DAFTAR PUSTAKA

Ashry, H. S. Al, & Modrykamien, A. M. (2014). *Humidification during Mechanical Ventilation in the Adult Patient*. *2014*.

Buckley, D., & Gillham, M. (2007). Invasive Respiratory Support. In *Cardiothoracic Critical Care* (Second Edi). https://doi.org/10.1016/B978-075067572-7.50032-1

Ch, J. A. C., Gianpaul, C., Ruiz, R., Jesús, B., Agreda, M., Anaya, S. C., … Cisneros, J. De. (2021). *A Low-cost Air Flow Sensor / transducer for Medical Applications : Design and Experimental Characterization*. (January), 1–23. https://doi.org/10.20944/preprints202101.0404.v1

Chi, Y., He, H., & Long, Y. (2020). A simple method of mechanical power calculation : using mean airway pressure to replace plateau pressure. *Journal of Clinical Monitoring and Computing*, (0123456789). https://doi.org/10.1007/s10877-020-00575-y

Galbiati, C., Abba, A., Agnes, P., Amaudruz, P., Arba, M., Ardellier-Desages, F., … Zardoni, A. (2020). *Mechanical Ventilator Milano (MVM): A Novel Mechanical Ventilator Designed for Mass Scale Production in Response to the COVID-19 Pandemic*. (Mvm). https://doi.org/10.1101/2020.03.24.20042234

Garmendia, O., Rodríguez-Lazaro, M. A., Otero, J., Phan, P., Stoyanova, A., Dinh-Xuan, A. T., … Farré, R. (2020). Low-cost, easy-to-build non-invasive pressure support ventilator for under-resourced regions: open source hardware description, performance and feasibility testing. *European Respiratory Journal*, 2000846. https://doi.org/10.1183/13993003.00846-2020

Giri, J., Kshirsagar, N., & Wanjari, A. (2021). Materials Today : Proceedings Design and simulation of AI-based low-cost mechanical ventilator : An approach. *Materials Today: Proceedings*, (xxxx), 4–9. https://doi.org/10.1016/j.matpr.2021.04.369

Hill, B., Lecturer, S., & Annesley, S. H. (2017). *Monitoring respiratory rate in adults*. 2017–2020.

Jürß, H., & Degner, M. (2018). A new compact and low-cost respirator concept for one way usage. *IFAC-PapersOnLine*, *51*(27), 367–372. https://doi.org/10.1016/j.ifacol.2018.11.612

Kenaan, M., & Hyzy, R. C. (n.d.). *50 - Mechanical Ventilation and Advanced Respiratory Support in the Cardiac Intensive Care Unit*. https://doi.org/10.1016/B978-0-323-52993-8.00050-3

Kerechanin, C. W., Cutchis, P. N., Vincent, J. A., Smith, D. G., & Wenstrand,

D. S. (2004). Development of field portable ventilator systems for domestic and military emergency medical response. *Johns Hopkins APL Technical Digest (Applied Physics Laboratory)*, *25*(3), 214–222.

Larsson, C., Larsson, C., & Yang, D. (2009). *Humidification of Inspiration Gas in a Ventilator Humidification of Inspiration Gas in a Ventilator*.

MHRA. (2020). *Rapidly manufactured ventilator system specification*. (April), 1–7. Retrieved from https://www.gov.uk/government/publications/coronavirus-covid-19-ventilator-supply- specification/rapidly-manufactured-ventilator-system-specification

Mohsen Al Husseini, A., Ju Lee, H., Negrete, J., Powelson, S., Tepper Servi, A., & Slocum, A. H. (2010). Design and prototyping of a low-cost portable mechanical ventilator. *Journal of Medical Devices, Transactions of the ASME*, *4*(2), 1–1. https://doi.org/10.1115/1.3442790

Pettenuzzo, T., & Fan, E. (2017). 2016 year in review: Mechanical ventilation. *Respiratory Care*, *62*(5), 629–635. https://doi.org/10.4187/respcare.05545

S, J. M. K. B., S, M. M. S. B., A, D. C. S. B., Msme, S. S., Sammour, I., Chse, F., … Design, R. L. C. (2020). *Design and performance testing of a novel emergency ventilator for in-hospital use*. *56*(September), 42–51. https://doi.org/10.29390/cjrt-2020-023.Background

Shunker, S., & Icu, L. (2016). *Mechanical Ventilation Learning Package*.

