Code Life Ventilator Challenge

Design Notes

Emergency Mechanical Ventilator

Team:

Nicolas Mouret (Mechanical Design, System integration)

Evgeny Kirshin (Electronics, Firmware/Software Design, System Integration)

Stephane Delisle (Medical Expert Consultant)

Martin Girard (Medical Expert Consultant)

Michel Troli (Mechanical System Design Consultant)

Support:

Igor Kilunov (Embedded Firmware Development – User Interface)

Mikhail Dimitras (3D printing support)

Last update: April 1, 2020

Contents

Design Features

- Two implementation options: 1) with integrated LCD as user interface; 2) With external PC as user interface;
- Works in Assisted Control mode: ventilation triggered by either minimum Respirations per Minute (RPM) setting or by patient inhale;
- Display of pressure graph, output of alarms - visual and sound (User Interface/speakers);
- Calculation and display of peak airway pressure;
- Adjustment of RPM (User Interface);
- Selection of two Inspiratory Flow rates - 44 L/min or 57 L/min (User Interface);
- System Setup and Calibration (User Interface);
- Tidal volume adjustment (mechanical)
- PEEP pressure adjustment (mechanical valve);
- System is designed with component availability in mind.

Safety features:

- Emergency Stop button (User Interface, emergency shutdown relay);
- Emergency Stop on Maximum Pressure: both with pressure sensor and with safety valve;
- Uninterrupted power supply system (battery charged by power supply; system continues working when either power supply or battery fail);
- Two flow sensors for flow rate monitoring;
- Commands from User Interface are sent with acknowledgement and repetition (if command is lost).

Device Concept

Figure below illustrates the proposed system design.

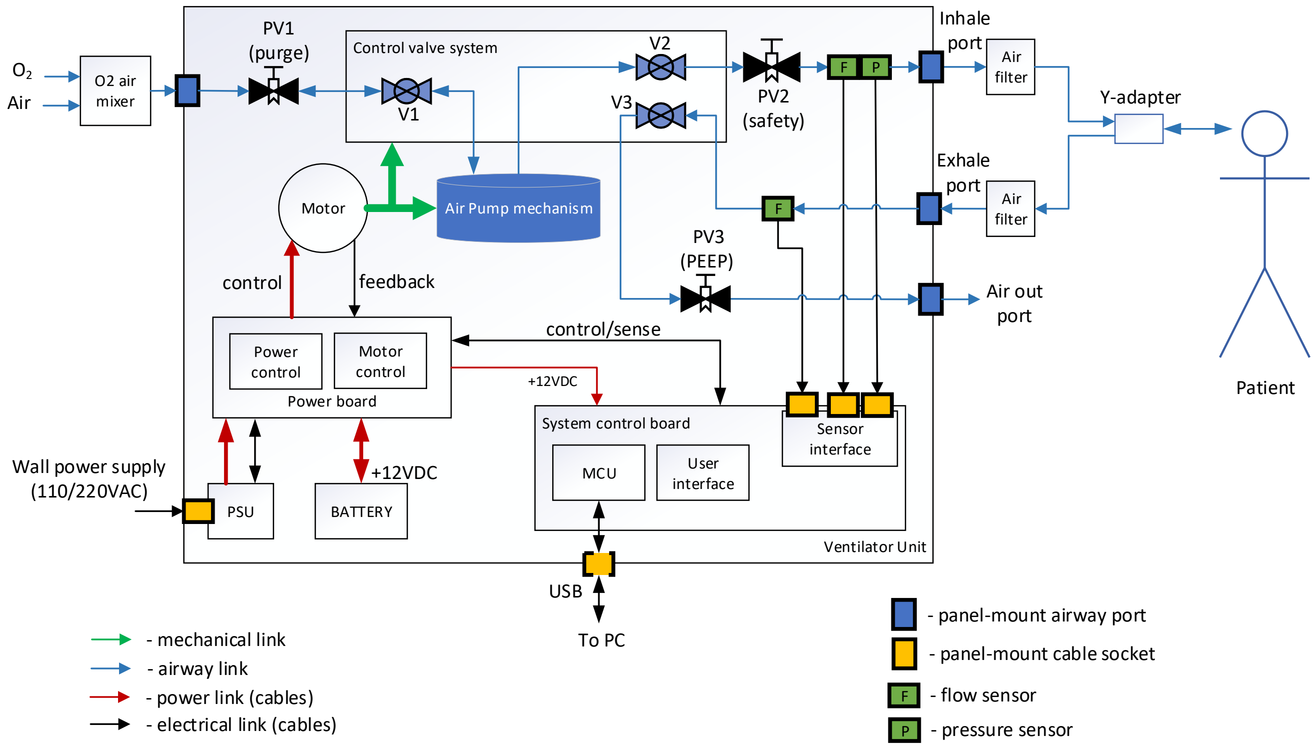


Figure 1 - Ventilator unit system diagram

The system consists of the following sub-systems:

- Unit enclosure
- Power board: provides 12V DC uninterrupted power to the system, as well as drives the main motor under the control of the system board.
- System control board: includes microcontroller board with optional LCD screen and USB port, sensor interface and user interface.
- Mechanical air inflation mechanism, consisting air pump and airway valve mechanism.

External components, such as O2 air mixer, air filters, Y-adapter and hoses are out of scope of our design.

Functional description

With one turn of the motor air pump creates pressure inside the compartment. By mechanically opening and closing air valves V1, V2, V3 at the correct phase of the motor turn, air flows from external O2 air mixer through valve V1 and V2 into the patient and through valve V3 outside the patient.

Valves PV1, PV2, PV3 are valves that open or close mechanically depending on the pressure in the system. Such valves can be mechanically adjusted by operator.

PV1 – helps purging residual air after valve V1 closes on inhale cycle. Mechanical adjustment in the control of V1 valve sets the amount of air entering the patient at inhale phase.

PV2 is a safety valve. It opens when pressure in the system exceeds 40 cmH2O.

PV3 is a valve to keep the PEEP pressure in the airway. It opens when air pressure exceeds certain limit (e.g. 5 cmH2O) allowing air to go out of the patient; it closes when the pressure becomes lower, keeping required PEEP level.

Pressure sensor together with two flow sensors (inhale and exhale airway paths) allow microcontroller control the system appropriately.

Unit Enclosure

Ventilator unit enclosure is a box with appropriate dimensions to fit all the components. The enclosure can be fabricated from steel and painted.

As a part of this project, a concept CAD model has been developed – see images below (some elements are missing: some hose ports, sensor connectors, power connector and USB port).

Internals:

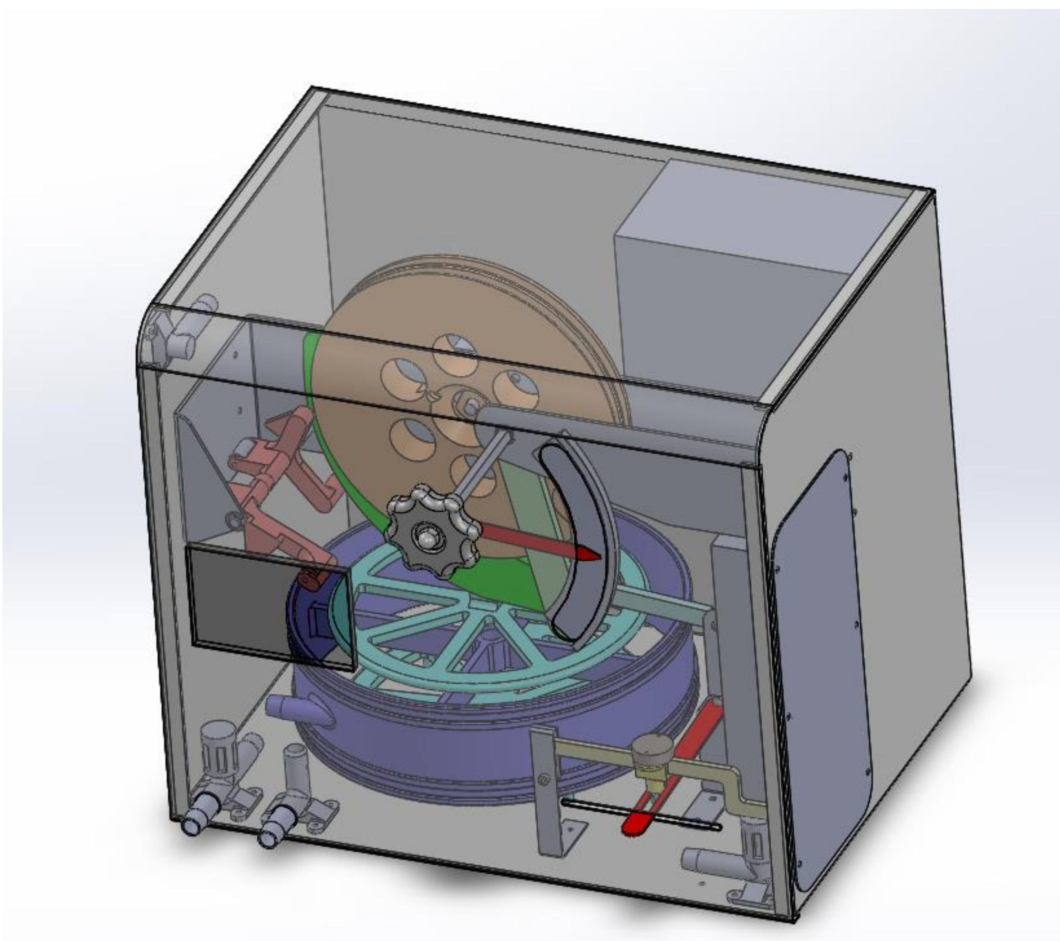


Figure 2 - Internals of the ventilator unit

Power

Ventilator unit has to be operational during 180 minutes without mains power supply, which means internal battery is a must.

The system should be continuously powered from the battery to enable uninterrupted work in case of external power loss. The battery has to continuously charge from external power supply when it is ready (both during standby and when the machine is operational).

We propose using automotive batteries, similar to shown below since they are easy to find in stores.

