

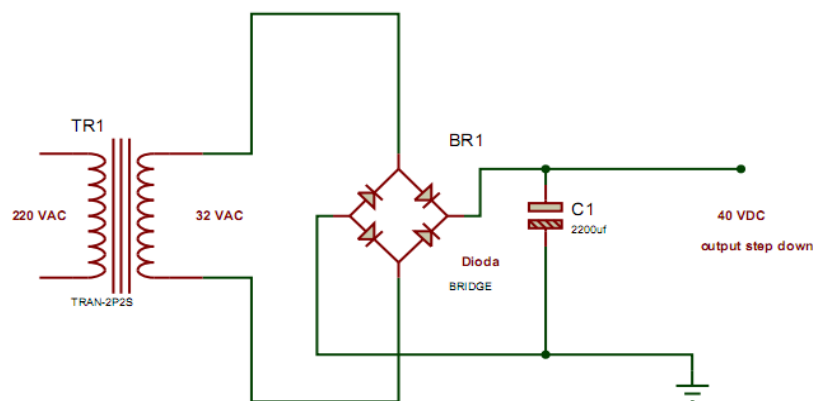
BAB IV

HASIL DAN PEMBAHASAN

4.1 Penjelasan Rangkaian

4.1.1 Rangkaian *Power Supply*

Berikut ini merupakan penjelasan dari rangkaian *power supply*:



Gambar 4.1 Rangkaian *Power Supply*

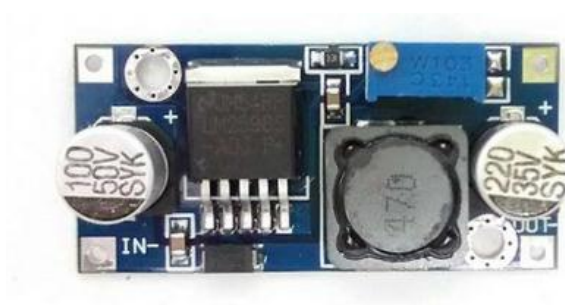
Pada rangkaian diatas menggunakan *transformator* berjenis *transformator AC step down*. Tegangan *input* dari PLN tersebut adalah 220 VAC dan tegangan *output* pada *transformator* adalah 32 VAC. Rangkaian diatas menggunakan dioda *bridge* yang digunakan untuk penyearah gelombang penuh dari tegangan AC menjadi tegangan DC. Dari dioda *bridge* dihubungkan dengan kapasitor, kapasitor digunakan untuk menyaring tegangan *outputan* tegangan DC dari tegangan AC.

4.1.2 Modul *Step down*

Berikut ini merupakan penjelasan dari modul *step down*:

Pada rangkaian *power supply* tegangan *output*-nya 40 VDC. Tetapi penulis memerlukan tegangan 30 VDC dan 5 VDC, sehingga penulis memerlukan modul

step down untuk menurunkan tegangan dari 40 VDC menjadi 30 VDC dan 5 VDC. Modul *step down* ini menggunakan IC LM2596. Dimana IC LM2596 adalah sirkuit terpadu/*integrated circuit* yang berfungsi sebagai *step down* DC *converter* dengan *current rating* 3A. Terdapat beberapa varian dari IC seri ini yang dapat dikelompokkan dalam dua kelompok yaitu versi *adjustable* yang tegangan keluarannya dapat diatur, dan versi *fixed voltage output* yang tegangan keluarannya sudah tetap / *fixed*. Pada modul diatas menggunakan seri IC *adjustable* yang tegangan keluarnya dapat diubah-ubah. Keunggulan modul *step down* LM2596 dibandingkan dengan *step down* tahanan resistor / potensiometer adalah besar tegangan *output* tidak berubah (stabil) walaupun tegangan input naik turun. Berikut merupakan gambar dari Modul *step down* LM2596.



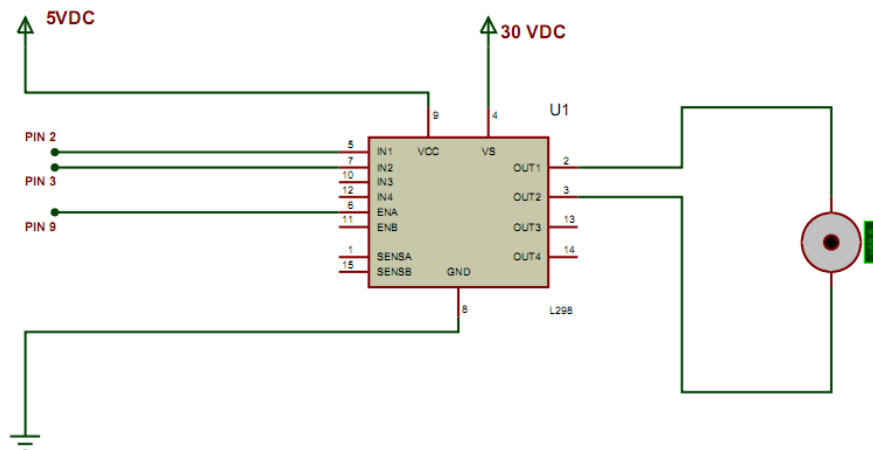
Gambar 4.2 Modul *step down* LM2596

4.1.3 Modul *Driver* Motor DC

Berikut ini merupakan penjelasan dari modul *driver* motor DC:

