2021 43rd Annual International Conference of the
IEEE Engineering in Medicine & Biology Society (EMBC)
Oct 31 - Nov 4, 2021. Virtual Conference

2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) | 978-1-7281-1179-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/EMBC46164.2021.9630882

MIT Emergency-Vent: An Automated Resuscitator Bag for the COVID-19 Crisis*

Teddy Ort[1], Nevan Hanumara[1], *Member IEEE*, Amado Antonini[1], Brandon Araki[1]
Murad Abu-Khalaf[1], *Member IEEE*, Michael Detienne[1], David Hagan[1], Kimberly Jung[1]
Aaron Ramirez[1], Shakti Shaligram[1], Coby Unger[1]
Albert Kwon[2], Alex Slocum Jr.[3], Christoph Nabzdyk[4], Dirk Varelmann[5], Jay Connor[6],
Daniela Rus[1], *Member IEEE*, Alexander Slocum[1]

***Abstract*— MIT's Emergency-Vent Project was launched in March 2020 to develop safe guidance and a reference design for a bridge ventilator that could be rapidly produced in a distributed manner worldwide. The system uses a novel servo-based robotic gripper to automate the squeezing of a manual resuscitator bag evenly from both sides to provide ventilation according to clinically specified parameters. In just one month, the team designed and built prototype ventilators, tested them in a series of porcine trials, and collaborated with industry partners to enable mass production. We released the design, including mechanical drawings, design spreadsheets, circuit diagrams, and control code into an open source format and assisted production efforts worldwide.**

***Clinical relevance*— This work demonstrated the viability of automating the compression of a manual resuscitator bag, with pressure feedback, to provide bridge ventilation support.**

I. Introduction

By March 2020 the novel coronavirus pandemic was stressing ICU capacity in Europe and especially Italy [1], where difficult decisions were made around rationing clinical care. Ventilator shortages became apparent and demand outstripped supply. Multiple efforts were launched worldwide to develop and deploy ventilators and substitutes of varying complexity by established medical device companies, numerous teams at universities, large and small companies, and home hobbyists.

One popular strategy was to automate the compression of a manual resuscitator bag, an FDA approved device found in ambulances and throughout clinical facilities. This approach was prototyped back in 2009 as a project in the MIT Medical Device Design course and presented at the 2010 Design of Medical Devices Conference [2]. However, in the absence of a pandemic, support was not available to continue the project.

*This project was supported by the generosity of private donors.
Materials available at `https://emergency-vent.mit.edu`

[1] Massachusetts Institute of Technology, Depts. of Mechanical Engineering, Computer Science & Artificial Intelligence, Electrical Engineering, Cambridge, MA, USA (e-mails: teddy@mit.edu; hanumara@mit.edu, rus@mit.edu, slocum@mit.edu, phone: +1.617.258.8541).
[2] Anesthesiology, Stanford University, Stanford, CA
[3] Plastic Surgery, Medical College of Wisconsin, Milwaukee, WI
[4] Anesthesiology & Critical Care, Mayo Clinic, Rochester, MN
[5] Anesthesiology & Critical Care, Brigham & Women's, Boston, MA
[6] Orthopedic Surgery, Mt. Auburn Hospital, Cambridge, MA

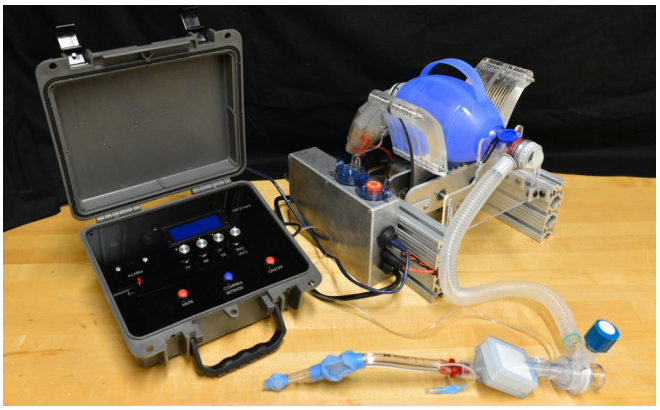

Fig. 1. The MIT Emergency Ventilator uses a robotic gripper to automate the task of manually squeezing a resuscitator bag to serve as a bridge ventilation device.

In early March 2020, numerous requests for plans and manufacturing information to replicate the 2010 design began arriving from around the world. Realising the inherent danger in releasing an untested design, but compelled to respond to enquiries, the course staff convened an emergency, multidisciplinary team comprised of doctors and mechanical, electrical, and software engineers and students and alumni. We recruited a team of leading anesthesiologists and critical care clinicians that provided key input into the updated requirements for a de minimis design and a use scenario as a bridge device to enable rationing of advanced ventilators. Together, we sought to answer the question: *Is it safe and feasible to ventilate a person by automating the compression of a manual resuscitator bag?*

Concurrently, we launched a website[5] to communicate our design and results as they developed, receive feedback, and foster discussion via an open forum. The information was accessed by tens of thousands worldwide and helped support multiple projects with shared principles but different designs as a function of local adaptations.

