

4.1 Pengujian Supply Daya Pada Sistem Elektrikal Sepeda Listrik

a

b

c

Berdasarkan hasil pengukuran tegangan dari multimeter seperti pada gambar 4.1, Terdapat selisih pengukuran sekitar 0,05V hingga 0.19V. Selain itu dilakukan juga pengukuran arus modul kontrol dengan Clamp meter seperti gambar 4.2



78

Tabel 4. 1 Pengukuran Output *Buck Converter*

Sumber Tegangan	Tegangan Referensi (V)	Tegangan Pengukuran (V)	Selisih (V)	Error (%)
Buck 12V	12	12,19	0,19	1,58%.
Buck 4,6V	4,6	4,65	0,05	1,09%
Buck 3,3V	3,3	3,3	0	0%

Berdasarkan pengukuran pada tabel 4.1, diketahui terdapat selisih error sebesar 1,09% hingga 1,58% antara tegangan referensi dari *buck converter* dengan tegangan pengukuran oleh multimeter. Angka error tersebut tergolong sangat kecil karena masih dibawah ambang toleransi tegangan yang dapat diterima oleh perangkat terkait. Selisih 0,05V untuk tegangan masukan *hall* sensor masih masuk kedalam batas amannya, sedangkan selisih 0,19V untuk IR2110 masih sangat aman untuk mengontrol keluaran MOSFET.

Tabel 4. 2 Perhitungan dan pengukuran daya modul *gate driver*

Pengujian	Perhitungan Total Daya	Pengukuran Total Daya	Selisih Daya	Error
Modul <i>Gate Driver</i>	5,09 Watt	4,51 Watt	0,58	11,3%

Berdasarkan tabel 4.2 tentang pengukuran dan perhitungan daya modul *gate driver*, hasil pengujian menunjukkan adanya selisih total daya dari hasil perhitungan awal dengan pengukuran. Total daya berdasarkan perhitungan lebih sebesar 5,09 sedangkan hasil pengukuran aktualnya memiliki nilai lebih rendah di 4,51 watt. Selisih daya antara keduanya sebesar 0,58 watt dengan presentase error sebesar 11,3%. Adanya selisih error ini merupakan hal yang wajar dalam implementasi rangkaian sistem elektronika, dan bisa di akibatkan oleh berbagai faktor seperti

konsumsi daya operasional yang tidak selalu maksimal seperti pada arus arus perhitungan dan adanya rugi daya akibat *switching* dan resistansi internal pada rangkaian.

4.2 Pengujian *Hall* Sensor

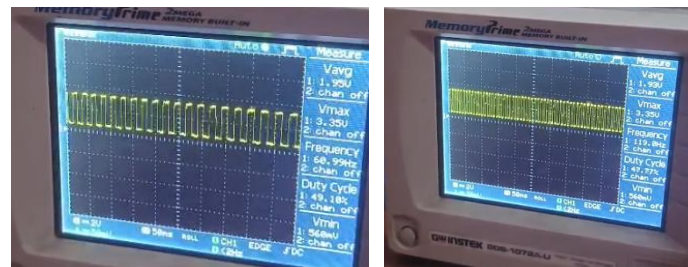
Pada bagian *hall* sensor, dilakukan dua buah pengujian yakni pengujian RC *Low pass filter* dan pengujian keluran sinyal *hall* sensor

4.2.1 Pengujian RC Low Pass Filter pada *hall* Sensor

Pengujian ini dilakukan untuk membandingkan keluaran sinyal *hall* sensor sebelum dan sesudah diberi RC *Low Pass Filter* untuk melihat bentuk keluaran sinyal pada osiloskop. Pengujian ini dilakukan dengan membandingkan bentuk sinyal keluaran pada osiloskop serta tegangan maksimal dari sinyal keluarannya dengan kecepatan motor di 300 RPM.



Gambar 4. 3 Keluaran *hall* sensor sebelum diberi LPF



Gambar 4. 4 a.) sinyal *hall* setelah fiberi LPF dengan kecepatan 10 Km/h b.) simyal *hall* setelah fiberi LPF dengan kecepatan 20 Km/h

Tabel 4. 3 perbandingan parameter *hall* sensor tanpa filter dan dengan filter

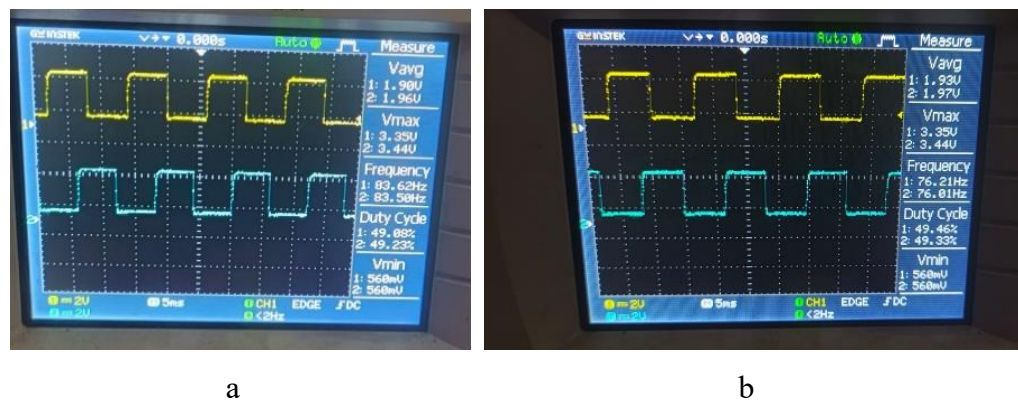
Kondisi <i>Hall</i>	Vin	Vmax	Fmax
Tanpa Filter	4,6 V	5,59 V	4.471 Hz
Dengan Filter	4,6 V	3.35 V	119 Hz

Pada Tabel 4.3 menunjukkan bahwa RC *Low pass filter* dapat meredam spike tegangan dan frekuensi yang cukup tinggi apabila kita bandingkan dengan tanpa filter. Pada saat motor BLDC di aliri listrik dan berputar, maka akan menghasilkan *Back EMF* yang menyebabkan *spike* atau lonjakan tegangan pada keluaran *hall* sensor. Hal tersebut tidak baik karena dapat mempengaruhi pembacaan kondisi *hall* sensor untuk kebutuhan komutasi. Selain itu, pin input STM32 memiliki toleransi tegangan input analog di angka 3,6 V. Jika pin input tersebut terus-menerus terkena tegangan yang lebih tinggi dari 3,6V maka dapat merusak mikrokontroler tersebut. *Switxhing* frekuensi tinggi pada inverter juga dapat menyebabkan EMI yang menyebabkan *noise* frekuensi dari sinyal keluaran *hall* sensor sehingga mikrokontroler tidak dapat membaca kondisi nyala atau matinya *hall* sensor dengan baik. Untuk mengatasi *noise* pada tegangan dan frekuensi keluaran *hall* sensor, kami memasang RC *Low pass filter* dengan resistor 10K Ohm dan kapasitor ceramic 100 nF. Nilai frekuensi *cut-off* yang dapat diloloskan dihitung berdasarkan persamaan 2.8 dan di dapati nilai sebesar 160 Hz.

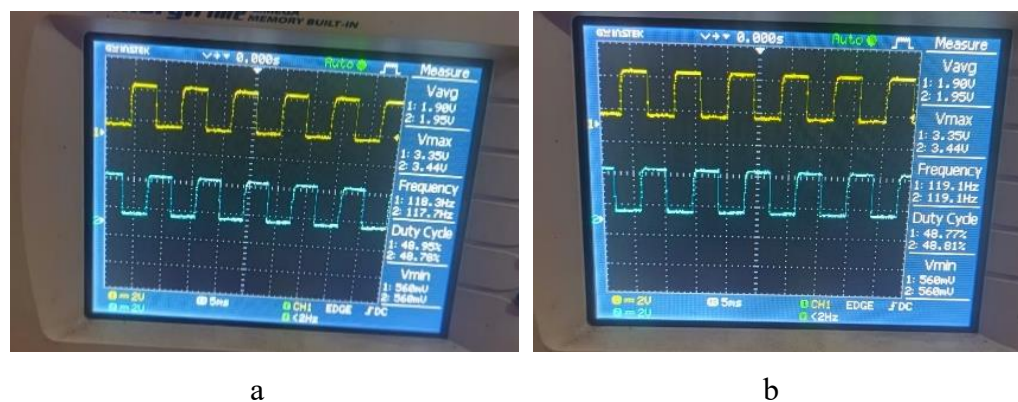
RC *Low pass filter* yang di pasang akan melewatkan sinyal dibawah 160Hz dan meredam habis sinyal di atasnya. Sehingga frekuensi sinyal yang masuk pada *hall* sensor menjadi lebih stabil dan bersih. Selain itu, *spike* tegangan yang mucul akibat *back EMF* akan diserap oleh kapasitor ceramic dan membuang lonjakannya ke ground. Sehingga tegangan yang keluar setelah terkena filter menjadi stabil di angka 3,35V yang mana masih batas aman untuk masuk ke pin analog input STM32F401CCU6.

4.2.2 Pengujian Keluaran *Hall* sensor

Pengujian ini dilakukan untuk melihat keluaran *hall* sensor dari osiloskop dengan melihat tegangan maksimal, tegangan rata-rata, dan frekuensi untuk tiap *hall* sensor yang telah di filter. Pengujian ini dilakukan dengan membandingkan dua buah *hall* sensor pada RPM 200 dan 300 untuk mengetahui nilai masing-masing parameter pengujian pada RPM yang berbeda. Berikut untuk data-data perbandingannya.



Gambar 4. 5 a.)Pengujian Osiloskop *Hall* Sensor 1 dan 2 200 RPM b.) Pengujian Osiloskop *Hall* Sensor 1 dan 3 200 RPM



Gambar 4. 6 a.)Pengujian Osiloskop *Hall* Sensor 1 dan 2 300 RPM b.) Pengujian Osiloskop *Hall* Sensor 1 dan 3 300 RPM

Tabel 4. 4 Data pengujian *Hall* sensor

Sensor	200 RPM			300 RPM		
	Vmax (V)	Vavg (V)	Frekuensi (Hz)	Vmax (V)	Vavg (V)	Frekuensi (Hz)
<i>Hall 1</i>	3,35	1,90	83,62	3,35	1,90	118,3
<i>Hall 2</i>	3,44	1,96	83,50	3,44	1,95	117,7
<i>Hall 3</i>	3,44	1,97	76,01	3,44	1,95	119,1

Berdasarkan tabel 4.4 Dapat dilihat bahwa tegangan maksimal dan tegangan rata-rata *hall* sensor 1 memiliki sedikit selisih dengan *hall* sensor 2 dan 3 yang nilainya cukup mirip. *Hall* tersebut dikreanakan adanya perbedaan nilai resisitor yang digunakan karena pastinya ada selisih nilai resistansi antara label dan aktualnya. Namun, selisih nilai tersebut tidak berpengaruh ke pembacaan sinyal *hall* sensor ke mikrokontroler karena pembacaan digital STM32f401 hanya melihat nilai tegangan di atas atau dibawah nilai *threshold*, nilai 3,35V sudah memenuhi ambang batas *threshold* sehingga logikanya di anggap *High*. Pada pengukuran nilai Vavg, nilai pada masing-masing *hall* terukur sama meskipun RPM nya berbeda dikarenakan adanya RC *Low pass filter* yang mengubah pulsa keluaran *Hall* menjadi tegangan yang lebih halus . Untuk nilai frekuensi masing-masing *hall* sensor terukur mirip-mirip satu sama lain pada pengujian di RPM yang sama, selain itu pengujin ini juga membuktikan kalau semakin cepat putaran motor, maka frekuensi yang keluar dari *hall* sensor juga semakin tinggi.

4.3 `Pengujian Komutasi

Pada pengujian komutasi, dilakukan dua buah pengujian yakni pengujian keluaran dari IC IR2110 untuk melihat karakteristik sinyal keluaran IR2110 pada pin VS, HO, dan LO.

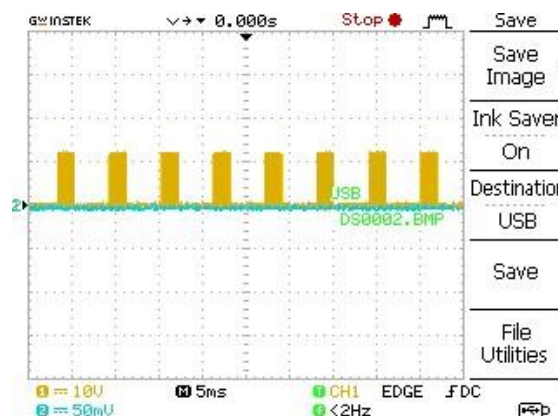
4.3.1 Pengujian IC IR2110

Pada pengujian IC IR2110 dilakukan dengan melihat output keluaran dari IC IR2110 pada pin yang terhubung dengan inverter. Pin yang di ukur pada pengujian ini yakni pin HO, VS, dan LO untuk memastikan sinyal yang dihasilkan oleh IC bekerja dengan baik sehingga dapat mengontrol MOSFET pada saat melakukan *switching* tegangan. Pada pengujian ini, IR2110 diberi tegangan sumber 12V untuk beroperasi dengan tegangan inverter 12V tanpa memberi beban ke inverter. Untuk hasil pengujiannya dapat dilihat pada gambar dibawah ini.

Tabel 4. 5 Pengukuran Output IC IR2110

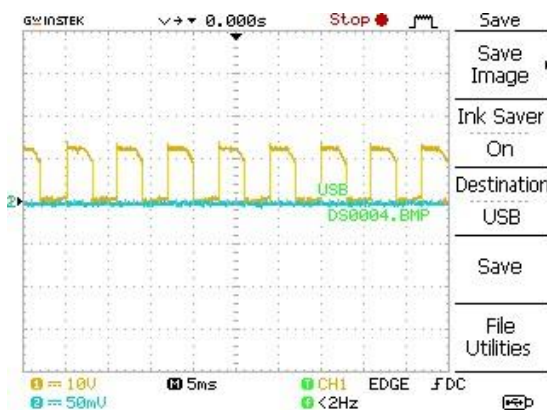
Output	Frekuensi (Hz)	Vref (V)	Vout(V)	Error Vout (%)
LO	10.000	12	11,6	3,3%
VS	210	12	13,1	9,1%
HO	210	24	23,2	3,3%

Tabel 4.6 menampilkan nilai frekuensi serta tegangan yang dihasilkan oleh HO, VS, dan LO IC IR2110. Untuk analisa keluaran pada osiloskop dapa dilihat pada gambar 4.8, 4.9, dan 4.10.



