4 Alphasense Air Pollutant Sensors for COSMOS and ESCI 122

David Durkin
Peter Weiss

June 26th, 2019 to September 3rd, 2019

Abstract

Four Alphasense Toxic Air sensors that measure the concentration of carbon monoxide (CO), nitric oxide (NO), ozone (OX), and nitrogen dioxide (NO_2) were compiled onto one circuit board interfaced with an RTC, microSD card, and Arduino MEGA for discrete data collection and storage. The sensors attached to a container for gas sampling can measure air pollutants in ppb with varying ranges and resolutions for each sensor.

Introduction

The appending of sensors to one apparatus was set forth by Peter Weiss for initial instructional use for the UCSC COSMOS Program in the Summer of 2019, and for Earth Science 122: Air Pollution, to be taught by Professor Weiss in the Fall of 2019. Each Alphasense 4-Electrode Individual Sensor Board (ISB) measures toxic air pollutants with a resolution of about 15ppb, and up to about 10ppm, depending on the sensor. Without a commercially available board to interface to all four sensors, one was built by David Durkin, using components for noise reduction, time keeping, and data storage. Initial use of the board will be for lab experiments, but further installation in high pollutant areas can be used for monitoring air quality for academic or non-academic purposes.

Having a brief understanding of how the Alphasense air monitors operate gives a foundation for understanding the set up of the board and code. Schematics, code, and pictures can be found at the end of this document, with all files on a GitHub repository linked in the appendix.

Alphasense Air Monitors

The Alphasense toxic gas sensor usd electrochemical cells to detect amperometric changes, that is changes to current, running through the sensor. The change in current is linearly proportional to a volumetric change in the gas present. There are three electrodes in the sensor, two of notable importance; the working electrode (WE), reference or auxiliary electrode (AE), and counter electrode. For CO, NO, and OX, oxidation occurs on the surface of the sensor, whereas a reduction occurs with NO_2. The counter electrode tries to balance the reaction of the WE, and will produce other compounds like H_2O or CO_2 in a reaction that compliments the WE's reaction. The counter electrode is not interfaced to the sensors outputs, and its voltage/current is not of notable concern (AAN 104).

The auxiliary electrode (AE), however, is important. The circuitry and electrochemical cell for the AE are identical to the WE, but the AE is closed off from air to be used as a reference and control to the WE. The voltage put out from the AE to the sensor is subtracted from the WE voltage, as seen below in the equations for solving for concentration.

The sensors are susceptible to electromagnetic fields, and design consideration from Alphasense suggest using shielding around the sensors themselves for more stable result. This was not implemented since the effect would be minute and no major emitters of magnetic waves are present in most experiments that the sensors will be used for (AAN103). The sensors do however contain liquids within the device for the chemical reaction, and are sensitive to movement. Avoid bumping or moving the device to ensure stable results. When exposing the sensors to gas, ensure air flow, if any, is not directly normal or perpendicular to the screen on the sensor, but runs parallel over the mesh (AAN 010). Further information about extreme environmental conditions, interfering gasses, and additional documentation can be found under the Toxic section of the application notes available online (http://www.alphasense.com/index.php/safety/application-notes/).

Interfacing the Sensors

The sensors each have 6 pins, 2 for power (+/-), 2 for the working electrode (+/-), and 2 for the auxiliary electrode (+/-), with a shared internal ground. The working electrode outputs a value to the analog pin of the Arduino which reads between 0-1023, as does the auxiliary electrode (AE). This value is then converted to mV based on the reference voltage of the Arduino, which is a standard 5V. Upon measuring, the Arduino was supplying 5.074V to the sensors, which then becomes the reference used in converting the value from the WE and AE to mV (line 35 of code). Multiplying the analog value of WE and AE by 5.074V/1023 gives us the mV value of each.

With the four sensor came a calibration sheet with specific value pertaining to each sensor. Alphasense measures each sensor for specific zero offset of the WE and AE. The calibration and sensitivity values are specific to the sensor we are using. Swapping one sensor for a new one will require adjustment to the code (line 16-33). The sensitivity and electronic zero are listed below.