Many other efforts were simultaneously launched, both by industry and other research institutions. Some, such as a NASA's VITAL [3], aimed to redesign the conventional ventilator from the ground up by reducing the complexity and using fewer parts. Philips, a market leader in respiratory care,

embarked on converting a bilevel positive airway pressure (BiPAP) machine into a ventilator, now the Philips E30 [4]. Other groups also utilized resuscitator bags including the MADVent [5], which uses a lanyard wrapped around a motor pinion to pull a compressor arm, the Ambovent [6], which uses a single rotating compressor, the ApolloBVM [7], which uses a rack and pinion, and the Coventor [8], which employs a crank and slider. A concern with many of these designs and the 2010 MIT project [2] is the risk of premature bag failure due to friction from rubbing and bending while squeezing the bag. The Emergency-Vent uses a dual-hand robotic gripper design to ensure the bag is compressed naturally on both sides as it would when squeezed by a human hand.

II. Design Requirements

Existing state-of-the-art hospital ventilators have a large number of features and settings that allow a clinician a high degree of customization, but they cost tens of thousands of dollars. To simplify the design for emergency use, we worked with our clinical team to identify the minimum set of requirements to safely ventilate a patient with COVID-19 and, more generally, Acute Respiratory Distress Syndrome (ARDS).

A. Ventilation Settings

Two main ventilation strategies exist: volume and pressure control. We focused on the former, since a resuscitator bag inherently supplies a volume of air, and identified five required settings: Tidal Volume, Respiratory Rate, Inspiration to Expiration Ratio, Positive End-Expiratory Pressure (PEEP), and Assist Control Trigger Threshold. These must be set and adjusted by a clinician based on monitoring a patient's vital signs, particularly blood oxygenation. We sought to keep the interface flexible, straightforward and reliable.

1) Tidal Volume: The volume of air supplied to the patient during the inhalation phase of the breathing cycle should be adjustable between 200 and $800\,\mathrm{mL}$. An initial setting can be made based on body weight as per the NIH's ARDSnet protocol [9].

2) Respiratory Rate: The ventilator must be capable of providing 6 – 40 breaths/minute (BPM). A normal respiratory rate is typically $18\,\mathrm{BPM}$, but patients being weaned off ventilators require lower rates, while patients suffering from severe ARDS require higher rates, with small tidal volumes to reduce the risk of barotrauma (damage due to over pressure).

3) Inspiration:Expiration (I:E) Time Ratio: A typical value for this setting is 1:2 whereby inhalation occupies a third of the breathing cycle. Desired values can range from 1:1 to 1:4.

4) PEEP: At the end of exhalation, lungs return to atmospheric pressure. During ventilation, setting a nonzero PEEP typically ranging from $5{-}15\,\mathrm{cm\,H_2O}$ prevents complete exhalation and helps keep the aveoli open. This parameter is set with a valve on the resuscitator bag.

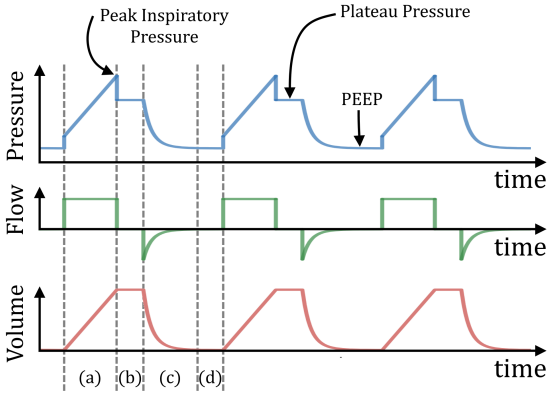

Fig. 2. Pressure, flow, and volume waveforms of the breathing cycle during volume control ventilation showing the four phases: (a) Inspiration, (b) Inspiratory Hold, (c) Expiration, and (d) Expiratory Hold.

5) Assist Control Trigger Threshold: When patients are sedated, but not paralysed, Assist Control (AC) mode provides a breath when either the patient attempts to inhale spontaneously or a timer runs out. This avoids asynchrony with the ventilator. An inhalation attempt can be detected via a pressure drop with a threshold range of $2-5\,\mathrm{cm\,H_2O}$ below PEEP.

B. Waveform Calculation

The breathing cycle can be divided into four distinct regions:

a) **Inspiration:** Air is pumped into the lungs under driving pressure.
b) **Inspiratory hold:** The lungs hold the full volume of air and the pressure settles at plateau pressure.
c) **Expiration:** The pressure is released and the air flows out of the lungs.
d) **Expiratory hold:** Pressure is held at PEEP until it is time to start a new breathing cycle.

