

## BAB IV

### IMPLEMENTASI DAN ANALISIS

#### 4.1 Perangkat Keras

Perangkat keras yang terdapat pada sistem *Prototype electric supercharger turbine* ini terbagi menjadi dua bagian yang saling berkaitan, yaitu elektronik dan bagian konstruksi.

##### 4.1.1 Bagian Elektronik

Bagian elektronik terdiri dari empat bagian yaitu Pulser, Arduino Uno, *motor brushless DC*, dan ESC. Pada bagian ini akan dibahas mengenai rangkaian elektronik yang digunakan untuk mengontrol sistem kerja pada *Prototype electric supercharger turbine*.

Rangkaian elektronik pada *Prototype electric supercharger turbine* ini terdiri dari *input*, proses dan *output*, seperti sistem kontrol dengan Arduino Uno. Bagian *input* merupakan bagian utama suatu alat, bagian utama yang digunakan merupakan pembacaan putaran mesin kendaraan menggunakan pulser.



**Gambar 4.1** Pulser

Bagian proses merupakan bagian kontrol dan pengolahan data menggunakan Arduino Uno dengan mikrokontrollnya ATmega328 dimana *input* berasal dari pulser.

#### 4.1.1.1 Pulser

Pulser adalah besi bermagnet dililit kawat tembaga halus yang berfungsi sebagai *speed sensor* sementara *outputnya* berupa besar kecilnya tegangan induksi yang dihasilkan yang merupakan informasi mengenai putaran mesin. Tegangan *output* pulser sebesar 0,3 - 0,7 volt. Sementara mikrokontroller dapat mengenali tegangan sebesar 0 volt sebagai signal digital *low*, atau tegangan 5 volt sebagai signal digital *high*. Sedangkan untuk mengenali nilai tegangan antara 0 volt hingga 5 volt, diperlukan *Analog-to-Digital Converter* atau biasa disebut ADC. Untuk mempermudah pembacaan, maka digunakan op-amp LM358 untuk meningkatkan nilai *output* pulser.

Sedangkan prosedur untuk pembacaan tegangan dapat dilakukan dengan langkah-langkah sebagai berikut :

- Pulser mengirim tegangan pada op-amp LM358.
- Op-amp LM358 meningkatkan tegangan 100x.
- *Output* dibaca dengan ADCpulser.
- ADCpulser dikalibrasi dengan ESC (ADCesc).
- ADCesc dikalibrasi dengan *motor brushless*

Untuk pengambilan data dari Pulser dengan menggunakan program ADC (*Analog-to-Digital Converter*) adalah sebagai berikut :

```
int sensorPin = A0;    // select the input pin for the potentiometer

int ledPin = 13;      // select the pin for the LED

int sensorValue = 0;  // variable to store the value coming from the
sensor

void setup() {

    // declare the ledPin as an OUTPUT:

    pinMode(ledPin, OUTPUT);

    Serial.begin(9600);

}

void loop() {

    // read the value from the sensor:

    sensorValue = analogRead(sensorPin);

    Serial.println(sensorValue);

    delay(1);

    // turn the ledPin on

    digitalWrite(ledPin, HIGH);

    // stop the program for <sensorValue> milliseconds:

    delay(sensorValue);

    // turn the ledPin off:

    digitalWrite(ledPin, LOW);

    // stop the program for for <sensorValue> milliseconds:
```

Sedangkan untuk mengkalibrasikan data output pulser dengan ESC (*elektronik speed control*) adalah sebagai berikut :

```
#include <Servo.h>

Servo myESC;

int sensorPin = A0; // select the input pin for the potentiometer

int ledPin = 13;    // select the pin for the LED

int sensorValue = 0; // variable to store the value coming from
the sensor

int rpm,x;

int escpwm;

void setup() {

  // declare the ledPin as an OUTPUT:

  pinMode(ledPin, OUTPUT);

  Serial.begin (9600);

  myESC.attach(9);}

void loop() {

  // read the value from the sensor:

  sensorValue = analogRead(sensorPin);

  rpm = sensorValue* x + 2;

  escpwm = map(rpm, 0, 1023, 0, 360);

  myESC.write(escpwm); // sets the servo position according to
the scaled value

  delay(15);
```

```
Serial.println(sensorValue);

delay(15);

Serial.println(sensorValue);

//delay(500);

// turn the ledPin on

digitalWrite(ledPin, HIGH);

// stop the program for <sensorValue> milliseconds:

//delay(sensorValue);

// turn the ledPin off:

digitalWrite(ledPin, LOW);

// stop the program for for <sensorValue> milliseconds:

}
```

