OSVent

- An Open Source, Emergency Ventilator Controller based on Arduino

Ashhar Farhan,
Gordon L. Gibby MD,
Krishna Prasad Mulavisala, MD FRCA

OSVent – An Open Souce, Emergency Ventilator Controlle based on Arduino

Copyright Notice

WARNING

The described system is only meant for evaluation. It hasn't been approved for medical use yet. Be warned that it maybe illegal to use this on Covid patients. It may also be life threatening if the system malfunctions.

Introduction

The Covid-19 Pandemic has stressed the ventilator availability world-wide. This project is an attempt to build a ventilator controller using electronics readily available in many different geographies. Many teams of engineers, doctors and amateurs are racing around the world to develop various mechanisms to ventilate people with respiratory distress. All these systems will need a reliable, easily-manufactured electronics controller that can be made with a variety of components. This paper describes such a controller and how it can be used to turn the household vacuum cleaner (or, prefeably, a suitable high pressure medical gas system) into an emergency patient ventilation system.

Basics of Medical Ventilation

Here is a very brief theory of medical ventilation. Medical ventilators are used when a patient is unable to breathe freely on their own.

- The patient is administered between around 10 to 15 breaths per minute. The OSVent can be setup to control this between 5 and 30 bpm (***Breaths per minute***)

- Each breath has an inhalation phase and an exhalation phase. Usually the duration of the inhalation and exhalation phases are in a radio of 1:2, but this ratio may be optimized with other values. The ratio of durations of inhalation to exhalation time periods is called the ***I/E Ratio***

- During the Inhalation, the air (or a mixture of air and oxygen) is forced into the lungs under carefully controlled pressure. The exact pressure needed is typically measured in centimetres of water level, relative to the atmospheric pressure. So, when a doctor prescribes a pressure of 15 cm H_20, it is assumed to be positive pressure with respect to the atmospheric pressure. The ventilator must be able to specify and control the ***pressure in millimetres of water***.

How the OSVent works

The OSVent works by: (a) switching supply of air compression on and off; and (b) monitoring the air pressure in the ventilation tube. Figure 1 below illustrates this

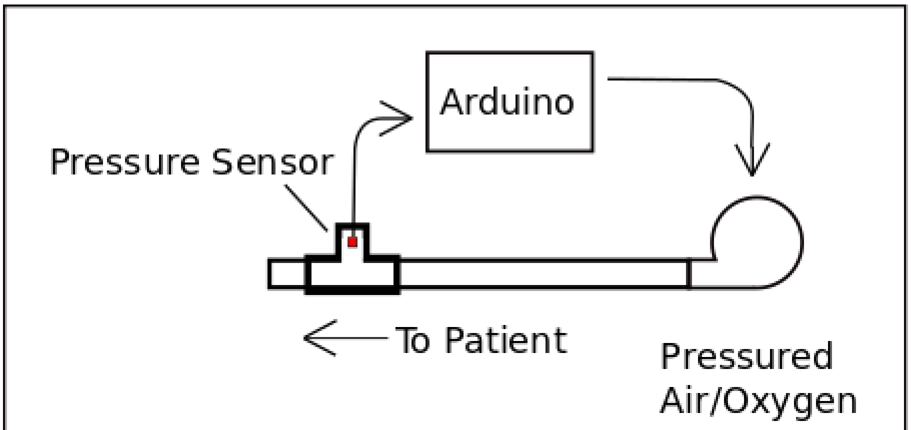


Figure 1

We use a very commonly available air pressure sensors: the BMP180 or the BMP280, manufactured by Bosch Corporation, to monitor the air-pressure inside the tube.

There are three ways to switch the supply of compressed air on and off:

A. **Using Pressurized Oxygen in the Hospital:** Hospitals may have a source of compressed mixture of air and oxygen available. A solenoid operated valve is connected in-line with the pressure regulated gas to switch the inhalation on and off under the Arduino control. In a reciprocal action, another exhalation valve can be turned on to evacuate the stale gas from the patient. It is important that the resistance of the exhalation valve be very low, eg., allowing 50 liter/min at 1 cm H_2O pressure drop.

Commercial valves in the AMBU® system accomplish this, and in intensive care units, pneumatically controlled valves control exhalatio. In emergency cases, a solenoid operated exhalation valve may not be needed and a simple passive valve can be used.

B. **AMBU bags:** A number of open source ventilators have used some mechanism to automate the pumping of an Ambu bag using either a stepper motor or a reversible DC motor.

C. **Vacuum Cleaner:** In extreme cases where neither the AMBU bags nor the pressurized gas is available, a sanitized and carefully cleaned vacuum cleaner’s exhaust (with a suitable particulate filter) can be used as an air compressed source. The power to the vacuum cleaner is switched on during the inhalation phase.