As a power supply unit, it is convenient to use computer power supplies – they offer high current for charging the battery, as well as powering the system when battery is not operational (e.g. during battery change process).

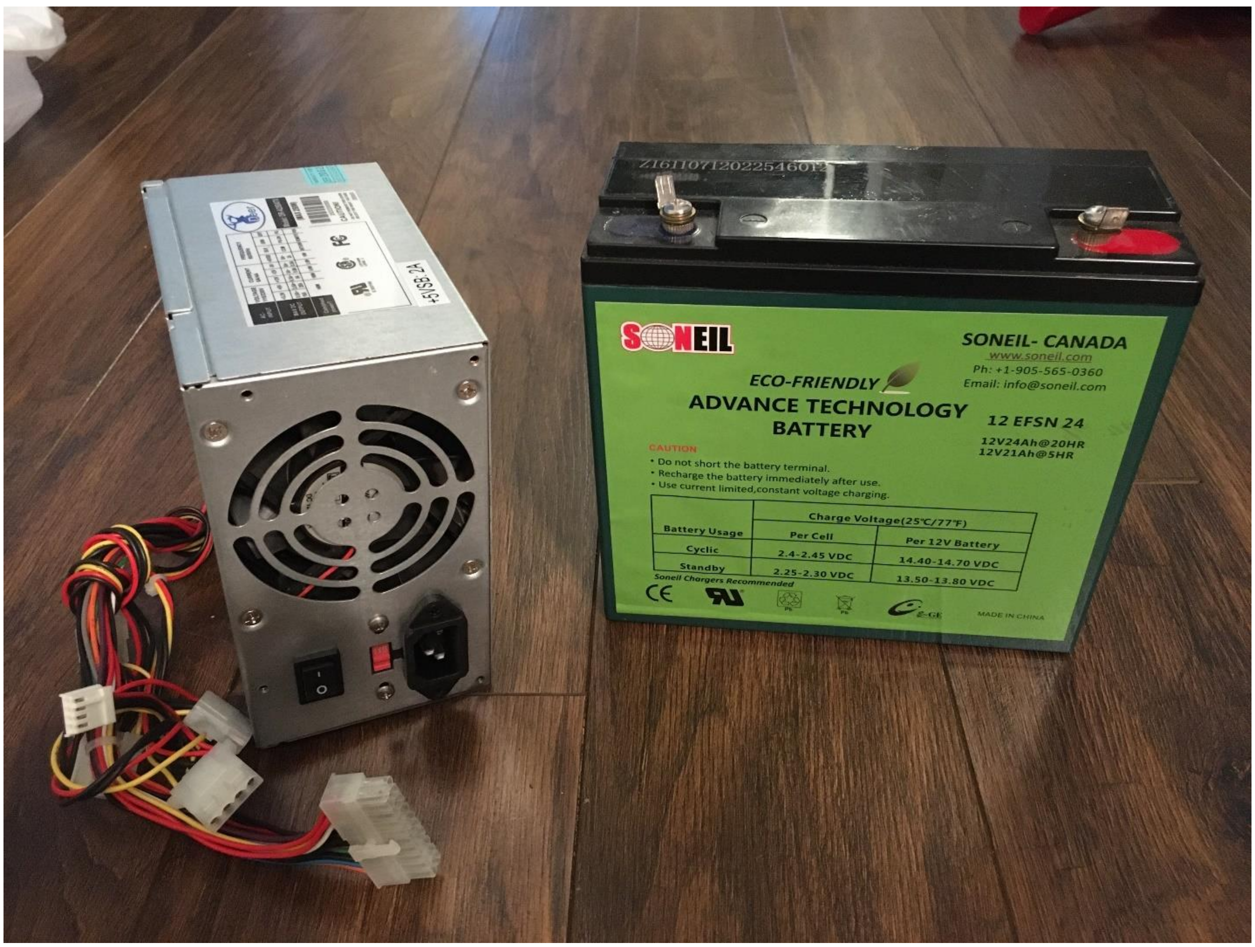


Figure 3 - Computer power supply and automotive 12V battery as main power components

Power Board

The diagram of the prototype power board is shown below:

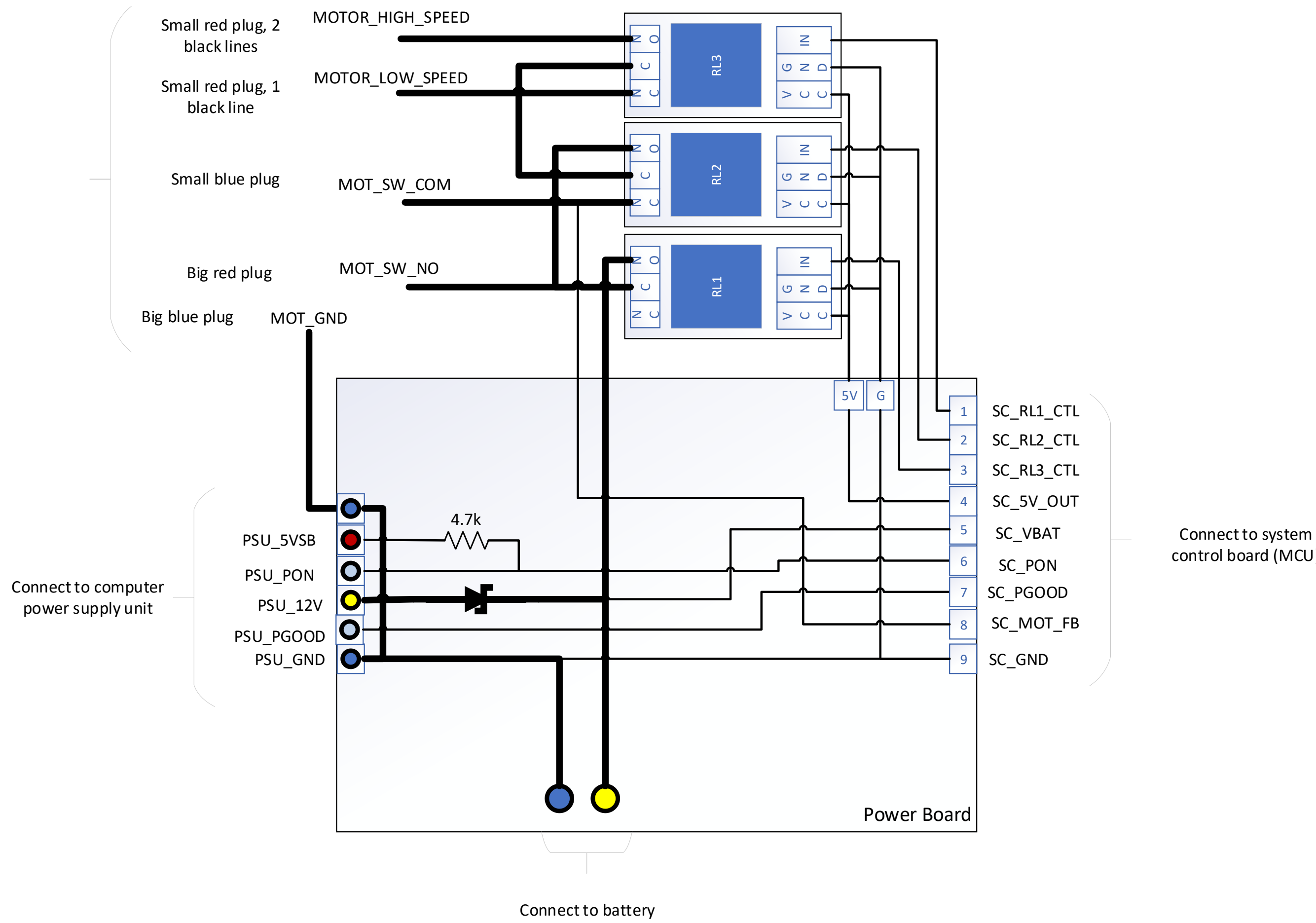


Figure 4 - Ventilator Power Board Prototype connectivity diagram

The motor is connected to the system through a system of three relays:

- RL1 – enables/disables power to the motor (used for emergency shutdown)
- RL2 – used for driving the wiper motor (see diagram below)
- RL3 – used for switching motor LOW/HIGH speed (this results in slower/faster respiratory flow, e.g. in our design these numbers are 44 L/min and 57L/min).

Here is an example of the relays that were used in the design:

Figure 5 - Example of used relays

These are available for ordering on Amazon:

https://www.amazon.ca/gp/product/B00ZR3B252/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1

Other options:

https://www.digikey.ca/product-detail/en/seeed-technology-co-ltd/103020012/1597-1090-ND/5482568

https://www.digikey.ca/product-detail/en/dfrobot/DFR0251/1738-1175-ND/6588596

Some elements, such as capacitors and protection fuse, are not shown on the diagram.

Such connectivity enables the system to work either when external power is lost or when battery needs to be urgently changed. The microcontroller monitors both battery voltage and the presence of external power supply and reacts with warnings.