Pengkalibrasian data *output* pulser dan *electronic speed control* berfungsi untuk menyesuaikan putaran mesin motor otto dengan *motor brushless*, sedangkan dalam pembuatan *prototype electric supercharger turbine* kecepatan *motor brushless* diatur atau di set pada saat Rpm motor otto 1500 (pelan), 5000 (sedang), 10.000 (cepat). Ketentuan pengaturan kecepatan motor brushless berdasarkan pembukaan skep pada *intake manifold*. Kecepatan motor brushless dapat diatur sesuai yang diinginkan dengan mengubah program kalibrasi pada ESC (*elektronik speed control*).

#### 4.1.1.2 Arduino Uno

Arduino Uno adalah *board* Arduino berbasis ATmega328 yang mempunyai kelebihan dibanding Arduino yang berbasis ATmega168 yaitu mempunyai DC *power jack*. Dalam pembuatan alat ini menggunakan Arduino Uno dengan mikrokontroler 328.

Seperti yang sudah dipaparkan pada BAB II tentang dasar teori Arduino Uno. Arduino Uno memiliki 14 pin digital I/O 6 diantaranya untuk mendukung *output* PWM, 6 pin Analog *input*, untuk tegangan operasi Arduino adalah 5 volt tegangan *input* 7-12 volt, dan untuk tegangan *output* 6-20 volt.

Dari setiap digit pin pada Arduino dapat digunakan sebagai *input* dan *output* dengan menggunakan fungsi dari `pinMode`, `digitalWriter`, dan `digitalread` yang beroperasi pada tegangan 5 volt. Setiap pin dapat menerima lebih dari 40 mA dan memiliki sebuah *internal pull-up* resistor dari 20-50 kOhms. Sebagai tambahan, beberapa pin memiliki fungsi yang special:

- 1) Serial : 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan memancarkan (TX) serial data TTL (*Transistor-Transistor Logic*). Kedua pin ini dihubungkan ke pin-pin yang sesuai dari chip Serial Atmega8U2 USB-ke-TTL.
- 2) External Interrupts : 2 dan 3. Pin-pin ini dapat dikonfigurasi untuk dipicu sebuah *interrupt* (gangguan) pada sebuah nilai rendah, suatu kenaikan atau penurunan yang besar, atau suatu perubahan nilai. Lihat fungsi `attachInterrupt()` untuk lebih jelasnya.
- 3) PWM : 3, 5, 6, 9, 10, dan 11. Memberikan 8-bit PWM *output* dengan fungsi `analogWrite()`.

- 4) SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin-pin ini mensupport komunikasi SPI menggunakan *SPI library*.
- 5) LED : 13. Ada sebuah LED yang terpasang, terhubung ke pin digital 13. Ketika pin bernilai HIGH LED hidup, ketika pin bernilai LOW LED mati.

Arduino Uno mempunyai 6 *input* analog, diberi label A0 sampai A5, setiapnya memberikan 10 bit resolusi (contohnya 1024 nilai yang berbeda). Secara *default*, 6 input analog tersebut mengukur dari ground sampai tegangan 5 Volt, dengan itu mungkin untuk mengganti batas atas dari *range* dengan menggunakan pin AREF dan fungsi *analogReference()*. Di sisi lain, beberapa pin mempunyai fungsi khusus :

TWI : pin A4 atau SDA dan pin A5 atau SCL. Mensupport komunikasi TWI dengan menggunakan *Wire library*

Ada sepasang pin lainnya pada board :

- 1) AREF. Referensi tegangan untuk *input* analog. Digunakan dengan *analogReference()*.
- 2) Reset. Membawa garis LOW untuk mereset mikrokontroler. Secara khusus, digunakan untuk menambahkan sebuah tombol reset untuk melindungi dari gangguan di salah satu *board*.