The vacuum cleaners do not have any air pressure regulation and it is vitally important that the air pressure must be monitored for this to work safely. A mechanical over-pressure release can be utilized.

OS Vent Hardware

The OSVent Hardware is based on a low cost Arduino Nano. The schematic is given else where in this paper. Additionally, it has the following peripherals:

- A 16x2 LCD display. This was chosen for its high availability in India and internationally as well
- 2 x solenoid/relay/motor drivers that can source up to 1A of 12VDC power
- 4 x Front-panel buttons
- A TRIAC based AC on/off switch to control AC powered air compressor
- A 9-pin D connector with I2C lines that can be reprogrammed as analog inputs, 5v and 3.3v lines for powering external sensors
- A 4 wire stepper motor interface for working a NEMA17 or NEMA23 stepper motor
- A BMP280/BMP180 pressure sensor made by Bosch that is connected very near the mouth-piece

The solenoid drivers have one side held at ground potential for safety.

All the components were carefully evaluated for their availability before being chosen for the OSVent.

Pressure Sensor

The BMP180/280 is a pressure sensor that is readily available in the electronics components supply shops in India and abroad at a very nominal price. These are the world’s largest selling pressure sensors that are very commonly used by the smart phones and quadcopter drones to approximate the altitude.It is a field proven sensor that senses only the absolute atmospheric pressure and measures it in Pascals. We will use software to measure the pressure in centimetres of water H_20.

OS Vent Software

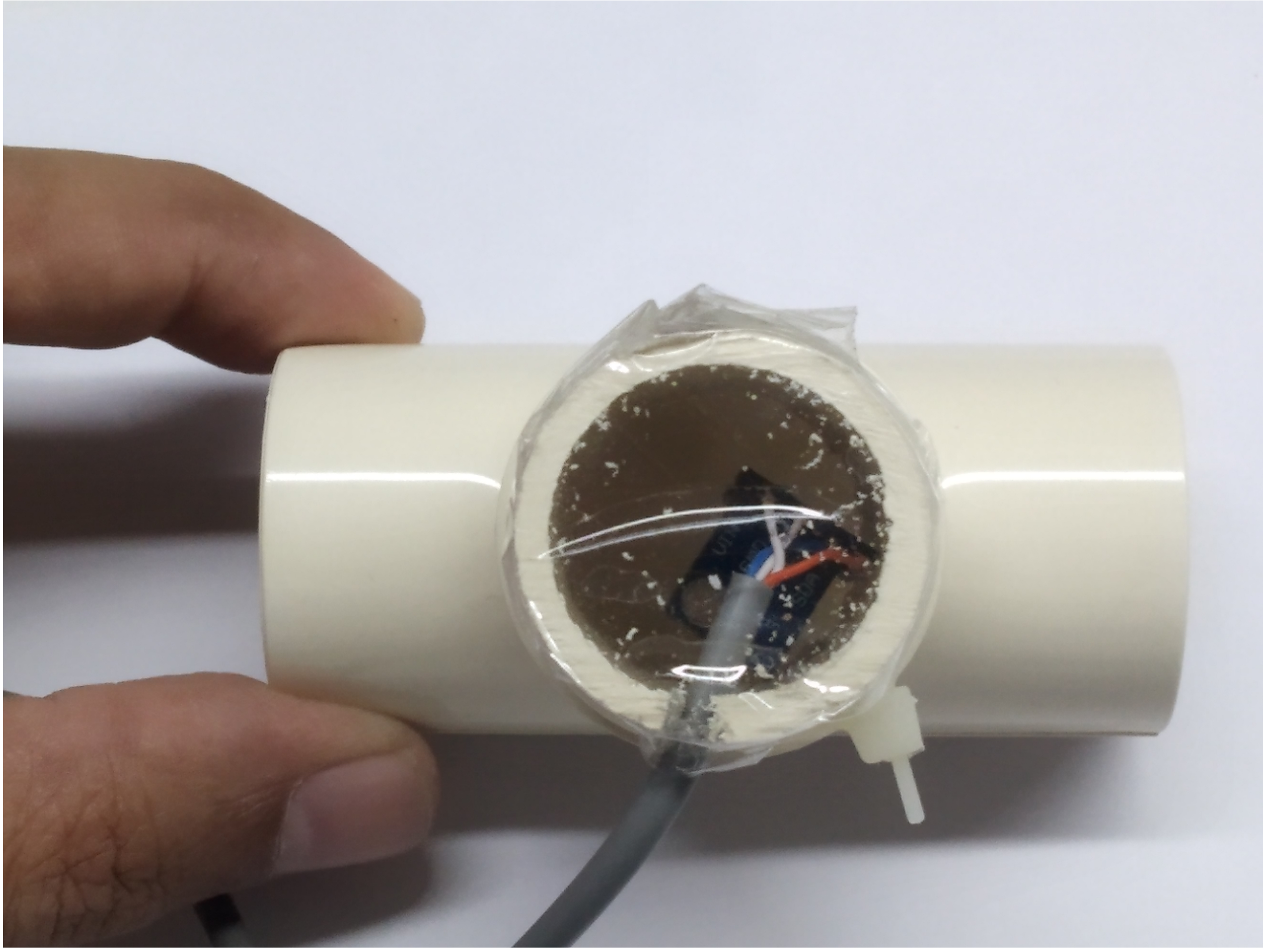
The OS Vent software is designed for safe, reliable working. The software has quite a few unique features:

- Watchdog timer: The watch-dog timer is a hardware circuitry outside the main CPU circuit that resets the CPU if it hangs up for 10 seconds without rearming the watch-dog timer. This protects the OSVent from surges, brown-outs (when the voltage dips below the operating threshold) and EMI spikes that can corrupt the CPU and the RAM.

- Time slicer: The system is written as a number of tasks that are executed in slices, every 100 millisecond. The user interface to check buttons, the alarm, the vent loop itself are all executed one time-slice at a time, in a loop. This architecture makes each module insulated from the other and helps in making the system auditable.

The Pressure sensor

The BMP280/BMP180 pressure sensor is the key to safe and reliable ventilation cycle. The sensor is commonly, locally available premounted on a PCB. We solder an Ethernet cable with four cores to the sensor board (Ground, 3.3V, SCL and SDA) and mount it inside a water plumbing T joint. The sensor is inserted through a slot into the side of the middle opening of the T and the opening is covered with two layers of cellotape and held down with a zip tie.



At reset, before the beginning of any vent cycle, the ambient temperature and the absolute pressure is measured. This is kept as reference for measuring the differential pressure at all phases of the venting cycle.

The Venting cycle

At the start, the baseline pressure (atmospheric pressure) is measured and the relative pressure is measured as an offset from this baseline pressure.

Venting function is implemented in a cycle of nominally 50 steps or phases. If an I/E ratio of 1:1 has to be maintained, then the vent pressure is switched on for 25 steps and turn off for the next 25 steps. The process repeats. At each step, the pressure is measured and plotted. If the pressure exceeds the the programmed pressure limit by 20%, the high pressure alert is set and the audio alarm immediately goes on.

The Front Panel

The front panel has four buttons:

1. Vent On/Off Button
2. Function button (choose between *Breaths per minute*, *Pressure* and *I/E Ratio*)
3. Two buttons to increase or decrease the function chosen by the function button.

The Circuit board

The circuit board has been designed to have minimum wiring required. Except for the power supply, all the other electronics including the display, the buttons and the controller are on a single double sided PCB. We chose to keep the PCB tracks to a minimum of 25 millis to take into account variations in production technology.