Modul *driver* motor mendapatkan *input* tegangan 30 VDC dari *power supply* untuk menghidupkan *micromotor* dan *input* tegangan 5 VDC dari *power supply* untuk menghidupkan IC L298N. Pin 1 dan 2 pada *driver* motor sebagai *input* arduino pada pin 2 dan 3 untuk mengatur pergerakan *micromotor* dan pin *enable* A pada *driver* motor sebagai *input* pin 9 pada arduino untuk mengatur

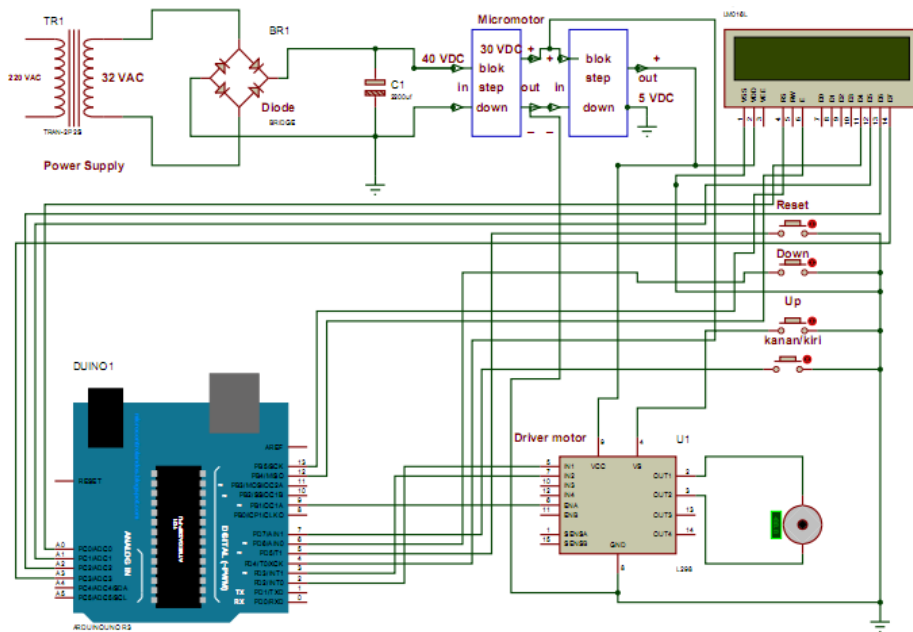
kecepatan *micromotor* dengan mengatur PWM pada arduino. Pin *output* 1 dan *output* 2 pada *driver* motor digunakan sebagai *input* bagi *micromotor*. Berikut ini merupakan gambar dari Rangkaian modul *driver* motor.



Gambar 4.3 Rangkaian modul *driver* motor

4.1.4 Rangkaian Keseluruhan

Berikut ini merupakan penjelasan dari rangkaian keseluruhan alat:



Gambar 4.4 Rangkaian Keseluruhan

Rangkaian *power supply* menggunakan *transformator step down*. *Input* dari *transformator* adalah 220 VAC langsung dari PLN. *Output* dari *transformator* adalah 32 VAC. Penulis memerlukan tegangan 5 VDC dan 30 VDC, sehingga *transformator* dirangkai dengan dioda *bridge* dan kapasitor. Dioda *bridge* digunakan untuk penyearah gelombang penuh untuk menjadikan tegangan AC menjadi tegangan DC dan kapasitor digunakan untuk menyaring frekuensi *outputan* tegangan DC dari *input* tegangan AC. Dari rangkaian ini masuk ke dalam modul *step down*, modul *step down* menggunakan IC LM2596. Dimana modul *step down* ini mempunyai *input* tegangan sampai 40VDC dan *output* dapat di *setting* sesuai keperluan yaitu antara 1,2 – 37 VDC. Arus keluaran pada modul ini adalah 3 *ampere*. Penulis mengatur *output* dari modul *step down* tegangan 5 VDC dan tegangan 30 VDC. Tegangan 5 VDC digunakan untuk *input* pada arduino, IC L298N dan LCD. Sedangkan, tegangan 30 VDC digunakan untuk men-*supply micromotor*.

Driver motor digunakan untuk mengontrol *micromotor* yang diatur oleh arduino. Didalam *driver* motor terdapat IC L298N yang digunakan untuk mengatur kecepatan dan putaran *micromotor*. Terdapat 6 pin *input* yaitu untuk arduino dan 4 pin *output* untuk motor. Disini penulis menggunakan 3 pin *input* untuk arduino dan 2 pin *output micromotor*. Kemudian pengaturan kecepatan *micromotor* dan putaran *micromotor* dengan menggunakan tombol *up/ down* dan tombol kanan / kiri. Tombol kanan / kiri diletakkan pada pin 7 pada arduino, sedangkan tombol *up / down* diletakkan pada pin 4 dan 6 dan menggunakan tombol *reset* yang diletakkan pada pin 5 pada arduino. Tombol-tombol inilah yang

menjadi *input* dari modul arduino, dimana semua perintah adalah *output* dari modul arduino yang telah diprogram untuk mengontrol pergerakan *micromotor*.