Gambar 4. 7 keluaran LO IR2110 pada osiloskop

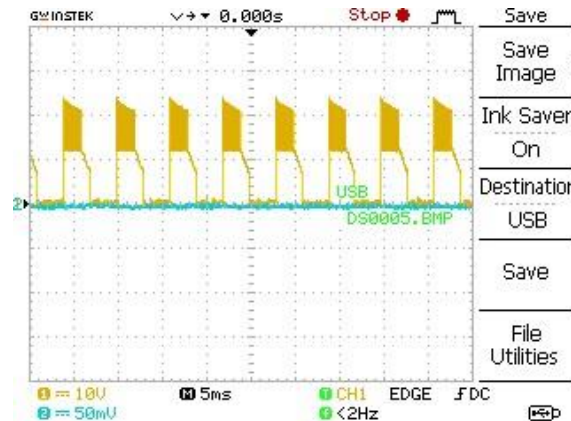
Pada gambar 4.8 dapat dilihat kalau keluaran LO dari IC IR2110 berbentuk menghasilkan gelombang kotak (*square wave*) yang sesuai dengan karakteristik sinyal PWM. Sinyal keluaran LO yang di hasilkan menunjukkan transisi logika *High* dan *Low* yang baik sehingga MOSFET bagian *Low side* dapat di kontrol dengan baik. Tegangan pada saat high yang terukur pada osiloskop bernilai 11,6V yang mmana selisih 3,3% dari tegangan refrensinya yakni 12V. Angka tersebut masih tergolong cukup ideal untuk mengaktifkan *gate* MOSFET sehingga MOSFET tidak cepat panas dan dapat membuka dengan sempurna. Untuk frekuensi yang dihasilkan juga sama dengan frekuensi yang diatur pada STM32F401CCU6 yakni di angka 10KHz.



Gambar 4. 8 keluaran sinyal VS IR2110 pada osiloskop

Pada gambar 4.9 dapat dilihat kalau bentuk gelombang pada VS atau *node switching* MOSFET tidak berbentuk kotak sempurna namun agak menyerupai trapesium, hal tersebut dikarenakan adanya kapasitansi parasitik MOSFET, resistor *gate*, serta waktu pengisian dan pengosongan muatan *gate* MOSFET yang kurang cepat. hal tersebut membuat bentuk sinyal keluarannya berubah secara bertahap sehingga menghasilkan sisi naik (*rising edge*) dan sisi turun (*falling edge*) yang tidak tegak lurus. Kendati demikian, sinyal tersebut masih menunjukkan proses *switching* pada inverter 3 fasa yang normal. Teganagn pada saat *high* terbaca di angka 13,1V dengan selisih eror sebesar 9,1% dari tegangan

refrensinya 12V. Selain itu, dapat dilihat kalau sinyal keluaran pada VS sangat minim noise sehingga tanpa adanya *overshoot* yang tinggi dan tegangan pada saat kondisi *low* yang mendekati 0V. Frekuensi yang terbaca di angka 210 Hz yang mana frekuensi tersebut merupakan frekuensi switching dari MOSFET.



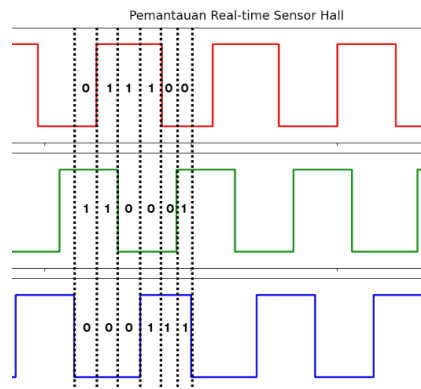
Gambar 4. 9 Keluaran sinyal HO IR2110 pada Osiloskop

Pada gambar 4.10 sinyal keluaran HO pada osiloskop menunjukkan karakteristik PWM dengan menggunakan tegangan refrensi VS sebagai acuannya. Dapat dilihat kalau PWM pada pin HO aktif saat VS dalam kondisi HIGH. Hal tersebut membuat nilai Tegangannya menjadi lebih tinggi dibanding tegangan LO. Keluaran sinyal tersebut juga membuktikan kalau *Bootstrap* pada IR2110 bekerja dengan baik. Hal tersebut sangatlah penting karena pada inverter 3 fasa, Source pada top MOSFET tidak terhubung dengan *Ground*, melainkan dengan terhubung ke beban atau fasa motor, hal tersebut membuat V_{gs} agar MOSFET top dapat bekerja harus lebih tinggi dari tegangan fasanya. Tegangan yang terukur ada di 23,1V yang mana menghasilkan selisih eror sebesar 3,3% dari refrensinya 24V. Angka tegangan 12V di dapat dari penjumlahan antara tegangan VS dengan keluaran dari pin HO. Hal tersebut di buktikan dari keluaran sinyal HO yang mana teganga PWM berada di atas tegangan VS saat aktif, bukan di GND. Hasil osiloskop mencatat frekuensi dari HO sama dengan frekuensi dari VS, sehingga yang tercatat bukanlah frekuensi PWM melainkan frekuensi dari

switching MOSFET. jika ingin mengukur frekuensi PWM, *probe* GND harus di letakkan ke referensi yakni pin VS, namun hal tersebut tidak bisa di lakukan karena apa bila VS terhubung dengan Ground dari Probe maka dapat menimbulkan *short* dan merusak *inverter*.

4.3.2 Pengujian komutasi Six-Step

Pengujian ini dilakukan untuk membaca sinyal keluaran dari *hall* sensor yang telah di filter. Sinyal keluaran dari tiga buah hal sensor ini akan digunakan untuk menentukan proses komutasi yang mana akan membuat motor BLDC berputar. Pengambilan data pada pengujian ini dilakukan menggunakan GUI yang dibuat khusus untuk menampilkan data keluaran serial monitor yang berisi status dari masing-masing *hall* sensor. Untuk keluaran sinyalnya dapat dilihat pada gambar 4.7 Dibawah ini



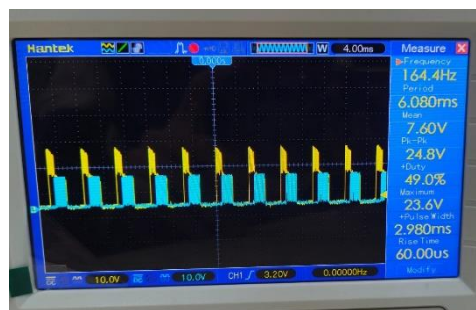
Gambar 4. 10 pembagian state hall sensor pada HMI

Berdasarkan gambar 4.7 Dapat dilihat kalau ketiga sinyal dari *Hall* Sensor (A, B, dan C) menghasilkan gelombang kotak (*square wave*) yang saling bergeser fase atau berbeda waktu aktifnya. Melalui penandaan garis vertikal pembatas *state*, terlihat jelas bahwa satu siklus elektrik penuh terbagi menjadi enam segmen operasional diskrit, di mana masing-masing segmen merepresentasikan rentang sudut sebesar 60 derajat pada komutasi *six-steps*, Untuk urutan aktivasi sensor dapat dilihat pada tabel 4.5 Berikut.

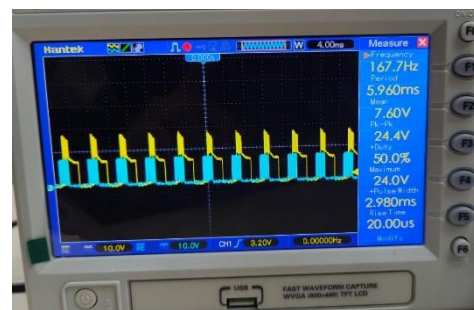
Tabel 4. 6 Hasil *truth* Tabel komutasi Six-step

Komutasi	Hall 1	Hall 2	Hall 3	MOSFET aktif
Step 1	0	1	0	Q1 dan Q4
Step 2	1	1	0	Q1 dan Q6
Step 3	1	0	0	Q3 dan Q6
Step 4	1	0	1	Q3 dan Q2
Step 5	0	0	1	Q5 dan Q2
Step 6	0	1	1	Q5 dan Q4

Tabel 4.5 memaparkan urutan proses komutasi *six-step commutation* dari ketiga sensor *Hall Effect* yang terpasang pada motor BLDC. Berdasarkan data tabel tersebut, satu siklus elektrik penuh yakni 360 derajat elektrik, terbagi secara simetris menjadi enam tahapan diskrit, di mana masing-masing tahapan merepresentasikan rentang pergerakan sudut sebesar 60 derajat elektrik. Pada setiap tahapannya, ketiga sensor *hall* secara sinkron menghasilkan urutan kode biner 3-bit yang spesifik dan berurutan, yaitu: 010 (Step 1), 110 (Step 2), 100 (Step 3), 101 (Step 4), 001 (Step 5), dan ditutup dengan 011 (Step 6). Transisi pola biner ini terjadi secara konsisten tanpa ada yang terlewati. Tidak ditemukannya kombinasi logika nol secara keseluruhan (000) maupun satu secara keseluruhan (111) di seluruh rentang komutasi menegaskan bahwa ketiga sensor *Hall* pada stator diposisikan tepat sebesar 120 derajat secara fisik.



a



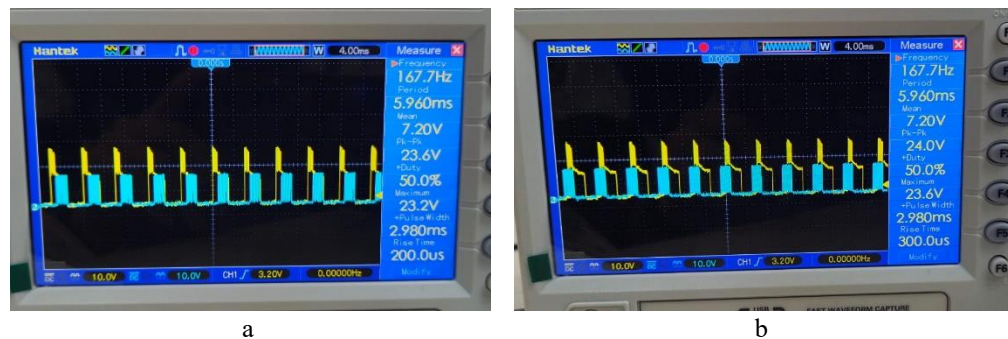
b

Gambar 4. 11 a.) Keluaran osiloskop Step 1 b.) Keluaran osiloskop Step 2

Dapat dilihat pada Gambar 4.11 a yang menampilkan profil sinyal elektrikal pada saat Step 1 aktif, di mana mikrokontroler STM32F401 mendeteksi

kombinasi logika sensor Hall 010 (hanya Hall B yang berlogika *High*) sehingga memicu aktivasi MOSFET Q1 pada *High-side* dan Q4 pada *Low-side* secara bersamaan. Selanjutnya, Gambar 4.11b memperlihatkan transisi halus ke Step 2 akibat perubahan posisi rotor yang menghasilkan logika Hall 110 (Hall A dan Hall B berlogika *High*) Sesuai dengan tabel kebenaran komutasi sistem, mikrokontroler merespons secara instan dengan menonaktifkan Q4 dan mengaktifkan Q6 pada *Low-side*, sementara status Q1 pada *High-side* tetap dipertahankan ON guna menjaga kontinuitas aliran arus pada lilitan stator.

Visualisasi osiloskop pada kedua fase ini membuktikan stabilitas sinyal PWM pada komutasi *Six-step*. Parameter *duty cycle* yang tertahan konsisten di angka 49% hingga 50% memvalidasi keberhasilan algoritma pembatasan pada perangkat lunak untuk mencegah lonjakan arus *inrush* berlebih pada *inverter*. Selain itu, frekuensi komutasi yang stabil di rentang 164,4 Hz hingga 167,7 Hz serta transisi gelombang yang bersih dari distorsi membuktikan sinkronisasi yang presisi antara putaran mekanis rotor dan perpindahan fasa elektrik. Hal ini membuktikan IC *gate driver* IR2110 mampu mengeksekusi pergantian *state* secara mulus dan aman tanpa adanya risiko *shoot-through*.



Gambar 4. 12 a.) Keluaran osiloskop Step 3 b.) Keluaran osiloskop Step 4

Dapat dilihat pada gambar 4.12a yang menampilkan sinyal elektrikal saat Step 3 aktif, yang dipicu ketika mikrokontroler STM32F401 mendeteksi transisi kombinasi logika sensor Hall menjadi 100 (hanya Hall A yang berlogika *High*). Sesuai dengan tabel kebenaran komutasi, kondisi ini menginstruksikan sistem

untuk mematikan MOSFET Q1 dan mengaktifkan MOSFET Q3 pada *High-side*, sementara MOSFET Q6 pada *Low-side* tetap dipertahankan dalam kondisi ON. Selanjutnya, Gambar 4.12b menunjukkan kelanjutan siklus ke Step 4 akibat pergerakan putaran rotor yang mengubah logika Hall menjadi 101 (Hall A dan Hall C berlogika *High*). Pada fase ini, mikrokontroler merespons dengan memutus arus pada Q6 dan mengaktifkan MOSFET Q2 pada *Low-side*, sedangkan status Q3 pada *High-side* tetap dipertahankan ON untuk menjaga kesinambungan aliran arus menuju fasa stator motor BLDC.

Visualisasi osiloskop pada kedua fase ini kembali mengonfirmasi stabilitas sinyal PWM yang menumpang pada profil komutasi trapesiodal. Parameter *duty cycle* yang tertahan konsisten di angka 50,0% memvalidasi keberhasilan algoritma pembatasan pada perangkat lunak untuk mencegah lonjakan arus *inrush* berlebih pada *inverter*. Selain itu, frekuensi komutasi yang stabil di 167,7 Hz serta transisi gelombang yang bersih dari distorsi membuktikan sinkronisasi yang presisi antara putaran mekanis rotor dan perpindahan fasa elektrik. Hal ini membuktikan IC *gate driver* IR2110 mampu mengeksekusi pergantian *state* tanpa terjadi *shoot-through*.