Figure below shows how power board has been implemented for our prototype:

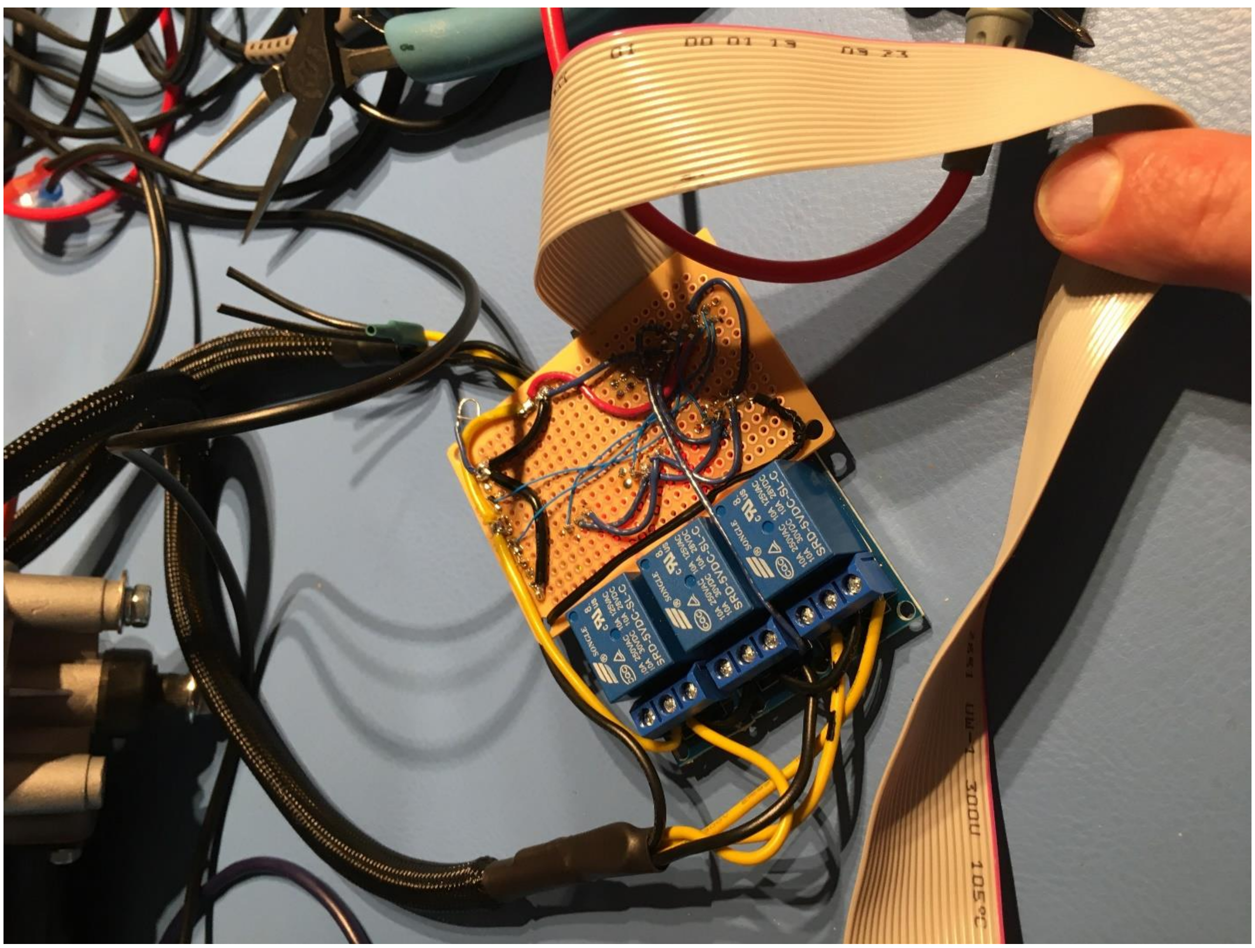


Figure 6 – Implemented prototype of the power board

Schematic of a power board is included below (and attached in a separate PDF file). A PCB board can be quickly designed and fabricated for production versions of the Ventilator.

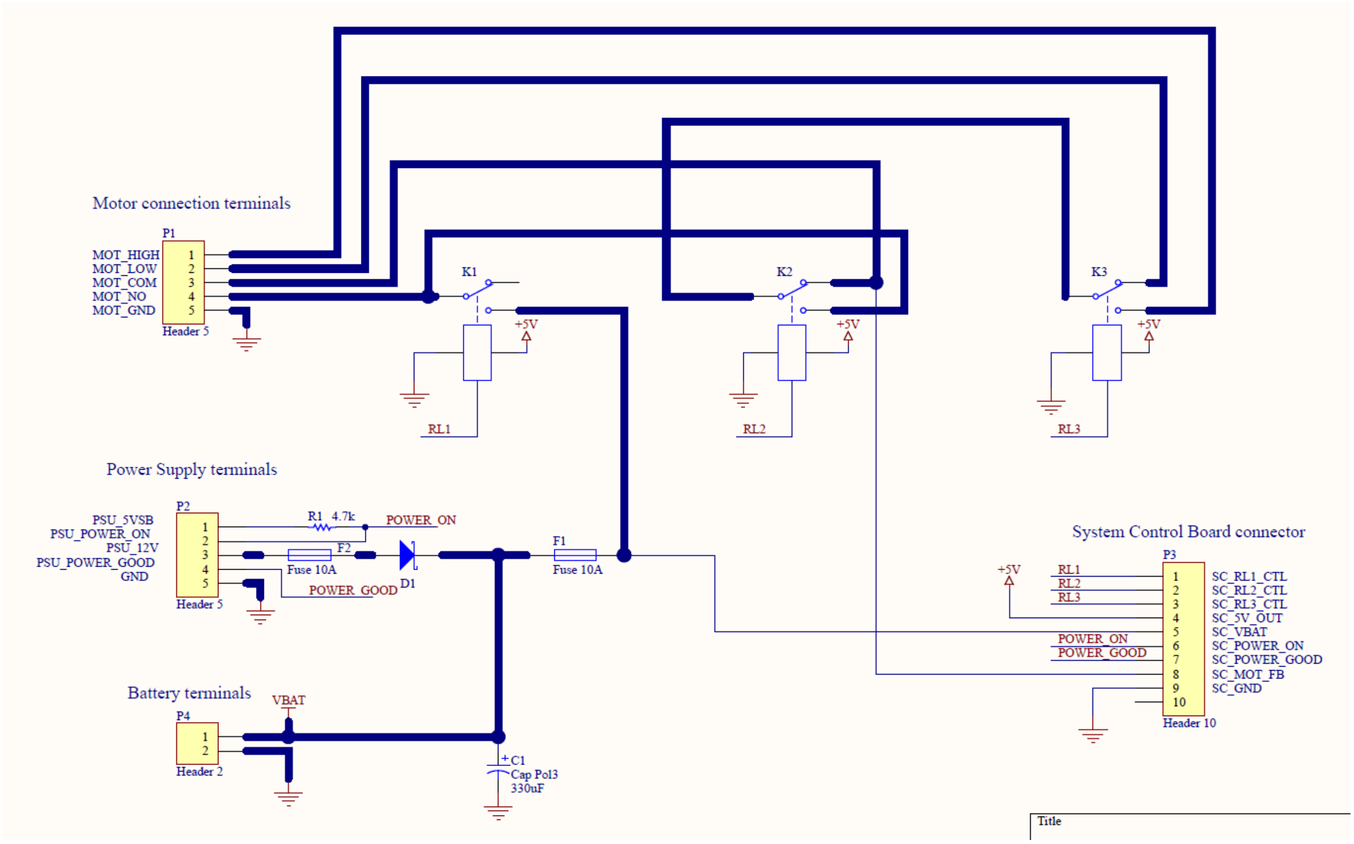


Figure 7 - Power Board Schematic (refer to separate PDF file for clear view)

Motor

Car wiper motor has been chosen as an available option:

https://partsavatar.ca/1989-dodge-lancer-new-wiper-motor-cardone-industries-85-387?gclid=Cj0KCQjw6_vzBRCIARIsAOs54z7syqEokNsfGXk8lTUL3MhSS556W7wEC_W9yQkqYQMvGfDrCNDB410aAsBmEALw_wcB

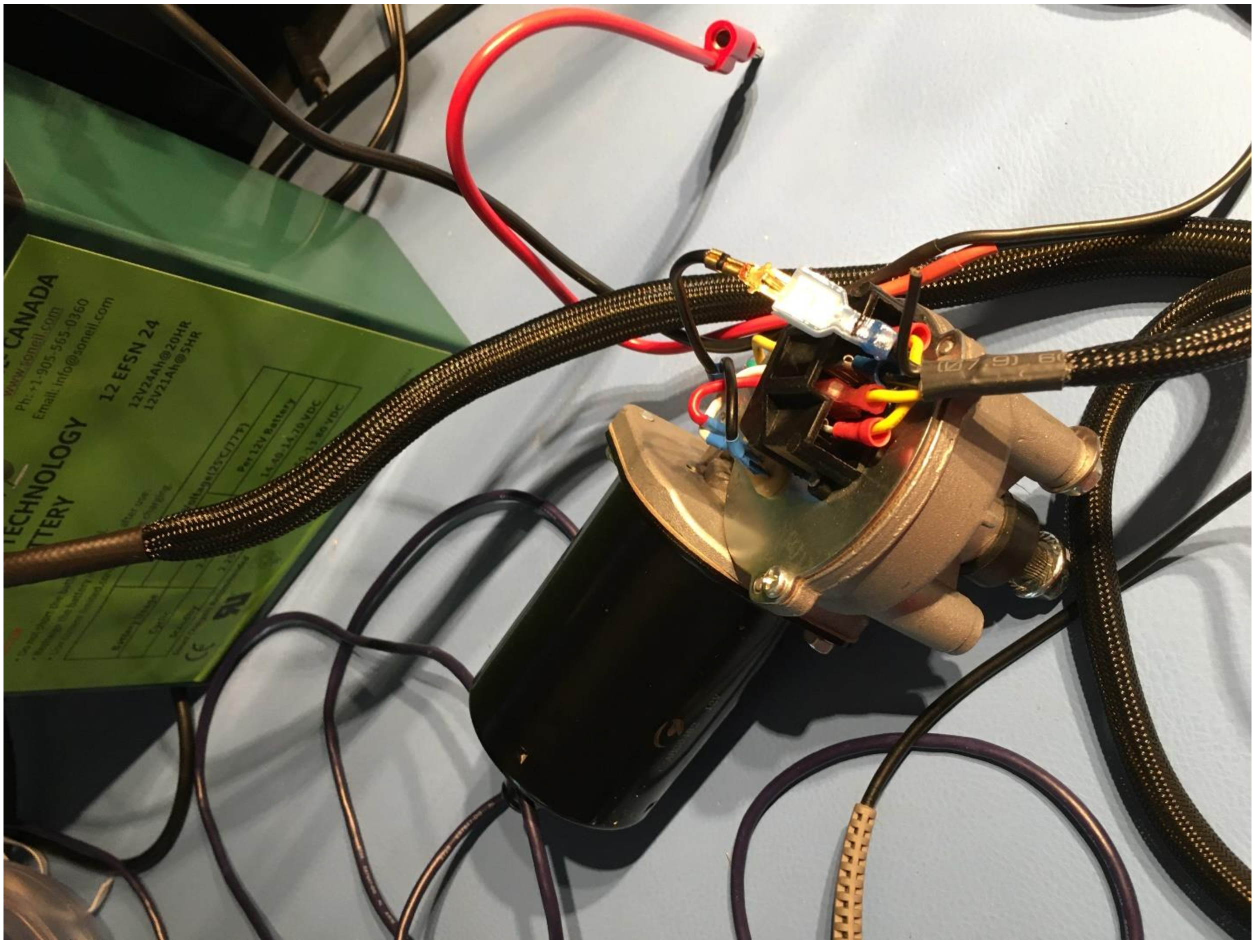


Figure 8 - Car windshield wiper motor used in the prototype

It can be any 12V wiper motor with equivalent functionality.