4.2 Pembahasan Program

```
#include <L298N.h>
#include <LiquidCrystal.h>
```

Gambar 4.5 *Listing* Program Memasukkan *Library*

Memasukkan *library* L298N untuk mendefinisikan *driver* motor agar dapat diprogram dalam arduino dan memasukkan *library* LCD sebagai penampil kecepatan dan arah putaran *micromotor*.

```
const int ENA = 9;
const int IN1 = 2;
const int IN2 = 3;

L298N driver (ENA, IN1, IN2, 0, 0, 0);
```

Gambar 4.6 *Listing* Program Inisialisasi *Driver* Motor L298N

Dalam *driver* motor penulis menggunakan variabel pin ENA, IN1, dan IN2 pada *driver* motor L298N untuk dimasukkan pada *input* pin pada *board* arduino agar L298N dapat diprogram untuk menggerakkan *micromotor*. Dimana *input driver* motor merujuk pada pin 2, 3 dan 9 arduino. Sehingga pin-pin tersebut di tulis pada *sketches* seperti diatas.

```
const int pinBt1 = 6; //down
const int pinBt2 = 4; //up
const int pinBt3 = 7; //kanan & kiri
const int pinBt4 = 5; // reset
```

Gambar 4.7 *Listing* Program Inisialisasi *Push Button*

Dalam modul penulis menggunakan 4 tombol untuk menjalankan perintah. Variabel `pinBt1` merupakan konstanta dalam bentuk *integer* yang merujuk pada pin 6 *board* arduino digunakan sebagai tombol *down* untuk mengurangi kecepatan *micromotor*. Variabel `pinBt2` merupakan konstanta dalam bentuk *integer* yang merujuk pada pin 4 *board* arduino digunakan sebagai tombol *up* untuk menambah kecepatan *micromotor*. Variabel `pinBt3` merupakan konstanta dalam bentuk *integer* yang merujuk pada pin 7 *board* arduino digunakan sebagai tombol kanan/kiri untuk mengatur putaran *micromotor* putar kanan atau putar kiri. Sedangkan, Variabel `pinBt4` merupakan konstanta dalam bentuk *integer* yang merujuk pada Pin 5 *board* arduino digunakan sebagai tombol *reset* digunakan untuk mengembalikan kecepatan *micromotor* pada 0 rpm.

```

lcd.begin(16,2);
pinMode(pinBt1, INPUT);
pinMode(pinBt2, INPUT);
pinMode(pinSPEED, OUTPUT);
pinMode(pinBt3, INPUT);

digitalWrite(pinBt1, HIGH);
digitalWrite(pinBt2, HIGH);
digitalWrite(pinBt3, HIGH);

```

Gambar 4.8 *Listing* Program inialisasi *input* dan *output Push Button*

LCD digunakan sebagai penampil kecepatan *micromotor* dengan menulis coding `lcd.begin(16,2)` karena menggunakan modul LCD 2X16.

Fungsi `pinMode` digunakan untuk memberitahu status `pinBt1`, `pinBt2`, dan `pinBt3` bahwa sebagai *input*. Sehingga, *microcontroller* tidak akan membaca logika pin tersebut tetapi hanya akan menulis logika tersebut. Demikian, fungsi `pinMode` pada logika `pinSPEED`, tetapi pada logika

`pinSPEED` dijadikan sebagai *output*. `pinSPEED` merupakan variabel pin ENA yang digunakan untuk output sinyal PWM pada sebagai pengatur kecepatan putaran *micromotor*.

Tombol *up* / *down* dan kanan/kiri merupakan *input* untuk mengatur kecepatan dan arah putaran *micromotor*, maka fungsi `digitalWrite` harus diberi nilai `HIGH` atau nilai dari 1 yang artinya jika tombol ditekan maka tombol tersebut akan berfungsi. Fungsi `digitalWrite` merupakan *output* yang bernilai digital, dimana fungsi `digitalWrite` mempunyai nilai `HIGH` dan *low*. `HIGH` merupakan nilai dari 1 atau mendapatkan tegangan 5 volt yang artinya jika tombol ditekan maka *micromotor* akan hidup atau aktif. Sedangkan, jika *low* merupakan nilai 0 atau mendapatkan tegangan 0 volt yang artinya jika ditekan *micromotor* akan mati atau pasif. Oleh karena itu tombol *up* / *down* dan tombol kanan/kiri mempunyai nilai `HIGH` yang artinya jika ditekan tombol *up* / *down* dan tombol kanan / kiri akan hidup atau aktif untuk menjalankan perintah.

```

if (digitalRead(pinBt1)==LOW)
{
    SPEED--;
}

```

Gambar 4.9 Listing Program Tombol *Down*

`pinBt1` adalah variabel yang digunakan untuk tombol *down*. Tombol *down* merupakan tombol yang digunakan untuk menurunkan kecepatan *micromotor*. Tombol *down* diberi nilai *low* pada fungsi `digitalRead`, karena tombol *down* juga tidak akan langsung aktif. Tombol akan aktif jika ditekan terlebih dahulu dan fungsi *digital write* bernilai `HIGH` untuk mengaktifkannya.