The timing of each of these phases must be controlled in order to achieve the desired ventilation parameters, and it is important to prevent small timing errors in each phase from accumulating, resulting in an incorrect breathing rate. Hence, the overall cycle timing must be independent of the individual phase timing specification. In Fig 2 we show pressure, flow, and volume over time during these four phases of the waveforms.

In Sec II-A we described the input ventilator settings specified by the operator. Here we show how those settings are used to calculate the timing requirement of each of the four phases of the breathing cycle. Later, in Sec IV we describe how these specified timings are achieved by the control system. For ease of notation, we denote the given ventilator settings as Tidal Volume (V_T), Respiratory Rate (RR), I:E Ratio (IE), PEEP (P_{PEEP}), and Pressure Trigger (P_T). Note, that here $IE \in [1, 4]$ is a real number such that $I{:}E = 1{:}IE$. We first calculate the desired period of the waveform as $T = 60/RR$. Next, we define the inspiratory hold duration as a brief pause before exhalation is allowed

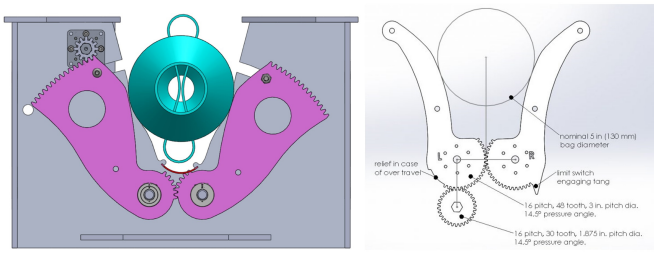

Fig. 3. Gripper with left *Top Drive* and right *Bottom Drive* designs.

to commence. Based on clinical recommendation we fixed this value as $T_h = 100\,\mathrm{ms}$. Thus, we can now calculate the duration of the inspiratory phase as $T_{in} = T/(1+IE) - T_h$. Finally, the total duration of the expiratory phase including the expiratory hold is $T_{ex} = T - (T_{in} + T_h)$. Since exhalation happens passively, we simply wait until T_{ex} has elapsed at which point the next breathing cycle is triggered.

C. Pressure Measurement

While this device is a volume control ventilator, we still need a pressure sensor for AC mode, clinical feedback, and fault detection. Three key pressures must be measured and displayed for each breathing cycle (see Fig 2). While these values are not directly controlled, they are used to trigger critical alarm conditions, as described in Sec IV-D.

1) Peak Inspiratory Pressure: P_{ip} is measured throughout the inspiratory and inspiratory-hold phases. At the start of each breathing cycle it is reset, and at the end of the inspiratory phase, the maximum detected value is shown. To avoid barotrauma, this value must remain under $40\,\mathrm{cm\,H_2O}$, otherwise an over pressure alarm triggers.

2) Plateau Pressure: P_{plat} is reached at the end of the inspiratory hold. At that point, the flow of air into the lung has stopped, thus the driving pressure has subsided and P_{plat} is typically lower than P_{ip}. However, a large gap can indicate obstructions in the tubes or airway, therefore this value is important both for triggering alarms and as feedback to the physician.

3) PEEP: While the clinician manually adjusts the PEEP valve, we measure this pressure for display, assist control, and fault detection.

III. Mechanical Design

A. Functional Requirements

The mechanical design is driven by the following set of functional requirements:

a) Compression: The bag must be gently squeezed from both sides, as is done manually, to maximise air expelled without damaging the bag.
b) Cycles: The system must operate continually for at least 14 days. An RR of up to $40\,\mathrm{BPM}$ translates to 806,400 cycles.
c) Power: The power delivered by the motor(s) must ventilate a patient in the worst possible scenario.
d) Support: The bag fixture must have axial compliance to allow for deformation of the bag, and it also must avoid jostling the oxygen and breathing tubes.
e) Fail-safe operation: If the machine fails, a clinician must be able to immediately remove the bag and proceed with manual ventilation independent of the machine.
f) Simplicity & Flexibility: The design should allow other teams to rapidly adapt it to local supply chains.

B. Implementation

In our first review of the past design, we identified that a single actuator squeezing the bag against a rigid surface would stress and rub the bag and cause it to tug on the oxygen line and breathing tube. Therefore, we created a symmetrical double gripper design that evenly squeezes the bag from each side, based on simple pivoting members (fingers) with convex surfaces that create rolling contact. These fingers are geared together at their base pivot points, so that they are driven by a single motor, thus avoiding the challenge of tandem control.

Two variants were developed in parallel (Fig 3): The *Bottom Drive*, which placed the motor beneath the fingers' pivot point and drove the fingers with a pinion, and the *Top Drive*, which placed the motor in a frame above the level of the fingers and drove a gear section at the tip of one of the fingers. The former was a more compact design but required a higher torque DC gearmotor, while the latter was larger but had a greater mechanical advantage and could use a lower torque motor. Both variants used the same control strategy.

