Title: VentMon: and Open Source Inline Ventilator Test Fixture and Monitor

Authors: Robert L. Read, Lauria Clarke, Geoff Mulligan

Affiliations: Public Invention with support from the *Protocol Labs* and the *Mozilla Foundation*

Contact email: read.robert@gmail.com

Abstract:

Humanitarian engineers responded to the pandemic ventilator shortage of March, 2020, by beginning over 100 open source ventilator projects. Inexpensive ventilator test equipment facilitates projects forced to be geographically distributed by lockdowns. The VentMon is a modular, open source, IoT-enabled tester that plugs into a standard 22mm airway between a ventilator and a physical test lung to test any ventilator by graphing flow, pressure, oxygen fraction, humidity and temperature at a data lake accessible to all team members. This is the data needed by a test team, but also the data desired by a clinician in actual treatment of patients. The open source design of the VentMon and its firmware and cloud-based software may allow it to be used as a component of modular ventilators to provide a clinical readout. This software system has been designed to be as modular and composeable as possible, for example by defining new open published standards, just as the hardware is composeable and modifiable, forming an open system or eco-system of ventilation data. Thanks to grants, 20 VentMons have been given away free of charge to pandemic response teams building open source ventilators, some of which have been used successfully.

Keywords: COVID-19, open source medical device, ventilator, pandemic response

Specifications table:

Hardware name	VentMon
Subject area	Educational Tools and Open Source Alternatives to Existing Infrastructure
Hardware type	Field measurements and sensors, Electrical engineering and computer science
Open source license	MIT License, CERN-OHL-S
Cost of hardware	$280
Source file repository	VentMon: https://doi.org/10.5281/zenodo.4079170 PIRDS (Public Invention Respiration Data Standard): http://doi.org/10.5281/zenodo.4079377 PIRDS logger: Zenodo. http://doi.org/10.5281/zenodo.4079382 SFM3X00: http://doi.org/10.5281/zenodo.4079355 VentDisplay: http://doi.org/10.5281/zenodo.4079374 *DOI URL to an approved source file repository: Mendeley Data, the OSF, or Zenodo (instructions). For example: https://doi.org/10.5281/zenodo.3346799* *If there is no external repository write " Available in the article "*

1. Hardware in context

The Coronavirus pandemic has created a global shortage of ventilators.Ventilators are expensive to manufacture and during uncertain times supply chains can be disrupted increasing the cost and scarcity of these devices. Since the beginning of the pandemic in March, there has been a large movement to develop cheaper, more supply chain resilient ventilator. The urgency and goals of this movement have shifted as the global understanding of the virus has evolved. Regardless of the current state of this movement, however, the use of open source medical devices in resources limited emergency situations – common during a global pandemic – is a topic that has come to the forefront of may conversations about preparedness and treatment. Medical devices require rigorous testing before they can be used on the general public. Open source medical devices require the same level, if not more scrutiny. If an open source movement is to succeed in this field, the tools to test and validate medical devices should also be community based. The goal of VentMon is to provide an equally open source, community based verification solution to increase the efficiency and accuracy of the development process for socially distanced, potentially international teams making open source ventilator devices.

2. Hardware description

The VentMon plugs directly into a standard 22mm adult airway. As air or medicaly gases pass through the VentMon, it records flow and pressure about 40 times a second, and occasionally records temperature, humdity, and O2 fraction. The result is that when placed in the airway between an invasive or non-invasive ventilator under test and physcial test lung, it sends this data to a public data lake using a clearly defined respiration standard. This data can be graphically rendered either live or statically to provide a display that is typical of the clinical display of advance ventilators. In this way it allows sophisticated testing of ventilators. This device is therefore similar in function to an ASL5000 or a Michigan Test Lung device, which tend to start at USD$10,000.

A special challenge of teams building open source pandemic ventilators is that they tend to be poorly funded, but also geographically distributed. The VentMon is an IoT-enbabled device which sends its data to a public data lake, where a team member on a different continent can view it live in real time as a collleagure makes adjustments to the ventilator.

The VentMon thus offer benefits to teams developing ventilation devices:

- It is inexpensive.
- It is completely open and part of a modular hardware/software system that can be customized and extended.
- It is allows remote collaboration.

Moreover, being completely open, it supports modifcation, extension, and importantly but only potentially at present, integration into a complete ventilator. A ventilator that chose to offer a graphical trace of pressure and flow, which are generally demanded by clinicians for all but emergency transport situations, could incorporate the open source VentMon into its complete design.

Although the reliance on the internet is a strength for a distributed engineering team, it would be a disadvantage it a field hospital. Although aiming to eventually be safe enough and reliable to be used as a monitor on human patients, it requires much improvement before it is ready for that intended use.