Besar nilai kecepatan *micromotor* yang diatur tombol *down* ditampilkan pada LCD di kolom 0 dan di baris 1. `lcd.print` merupakan perintah yang digunakan untuk menampilkan tulisan berapa kecepatan micromotor berkurang saat tombol *down* ditekan.

```
else if (digitalRead(pinBt2)==LOW)
{
    SPEED++;
}
```

Gambar 4.10 *Listing Program Tombol up*

Fungsi `digitalRead` merupakan input yang bernilai digital untuk menyalakan *micromotor*. Fungsi `digitalRead` mempunyai nilai *low* dan *high*. *Low* merupakan nilai 0 atau mendapatkan tegangan 0 -1,5 volt yang artinya tombol tidak akan langsung aktif. Sedangkan, *HIGH* merupakan nilai 1 atau tombol akan mendapatkan tegangan 5-3 volt yang artinya langsung aktif. `pinBt2` adalah variabel yang digunakan untuk inisial tombol *up*.

Tombol *up* merupakan tombol yang digunakan untuk menambah kecepatan *micromotor* setiap kali ditekan. Jadi, fungsi `digitalRead` diberi nilai *low* karena tombol *up* tidak langsung aktif. Tombol *up* aktif jika ditekan terlebih dahulu dan pada fungsi `digitalWrite` bernilai *HIGH*. Oleh karena itu tombol *up* mempunyai nilai *low* pada fungsi `digitalRead` karena tombol tidak langsung aktif, tetapi tombol akan aktif setiap di tekan dan fungsi *digital write* diberi nilai *HIGH*. Perubahan kecepatan yang bertambah saat ditekan tombol *up* ditampilkan pada LCD dengan letak di kolom 0 dan di baris 1. `lcd.print` merupakan

perintah yang akan menampilkan tulisan berapa kecepatan yang akan bertambah saat tombol *up* ditekan.

```

if (digitalRead(pinBt3)==LOW)
{
  driver.turn_right((SPEED*100/255),time_delay)
  digitalWrite(pinSPEED,LOW);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Putar Kanan ");
  lcd.setCursor(0,1);
  lcd.print("Speed = ");
  lcd.print(persenSpeed);
}
else
{
  driver.turn_left((SPEED*100/255),time_delay)
  digitalWrite(pinSPEED,LOW);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Putar Kiri ");
  lcd.setCursor(0,1);
  lcd.print("Speed = ");
  lcd.print(persenSpeed);
}

```

Gambar 4.11 *Listing* Program Tombol Kanan/ Kiri

`pinBt3` merupakan variabel dari tombol kanan / kiri. Tombol kanan / kiri digunakan untuk mengatur arah putaran *micromotor*. `driver.turn_right` adalah coding yang digunakan pada arduino untuk memerintah *driver* motor L298N. Jadi, saat `pinBt3` ditekan motor akan berputar ke arah kanan. Sedangkan, `driver.turn_left` merupakan coding yang digunakan arduino untuk memerintah *driver* motor L298N agar `pinBt3` saat ditekan motor akan berputar ke arah kiri. Variabel `pinSPEED` merupakan nama dari ENA yang digunakan untuk *input driver* motor L298N dan juga *output* untuk mengatur PWM yang digunakan setiap besar nilai kecepatan *micromotor* yang dipakai oleh operator.

`lcd.clear` digunakan untuk menghapus karakter secara langsung pada tulisan putar kanan dan putar kiri di LCD. `lcd.setCursor` digunakan untuk letak posisi tulisan putar kanan dan putar kiri pada posisi koordinat (0,0) dan letak posisi kecepatan *micromotor* pada posisi koordinat (0,1). `lcd.print` digunakan untuk menulis karakter putar kanan / kiri dan menuliskan karakter kecepatan *micromotor* sesuai keinginan operator.

```

if (digitalRead(pinBt4)==LOW)
{
    SPEED = 0;
    lcd.clear();
    lcd.setCursor(0,1);
    digitalWrite(pinBt4, HIGH);
}

```

Gambar 4.12 *Listing Program Tombol Reset*

`pinBt4` merupakan variabel dari tombol *reset*. Tombol *reset* digunakan untuk mengembalikan kecepatan *micromotor* ke 0 rpm saat selesai digunakan. Jadi saat menggunakan *micromotor* dengan posisi awalnya menggunakan kecepatan dengan nilai yang besar tidak harus menurunkan ke 0 rpm dengan menekan tombol *down* terus menerus cukup dengan menekan tombol *reset* ia akan kembali pada 0 rpm. Perintah tombol *reset* ini ditampilkan pada layar LCD di kolom 0 dan di baris 1. Saat tombol *reset* ditekan akan muncul tulisan 0 diperintah oleh `lcd.print`. Fungsi `digitalWrite` bernilai `HIGH` karena saat ditekan ia langsung aktif untuk menjalankan perintah.

```

analogWrite(pinSPEED, SPEED);
persenSpeed = (SPEED*100)/255;//tampilan
delay(30);
{
if (persenSpeed<=10)
{
    persenSpeed=5000;
}
if (persenSpeed<=15)
{
    persenSpeed=10000;
}
if (persenSpeed<=22)
{
    persenSpeed=15000;
}
if (persenSpeed<=35)
{
    persenSpeed=20000;
}
if (persenSpeed<=68)
{
    persenSpeed=25000;
}
}

```

Gambar 4.13 *Listing* program tampilan

`analogWrite` digunakan untuk membaca nilai putaran dan kecepatan *micromotor* yang digunakan sesuai *settingan* yang diinginkan operator. `persenSpeed` digunakan untuk penampil kecepatan pada LCD yang di *setting* operator sesuai kebutuhan. `persenSpeed` diatas diatur dari 5000 – 25000 rpm dengan kenaikan 5000/rpm.