The Bottom Drive was executed in a modular 80/20-based aluminum frame as a development platform that could be adjusted and easily replicated around the world. The Top Drive unit was configured around a box frame made of tab-in-slot sheet metal parts, which lent itself to an array of industrial fabrication modalities in steel and aluminum. Care was taken to select materials and gear dimensions to avoid fatigue failures, with potentially millions of cycles of reciprocating motion.

IV. Electronics and Control System

The electrical system design focused on using affordable parts, with multiple variants available in large quantities. Fig 4 shows a schematic of the main electrical hardware, with less than 20 total parts. Here we list the requirements for these components and note any pertinent details.

A. Electrical Design

1) Microcontroller: The main microcontroller used in this design is the widely available Arduino Mega [10]. However, any Atmega based chip would be compatible with both the hardware and software.

2) Motor Selection: Speed and torque output were the main considerations for choosing the motor, along with low cost, high availability, and ability to be back-driven so the bag can be manually removed in an emergency. To determine the maximum speed, ω_{motor}, we use the worst case from

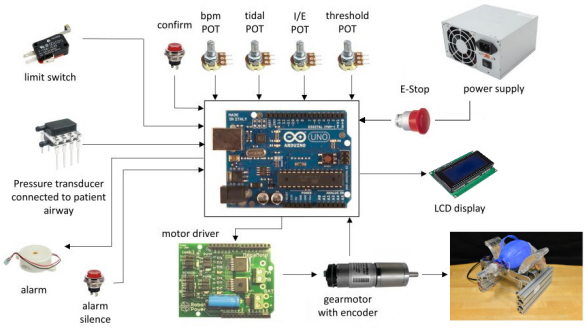

Fig. 4. A schematic of the various electrical components necessary to build the robotic gripper, selected and designed with an emphasis on using only inexpensive and widely available parts to ensure rapid scalability.

the waveform specification in Sec II-B. With $IE \leq 1$, the inspiration phase is never longer than the expiration so the motor must move the fastest in that phase. Therefore:

$$T_{\text{in}} = T/(1+IE) - T_h = 60/RR(1+IE) - T_h \quad (1)$$

In the worst case, $RR = 40\,\text{BPM}$ and $IE = 4$, then (1) yields $\min T_{\text{in}} = 200\,\text{ms}$. Based on calibration (Sec IV-C.3) we found a required finger sweep of $\theta_{\text{finger}} = 27° = 0.47\,\text{rad}$ to achieve the full required $V_T = 800\,\text{mL}$. Therefore, the worst case motor speed is $\omega_{\text{motor}} = (r \cdot \theta_{\text{finger}})/T_{\text{in}} = 3.76\,\text{rad/s} \approx 35RPM$, where $r = 48/30$ is the gear ratio from the motor to the finger shaft on the *Bottom Drive*.

We determine the maximum required torque, $\max \tau_{\text{motor}}$, from the force the finger must exert on the bag to achieve the maximum required pressure of $\max P_{\text{airway}} = 40\,\text{cm}\,H_2O$ and the finger's lever arm, $l_{\text{finger}} = 12\,\text{cm}$, accounting for losses as follows: The force is pressure times contact area ($A_{\text{bag}} \approx 100\,\text{cm}^2$) so we have the torque at each finger

$$\max \tau_{\text{finger}} = \max P_{\text{airway}} \cdot A_{\text{bag}} \cdot l_{\text{finger}} \cdot \eta_{\text{bag}}^{-1} \approx 9.4\,\text{N m} \quad (2)$$

where we assume an efficiency $\eta_{\text{bag}} = 50\%$ for converting the mechanical force into pressure. Finally, accounting for 2 fingers, the gear ratio r, and a similar 50% gear efficiency, we obtain the maximum torque at the motor shaft

$$\max \tau_{\text{motor}} = 2\eta_{\text{gear}}^{-1} \max \tau_{\text{finger}} \approx 20\,\text{N m} \quad (3)$$

Thus we chose a motor-gearbox combination for which these values of τ_{motor} and ω_{motor} fall within the torque-speed curve ensuring the motor is capable of delivering the specified waveform. For this prototype, we selected a gearmotor with integrated encoder, the AndyMark PG188.

3) Motor Driver: The motor driver is a standalone microcontroller responsible for the low-level controllers (Sec IV-C.2). While it may have been feasible to implement the motor controller on the Arduino itself, we found the lack of a threading module on the Arduino an impediment to real-time simultaneous control of the motor, user interface, safety alarms, and high-level control. Therefore, we recommend a dedicated motor controller. In our case we used a BasicMicro Roboclaw Solo 30A.

4) Encoder: An incremental quadrature rotary encoder is integrated into the gear motor and used by the controller to provide feedback for the PID controllers (IV-C.2). Note that it is important to position the encoder at the drive shaft rather than on the finger shaft to obtain high resolution feedback. Since the drive shaft rotates at a much higher rate than the fingers, even though our encoder only has 7 poles, we receive 700 counts in a single $30°$ finger sweep.

5) Limit Switches: A limit switch is used to home the motor, with the arms in the open position, each time the machine is turned on. This also serves as a safety stop in case of over travel.