Such motor offers good torque and will be able to drive the pump.

Here is a connectivity diagram to control such motor:

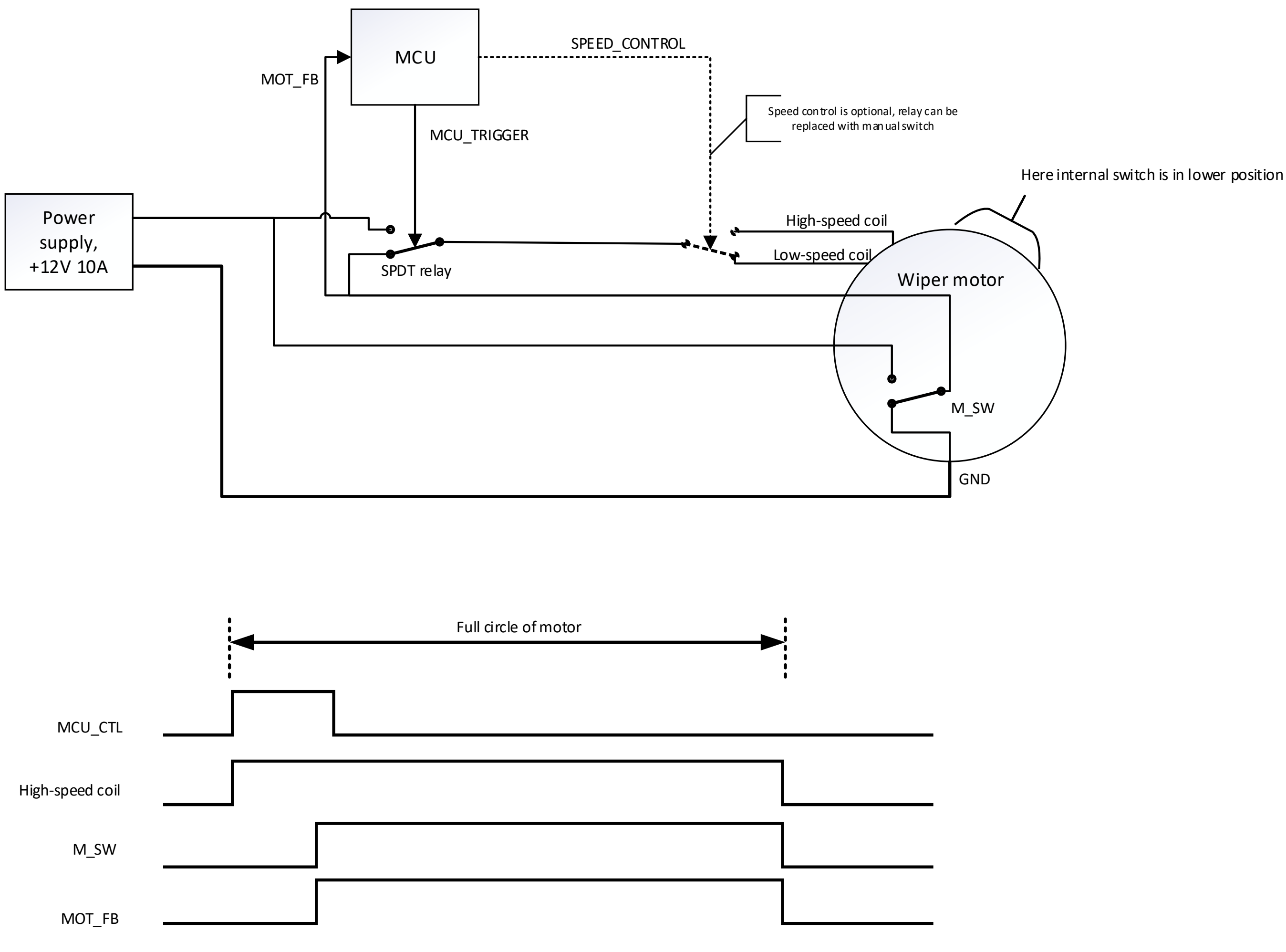


Figure 9 - Motor control diagram

System Control Board

System controller is a board or a set of connected boards providing the following functions:

- Reading sensor and processing the signals.
- Control motor based on sensor readings and commands/settings from operator.
- Provide user interface in order to:
  - Switch device operating mode
  - Display essential information (pressure graph, max. pressure, current settings, etc.)
  - Output alarms (visual and sound)
- Monitor and control power system

There are many ways of implementing system controller.

We will provide two options:

1. Standalone system with GUI
   - Only ventilator unit is required
   - The unit has an LCD screen with touch screen
   - The LCD is used to visualize pressure graph, system parameters and alarms.
   - Buzzer is used to make the alarms audible.
2. System with extended user interface
   - A PC (laptop or desktop) with Windows operating system is required in addition to the ventilator unit.
   - Ventilator unit connects to a USB port of the PC.
   - Windows application with GUI is designed to control the unit, visualize information and output alarms (speakers or PC buzzer is required).

Implementation of system controller

Option with integrated LCD screen can be implemented using a development board with an LCD. The advantage of such approach is the use of off-the-shelf components and their availability

We provide two options as examples: STM32F429-DISCOVERY board and STM32F746G-DISCO.

STM32F429-DISCOVERY:

Figure 10 - STM32F429-DISCOVERY as a platform for a system controller with LCD touch screen

The board is orderable at Digikey: https://www.digikey.ca/product-detail/en/stmicroelectronics/STM32F429I-DISC1/497-16140-ND/5731713

Amazon: https://www.amazon.ca/32F429IDISCOVERY-STM32F429I-DISCO-Discovery-STM32F429ZIT6-Development/dp/B01EJ3ASAK/ref=sr_1_2?keywords=stm32f429-discovery&qid=1585415446&sr=8-2

Mouser: https://www.mouser.ca/ProductDetail/STMicroelectronics/STM32F429I-DISC1?qs=sGAEpiMZZMvJkDqKJH80dBW9Ucpez89T%252ByheH8hFi38%3D

The board does not have integrated 12V->5V regulator, so external DC-DC converter will be required. LM317 is an option for this application: https://www.digikey.ca/product-detail/en/texas-instruments/LM317DCYR/296-12602-1-ND/443738

- The firmware for LCD user interface is under development.
- The control/sensing firmware can be easily ported from STM32-NUCLEO64 platform.

Another option would be to use a development board with larger LCD touch screen - **STM32F746G-DISCO**.

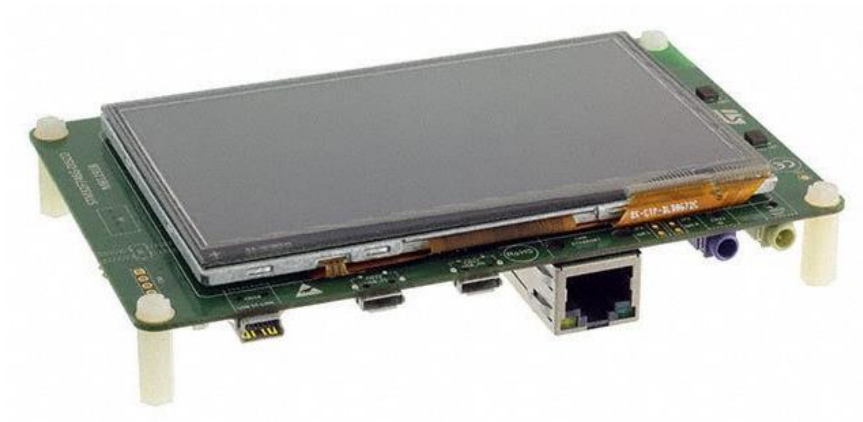


Figure 11 - STM32F746G-DISCO board

The board is available for ordering from popular distributors:

https://www.digikey.ca/product-detail/en/stmicroelectronics/STM32F746G-DISCO/497-15680-5-ND/5267791

Version without integrated LCD can be implemented based on a relatively simple development boards. One example is **STM32-NUCLEO64** boards – see below.

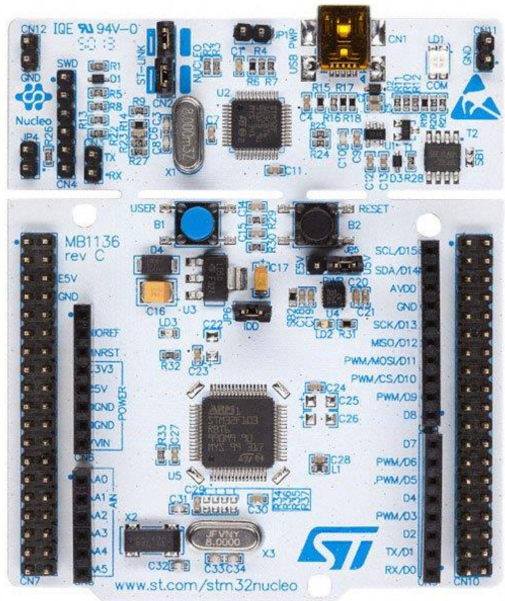


Figure 12 - STM32-NUCLEO64 development board as a platform for a ventilator system controller

These boards have several advantages:

- Low-cost: 17-25$CAD on Digikey, depending on installed microcontroller.
- Available on Digikey, Mouser, Amazon, etc.
- Simple enough for developers (firmware can be easily ported from one MCU to another by experienced developers, instructions to be provided).
- The board has integrated programmer/debugger (STLink V2) -> no need external programmer for flashing firmware.
- The board has integrated regulators -> can be powered directly from 12V supply with 5V and 3.3V available from the onboard regulators to power external sensors.
- STLink V2 debugger provides Virtual COM port functionality, which allows connecting this board to a PCB over USB and exchange data.

