

Specifications for simple open source mechanical ventilator

Julian Botta, Johns Hopkins Emergency Medicine Resident PGY-3
Twitter: @julianbotta

This is a living document intended to give non-clinicians/non-respiratory therapists an idea of key ventilator features and one proposed simplified design. I encourage other healthcare professionals who are very familiar with ventilators and their use to give me feedback using the comments feature to improve these specifications.

Hardware:

Connection to endotracheal tube: 15mm inner diameter universal connector (ideally slightly conical, slightly wider at the opening to allow for an easy friction-fit while being smooth to allow easy removal, ~15-20mm long)

Ventilator linkage: short-length corrugated tubing (easy flexibility but should not expand or collapse with internal pressures -100 to +100cmH₂O (+/-1.42 PSI), 1-2 feet, 15mm outer diameter universal (cylindrical, 15-20mm long). Because this design does not use separate channels for inhalation and exhalation, the tubing should have minimal volume so as not to contribute to dead space, while being wide enough to avoid adding significant flow resistance (at least 15mm internal diameter).

In-line pressure sensor: low-cost disposable pressure sensor capable of sensing -100 to +100 cmH₂O, with a sampling rate of at least 10 times/second, ideally 100 times/second. The sensor should be shielded from moisture that condenses in the tubing. The portion that will be exposed to the patient's exhalations will have to be disposable (if a thin, flexible diaphragm could be used to conduct pressure accurately, this may lower the cost).

In-line flow sensor: low-cost disposable flow sensor capable of sensing flows 0 to 3L/min, updating at least 10 times/second, ideally 100 times/second. Will be used to calculate volume delivered and detect patients taking a spontaneous breath.

Exhalation blocker: allows inhalation freely, but can be used to block exhalation temporarily for an inspiratory hold (breath hold) maneuver (see below). Can be accomplished either with a solenoid-actuated one-way valve (must default to unblocked during power failure), or a spring-loaded button that physically seals the exhalation port and activates an electronic button to tell the computer to measure a plateau pressure and alarm if blocked longer than 10 seconds.

Standard bag-valve with addition of a PEEP valve on the exhaust port.

The PEEP valve increases resistance to exhalation, allowing the maintenance of a pressure ranging from 0 to 30cmH₂O (typically 5-15) against exhalation. Typically made with a spring holding the exhalation flutter valve shut until enough pressure builds up behind the valve to push it open. The spring is tensioned such that the pressure is adjustable as specified above. Many bag-valves come with a PEEP valve, so manufacture of this piece is not a priority, but PEEP is a crucial element of ventilating patients with many severe lung illnesses.

A very fine filter (N95 spec or higher) over the exhaust port would help make the environment much safer for the healthcare team.

Bag actuator: It should be possible to generate at least 50cmH₂O of pressure relatively quickly, but with very fine control of the pressure and little momentum. The ventilator has to be able to stop itself instantly once either a certain pressure threshold or a specific volume is delivered. It should be able to deliver its maximal breath capacity of at least 900mL in as little as 0.4 seconds.

Bag actuator types:

- Strap: A strap holding the bag against a stationary surface tightens and releases, pulled by being wrapped around a cylinder by a stepper motor.
- Linear: a piston drives a plate against the bag on a stationary surface
- Clamp: one arm pivots to squeeze the bag against a stationary surface, or two arms pinch together around the bag