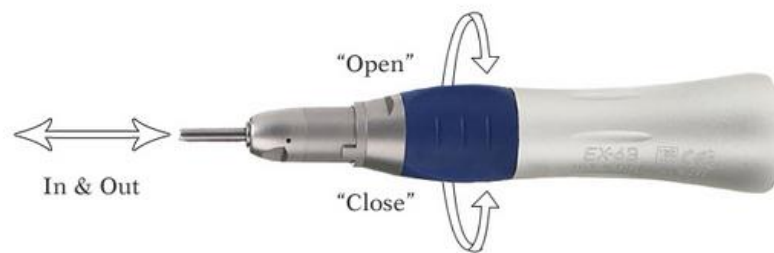
4.3 Standar Operasional Pemakaian (SOP)

4.3.1 Persiapan Alat

a. Petunjuk Pemasangan mata bor

1. Buka kunci dengan memutar bagian warna biru pada *chuck ring* kearah kiri.

2. Masukkan mata bor kedalam *handpiece micromotor*, mata bor ketika dimasukkan sambil diputar.
3. Setelah selesai memasukkan mata bor putar kembali bagian biru pada *chuck ring* kearah kanan.



Gambar 4.14 Cara memutar *chuck ring*

- b. Masukkan kabel *handpiece micromotor* kedalam colokan *micromotor* pada *box micromotor*.
- c. Colokkan *footswitch* pada konektor *footswitch* sebagai penggerak *micromotor*.
- d. Tempatkan *dental micromotor* sedekat mungkin dengan operator.
- e. *Setting dental micromotor* sesuai kebutuhan.

4.3.2 Penggunaan Alat

- a. Colokkan kabel *power AC* pada tegangan 220 VAC.
- b. Tekan tombol *power ON/OFF*.
- c. Tekan tombol pemilihan arah putaran yang diinginkan. Putar kanan atau putar kiri (tombol biru). Jangan menekan tombol pemilihan putaran pada saat *micromotor* sedang bergerak / digunakan karena dapat menyebabkan kerusakan pada *driver motor*.

- d. *Setting* kecepatan *micromotor* sesuai kehendak dengan menekan tombol *up*(tombol kuning) untuk menambah kecepatan *micromotor* dan tombol *down* (tombol hijau) untuk mengurangi kecepatan *micromotor*.
- e. Tekan tombol *reset* (tombol merah) untuk berhenti.
- f. Setelah selesai tindakan tekan tombol *power On/Off* untuk mematikan *dental micromotor* dan rapikan alat pada tempat semula.

4.4 Hasil Penelitian

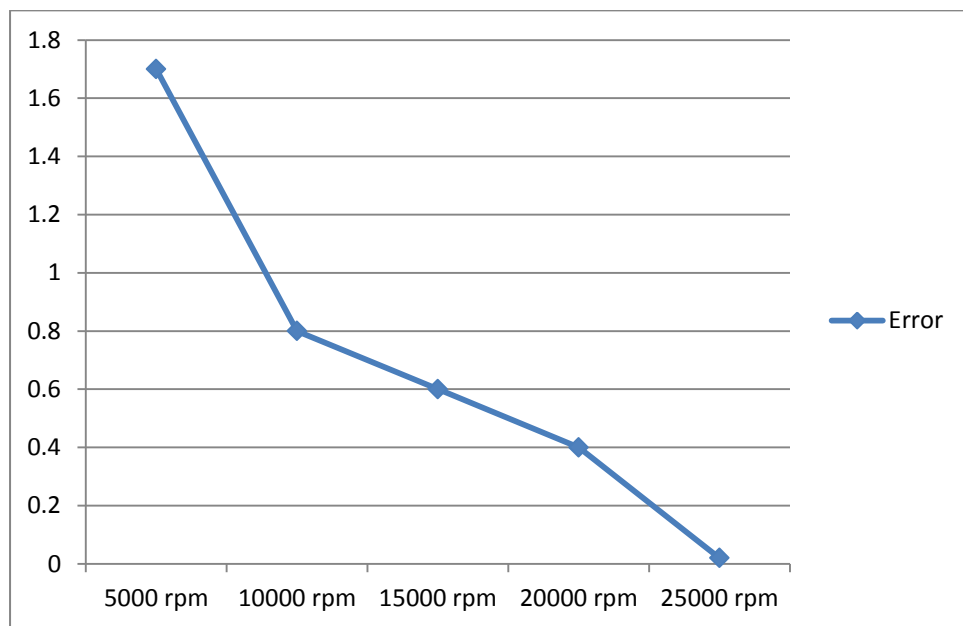
4.4.1 Pengukuran Putaran Kanan

Tabel 4.1 Hasil Pengukuran Modul pada Putaran Kanan

No	Hasil <i>Setting</i> Tachometer (Rpm)				
	5000	10000	15000	20000	25000
1	4997	10230	15053	20034	25043
2	4992	10105	15142	20054	24973
3	5179	10063	15078	20052	24974
4	5190	10024	15118	20067	25019
5	5188	10056	15132	20073	24992
6	5150	10094	15145	20080	24987
7	5108	10072	15035	20044	24959
8	5138	10081	15096	20134	24973
9	5148	10178	15123	20127	24926
10	5169	10020	15001	20166	24950
11	5145	9979	14965	20172	25044
12	5031	10043	15146	19970	25025
13	5022	10090	15117	20070	24997
14	5080	10162	15102	20035	24972
15	5054	10115	15126	20051	24971
16	5022	10110	15088	20127	25019
17	5004	10115	15178	20086	24992
18	5028	10067	15160	20123	25030
19	5017	10039	14982	20126	25015
20	5027	10037	14953	19925	25023
Rata-Rata	5084.45	10084	15087	20075.8	24994.2
Error	1.7%	0.8%	0.6%	0.4%	0.02%

a. Grafik putaran kanan

Grafik nilai *error* dengan *setting* kecepatan pada putaran kanan dapat dilihat pada gambar grafik dibawah ini,