The board is available for ordering:

Digikey: https://www.digikey.ca/products/en/development-boards-kits-programmers/evaluation-boards-embedded-mcu-dsp/786?FV=1742%7C380852%2C-8%7C786&quantity=0&ColumnSort=0&page=1&k=NUCLEO&pageSize=25&pkeyword=NUCLEO

Mouser: https://www.mouser.ca/ProductDetail/STMicroelectronics/NUCLEO-L010RB?qs=%2Fha2pyFaduhrZTFttm%252BiYif0TrImutaxfqJmldbHgzTfetnr%252Bdutxg%3D%3D

Amazon: https://www.amazon.ca/STM32-Nucleo-Development-STM32F446RE-NUCLEO-F446RE/dp/B01I8XLEM8/ref=sr_1_1?crid=2PBBKFRY5RD67&keywords=stm32+nucleo-64&qid=1585415484&sprefix=stm32-nucleo%2Caps%2C241&sr=8-1

A schematic of the system control board is shown below. The board can be assembled either using separate components installed on adapter boards and perf board used to install all necessary components or a PCB can be designed and fabricated within ~2 weeks for production versions.

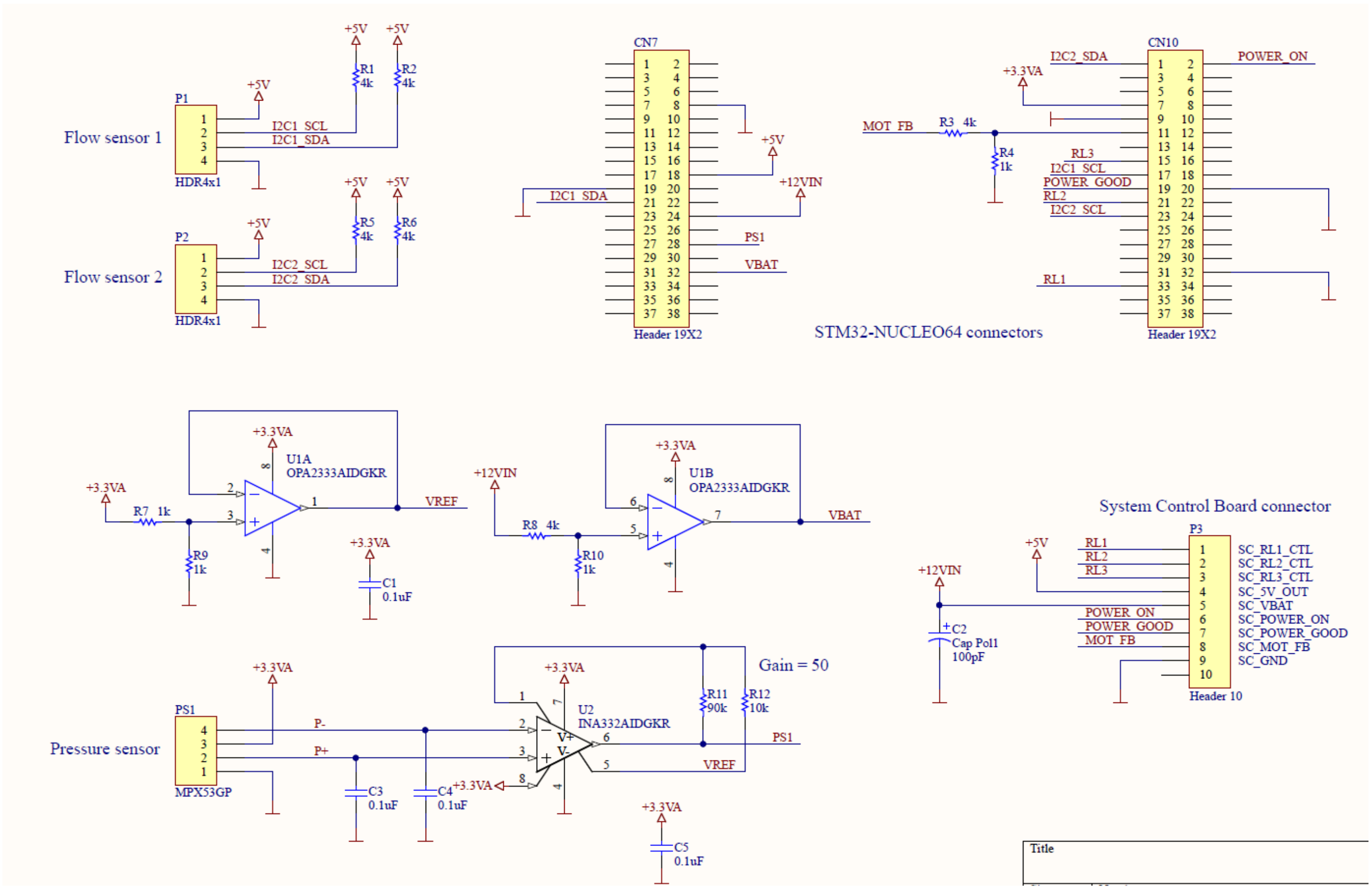


Figure 13 - System Control Board schematic (refer to separate PDF file for clear view)

Sensor interface

One pressure sensor and two flow sensors are used in this design.

Examples of a pressure sensor that can be used in the design:

https://www.digikey.ca/product-detail/en/honeywell-sensing-and-productivity-solutions/TBPDANS005PGUCV/480-5858-ND/4696898

https://www.digikey.ca/product-detail/en/honeywell-sensing-and-productivity-solutions/TBPDANS015PGUCV/480-5852-ND/4696892

https://www.digikey.ca/product-detail/en/nxp-usa-inc/MPX53GP/MPX53GP-ND/684666

The pressure sensors are connected to the microcontroller's ADC through an instrumentation amplifier. For example, INA332 can be used for this purpose (refer to schematic):

https://www.digikey.ca/product-detail/en/texas-instruments/INA332AIDGKR/296-41265-1-ND/5222729

Here is an example of a flow sensor that can be used in the design:

https://www.digikey.ca/en/products/detail/honeywell-sensing-and-productivity-solutions/HAFUHH0010L4AXT/4696827

Flow sensors are connected to the microcontroller through two I2C interfaces (refer to schematic).

Firmware

The main firmware for the prototype is implemented for the STM32-NUCLEO64 board described above. The firmware controls power system and the motor, collects information from pressure sensors (and potentially from flow sensors), communicates with the PC software. This software can be easily ported to the boards with the LCD screens described above.

Source code of the firmware is developed in STM32CubeIDE environment and submitted with the design files. A programmer is not required to flash the firmware to the STM32-NUCLEO64 board since there is integrated STLinkV2 programmer/debugger on the board.

Block-diagram of the firmware is shown below.

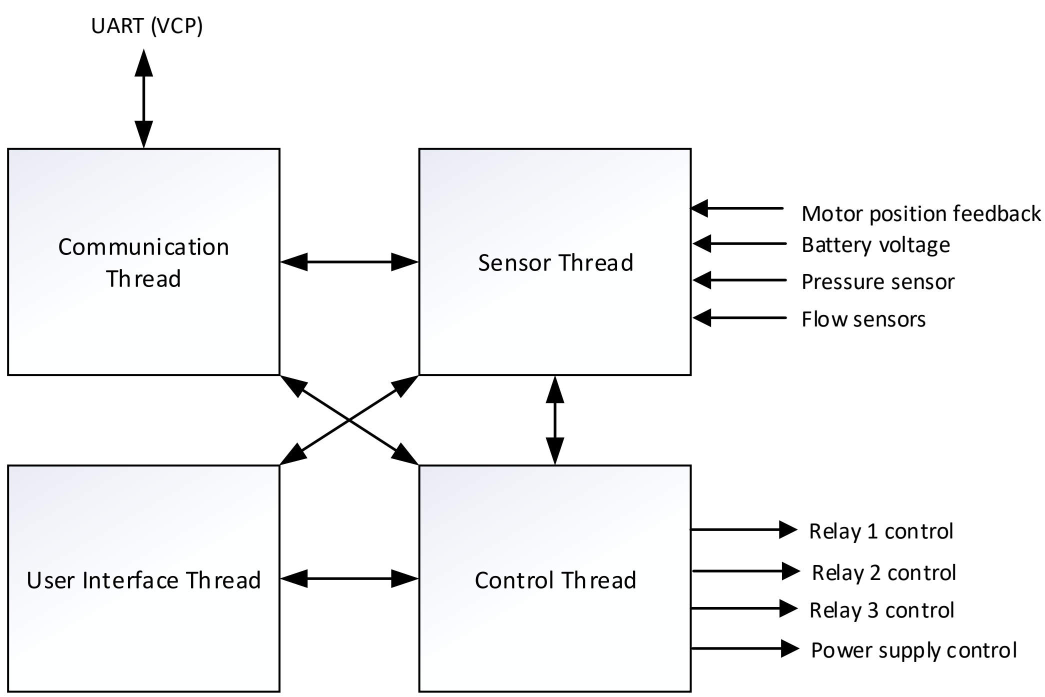


Figure 14 - Firmware block-diagram

The firmware is based on the FREERTOS operating system. There are 4 threads in the system:

- Sensor thread: acquires information from all sensors and collects all information from power system and motor feedback.
- Control thread: implements main control logic - analyzes sensor signals and controls motor and relays).
- Communication thread: receives commands from PC application; sends confirmations to the commands from PC application;
- User Interface thread: LEDs/buttons for simple implementation; LCD screen for implementation with the display integrated in the unit.