Sensor Type	WE Electronic zero	AE Electronic zero	Sensitivity
Carbon Monoxide Sensor (CO)	355mV	351mV	.401(mV/ppb)
Nitrogen Dioxide Sensor (NO_2)	290mV	288mV	.471(mV/ppb)
Ozone + NO2 Sensor (OX)	227mV	219mV	.277(mV/ppb)
Nitric Oxide Sensor (NO)	229mV	227mV	.354(mV/ppb)

The full sheet of specific values is located in the Appendix. With the output value from the analog pin, we subtract the WE electronic zero and the AE electronic zero from the WE and AE values. These are then the corrected values, and finding the difference between those, we get the actual change in voltage (delta V), from which we multiply by the reciprocal of sensitivity to get our value in ppb. Below is an example to the equation and serial monitor output.

$$Concentration\ (ppb)\ =\ [(WE(mV) - WE_o(mV)) - (AE(mV) - AE_o(mV))] \div [Sensitivity(\tfrac{mV}{ppb})]$$

```
CO WE Value:     125       CO AE Value: 67
Voltage WE(mV):  255.95    Voltage AE(mV): -23.53
Delta V(mV):     279.48    CO Concentration: 696ppb
```

The equation can be simplified to resemble a linear equation(y = mx +b) as follows

$$Concentration\ (ppb)\ =\ [WE(mV) - AE(mV)] \div Sensitivity(\tfrac{mV}{ppb}) + [AE_o(mV) - WE_o(mV)] \div Sensitivity(\tfrac{mV}{ppb})$$

Thus the difference between WE and AE is the only dependent variable. Calculations to solve for concentrations can be found on lines 122-185. Each equation would be as follows,

$$CO(ppb)\ =\ ([WE - AE] \div .401(mV/ppb))\ + 44.88$$

$$NO_2(ppb)\ = ([WE - AE] \div .227(mV/ppb))\ + 8.81$$

$$OX\ =\ ([WE - AE] \div .354(mV/ppb))\ +\ 2.824$$

$$NO(ppb)\ =\ ([WE - AE] \div .471(mV/ppb))\ - 38.21$$

The sensors are marketed as having high resolution for precise ppb measurements. However, with only 1023 discrete points and a range of roughly 10ppm, the change in concentration can be a bit course. The values listed below correspond to a change in one analog point to the relative change in concentration,

Δ Value	Δ ppb
CO	12ppb
NO_2	12ppb
OX	14ppb
NO	10ppb

Each sensor must warm up before reporting accurate values. Upon initial purchase of a new sensor or after a long period of removal, the warm up time is roughly 2 hours for CO, NO_2, and OX, and 12 hours for NO. After first use, warm up time significantly drops to 10 min theoretically, and 20-30 minutes experimentally for CO, NO_2, and OX. According to the data sheet, NO_2 still requires 12 hours of warm up even after repeated use.

The baseline for these values is also not always accurate. While NO and NO_2 should be zero in the absence of a combustion or pollutant sources, initial reading showed values of 20 and 50 ppb respectively. This was resolved by manually calibrating those values to be zero after exposing the sensor to ambient conditions. For OX however, values are typically 10ppb indoors, and 40-50ppb outdoors depending on conditions. Upon sampling air from both environments, values were adjusted to read ~16ppb indoors and ~40ppb outdoors, yet the issue of stability and resolution came up again.

PCB

To stabilize values, decoupling capacitors were put near the output of the WE and AE pins on the circuit board. Alphasense suggests using these bypass capacitors of 10nF and 100nF to reduce noise, calming any hiccups or ripples in the current. The DS1307 RTC was used for time keeping, along with a CR1220 button cell battery. The battery allows for time to be kept even when the board is not powered. A microSD card reader from Adafruit is used to store data on a 16Gb card. All components are soldered to a custom PCB that sits atop the Arduino MEGA. The interface is powered by the Arduino that uses a USB channel included with most microcontrollers. The sensors are attached with jumper cables, and each component is labelled on the silk screen. Schematics, OrCAD images, and pictures can be found in the Appendix, as well as a link to the github that contains the gerber files.

Code

The code, written in C on an Arduino file, uses a few libraries for the RTC and SPI for the SD card. All initial values for the sensors, including their sensitivity and zeros, are declared in the beginning for easy customization if components are switched out (16-33). Other changeable values include the input voltage(35), timing between intervals(36), and sampling set (46) . Apart from that, little else should be changed when uploading code to a new board or reviewing how the code operates.