Gambar 4. 13 a.) Keluaran osiloskop Step 5 b.) Keluaran osiloskop Step 4

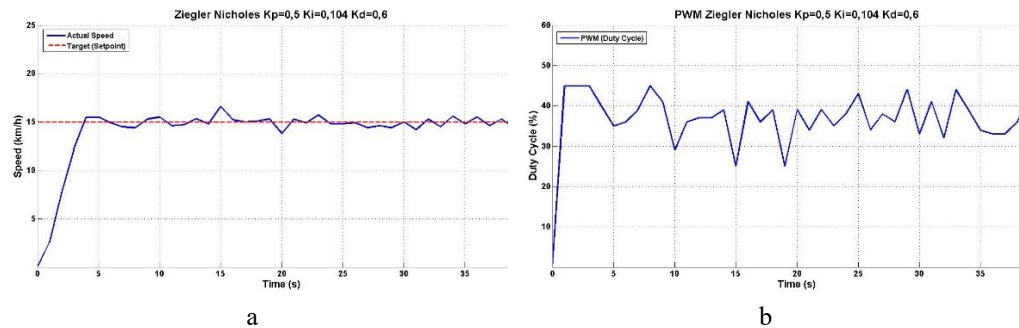
Gambar 4.13a menampilkan sinyal elektrik pada saat Step 5 aktif, yang terjadi ketika mikrokontroler STM32F401 mendeteksi kombinasi logika sensor Hall 001 (hanya Hall C yang berlogika High). Merujuk pada tabel kebenaran

komutasi, transisi ini menginstruksikan sistem untuk mematikan MOSFET Q3 dan mengaktifkan MOSFET Q5 pada High-side, sedangkan MOSFET Q2 pada Low-side tetap dipertahankan ON. Selanjutnya, Gambar 4.13b memperlihatkan fase penutup siklus yakni Step 6, yang terpicu akibat pergerakan rotor yang mengubah logika Hall menjadi 011 (Hall B dan Hall C berlogika High). Pada titik ini, mikrokontroler memutus arus pada Q2 dan mengaktifkan MOSFET Q4 pada Low-side, sementara status Q5 pada High-side tetap dipertahankan ON sebelum siklus nantinya kembali berulang ke Step 1.

Visualisasi osiloskop pada kedua fase akhir ini secara konsisten mengonfirmasi stabilitas sinyal PWM. Parameter *duty cycle* yang persis tertahan di angka 50,0% memvalidasi keandalan algoritma pembatasan pada perangkat lunak dalam melindungi komponen inverter dari potensi lonjakan arus *inrush* berlebih. Selain itu, frekuensi komutasi yang tetap stabil di 167,7 Hz dengan transisi gelombang yang bebas dari distorsi membuktikan bahwa sinkronisasi antara putaran mekanis rotor dan perpindahan fasa elektrik berjalan secara presisi hingga akhir siklus. Kestabilan bentuk gelombang ini juga menegaskan kembali bahwa IC gate driver IR2110 beroperasi secara optimal dalam mengeksekusi pergantian *state* yang mulus dan aman tanpa memicu risiko tegangan tembus (*shoot-through*).

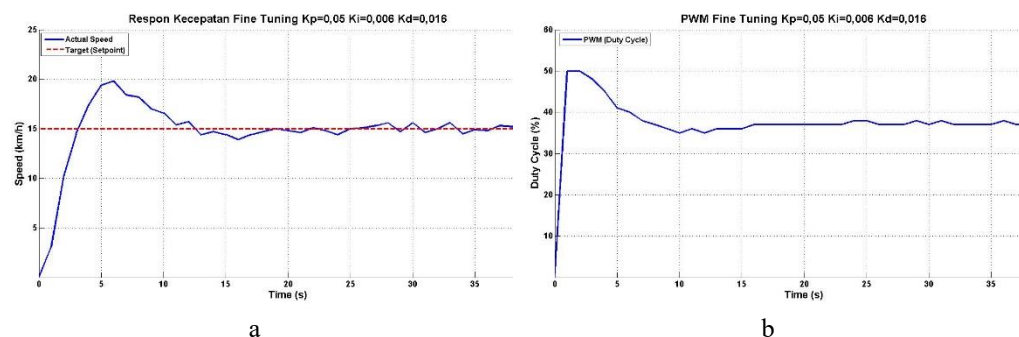
4.4 Pengujian Respon Transien kendali PID

Melalui Metode Ziegler Nichols Orde 1, diperoleh nilai K_p sebesar 0.1, K_i sebesar 0.104, dan K_d sebesar 0.6 untuk nilai konstantanya. Kemudian parameter tersebut di terapkan ke sistem dengan menggunakan *set point* pengujian kecepatan di 15 Km/h. pada pengujian ini data kecepatan dan PWM aktual sistem di ambil 1 detik lalu di ubah menjadi kurva untuk mempermudah analisa. Setelah diujikan, didapati respon transien sistem dan PWM seperti pada gambar 4.14 dan 4.15 Berikut.



Gambar 4. 14 a.)Respon kecepatan PID Ziegler Nichols b.)Respon PWM PID Ziegler Nichols

Berdasarkan respons kecepatan dan PWM pada gambar 4.14, penggunaan parameter tersebut menunjukkan kendala yang krusial pada tingkat kestabilan aktuator. Meskipun grafik kecepatan terlihat mampu merespons dan mengikuti *set point*, sinyal *duty cycle* PWM berosilasi secara ekstrem dan sangat fluktuatif di rentang 25% hingga 45%. Motor BLDC pada penelitian ini memiliki batasan daya dorong minimal yakni apabila nilai *duty cycle* turun dibawah ambang batas 32%, motor akan kehilangan torsi secara drastis dan berhenti berputar. Sebaliknya, ketika kontroler tiba-tiba melonjakkan *duty cycle* kembali ke atas 32% untuk mengejar ketertinggalan *error*, motor akan menerima sentakan daya secara paksa. Osilasi tajam dengan naik turun drastis yang terus berulang ini memicu ketidakstabilan lonjakan arus yang sangat berbahaya, karena berpotensi merusak komponen *switching* terutama MOSFET IRFP250 pada *inverter 3-fasa half-bridge* akibat tekanan elektrik dan termal yang berlebihan. Untuk itu perlu dilakukan penyesuaian ulang (*fine-tuning*) dan menghasilkan respon seperti gambar 4.15 berikut.



Gambar 4. 15 a.)Respon kecepatan PID *Fine Tuning* b.)Respon PWM PID *Fine Tuning*

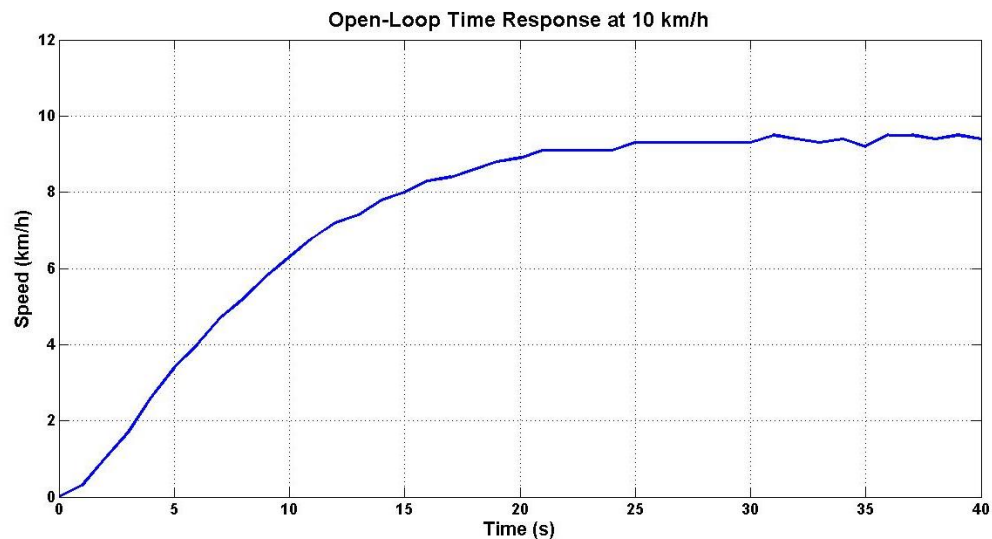
Merujuk pada referensi jurnal tinjauan pustaka, metode *fine-tuning* ini diaplikasikan dengan mengecilkan semua nilai parameter awal menjadi sekitar 10% dari hasil perhitungan *Ziegler-Nichols* guna menekan respons PWM yang terlalu agresif. Khusus untuk parameter K_i , nilainya dimodifikasi menjadi mendekati 10% dari nilai K_p yang baru agar sistem tidak merespons akumulasi *error* secara berlebihan, sehingga diperoleh nilai parameter akhir $K_p = 0,05$, $K_i = 0,006$, dan $K_d = 0,016$. Hasil dari modifikasi parameter ini direpresentasikan pada gambar 4.15a untuk respon kecepatan dan 4.15b untuk respon PWM. Melalui penalaan ini, fluktuasi sinyal kendali berhasil dieliminasi sehingga keluaran PWM menjadi jauh lebih mulus dan secara konsisten bertahan di atas batas kritis 32%. Stabilitas *duty cycle* ini memastikan torsi motor BLDC terdistribusi secara kontinu tanpa adanya pemutusan arus yang mendadak, sehingga kinerja kelistrikan *inverter* menjadi jauh lebih stabil dan aman. Sebagai *trade-off* dari penghalusan sinyal kendali PWM tersebut, respons kecepatan motor menunjukkan munculnya efek *overshoot* pada fase transien awal, munculnya *overshoot* transien ini merupakan langkah kompromi yang jauh lebih ideal dan rasional dibandingkan memaksakan parameter *Ziegler-Nichols* murni yang sangat berisiko menurunkan umur pakai dan ketahanan perangkat keras akibat arus yang tidak stabil.

4.5 Pengujian *Open-loop*

Pengujian sistem openloop dilakukan dengan memberikan nilai PWM tertentu yang dapat menghasilkan putaran roda mendekati atau menyamai dengan setpoint yang diberikan. Pada pengujian ini, kecepatan motor BLDC di *logger* oleh sensor kecepatan dari *hall* sensor lalu dikirimkan dengan komunikasi UART ke laptop untuk di tampilkan di GUI dan disimpan dengan format .csv untuk di buat grafik pada excell. Untuk pengujiannya sendiri terdapat empat setpoint yang di uji yakni 10 Km/Jam, 15 Km/Jam, 20 Km/Jam, dan 25 Km/Jam.

4.5.1 Pengujian *Open-loop* kecepatan 10 Km/h

Pada pengujian ini, Sistem diberi *Duty cycle* sebesar 34% dimana nilai tersebut akan menghasilkan kecepatan mendekati atau menyamai 10 Km/Jam. Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.16 berikut.



Gambar 4. 16 respon kecepatan openloop 10 Km/h

Berdasarkan gambar 4.16 dapat dilihat kalau pada implementasi *Duty cycle* 34% pada sistem *Open-loop* dapat menghasilkan kecepatan yang mendekati 10 Km/h namun belum bisa mencapai 10 Km/h. Untuk melihat respon sistem secara akurat, dilakukan perhitungan parameter respon waktu berdasarkan kurva kecepatan pada gambar 4.16 meliputi *Rise Time*, *Overshoot*, *Settling time*, *Steady-state error*.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *open-loop* dengan set point 10 km/h selama 15 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19, tidak ada *overshoot* yang tercatat dikarenakan tidak ada nilai yang melebihi *steady-state* nya saat kondisi transien.

c. *Settling Time*

Sistem stabil pada *set point* 10 Km/h. menggunakan persamaan 2.20 didapatkan waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-20.

d. *Steady-state Error*

Berdasarkan perhitungan dengan persamaan 2.21, didapatkan *steady-state error* sebesar 0,6 km/h atau sekitar 6% lebih kecil dari *set point* yang diharapkan.

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.7 untuk mempermudah penganalisaan.

Tabel 4. 7 Parameter respon waktu sistem *Open-loop* 10 Km/h

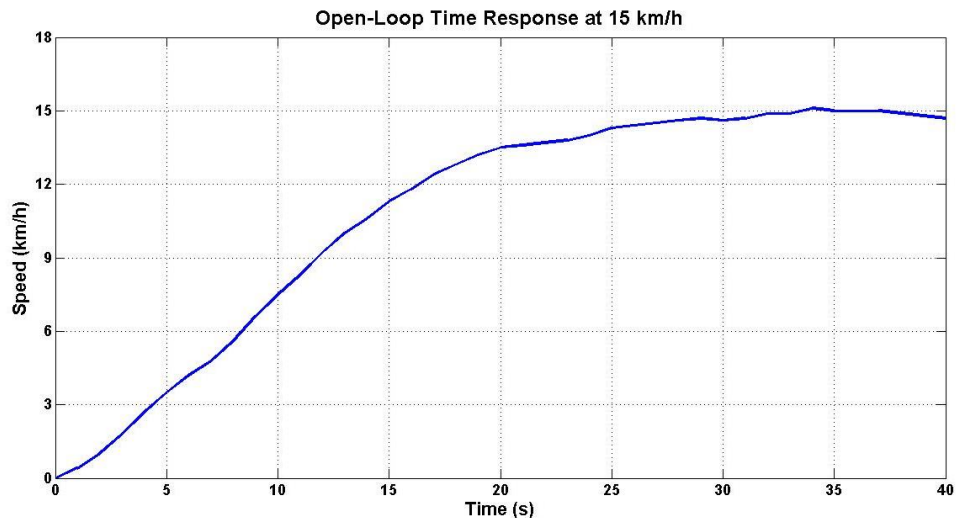
SetPoint (Km/h)	<i>Duty</i> <i>Cycle</i> (%)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady- state Error (Km/h)
10	34	15	20	0	0,6

Berdasarkan tabel 4.7 dapat dilihat jika pada pengujian sistem *open-loop* dengan *duty cycle* 34% secara konstan menghasilkan kurva kecepatan motor dengan tipikal keluaran sistem *open-loop*. Kecepatan motor meningkat secara perlahan tanpa mengalami lonjakan lonjakan kecepatan, hal tersebut dibuktikan dengan nilai *overshoot* 0% yang menandakan kecepatan sistem tidak pernah melebihi nilai *set point* nya. Namun respon sistem tergolong cukup lambat yang

dibuktikan dengan *rise time* selama 15 detik untuk mencapai 90% dari kecepatan stabilnya, serta *settling time* yang membutuhkan waktu 20 detik agar fluktuasi kecepatan dapat masuk ke dalam batas 5% toleransi kecepatan tunaknya. Selain itu, terdapat *steady-state error* sebesar 0,6 Km/h karena kecepatan sistem tertahan pada kondisi tunak di kecepatan 9,4 Km/h sehingga sistem tidak dapat menjangkau *setpoint* 10 Km/h yang diberikan.

4.5.2 Pengujian *Open-loop* kecepatan 15 Km/h

Pada pengujian ini, Sistem diberi *Duty cycle* sebesar 38% dimana nilai tersebut akan menghasilkan kecepatan mendekati atau menyamai 15 Km/Jam. Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.13 berikut.