User interface

Windows Application UI

Windows application consists of three main tabs:

- Main – for regular operation
- Setup – for setting up the system at the beginning of operation
- Service – for installation and calibration

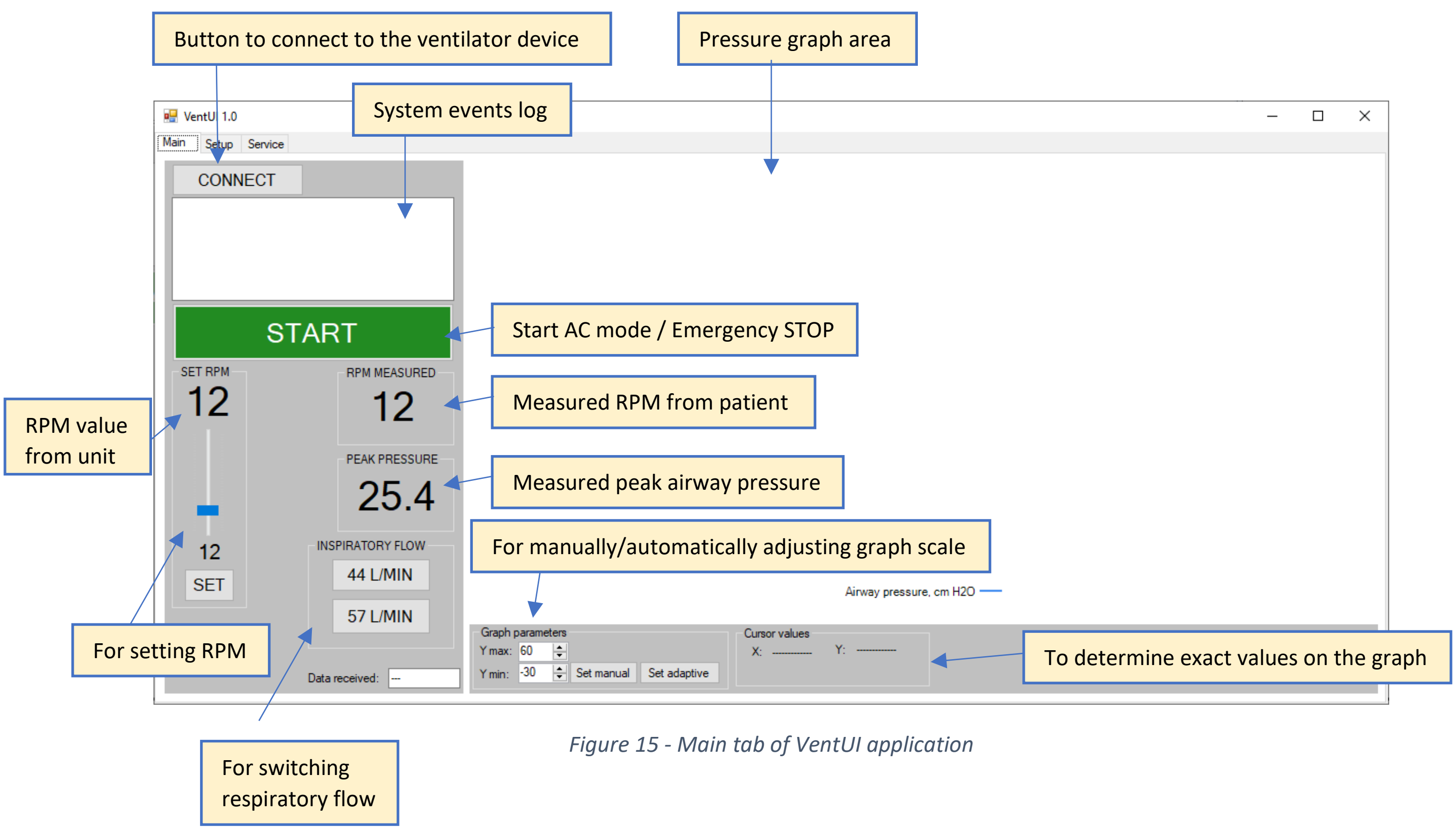


Figure 15 - Main tab of VentUI application

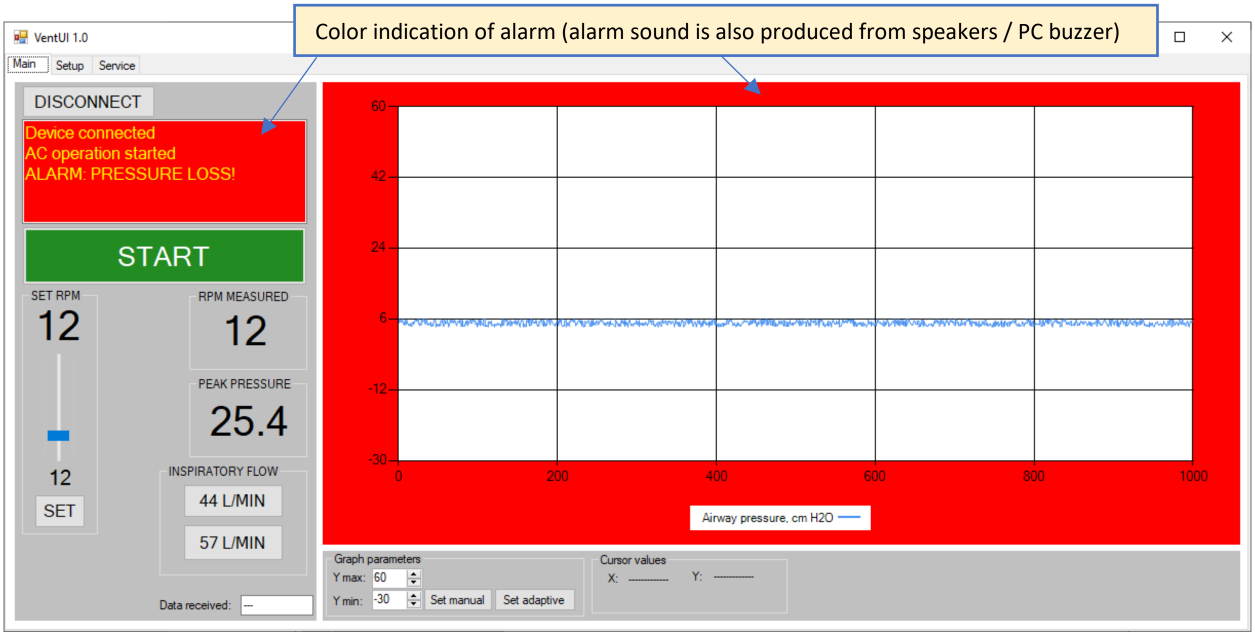


Figure 16 - User interface in case of Alarm

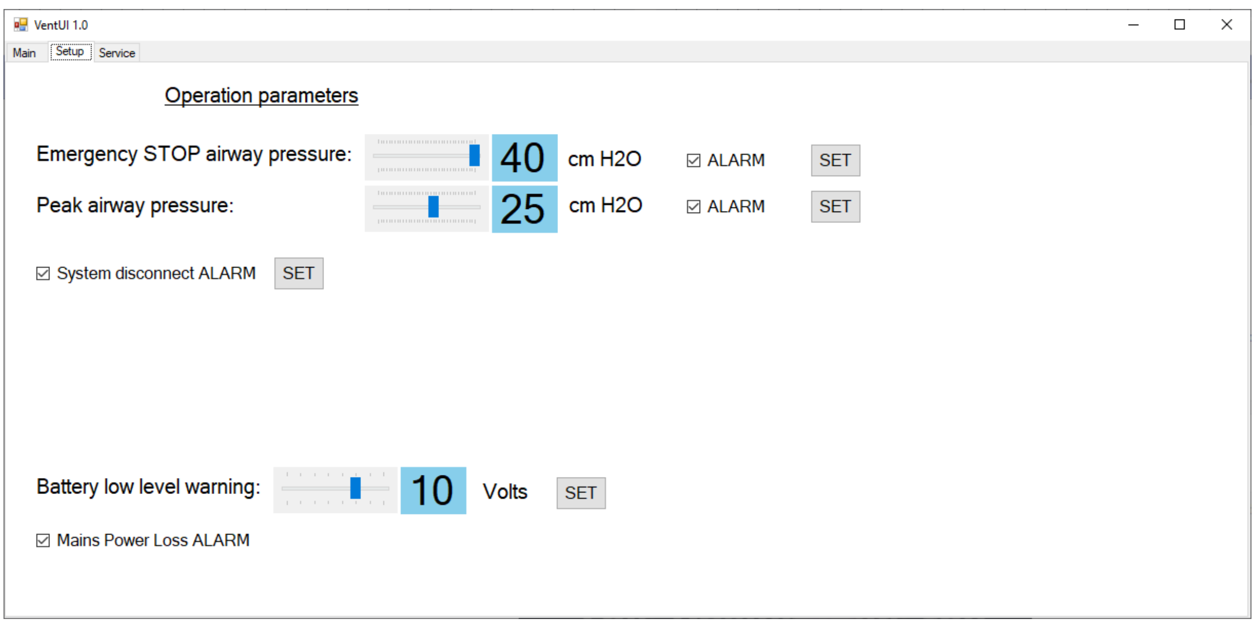


Figure 17 - Setup tab of VentUI application

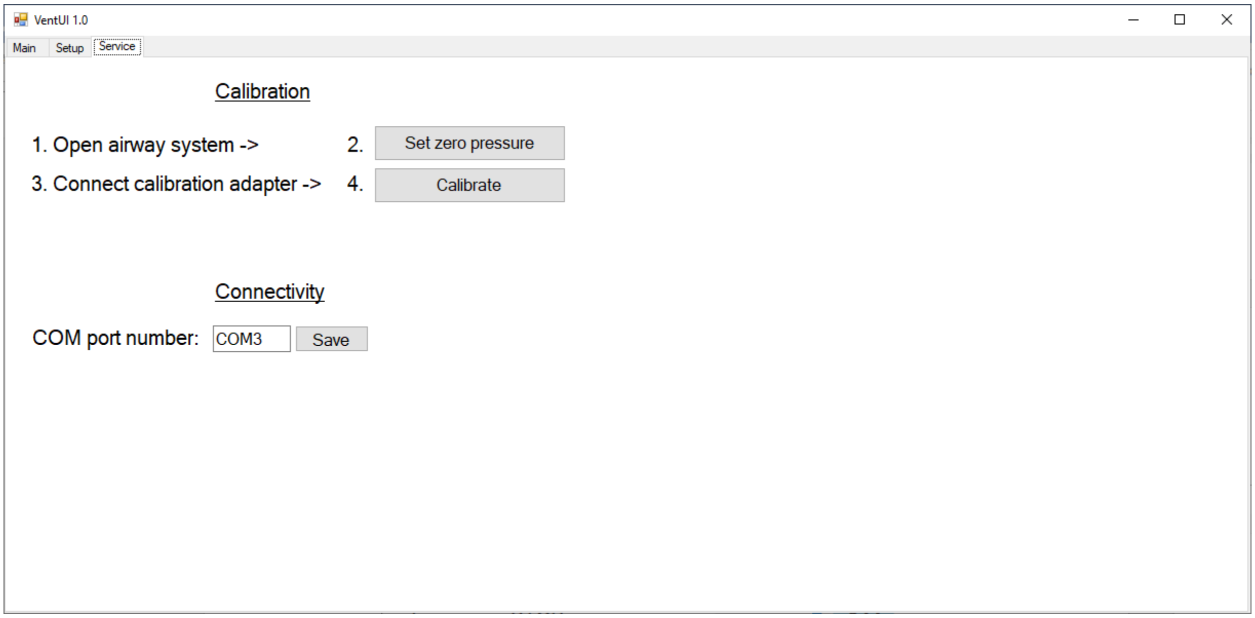


Figure 18 - Service tab of VentUI application