The main body of the code uses case states to ensure timing errors do not occur when tasks are performed, such as sampling, displaying data to the serial monitor, and writing to the SD card. The serial monitor is useful in reading the data as it comes in, and can be accessed through the Arduino IDE, and opened by either clicking the icon on the top right of the window or the shorthand command of Ctrl+Shift+M. The monitor will display whether the SD card is initialized, followed by the date, trial number, and headers for the different gases. Following this, data will be taken every second and display to the monitor, as well as written to the SD card.

For time time interval, the set time is every 1 second (36). Within this second, 75 data points are collected from each sensor, and the mode, or most recurring number is chosen to ensure accuracy (231-246). The number of data points is determined by a variable “n” that can be changed(46). Statistically, no less than 50 data points should be taken to achieve accurate values for the mode function. The set of 75 data points is collected in an interval of 10ms. Depending on the specified number of data points, the time interval between displaying to the monitor will be the same (208). Runtime is not accurately accounted for in the timing of the code, so data points are not exactly 1 second apart. Note the timestamp next to each trial to ensure when graphing that the information is conveyed correctly.

The data is written to the SD card in a different format than what is displayed, so that the columns of data can be transferred over to an excel sheet. Data to SD card is delimited by a comma. As follows, it is "Trial, ppb CO, ppb NO2, ppb OX, ppb NO, date (m/d/y), time (h:m:s)", exporting instructions can be found here (https://spreadsheeto.com/text-to-columns).

The "write" state uses the standard example "ReadWrite" code from Arduino to store data to the SD card. The serial monitor will display "Saved" if data is saved, and an error message if not. The error message will also occur if the microSD card is not inserted. Do not remove the microSD card from the board unless the board is not sampling or is powered off, this can corrupt or destroy previous data written to the card.

Conclusion

The board constructed is useful in consolidating multiple air pollutant sensors to one apparatus for academic use, and furthermore for monitoring air quality in a multitude of applications. The board simplifies the project that would otherwise be a rats nest of cables branching out from a breadboard, while also offering accurate time keeping and data storage functionality. The sleek simplistic design takes away a distracting cluster of chords to allow the user to track and chart toxic gases in the air, allowing students to perform labs and experiments free from the fear of unplugging modules and shorting wires.