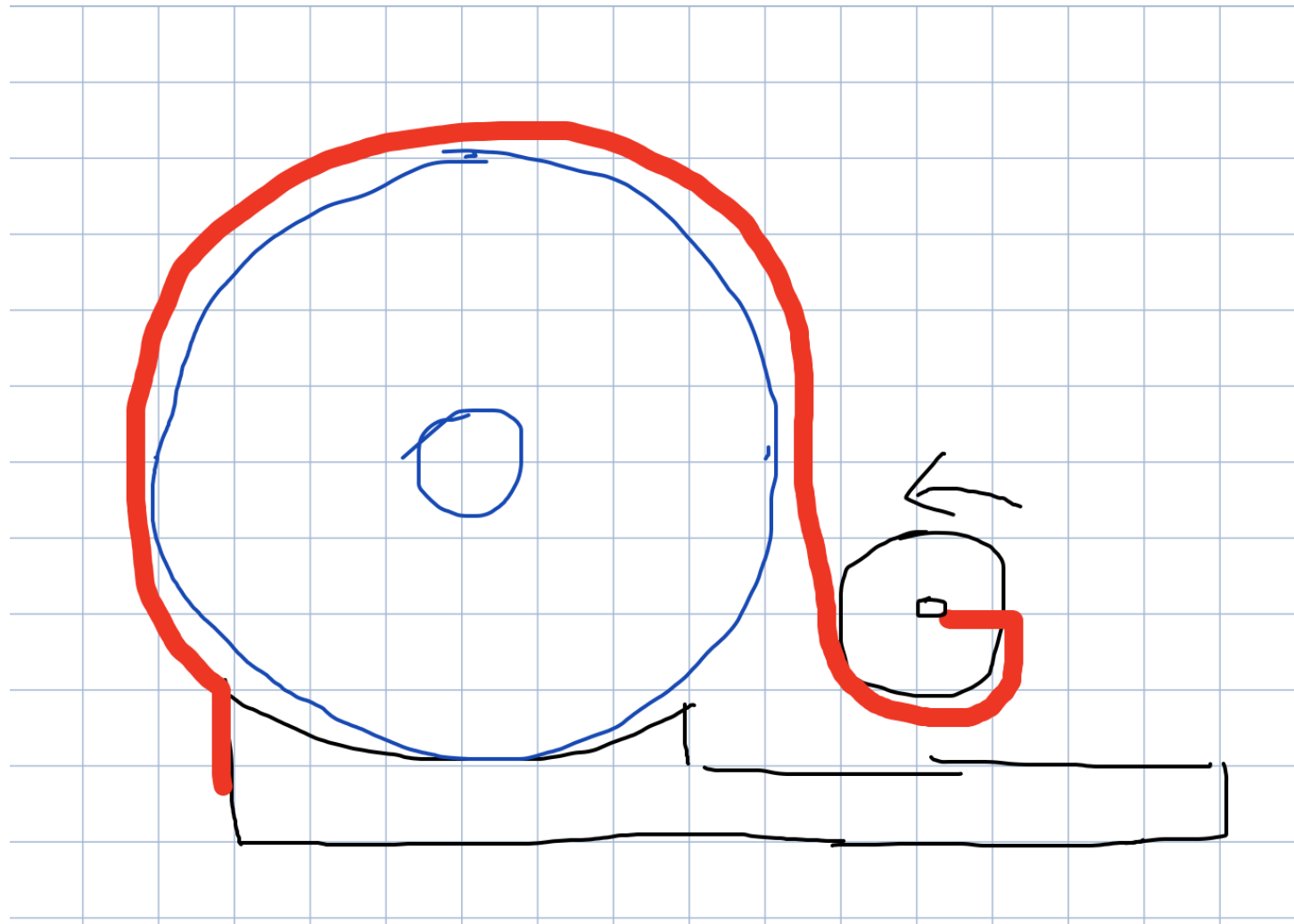


Fig: strap compression: a strap (red) passing over the bag-valve (blue) is pulled by a stepper motor that turns a drum (small black circle).

Electronics:

Raspberry Pi or possibly an Arduino (Raspberry Pi would likely be able to make this a more stand-alone device, and more easily improved in the future.)

Screen with touch or knob/button interface
Speaker for alarms
Battery backup (rechargeable, including battery manager)

Software:

Ventilators have different modes, the most basic of which are:

- Volume Control (aka Assist Control): ventilator delivers a breath of a fixed volume. Does not take pressure into consideration.
- Pressure Control: ventilator delivers a breath until a certain pressure is reached and stops, does not take delivered volume into consideration.

They also have different features:

- Triggered breaths: when the vent detects the beginning of inhalation, it delivers a breath
 - Detected by beginning of flow towards the patient when no breath is being delivered (caused by patient inhaling through the circuit)
 - There must also be a back-up rate, where the ventilator takes over if the patient is breathing too slowly or not at all (determined by the amount of time between breaths)
- Alarms for
 - Pressure too high (for volume control modes)
 - Volume delivered too low (for pressure control modes, requires setting a goal minimum volume)
 - Power failure
 - Tube disconnect (no resistance or change in pressure when giving a breath)
 - Nice to have: alarm for breath stacking (incomplete exhalation causing air trapping from multiple breaths)
- Inspiratory hold: gives a breath and prevents exhalation temporarily while a button is pressed
- Measurement and display (ideally numerically and with a graph) of pressure, tidal (breath) volume, plateau pressure (where the pressure settles during an inspiratory hold)
- Control over
 - Respiratory rate
 - PEEP (not software controlled in this case, see above)
 - Tidal volume or pressure (depending on ventilator mode)
 - Inspiratory time: time over which a breath is delivered

◦