Gambar 4. 17 Respon kecepatan sistem *Open-loop* 15 Km/h

Berdasarkan gambar 4.17 dapat dilihat kalau pada implementasi *Duty cycle* 38% pada sistem *Open-loop* dapat menghasilkan kecepatan yang mendekati 15 Km/h namun belum bisa mencapai 15 Km/h. Untuk melihat respon sistem secara akurat, dilakukan perhitungan parameter respon waktu berdasarkan kurva

kecepatan pada gambar 4.17 meliputi *Rise Time*, *Overshoot*, *Settling time*, *Steady-state error*.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *open-loop* dengan set point 15 km/h selama 17,5 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19, tidak ada *overshoot* yang tercatat dikarenakan tidak ada nilai yang melebihi *steady-state* nya saat kondisi transien.

c. *Settling Time*

Sistem stabil pada pada *set point* 15 Km/h. menggunakan persamaan 2.20 didapati waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-25.

d. *Steady-state Error*

Berdasarkan perhitungan dengan persamaan 2.21, didapati *steady-state error* sebesar 0,3 km/h atau sekitar 2% lebih kecil dari set point yang di harapkan.

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.8 untuk mempermudah penganalisaan.

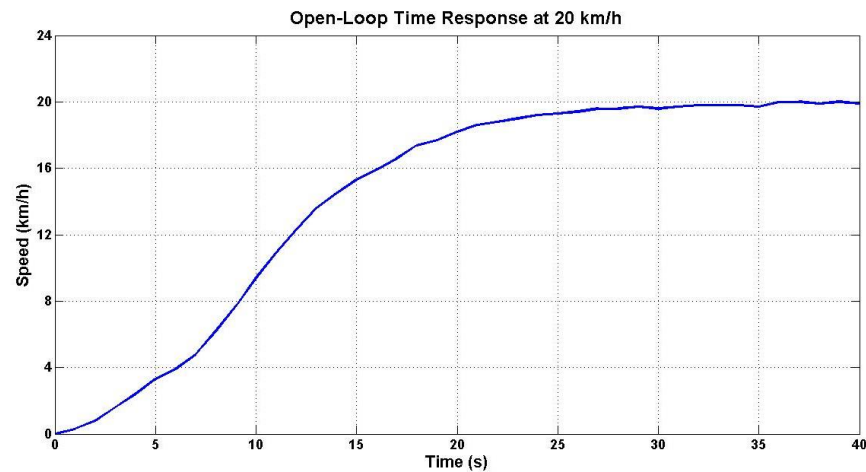
Tabel 4. 8 Parameter respon waktu sistem *Open-loop* 15 Km/h

SetPoint (Km/h)	<i>Duty</i> <i>Cycle</i> (%)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady- state Error (Km/h)
15	38	17,5	25	0	0,3

Berdasarkan tabel 4.8 dapat dilihat jika pada pengujian sistem *open-loop* dengan *duty cycle* 38% secara konstan menghasilkan kurva kecepatan motor dengan tipikal keluaran sistem *open-loop*. Kecepatan motor meningkat secara perlahan tanpa mengalami *overshoot* yang di buktikan dengan dengan *overshoot* sebesar 0%. Namun respon sistem tergolong cukup lambat yang dibuktikan dengan *rise time* selama 17,5 detik untuk mencapai 90% dari kecepatan stabilnya, serta *settling time* yang membutuhkan waktu 25 detik sama seperti pada *set point* 15 Km/h agar fluktuasi kecepatan dapat masuk ke dalam batas toleransi kecepatan tunaknya. Nilai *steady-state error* tergolong kecil di angka 0,3 Km/h dengan kecepatan rata-rata pada saat tunak di angka 14,7 Km/jam yang membuktikan akurasi nya cukup baik. Nilai *steady-state error* yang tergolong kecil apabila dibandingkan dengan *set point* 10 km/h kemungkinan dikarenakan nilai *duty cycle* yang diberikan relatif besar sehingga menghasilkan daya dorong listrik dan torsi yang cukup besar untuk melawan gaya gesek mekanis pada motor. Di sisi lain, parameter *steady-state error* pada pengujian ini tergolong sangat kecil, yakni hanya di angka 0,3 km/jam, di mana kecepatan rata-rata motor pada saat mencapai kondisi tunak stabil pada angka 14,7 km/jam. Nilai *error* statis yang menyusut cukup jauh apabila dibandingkan dengan pengujian *set point* 10 km/jam ini kemungkinan besar dikarenakan proporsi *duty cycle* 38% sudah mampu menyuplai tegangan rata-rata yang ideal bagi karakteristik lilitan motor.

4.5.3 Pengujian *Open-loop* Kecepatan 20Km/h

Pada pengujian ini, Sistem diberi *Duty cycle* sebesar 42% dimana nilai tersebut akan menghasilkan kecepatan mendekati atau menyamai 20 Km/Jam. PWM konstan yang diberikan akan menghasilkan putaran pada motor BLDC dan kecepatan akan terus naik hingga mencapai kondisi *steady-state* nya. Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.18 berikut.



Gambar 4. 18 Respon kecepatan sistem *Open-loop* 20 Km/h

Berdasarkan gambar 4.18 dapat dilihat kalau pada implementasi *Duty cycle* 42% pada sistem *Open-loop* dapat menghasilkan kecepatan yang mendekati 20 Km/h namun belum bisa mencapai 20 Km/h. untuk melihat respon sistem secara akurat, dilakukan perhitungan parameter respon waktu berdasarkan kurva kecepatan pada gambar 4.18 meliputi *Rise Time*, *Overshoot*, *Settling time*, *Steady-state error*.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapatkan *rise time* sistem *open-loop* dengan set point 20 km/h selama 16,5 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19, tidak ada *overshoot* yang tercatat dikarenakan tidak ada nilai yang melebihi *steady-state* nya saat kondisi transien.

c. *Settling Time*

Sistem stabil pada *set point* 25 Km/h. menggunakan persamaan 2.20 didapatkan waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-22.

d. *Steady-state Error*

Berdasarkan perhitungan dengan persamaan 2.21, didapati *steady-state error* sebesar 0,2 km/h atau sekitar 1% lebih kecil dari *set point* yang di harapkan.

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.9 untuk mempermudah penganalisaan.

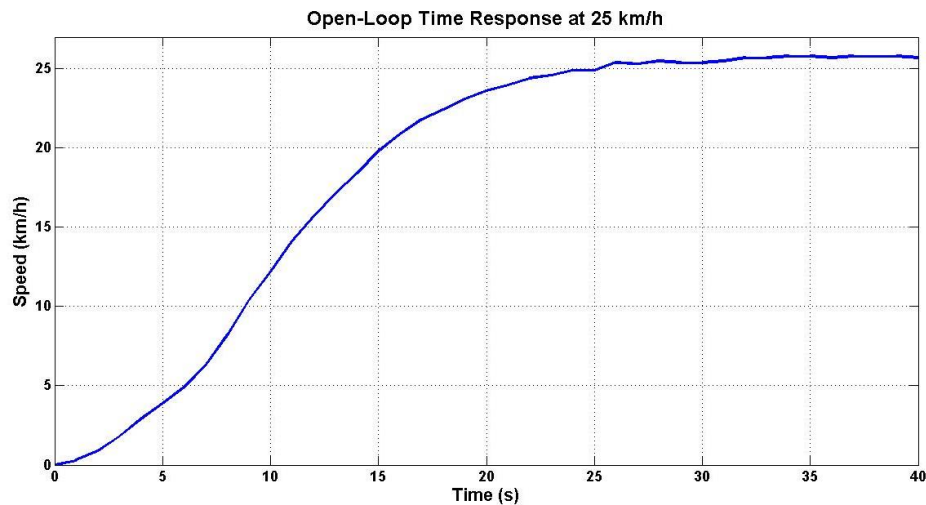
Tabel 4. 9 Parameter respon waktu sistem *Open-loop* 20 Km/h

SetPoint (Km/h)	<i>Duty</i> <i>Cycle</i> (%)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady- state Error (Km/h)
20	42	16,5	22	0	0,2

Berdasarkan tabel 4.9 dapat dilihat jika pada pengujian sistem *open-loop* dengan *duty cycle* 42% secara konstan menghasilkan kurva kecepatan motor dengan tipikal keluaran sistem *open-loop*. Kecepatan motor meningkat secara perlahan tanpa mengalami lonjakan yang dibuktikan dengan nilai *overshoot* di 0%. Respon sistem tergolong cukup lambat yang dibuktikan dengan *rise time* selama 16,5 detik untuk mencapai 90% dari kecepatan stabilnya, serta *settling time* yang membutuhkan waktu 22 detik agar fluktuasi kecepatan dapat masuk ke dalam batas toleransi kecepatan tunaknya sebesar 5% dari 19,8 Km/h. Nilai *steady-state error* tergolong sangat kecil di angka 0,2 Km/h dengan kecepatan rata-rata pada saat tunak di angka 19,8 Km/jam dimana angka tersebut sudah mendekati setpoint yang diinginkan. Namun tetap saja sistem belum mampu untuk menyentuh nilai *set poin* yang ditetapkan.

4.5.4 Pengujian *Open-loop* Kecepatan 25 Km/h

Pada pengujian ini, Sistem diberi *Duty cycle* sebesar 46% dimana nilai tersebut akan menghasilkan kecepatan mendekati atau menyamai 25 Km/Jam. Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.19 berikut.



Gambar 4. 19 Respon kecepatan sistem *Open-loop* 25 Km/h

Berdasarkan gambar 4.19 dapat dilihat kalau pada implementasi *Duty cycle* 46% pada sistem *Open-loop* dapat menghasilkan kecepatan yang melebihi 25 Km/h. untuk melihat respon sistem secara akurat, dilakukan perhitungan parameter respon waktu berdasarkan kurva kecepatan pada gambar 4.14 meliputi *Rise Time*, *Overshoot*, *Settling time*, *Steady-state error*.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *open-loop* dengan set point 25 km/h selama 15 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19, tidak ada *overshoot* yang tercatat dikarenakan tidak ada nilai yang melebihi *steady-state* nya saat kondisi transien.

c. *Settling Time*

Sistem stabil pada pada *set point* 25 Km/h. menggunakan persamaan 2.20 didapati waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-23.

d. *Steady-state Error*

Berdasarkan perhitungan dengan persamaan 2.21, didapati *steady-state error* sebesar 0,8 km/h atau sekitar 3,2% lebih besar dari *set point* yang di harapkan.

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.10 untuk mempermudah penganalisaan.

Tabel 4. 10 Parameter respon waktu sistem *Open-loop* 25 Km/h

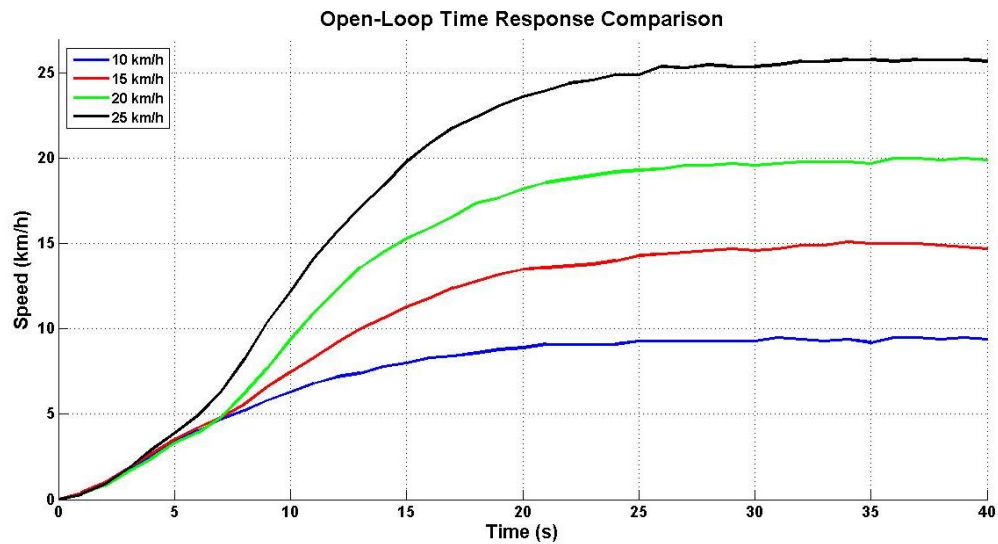
SetPoint (Km/h)	<i>Duty Cycle</i> (%)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady- state Error (Km/h)
25	46	15	23	0	0,8

Berdasarkan tabel 4.10 dapat dilihat jika pada pengujian sistem openloop dengan *duty cycle* 46% secara konstan menghasilkan kurva kecepatan motor dengan tipikal keluaran sistem *Open-loop* dengan *set point* di angka 25,8 Km/h. Kecepatan motor meningkat secara perlahan tanpa mengalami lonjakan yang dibuktikan dengan nilai *overshoot* di 0%. Respon sistem tergolong cukup lambat yang dibuktikan dengan *rise time* selama 15 detik untuk mencapai 90% dari kecepatan stabilnya. *Settling time* agar sistem stabil membutuhkan waktu 23 detik agar fluktuasi kecepatan dapat masuk ke dalam batas toleransi kecepatan tunaknya. Nilai *steady-state error* tergolong kecil di angka 0,8 Km/h dengan kecepatan rata-rata pada saat tunak di angka 25,8 Km/jam, nilai tersebut lebih

tinggi di banding *set poin* yang di harapkan yakni 25 Km/h. Jika dibandingkan dengan pengujian yang menggunakan *set point* dibawah 25 Km/h, pengujian ini memiliki *rise time* dan *settling time* yang lebih cepat di banding pengujian sebelumnya, namun *overshoot* yang dihasilkan merupakan yang tertinggi.

4.5.5 Perbandingan Respon Kecepatan *Open-loop*

Pada pengujian ini, grafik respon dari sistem *Open-loop* untuk kecepatan 10 km/h, 15 km/h, 20 km/h, dan 25 km/h akan di bandingkan untuk melihat karakteristik alami dari dinamika pergerakan motor BLDC sebelum diberikan algoritma kontrol PID. Melalui perbandingan keempat kurva tersebut, dapat dievaluasi tren parameter transien sistem serta membandingkan parameter respon waktu pada tiap *set point* kecepatannya. Untuk kurva perbandingan respon kecepatannya dapat dilihat pada gambar 4.15 berikut.



Gambar 4. 20 Perbandingan kurva respon kecepatan *Open-loop* pada seluruh varian set point

Untuk parameter respon waktu pada tiap *set point* kecepatan pada pengujian ini, dapat dilihat pada tabel 4.11 dibawah ini.