Appendix

Github links

https://github.com/daviddurkins/4AirPollutantSensors

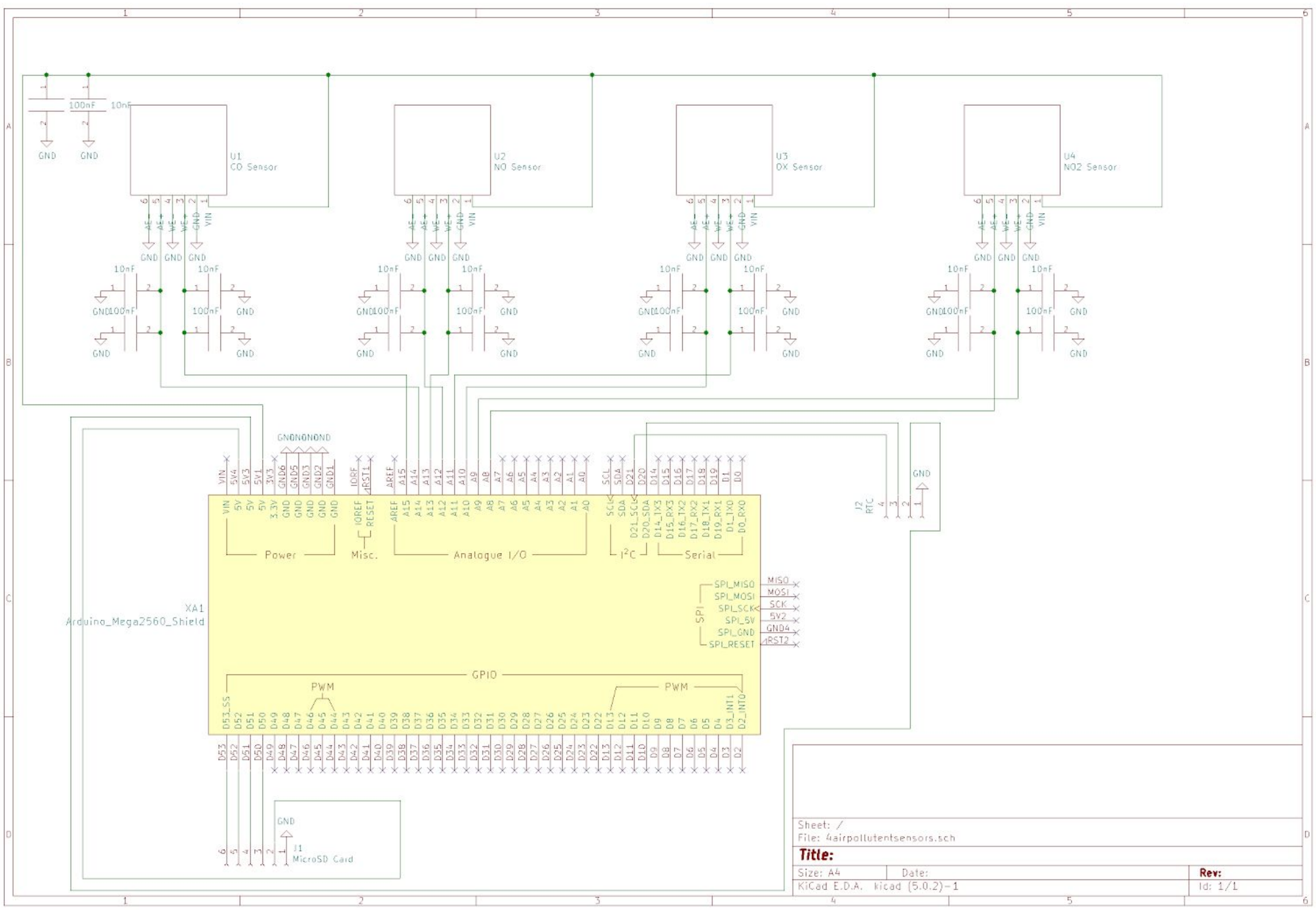
Link to application note for the Alphasense ISB

http://www.alphasense.com/index.php/safety/application-notes/