Gambar 4.15 Grafik *setting* putaran kanan

b. Analisa Perhitungan Putaran Kanan

1. Rata-rata Data Modul

a) Rata-rata pada 5000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{4997+4992+5179+5190+5188+5150+5108+5138+5148+5169+5145+5031+5022+5080+5054+5022+5004+5028+5017+5027}{20}$$

$$(\bar{X}) = \frac{101689}{20}$$

$$(\bar{X}) = 5084,45$$

b) Rata-rata pada 10000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{10230+10105+10063+10024+10056+10094+10072+10081+10178+10020+9978+10043+10090+10162+10115+10110+10115+10067+10039+10037}{20}$$

$$(\bar{X}) = \frac{201680}{20}$$

$$(\bar{X}) = 10084$$

c) Rata-rata pada 15000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{15053+15142+15078+15118+15132+15145+15035+15096+15123+15001+14965+15146+15117+15102+15126+15088+15178+15160+14982+14953}{20}$$

$$(\bar{X}) = \frac{301740}{20}$$

$$(\bar{X}) = 15087$$

d) Rata-rata pada 20000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{20034+20054+20052+20067+20073+20080+20044+20134+20127+20166+20172+19970+20070+20035+20051+20127+20086+20123+20126+19925}{20}$$

$$(\bar{X}) = \frac{401516}{20}$$

$$(\bar{X}) = 20075,8$$

e) Rata-rata pada 25000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{25043+24973+24974+25019+24992+24987+24959+24973+24926+24950+25044+25025+24997+24972+24971+25019+24992+25030+25015+25023}{20}$$

$$(\bar{X}) = \frac{499884}{20}$$

$$(\bar{X}) = 24994,2$$

2. Error

a) 5000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{5084,45 - (5000)}{5084,45} \times 100\%$$

$$\%Error = \frac{84,45}{5084,45} \times 100\%$$

$$\%Error = 0,0166 \times 100\%$$

$$\%Error = 1,7\%$$

b) 10000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{10084 - (10000)}{10084} \times 100\%$$

$$\%Error = \frac{84}{10084} \times 100\%$$

$$\%Error = 0,0083 \times 100\%$$

$$\%Error = 0,8\%$$

c) 15000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{15087 - (15000)}{15087} \times 100\%$$

$$\%Error = \frac{87}{15087} \times 100\%$$

$$\%Error = 0,0057 \times 100\%$$

$$\%Error = 0,6\%$$

d) 20000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{20075,8 - (20000)}{20075,8} \times 100\%$$

$$\%Error = \frac{75,8}{20075,8} \times 100\%$$

$$\%Error = 0,0037 \times 100\%$$

$$\%Error = 0,4\%$$

e) 25000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{24994,2 - (25000)}{24994,2} \times 100\%$$

$$\%Error = \frac{-5,8}{24994,2} \times 100\%$$

$$\%Error = 0,0002 \times 100\%$$

$$\%Error = 0,02\%$$

3. Rata-rata *Error* Keseluruhan Pada Putaran Kanan

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{1,7\% + 0,8\% + 0,6\% + 0,4\% + 0,02\%}{5}$$

$$(\bar{X}) = 0,7\%$$

c. Uraian Hasil Pengukuran

Tabel 4.1 diatas merupakan tabel hasil data perbandingan pengukuran rpm pada posisi putaran kanan dengan menggunakan tachometer. Tachometer merupakan alat yang digunakan sebagai pengukur kecepatan putaran motor. Dimana pada modul ini pengukuran putaran yang dilakukan mulai dari 5000 rpm-25000 rpm dengan kenaikan 5000 rpm. Pada *setting* putaran 5000 kecepatannya mulai dari 0-5000 rpm. *Setting* 10000 rpm mulai dari 5000-10000 rpm. *Setting* 15000 rpm mulai 10000-15000 rpm. *Setting* 20000 rpm mulai dari 15000-20000 rpm. Dan *setting* 25000 mulai dari 20000-25000 rpm. Tetapi, pada pengambilan data penulis mengambil data *setting* nilai putaran yang maksimal. Sehingga, data yang diambil hanya 5000 rpm, 10000 rpm, 15000 rpm, 20000 rpm, dan 25000 rpm.

Gambar 4.15 grafik putaran kanan diatas menunjukkan bahwa nilai *error* setiap *setting* berbeda-beda. Pada kecepatan 5000 rpm mengalami nilai *error* yang cukup besar yaitu 1,7%. Nilai *error* semakin menurun setiap *setting* kecepatannya naik. Jadi, kecepatan rpm yang tinggi baik digunakan.

