	Data	Signatures	
Author: Celso Monteiro	17/04/2020		Title: **The eAR System – Ventilator SDK**
Verif.:			
Aprov.:			

1. Scope

The scope of this document is to describe the creation of the eAR system. The eAR system is a low-cost SDK for inclusive ventilator systems developed in around 30 days during the CoVid19 pandemics by a self-funding voluntary multidisciplinary team in Brazil. We'll present the design alternatives, choices, results and limitations, focused on AMBU automation. This is far from being the best system but it's the best we can do now on what we have available.

All electronics hardware, mechanical, and software documents are available in open hardware, open software basis in the Githut https://github.com/RespiradorHacker/Projeto-EAR-Celso

This design can be changed at any moment without previous announcement.

Disclaimer: we have no responsability about any device built upon information from eAR project and we emphasize that devices must be built in strict compliance to local regulations, following expert recommendations and they must be homologated by local regulatory agencies.

2. Document Control

- April 17, 2020: Document emission R00

3. References

- Open Source COVID19 Medical Supplies facebook: https://www.facebook.com/groups/670932227050506/
- COVID-19 Air BRASIL - Fast production of assisted ventilation devices https://www.facebook.com/groups/235476464265909/
- RDSV200320 Emergency Automatic Respirator Specification R00 – Preliminary
- RDSV200322 EAR Hardware Proposition Draft R00 – PTBR
- RDSV200327 EAR Project Briefing PTBR R04
- **MIT E-Vent – "Key Ventilation Specifications"**
- **MHRA (UK) Rapidly Manufactured Ventilator System**

4. Abbreviations and Terms

- MIT: Massachusetts Institute of Technology (USA)
- MHRA: Medicines & Healthcare products Regulatory Agency (UK)

- AMBU: in this scope is just a silicone bag
- BDC: Brushed Direct Current
- FIO2: Fraction of Inspired O2 defines proportion of O2 in inspired air
- ECU: Electronic Control Unit
- PEEP: Positive End Expiratory Pressure
- PTH: Plated Through Hole, electronics technology
- SPI: Serial Peripheral Interface
- WIP: Work in Progress
- SDK: system development kit

5. Design Reasons

There are several AMBU automation solutions, why do we need another one? All AMBU automation we could find are based on materials available in research labs. These materials were not found normally in emerging countries like Brazil and especially on the worldwide uprising of Covid19. Detours on purchasing are announced in a daily basis for all resources targeted to fight Covid19.

So eAR basic concept is creating a system that could be built using available components and local resources. It should be a possibility for any town, including small towns in emerging countries.

6. eAR Design - Hardware

The AMBU bag was the enriched air propeller chosen because it was available at all places, used by emergency medical care worldwide.

We needed something to compress the AMBU so at first we were considering an indirect compressed air based AMBU compressing device but this solution needed a dedicated chamber, more complicated to build and we considered that some small hospitals had few available spots of compressed air.