Data Sheets for the A4 series of air pollutant monitors (CO-B4, NO_2-B4, OX-B4, NO-B4)

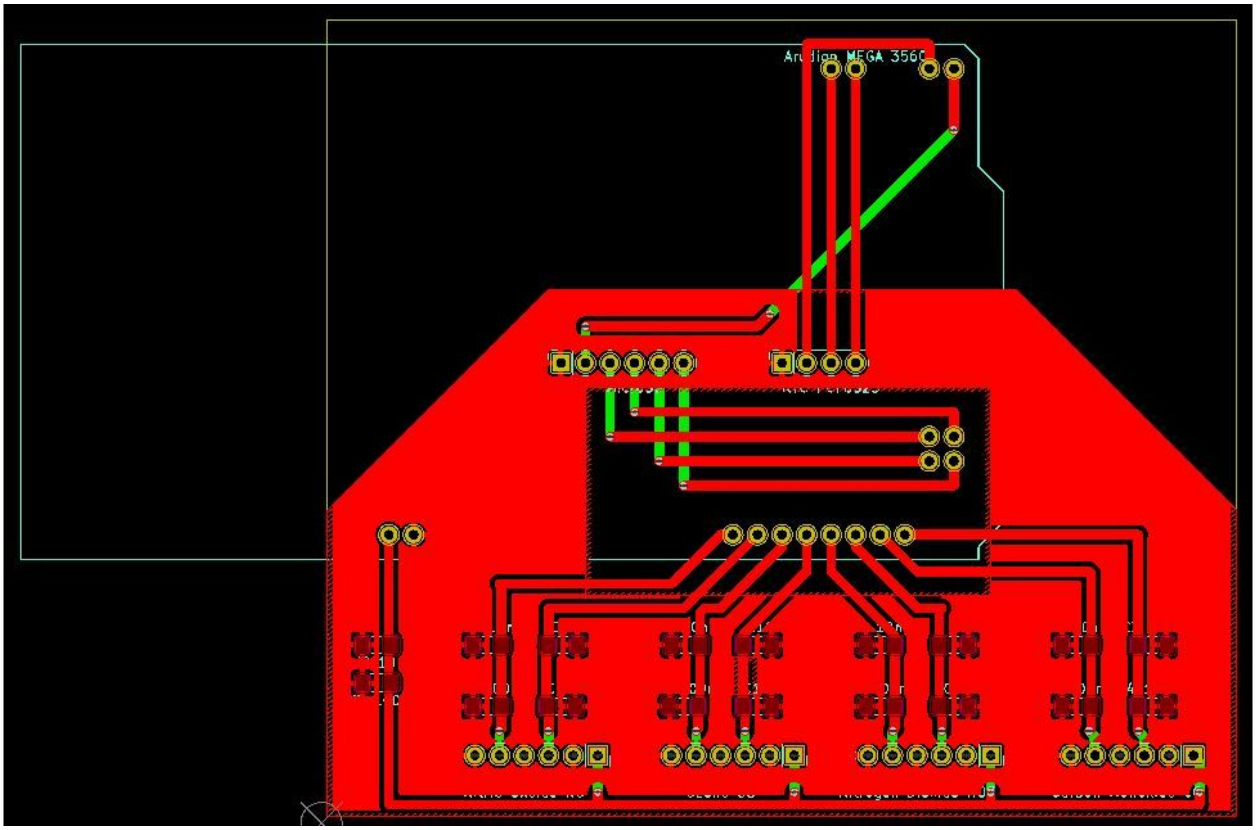
http://www.alphasense.com/index.php/safety/downloads/