#### **4.1.1.3 Motor Brushles**

Perangkat *elektromagnetis* yang mengubah energi listrik menjadi energi mekanik adalah motor listrik. Motor listrik memiliki dua jenis yaitu motor listrik DC dan motor listrik AC. Motor listrik AC biasanya digunakan untuk penggerak alat pada industri. *Motor brushless* DC ini dapat bekerja

ketika stator diberikan arus 3 fasa. Akibat arus yang melewati kumparan pada stator maka timbul medan magnet.

Dalam pembuatan *prototype electric supercharger turbine* ini *motor brushless* DC sebagai penggerak *propeller*. Propeller ini berfungsi untuk memampatkan udara yang masuk ke *intake manifold*.

#### **4.1.1.4 ESC (*Electronic Speed Control*)**

Seperti yang telah dijelaskan pada BAB II *Electric Speed Control* adalah Sebuah modul rangkaian elektronik yang fungsinya mengatur putaran pada motor sesuai ampere yang dibutuhkan oleh motor. Fungsi ESC pada *prototype electric supercharger turbine* ini adalah sebagai *driver motor brushless* DC.

#### **4.1.2 Bagian Kontruksi**

Bentuk kontruksi dari alat *prototype electric supercharger turbine* ini berbentuk tabung berbahan fiber yang telah di modifikasi untuk penempatan *motor brushless* dan *propeller*. Berikut bentuk kontruksi dari *prototype electric supercharger turbine*.



**Gambar 4.2** Bentuk Kontruksi Alat

Bagian-bagian dari konstruksi *prototype electric supercharger turbine* yaitu :

1) *Motor Brushless DC*

Fungsi dari *motor brushless* dalam *prototype electric supercharger turbine* ini adalah sebagai penggerak *propeller* untuk meningkatkan volume tekanan udara pada *intake manifold*.

2) *Propeller*

Fungsi dari *propeller* adalah untuk menghasilkan tekanan udara.

#### 4.2 Operasi Perangkat Lunak

Operasi perangkat lunak *prototype electric supercharger turbine* adalah ketika kendaraan dihidupkan, maka sistem elektronik dari *prototype* alat *electric supercharger turbine* memulai proses kerjanya. Pertama, pulser akan mendeteksi putaran mesin dengan cara membaca putaran tonjolan pada *pick up coil*. Kemudian proses berlanjut pada tahap berikutnya yaitu *output* pulser akan dinaikkan 100x dengan menggunakan op-amp LM358. Setelah itu data dari op-amp LM358 akan diolah dengan Arduino Uno menggunakan ADC pulser lalu dikalibrasikan dengan ADCesc untuk mengendalikan kecepatan *motor brushless* yang diatur atau diset sesuai Rpm motor otto 1500 (pelan), 5000 (sedang), dan 10.000 (cepat). *Motor brushless* dikendalikan sesuai dengan lamanya motor otto menyala. Sementara program akan aktif ketika kontak motor otto posisi ON.

#### 4.3 prinsip kerja

Prinsip kerja dari *prototype* alat *electric supercharger turbine* ini disesuaikan dengan data-data yang ada pada BAB II. *Prototype electric supercharger turbine* dioperasikan secara otomatis mengikuti kontak motor

otto. Jadi pada saat kontak ON maka program ON, sedangkan saat kontak OFF maka program OFF. Kecepatan putar *motor brushless* di set dalam tiga step yaitu pada Rpm kendaraan 1500, 5000, dan 10.000. Berdasarkan referensi yang ada, penggunaan *prototype electric supercharger turbine* dapat diaplikasikan pada kendaraan 150 cc.

Pada saat Rpm motor otto menunjukkan angka 1500 maka kecepatan *motor brushless* pelan, selanjutnya saat Rpm motor otto menunjukkan angka 5000 maka kecepatan *motor brushless* sedang, dan saat Rpm motor otto menunjukkan angka 10.000 maka kecepatan *motor brushless* cepat (Max).

#### **4.4 Pengoperasian Alat**

Pertama *jumper* kabel positif pulser, lalu hubungkan dengan op-amp LM358, lalu pasang adaptor 12 volt sebagai catu daya mikrokontroller, op-amp LM358, dan ESC (*elektronik speed control*). Kemudian untuk memulai *prototype electric supercharger turbine*, pada saat kontak motor otto posisi ON, maka proses kerja *prototype electric supercharger turbine* akan berjalan. Apabila terjadi *error* pada sistem kerja alat, maka bisa ditekan tombol reset pada *board* Arduino Uno. Untuk mengakhiri proses kerja *prototype electric supercharger turbine* adalah dengan memastikan kontak motor otto dalam kondisi OFF kembali.