3. Design files

TODO Need to make hyperlinks colored in *Location of File* column

3.1 Design Files Summary

Design file-name	**File type**	**Open source license**	**Location of the file**
Embedded Firmware	C++ Source Code	MIT License	VentMon Firmware
PIRDS Data Viewer	HTML Source Code	MIT License	Vent Display
PIRDS Data Logger	C Source Files	MIT License	PIRDS Logger
VentMon T0.3 PCB	PCB Design Files	MIT License	VentMon PCB
Pressure Sensor and Airway Adaptor	STL File	MIT License	VentMon Adaptors
PIRDS Data Standard			

For each design file listed above, include a short description of the file here (one or two sentences)

4. Bill of materials

Current Link: VentMon HardwareX BOM

- BOM needs to be converted to PDF and uploaded per instructions:
  For a complex Bill of Materials, the complete Bill of Materials (editable spreadsheet file e.g., ODS file type or PDF file) can be uploaded in an open access online location such as the Open Science Frameworks repository. Include the link here. Alternatively, the Bill of Materials can be uploaded at the time of submission on the online Elsevier submission interface as supplementary material.

5. Build instructions

Two versions of VentMon can be assembled depending on availability of parts and time. The most physically robust and complete version of VentMon requires the purchase and manufacture of a PCB as well as a number of 3D printed plastic parts. The designs of the PCB and 3D printed parts are fully open; in practice, due to the fact that PCBs have a high set up cost, it makes most sense to use the PCB if you are bulding 10 or more VentMons. This version of the device requires the least amount of time to assemble and contains fewest discrete components. Due to the high cost of PCB manufacture and assembly, VentMon can also be created using off the shelf components readily available from DIY electronics suppliers. This version requires significantly more assembly time. We believe this has been done twice by third parties with little interaction with us. Both assembly procedures are outlined below.

5.1 PCB Based VentMon

This version of VentMon requires two 3D printed parts – one encapsulated an on-board pressure sensor and one is an airway adaptor – as well as a PCB assembly. Before beginning the assembly process make sure that you have manufactured those three parts.

1. PCB Assembly

   1.1 Add standoffs to PCB

      1.1.1 take hardware and put it in the holes

   1.2 Mount sensor enclosure for BME280 pressure sensor

      1.2.1 Insert plastic part into mounting holes on PCB to check fit and alignment. The barb should face toward the outer edge of the PCB.

      1.2.2 Apply glue to bottom edge and mounting pegs of plastic.

      1.2.3 Insert plastic back into holes being careful not to get any glue on the sensor.

      1.2.4 Allow 24 hours for glue to cure before attaching a hose to the barb.

2. Enclosure

3. Flow Sensor Assembly

4. Oxygen Sensor Assembly

5. Final Assembly

5.2 COTS Based VentMon

1. Qwiic Shield Assembly

   See Figure 2 on page 6 for final assembly of Qwiic Shield.

   1.1 Trim two pieces of male header pins that are two positions long and one piece that is four positions long.

   1.2 With the connector side of the board facing up, solder one of the two position headers into the two holes labelled SDA and SCL.

   1.3 Solder the other two position header into the last two holes on the row along the opposite side.

   1.4 Solder the four position header into the last four holes along the right side. This should provide connections to the GND and 3V3 through holes.

   1.5 Connect both sides of the solder bridge to engage the pullup resistors for the I^2C bus.

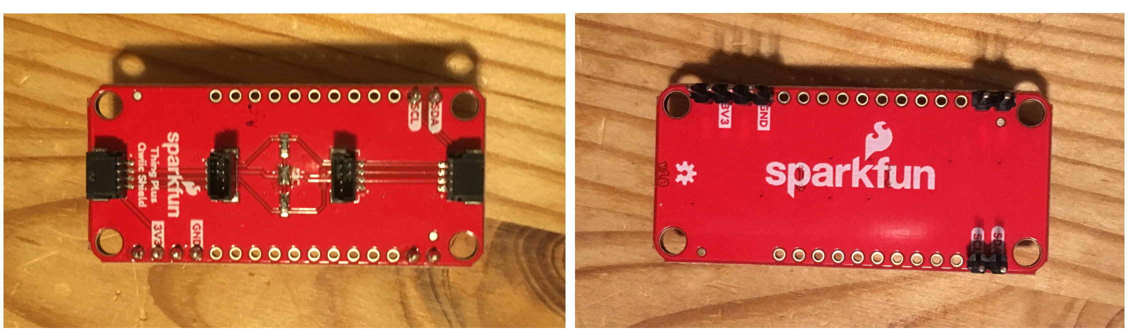

Figure 1: SparkFun Qwiic Shield with male headers and I^2C pullups connected

2. Enclosure

   Before starting this step download and print the drill template. See Figure **??** on page **??** for finished hole pattern.