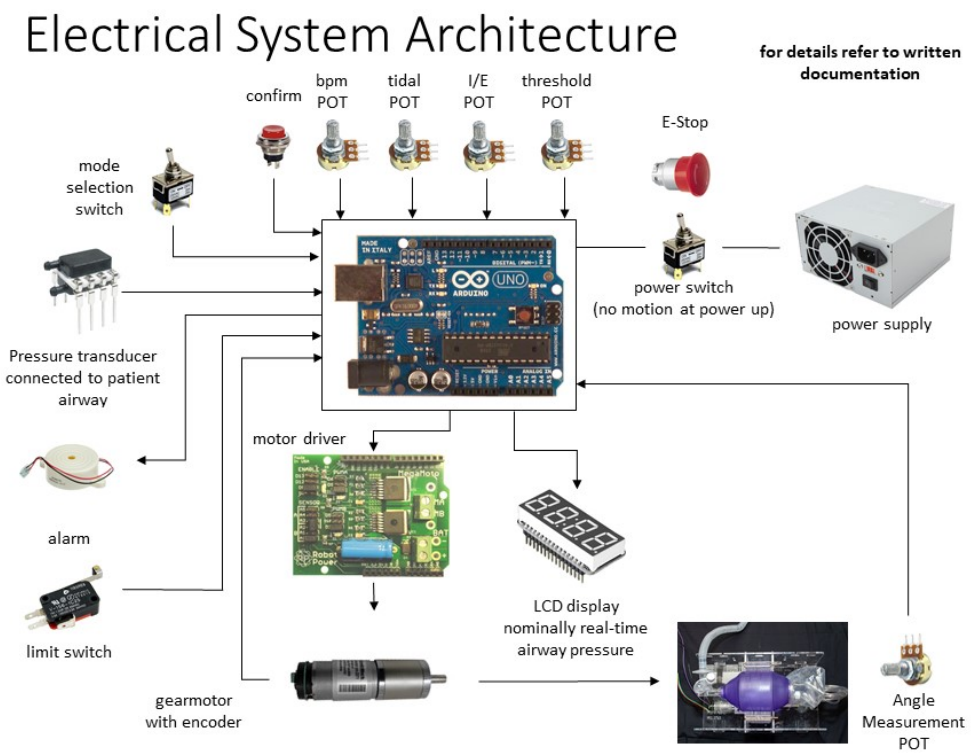
So we switched to electric motor possibilities. There are many alternatives that could do the job, we've considered NEMA type stepper motors, car windshield and car window motors. We had previous experience with all three motors. NEMA motors are the only brushless option, but we discarded because of availability in smaller towns. Windshield motors were discarded because they are power hungry, heavy and they were not built to reverse operation. We needed reversal because we wanted to control Tidal volume without mechanical changes. Car windows were the most available (up to 4 on each car and million cars spread through the country) and they had excess torque, low power consumption and reversal capability. There's a life limit associated with this choice, although these motors work on worm gears and they have rubber couplers, they are brushed motors, and they wear. Due to low load condition we've estimated around 1 million cycles (to be confirmed). Possibly it'll last longer than the AMBU.

MODEL		VOLTAGE	NO LOAD		STALL			WEIGHT
		RATED	SPEED	CURRENT	TORQUE		CURRENT	g
		V	r/min	A	N·m	kg·cm	A	
	JC-578VA-4720 Standard Specifications	12	92	1.3	9.1	93.0	24.0	508

Motor used in eAR system

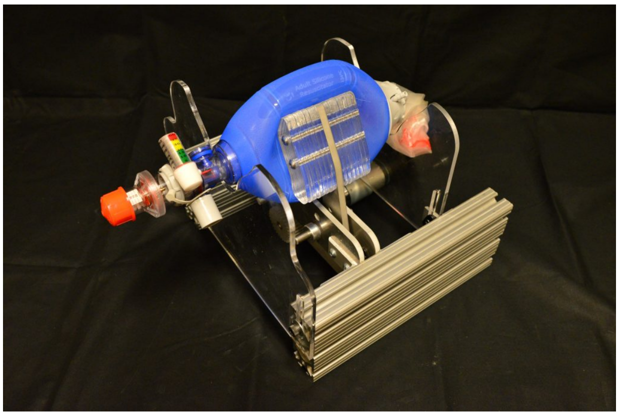
We've built mechanical prototype 1, it was similar to an Israeli prototype, with direct traction compression arm. It was brute and hard to control. We've changed to mechanical prototype 2, this model used the window mechanism original pulleys and steel cable. It was controllable but the steel cable needed to be kept tight, turning it mechanically complicated. The mech prototype 3 introduced gears between the arm and the motor gear with approximately 1:4 reduction. It worked smoothly even under 3V and so far we've been enhancing this model.

At some point between mechanical prototype 2 and 3, we started to stablish preliminary specifications. Since the beginning MIT E-Vent design was inspiring, but we found the MIT specs were insufficient. So we found Rapidly Manufactured Ventilator System from MHRA (UK) and we considered it as the main specification. The eAR design now had a main spec: MHRA and a pathway: MIT E-Vent.