#### **4.5 Pengujian Alat**

Pengujian alat terdapat lima tahapan yaitu pengujian rangkaian, pengujian pulser, pengujian Arduino uno, pengujian ESC (*elektronik speed control*), pengujian *motor brushless* dan validasi sistem.

## 4.5.1 Pengujian Rangkaian

Pengujian rangkaian dilakukan pada rangkaian catu daya, op-amp LM358, dan rangkaian mikrokontroller.

### 4.5.1.1 Pengujian Catu Daya

#### 1. Pengujian Rangkaian Catu Daya

*Protoype electric supercharger turbine* ini menggunakan catu daya accu 12 volt untuk menjalankan sistem mikrokontroller, op-amp LM358, dan ESC (*elektronik speed control*). Tegangan yang digunakan adalah tegangan langsung dari accu motor oto tanpa adanya perubahan.

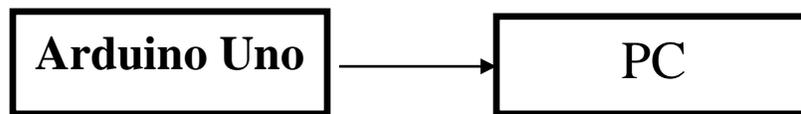


**Gambar 4.3** Pengujian tegangan catu daya accu

Pada gambar 4.3 terlihat tegangan catu daya accu sebesar 12.71 volt. Tegangan tersebut merupakan tegangan sumber yang sudah dapat digunakan langsung oleh sistem mikrokontroller, op-amp maupun ESC (*elektronik speed control*).

#### 4.5.1.2 Pengujian Arduino Uno

Berikut ini adalah diagram blok untuk pengujian Arduino Uno dengan komputer.

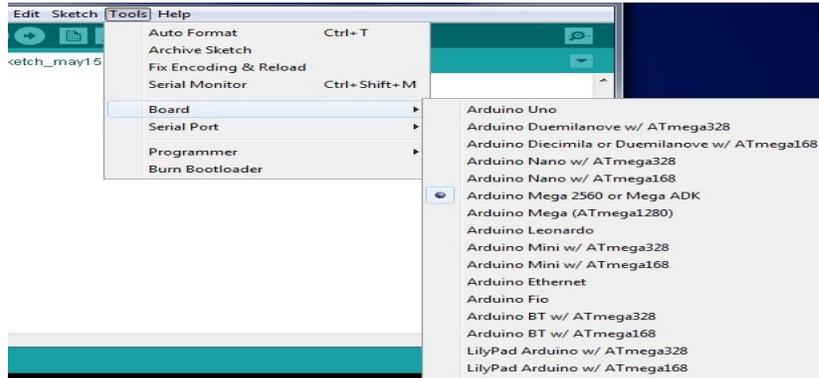


**Gambar 4.4** Diagram blok pengujian mikrokontroller

Walaupun arduino dapat beroperasi dengan mendapat *input* daya dari komputer, namun hal itu bukan berarti Arduino dapat berkomunikasi dengan komputer tersebut. Untuk memastikan bahwa Arduino uno telah terpasang dengan benar dan dapat berkomunikasi dengan interaktif maka perlu dilakukan pengujian.