   2.1 Cut template and tape it to the enclosure.

   2.2

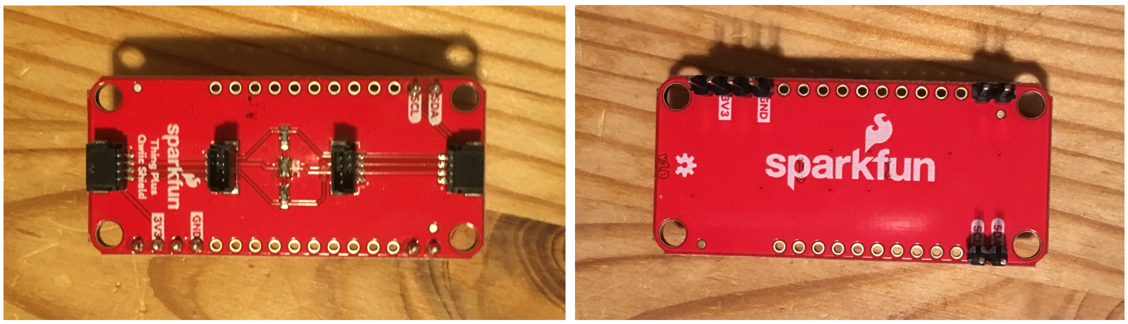

Figure 2: SparkFun Qwiic Shield with male headers and I^2C pullups connected

3. Pressure Sensor Assembly

   3.1 Drill out mounting holes

      3.1.1 Using the 7/64? drill bit, carefully enlarge the mounting holes in each corner of the sensor

      3.1.2 You can do all four or just two - one on the left side of one board and one on the right side of the other

   3.2 Cut trace from SDO through hole to sensor

      3.2.1 Carefully cut the trace coming from the SDO hole on the back side of one board

      3.2.2 This board will be used to measure the airway pressure

   3.3 Assemble Cable

      3.3.1 Cut one connector off the Qwiic cable

      3.3.2 Strip and tin the last ?? of the leads

   3.4 Connect Sensors to Cable

      3.4.1 Stack the two sensors on the breadboard and use two paper clips to anchor them

      3.4.2 Be sure the airway board with the cut trace is facing down

      3.4.3 Starting with Vin or GND insert the tinned lead through the appropriate hole and apply solder

      3.4.4 Flip the sensor assembly over and make sure solder has flowed through both through holes, adding some if needed

      3.4.5 Don?t worry if the solder does not flow perfectly

      3.4.6 Pin the assembly back down and move to another connection following the connection table below

      3.4.7 Repeat the steps above until the two boards have been secured and all leads are connected

3.4.8 Re-flow all the joints to ensure a good connection

Signal	Qwiic Color
GND	black
VIN	red
SDA	blue
SCL	yellow

Table 1: pressure sensor wiring

3.5 Add address jumper to ambient pressure sensor

3.5.1 Take a ?? piece of hookup wire and strip both ends

3.5.2 Flow the solder in the GND through hole and poke one end of the hook up wire into the hole

3.5.3 Put the other end of the wire into the SDO through hole and apply solder

4. Flow Sensor Assembly

4.1 Assemble Cable

The flow sensor requires a 5 V input, however we will run the I^2C bus at 3.3 V necessitating this cable.

4.1.1 Cut the Qwiic cable in half

4.1.2 Cut off the red conductor near the base of the connector

4.1.3 Cut a 30 cm length from the USB cable

4.1.4 Strip the jacket back 1 1/2" from one end

4.1.5 Strip the first 1/4" of each conductor on one end and tin

4.1.6 Twist a small portion of the shield into the black GND conductor

4.1.7 Place a small piece of heat shrink on each conductor of the Qwicc cable and push them toward the connector end

4.1.8 Solder the ends of the Qwiic cable to the ends of the USB cable following Table 2.

USB	Qwiic	Signal
black	black	GND
green	blue	SDA
white	yellow	SCK
red	single 6" wire	5V

Table 2: flow sensor wiring

4.1.9 Connect the red wire to the red condutor in the USB cable and solder the extra long male header pin to the end

4.1.10 Cover the finished joints with a short 1/4" piece of heat shrink

4.2 Connect cable to sensor

4.2.1 Apply a small quantity of solder to the four center pads on the SFM3400 flow sensor

4.2.2 Remove 5/8" of the jacket on the bare end of the USB cable

4.2.3 Strip back 1/8" from each conductor and tin the ends

4.2.4 Thread the cable through the hole in the sensor as shown in the photo and make sure the tinned leads can reach the sensor pads

4.2.5 Connect the end of each lead to the pads you have prepared by touching the tip of the soldering iron to the pad and then placing the tinned lead onto it.