Windows application is developed using Microsoft Visual Studio Community Edition (free) using C# language and Windows Forms library.

LCD user interface

LCD screen UI prototype is shown below. It allows user to control the system with a touch-screen. The interface is similar to the Windows application with a subset of functions and more compact layout.

Figure 19 - User interface on integrated LCD screen

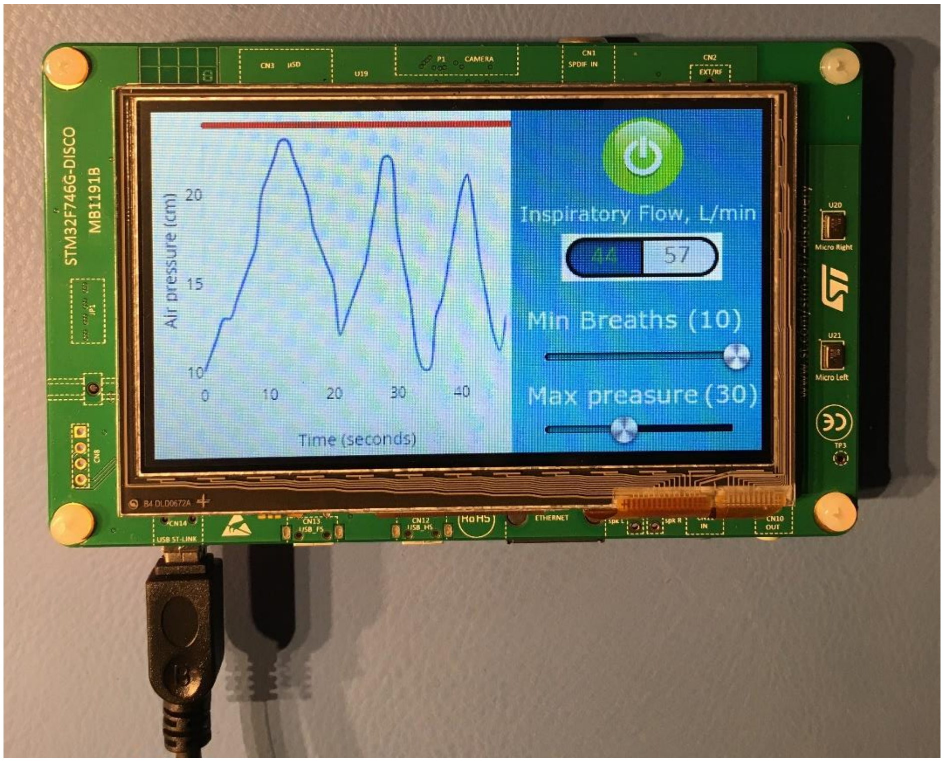


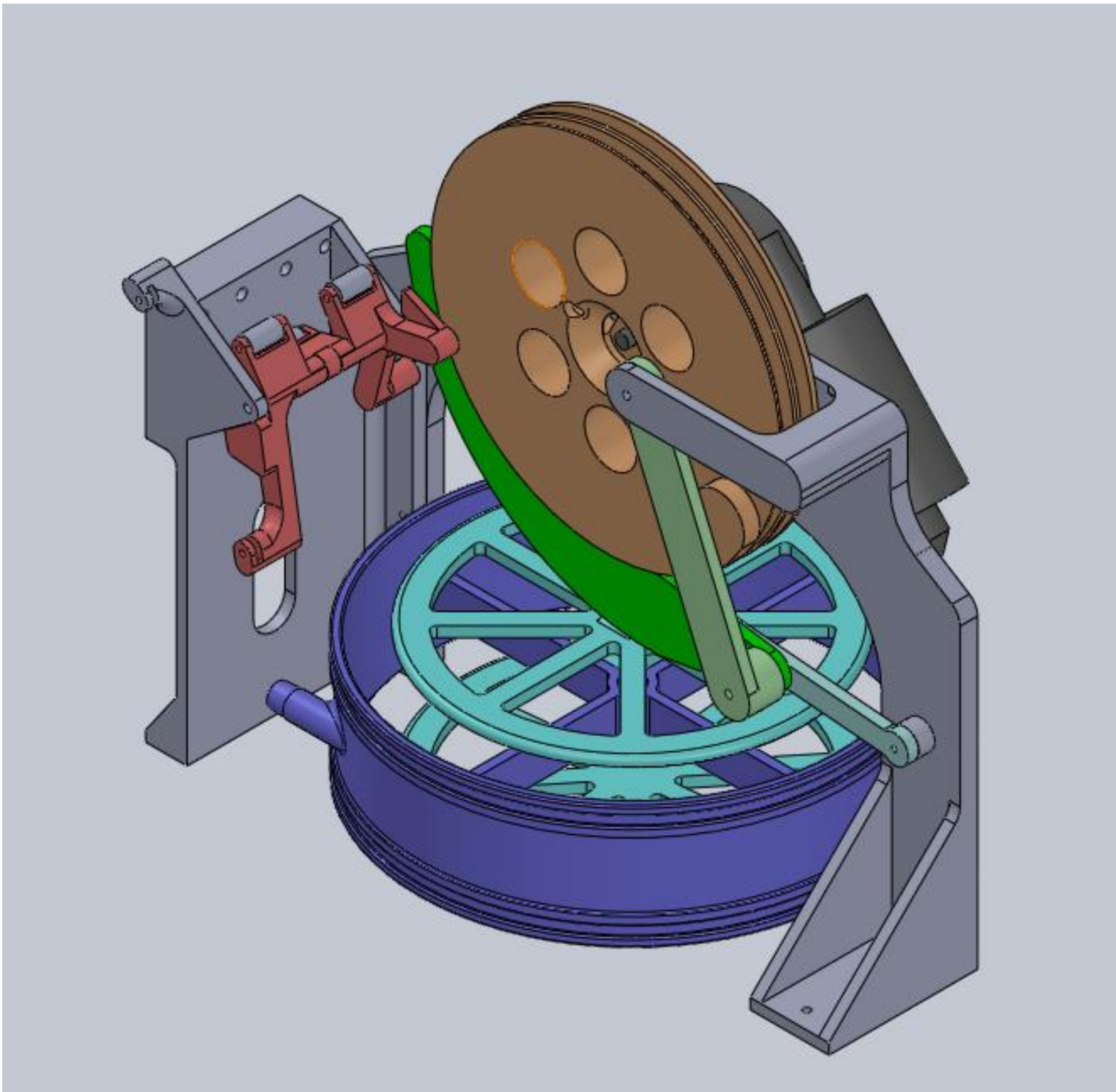
Figure 20 - LCD user interface implemented for STM32F746G-DISCO board