Tabel 4. 11 komparasi parameter respon waktu sistem *Open-loop* 10 Km/h

SetPoint (Km/h)	<i>Duty Cycle</i> (%)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady- state Error (Km/h)
10	34	15	20	0	0,6
15	38	17,5	25	0	0,3
20	42	16,5	22	0	0,2
25	46	15	23	0	0,8

Berdasarkan gambar 4.16 dan tabel 4.11, hasil pengujian *Open-loop* pada motor BLDC dilakukan menggunakan metode tanpa beban, di mana posisi roda belakang terangkat menggunakan *paddock*. Hasil pengujian menunjukkan hubungan yang linier dan konsisten antara *duty cycle* dan respons kecepatan akhir motor BLDC. Setiap peningkatan target kecepatan sebesar 5 km/jam diberikan kenaikan *Duty Cycle* secara proporsional sebesar 4%, membuktikan bahwa sistem kelistrikan merespons masukan secara baik dan logis. Dari segi dinamika transien, semua *set point* respon sistem memberikan bentuk kurva *overdamped* dengan *Maximum Overshoot* sebesar 0% pada seluruh rentang kecepatan. *Rise Time* sistem berada di kisaran 15–17,5 detik dan *Settling Time* di kisaran 20–25 detik, angka tersebut tidak berbeda drastis meskipun kecepatan cukup berbeda jauh. Hal ini membuktikan kalau perbedaan *duty cycle* tidak begitu berpengaruh pada *settling time*, namun berpengaruh pada kecepatan maksimal yang dapat dihasilkan oleh motor. Semakin besar *duty cycle*, maka semakin tinggi juga kecepatannya.

Pada fase transien awal di detik ke-0 hingga ke-5 memperlihatkan garis yang berhimpitan pada seluruh kurva respons kecepatan, di mana laju akselerasi terlihat cukup identik pada semua *set point* kecepatan. Dari segi hardware, IR IR2110 dengan IRFP250N tidak memiliki fitur yang dapat membatasi arus

keluaran. Alasan mengapa hal tersebut terjadi adalah karena penggunaan filter EMA pada keluaran RPM dan adanya momen inersia yang besar saat akselerasi. Dikarenakan sistem pembacaan *hall* sensor menggunakan perhitungan jumlah *tick* dalam rentang waktu tertentu, adanya keterlambatan pembacaan satu *tick* saja dapat menghasilkan error sebesar 25 RPM, hal tersebut yang membuat sistem sulit mencapai *steady-state*. Oleh sebab itu di aplikasikan filter EMA dengan a sebesar 0,2 sehingga sistem hanya akan mengambil 20% dari RPM yang baru, dan mempertahankan 80% nilai RPM sebelumnya.

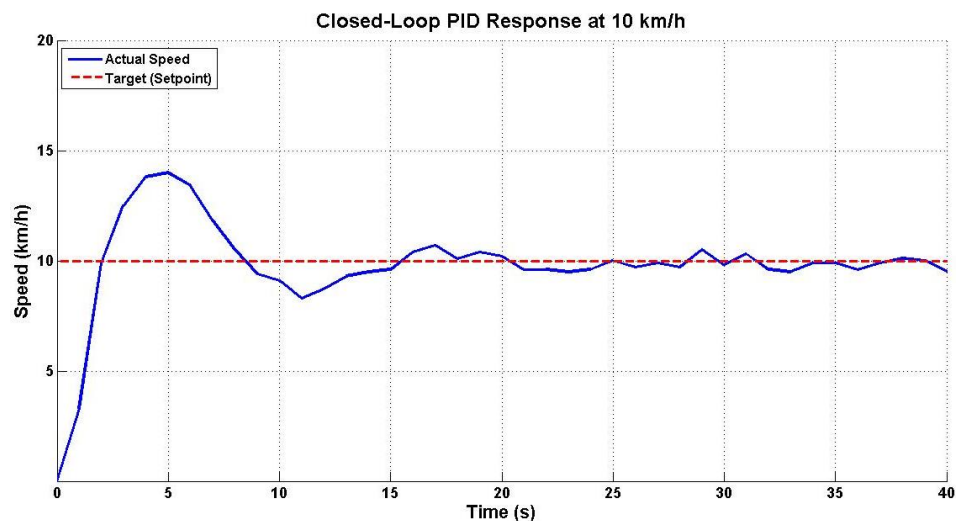
Filter tersebut dapat membuat respon kecepatan menjadi lebih halus dan mengatasi masalah error RPM, karena yang dipakai *exponential moving average*, jadi tidak memberikan delay yang tinggi seperti *moving average* biasa. Namun, pada saat akselerasi motor BLDC memiliki momen inersia yang cukup besar, hal tersebut membuat sebagian besar torsi yang di hasilkan motor terserap untuk mengalahkan momen inersia dari BLDC sehingga pada fase awal, kenaikan kecepatan tidak secepat ketika motor sudah memiliki momentum. Oleh sebab itu, pada fase awal, kenaikan kecepatan tidak berbeda cukup jauh meskipun *duty cycle* yang di pakai berbeda, ditambah adanya filter EMA yang hanya menambahkan 20% dari kecepatan tersebut sehingga grafik dari *set point* yang berbeda memiliki respon yang sama pada 5 detik awal. Akan tetapi setelah BLDC mendapatkan momentumnya dan inersia dapat di atasi, perbedaan *duty cycle* mulai terlihat responnya, *set point* dengan *duty cycle* yang lebih tinggi, kecepatannya cenderung naik lebih cepat. Hal tersebutlah yang membuat pada 5 detik awal grafik respon kecepatan tiap *set point* terlihat berhimpitan, namun setelah 5 detik dimana motor sudah dapat momentum, maka respon asli tiap setpoint mulai terlihat dikarenakan inersia yang harus di lawan oleh torsi tidak sebesar fase 5 detik awal.

4.6 Pengujian *Close-loop*

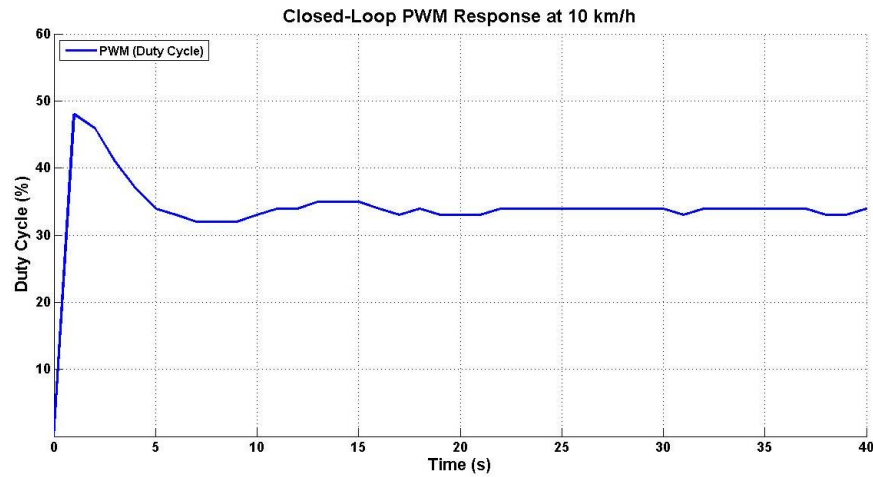
Pengujian sistem *close-loop* dilakukan dengan menggunakan sistem kontrol PID dengan parameter yang di dapat dari metode *fine tuning* $K_p = 0,05$ $K_i = 0,006$ dan $K_d = 0,016$ dengan menggunakan *feedback* kecepatan dari pembacaan sensor kecepatan dengan *hall* sensor agar sistem dapat berjalan stabil pada *set point* kecepatan yang di harapkan. Pada pengujian ini *duty cycle* PWM dibatasi di angka 50% Sama seperti pengujian *Open-loop*, pengujian ini terdapat empat setpoint yang di uji yakni 10 Km/Jam, 15 Km/Jam, 20 Km/Jam, dan 25 Km/Jam. Pada pengujian ini, kecepatan motor BLDC di *logger* oleh sensor kecepatan dari *hall* sensor lalu dikirimkan dengan komunikasi UART ke laptop untuk di tampilkan di GUI dan disimpan dengan format .csv untuk di buat grafik pada excell.

4.6.1 Pengujian *Close-loop* kecepatan 10 Km/h

Pada pengujian ini, sistem diberikan kontrol PID dengan *set point* kecepatan di 10 Km/h. Lalu sistem di nyalakan untuk mengetahui grafik respon sistemnya Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.21 berikut.



Gambar 4. 21 Respon kecepatan sistem *Close-loop* 10 km/h



Gambar 4. 22 Kurva respon PWM *Close-loop* 10 km/h

Berdasarkan gambar 4.21 dan 4.22 dapat dilihat kalau implementasi PID pada setpoint 10 Km/h menghasilkan grafik respon kecepatan yang mampu untuk mencapai *set point* yang diberikan. Selain itu, respon dari *duty cycle* yang terlihat pada gambar 4.17 cukup stabil sehingga rugi-rugi daya saat proses *switching* menjadi minimal. Hal ini membuat beban kerja serta pelepasan energi panas pada enam buah MOSFET IRFP250 menurun secara drastis, sehingga suhu operasional rangkaian tetap terjaga dan terhindar dari potensi kerusakan permanen akibat *overheat*. Dari segi dinamika laju kendaraan, profil PWM yang secara konsisten dipertahankan di atas ambang batas kritis 32% memastikan aktuator tidak pernah kehilangan daya dorong magnetiknya secara mendadak. Hasilnya, pergerakan rotasi motor BLDC menjadi sangat halus dan mampu mempertahankan traksi kecepatan secara konstan dengan efisiensi energi yang optimal. Untuk menilai respon kecepatan sistem dilakukan pengukuran parameter respon waktu dengan perhitungan sebagai berikut.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *close-loop* dengan set point 10 km/h selama 2 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19 *overshoot* yang tercatat mencapai 40%

c. *Settling Time*

Sistem stabil pada *set point* 10 Km/h. menggunakan persamaan 2.20 didapati waktu untuk sistem mulai stabil dengan toleransi 5% yakni dimulai dari detik ke-18.

d. *Steady-state Error*

Sistem *close-loop* mampu mengeliminasi *steady-state error* berdasarkan perhitungan dengan persamaan 2.21 dikarenakan sistem mampu menjaga kecepatan untuk tetap berada pada *set point*

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.12 untuk mempermudah penganalisaan.

Tabel 4. 12 Parameter respon waktu sistem *Close-loop* 10 Km/h

Setpoint (Km/h)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)
10	2	18	40	0

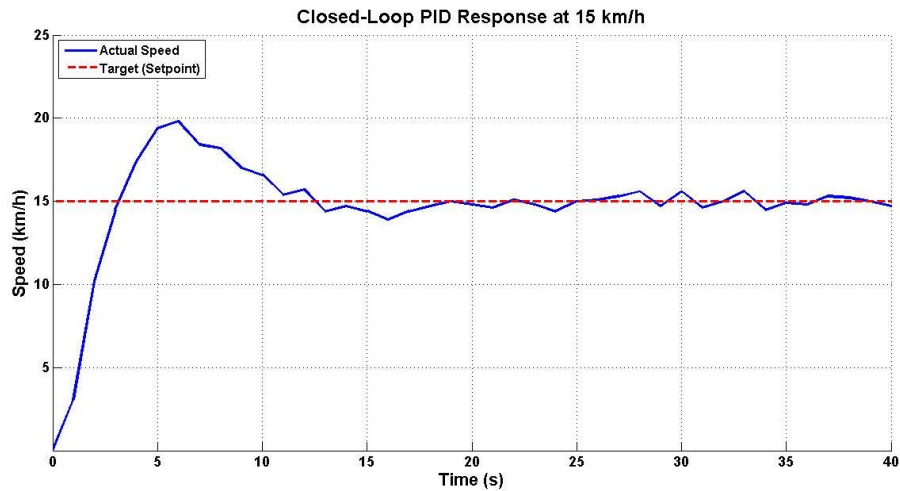
Berdasarkan Gambar 4.21 dan tabel 4.12 tentang parameter respon waktu sistem, respons sistem kontrol kecepatan motor BLDC untuk mencapai *setpoint* 10 km/jam menunjukkan karakteristik respons *underdamped*. Sistem menghasilkan reaksi awal yang sangat cepat, dibuktikan dengan nilai *rise time* sebesar 2 detik. Meskipun demikian, akselerasi awal yang agresif tersebut menimbulkan adanya *overshoot* yang cukup tinggi hingga mencapai 40%, di mana kecepatan aktual menyentuh titik puncak sekitar 14 km/jam pada detik ke-5. Setelah melewati fase transien dan beberapa osilasi, *Settling time* baru di capai

pada waktu 18 detik untuk meredam fluktuasi dan masuk ke dalam rentang toleransi kestabilan. Terlepas dari adanya lonjakan pada fase awal, kinerja algoritma PID pada pengujian ini dinilai bekerja dengan sangat baik, yang dibuktikan secara telak oleh keberhasilannya dalam mengeliminasi simpangan statis (*steady-state error*) secara penuh. Nilai *steady-state error* yang tercatat mendekati 0 km/jam menandakan bahwa aksi pengontrolan telah sukses mengakumulasi dan mengoreksi sisa *error*, sehingga kecepatan aktual motor pada akhirnya berhimpit sempurna dengan garis kecepatan referensi yang diinginkan dan mampu dipertahankan secara konstan untuk durasi operasional selanjutnya.

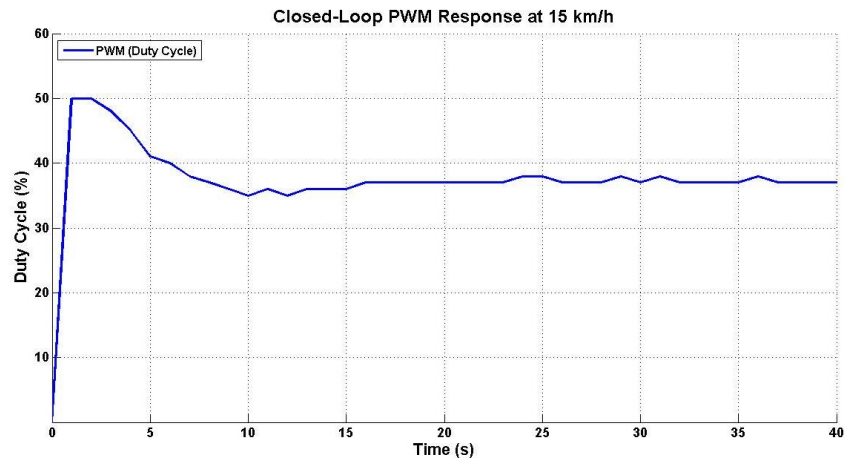
Dinamika kecepatan ini berkorelasi langsung dengan grafik respons *duty cycle*. Pada gambar 4.22 dapat dilihat di detik-detik awal, PID memberikan sinyal kendali PWM yang agresif hingga *duty cycle* menembus sekitar 48% guna menghasilkan torsi awal untuk mencapai *setpoint*. Begitu sistem mendeteksi terjadinya *overshoot*, nilai *duty cycle* diturunkan secara drastis ke kisaran 32-33% untuk membuat putaran roda jadi lebih pelan. Setelah memasuki kondisi *steady-state*, sinyal *duty cycle* menemukan titik keseimbangannya dan stabil di kisaran 34-33%. Secara keseluruhan, pengontrol PID telah berhasil menjaga akurasi kecepatan secara absolut pada jangka panjang, namun nilai *overshoot* transien sebesar 40% mengindikasikan bahwa akselerasi di awal yang cukup tinggi akan mempengaruhi kehalusan start dari motor BLDC yang digunakan.