Tehrani, F. T. (2008). Automatic control of mechanical ventilation. Part 2: The existing techniques and future trends. *Journal of Clinical Monitoring and Computing*, *22*(6), 417–424. https://doi.org/10.1007/s10877-008-9151-y

Tsuzuki, M. S. G., Martins, T. C., Takimoto, R. T., Tanabi, N., Sato, A. K., Scaff, W., … Silverio, T. B. (2021). Mechanical Ventilator VENT19. *Polytechnica*, (Oxford 2020). https://doi.org/10.1007/s41050-021-00031-z

Warner, M. A., & Patel, B. (2012). Mechanical Ventilation. In *Benumof and Hagberg's Airway Management: Third Edition* (Third Edit). https://doi.org/10.1016/B978-1-4377-2764-7.00048-8

LAMPIRAN

Lampiran 1. Program ventilator

code_venus_rev1 | Arduino 1.8.13

File Edit Sketch Tools Help

code_venus_rev1

```
#include <ezButton.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <EEPROMEx.h>
#include <EEPROMVar.h>
#include "Arduino.h"

void disp(int input,byte col,byte row);
void setup();
void loop();

LiquidCrystal_I2C lcd(0x27,20,4);

ezButton pbset(2);
ezButton pbleft(3);
ezButton pbup(4);
ezButton pbright(5);
ezButton pbdown(6);
ezButton pbok(7);

const byte highpress = 44;
bool highpresstate = LOW;
unsigned long sebelum1 = 0;
long highpresson;
long highpressoff;

const byte lowpress = 45;
bool lowpresstate = LOW;
unsigned long sebelum2 = 0;
long highpresson;
long highpressoff;

const byte lowpress = 45;
bool lowpresstate = LOW;
unsigned long sebelum2 = 0;
long lowpresson;
long lowpressoff;

unsigned long sebelum3 = 0;
const long sampling = 100;

char tv_buf[9];
char bpm_buf[9];
char ie_buf[9];
```

```
ezButton pbleft(3);
ezButton pbup(4);
ezButton pbright(5);
ezButton pbdown(6);
ezButton pbok(7);

const byte highpress = 44;
bool highpresstate = LOW;
unsigned long sebelum1 = 0;
long highpresson;
long highpressoff;

const byte lowpress = 45;
bool lowpresstate = LOW;
unsigned long sebelum2 = 0;

//const float ADC_mV = 4.8828125;       // convesion multiplier from Arduino ADC value to voltage in mV
const float ADC_mV = 4.8828125;       // convesion multiplier from Arduino ADC value to voltage in mV
const float SensorOffset = 200.0;     // in mV taken from datasheet
const float sensitivity = 4.413;      // in mV/mmH2O taken from datasheet
const float mmh2O_cmH2O = 10;         // divide by this figure to convert mmH2O to cmH2O
const float mmh2O_kpa = 0.00981;      // convesion multiplier from mmH2O to kPa
const float mmh2O_pa = 9.81;      // convesion multiplier from mmH2O to Pa

void disp(int input,byte col,byte row){
  char buff[9];
  if (input <10){
    lcd.setCursor(col+1,row);lcd.print("    ");
    lcd.setCursor(col,row);lcd.print(itoa(input,buff,10));
  }else if ((input>=10)&&(input<100)){
    lcd.setCursor(col+2,row);lcd.print("   ");
    lcd.setCursor(col,row);lcd.print(itoa(input,buff,10));
  }else if ((input>=100)&&(input<1000)){
    lcd.setCursor(col+3,row);lcd.print("  ");
    lcd.setCursor(col,row);lcd.print(itoa(input,buff,10));
  }else if (input>=1000){
    lcd.setCursor(col+4,row);lcd.print(" ");
    lcd.setCursor(col,row);lcd.print(itoa(input,buff,10));
  }else {
    lcd.setCursor(col,row);lcd.print(itoa(input,buff,10));
  }
}

void setup() {
  pinMode(highpress,OUTPUT);
  pinMode(lowpress,OUTPUT);
  digitalWrite(highpress,LOW);
  digitalWrite(lowpress,LOW);

  pbset.setDebounceTime(50);
  pbleft.setDebounceTime(50);
  pbup.setDebounceTime(50);
  pbright.setDebounceTime(50);
  pbdown.setDebounceTime(50);
  pbok.setDebounceTime(50);

  Serial.begin(115200);
```

```
  lcd.init();
  lcd.init();
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("  >>> VENUS-02 <<<  ");
  lcd.setCursor(0,1);
  lcd.print("  VENTILATOR UNHAS  ");
  delay(1000);
  lcd.clear();
  tv = EEPROM.readInt(addr_tv);
  bpm = EEPROM.readInt(addr_bpm);
  ie = EEPROM.readInt(addr_ie);
}

void loop() {
  pbset.loop();pbleft.loop();pbup.loop();pbright.loop();pbdown.loop();pbok.loop();

  switch (state){
    case 0  :
              lcd.setCursor(0,0);lcd.print("HOME");
              lcd.setCursor(0,1);lcd.print("TV  :");disp(tv,5,1);
              lcd.setCursor(0,2);lcd.print("BPM :");disp(bpm,5,2);
              lcd.setCursor(0,3);lcd.print("IE  :1/");disp(ie,7,3);
              if(pbok.isPressed()){
                lcd.clear();
                state = 1;
              }else {
                state = 0;
              }
              break;

    case 1  :
              lcd.setCursor(0,0);lcd.print("Set Volume");
              lcd.setCursor(0,1);lcd.print("TV  :");disp(tv,5,1);
              lcd.setCursor(0,2);lcd.print("BPM :");disp(bpm,5,2);
              lcd.setCursor(0,3);lcd.print("IE  :1/");disp(ie,7,3);
              if (tv>800){
                tv = 800;
              }else if (tv<100){
                tv = 100;
              }else {
                tv = tv;


             if(pbup.isReleased()){
               tv=tv+10;
             }else if(pbdown.isReleased()){
               tv=tv-10;
             }else if(pbleft.isReleased()){
               lcd.clear();
               state = 0;
             }else if(pbright.isReleased()){
               lcd.clear();
               state = 2;
```