Wiring Schematic*

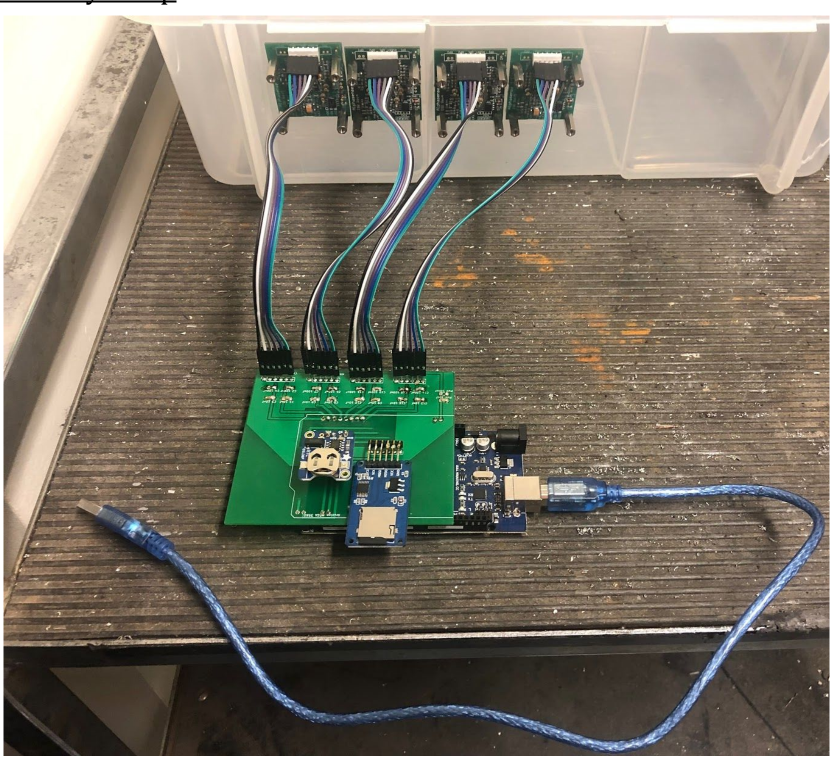


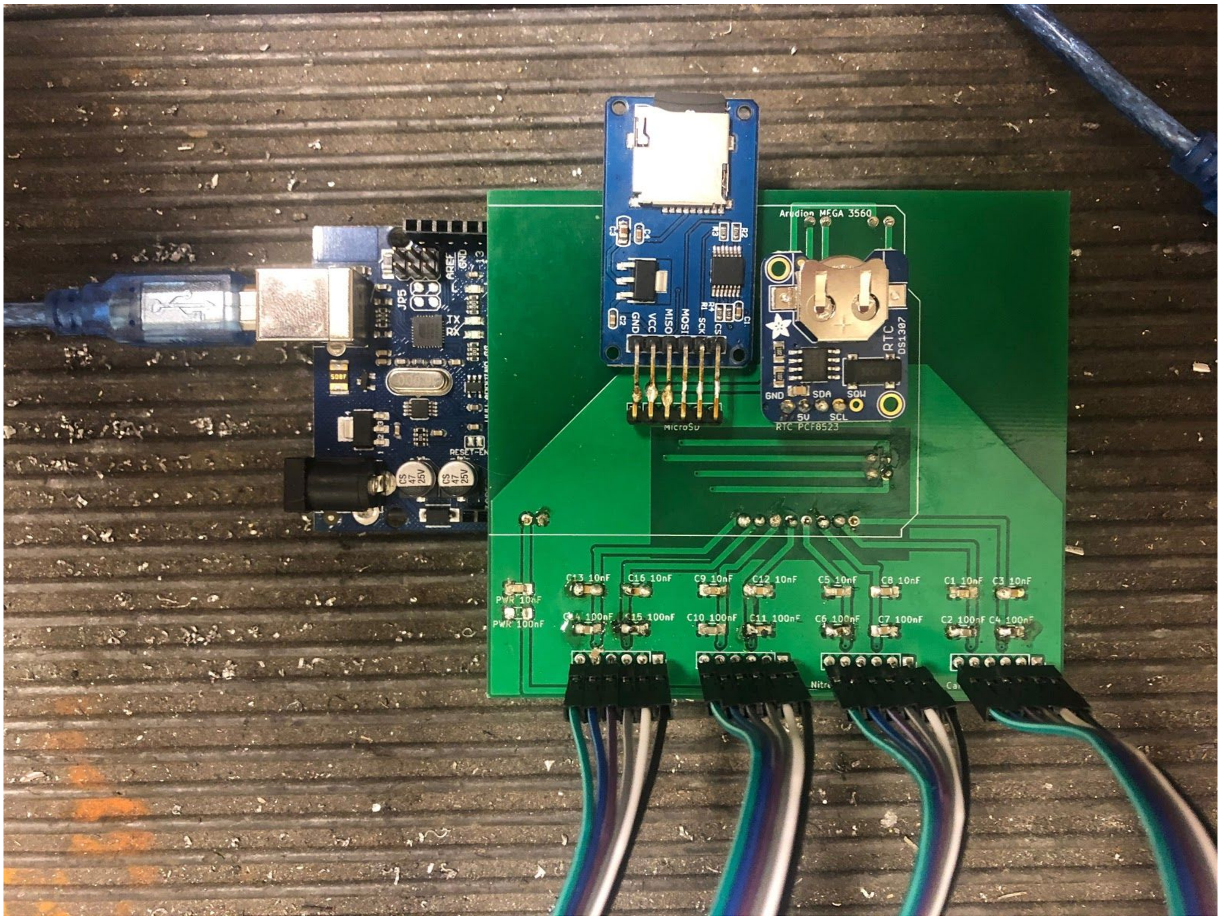
*Higher quality images of the schematic can be found on the github

OrCAD image



Actually setup





Calibration sheet given by Alphasense

CUSTOMER	Sensor Type	Gain (mV/nA)	SENSOR NUMBER	WE Zero (mV)	Aux Zero (mV)	WE Sensor (nA/ppm)	Sensitivity (mV/ppb)	ELECTRONIC ZERO (WE) (mV)	ELECTRONIC ZERO (AUX) (mV
Uni of California - Santa Cruz	CO B4	0.8	162482654	324	342	501	0.401	355	351
Uni of California - Santa Cruz	NO B4	0.8	160660809	345	328	589	0.471	290	288
Uni of California - Santa Cruz	NO2 B43F	-0.73	202481508	190	192	-380	0.277	227	219
Uni of California - Santa Cruz	OX B431	-0.73	204660047	221	222	-485	0.354	229	227

Code

https://github.com/daviddurkins/4AirPollutantSensors