6) Pressure Transducer: The pressure transducer is the key feedback sensor for the patient airway. It is crucial that it can detect high pressures ($> 40\,\text{cm}\,H_2O$) that cause barotrauma. Therefore, the sensor must be able to measure accurately at least up to that point. Since all airway pressures are measured with respect to ambient temperature, a relative rather than absolute sensor should be used. Finally, in order to use AC mode, the sensor must be able to measure *negative* pressures. From Sec II this could be as low as $-5\,\text{cm}\,H_2O$ if $P_{PEEP} = 0$.

7) Power Supply: Although we obtained a rough power estimate from the motor and microcontroller requirements, we recommend measuring the required power empirically at the maximum rated load. For our system we measured a maximum draw of 5A at 12V or 60W, which agreed with our estimate.

8) User Interface: We incorporate several peripherals into the user interface including: 1 LCD display, 1 emergency stop button, 1 alarm buzzer, 1 alarm visual indicator LED, 3 touch buttons (start, stop, and alarm silence), and 4 potentiometers for setting V_T, RR, IE, and P_T.

9) Printed Circuit Board: We designed a custom printed circuit board which is used to connect all of the previously listed components quickly and securely. The circuit board schematic is freely available on our website for reproducing the board at any PCB mill.

B. High Level Control System

The main function of the control system is to take the waveform parameters calculated in Sec II-B from the operator settings and control the motor to produce a matching waveform. The interface must also be monitored for changes to the settings along with other operating parameters to detect unsafe conditions and respond appropriately. Here, we describe the function of the finite state machine (FSM) shown in Fig 5 which controls normal operation. In Sec IV-D, we describe the safety and alarm controller which handles fault detection. The homing procedure is always run on device startup and doesn't repeat unless triggered by an alarm condition. After homing, the following five states, representing the stages of the breathing cycle, repeat indefinitely in normal operation.

1) Inspiration State: This first state initializes $t = 0$. The motor is commanded to move to the position corresponding to the V_T and arrive at time T_{in} where V_T, T_{in} are obtained

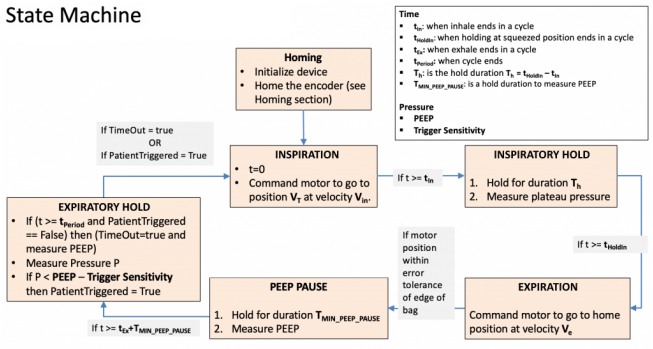

Fig. 5. A finite state machine is used represent the consecutive phases of the breathing cycle.

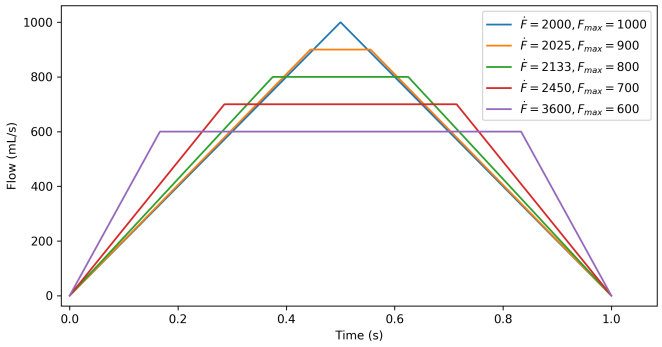

Fig. 6. A family of decelerating flow profiles generated by Eqs (4,5). While all of the curves deliver the same $V_T = 500\,\mathrm{mL}$, they illustrate the trade-off between minimizing acceleration $\dot{F}$ and max flow rate F_{max}.

from the waveform values calculated in Sec II-B. We describe our choice of velocity profile in detail in Sec IV-C.1.

2) Inspiratory Hold State: The fingers pause for T_h, after which the FSM immediately transitions to the exhalation phase. If the fingers have delivered the full volume, they maintain their position, however, if there is a position error, it is corrected at the beginning of this state. Before leaving this state the P_{ip} and P_{plat} are measured as described in Sec II-C.

3) Expiration State: During the expiration phase, the fingers travel to their fully open position. The actual exhalation is a passive process governed by the lung pressure as well as compliance and resistance of the airway.

4) PEEP Pause State: This is a timing state, physically equivalent to the *Expiratory Hold* state. It is implemented to ensure that the expiratory hold lasts long enough to obtain a reliable PEEP measurement before a patient inspiration can trigger a state change when AC mode is active.