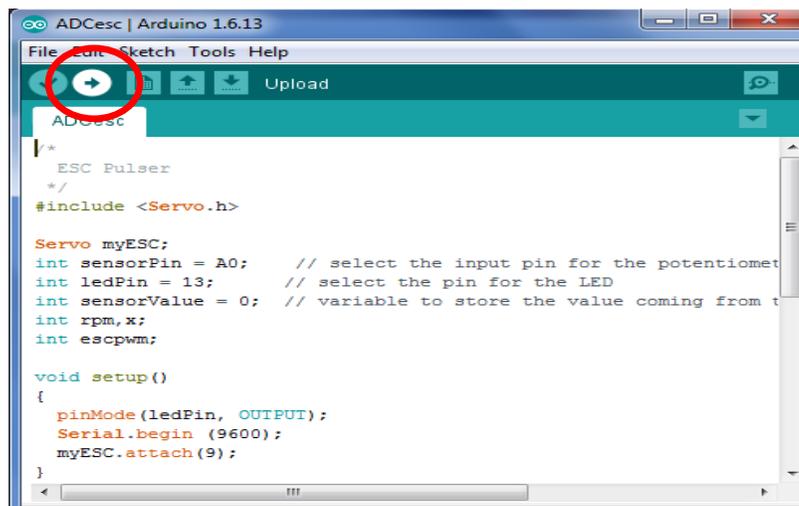
- 1) Jalankan IDE Arduino dengan menjalankan file *arduino.exe* pada software Arduino. Program ini adalah program *Java* walaupun terlihat seperti program windows pada umumnya. Jika terjadi sebuah pesan kesalahan kemungkinan besar komputer tersebut belum terinstal *Java Development Kit (JDK)*.

- 2) Jalankan menu Tools → Board, kemudian pilih jenis papan yang akan digunakan.



**Gambar 4.5** Pemilihan board arduino

- 3) Pada *Toolbar* klik tombol *Upload* untuk memuat *sketch* tersebut ke dalam papan Arduino.



**Gambar 4.6** Upload program

setelah melakukan *upload sketch* sukses maka akan ditandai dengan adanya pesan seperti berikut.

```
myServo.write(servoPwm); // sets the servo position
delay(15);
Serial.println(sensorValue);
```

Done uploading.

Sketch uses 3,344 bytes (10%) of program storage space. Maximum is 32  
Global variables use 237 bytes (11%) of dynamic memory, leaving 1,811

**Gambar 4.7** Pesan program sukses di upload

### 4.5.1.3 Pengujian Pulser

Pengujian yang dilakukan pada pulser dengan cara menjumper kabel positif pulser dan pengukuran menggunakan multimeter digital dikarenakan *output* pulser berupa tegangan. Metode pengukuran pulser dibagi dalam tiga step. saat Rpm motor otto 1500, 5000, dan 10.000. Berikut ini adalah gambar tegangan pulser pada saat Rpm motor otto 1500, 5000, dan 10.000.



**Gambar 4.8** Tegangan pulser saat Rpm 1500



**Gambar 4.9** Tegangan pulser saat Rpm 5000



**Gambar 4.10** Tegangan pulser saat Rpm 10.000

#### 4.5.1.4 Pengujian Op-amp LM358

Op-amp berfungsi sebagai penguat tegangan *output* pulser agar dapat dibaca oleh Arduino. Pengujian Op-amp dengan menggunakan multimeter digital dengan cara menyambungkan *output* pada pulser ke *input* Op-amp. Selanjutnya Op-amp disambungkan dengan catu daya

Berikut adalah gambar hasil pengujian Op-amp LM358 pada saat Rpm motor otto 1500, 5000, dan 10.000.



**Gambar 4.11** Tegangan Op-amp saat Rpm 1500



**Gambar 4.12** Tegangan Op-amp saat Rpm 5000



**Gambar 4.13** Tegangan Op-amp saat Rpm 10.000

#### **4.5.1.5 Pengujian Motor Brushless dan ESC (*Electronic Speed Control*)**

Pengujian motor brushless dan ESC bertujuan untuk dapat mencari nilai besaran pulsa untuk memutarakan *motor brushless* sehingga dapat

menggerakkan *propeller*. Pengujian ini dilakukan guna mendapatkan nilai yang tetap agar dalam pemrograman lebih mudah menentukan nilai tegangan yang digunakan. Hasilnya dapat dilihat pada tabel 4.1 berikut.

**Tabel 4.1** Hasil pengujian nilai untuk motor brushless

NO	Tegangan output Op-amp	Motor Brushless
1	0 - 1 Volt	Berputar pelan
2	2.27 Volt	Berputar sedang
3	2.65 Volt	Berputar cepat

Keterangan: nilai diatas adalah nilai tegangan *output* Op-amp yang diberikan. Dimana nilai diatas didapat dari pengujian dan analisa.

#### 4.5.2 Validasi Sistem

Pengecekan oprasional kerja alat secara berulang-ulang dan menyeluruh disebut dengan validasi sistem. Validasi dilakukan untuk membuktikan bahwa semua bagian dan komponen serta program apakah telah sesuai dengan yang diharapkan. Hasil validasi terhadap keseluruhan sistem disediakan dalam bentuk tabel berikut ini.