A Vacuum Cleaner based ventilator

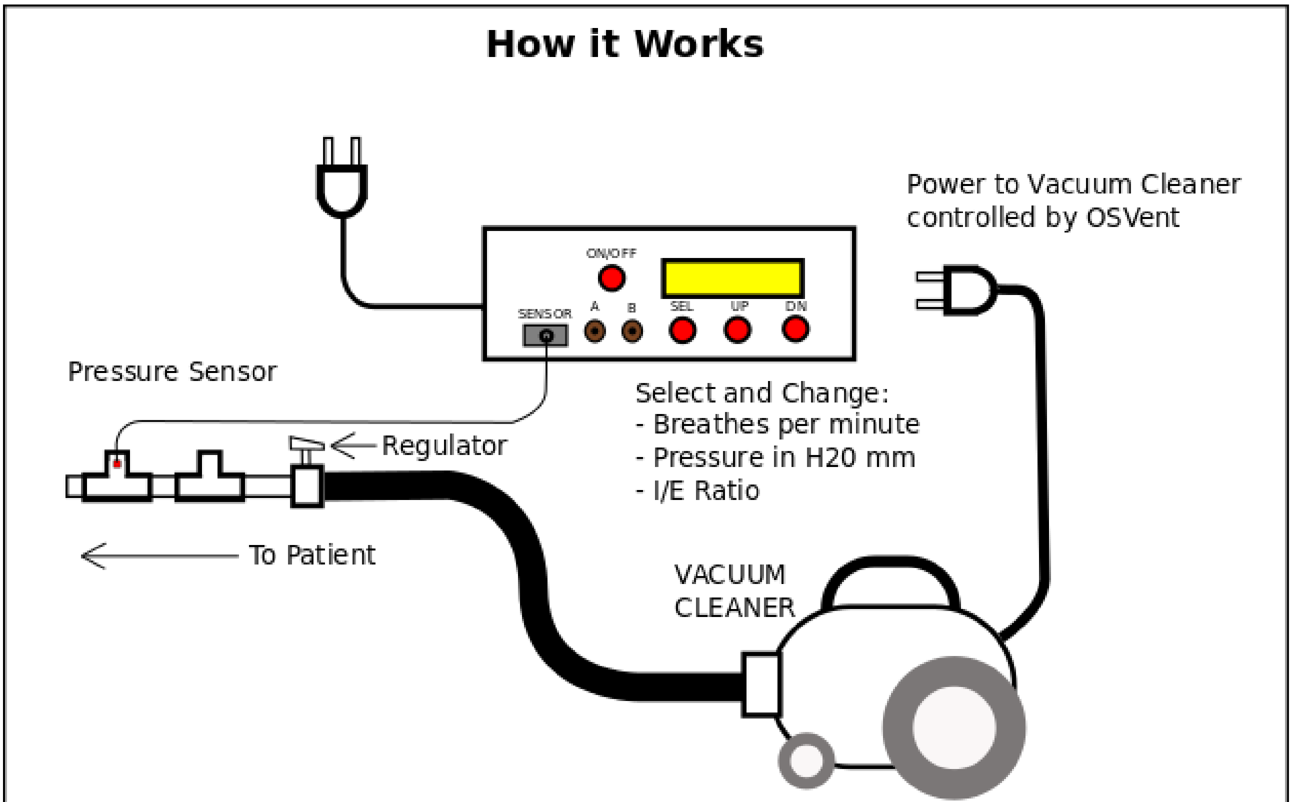


Figure 2

A home-class vacuum cleaner of an Indian brand that was available in the home lab was used to develop a ventilation system. It can be built purely from simple plumbing hardware available from any construction store.

The broad schematic of the system is show in Figure 2.

The system consists of this:

1. The vacuum cleaner is plugged into the back of the OSVent such that the OSVent can turn the vacuum cleaner on and off under program control

2. The exhaust of the vacuum cleaner is connected via a water valve to control the maximum pressure that exits from the pipe

3. A Tee fitting with the attached to the regulator and this followed by another Tee into which the pressure sensor is mounted. The pressure sensor is mounted well inside the Tee such that is not exposed the air flow across the Tee

4. The other end of the Tee is interfaced with an adapter that fits ventilation mouth piece.

This is how it works:

1. At the beginning of the vent cycle, the inhalation phase starts by turning on the vacuum cleaner.

2. The air pressure is continuously monitored. If the pressure goes 20% higher than the programmed amount, the alarm is set and the display status shows ‘HI’. At this point, the attendant can turn the regulator handle back to reduce the air pressure.

3. The first Tee has the side fully open. This allows some air pressure to escape. However, there is ample pressure from all the tested vacuum cleaners to overcome the leakage.

4. The pressure is cut-off by turning of the vacuum cleaner as soon as the inhalation phase ends.

5. As the pressure cuts off, the lungs passively collapse back as the atmospheric pressure is lower than the pressure in the lungs and the exhaled air escapes from the open Tee’s side opening.

6. At the end of the cycle, the cycle is analyzed for low pressure to signal patient disconnection.

The video of such a system is available on YouTube at https://youtu.be/H8-BbmRZL2A

Concluding Remarks

The OSVent is a very flexible system that is quite adaptive to different mechanisms of ventilations that maybe deployed around the world depending upon what is locally available. Though vacuum cleaner based ventilation system is quite a useful innovation for emergency situations, we hope that the standard system of deployment will be based on solenoids that will regulate the flow of piped oxygen/air mixtures from the hospital. Additional safety mechanisms that are being developed include anti-asphyxia mecchanical valves (not needed in this version with an open Tee); electronic over pressure alarms; low-series-resistance pneumatically controlled exhalation valve; electronic flow measurement and alarms; and low-pressure/disconnect alarms. The prototype shown is acknolwedged to be rather primitive, but possiblly life saving in emergent circumstances.

The lack of air flow measurement was due to the unreliable supply of differential air pressure sensors like the NXP’s MPX7002DP. Engineering is always a balancing act between the budget, availability and desirability. With the OSVent Controller, we hope that we have hit a sweet spot in this life saving endeavor.

References:

1. https://groups.io/g/VentilatorDevelopers/
Discussion Group dealing with developments of Open Source Venitlators

2. https://youtu.be/H8-BbmRZL2A
Video demonstrating a working OSVent

3. https://github.com/afarhan/osventproto
Latest source code of the experimental ventilator, schematics

Author contacts:

Ashhar Farhan, farhan@exseed.com
Gordon L. Gibby, ocvacuumtubes@gmail.com
Krishna Prasad Mulavisala, kpmulavisala@gmail.com

OSVent – An Open Souce, Emergency Ventilator Controlle based on Arduino