4.2.6 Refer to the SFM3400 datasheet to be sure you are connecting them in the proper position.

4.3 Assemble sensor adapters

4.3.1 Cut a 1 1/2" section of petex tubing

4.3.2 Insert the tubing into the wide end of the flow sensor

4.3.3 Insert the frosted 15 mm to 22 mm adapter over the petex

4.3.4 Add the clear 15 mm to 22 mm adapter over the small end of the flow sensor

4.3.5 Place the bleed adapter over the clear adapter

4.3.6 Use electrical tape to secure any loose connections

5. Final Assembly

5.1 Glue oxygen hose to pressure sensor assembly

5.1.1 Cut a section of oxygen hose approximately the same length as the flow sensor cable

5.1.2 Make sure your cut is flat on the end

5.1.3 Place one end of the hose over the airway pressure sensor and apply glue around the edges

5.1.4 Be very careful not to get glue on the sensor and make sure that the hose is securely attached to the pcb

5.2 Install pressure sensor assembly

5.2.1 Add two standoffs to opposite corners of the pressure sensor assembly

5.2.2 These should correspond to the mounting holes you drilled in the enclosure

5.2.3 Thread the bare end of the oxygen hose through its hole in the enclose from the inside until the sensor assembly is pressed against the wall of the enclosure

5.2.4 Attach the standoffs into the enclosure from the outside using two machine screws

5.3 Zip tie sensor cable to enclosure

5.3.1 Thread the small zip tie through the two mounting holes in the enclosure

5.3.2 Push the qwiic connector end of the flow sensor cable through its hole in the enclosure. It?s a tight fit.

5.3.3 Zip tie the cable to the side of the enclosure making sure that the jacket of the USB cable is engaged by the zip tie

5.4.1 Install ethernet shield and processor

5.4.2 Attach standoffs to the four corners of the ethernet shield

5.4.3 Screw the standoffs into the enclosure from the bottom

5.4.4 Place the ESP32, OLED wing, and Qwiic shield on top of the ethernet shield as depicted in the diagram

5.4.5 Plug the flow sensor and pressure sensor assembly into the qwiic shield

5.4 Add a small piece of electrical tape every 5 inches attaching the oxygen tubing and flow sensor cable

5.5 Wrap a piece of electrical tape around the flow sensor covering the exposed pads of the flow sensor

5.6 Add identification card on the next page card to enclosure.

6. Operation instructions

Provide detailed instructions for the safe and proper operation of the hardware.

- *Step-by-step operational instructions for operating the hardware.*
- *Use visual instructions as necessary.*
- *Highlight potential safety hazards.*

7. Validation and characterization

Demonstrate the operation of the hardware and characterize its performance over relevant critical metrics

- *Demonstrate the use of the hardware for a relevant use case.*
- *If possible, characterize performance of the hardware over operational parameters.*
- *Create a bulleted list that describes the capabilities (and limitations) of the hardware. For example consider descriptions of load, operation time, spin speed, coefficient of variation, accuracy, precision and etc.*

Because the VentMon produces pressure and flow graphs at about 25 samples per seconds which is fast compared to the physiological process of breathing, it is easy to have confidence in the basic dynamic function of the VentMon by simply breathing through it into a plastic test lung. One can easily see that flow and pressure dynamically match the breath or changes in the mechanical ventilation, or even changes induces by changes in the resistance of compliance of the test lung.

The flow and pressure sensors used in the VentMon require not calibration. Flow was checked with a graduated syringe and found to be quite accurate. Pressure sensors can be double checked by checking ambient pressure and, for example, computing altitude of the physcial location.

Other aspects of ventilation such as respiration rate, peak inspiratory pressure, Inhalation to Exhalation ratio, work of breathing, etc. are dependenent on the VentDisplay software which are a module not technically part of the VentMon hardware. This software has been shown to be valuable, although it is dependent on accurate software decisions about the beginning and the ending of a breath, which can be subtle in some situations.

8. Acknowledgements

Robert L. Read conceived of and wrote most of the firmware for the VentMon. Geoff Mulligan created the initial cloud-based IoT appoach which became the PIRDS-logger. Lauria Clarke did most of the hardware design, including the assembly instructions and the printed circuit board based on an open source design from Adafruit[**?**, **?**,] and the SFM3X00 software repo for encapsulating

Sensirion flow sensors in the Arduino environment. This paper was written by Robert L. Read and Lauria Clarke.

The Mozilla Open Source Software foundation gave Public Invention a grant that allowed us to develop the hardware and provide 20 VentMons to open source teams around the world free-of-charge, including paying shipping costs. Protocol Labs provided Public Invention a grant that has been used web communication and conference costs.

We would like to thank early adopters of the VentMon on pandemic ventilator teams such as the ARMEE[] and DIY-Beatmungsgerät.de[] who gave us valuable bug reports and feedback.

Student volunteers on the COVID19-vent-list project[] helped to make the need for the VentMon clear by assessing over 100 pandemic ventilator teams.

9. Declaration of interest

None.