4.6.2 Pengujian *Close-loop* kecepatan 15 Km/h

Pada pengujian ini, sistem diberikan kontrol PID dengan *set point* kecepatan di 20 Km/h. Lalu sistem di nyalakan untuk mengetahui grafik respon sistemnya. Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.18 berikut.



Gambar 4. 23 Respon kecepatan sistem *Close-loop* 15 km/h



Gambar 4. 24 Kurva respon PWM *Close-loop* 15 km/h

Berdasarkan gambar 4.23 dan 4.24 dapat dilihat kalau implementasi PID pada setpoint 10 Km/h menghasilkan respons dari *duty cycle* yang terlihat pada Gambar 4.24 menunjukkan pola yang stabil setelah melewati fase saturasi awal, sehingga rugi-rugi daya saat proses *switching* menjadi minimal sehingga pelepasan energi panas pada enam buah MOSFET IRFP250 menurun secara drastis, sehingga suhu operasional rangkaian tetap terjaga dan terhindar dari potensi kerusakan permanen akibat *overheat*. Dari segi dinamika laju kendaraan,

profil PWM yang secara konsisten dipertahankan di kisaran 37% hingga 38% ini berada dengan aman di atas ambang batas kritis 32%, memastikan aktuator tidak pernah kehilangan daya dorong magnetiknya secara mendadak. Hasilnya, motor BLDC mampu mempertahankan traksi kecepatan secara konstan dengan efisiensi energi yang optimal. Untuk menilai respons kecepatan sistem, dilakukan pengukuran parameter respons waktu dengan perhitungan sebagai berikut.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *close-loop* dengan set point 15 km/h selama 2,5 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19 *overshoot* yang tercatat mencapai 32%.

c. *Settling Time*

Sistem stabil pada *set point* 15 Km/h. menggunakan persamaan 2.20 didapati waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-17.

d. *Steady-state Error*

Sistem *close-loop* mampu mengeliminasi *steady-state error* berdasarkan perhitungan dengan persamaan 2.21 dikarenakan sistem mampu menjaga kecepatan untuk tetap berada pada *set point*

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu pada tabel 4.13 untuk mempermudah penganalisaan.

Tabel 4. 13 Parameter respon waktu sistem *Close-loop* 15 Km/h

Setpoint (Km/h)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)
15	2,5	17	32	0

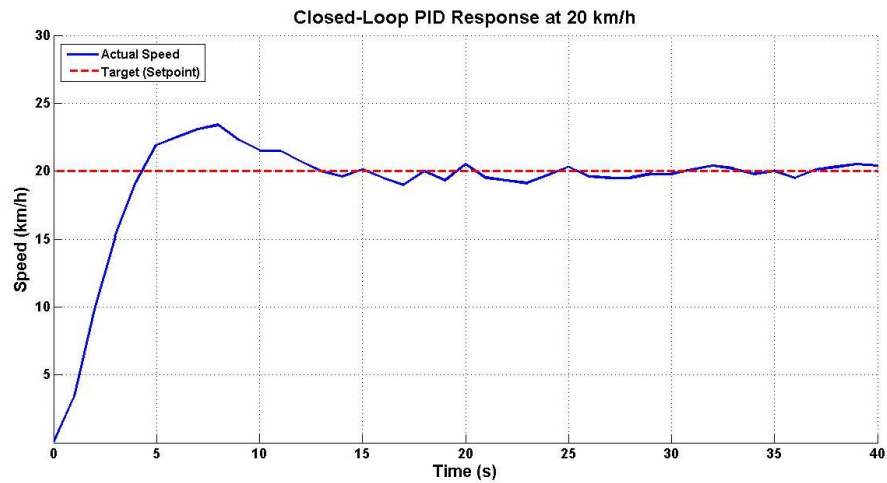
Berdasarkan Gambar 4.18 dan tabel 4.13 tentang parameter respon waktu sistem, respons sistem kontrol kecepatan motor BLDC untuk mencapai *setpoint* 15 km/jam menunjukkan karakteristik respons *underdamped*. Sistem menghasilkan reaksi awal yang cukup cepat, dibuktikan dengan nilai *rise time* sebesar 2,5 detik. Meskipun demikian, akselerasi awal yang agresif tersebut menimbulkan adanya *overshoot* sebesar 32% yang tergolong lumayan, di mana kecepatan aktual menyentuh titik puncak sekitar 19,8 km/jam pada detik ke-6. Setelah melewati fase transien dan beberapa osilasi, *Settling time* baru di capai pada waktu 18 detik untuk meredam fluktuasi dan masuk ke dalam rentang toleransi kestabilan. *Steady-state error* yang mendekati 0 km/jam, yang berarti kecepatan aktual pada akhirnya berhimpit dengan kecepatan referensi yang diinginkan, membuktikan kalau PID bekerja cukup baik pada sistem ini.

Karakteristik lonjakan kecepatan tersebut berbanding lurus dengan profil sinyal kendali yang ditunjukkan pada grafik respons *duty cycle* pada gambar 4.24. Pada empat detik pertama pengujian, kompensator PID secara instan memerintahkan sinyal *duty cycle* pada nilai batasan maksimalnya, yakni di angka 50%, untuk menaikkan kecepatan motor hingga menyentuh nilai *set point*. Begitu kecepatan aktual terdeteksi melampaui *setpoint*, kontroler memberikan aksi pengereman elektrik dengan menekan nilai *duty cycle* secara tajam hingga menyentuh level 35%. Setelah kecepatan masuk ke dalam zona toleransi kestabilan, sistem PID menjaga *duty cycle* agar tetap di angka sekitar 37% hingga 38% agar sistem syabil. Osilasi yang kecil pada *duty cycle* di fase tunak ini merupakan bentuk kompensasi aktif dari parameter integral dan derivatif untuk agar kecepatan sepeda listrik dapat terus dipertahankan di angka 15 km/jam.

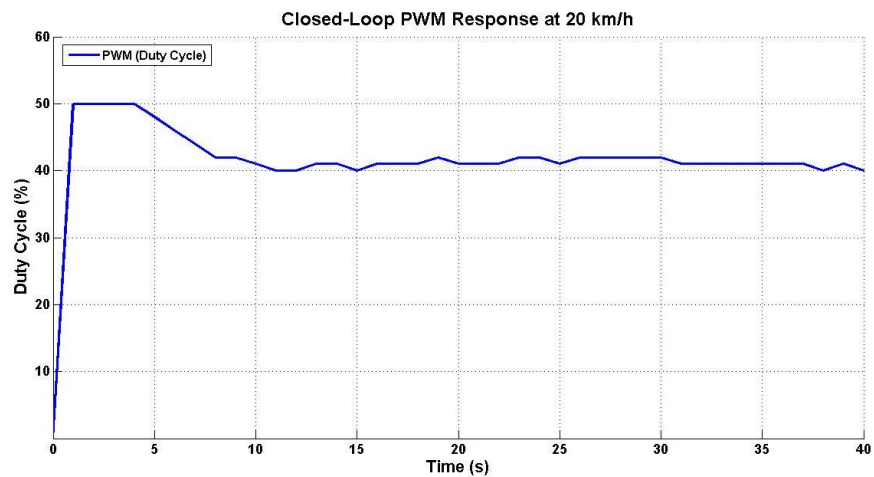
4.6.3 Pengujian *Close-loop* kecepatan 20 Km/h

Pada pengujian ini, sistem diberikan kontrol PID dengan *set point* kecepatan di 20 Km/h. Lalu sistem di nyalakan untuk mengetahui grafik respon

sistemnya Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.20 berikut.



Gambar 4. 25 Respon kecepatan sistem *Close-loop* 20 km/h



Gambar 4. 26 Kurva respon PWM *Close-loop* 20 km/h

Berdasarkan gambar 4.25 dan 4.26 dapat dilihat kalau implementasi PID pada setpoint 20 Km/h menghasilkan grafik respons dari *duty cycle* yang terlihat pada Gambar 4.26 cukup stabil di kisaran 41% hingga 42% setelah melewati fase saturasi awal, sehingga rugi-rugi daya saat proses *switching* menjadi minimal. Hal ini membuat beban kerja serta pelepasan energi panas pada enam buah

MOSFET IRFP250 menurun secara drastis, sehingga suhu operasional rangkaian tetap terjaga dan terhindar dari potensi kerusakan permanen akibat *overheat*. Dari segi dinamika laju kendaraan, profil PWM yang secara konsisten dipertahankan di atas ambang batas kritis 32% ini memastikan aktuator tidak pernah kehilangan daya dorong magnetiknya secara mendadak. Hasilnya, pergerakan rotasi motor BLDC menjadi sangat halus dan mampu mempertahankan traksi kecepatan secara konstan dengan efisiensi energi yang optimal. Untuk menilai respon kecepatan sistem, dilakukan pengukuran parameter respon waktu dengan perhitungan sebagai berikut.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *close-loop* dengan set point 20 km/h selama 4 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19 *overshoot* yang tercatat pada pengujian 20 km/h sebesar 17%.

c. *Settling Time*

Sistem stabil pada *set point* 20 Km/h. menggunakan persamaan 2.20 didapati waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-13.

d. *Steady-state Error*

Sistem *close-loop* mampu mengeliminasi *steady-state error* berdasarkan perhitungan dengan persamaan 2.21 dikarenakan sistem mampu menjaga kecepatan untuk tetap berada pada *set point*.

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.14 untuk mempermudah penganalisaan.

Tabel 4. 14 Parameter respon waktu sistem *Close-loop* 20 Km/h

Setpoint (Km/h)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)
20	4	13	17	0

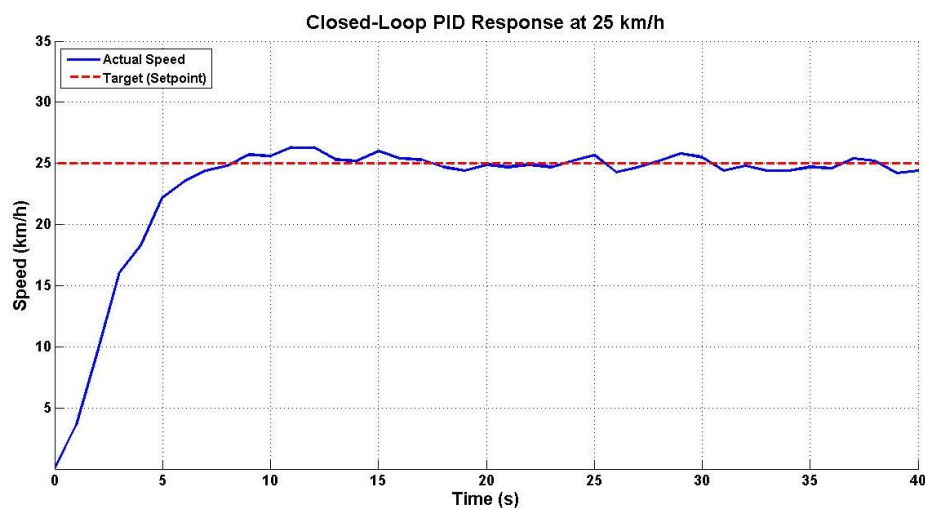
Berdasarkan Tabel 4.14, pengujian respons sistem kontrol kecepatan pada *setpoint* 20 km/jam menunjukkan karakteristik *underdamped*. *Rise time* yang dibutuhkan oleh sistem untuk mencapai *set point* yakni sebesar 4 detik, nilai tergolong cepat untuk waktu akselerasi motor mencapai *set point*. Namun, momentum dari akselerasi tersebut memicu terjadinya *overshoot* sebesar 17%, yang menyebabkan kecepatan aktual mencapai angka 23,4 km/jam pada sekitar detik ke-8. Setelah melewati fase lonjakan dan osilasi transien tersebut, aksi kompensasi dari parameter PID perlahan meredam fluktuasi secara efektif sehingga *settling time* dapat dicapai pada waktu 13 detik, durasi di mana sistem mulai mampu mempertahankan laju kendaraan agar secara konsisten berada di dalam rentang pita toleransi kestabilan 5%. Sama halnya dengan hasil pengujian pada *set point* kecepatan yang lebih rendah, kinerja optimal dan keandalan dari kontroler PID pada pengujian ini terbukti secara meyakinkan dari kemampuannya dalam mengeliminasi simpangan statis (*steady-state error*) menjadi 0 km/jam. Pada fase tunak atau *steady-state* ini, kecepatan kendaraan berhasil dijaga secara presisi agar terus berhimpit dengan garis *set point* 20 km/jam, yang mengindikasikan bahwa kontrol integral telah bekerja dengan sempurna dalam mengoreksi sisa *error* akibat rugi-rugi gesekan mekanis tanpa membiarkan kecepatan motor melambat.

Berdasarkan gambar 4.26, dapat dilihat kalau dinamika respons kecepatan sistem berbanding lurus dengan sinyal kendali. Pada 4 detik pertama, sinyal *duty cycle* langsung naik maksimal dan tertahan secara mendatar di nilai 50% yang

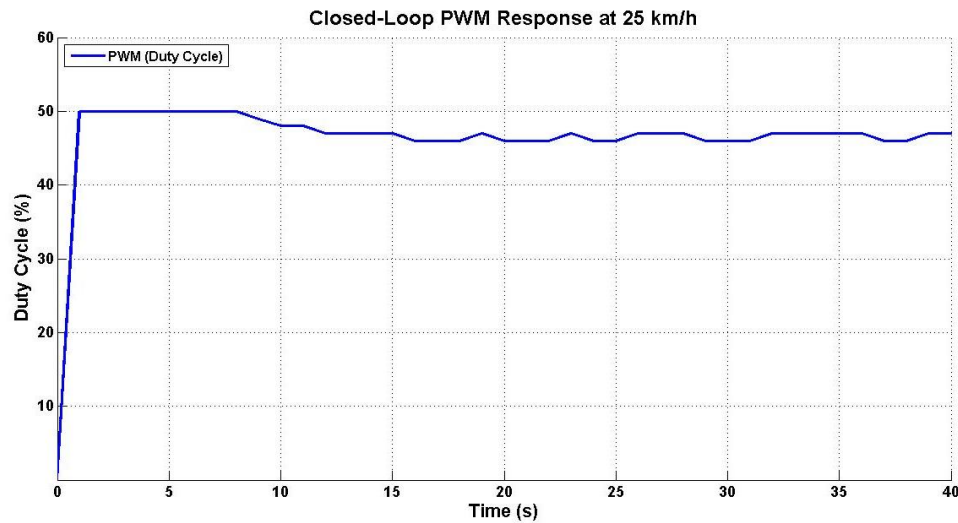
mana merupakan batas maksimal sebagai respon untuk mengatasi inersia mekanis di awal, hal tersebut membuktikan bahwa sistem menjalankan fungsi pembatasan sinyal kendali maksimal dengan baik sebagai upaya untuk mencegah beban arus berlebih pada *driver six steps*. Begitu kecepatan menembus angka 20 km/jam, kontroler merespons dengan menurunkan *duty cycle* secara signifikan ke kisaran 40% untuk menurunkan kecepatan motor. Setelah memasuki kondisi *steady-state*, *duty cycle* menemukan nilai idealnya dan stabil di rentang 41% hingga 42%. Secara komparatif, persentase *overshoot* pada pengujian 20 km/jam ini cenderung menurun dibandingkan pengujian pada *setpoint* yang lebih rendah, yang menandakan bahwa parameter kendali PID ($K_p = 0,035$ $K_i = 0,004$ dan $K_d = 0,016$) memiliki stabilitas operasional yang sedikit lebih baik pada rentang kecepatan tinggi, meskipun efek akselerasi awal masih agak tinggi.