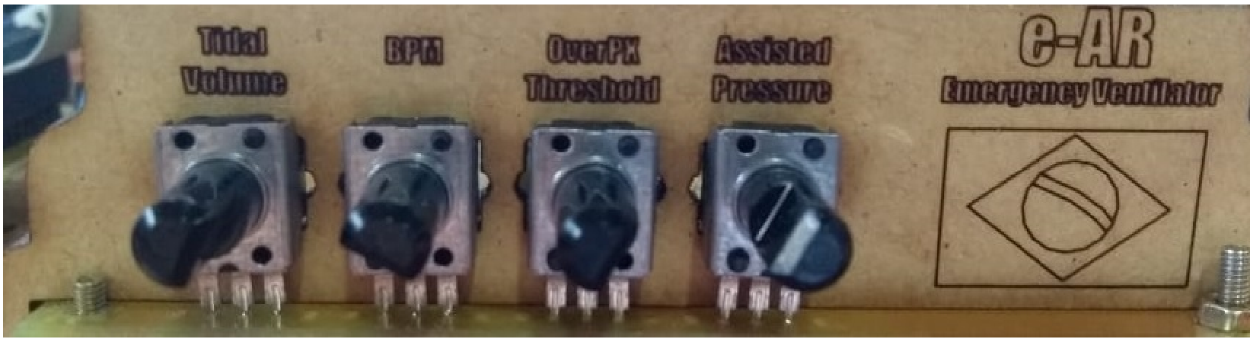
E-Vent Hardware Architecture

Why we did not use the MIT as is? MIT E-Vent was nice, but the BDC chosen motor was not available here. MIT motor has encoder and planetary gears, and locally available planetary motors use tiny plastic gears which could be unreliable in this application. There were no schematics or lines of code. So we adapted eAR hardware to what we could afford.



MIT E-Vent Ambu Compressor

We've kept the 4 potentiometers of MIT Design, added failsafe resistors for failure mode open when operating. We replaced I/E for Assisted Pressure Threshold on pot, according to MIT recommendation.

