Algorithm 1: Read Sensors

Result: Reads and converts sensor values (every 50ms)
//to be added in an always executing loop;

if *sampling period has elapsed* **then**
    read pressure value;
    read flow value;
    read temp value;
    filter all value with low pass filter;
    convert all values to engineering unit;
end

Algorithm 2: Breath Cycle (Breathe-In)

Result: Sets the maximum speed and acceleration of stepper for breathe-in phase
//to be added in always executing loop;

* $breathePeriod(ms) = \frac{60{,}000}{requiredBreathePerMinute}$);

if *breathe Period has elapsed* **then**

calculate breathe in time (needs improvement);

$$breratheInTime = \frac{breathePeriod}{2}$$

calculate breathe out time (needs improvements);

$$breratheOutTime = breathePeriod - breratheInTime$$

calculate breathe in speed i.e (liters required/sec) (needs improvement);

$$breratheInSpeed = \frac{volume}{breratheInTime}$$

set maximum stepper speed as;

$$stepperMaxSpeed = (motorSpeedFor1liter/sec) * (breratheInSpeed)$$

set maximum stepper acceleration as;

$$stepperMaxAcceleration = breathInSpeed{*}breathInSpeed{*}motorAcceleration$$

move stepper to ;

$$stepperMoveTo = (volume) * (motorVolumeRatio)$$

where

$$motorVolumeRatio = (distanceMovedInOneStep) * (volumeMovedInOneStep)$$

calculate breathe out speed i.e (liters removed/sec) (needs improvement);

$$breratheOutSpeed = \frac{volume}{breratheOutTime}$$

end

Algorithm 3: Breath Cycle (Breathe-out)

Result: Sets the maximum speed and acceleration of stepper for breath-out phase
//to be added in always executing loop;

if *breathe-in Period has elapsed* **then**

set maximum stepper speed as;

$$stepperMaxSpeed = (motorSpeedFor1liter/sec) * (breratheOutSpeed)$$

set maximum stepper acceleration as;

$$stepperMaxAcceleration = breratheOutSpeed * breratheOutSpeed * motorAcceleration$$

move stepper back to 0;

end