**Tabel 4.2** Hasil validasi keseluruhan

NO	Kerja Alat	Kondisi	Deskripsi Kerja	Status
1	Kontak Motor	OFF	Alat dalam kondisi OFF	OK
		ON	Alat dalam kondisi ON	OK
2	Arduino Uno	OFF	Indikator mati	OK

**Tabel 4.2** Hasil validasi keseluruhan (lanjutan)

		ON	Indikator hidup	OK
3	Pulser	Rpm 1500	Tegangan output 0.002 Volt	OK
		Rpm 5000	Tegangan output 0.011 Volt	OK
		Rpm 10.000	Tegangan output 0.015 Volt	OK
4	Op-amp LM358	Rpm 1500	Tegangan output 1.01 Volt	OK
		Rpm 5000	Tegangan output 2.27 Volt	OK
		Rpm 10.000	Tegangan output 2.65 Volt	OK
5	Motor Brushless dan ESC	OFF	Motor brushless dalam kondisi OFF	OK
		ON	Motor brushless berputar pelan, sedang dan cepat	OK
6	Tombol Reset	ON	Mereset Mikrokontoller dan mengulang kerja program dari awal	OK

Berdasarkan tabel diatas, seluruh bagian sistem telah berjalan dengan normal seperti yang diharapkan.

#### **4.6 Implementasi Alat**

Setelah dilakukan pengujian dan validasi sistem, maka tahap selanjutnya dilakukan implementasi alat terhadap *prototype electric supercharger turbine*.

Implementasi yang dilakukan adalah dengan mengaktifkan sistem pada kendaraan. Dalam hal ini pengukuran dan pengambilan data dilakukan

pada *dynotest* sebelum dan sesudah penggunaan alat dengan penentuan *variable* Rpm motor otto, *horse power*, dan torsi kendaraan.

Data ditunjukkan dalam bentuk tabel sebagai berikut :

1. pengambilan data *dynotest* tanpa *prototype electric supercharger turbine*

**Tabel 4.3** pengambilan data *dynotest* tanpa *electric supercharger turbine*

NO	RPM	<i>Horse Power</i> (Hp)	Torsi (N*M*M)
1	6250	8.7	9.21
2	6500	8.9	9.42
3	6750	9.1	9.57
4	7000	9.9	10.01
5	7117	10.1	10.09
6	7250	10.1	9.86
7	7500	9.8	9.25
8	7750	8.9	8.15
9	8000	8.7	7.72
10	8250	8.8	7.59
11	8500	9.3	7.74
12	8750	9.8	7.94
13	9000	10.4	8.18
14	9250	10.7	8.17
15	9500	10.8	8.08
16	9663	11.0	8.09
17	9750	10.9	7.93
18	10000	10.5	7.41
19	10250	9.9	6.86

Berdasarkan tabel 4.3 pengujian kendaraan pada *dynotest* tanpa menggunakan *prototype electric supercharger turbine* pada rpm 6250 menghasilkan *horse power* sebesar 8.7 Hp dan torsi 9.21 Nm. sedangkan pada rpm 9663 mencapai titik tertinggi yang menghasilkan *horse power* sebesar 11.0 Hp dan torsi 8.09 Nm. dan pada rpm 11500 menghasilkan *horse power* sebesar 8.5 Hp dan torsi 5.19 Nm.