Mechanical System

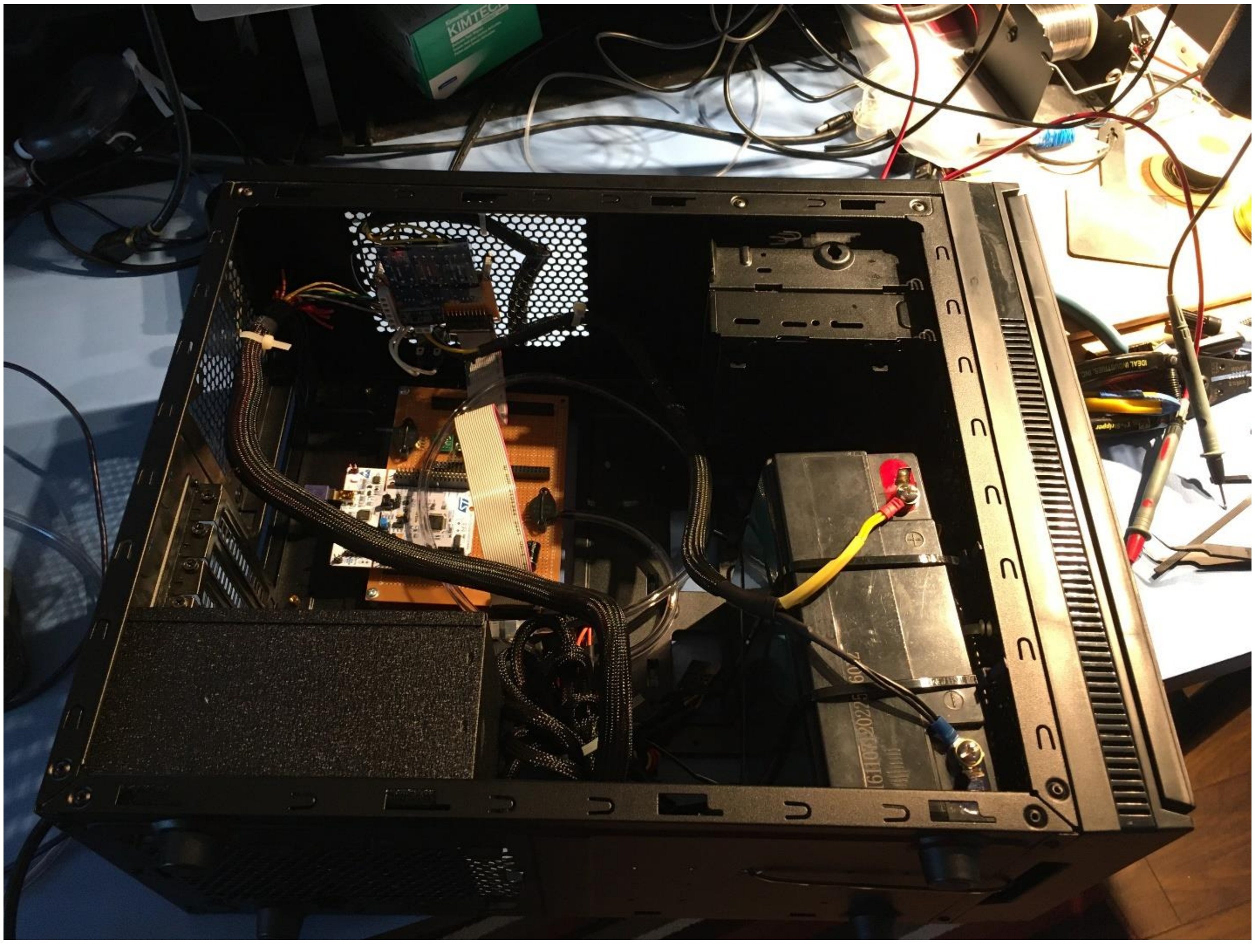
Mechanical system is described in a separate document.

Appendix

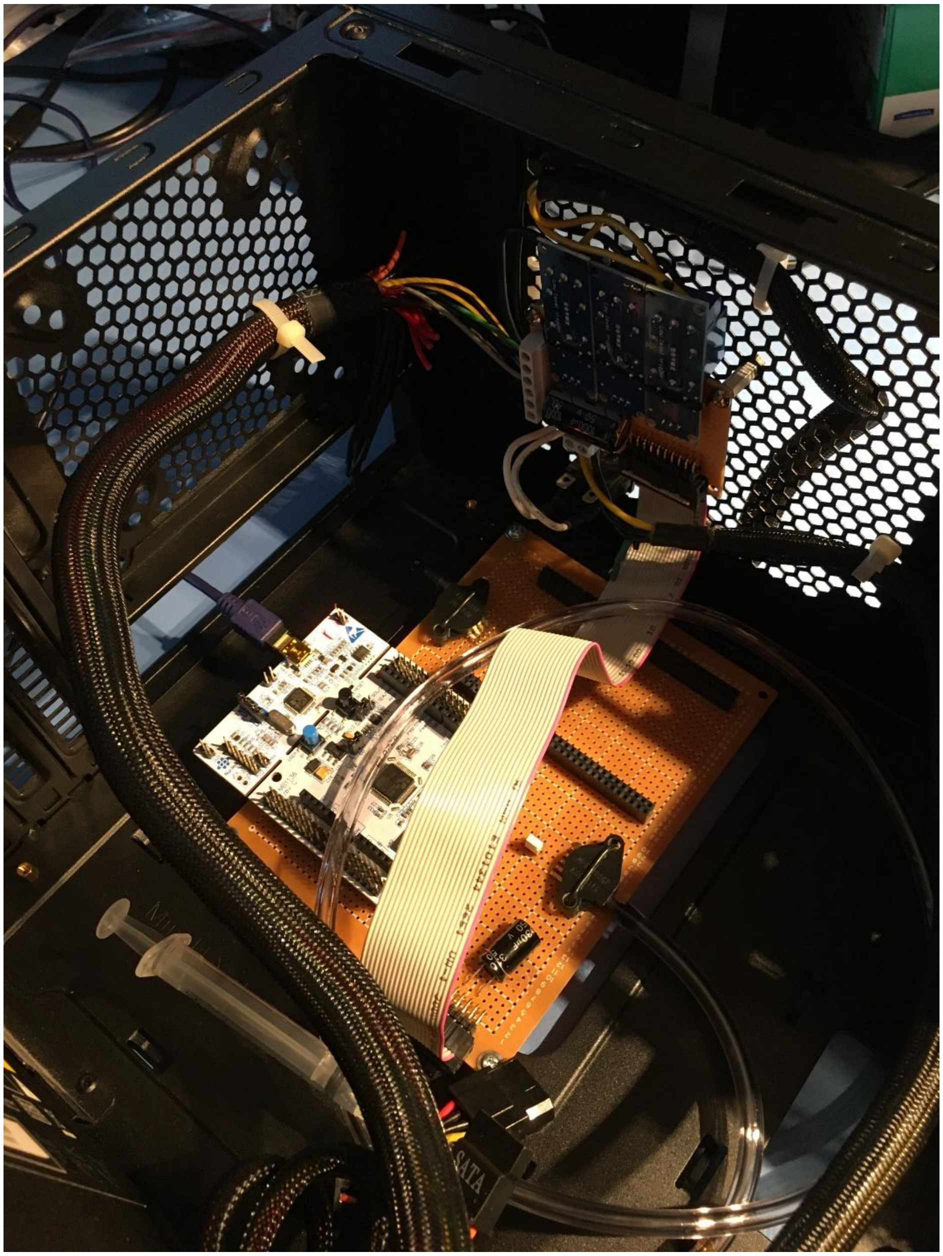
The core of Mechanical system:



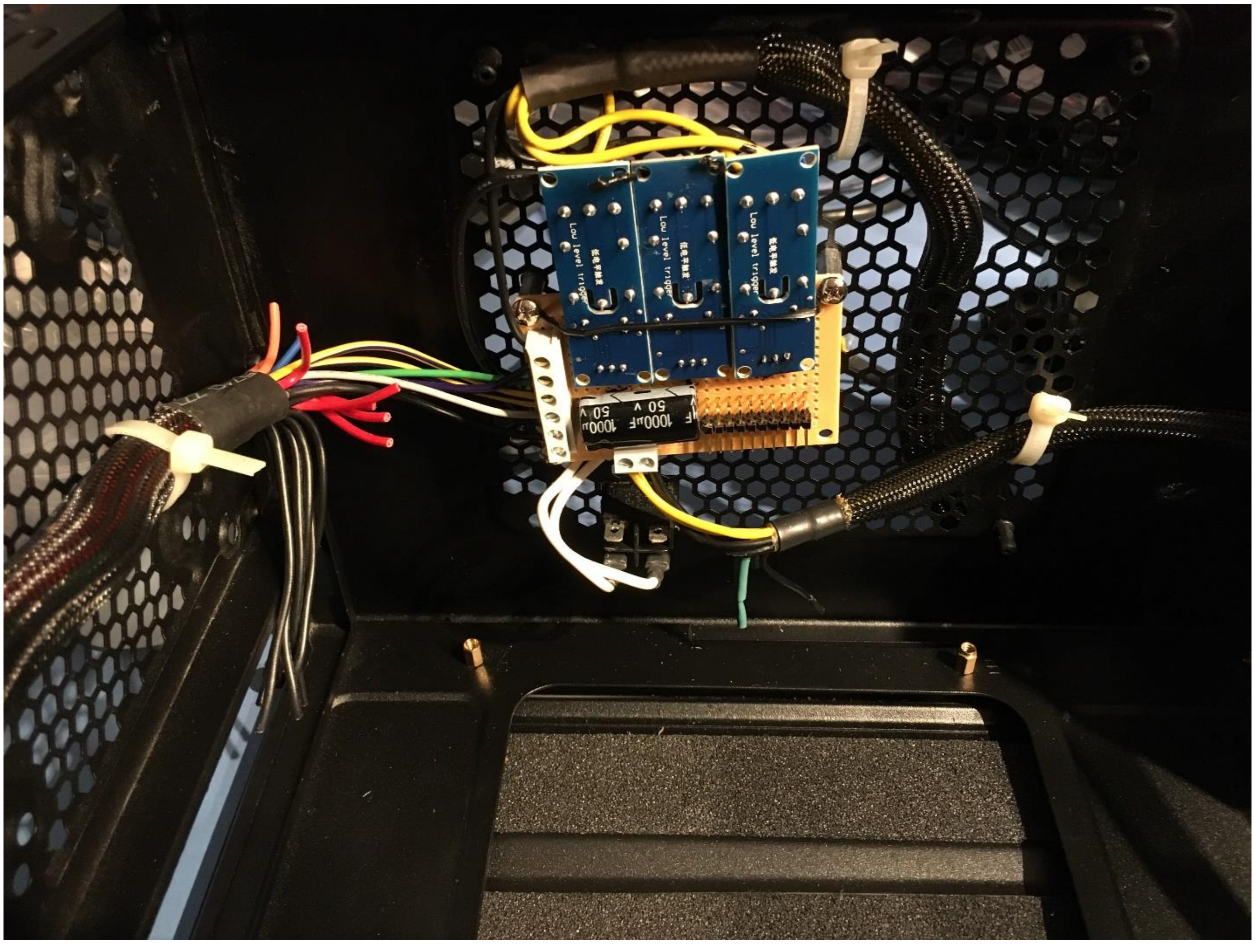
Assembled Electronics and Power system:



System Control and Power board prototypes:



Power board prototype:



Wiper motor connected to the system: