier generation BMP180 Bosch sensor. The sensor clearly has a wide operating range despite only being specified down to 300 hPa, and a remarkably rapid transient response of more than 10 hPa per second.

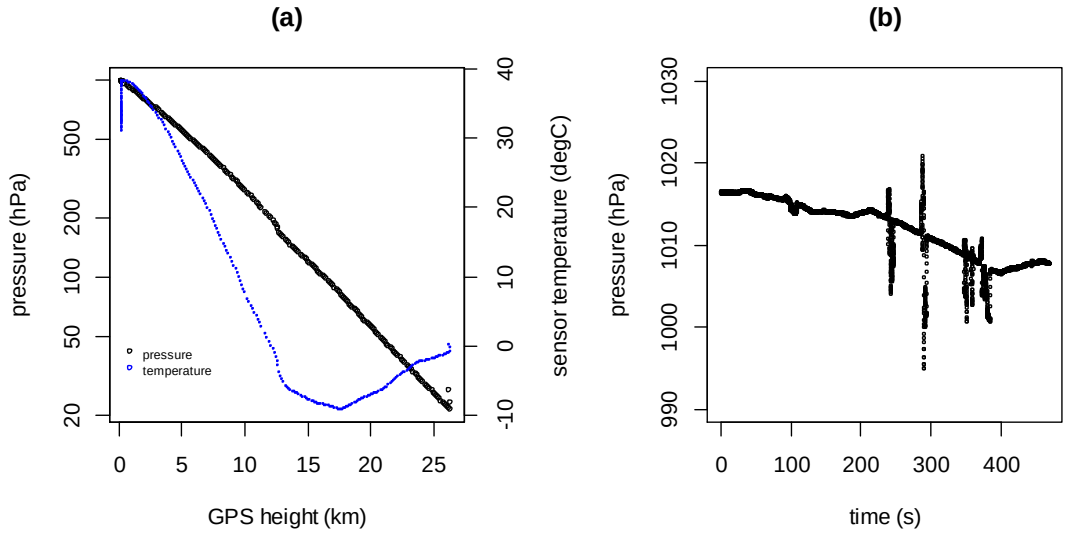


Figure 2. Pressure measurements obtained using a Bosch BMP180 pressure sensor (a) from a balloon package (also carrying a GPS receiver) launched from Zaragoza, Spain in September 2014 (Makhmutov et al, 2015) and (b) passing through railway tunnels after leaving London King's Cross station.

Further construction notes

The parts and software needed are readily obtained from the internet. Depending on the suppliers and delivery terms chosen, the principal items needed of the Arduino, LCD and BME280 can be obtained for about £10. The Arduino system and IDE is well supported[1], including help with installing libraries[2]. A simple test of the IDE connection and the Arduino Nano is provided by downloading the Blink program supplied as an example with the IDE, which, when functioning correctly, will cause one of the integral LEDs on the Nano to flash. Note that the unsoldered connections suggested are not very robust, and strange values may result with poor connections: this can be remedied by adjusting the connectors to press the Nano or BME280 pins ever more firmly, then pushing the Nano's reset button to allow the initialisation to run again, until more sensible values are shown. To test that both devices are properly connected, an I^2C scanner program can be downloaded to the Nano: the LCD should report 0X3C and the BME280 sensor 0X76.

Parts: plugboard; BME280 sensor on GY-BME280 breakout board; Arduino Nano; 0.96 inch 128x64 pixel I^2C SSD1306 LCD display (e.g. AZOLED12864-1000); 9V battery and battery clip; colour-coded single core wires can be salvaged from a short piece of telephone cable

Software: Arduino IDE; TinyBME280 library; Adafruit SSD1306 library; *PTUWeather* logging code (downloaded from https://tinyurl.com/t859pms); USB to mini-B programming lead. If a clone Arduino Nano is used, additional USB drivers may be required.

Disclaimer

I have no association with Arduino or Bosch, and other arrangements of different sensors and development boards to achieve similar results are of course also possible. The experiments suggested here are undertaken at your own risk. Detailed correspondence cannot be entered into individually, but further ideas and frequently asked questions will be addressed at https://tinyurl.com/yxch7lz5 .

References

M.W. Airey, R.G. Harrison, K.A. Nicoll, P.D. Williams, and G.J. Marlton, 2017. A miniature oscillating microbalance for sampling ice and volcanic ash from a small airborne platform *Rev Sci Instrum* 88, 086108 10.1063/1.4998971

Bosch, 2020. https://www.bosch-sensortec.com/products/environmental-sensors/humidity-sensors-bme280/

R.G. Harrison, 2000. A temperature-compensated meteorological barometer *Rev Sci Instrum* **71**, 4, 1909-1910

R.G. Harrison and C.D. Westbrook, 2016. Microcontrollers for data logging in Environmental Physics, *Geophysical Research Abstracts*, 18, EGU2016-8495-1

[1] https://www.arduino.cc/en/main/software
[2] https://www.arduino.cc/en/guide/libraries

V. Makhmutov, G. Bazilevskaya, Y. Stozhkov, M. Philippov, Y. Yair, R. Yaniv, G. Harrison, K. Nicoll, K. Aplin, 2015. Cosmic ray measurements in the atmosphere at several latitudes in October 2014, *Proc 34th International Cosmic Ray Conference*, The Hague, The Netherlands

G.J. Marlton, R.G. Harrison, K.A. Nicoll and P.D. Williams, 2015. A balloon-borne accelerometer technique for measuring atmospheric turbulence *Rev Sci Instrum* **86**, 016109, http://dx.doi.org/10.1063/1.4905529