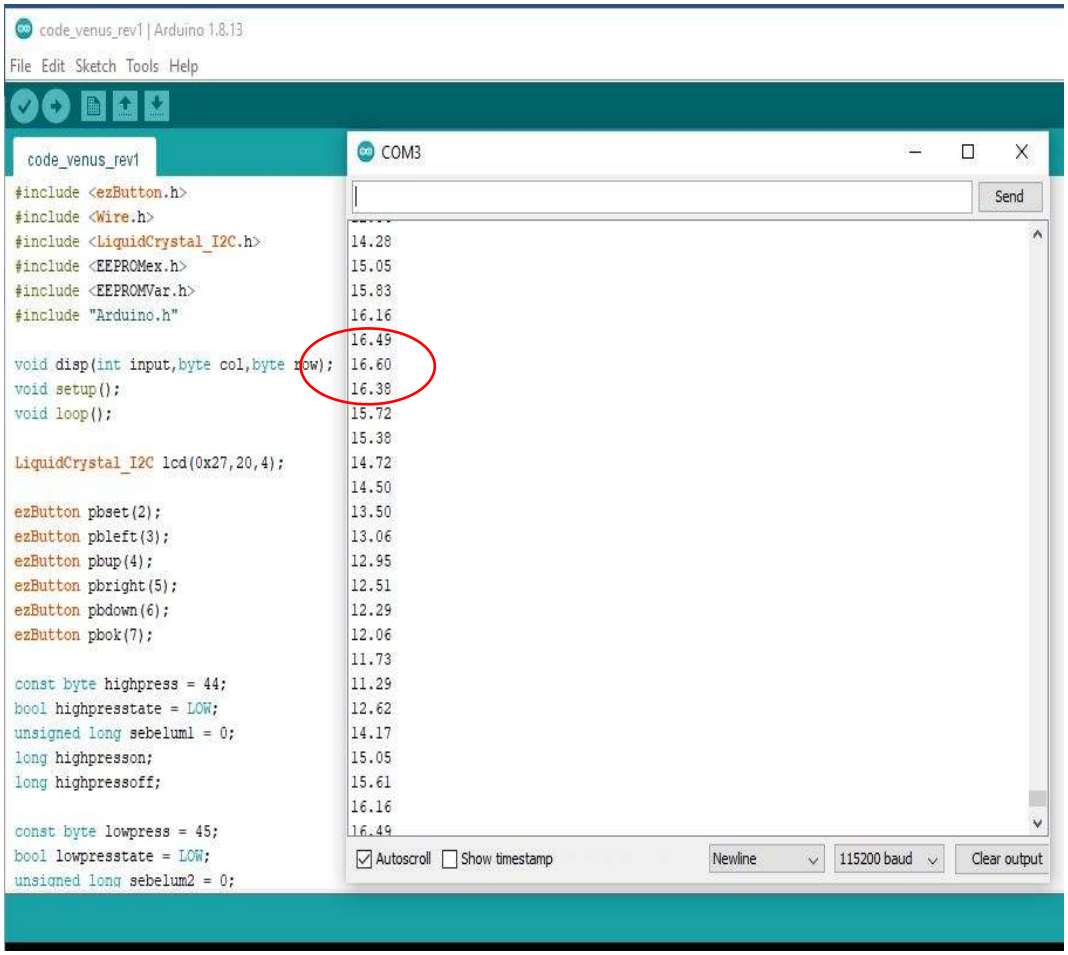
```
    }else if(pbok.isPressed()){
      EEPROM.writeInt(addr_tv,tv);
      lcd.setCursor(15,0);lcd.print("Saved");
      state = 1;
    }else {
      state = 1;
    }
    break;

case 2  :
    lcd.setCursor(0,0);lcd.print("Set BPM");
    lcd.setCursor(0,1);lcd.print("TV  :");disp(tv,5,1);
    lcd.setCursor(0,2);lcd.print("BPM :");disp(bpm,5,2);
    lcd.setCursor(0,3);lcd.print("IE  :1/");disp(ie,7,3);
    if (bpm>40){
      bpm = 40;
    }else if (bpm<8){
      bpm = 8;
    }else {
      bpm = bpm;
    }

    if(pbup.isReleased()){
      bpm++;
    }else if(pbdown.isReleased()){
      bpm--;
    }else if(pbleft.isReleased()){
      lcd.clear();
      state = 1;
    }else if(pbright.isReleased()){
      lcd.clear();
      state = 3;
    }else if(pbok.isPressed()){
      EEPROM.writeInt(addr_bpm,bpm);
      lcd.setCursor(15,0);lcd.print("Saved");
      state = 2;
    }else {
      state = 2;
    }
    break;

case 3  :
    lcd.setCursor(0,0);lcd.print("Set I/E");
    lcd.setCursor(0,1);lcd.print("TV  :");disp(tv,5,1);
    lcd.setCursor(0,2);lcd.print("BPM :");disp(bpm,5,2);
    lcd.setCursor(0,3);lcd.print("IE  :1/");disp(ie,7,3);
    if (ie>5){
      ie = 5;
```

```
      }else if (ie<1){
        ie = 1;
      }else {
        ie = ie;