Nilai *error* yang dihasilkan pada putaran kanan tidak melebihi ambang batas nilai *error*, karena nilai maksimal *error* dari rpm adalah 10%. Tetapi pada *setting* putaran di 5000 rpm mempunyai nilai *error*

tertinggi yaitu 1,7%. Sedangkangkan, pada putaran diatas 5000 rpm nilai *error*nya dibawah 1%. Penyebab *error* yang tidak stabil dikarenakan *human error*, peletakkan lempengan untuk pengukuran yang kurang stabil, dan kurang presisinya antara tachometer dan *micromotor*. Dari data yang dihasilkan tersebut, bahwa putaran kanan pada 5000 rpm kurang efisien untuk digunakan sedangkan diatas 5000 rpm baik digunakan. Tetapi, secara keseluruhan putaran kanan layak digunakan untuk pembentukan dan pemolesan *protesa* gigi karena rata-rata *error* dari data yang dihasilkan 0.7%.

4.4.2 Pengukuran Putaran Kiri

Berikut ini merupakan tabel hasil pengukuran modul pada putaran kiri yang ditunjukkan pada tabel 4.2.

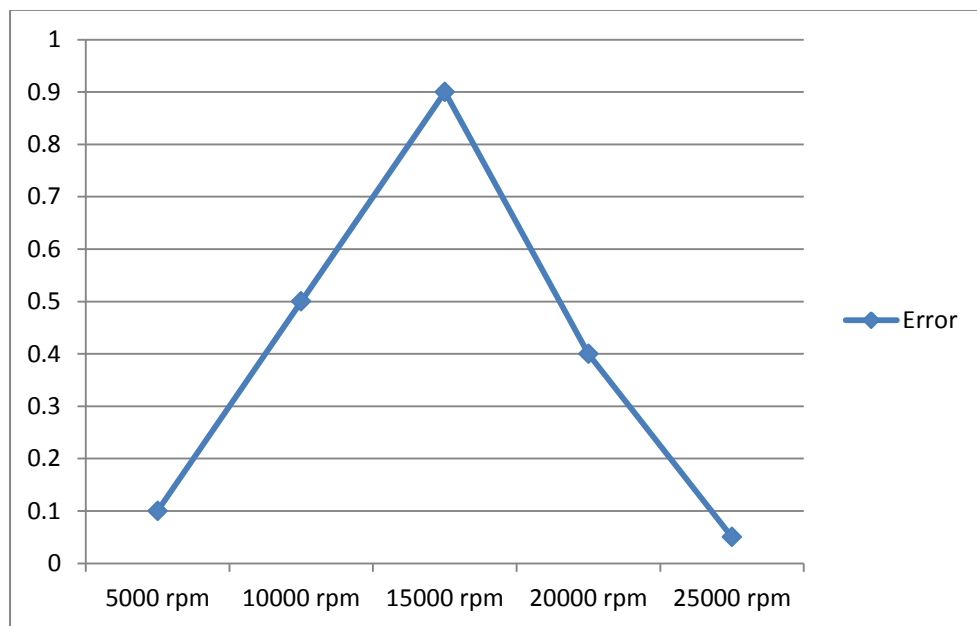
Tabel 4.2 Hasil Pengukuran Modul pada Putaran Kiri

No	Hasil Setting Tachometer (Rpm)				
	5000	10000	15000	20000	25000
1	4910	10040	15082	20082	24933
2	4905	10053	15072	20122	24924
3	4920	9982	15050	20135	24991
4	4930	10050	15135	20049	24993
5	4970	10080	15059	20027	25025
6	4990	10023	15140	20011	25019
7	5044	10065	15177	20032	24991
8	5035	10054	15098	20055	24970
9	4990	10099	15179	20051	25020
10	5007	10031	15124	20020	25006
11	5049	10071	15170	20013	24960
12	5001	10053	15271	20096	24933
13	4973	10224	15218	20124	25003
14	4933	10005	15214	20113	24981
15	5048	10018	15217	20070	24984
16	4971	10001	15099	20023	24983
17	5031	10002	15120	20165	24967
18	5069	10048	15122	20103	24985

19	5058	10024	15055	20083	25025
20	5030	10028	15152	20087	25012
Rata-rata	4993.2	10047.5	15137.7	20073.05	24985.25
Error	0.1%	0.5%	0.9%	0.4%	0.05%

a. Grafik Putaran Kiri

Grafik nilai *error* dengan *setting* kecepatan pada putaran kiri dapat dilihat pada gambar grafik dibawah ini,



