RESEARCH ARTICLE OPEN ACCESS

Manuscript received December 31, 2022; revised January 21, 2023; accepted February 02, 2023; date of publication February 25, 2023
Digital Object Identifier (**DOI**): https://doi.org/10.35882/ijeeemi.v5i1.263

How to cite: Levana Forra Wakidi, Bambang Guruh Irianto, Abd. Kholiq, Eko Dedi Prasetyo, Chandrasekaran P, "QRS Detection On Heart Rate Variability Readings using Two Moving Average Methods", Indonesian Journal of Electronics, Electromedical Engineering, and Medical Informatics, vol. 5, no. 1, pp. 30–37, February. 2023.

Pressure Sensor Stability Analysis of Positive End Expiratory Pressure Parameters in Flow Analyzer Design

Levana Forra Wakidi[1], Bambang Guruh Irianto[1], and Abd. Kholiq[1], Eko Dedi Prasetyo[1], Chandrasekaran P[2]

[1] Department of Electrical Engineering Poltekkes Ministry of Health, Surabaya, Pucang Jajar Timur 10, Surabaya, 60245, Indonesia
[2] K.S.Rangasamy College of Arts and Science Autonomous, India

Corresponding author: lep.forra@gmail.com

ABSTRACT The Positive End Expiratory Pressure (PEEP) parameter is a parameter that must be considered in the process of determining the patient's condition, a safe threshold, and must be in accordance with the settings. However, the PEEP value on the ventilator often does not match the settings so that the measuring instrument capable of detecting PEEP on the ventilator is the Flow Analyzer. The purpose of this study was to design a Flow Analyzer using the MPX2010 sensor to analyze the stability of the PEEP parameters on the ventilator. The main contribution of this research is the design of a simple Flow Analyzer device with stable monitoring of PEEP parameters and the availability of many required setting options. This study used PEEP settings of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O. In this case, data were collected using a ventilator with VCV (Volume Control Ventilation) and PCV (Pressure Control Ventilation) modes. The tool used for reference from standard measurements was the Standard Flow Analyzer tool. The results of this study indicated that the measurement accuracy of PEEP parameters with the Flow Analyzer module at each PEEP setting had the smallest error of ±0% at 0 cmH2O setting so that it also had the smallest value of 0 by standard. deviation and uncertainty (UA) value 0 at each setting. Meanwhile, the Flow Analyzer measurement module had the largest error in the 5 cmH2O setting, which was ±13.2% with the largest correction value of 0.77. Based on the data obtained, the monitoring of the PEEP parameter was considered quite stable even though the value was still out of tolerance. Therefore, the monitoring of PEEP stability parameters can be implemented during the ventilator calibration process in order to analyze damage and reduce the time of damage to the ventilator.

INDEX TERMS PEEP; Calibration; PCV Mode; VCV Mode; Stability.

I. INTRODUCTION

Positive End Expiratory Pressure (PEEP) is a parameter on the ventilator for oxygenation of patients by giving positive pressure at the end of breathing to keep the pulmonary alveoli from collapsing so that the lungs do not experience atelectasis. PEEP is useful so that the diffusion of oxygen and carbon dioxide can still take place in the pulmonary alveoli. In addition, another impact of using PEEP is a decrease in venous return or filling of the heart [1][2]. This pressure acts as a buffer (stent) that keeps the small airways in the lungs open at the end of expiration [3]. In this case, the recommended PEEP setting value is 5 - 25 cmH2O [4][5]. The use of higher PEEP values can increase alveolar recruitment, reduce stress, lung strain, and prevent atelectrauma in some patients [6][7][8]. However, the application of high-pressure PEEP can also increase the risk of alveolar overdistention and pulmonary vascular resistance [9][10]. The monitoring of PEEP parameters on the ventilator must be carried out strictly in order to determine the patient's condition, evaluate the safe threshold, and must comply with the setting values [11][12]. The PEEP value of patients monitored on a ventilator often does not match the setting value [13]. The occurrence of damage to the ventilator in the inspiratory and expiratory control sections can result in disrupted service to patients and losses from the use of other ventilators. Research on the stability of the pressure sensor on the PEEP parameter will help analyze equipment damage and reduce the breakdown time of the ventilator so that services for patients with respiratory failure can be provided properly and quickly.

In 2020, Paolo Marchionni et al. conducted a study to find the optimal PEEP value by looking at the SpO2 value. This study used an experimental method involving 5 patients with different PEEP settings so that the optimal PEEP value was 8 CmH2O with a median SpO2 value of 98% [14].

In the same year, M. Jaber et al. carried out a study on the development of a computational model to study the impact of different ventilation modes, employing Simulink/MATLAB [15].

In 2012, Göran Cewers et al. studied the ventilator control to deliver gas flow and relate the gas flow from the user's airway to the barrier provided in the line between the pressure sensor and the patient in such a way. In this case, the first volume was determined in the line between the first valve and the limiter. Meanwhile, the second volume was determined in the line between the patient and the restrictor. The flow of gas expelled from the expiratory limb was limited by using the exhaust valve for the need to maintain a certain pressure in the patient's lungs at the end of expiration which is called Positive End Expiratory Pressure (PEEP) [16]. This research still needs to develop a monitoring system for graphs of flow rate and pressure variations during the breathing cycle of constant flow rate in real-time. Furthermore, in 2015, Pu Zhang et al. designed a calibrator to increase the accuracy of the latest integration ventilator testing, which involved a calibration module with static parameters (gas flow rate and pressure) and dynamic parameters (tidal volume, peak airway pressure, and positive end-expiratory pressure (PEEP)) [17]. This study developed a system that can track the results of calibration or verification of ventilator testers with high accuracy and used the appropriate software for its function. However, this study did not mention the type of each sensor and the measurement error value of each parameter. In 2018, Sara Zulfiqar et al. designed a prototype low-cost portable ventilator with a breath control feedback path from two self-calibrating sensors. In this case, users can enter setting data via the Graphical User Interface (GUI) on the touch screen module. In addition, it also functions to display the signal being controlled. Data from the sensor were optically isolated and converted into a PWM signal for precise readings. The system was controlled digitally with various settings so that it can be adjusted to the patient's needs. The air pump used a DC piston, with a modified converter as speed controller, and PID tuned so that it can be replaced with a similar pump. The pressure sensor used MPX4250DP for negative and positive pressure parameters such as PEEP, while the airflow rate sensor used AWM720P1 for the respiratory rate per minute (BPM) parameter [18][19]. However, this study did not explained the results of measuring tools with comparison tools so that the error value for each parameter measurement was not yet known.

In 2020, H. Y. Al-Hetari et al. conducted a study using a single-compartment lung model VCV mode during real-time mechanical ventilation using pressure signals. This mathematical model described lung volume and compliance correctly by considering positive end-expiratory pressure (PEEP) values. The model was implemented using the LabVIEW tool and can be used to monitor volume, flow, and fit as output from the model. Two trials were carried out on the proposed lung model in three volume input scenarios (400, 500, and 600 ml), where each trial taking into account the PEEP value [20].

Furthermore, R. Nadeem et al. in 2019 conducted a study of the maximum tolerable PEEP in mechanically ventilated patients with PEEP parameter settings of 0, 5, 8,11, 14, 17, 20, 23, 26, 29 cmH2O. Based on this study, it was concluded that under controlled conditions, higher PEEP levels were well tolerated and the PEEP setting of 29 CmH2O was the highest PEEP setting that could be tolerated in patients [21][22].

Based on the identification of the problems above, this study designed a Flow Analyzer using the MPX2010 sensor to analyze the stability of the PEEP parameter on the ventilator by displaying a graph of the respiratory rate in real-time on the TFT LCD. Therefore, in this research, it is expected that readers will know:

a. Monitoring the stability of the patient's respiratory rate in real-time clearly and easily.
b. The design can be used as a ventilator calibration tool, especially for the PEEP parameter.
c. Monitoring of PEEP stability parameters can be applied during the ventilator calibration process which needs to be carried out to analyze damage and reduce the time for ventilator damage.

This study produced a Flow Analyzer design using the MPX2010 sensor to analyze the stability of the PEEP parameters on the ventilator with settings of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O. Meanwhile, the data analysis was conducted using a ventilator with VCV (Volume Control Ventilation) and PCV (Pressure Control Ventilation) modes showing that monitoring of PEEP parameters was quite stable.

II. MATERIALS AND METHOD

A. SETTING TEST

This study used a standard ventilator and Flow Analyzer which was connected to a module using a Y-Piece. Sampling was carried out with VCV (Volume Control Ventilation) and PCV (Pressure Control Ventilation) modes with 5 times data collection.

1) MATERIALS AND TOOLS

This study used the MPX2010 sensor as a pressure sensor given from the ventilator. The MPX2010 sensor is a silicon

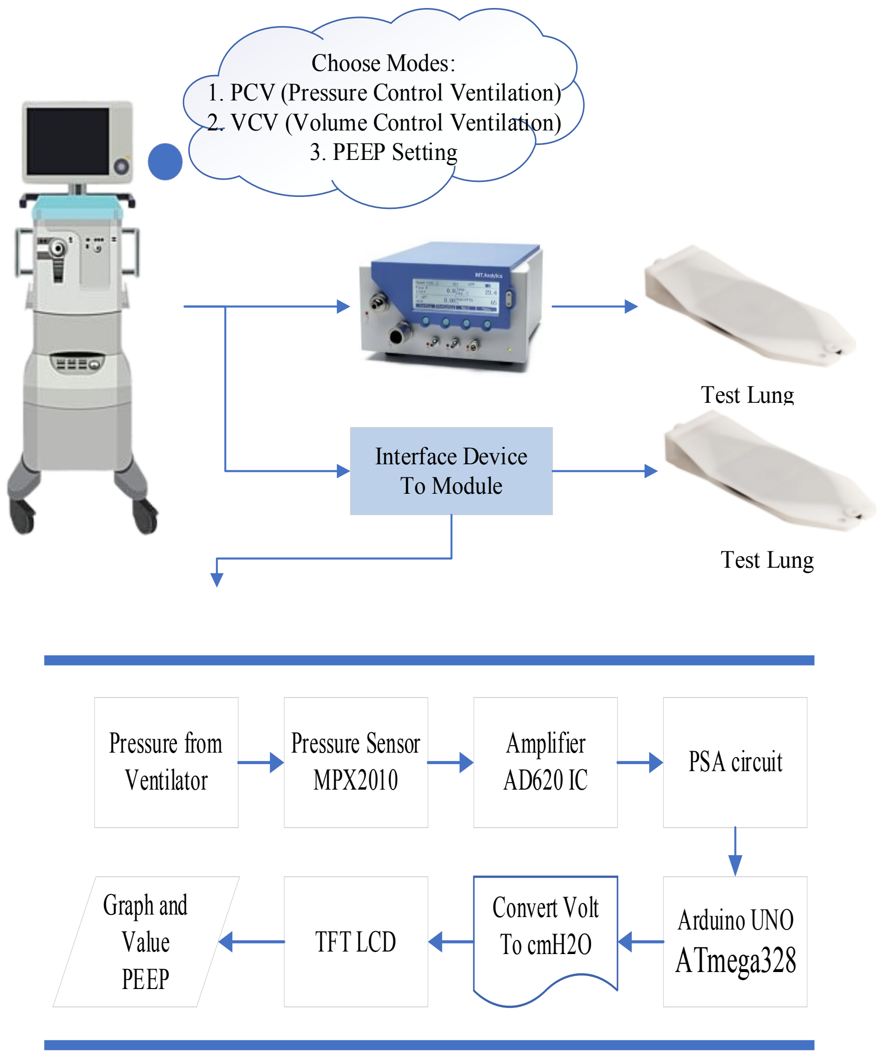


FIGURE 1. Block Diagram of Flow Analyzer Module

piezoresistive pressure sensor with an excellent, accurate, and linear voltage output directly proportional to the applied pressure [23]. Meanwhile, the microcontroller used to process the data was Arduino UNO R3. Arduino Uno is an ATmega328 based microcontroller board. Arduino has 14 input/output pins of which 6 can be used as PWM outputs, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button [24]. In order to send the data sending data, Nextion Enhanced Thin Film Transistor was used to display graphs and PEEP values. This Nextion has a touchscreen feature that can present various information on the screen, as well as control, analyze, and collect data by several sensors [25].

2) EXPERIMENT

In this study, after the design is complete, a test is carried out on the response of the MPX2010 pressure sensor to detect the PEEP value. The ventilator will be set to VCV or PCV mode and then the PEEP value will be set to be tested. The PEEP settings used are 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O. Furthermore, the ventilator will apply pressure to the standard Flow Analyzer tool and the module simultaneously via a Y-piece connection. The standard Flow Analyzer tool will detect the PEEP value as a reference and comparison of the PEEP value as well as the pressure graph detected from the Flow Analyzer module which will be displayed on the Thin Film Transistor Liquid Crystal Display in real time. The data obtained from the Flow Analyzer module will be analyzed based on the average value, standard deviation, error, uncertainty (UA), and correction.

B. BLOCK DIAGRAM

The system works according to FIGURE 1. The system starts working when the appliance is ON (ON). The Arduino Nano Microcontroller IC will initialize the connected hardware including the Thin Film Transistor Liquid Crystal Display, and the pressure sensor. The pressure sensor will measure how much pressure is given by the ventilator to the patient's lungs which has been adjusted by setting the PEEP parameter. This pressure sensor works Arduino Uno Microcontroller IC processes the pressure sensor readings and displays it on the Thin Film Transistor liquid crystal display in the form of numbers and graphics.

C. FLOW CHART

The program that was built based on the flowchart as shown in FIGURE 2. created a module on the Y-piece that was connected to the Flow Analyzer and Test Lung, then sets PEEP and other parameters on the ventilator. In order to start the system, press the ventilation start button on the ventilator, the pressure sensor then detected the pressure on the hose in the form of voltage which was then converted into pressure (cmH2O). The result of the conversion was further displayed in the form of PEEP values and graphs on the Thin Film Transistor liquid crystal display.

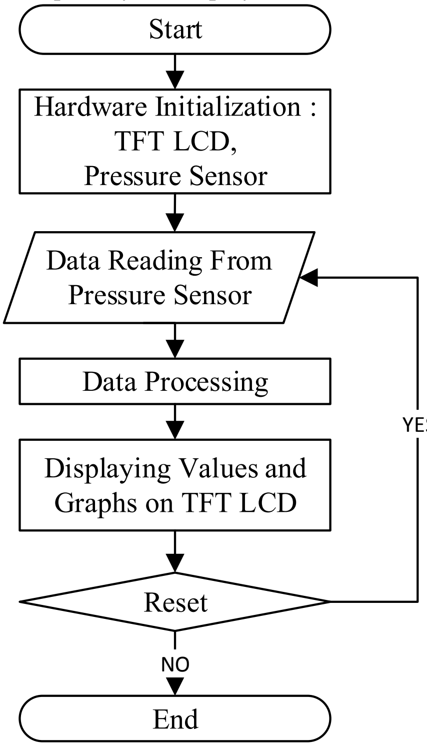


FIGURE 2 **Flow Chart of Flow Analyzer Module**

III. RESULT

In this study, the ventilator has been tested using the Standard Flow Analyzer and the Flow Analyzer module. The flow analyzer module design can be seen in FIGURE 3. Here are the results of the study:

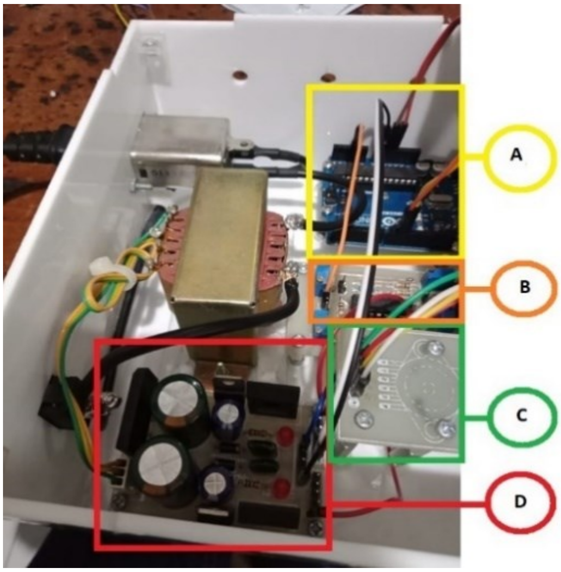


FIGURE 3. **Flow Analyzer Module Design**

1) DESIGN

Based on FIGURE 3, the system circuit consists of a power supply circuit in part D as a voltage source for the entire circuit, the MPX2010 sensor circuit as in part C which was connected to the AD620 IC to amplify the difference in part B, and the PSA circuit to condition the signal output of the AD620 IC so that it matches the input voltage range for the microcontroller. Microcontroller in part A, the Arduino UNO R3 was used to process the circuit output to be converted into cmH2O and send the output data to be displayed on the Thin Film Transistor Liquid Crystal Display in the form of a graph and PEEP value.

2) LISTING PROGRAM CONVERSION VALUE CMH2O

In this study, the microcontroller used was Arduino UNO R3 which consists of a program to change the analog voltage value read from the sensor connected to pin A2 to be converted into digital data and displayed with cmH2O pressure units. In addition, the data were also converted and visualized with graphs which can view the stability of the sensor, and detect the value of the PEEP measurement. The program to convert values to cmH2O pressure units was intended for Program Listing 1, namely the voltage value measured on the sensor (analog voltage) calculated from the sensor value multiplied by the maximum ADC value of 5.0 Volts divided by the value of 10 data bits (1023.0). Meanwhile, the pressure value (kPa) can be calculated by the formula ((measured pressure value/ maximum stress value (5.0))-0.04)/0.09. Meanwhile, the formula for converting the kPa pressure value to cmh2o units is (pressure value (kPa)*10.1972) + 6.83 based on the provisions. This program also consists of an Arduino initialization program and a smoothing program to make it more stable.

Program Listing 1. Conversion Value cmH2O Program

```
void setup()
{
Serial.begin(9600);
NexSerial.begin(9600);
pinMode (A2,INPUT);
for (int thisReading = 0; thisReading < numReadings;
thisReading++) {
   readings[thisReading] = 0;}
delay (0);
}
void loop()
{
  sensorValue = analogRead(inputadc);
  tegangan = (sensorValue *(5.0/1023.0)); //konversi
ADC ke Volt
  kpa = ((tegangan/5.0)-0.04)/0.09; //konversi Volt ke
kPa
  konversi_cmh2o = (kpa*10.1972)+6.83; //konversi kPa
ke cmH2O dan adjustment
  if(konversi_cmh2o>19){
   konversi_cmh2o=konversi_cmh2o-6.3;
   }

  int grafikvalue = konversi_cmh2o;
    total = total - readings[readIndex]; // read from the
  sensor:
    readings[readIndex] = grafikvalue; // add the reading
  to the total:
    total = total + readings[readIndex]; // advance to the
  next position in the array:
    readIndex = readIndex + 1; // if we're at the end of
  the array...
    if (readIndex >= numReadings) { // ...wrap around to
  the beginning:
     readIndex = 0;}
  average = total / numReadings; // calculate the average
```

3) LISTING OF TFT LCD DISPLAY CHART PROGRAM

The listing program to display graphics on the Thin Film Transistor Liquid Crystal Display was displayed in Program Listing 2. This program consists of a program for sending data to be displayed in graphical form on the Thin Film Transistor Liquid Crystal Display in real time which was read from the sensor. The Thin Film Transistor Liquid Crystal Display also displays the cmH2O Converted Value Program. Monitoring PEEP values will be easier.

Program Listing 2. Chart Display Thin Film Transistor Liquid Crystal Display Program

```
    NexSerial.print("add 6,0,"+String(average));
  //Pengiriman grafik nextion
    NexSerial.write(0xff);
    NexSerial.write(0xff);
    NexSerial.write(0xff);
  delay(1);
```

4) LISTING DISPLAYS PEEP ON TFT LCD PROGRAM

The listing program for displaying the PEEP value on the Thin Film Transistor Liquid Crystal Display is shown in Listing Program 4. which consists of a program that displays the PEEP value in real time.

Program Listing 4. Listing Displays PEEP on Thin Film Transistor Liquid Crystal Display Program

```
 Nexserial.print("t1.txt=\"");  //perintah untuk nextion di t1
 Nexserial.print(PEEP);         //memunculkan hasilPEEP
 Nexserial.print("\"");
 Nexserial.write(0xff);
Nexserial.write(0xff);
Nexserial.write(0xff);
```

5) PEEP READING PROGRAM LISTING FROM CHART

The listing program for detecting the PEEP value of the entire data is shown in Listing Program 3. which consists of the program detecting the PEEP value.

Program Listing 3. PEEP Reading Program Listing from Chart

```
if (flag==0&&konversi_cmh2o>datasave){
  datasave=konversi_cmh2o;
}
if (flag==0&&konversi_cmh2o<datasave){
  flag=1;
  datasave=konversi_cmh2o;
 }
if (flag==1&&konversi_cmh2o<datasave){
  datasave=konversi_cmh2o;
 }
if (flag==1&&konversi_cmh2o>datasave){
  PEEPa=datasave;
  flag=2;
 }
if (flag==2&&PEEPa<konversi_cmh2o){
  PEEP=PEEPa;
  flag=2;
}
if (NexSerial.available()>0){
   String Received = NexSerial.readString();
   if (Received[0] == 'r' ){konversi_cmh2o=0;flag=0;
    }
 }
```

6) PEEP PARAMETER MEASUREMENT RESULTS WITH VCV (VOLUME CONTROL VENTILATION) AND PCV (PRESSURE CONTROL VENTILATION) MODES

FIGURE 4 shows that the measurement was carried out 5 times (X axis) at each PEEP parameter setting of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O (Y axis), obtaining that it has a stable output. However, there are differences in the output value of the module with the settings on the ventilator. The value of the difference is still within the allowable tolerance limit of 10% of the measurement results.

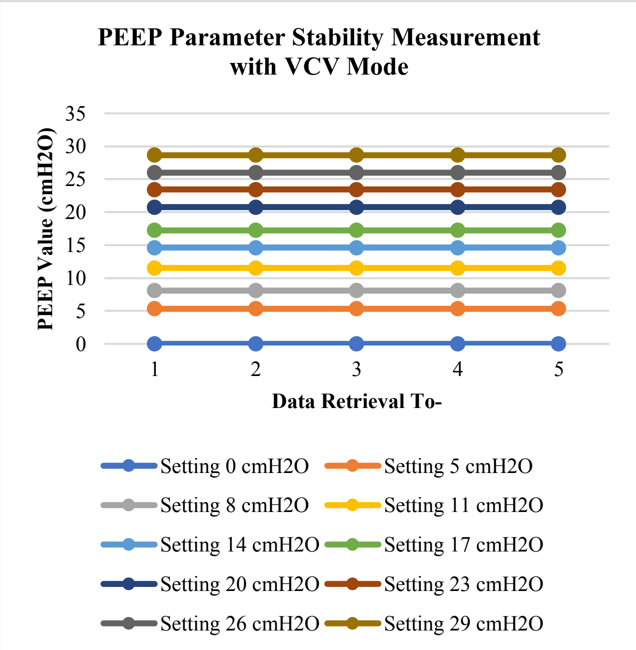


FIGURE 4. PEEP Parameter Measurement Results with VCV Mode.

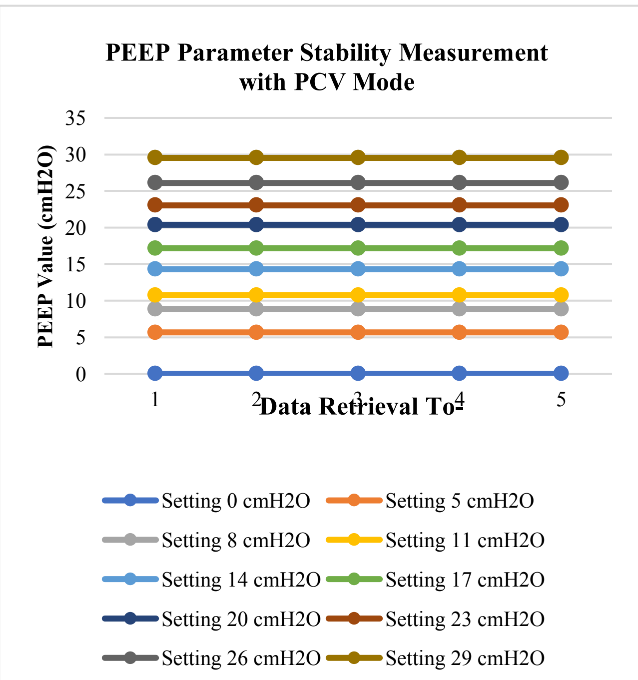


FIGURE 5. PEEP Parameter Measurement Results with PCV Mode.

FIGURE 5 shows that after 5 measurement attempts (X axis) at each PEEP parameter setting of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O (Y axis), it has a stable output. There was a difference in the output value of the module with the settings on the ventilator. Based on these differences, there were 2 data settings that are out of range, namely the 5 cmH2O setting of 13.2% and 8 cmH2O of 10.87%. In addition, the value of the difference was still within the tolerance range of 10%. FIGURE 4 and FIGURE 5 prove that the PEEP parameter setting measurement had a stable output using the MPX2010 pressure sensor based on the mode applied, namely VCV (Volume Control Ventilation) and PCV (Pressure Control Ventilation) on the Ventilator System.

7) ERROR ANALYSIS, STANDARD DEVIATION, UNCERTAINTY (UA), AND FLOW ANALYZER MODULE CORRECTION

The measurement of the Flow Analyzer module was carried out using VCV (Volume Control Ventilation) and PCV (Pressure Control Ventilation) modes with PEEP settings of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O in each mode. The following are the results of the comparison of errors in VCV and PCV modes which are shown in Table III. In terms of the ventilator, the standard deviation is shown in Table IV, the uncertainty (UA) is in Table V, and the correction values are shown in Table VI. Flow Analyzer module measurement error in TABLE 1 shows that the highest measurement error is 6.6% in the VCV mode, which is at the 5 cmH2O settings, while in the PCV mode it is 13.2% at the 5 cmH2O settings. Meanwhile, the lowest error value from both VCV and PCV measurements is in the 0 cmH2O setting, which is 0%. Meanwhile, if we look at the average error of the two modes, it can be seen that the error value in VCV mode is 2.45% lower than PCV mode, which is 3.16%.

TABLE 1
Comparison of PEEP parameters error value between VCV and PCV mode.

Setting PEEP (cmH2O)	Mode Ventilation	
	VCV (%)	PCV (%)
0	0	0
5	6.6	13.2
8	1.12	10.875
11	4.81	0.0227
14	4.43	2.27
17	1.35	1
20	3.85	1.9
23	1.96	0.130
26	0.038	0.38
29	1.24	1.82
Average ($\overline{X}$)	2.54	3.16

The standard deviation of PEEP measurements with setting values of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O as shown in TABLE 2 shows that each setting in VCV and PCV modes is 0 and does not exceed the average value. So, it can be concluded that the flow analyzer was good.

TABLE 2
Comparison of the standard deviation of the PEEP parameter between the VCV and PCV modes.

Setting PEEP (cmH2O)	Mode Ventilation	
	VCV	PCV
0	0	0
5	0	0
8	0	0
11	0	0
14	0	0
17	0	0
20	0	0
23	0	0
26	0	0
29	0	0
Average ($\overline{X}$)	0	0

TABLE 2 also shows the uncertainty value (UA) of the PEEP parameter setting of 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O to see how much deviation (accuracy) the Flow Analyzer module has in reading the PEEP value. Relative uncertainty is closely related to measurement accuracy, which can be stated if the smaller the uncertainty, the higher the accuracy. The same deviation and UA values are obtained for each PEEP setting with a small value of 0.

TABLE 3
Comparison of the standard deviation of the PEEP parameter between the VCV and PCV modes.

Setting PEEP (cmH2O)	Mode Ventilation	
	VCV (%)	PCV (%)
0	0	0
5	0.33	0.66
8	0.09	0.87
11	0.53	-0.25
14	0.62	0.29
17	0.23	0.17
20	0.77	0.38
23	0.45	0.03
26	-0.01	0.1
29	-0.36	0.53
Average ($\overline{X}$)	0.284	0.212

The existence of a correction value indicates an error in the system. So, the closer the correction value is to 0, the better the tool will work. Based on TABLE 3, the correction value for measuring the PEEP parameter with the settings 0, 5, 8, 11, 14, 17, 20, 23, 26, and 29 cmH2O, the largest in the VCV mode was found in the PEEP setting 20 cmH2O, which is 0.77. As for the PCV mode, the setting is 8 cmH2O, which is 0.87. For the smallest correction value in the VCV and PCV modes, it is at the 0 cmH2O setting, which is 0.

VI. DISCUSSION

The MPX2010 sensor will read the pressure value on the Ventilator Servo-I when the ventilator is working. The function of the pressure sensor is to read the pressure value data that will be displayed via a graph and read the PEEP value from the entire data. In VCV (Volume Control Ventilation) mode, the PEEP value was stable. The setting value which has the lowest error value of 0% is at 0 cmH2O setting and the highest error value at 5 cmH2O is 6.6%, while the distribution of data in each set is the same, with a standard deviation value of 0 which also produced an uncertainty of 0. An uncertainty value of 0 means that the stability of the results was good because there was no change in each measurement. Meanwhile, the correction value was also considered good with the highest value of 0.77 at the PEEP setting of 20 cmH2O which means the measurement accuracy was still fairly good. Furthermore, in PCV (Pressure Control Ventilation) mode, the setting value which has the lowest error value of 0% is 0 cmH2O setting and the highest error value at 5 cmH2O is 13.2%, while the distribution of data in each set is in PCV mode (Pressure control ventilation). the same as the standard deviation of 0 which also produces an uncertainty of 0. An uncertainty value of 0 can be interpreted if the stability of the results is good because there is no change in each measurement. Meanwhile, the correction value is still said to be good with the highest value of 0.87 at the PEEP setting of 8 cmH2O which means the measurement accuracy is still fairly good. These results indicated that the measurements on the module are fairly stable, with a fairly good level of accuracy with the highest error value of 13.2% at the PEEP setting of 5 cmH2O and the largest correction value of 0.87 at the PEEP setting of 8 cmH2O. This is different from the study conducted by Paolo Marchionni et al., which obtained results when the optimal PEEP value was at 8 cmH2O. For each setting, the PEEP value setting in each mode is in accordance with research conducted by R. Nadeem et al., namely 0, 5, 8.11, 14, 17, 20, 23, 26, 29 cmH2O [18].

This research has a problem in using the reset button on the TFT LCD which was less sensitive so it needs to be pressed many times. We recommend using another program that makes the reset button more sensitive so it does not need to be pressed repeatedly. Another problem experienced by the author is that the higher the conversion value, the further the difference. We recommend using another sensor that only uses 1 output so that it can be processed directly with the microcontroller without the need to add another circuit that can affect the value of the circuit output data and the effects of using 1 input or 2 inputs. In addition, the value and conversion formula as well as the type of sensor used and the function of each part need to be studied further because the discussion of the conversion formula to cmH2O units and others is not discussed in the study.

VII. CONCLUSION

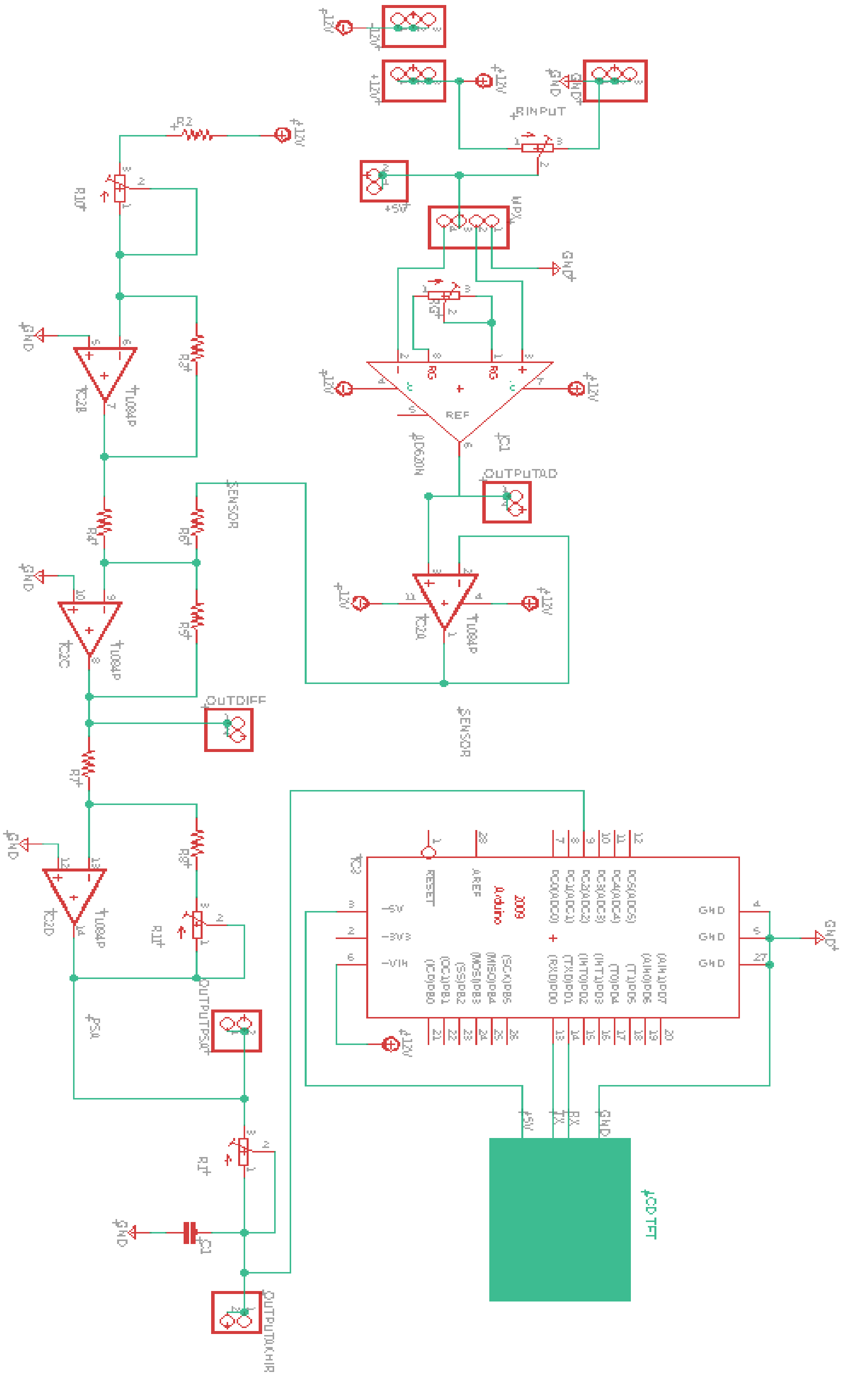
This study aims to design a Flow Analyzer using the MPX2010 sensor to analyze the stability of the PEEP parameters in the VCV (Volume Control Ventilation) and PCV (Pressure Control Ventilation) modes on the ventilator. Based on the data obtained, the PEEP reading value is close to the Standard Flow Analyzer measurement, namely the PEEP setting of 0 cmH2O with an error value of 0%. This applies to both VCV and PCV modes. Likewise, the value of the standard deviation and uncertainty (UA) in all PEEP settings is 0. Meanwhile, the smallest correction value in VCV mode is found in the 26 cmH2O PEEP setting, which is 0.01, and in PCV mode, it is found in the 23 cmH2O PEEP setting, which is 0.03. In the future, the authors hope that this research project can be developed by adding additional monitoring of the Inspiratory and Expiratory Ratio (I:E Ratio), Oxygen Fraction (FiO2) parameters and others which are useful for calibrating and detecting damage to the ventilator.

REFERENCES

[1] The Intensive Care Foundation, *Handbook of Mechanical Ventilation: A User's Guide Ventilation*. 2015.
[2] J. W. Kreit, *Mechanical Ventilation*, Second., vol. 13, no. 1. New York, 2013.
[3] B. Kumar, *Handbook of Mechanical Ventilation*. 2011. doi: 10.5005/jp/books/11466.
[4] L. Gattinoni *et al.*, "Positive end-expiratory pressure: How to set it at the individual level," *Ann. Transl. Med.*, vol. 5, no. 14, pp. 1–10, 2017, doi: 10.21037/atm.2017.06.64.
[5] A. Das, P. P. Menon, J. G. Hardman, and D. G. Bates, "Optimization of mechanical ventilator settings for pulmonary disease states," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 6, pp. 1599–1607, 2013, doi: 10.1109/TBME.2013.2239645.
[6] M. C. Sklar, B. K. Patel, J. R. Beitler, T. Piraino, and E. C. Goligher, "Optimal Ventilator Strategies in Acute Respiratory Distress Syndrome," *Semin. Respir. Crit. Care Med.*, vol. 40, no. 1, pp. 81–93, 2019, doi: 10.1055/s-0039-1683896.
[7] T. Bluth *et al.*, "Effect of Intraoperative High Positive End-Expiratory Pressure (PEEP) with Recruitment Maneuvers vs Low PEEP on Postoperative Pulmonary Complications in Obese Patients: A Randomized Clinical Trial," *JAMA - J. Am. Med. Assoc.*, vol. 321, no. 23, pp. 2292–2305, 2019, doi: 10.1001/jama.2019.7505.
[8] F. Gordo and I. Conejo, "What PEEP level should I use in my patient?," *Med. Intensiva*, vol. 41, no. 5, pp. 267–269, 2017, doi: 10.1016/j.medin.2016.10.010.
[9] E. Fan *et al.*, "An official American Thoracic Society/European Society of intensive care medicine/society of critical care medicine clinical practice guideline: Mechanical ventilation in adult patients with acute respiratory distress syndrome," *Am. J. Respir. Crit. Care Med.*, vol. 195, no. 9, pp. 1253–1263, 2017, doi: 10.1164/rccm.201703-0548ST.
[10] R. Carrasco Loza, G. Villamizar Rodríguez, and N. Medel Fernández, "Ventilator-Induced Lung Injury (VILI) in Acute Respiratory Distress Syndrome (ARDS): Volutrauma and Molecular Effects," *Open Respir. Med. J.*, vol. 9, no. 1, pp. 112–119, 2015, doi: 10.2174/1874306401509010112.
[11] P. L. Silva and P. R. M. Rocco, "The basics of respiratory mechanics: ventilator-derived parameters," *Ann. Transl. Med.*, vol. 6, no. 19, pp. 376–376, 2018, doi: 10.21037/atm.2018.06.06.
[12] M. Vargas, Y. Sutherasan, C. Gregoretti, and P. Pelosi, "PEEP role in ICU and operating room: From pathophysiology to clinical practice," *Sci. World J.*, vol. 2014, 2014, doi: 10.1155/2014/852356.
[13] A. M. Mercat and J. C. M. Richard, "Positive End-Expiratory Pressure Setting in Adults With Acute Lung Injury and Acute Respiratory Distress Syndrome," *Surv. Anesthesiol.*, vol. 52, no. 6, pp. 272–273, 2008, doi: 10.1097/01.sa.0000318641.66249.02.
[14] P. Marchionni, A. Galli, V. P. Carnielli, and L. Scalise, "A novel measurement technique for the assessment of best positive end-expiratory pressure in newborn patient," *IEEE Med. Meas. Appl. MeMeA 2020 - Conf. Proc.*, 2020, doi: 10.1109/MeMeA49120.2020.9137151.
[15] T. Sujeeth, "Low Cost Mechanical Ventilator with Controlled PEEP," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. VI, pp. 2273–2278, 2021, doi: 10.22214/ijraset.2021.34662.
[16] G. Cewers, "Expiratory Pressure Regulation in a Ventilator," *Patent*, vol. i, no. 12, 2012.
[17] P. Zhang, J. Sun, G. Wan, and W. Liu, "Development of Ventilator Tester Calibration Equipment," *IET Conf. Publ.*, vol. 2015, no. CP680, 2015, doi: 10.1049/cp.2015.0790.
[18] S. Zulfiqar, H. Nadeem, Z. Tahir, M. Mazhar, and K. M. Hasan, "Portable, Low Cost, Closed-Loop Mechanical Ventilation Using Feedback from Optically Isolated Analog Sensors," *IEEE Reg. 10 Annu. Int. Conf. Proceedings/TENCON*, vol. 2018-Octob, no. October, pp. 1913–1916, 2019, doi: 10.1109/TENCON.2018.8650277.
[19] I. Freescale Semiconductor, *Tehnical Data MPX4250 Series*. 2009.
[20] A. C. McKown, M. W. Semler, and T. W. Rice, "Best PEEP trials are dependent on tidal volume," *Crit. Care*, vol. 22, no. 1, pp. 21–23, 2018, doi: 10.1186/s13054-018-2047-4.
[21] R. Nadeem, K. Ahmed, L. Salama, and A. M. Elhoufi, "Tolerable Maximum Positive End Expiratory Pressure in Mechanically Ventilated Patients and Its Impact on Blood Flow across Cardiac Valves: Index Case Report of a Physiology Study," *Dubai Med. J.*, vol. 2, no. 4, pp. 163–167, 2019, doi: 10.1159/000504045.
[22] H. Kaykisizli and V. Ciftci, "Reference standard design for flow calibration of mechanical ventilator and lung simulator calibrators," *TÜBİTAK UME J.*, vol. 10, pp. 1–4, 2010, doi: 10.1109/biyomut.2010.5479874.
[23] F. I. Semiconductor, "Freescale Semiconductor 200 kPa On-Chip Temperature Compensated Silicon Pressure Sensors," *NPX Datasheet*, pp. 2006–2008, 2008.
[24] C. Rajan, B. Megala, A. Nandhini, and C. R. Priya, "A Review : Comparative Analysis of Arduino Micro Controllers in Robotic Car," *Int. J. Mech. Aerospace, Ind. Mechatronics Eng.*, vol. 9, no. 2, pp. 371–380, 2015.
[25] A. C. Bento, "An Experiment with Arduino Uno and Tft Nextion for Internet of Things," *2018 Int. Conf. Recent Innov. Electr. Electron. Commun. Eng. ICRIEECE 2018*, no. July 2018, pp. 1238–1242, 2018, doi: 10.1109/ICRIEECE44171.2018.9008416.

ATTACHMENT

1. The Circuit of Flow Analyzer



2. Program Listing

- Inisialisasi Program Arduino

```
#include <SoftwareSerial.h>
SoftwareSerial NexSerial(9,10); //RX, TX arduino
const int numReadings = 5;
int readings[numReadings];      // the readings from the analog input
int readIndex = 0;              // the index of the current reading
int total = 0;                  // the running total
int average = 0;                // the average
float inputadc = A2;            //INPUT
float sensorValue,tegangan,kpa,konversi_cmh2o;
float datasave;
float PEEP,PEEPa;
int flag=0;
```

- Program Awal yang Dijalankan Arduino

```
void setup()
{
Serial.begin(9600);
NexSerial.begin(9600);
pinMode (A2,INPUT);
for (int thisReading = 0; thisReading < numReadings; thisReading++) {
   readings[thisReading] = 0;}
delay (0);
}
```

- Program Konversi

```
void loop()
{
  sensorValue = analogRead(inputadc);
  tegangan = (sensorValue *(5.0/1023.0)); //konversi ADC ke Volt
  kpa = ((tegangan/5.0)-0.04)/0.09; //konversi Volt ke kPa
  konversi_cmh2o = (kpa*10.1972)+4.53; //konversi kPa ke cmH2O dan adjustment
  if(konversi_cmh2o>19){
   konversi_cmh2o=konversi_cmh2o-3.7;
   }
int grafikvalue = konversi_cmh2o;
  total = total - readings[readIndex]; // read from the sensor:
  readings[readIndex] = grafikvalue; // add the reading to the total:
  total = total + readings[readIndex]; // advance to the next position in the array:
  readIndex = readIndex + 1; // if we're at the end of the array...
  if (readIndex >= numReadings) { // ...wrap around to the beginning:
   readIndex = 0;}
   average = total / numReadings; // calculate the average
```

- Program Menampilkan Grafik Pada LCD TFT

```
  NexSerial.print("add 6,0,"+String(average)); //Pengiriman grafik nextion
  NexSerial.write(0xff);
  NexSerial.write(0xff);
  NexSerial.write(0xff);
  delay(1);
```

- Program Pembacaan Nilai PEEP dari Grafik

```
if (flag==0&&konversi_cmh2o<datasave){
  datasave=konversi_cmh2o;
 }
```

```
if (flag==0&&konversi_cmh2o>datasave){
  flag=1;
  datasave=konversi_cmh2o;
 }
if (flag==1&&konversi_cmh2o>datasave){
  datasave=konversi_cmh2o;
 }
if (flag==1&&konversi_cmh2o>datasave){
  PEEPa=datasave;
  flag=2;
 }
if (flag==2&&PEEPa<konversi_cmh2o){
  PEEP=PEEPa;
  flag=2;
}
if (NexSerial.available()>0){
   String Received = NexSerial.readString();
   if (Received[0] == 'r' ){konversi_cmh2o=0;flag=0;
    }
 }
```

- Program Menampilkan Nilai PEEP Pada LCD TFT

```
NexSerial.print("t1.txt=\"");  //perintah untuk nextion di t1
 NexSerial.print(PEEP);         //memunculkan hasilPEEP
 NexSerial.print("\"");
 NexSerial.write(0xff);
 NexSerial.write(0xff);
 NexSerial.write(0xff);
```