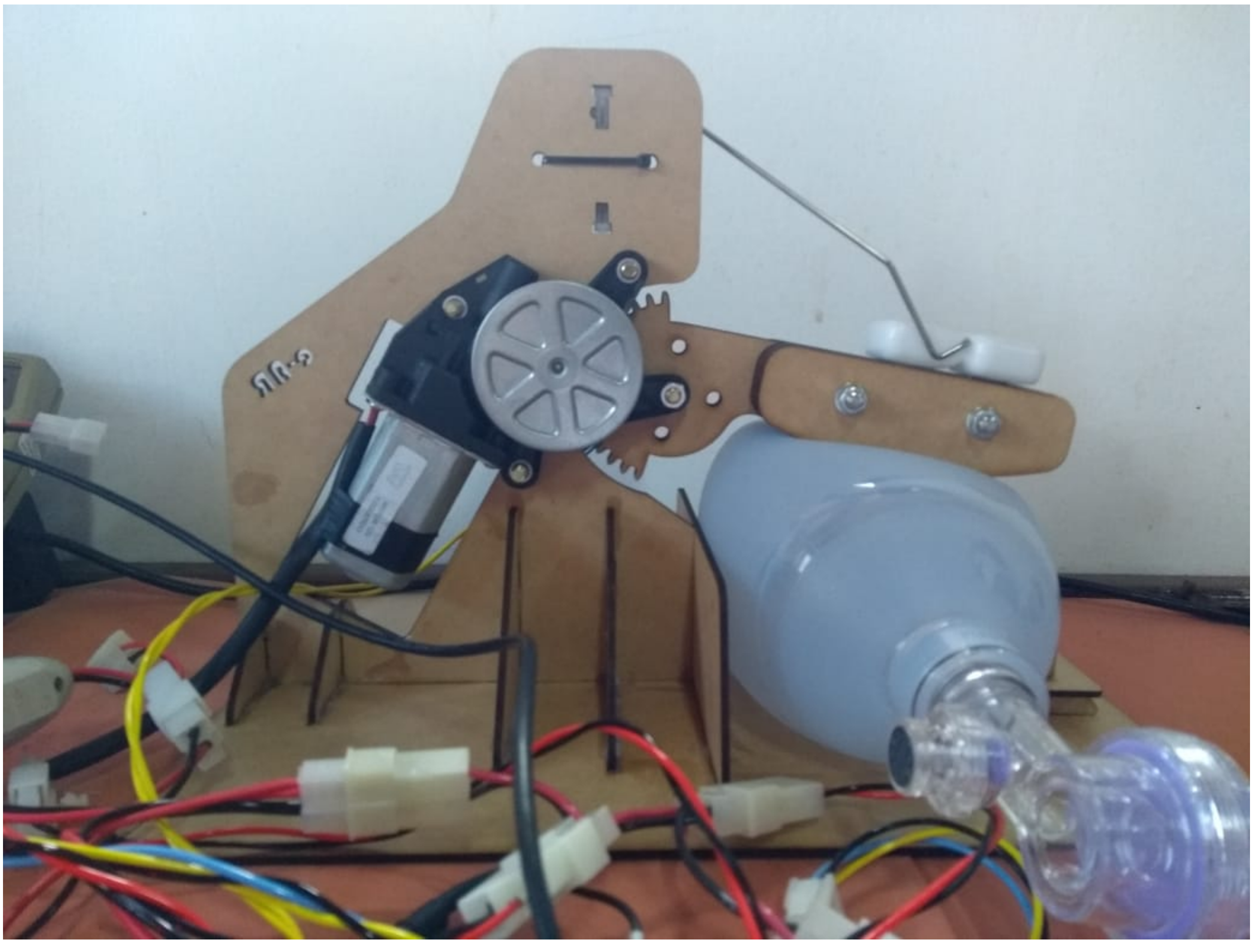


eAR prototype front panel main controls

We did not use MIT display because we do not consider 7-segment displays adequate for instant visibity, so we depend on external pressure viewer which could be an electronic bargraph panel, a mechanical set or a GUI, currently being developed.

About the MCU, we chose Arduino Pro Mini due to availability and small DIP24 footprint. We are not 100% sure that Arduino will meet all reliability issues so we added some extras to improve reliability like some ESD protection and an external hardware watchdog timer. If Arduino ultimately fails validation, we can provide a PCB with another processor using Arduino's DIP24 footprint.

We were not able to find an available superb quality angle measurement pot as MIT did. So we started using a revolution sensor and end-of-course to read AMBU compressing arm position. But motor speed changed according to arm travel and effort. So we've introduced a car tank float sensor as an absolute arm position sensor. I've designed in nineties the production tester for the GM Corsa tank float sensor for Indebras, so we know the sensor and the failure modes. Although the Corsa sensor was over 20 years old, it worked flawlessly. These sensors usually fail open so it's easy to notice intermittent failures so we can replace sensors before catastrophic failure. In order to avoid problems with this type of sensor, we recommend observing the curve of the sensor during movement. In poor quality sensors, it is easily noticed flatness at some points, this is due to grey sensor manufacturers "enhancing" carmakers checkpoints (e.g 1/4, ½, ¾) In mechanical prototype 3.1 we've upgraded to Gol G3 sensor, with increased resolution and linearity.