4.6.4 Pengujian *Close-loop* kecepatan 25 Km/h

Pada pengujian ini, sistem diberikan kontrol PID dengan *set point* kecepatan di 25 Km/h. Lalu sistem di nyalakan untuk mengetahui grafik respon sistemnya. Setelah di jalankan, motor BLDC menghasilkan grafik kecepatan sesuai gambar 4.22 berikut.



Gambar 4. 27 Respon kecepatan sistem *Close-loop* 25 km/h



Gambar 4. 28 Kurva respon PWM *Close-loop* 25 km/h

Berdasarkan gambar 4.22 dan 4.23 dapat dilihat kalau implementasi PID pada setpoint 20 Km/h menghasilkan grafik respons dari *duty cycle* yang terlihat pada Gambar 4.23 cukup stabil pada rentang 46% hingga 47% setelah fase akselerasi terlewati, sehingga rugi-rugi daya saat proses *switching* menjadi minimal. Hal ini membuat beban kerja serta pelepasan energi panas pada enam buah MOSFET IRFP250 menurun secara drastis, sehingga suhu operasional rangkaian tetap terjaga dan terhindar dari potensi kerusakan permanen akibat *overheat*. Dari segi dinamika laju kendaraan, profil PWM yang secara konsisten dipertahankan jauh di atas ambang batas kritis 32% memastikan aktuator tidak pernah kehilangan daya dorong magnetiknya secara mendadak. Hasilnya, pergerakan rotasi motor BLDC menjadi sangat halus dan mampu mempertahankan traksi kecepatan secara konstan dengan efisiensi energi yang optimal. Untuk menilai respon kecepatan sistem, dilakukan pengukuran parameter respon waktu dengan perhitungan sebagai berikut.

a. *Rise Time*

Berdasarkan perhitungan menggunakan persamaan 2.18 didapati *rise time* sistem *close-loop* dengan set point 25 km/h selama 7 detik.

b. *Overshoot*

Berdasarkan perhitungan menggunakan persamaan 2.19 *overshoot* yang tercatat pada pengujian 25 km/h sebesar 5,2%.

c. *Settling Time*

Sistem stabil pada *set point* 25 Km/h. menggunakan persamaan 2.20 didapati waktu untuk sistem mulai stabil dengan toleransi 5% dimulai dari detik ke-13.

d. *Steady-state Error*

Sistem *close-loop* mampu mengeliminasi *steady-state error* berdasarkan perhitungan dengan persamaan 2.21 dikarenakan sistem mampu menjaga kecepatan untuk tetap berada pada *set point*.

Setelah dilakukan analisis dan perhitungan berdasarkan kurva respon kecepatan menggunakan parameter respon waktu, hasil perhitungannya disajikan pada tabel 4.15 untuk mempermudah penganalisaan.

Tabel 4. 15 Parameter respon waktu sistem *Close-loop* 25 Km/h

Setpoint (Km/h)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)
25	7	13	5,2	0

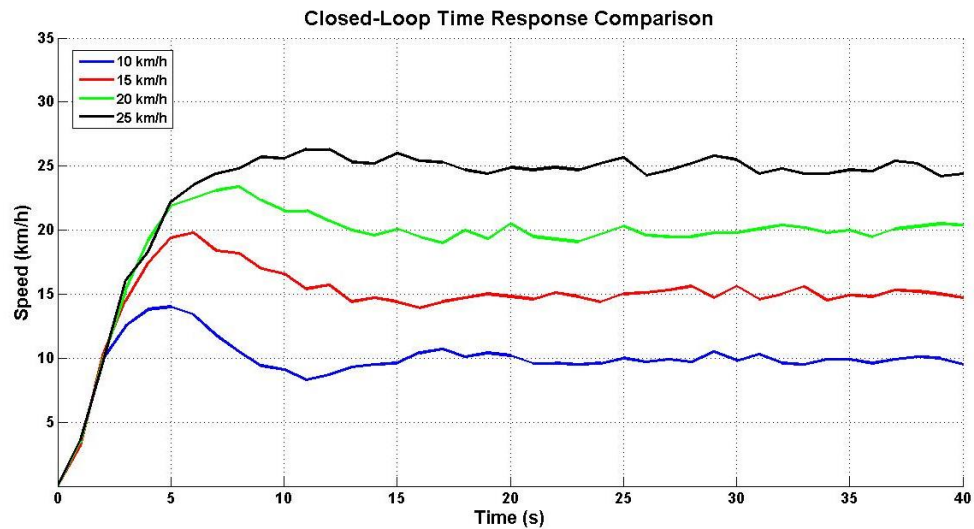
Berdasarkan Gambar 4.22 dan tabel 4.15, pengujian respons kecepatan dengan *setpoint* 25 km/jam menunjukkan kinerja sistem yang sangat stabil dan cukup ideal. Sistem mencatatkan *rise time* sebesar 7 detik. Meskipun membutuhkan durasi akselerasi yang sedikit lebih lama dibandingkan pengujian pada *setpoint* yang lebih rendah, karakteristik kurva yang melandai ini berhasil menekan lonjakan kecepatan pada saat akselerasi. Hal tersebut dibuktikan dengan nilai *overshoot* yang turun drastis menjadi hanya sebesar 5,2%. Pada

pengoperasian fisik sebuah sepeda listrik, minimnya persentase *overshoot* ini merupakan sebuah keunggulan karena laju motor saat mendekati target kecepatan menjadi jauh lebih mulus dan terhindar dari sentakan mekanis karena kecepatan yang terlalu tinggi di awal. Selanjutnya, *settling time* tercapai pada detik ke-13, dan PID berhasil menjaga kecepatan aktual sistem agar tetap berhimpitan dengan *set point* 25 Km/h sehingga diperoleh nilai *steady-state error* sebesar 0 km/jam.

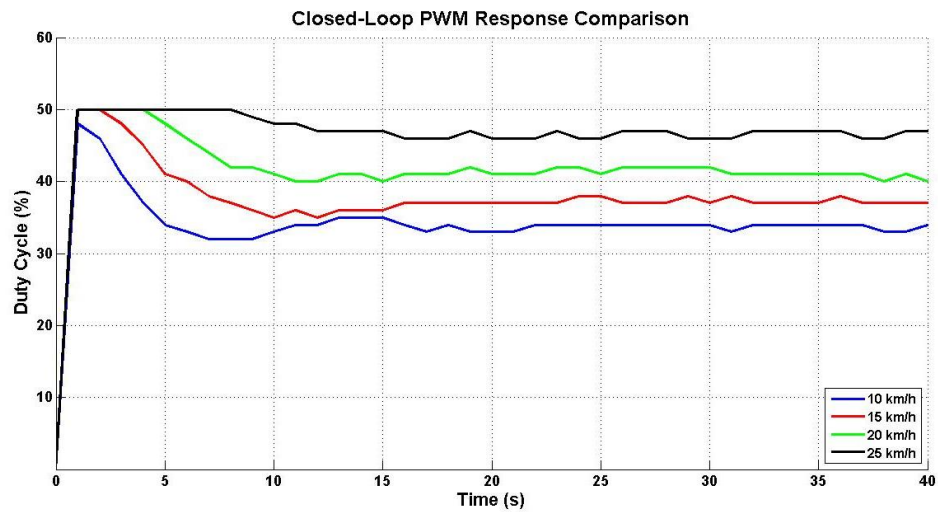
Pada fase transien yakni detik 0 hingga 8, sinyal *duty cycle* langsung melonjak dan mengalami saturasi pada batas atas 50%. Garis lurus mendarat ini menunjukkan bahwa program mikrokontroler telah menjalankan pembatasan sinyal kontrol secara efektif di angka 50% untuk menahan lonjakan suplai arus. Namun, hal tersebutlah yang membuat *rise time* sistem menjadi sedikit lebih lama di banding percobaan sebelumnya. Begitu kecepatan menyentuh target 25 km/jam, momentum pergerakan motor sudah berkurang, sehingga kontroler hanya perlu menurunkan *duty cycle* secara perlahan. Memasuki *isteady-state*, nilai *duty cycle* menemukan angka stabilnya pada kisaran 46% hingga 47%. Secara keseluruhan, pengujian ini membuktikan bahwa parameter PID yang dirancang bekerja dengan sangat baik pada *set point* kecepatan yang lebih tinggi.

4.6.5 Perbandingan Respon Kecepatan *Close-loop*

Pengujian ini menyajikan komparasi terhadap respons kecepatan sistem dalam konfigurasi *closed-loop*. Pengujian dilakukan menggunakan algoritma kendali Proportional-Integral-Derivative (PID) pada empat *set point* kecepatan yang sama, yaitu 10, 15, 20, dan 25 km/jam. Perbandingan ini bertujuan untuk memvalidasi efektivitas dari kontroler PID dalam mengeliminasi *steady-state error* secara presisi, serta mengevaluasi kemampuannya dalam memperbaiki parameter respon transien sistem seperti *Rise Time*, *Settling Time*, dan *Overshoot* di berbagai variasi kecepatan operasional.



Gambar 4. 29 Perbandingan kurva respon kecepatan *Close-loop* pada seluruh varian set point



Gambar 4. 30 Perbandingan kurva respon PWM *Close-loop* pada seluruh varian set point

Untuk mempermudah proses analisis, maka parameter respon waktu pada tiap *set point* kecepatan pada pengujian ini akan di tempat pada tabel, 4.16 dibawah ini.

Tabel 4. 16 Perbandingan parameter respon waktu sistem *Close-loop*

Setpoint (Km/h)	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)
10	2	18	40	0
15	2,5	17	32	0
20	4	13	17	0
25	7	13	5,2	0

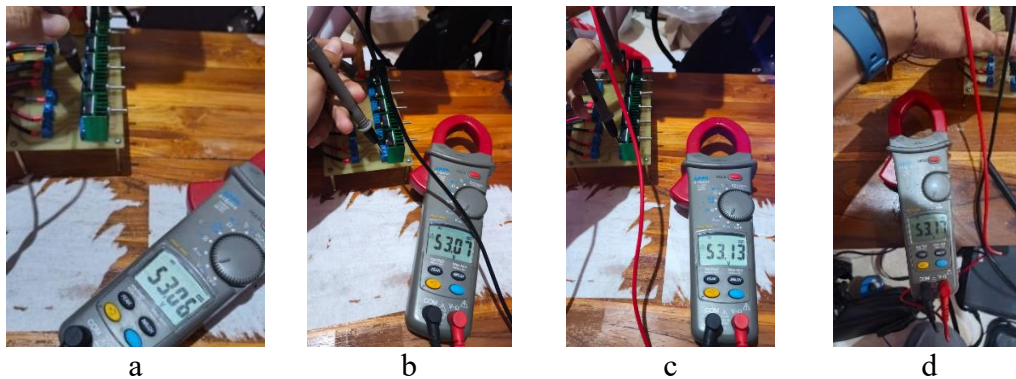
Berdasarkan grafik respons kecepatan 4.24 dan tabel parameter transien hasil pengujian 4.16, penerapan kontroler PID terbukti berhasil menghilangkan *steady-state error* pada seluruh *set poin* pengujian. Keberhasilan ini menyelesaikan permasalahan sistem *Open-loop* yang tmasih mempunyai *steady-state error* pada pengujiannya. Analisis lebih lanjut memperlihatkan adanya hubungan terbalik nilai *setpoint* dengan persentase *overshoot* pada tiap pengujian. Pada *setpoint* rendah (10 km/jam), sistem menghasilkan *overshoot* tertinggi mencapai 40% dan mencatatkan waktu *settling time* terlama, yakni 18 detik. Sebaliknya, seiring peningkatan *setpoint* menuju 25 km/jam, lonjakan transien teredam secara signifikan hingga 5,2% dengan *settling time* yang jauh lebih singkat di 13 detik. *Rise time* bertambah seiring naiknya *set poin* dari 2,5 detik menjadi 7 detik akibat rentang akselerasi yang lebih jauh dan pembatasan nilai *duty cycle* maksimal, penurunan *overshoot* pada kecepatan tinggi ini mengindikasikan bahwa nilai konstanta $K_p = 0,05$ $K_i = 0,006$ dan $K_d = 0,016$ yang diterapkan bekerja jauh lebih optimal dan stabil saat sistem dioperasikan pada kecepatan tinggi.

Untuk melindungi MOSFET dari potensi kerusakan akibat lonjakan arus *inrush* yang tinggi, diimplementasikan batasan maksimal *duty cycle* PWM sebesar 50%. Berdasarkan Gambar 4.25, terlihat bahwa pada fase transien awal, kalkulasi kontroler PID untuk *setpoint* 15, 20, dan 25 km/jam menghasilkan

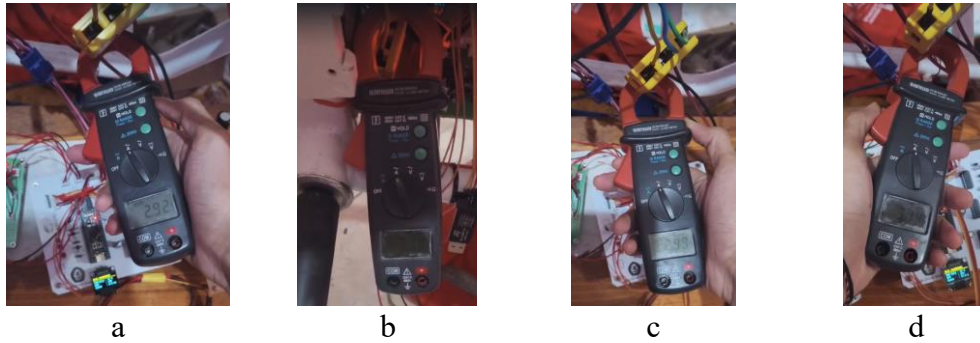
sinyal keluaran yang melampaui *duty cycle* yang diizinkan sistem, sehingga sistem secara paksa memotong sinyal tersebut dan menahannya di angka 50%. Kondisi saturasi ini berlangsung selama beberapa detik hingga kecepatan aktual mendekati target dan nilai error kecepatan mengecil. Tertahannya sinyal PWM di nilai puncak yang sama inilah yang menjadi penyebab utama mengapa kurva percepatan pada ketiga titik uji tersebut berhimpit pada fase awal akselerasi. Kurva PWM baru mulai terdistribusi dan menyesuaikan diri menuju nilai *steady-state* masing-masing yakni sekitar 34% untuk 10 km/jam hingga 46% untuk 25 km/jam ketika sistem telah keluar dari area saturasi.