5) Expiratory Hold State: In this state, the motor speed is zero and the system waits for the waveform timer to complete to start a new breathing cycle. In AC mode the transition can be triggered early if a spontaneous breath lowers the pressure to $P_{PEEP} - P_T$. Otherwise, the timer waits for T_{ex} as calculated in Sec II-B.

C. Motor Control

The low-level motor control takes place on the motor driver which receives commands from the main microcontroller via a serial connection. Each command consists of a tuple with two values: $cmd = (V_{T,goal}, t_{goal})$ Where $V_{T,goal}$ is the goal tidal volume and t_{goal} is the time allotted to reach that position.

1) Motion Profiles: Flow rate of air delivered is a function of finger rotation velocity. Therefore, it is not sufficient to simply ensure the correct volume is delivered in the set time. Rather, a carefully controlled velocity profile can reduce high flow rates which result in dangerously high P_{ip}. Furthermore, since the motor must accelerate to a nominal velocity and then decelerate to rest at the destination position, it is important to provide sufficient time for these accelerations to reduce motor torque and mechanical stresses on the mechanical components. For these reasons, we selected a hybrid triangular/trapezoidal flow profile, similar to [11], which has the advantage of reducing the P_{ip} by offsetting the maximum flow rate from the maximum volume, while also reducing mechanical stress on the motor. To do so, we derive triangular and trapezoidal flow profiles according to

$$F_{tri}(t) = \begin{cases} \frac{4V_T}{T_{\text{in}}^2} t, & 0 < t \leq \frac{T_{\text{in}}}{2} \\ \frac{4V_T}{T_{\text{in}}} \left(1 - \frac{t}{T_{\text{in}}}\right), & \frac{T_{\text{in}}}{2} < t \leq T_{\text{in}} \end{cases} \tag{4}$$

$$F_{trap}(t) = \begin{cases} F_{max} \frac{t}{T_a}, & 0 < t \leq T_a \\ F_{max}, & T_a < t \leq T_{\text{in}} - T_a \\ F_{max} \left(\frac{T_{\text{in}} - t}{T_a}\right), & T_{\text{in}} - T_a < t \leq T_{\text{in}} \end{cases} \tag{5}$$

where F_{max} is the flow at the maximum allowable motor velocity and $T_a = T_{\text{in}} - V_T/F_{max}$ is the acceleration and deceleration time. The flow profiles described in Eqs (4,5) are shown in Fig 6 for a range of flow limits $F_{max} = 600 - 1000\,\mathrm{mL}$. The key property of these equations is that they all satisfy $\int_0^{T_{\text{in}}} F(t)dt = V_T$, e.g. they deliver the specified volume V_T at the required time T_{in}. Intuitively, the triangular profile is ideal, in that it has the lowest acceleration (slope of the flow curve). However, we choose a trapezoidal shape when the triangular curve would exceed the maximum flow constraint derived from the mechanical velocity limits.

2) Position and Velocity Controllers: To actuate these profiles, we implemented a pair of PID loops in the motor controller, tuned for this device under load. The first PID controller tracks the desired velocity set point and ensures that the motor followed the flow profile described above. Once the motor is in the vicinity of the goal, the position PID controller is used to drive the final motor position so the correct volume of air is delivered. This two phase approach ensured that the stringent timing and position requirements were met and the correct volume of air was delivered not only by the specified time, but also at the specified rate.

3) Calibration: The final step in implementing high-level motion control was to convert tidal volume commands to motor angles. This conversion depends on the gear ratio, finger length, and the non-linear relationship between bag compression and volume of air displaced, which varies by

bag brand. We found that among different bags of the same manufacturing model, no recalibration was needed. Since the calibration takes less than an hour and only needs to be performed once for each bag model, we don't believe calibration will pose a significant obstacle.

We implemented a semi-automated calibration procedure. An uncalibrated resuscitator bag is placed in the device and the procedure automatically steps through a sequence whereby the arms are rotated to a series of fixed positions and the volume flow is measured and recorded. We found that a second order polynomial model captured the volume-angle relationship, with residuals typically below $1\,\mathrm{mL}$ RMS. It is important to note that other dynamics exist, related to airway resistance and lung compliance, which could affect the volume of air delivered. However, this model provides a good approximation and we leave the modeling of those affects for future work.

D. Safety and Alarm System

Designing any medical device requires identifying failure modes, such as a blockage or mechanical failure, and appropriate mitigation measures or alerts. Working with our clinical team, we identified typical faults as well as those specific to a bag-based ventilator. These, along with the desired alarm behaviours are shown in Table I. For each, we detail how it is detected, how it can be resolved, the appropriate device response, and what message should be displayed along with the audible and visual alarm cues.

We verified these alarm conditions by partially and fully blocking the patient tubing, disconnecting the tubing, dropping solid objects into the fingers to prevent them from closing, and removing and misplacing the emergency resuscitator bag from the fingers. In each of these tests, the alarms sounded, the appropriate message was displayed, and the corrective action as seen in Table I was triggered.