      }

      if(pbup.isReleased()){
        ie++;
      }else if(pbdown.isReleased()){
        ie--;
      }else if(pbleft.isReleased()){
        lcd.clear();
        state = 2;
      }else if(pbright.isReleased()){
        lcd.clear();
        state = 4;
      }else if(pbok.isPressed()){
        EEPROM.writeInt(addr_ie,ie);
        lcd.setCursor(15,0);lcd.print("Saved");
        state = 3;
      }else {
        state = 3;
      }
      break;

case 4  :
      tv = EEPROM.readInt(addr_tv);
      bpm = EEPROM.readInt(addr_bpm);
      ie = EEPROM.readInt(addr ie);
      ie = EEPROM.readInt(addr_ie);
      perioda =(60.0/bpm)*1000.0; // dalam ms
      ton = (1/(1.0+ie))*perioda;
      toff = perioda - ton;
      highpresson = tv;
      highpressoff = perioda - tv;
      lowpresson = ton;
      lowpressoff = toff;
      lcd.setCursor(0,0);lcd.print("TV:");lcd.print(tv);lcd.print(" ");lcd.print
      lcd.setCursor(0,1);lcd.print("T   :");lcd.print(perioda);
      lcd.setCursor(0,2);lcd.print("ton :");lcd.print(ton);
      lcd.setCursor(0,3);lcd.print("toff:");lcd.print(toff);
      if (pbok.isReleased()){
        lcd.clear();
        state = 0;
      }else if (pbset.isReleased()){
        sebelum1=millis();
        sebelum2=millis();
        state = 5;
      }
      break;
```