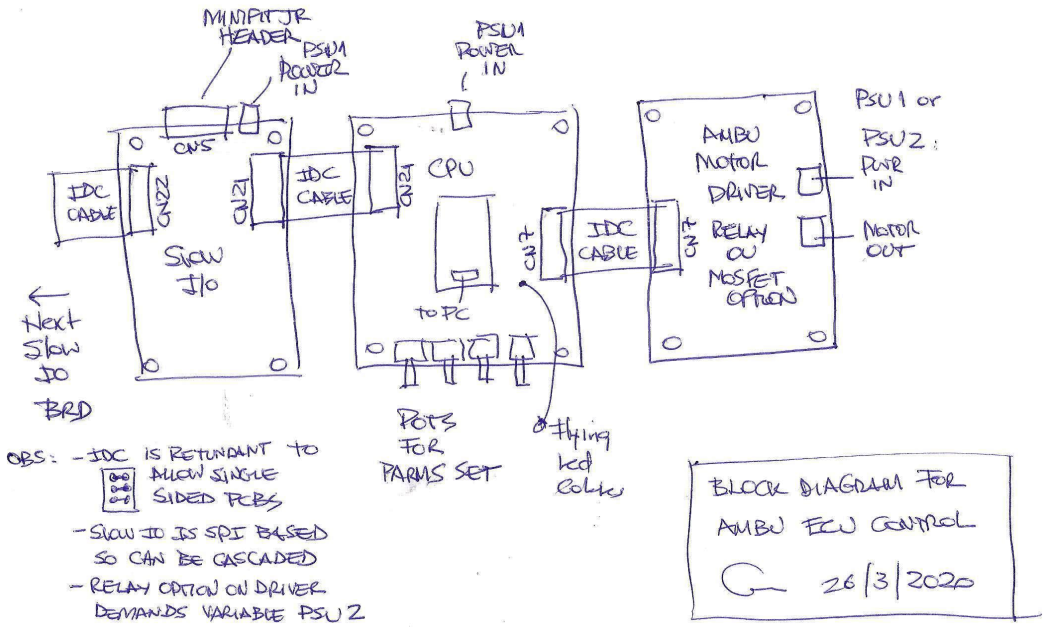


eAR mechanical prototype 3.1 mounted with MDF parts

We've directly correlated the Arm position Sensor with Tidal Volume because the arm displacement is related to Ambu compressing and it will tell the possible maximum displacement of enriched air for the patient. This is a restriction on eAR system, at least for now it does not measure airflow to pacient, it will only estimate the real quantity of dispensed air. This was done in order to keep system costs down because a reliable flowmeter would be the most expensive part of eAR system. If necessary we can review it, maybe using a custom built hot wire flowmeter. Our expectation is that the necessary Tidal volume will be set by the healthcare professional turning the Tidal volume knob clockwise and observing the patient chest cavity displacement. Other drawback from not using a flow sensor is that some asynchronies during assisted ventilation will be harder to identify, maybe we'll miss some of them.

Regarding the motor driver, we could not identify an available PWM H-bridge motor driver. There are popular drivers based on L298 but they are not specified to handle the 7-8 Amps peak motor current during motor start. We finally developed two driver versions: first model is a relay based motor driver that does not use PWM and you have to add an external power supply or a variable DCDC converter to set motor speed. The second model is a Mosfet based PWM that allows dynamic motor control. The design allows you to use any TO220, DPAK or D2PAK low level drive N-Channel mosfet that meets the current and voltage specs.

There are currently 2 versions of printed circuit boards: the first version is the inclusive one. It is a set of 3 PCBs, 10x10cm, single sided with PTH components. Anyone can buy the shelf 10x10cm copper PCB, use a DIY PCB method and get the boards done. The connections between the boards are handled by IDC Flat cables, with retundant vias to increase reliability. These flat cables can be obtained even on old PC floppy cables. We've downsized technology to allow this inclusive design. POC proved it works!



Early Inclusive version sketch

The second PCB version is a single 10x10cm FR4 board SMT version. This alternative allows higher volume production.

We used a standard 5V pressure sensor, NXP MPX5010, with 10 KPa sensivity but any sensor working on a range of 5-70cm H2O will work. This sensor is linear, so the resolution depends on the ADC we're currently using. The manufacturer's spec for pressure precision is 5% but we've achieved less than 1% precision in previous designs based on 2-points calibration procedure. This calibration procedure is very simple: we build some calibrated tubing, place the sensor in a stardard position regarding the water column and use the system to read raw readings of 2 points, e.g. 5 and 40 cm H2O. With this readings we enter raw readings as parameters on a provided Lookup Table Generator software (based on Arduino too in order to avoid extra software) so we have a now 2-points calibrated curve. We've got precision better than 1 cm H2O.

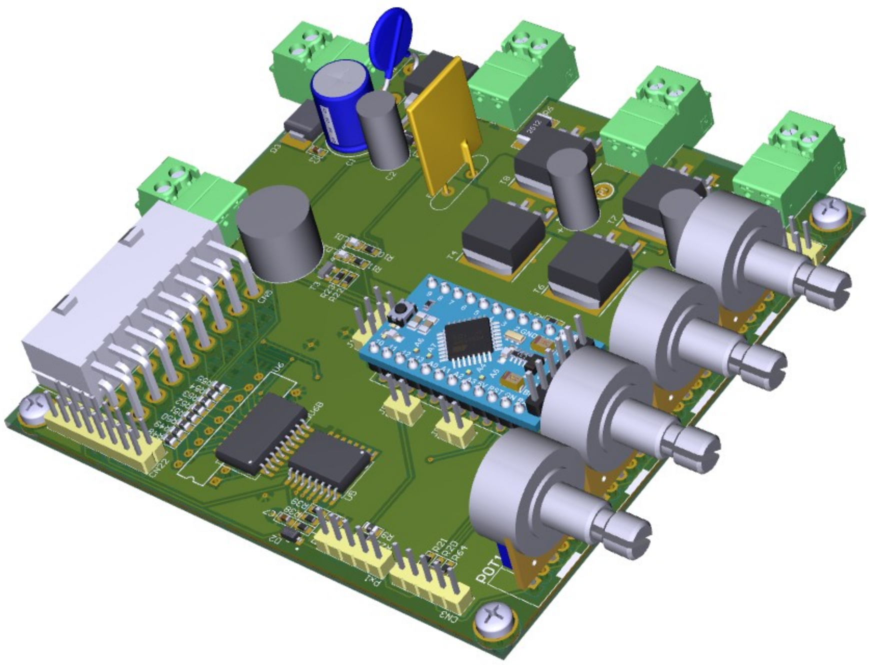
We also added some extra IO to eAR so we could read external switches and drive directly 12Vdc loads, e.g external relays or light loads. We used SPI interface so the expansion is modular based on 8 Inputs/8 Outputs base module. Using 4 of these inputs we already implemented I/E Ratio and Plateau Hold Time selection.

Mode selection switch is directly connected to Arduino. This switch defines forced mode or assisted ventilation.

Status as April 17, 2020:

- Inclusive eAR hardware (PTH single sided boards):
    - eAR ECU CPU Board: designed, tested, works
    - eAR ECU I/O Expansion Board: designed, tested, works.
    - eAR ECU Motor Driver Board – relay option: designed, tested, works. Rework required to handle arm sensor (added later) and changed HSA and LSB signals. Recommended use of Mosfet driver instead due to best performance and no extra PSU.
    - eAR ECU Motor Driver Board – mosfet option: designed, electronics works, untested PCBs.
    - eAR Pressure Sensor board: designed, electronics works, untested PCBs
    - eAR watchdog Timer Board: designed, electronics works, untested PCBs

- eAR hardware for mass production (FR4 SMT boards):
    - eAR ECU (CPU + IO + Mosfet motor Driver + watchdog Timer) expected release this week: designed, electronics works, untested PCBs
    - eAR Pressure Sensor board: designed (PTH), electronics works, untested PCBs



eAR Hardware for mass production – 10x10 cm Single Board SMT solution

7. eAR Design - Firmware

Arduino is a platform for learning embedded systems, with a nice IDE and one of the largest base of developers. It is not really a reference for real time designs. In order to implement the real-time needs for eAR, we proposed a very simple software architecture.

This architecture is based on an interrupt based timer, that generates the system timebase, which is currently 10ms.

We established an eternal loop like PLC software, and we added a deterministic task scheduler, or task sequencer. So we basically have one part of the firmware that runs every 10ms and 10 tasks that execute every 100ms.

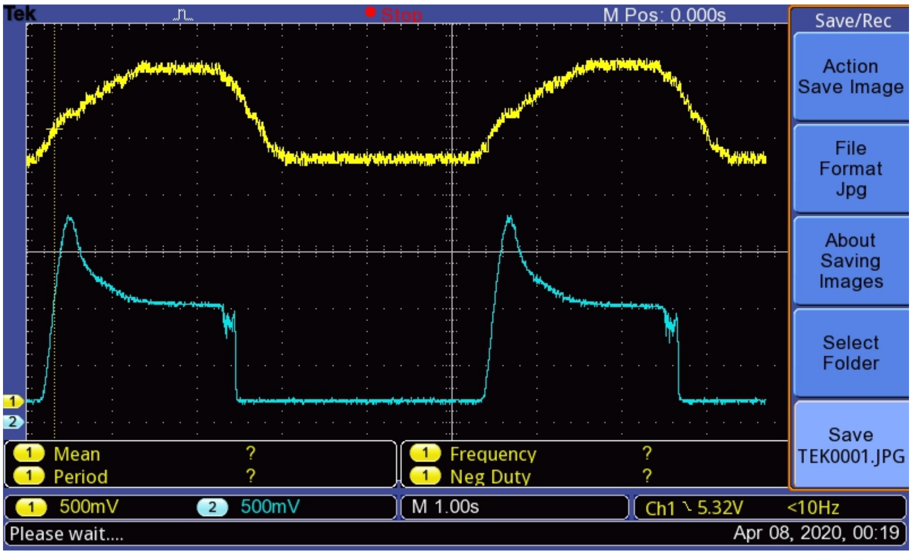
Notice that the tasks are not dynamically created, there is no RTOS structure or anything that could be stack or processing time intensive. This is a tiny 8-bit processor, there are no resources for that. Flash footprint is 32KB and so on.

Final arm position is tightly correlated to Tidal volume potentiometer so arm course is directly proportional to Tidal pot position.

BPM pot is translated into 8-30 BPM according to MIT proposed math functions. Total breathing period takes into account I/E ratio, inspiration time, plateau hold time and expiration time. All of them are variables. I/E ratio and Plateau hold time are switch setpoints, inspiration and expiration time are calculated from BPM, I/E ratio and Plateau hold time.

Float processing is avoided, notice that the Arduino MCU cannot even handle 16/8 division on hardware. Most math are handled using integer processing, sometimes lookup tables.

Software changes PWM in order to change pressure ramping speed according to inspiration time set.



First pressure profile

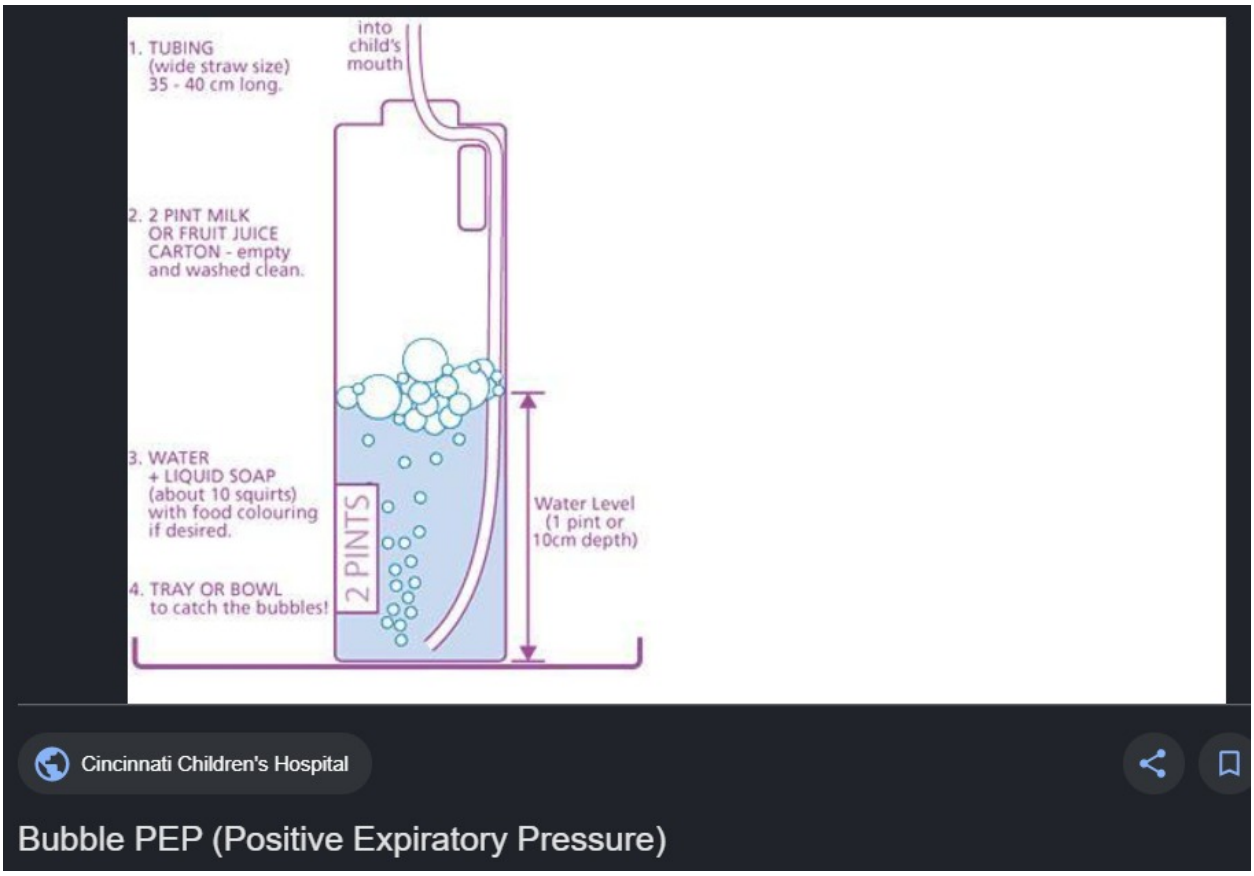
This is the first registered pressure profile. In yellow we see the arm position sensor and in blue we see the pressure sensor signal. The high peak – plateau delta is due to we were using a baloon and there is a negative resistance area on pressure using these devices. The pressure went down to zero because there was no PEEP valve in this setup.