V. Porcine Studies

We conducted four IACUC-approved porcine studies to guide the design effort by identifying issues early and to verify the overall bag-based strategy with clinicians and in comparison with a hospital-grade ventilator (Fig 7). (CBSET Inc., Lexington MA, Study: HME00087, IACUC Protocol: I00313)

1) Functional Test: This first study was conducted to evaluate the overall operation of the device and identify critical failure modes. Two Yorkshire swine, between 25 – 30 kg, were sedated and one was ventilated using a Puritan Bennet 840 and the other using the first iteration of the MIT Emergency Ventilator. This study enabled us to gather data on the pressures involved in ventilation and, consequently, the forces and stresses on the system. This first system, with a frame made of laser-cut acrylic failed, leading to a major redesign, along with improvements in the user interface; however, a comparison between swine was not possible.

2) Verification with Manual Resuscitator: This study again sought to validate the ventilation capabilities of the device. We tested an updated prototype on a single 27kg pig. First, the tidal volume was varied and measured with a spirometer to verify the ability to control volume of air delivered. Next, flow-volume plots were recorded with the porcine ventilated by the MIT Ventilator and compared to those produced by an experienced anesthesiologist manually pumping the resuscitator bag. Finally, the porcine was ventilated with our system for several hours and its arterial blood gases were measured. The anesthesiologist was able to control the blood gas parameters solely using the user interface on the device. Three days post study the subject was observed to be healthy.

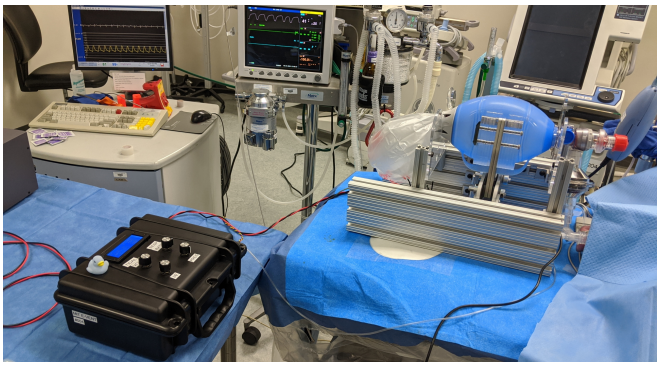

Fig. 7. A prototype undergoing testing on a live sedated pig.

3) Operational Training: We created a user manual detailing the function and control of the system and provided it to a volunteer anesthesiologist who had never seen the device. Working with an 81kg porcine, he was asked to set specific settings on the control box and collect flow-volume data, then to change settings to achieve specific pH ranges on arterial blood gases, and to mimic ventilator settings that he would normally provide to patients with ARDS. The anesthesiologist quickly became familiar with the control system, was able to complete the experimental tasks within 3 hours, and provided detailed design feedback.

4) Assist Control Mode: The final study was conducted on the same 81kg porcine as Study 3. Another volunteer clinician, who had never seen the device, was given the user manual, quickly became familiar with the controls and was able to successfully manipulate arterial blood gases as requested. Various alarm functions were checked. This study also tested Assist Control for the first time. The sedation was reduced and the minute ventilation reduced until the porcine started to take spontaneous breaths in between the mandatory volume control breaths. Assist Control mode was turned on and the threshold adjusted until the ventilator delivered assisted breaths in sync with the porcine subject, as described in Sec IV-B.

VI. Conclusion

We have demonstrated the viability of a low-cost ventilator that employs a robotic gripper to automate manual resuscitation bags. In this paper, we described in detail the design considerations that motivated our work and emphasize the importance of starting from clinical parameters, the safety

TABLE I
SAFETY ALARMS

Alarm Name	Triggering Condition	Clearing Condition	Response
Exceeded P_{ip} Pressure	$\mathbf{P > P_{max} = 40cmH20}$ Ensures the unit can never deliver air at a pressure higher than 40 cmH20 to avoid barotrauma.	One complete breathing cycle with no over pressure	Immediately open gripper completely, resume normal operation at start of breathing cycle
Low Pressure	$\mathbf{P_{plat} < P_{plat,min} = 5cmH20}$ Detects any disconnections or leaks and notifies the clinician to check the breathing tube.	Measured plateau pressure normal	Continue normal operation
High Resistive Pressure	$\mathbf{P_{ip} - P_{plat} > P_{resist,max} = 2cmH20}$ Notifies the clinician that there is unusual resistance in the breathing tube or airway.	One resistance pressure measurement in normal range	Continue normal operation
Over Current Fault	$\mathbf{I \geq I_{max} = 4.5A}$ Detects that the motor drawing excessive current, indicting that something may be stuck in the mechanism or there is a blockage in the breathing tube.	One complete breathing cycle without an overcurrent event	Immediately open gripper, repeat the homing procedure, resume normal operation
Tidal Volume Not Delivered	$\mathbf{V_{final} < V_{set} - V_{thresh} = 50mL}$ Detects when the commanded tidal volume is not delivered in the time allotted.	Delivered tidal volume in the correct range	Continue normal operation
Tidal Pressure Not Detected	$\mathbf{P_{peak} - P_{PEEP} < P_{tidal,min} = 5cmH20}$ (2 cycles) Detects situations where the pressure does not reflect inspiration, but there may not be a leak, such as a bag removed from the gripper.	One complete breathing cycle without a tidal pressure failure	Continue normal operation

and alarm systems required for use on real patients, and the results of our porcine studies.

In developing their "Emergency Use Resuscitator Systems Design Guidance" the AAMI COVID-19 Response Team drew heavily from our open source work [12]. Furthermore, as a testament to the power of the open source model to foster collaboration and distributed problem solving, several efforts worldwide have made use of our information, including groups in New York City [13], India, Ireland, Chile, Peru [14], Netherlands, Israel, Spain, Italy, Iran [15], and Afghanistan [16].

Looking forward, we see this technology as applicable to pandemics, mass casualty events, first responder use and transport scenarios, as well as step-up and step-down clinical care. Especially in lower resource markets, the success of teams worldwide suggest a more locally resilient approach to medical technology design, validation and production.

Finally, we hope that the unusual clinical – academic - industrial relationships, launched in this time of need, will endure post pandemic and contribute to increased interdisciplinary, collaborative and international approaches to problem solving.

REFERENCES

[1] "The extraordinary decisions facing italian doctors," https://www.theatlantic.com/ideas/archive/2020/03/who-gets-hospital-bed/607807/, 2020, accessed: 2020-10-30.

[2] A. M. Al Husseini, H. J. Lee, J. Negrete, S. Powelson, A. T. Servi, A. H. Slocum, and J. Saukkonen, "Design and prototyping of a low-cost portable mechanical ventilator," *Transactions of the ASME-W-Journal of Medical Devices*, vol. 4, no. 2, p. 027514, 2010.

[3] "How engineers at nasa jpl persevered to develop a ventilator," https://www.jpl.nasa.gov/news/news.php?feature=7661, 2020, accessed: 2020-10-30.

[4] "Philips respironics e30 ventilator," https://www.usa.philips.com/healthcare/medical-specialties/covid-19/sleep-and-respiratory-care-covid-19/e30-ventilator/, 2020, accessed: 2020-10-30.

[5] A. Vasan, R. Weekes, W. Connacher, J. Sieker, M. Stambaugh, P. Suresh, D. E. Lee, W. Mazzei, E. Schlaepfer, T. Vallejos *et al.*, "Madvent: A low-cost ventilator for patients with covid-19," *Medical devices & sensors*, vol. 3, no. 4, p. e10106, 2020.

[6] "Ambovent," https://ambovent.org/, 2020, accessed: 2020-10-30.

[7] "Appollobvm," http://oedk.rice.edu/apollobvm/, 2020, accessed: 2020-10-30.

[8] "A ventilator system built for rapid deployment," https://med.umn.edu/covid19Ventilator, 2020, accessed: 2020-10-30.

[9] "NIH NHLBI ARDS Clinical Network Mechanical Ventilation Protocol Summary," http://www.ardsnet.org/files/ventilator_protocol_2008-07.pdf, 2014, accessed: 2020-10-18.

[10] "Arduino mega 2560," https://store.arduino.cc/usa/mega-2560-r3, 2020, accessed: 2020-10-30.

[11] A. Alvarez, M. Subirana, and S. Benito, "Decelerating flow ventilation effects in acute respiratory failure," *Journal of critical care*, vol. 13, no. 1, pp. 21–25, 1998.

[12] "Emergency use resuscitator system (eurs) design guidance," https://www.aami.org/docs/default-source/standardslibrary/200424_cr503-2020_rev1-1.pdf, 2020, accessed: 2020-05-03.

[13] "New york needed ventilators. so they developed one in a month." https://www.nytimes.com/2020/04/20/technology/new-york-ventilators-coronavirus.html, 2020, accessed: 2020-05-03.

[14] J. Chang, A. Acosta, J. Benavides-Aspiazu, J. Reategui, C. Rojas, J. Cook, R. Nole, L. Giampietri, S. Pérez-Buitrago, F. L. Casado, and B. Castaneda, "Masi: A mechanical ventilator based on a manual resuscitator with telemedicine capabilities for patients with ards during the covid-19 crisis," *HardwareX*, vol. 9, p. e00187, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S246806722100016X

[15] "Iranian engineers develop open source ventilator," https://spectrum.ieee.org/the-institute/ieee-member-news/engineers-iran-open-source-ventilator, 2020, accessed: 2020-05-03.

[16] "Afghan all-girls robotics team designs low-cost ventilator to treat coronavirus patients," https://www.reuters.com/article/us-health-coronavirus-afghanistan-ventil/afghan-all-girls-robotics-team-designs-low-cost-ventilator-to-treat-coronavirus-patients-idUSKCN24L0WO, 2020, accessed: 2020-10-30.