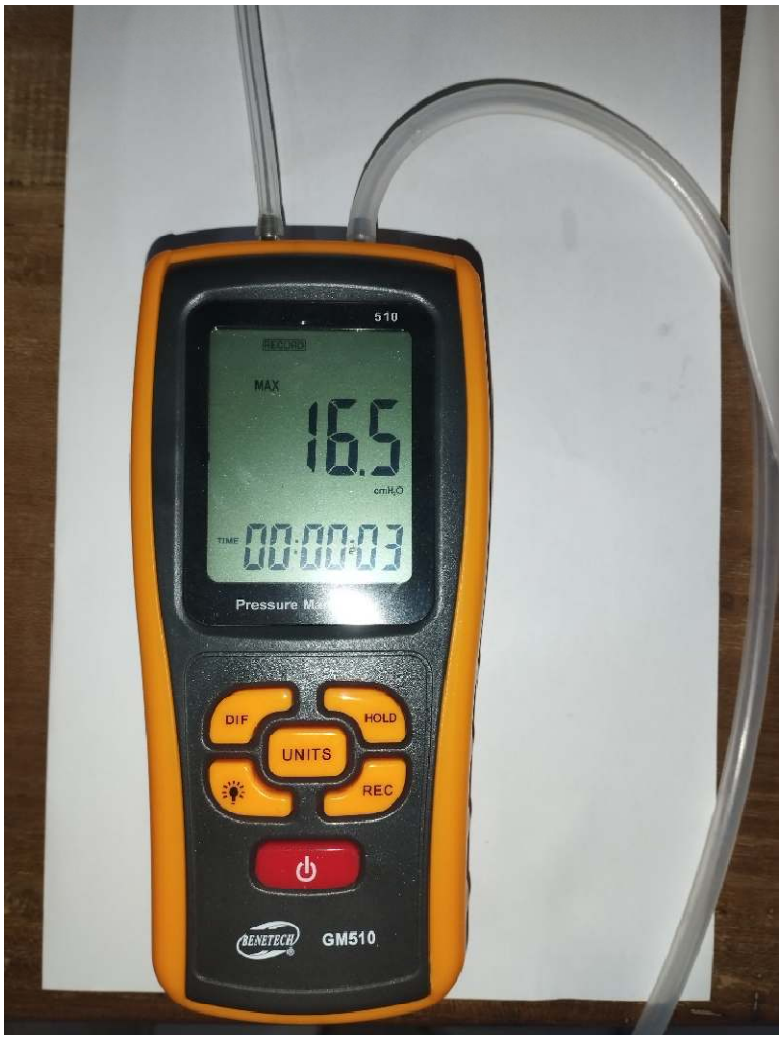
```
    case 5  :
              sekarang = millis();
              if((lowpresstate == HIGH) && (sekarang - sebelum2 >= lowpresson)){
                lowpresstate = LOW;
                sebelum2 = sekarang;
                digitalWrite(lowpress, lowpresstate);
              }else if ((lowpresstate == LOW) && (sekarang - sebelum2 >= lowpressoff)){
                lowpresstate = HIGH;
                sebelum2 = sekarang;
                digitalWrite(lowpress, lowpresstate);
              }
              if((lowpresstate == LOW) &&(highpresstate == HIGH) && (sekarang - sebelum1 >= highpresson)){
                highpresstate = LOW;
                sebelum1 = sekarang;
                digitalWrite(highpress, highpresstate);
              }else if ((lowpresstate == LOW) &&(highpresstate == LOW) && (sekarang - sebelum1 >= highpressoff)){
                highpresstate = HIGH;
                sebelum1 = sekarang;
                digitalWrite(highpress, highpresstate);
              }

              if(sekarang-sebelum3>=sampling){
                sebelum3 = sekarang;
                float deltaP = ((analogRead(A1) * ADC_mV - SensorOffset) / sensitivity * mmh2O_pa);     // result in Pa
                float Q = 5.42*sqrt(deltaP)/100;
                float cmh2o = (analogRead(A0) * ADC_mV - SensorOffset) / sensitivity / mmh2O_cmH2O;       // result in cmH2O
                lcd.setCursor(13,1);lcd.print("P:");lcd.print("     ");
                lcd.setCursor(15,1);lcd.print(cmh2o);
                lcd.setCursor(13,2);lcd.print("Q:");lcd.print("     ");
                lcd.setCursor(15,2);lcd.print(Q,3);
                lcd.setCursor(13,3);lcd.print(Q,3);
                Serial.println(cmh2o);
              }

              if (pbok.isPressed()){
                digitalWrite(highpress,LOW);
                digitalWrite(lowpress,HIGH);delay(5000);
                digitalWrite(lowpress,LOW);
                lcd.clear();
                state = 0;
              }
              break;
  }
}
```

Lampiran 2. Pengujian ventilator di RS. Dr. Wahidin Sudirohusodo

Lampiran 3. Pengukuran PIP dengan Sensor MPX 5010 DP



Lampiran 4. Pengukuran PIP dengan Manometer



Lampiran 5. Tabel hasil pengukuran tidal volume dan PIP

Solenoid Valve 1 *On* (s)	Volume Udara Terukur (ml)		*Peak Inspiratory Pressure* (cmH_2O)			
	Inspirasi	Ekspirasi	PEEP valve 5 cmH_2O	PEEP valve 10 cmH_2O	PEEP valve 15 cmH_2O	PEEP valve 20 cmH_2O
0.10	50	50	2.7	5.4	9.1	11.6
0.11	60	60	2.8	5.5	9.3	11.9
0.12	70	70	2.9	5.6	9.6	12.3
0.13	85	85	3.0	5.7	9.8	12.5
0.14	90	90	3.1	5.8	10.0	12.8
0.15	100	100	3.2	5.9	10.1	13.1
0.16	115	115	3.4	6.0	10.1	13.1
0.17	130	130	3.6	6.0	10.1	13.1
0.18	140	140	3.8	6.1	10.2	13.2
0.19	145	145	3.8	6.1	10.2	13.2
0.20	160	160	3.9	6.2	10.2	13.3
0.21	170	170	3.9	6.2	10.4	13.3
0.22	180	180	4.0	6.3	10.6	13.3
0.23	195	195	4.0	6.3	10.8	13.4
0.24	205	205	4.1	6.4	11.0	13.5
0.25	215	215	4.1	6.5	11.1	13.5
0.26	225	225	4.1	6.6	11.2	13.6
0.27	235	235	4.1	6.7	11.2	13.7
0.28	245	245	4.1	6.8	11.2	13.8
0.29	255	255	4.2	6.9	11.3	13.9
0.30	265	265	4.2	7.0	11.3	14.0
0.31	275	275	4.2	7.1	11.4	14.1
0.32	285	285	4.2	7.1	11.5	14.1
0.33	295	295	4.2	7.2	11.7	14.2
0.34	310	310	4.3	7.4	11.9	14.3
0.35	315	315	4.4	7.6	12.3	14.4
0.36	325	325	4.5	7.8	12.3	14.5
0.37	330	330	4.6	8.0	12.3	14.6
0.38	335	335	4.7	8.2	12.3	14.8
0.39	345	345	4.7	8.2	12.3	14.8
0.40	355	355	4.7	8.2	12.3	14.8
0.41	365	365	4.7	8.2	12.3	14.8
0.42	380	380	4.7	8.2	12.3	14.8
0.43	400	400	4.7	8.2	12.3	14.8
0.44	420	420	4.7	8.2	12.3	14.8

Lanjutan lampiran 5

Solenoid Valve 1 *On* (s)	Volume Udara Terukur (ml)		*Peak Inspiratory Pressure* (cmH_2O)			
	Inspirasi	Ekspirasi	PEEP valve 5 cmH_2O	PEEP valve 10 cmH_2O	PEEP valve 15 cmH_2O	PEEP valve 20 cmH_2O
0.45	440	435	4.8	8.2	12.4	15.0
0.46	455	450	4.8	8.2	12.5	15.1
0.47	465	460	4.8	8.2	12.5	15.1
0.48	475	475	4.8	8.2	12.6	15.1
0.49	480	480	4.8	8.2	12.6	15.1
0.50	490	485	4.8	8.3	12.6	15.1
0.51	505	500	4.9	8.3	12.6	15.1
0.52	515	515	5.0	8.4	12.6	15.1
0.53	530	525	5.1	8.4	12.6	15.1
0.54	545	545	5.2	8.5	12.6	15.1
0.55	565	465	5.3	8.6	12.6	15.1
0.56	575	475	5.3	8.7	12.6	15.1
0.57	585	480	5.3	8.8	12.6	15.1
0.58	590	490	5.3	8.9	12.6	15.1
0.59	610	605	5.3	9.0	12.8	15.2
0.60	620	620	5.4	9.0	13.0	15.3
0.61	630	625	5.4	9.0	13.2	15.4
0.62	650	645	5.5	9.0	13.2	15.4
0.63	665	665	5.5	9.0	13.2	15.4
0.64	675	670	5.6	9.0	13.2	15.4
0.65	685	685	5.6	9.0	13.3	15.5
0.66	700	690	5.6	9.0	13.4	15.6
0.67	715	710	5.6	9.0	13.5	15.7
0.68	725	720	5.7	9.0	13.6	15.7
0.69	740	740	5.7	9.0	13.6	15.7
0.70	750	745	5.7	9.1	13.6	15.8
0.71	765	760	5.9	9.1	13.6	15.9
0.72	775	770	5.9	9.1	13.7	16.1
0.73	785	780	5.9	9.2	13.7	16.2
0.74	805	805	5.9	9.2	13.7	16.3

Lampiran 6. Tabel hasil pengukuran PEEP

Volume Udara Terukur (ml)	*Positif end Expiratory Pressure* (cmH_2O)			
	PEEP *Valve* 5 cmH_2O	PEEP *Valve* 10 cmH_2O	PEEP *Valve* 15 cmH_2O	PEEP *Valve* 20 cmH_2O
50	0.9	2.9	6.3	7.6
60	0.9	2.9	6.4	7.9
70	0.9	3.0	6.5	8.3
85	1.0	3.1	6.7	8.6
90	1.0	3.1	6.9	8.8
100	1.0	3.1	7.0	9.2
115	1.1	3.2	7.0	9.2
130	1.1	3.2	7.0	9.2
140	1.1	3.3	7.1	9.3
145	1.1	3.3	7.1	9.3
160	1.1	3.3	7.1	9.3
170	1.1	3.3	7.1	9.3
180	1.1	3.3	7.2	9.3
195	1.1	3.3	7.2	9.3
205	1.1	3.3	7.2	9.3
215	1.1	3.3	7.2	9.3
225	1.1	3.3	7.2	9.3
235	1.1	3.3	7.2	9.3
245	1.1	3.3	7.3	9.3
255	1.1	3.3	7.3	9.4
265	1.1	3.3	7.3	9.4
275	1.1	3.3	7.3	9.4
285	1.1	3.3	7.3	9.4
295	1.1	3.4	7.5	9.5
310	1.1	3.5	7.7	9.5
315	1.1	3.6	7.7	9.5
325	1.1	3.7	7.7	9.5
330	1.1	3.8	7.7	9.5
335	1.1	3.8	7.7	9.5
345	1.1	3.8	7.7	9.5
355	1.1	3.8	7.7	9.5
365	1.1	3.8	7.7	9.5
380	1.1	3.8	7.7	9.5
400	1.1	3.8	7.7	9.5
420	1.1	3.8	7.7	9.5

Lanjutan lampiran 6

Volume Udara Terukur (ml)	*Positif end Expiratory Pressure* (cmH_2O)			
	PEEP *Valve* 5 cmH_2O	PEEP *Valve* 10 cmH_2O	PEEP *Valve* 15 cmH_2O	PEEP *Valve* 20 cmH_2O
440	1.1	3.8	7.8	9.5
455	1.1	3.8	7.8	9.5
465	1.1	3.8	7.8	9.5
475	1.1	3.8	7.8	9.5
480	1.1	3.8	7.8	9.5
490	1.1	3.8	7.8	9.5
505	1.1	3.8	7.8	9.5
515	1.1	3.8	7.8	9.5
530	1.1	3.8	7.8	9.5
545	1.1	3.8	7.8	9.5
565	1.1	3.9	7.8	9.5
575	1.1	3.9	7.8	9.5
585	1.1	3.9	7.8	9.6
590	1.1	3.9	7.8	9.7
610	1.1	3.9	7.8	9.8
620	1.1	3.9	7.8	9.9
630	1.1	3.9	7.8	9.9
650	1.1	3.9	7.9	10.0
665	1.1	3.9	7.9	10.0
675	1.1	3.9	7.9	10.0
685	1.1	3.9	7.9	10.0
700	1.1	3.9	7.9	10.0
715	1.1	3.9	7.9	10.0
725	1.1	3.9	7.9	10.0
740	1.1	3.9	7.9	10.2
750	1.1	3.9	7.9	10.4
765	1.1	3.9	7.9	10.6
775	1.1	3.9	7.9	10.8
785	1.1	3.9	7.9	10.9
805	1.1	3.9	7.9	10.9

Lampiran 7. Pengukuran PIP dan PEEP (RR dan I/E berbeda)

RR & I/E	V_T	PEEP Valve 5 cmH_2O		PEEP Valve 10 cmH_2O		PEEP Valve 15 cmH_2O		PEEP Valve 20 cmH_2O	
		PIP	PEEP	PIP	PEEP	PIP	PEEP	PIP	PEEP
10-1/1	805	6	0.4	9.5	2.2	14	6.1	16.3	8.6
10-1/2	805	5.8	0.4	9.5	1.8	13.2	5.9	16.3	8.3
10-1/3	805	5.8	0.4	9.6	2.0	13.5	5.9	16.5	7.3
10-1/4	805	6.0	0.6	9.8	2.6	13.9	6.5	16.4	7.5
10-1/5	805	5.8	0.4	9.4	2.0	13.8	6.0	16.4	7.5
20-1/1	805	5.9	1.1	9.2	3.9	13.7	7.9	16.3	10.9
20-1/2	805	6.4	1.6	9.5	3.9	13.9	8.0	16.6	10.8
20-1/3	805	6.0	1.3	9.4	3.8	13.7	8.3	16.5	10.8
20-1/4	805	6.2	1.6	9.8	4.3	14.2	8.9	16.2	10.2
20-1/5	805	6.5	1.7	9.7	4.0	14.3	8.4	16.4	10.5
30-1/1	805	5.5	1.2	9	4.6	13.4	9.7	15.9	11.7
30-1/2	805	5.7	1.8	9.1	4.9	13.5	9.8	15.8	12.3
30-1/3	805	5.9	2.4	9.6	5.4	13.7	9.7	16.2	12.4
30-1/4	805	6.1	2.6	10.0	5.8	13.8	9.9	16.1	12.5
30-1/5	805	5.9	2.6	9.6	5.6	13.3	9.1	16.1	12.2

Lampiran 8. Pengujian ventilator di BPFK

Lampiran 9. Data hasil pengujian tahap terakhir di BPFK Makassar

Solenoid Valve 1 *On* (detik)	V_T Terukur (ml)	*Peak Inspiratory Pressure* (cmH_2O)
0.10	81	5.8
0.20	235	13.3
0.30	357	17.8
0.40	409	19.1
0.50	432	20.8
0.60	599	26.6
0.70	621	34.1
0.80	638	42