2. Pengambilan data *dynotest* dengan *prototype electric supercharger turbine*

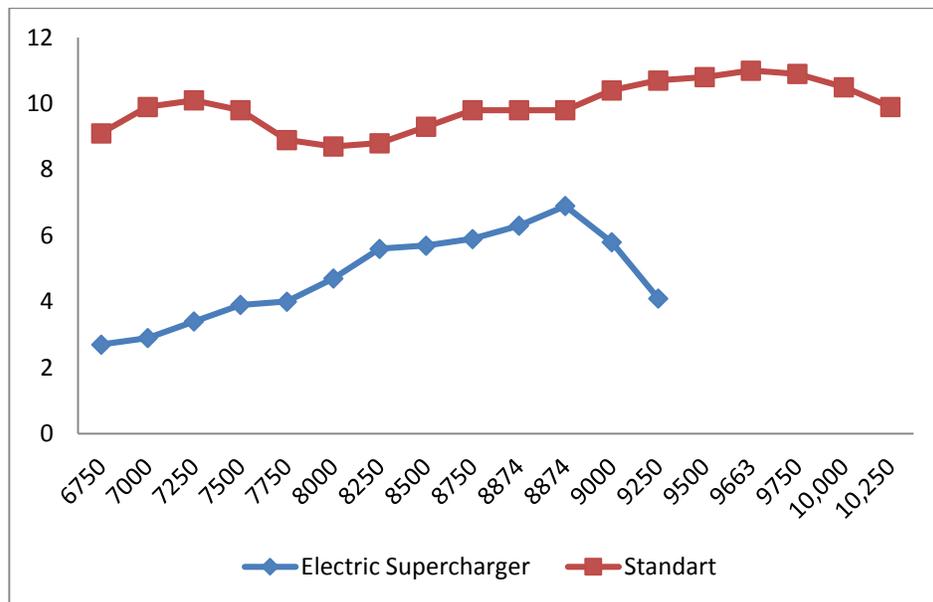
**Tabel 4.4** Pengambilan data *dynotest* dengan *electric supercharger turbine*

NO	RPM	Horse Power (Hp)	Torsi (N*M*M)
1	6750	2.7	2.81
2	7000	2.9	2.91
3	7250	3.4	3.33
4	7500	3.9	3.72
5	7750	4.0	3.68
6	8000	4.7	4.19
7	8250	5.6	4.83
8	8500	5.7	4.77
9	8750	5.9	4.79
10	8874	6.9	5.05
11	9000	5.8	4.56
12	9250	4.1	3.12

Berdasarkan tabel 4.4 pengujian kendaraan pada *dynotest* dengan menggunakan alat *prototype electric supercharger turbine* pada rpm 6750 hanya menghasilkan *horse power* sebesar 2.7 Hp dengan torsi 2.81 Nm, sedangkan pada rpm 8874 mencapai titik tertinggi dengan menghasilkan

*horse power* sebesar 6.9 Hp dan torsi 5.05 Nm, dan pada saat rpm 9250 menghasilkan *horse power* 4.1 Hp dan torsi 3.12 Nm.

**Gambar 4.14** Grafik perbandingan sebelum menggunakan alat dan saat penggunaan *prototype electric supercharger turbine* pada *dynotest*



Berdasarkan percobaan pertama dan kedua dapat disimpulkan bahwa menggunakan *prototype electric supercharger turbine* yang dilakukan pada *dynotest* dengan data pada tabel 4.4 maka hasil yang diperoleh belum mencapai target dikarenakan ruangan pada *dynotest* hampa udara dan sirkulasi udara kurang baik, sementara udara bebas sangat berpengaruh pada kinerja *prototype electric supercharger turbine*.

3. Pengambilan data perbandingan efisiensi bahan bakar sebelum dan saat menggunakan *prototype electric supercharger turbine*.

**Tabel 4.5** perbandingan efisiensi bahan bakar

NO	Odometer sebelum penggunaan alat	Odometer saat penggunaan alat
1	43277.8	43216.6
2	43298.8	43244.6
	21 Km	28 Km

Berdasarkan tabel 4.5 perbandingan efisiensi bahan bakar sebelum dan sesudah penggunaan *prototype electric supercharger turbine* dapat disimpulkan bahwa sebelum menggunakan *prototype electric supercharger turbine* kendaraan mampu menempuh jarak 21 Km dengan menggunakan 1 liter bensin. Dan saat menggunakan *prototype electric supercharger turbine* kendaraan mampu menempuh jarak 28 Km dengan menggunakan 1 liter bensin.

Perhitungan persentase efisiensi bahan bakar dapat menggunakan rumus sebagai berikut :

$$\frac{\text{Nilai Kenaikan}}{\text{Nilai sebelum kenaikan}} \times 100\%$$

$$\frac{7 \text{ Km}}{21 \text{ Km}} \times 100\% = 30 \%$$

Berdasarkan hasil perhitungan tersebut, dapat disimpulkan bahwa dengan penggunaan *prototype electric supercharger turbine* dapat menghemat bahan bakar sebesar 30%.