Maximum inspiration pressure is programmed by Overpressure threshold pot. If this threshold is set, eAR stops motor from pushing. Results are lower than 1 cm H2O.

External watchdog must be disconnected in order to reprogram the Arduino MCU. eAR hardware provides that if watchdog is disconnected the software can recognize it.

FiO2 was excluded from eAR scope because we could assemble a single FiO2 central for supplying enriched air for several eARs so we've decided to start a new independent design.

There is no Peep Control, it must be handled externally. We've found this inclusive alternative from Cincinnati Children Hospital that is called a Bubble Peep. You can control Peep pressure moving the tube up and down and even start decontamination phase if you put soap on the water. It's well known that soap adheres to Covid19.



Status as April 17, 2020:

- Implemented all functions necessary for forced mode ventilation, using 18% of MCU flash footprint and 31% of Ram footprint.
- Assisted mode is a work in progress

8. eAR GUI

Status as April 17, 2020:

- Under development, WIP

9. eAR Design – Preliminary Results

Preliminary results of eAR system:

- Preliminary BOM under USD 90, including ECU electronics, mechanical compressor arm and pressure sensor.

- Airway pressure measurement resolution and precision: 1 cm H2O

- Overpressure threshold resolution and precision: 1 cm H2O

- Airway pressure controlled cutoff: less than 1 cm H2O over programmed threshold

- BPM timings errors are unnoticeable. Range: 8-32 BPM

- Tidal Volume: 200-800ml range, 2 ml resolution, undetermined precision, expected repeatability 5%.

- Plateau Hold Time: 150, 250, 500ms and 1s, switch selectable.

- I/E ratio: 1:1, 1:2, 1:3 and 1:4, switch selectable.

- Power consumption: less than 10W@12V per eAR system

- Motor PWM range: 0-100%, 1% resolution.

- FiO2 is externally set. A single unit can feed several eAR systems. In a severe crisis scenario we can run several systems using a single O2 pressurized bottle and a 12V battery with a charger.

10. eAR Project – Partial list of the Design Team – Thanks

Project Contributors, as they joned this project
- Joannes Berque (FR): first design idea.
- Celso Monteiro: eletronics and firmware design, system conception and documentation.
- Diego Padilha and Lucas Pianizzer: electronics and mechanical design+prototyping.
- Marcelo Pires: electronic CAD, schematics capture & PCB Design.
- Nicolai Rutkevich: mechanical concepts.
- Bruno Afogliatto: decontamination concepts.
- Gustavo Ortenzi: motor drive alteratives.
- Suzuki: system concepts.
- Rodrigo Azevedo: organized Github Project.

- Douglas Esteves: interface with other projects.
- João Cardoso and Leonardo Afonso: Kicad Capture for Open Hardware.
- Andre Novelli e Victor Acioly: AMBU bags replacement.
- Flevio Alves: firmware design
- David Loos: electronics assembly

References:

- MIT E-Vent: human requirements and system conception ideas.
- MHRA (UK) Rapidly Manufactured Ventilator System

Healthcare professionals involved:
- Angelica Mammana M.D. helped stablishing specs and performed crytical analysis.
- Fernando Machado M.D. was the first contributing doctor.
- Lucio Flavio M.D.
- Ana Teresa Gama.

Project supporters:
- Fabio Knihs.
- Norberto Tomio.
- Gabriel Casagrande.
- Ivandro Ceccato.

Thanks to my family, special thanks to my dear wife Tissa and my children Joana, Gustavo and Leonardo, for their unlimited support and to all the people that prayed for this project, and God that inspired and guided me all the time.

Celso Ken Mori Monteiro, April 17, 2020.