Gambar 4.16 Grafik *Setting* Putaran Kiri

a. Analisa Perhitungan Putaran Kiri

1. Rata-rata Data Modul

a) Rata-rata pada 5000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{4910+4905+4920+4930+4970+4990+5044+5035+4990+5007+5049+5001+4973+4933+5048+4971+5031+5069+5058+5030}{20}$$

$$(\bar{X}) = \frac{99864}{20}$$

$$(\bar{X}) = 4993,2$$

b) Rata-rata pada 10000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{10040+10053+9982+10050+10080+10023+10065+10054+10099+10031+10071+10053+10224+10005+10018+10001+10002+10048+10024+10028}{20}$$

$$(\bar{X}) = \frac{200951}{20}$$

$$(\bar{X}) = 10047,55$$

c) Rata-rata pada 15000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{15082+15072+15050+15135+15059+15140+15177+15098+15179+15124+15170+15271+15218+15214+15217+15099+15120+15122+15055+15152}{20}$$

$$(\bar{X}) = \frac{302754}{20}$$

$$(\bar{X}) = 15137,7$$

d) Rata-rata pada 20000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{20082+20122+20135+20049+20027+20011+20032+20055+20061+20020+20013+20096+20124+20113+20070+20023}{20}$$

$$+20165+20103+20083+20087$$

$$(\bar{X}) = \frac{401461}{20}$$

$$(\bar{X}) = 20073,05$$

e) Rata-rata pada 25000 rpm

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{24933+24924+24991+24993+25025+25019+24991+24970+25020+25006+24960+24933+25003+24981+24984+24983+24967+24985+25025+25012}{20}$$

$$(\bar{X}) = \frac{499705}{20}$$

$$(\bar{X}) = 24985,25$$

2. Error

a) 5000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{4993,2 - (5000)}{4993,2} \times 100\%$$

$$\%Error = \frac{-6,8}{4993,2} \times 100\%$$

$$\%Error = -0,0013 \times 100\%$$

$$\%Error = 0,1\%$$

b) 10000 rpm

$$\%Error = \frac{Xn - (\bar{X})}{Xn} \times 100\%$$

$$\%Error = \frac{10047,5 - (10000)}{10047,5} \times 100\%$$

$$\%Error = \frac{47,5}{10047,5} X 100\%$$

$$\%Error = 0,0047 X 100\%$$

$$\%Error = 0,5\%$$

c) 15000 rpm

$$\%Error = \frac{X_n - (\bar{X})}{X_n} X 100\%$$

$$\%Error = \frac{15137,7 - (15000)}{15137,7} X 100\%$$

$$\%Error = \frac{137,7}{15137,7} X 100\%$$

$$\%Error = 0,0090 X 100\%$$

$$\%Error = 0,9\%$$

d) 20000 rpm

$$\%Error = \frac{X_n - (\bar{X})}{X_n} X 100\%$$

$$\%Error = \frac{20073,05 - (20000)}{20073,05} X 100\%$$

$$\%Error = \frac{73,05}{20073,05} X 100\%$$

$$\%Error = 0,0036 X 100\%$$

$$\%Error = 0,4\%$$

e) 25000 rpm

$$\%Error = \frac{X_n - (\bar{X})}{X_n} X 100\%$$

$$\%Error = \frac{24985,25 - (25000)}{24985,25} X 100\%$$

$$\%Error = \frac{-14,74}{24985,25} X 100\%$$

$$\%Error = - 0,0005 X 100\%$$

$$\%Error = 0,05\%$$

3. Rata- rata *Error* Keseluruhan Pada Putaran Kiri

$$(\bar{X}) = \frac{\sum Xi}{n}$$

$$(\bar{X}) = \frac{0,1\% + 0,5\% + 0,9\% + 0,4\% + 0,05\%}{5}$$

$$(\bar{X}) = 0,4\%$$

a. Uraian Hasil Pengukuran

Tabel 4.2 diatas merupakan tabel hasil data perbandingan pengukuran rpm pada posisi putaran kiri dengan menggunakan tachometer. Tachometer merupakan alat yang digunakan sebagai pengukur kecepatan motor. Dimana pada modul ini pengukuran putaran yang dilakukan mulai dari 5000 rpm-25000 rpm dengan kenaikan 5000 rpm. Pada *setting* putaran 5000 kecepatannya mulai dari 0-5000 rpm. *Setting* 10000 rpm mulai dari 5000-10000 rpm. *Setting* 15000 rpm mulai 10000-15000 rpm. *Setting* 20000 rpm mulai dari 15000-20000 rpm. Dan *setting* 25000 mulai dari 20000-25000 rpm. Tetapi, pada pengambilan data penulis mengambil data *setting* nilai putaran yang maksimal. Sehingga, data yang diambil hanya 5000 rpm, 10000 rpm, 15000 rpm, 20000 rpm, dan 25000 rpm.

Gambar 4.16 grafik diatas menunjukkan nilai *error* yang tidak stabil. Pada kecepatan 10000 rpm dan 15000 rpm mengalami kenaikan nilai *error* yaitu 0.5% dan 0.9%.. Tetapi, kenaikan nilai *error* ini masih

dapat digunakan dengan baik. Karena, kenaikan nilai *error* masih dalam batas ketentuan nilai *error*.

Nilai *error* yang dihasilkan pada putaran kiri tidak melebihi batas ketentuan nilai *error*, karena nilai maksimal *error* dari rpm adalah 10%. Putaran kiri pada modul menghasilkan nilai *error* yang sedikit walaupun kurang stabil. Penyebab *error* yang tidak stabil dikarenakan *human error*, peletakkan lempengan untuk pengukuran yang kurang stabil, dan kurang presisinya antara tachometer dan *handpiece micromotor*.

Dari data yang dihasilkan tersebut, bahwa putaran kiri pada modul dengan *setting* kecepatan putaran *micromotor* dari 5000-25000 rpm baik digunakan untuk pembentukan dan pemolesan *protesa* gigi karena rata-rata nilai *error* dari putaran kiri sangat minimum yaitu 0,4%.