4.7 Pengukuran Daya Inverter 3 Fasa

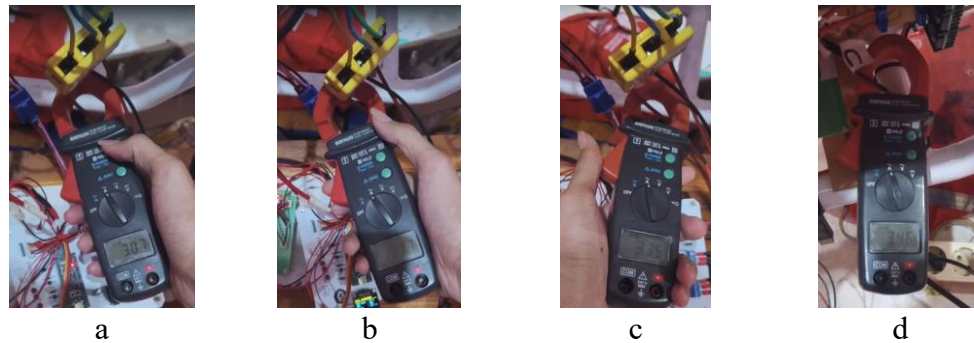
Pengujian ini dilakukan untuk pengukuran dan analisis daya kelistrikan pada rangkaian *driver* motor BLDC, yang mencakup data pengukuran tegangan, arus rata-rata, serta kalkulasi daya total. Data diambil pada sistem *Open-loop* dan *closed-loop* dengan PID pada 4 buah setpoint pengujian kecepatan 10, 15, 20, dan 25 km/jam pada saat keadaan *steady-state*. Data tegangan dan arus di ambil menggunakan multimeter dan clamp meter untuk mengetahui berapa tegangan sistem saat bekerja sesuai dengan gambar 4.31 dan pengukuran arus *Open-loop* 4.32, serta pengukuran arus *Close-loop* 4.33 dibawah ini. Lalu data tegangan dan arus akan di catat lalu di hitung untuk mendapatkan nilai daya operasionalnya.



Gambar 4. 31 a.) Tegangan 10 Km/h b.) tegangan 15 Km/h c.) tegangan 20 km/h d.) tegangan 25 km/h



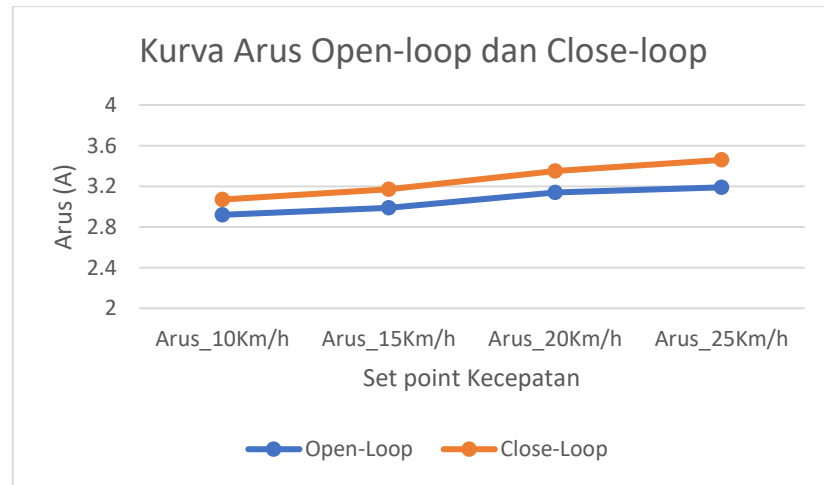
Gambar 4. 32 a.) Arus *Open-loop* 10 Km/h b.) Arus *Open-loop* 15 Km/h c.) Arus *Open-loop* 20 Km/h d.) Arus *Open-loop* 25 Km/h



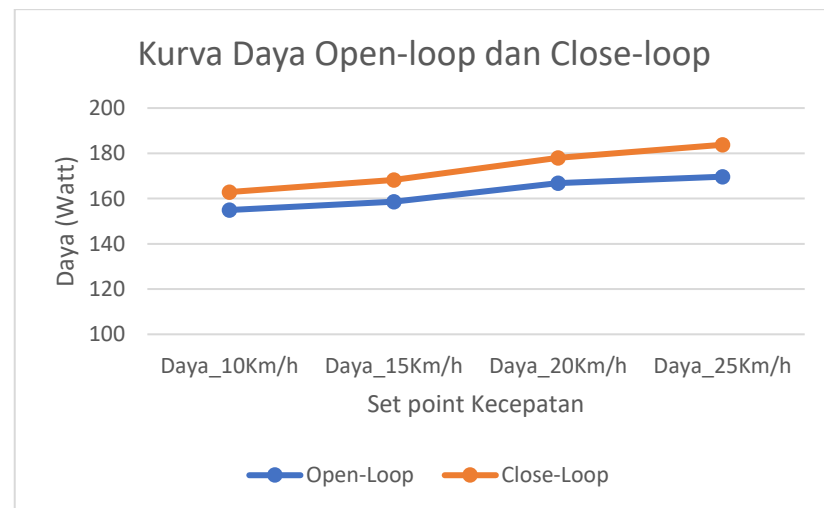
Gambar 4. 33 a.) Arus *Close-loop* 10 Km/h b.) Arus *Close-loop* 15 Km/h c.) Arus *Close-loop* 20 Km/h d.) Arus *Close-loop* 25 Km/h

Berdasarkan Gambar 4.31, Gambar 4.32, dan Gambar 4.33 pada lampiran pengujian, dilakukan pengukuran tegangan dan arus *driver* pada saat motor BLDC beroperasi dalam kondisi tunak (*steady-state*). Pada pengukuran tegangan masukan *inverter* yang ditunjukkan oleh Gambar 4.31, level tegangan DC dari baterai LiFePO4 terpantau sangat stabil melintasi keempat variasi *set point* kecepatan, dengan nilai yang hanya bergeser tipis di kisaran 53,06 V (pada 10 km/jam) hingga 53,17 V (pada 25 km/jam). Kestabilan ini membuktikan bahwa baterai mampu menopang *DC bus* dengan keandalan tinggi tanpa mengalami *voltage drop* yang berarti saat kecepatan operasional ditingkatkan. Sementara itu, pengukuran arus fasa menggunakan *clamp meter* pada Gambar 4.32 (pengujian *open-loop*) dan Gambar 4.33 (pengujian *closed-loop*) merepresentasikan karakteristik penarikan beban yang berbanding lurus dengan laju motor; semakin tinggi target kecepatan yang

ditetapkan, semakin besar pula arus yang dibutuhkan untuk mempertahankan torsi putaran.



Gambar 4. 34 Kurva Arus sistem *Open-loop* dan *Close-loop*



Gambar 4. 35 Kurva Daya sistem *Open-loop* dan *Close-loop*

Berdasarkan gambar 4.34 dan 4.35 dapat dilihat jika kenaikan arus berbanding lurus dengan kenaikan daya yang dihasilkan. Untuk mempermudah proses analisa, data tegangan, arus, dan daya akan ditampilkan pada tabel 4.17 berikut.

Tabel 4. 17 Data tegangan, arus, dan daya sistem *Open-loop* dan *Close-loop*

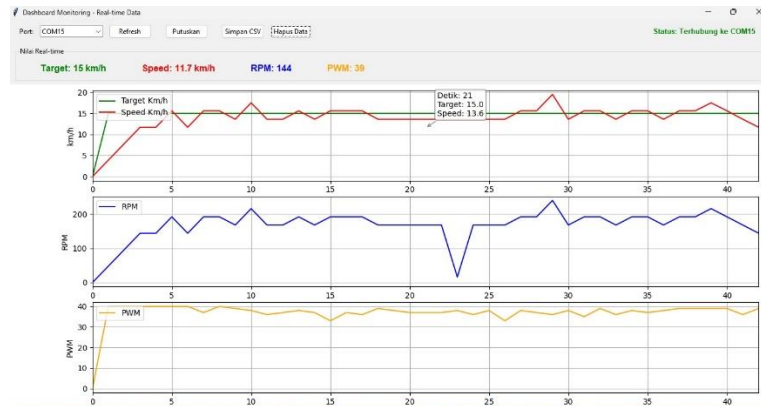
<i>Set point</i> (Km/h)	<i>Open-loop</i>			<i>Close-loop</i>		
	Tegangan (V)	Arus (A)	Daya (Watt)	Tegangan (V)	Arus (A)	Daya (Watt)
10	53,06	2,92	154,94	53,06	3,07	162,89
15	53,07	2,99	158,68	53,07	3,17	168,23
20	53,13	3,14	166,82	53,13	3,35	177,99
25	53,17	3,19	169,67	53,17	3,46	183,79

Berdasarkan tabel 4.17, Hasil pengujian menunjukkan bahwa sistem *closed-loop* secara konsisten menyerap daya yang lebih tinggi dibandingkan dengan sistem *Open-loop* pada seluruh *set point* kecepatan yang di uji. Contohnya pada *set point* 20 km/jam, sistem *closed-loop* mengonsumsi daya sebesar 177,99 W, meningkat sekitar 6,7% dari sistem *Open-loop* yang hanya mengonsumsi 166,82 W. Peningkatan konsumsi energi ini merupakan akibat dari diterapkannya algoritma PID pada sistem *Close-loop*. Pada sistem *Close-loop*, kontroler secara aktif melakukan komputasi dan mengirimkan PWM dengan maksimal nilai 50% pada fase akselerasi awal guna mempercepta *Rise Time*, serta terus-menerus melakukan koreksi osilasi untuk mempertahankan *steady-state* di angka nol. Sebaliknya, sistem *Open-loop* hanya memberikan PWM dengan *duty cycle* yang statis sejak awal pengujian, sehingga beban kerja dari catu daya cenderung lebih pasif dan menghasilkan rata-rata penarikan arus yang lebih rendah secara keseluruhan.

4.8 Pengujian Sensor Kecepatan

Pada pengujian sensor kecepatan. Kami membandingkan angka RPM yang dihasilkan oleh sensor kecepatan dengan tachometer untuk mengetahui apakah terdapat selisih pembacaan kecepatan antara sensor kecepatan yang digunakan dengan *tachometer*. Pengambilan data kecepatan aktual motor BLDC dilakukan dengan memanfaatkan sinyal *feedback* dari sensor Hall yang diolah melalui fungsi

interrupt pada mikrokontroler. Untuk pembacaan RPM mentah dapat dilihat pada gambar 4.36 berikut.



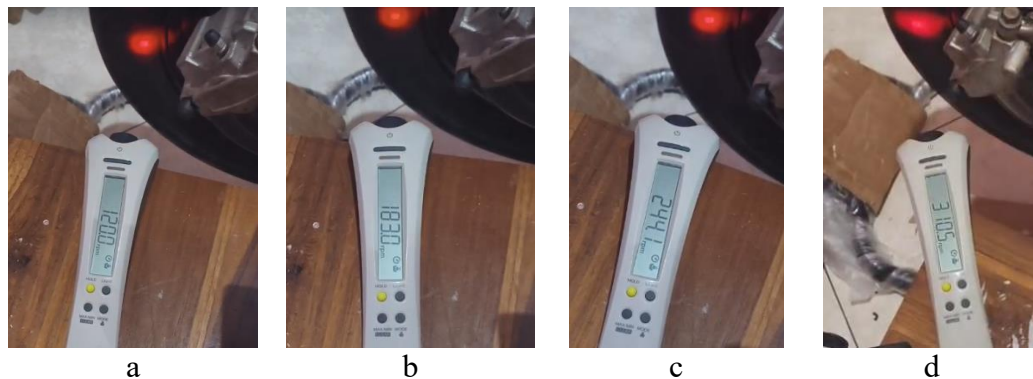
Gambar 4. 36 Respon Kecepatan sistem PID tanpa filter EMA

Pada gambar 4.36 terlihat jelas adanya fluktuasi ekstrem dan lonjakan pembacaan RPM yang tidak wajar. Instabilitas ini berakar dari kerentanan fungsi *interrupt* terhadap hilangnya deteksi sinyal masuk (*missed tick*) akibat siklus waktu *sampling* komputasi yang sangat cepat, yakni 100 ms. Pada rentang waktu tersebut, keterlambatan mikrokontroler dalam membaca satu sinyal saja akan langsung dikalkulasi sebagai deviasi *error* kecepatan yang masif, yaitu sebesar 25 RPM. Guna mengatasi masalah anomali pembacaan tersebut, diimplementasikan algoritma filter *Exponential Moving Average* (EMA) dengan pembobotan parameter *alpha* 0,2.



Gambar 4. 37 Respon Kecepatan sistem PID dengan filter EMA

Terlihat pada gambar 4.37, di mana grafik RPM dan kecepatan aktual menjadi jauh lebih mulus dan terbebas dari lonjakan data yang palsu. Keandalan data sensor kecepatan hasil filter EMA ini memegang peranan yang sangat krusial untuk sistem kontrol PID. Apabila data umpan balik dibiarkan cacat, kalkulasi kompensator PID akan bereaksi secara agresif terhadap fluktuasi palsu tersebut dan memicu perubahan sinyal *duty cycle* PWM yang bergetar dan tidak terkendali. Dengan adanya filter EMA, parameter kecepatan yang masuk ke dalam komparator PID menjadi representasi laju fisis motor yang sesungguhnya, sehingga kontroler mampu meredam *error* dengan tepat dan menghasilkan sinyal kendali PWM yang stabil untuk mempertahankan laju kendaraan secara halus. Setelah itu kecepatan motor aktual akan di ambil menggunakan *tachometer* kemudian di bandingkan dengan data kecepatan aktual dari *hall* sensor yang telah di beri filter EMA.



Gambar 4. 38 a.) tachometer *set point* 10 Km/h b.) tachometer *set point* 15 Km/h c.) tachometer *set point* 20 Km/h d.) tachometer *set point* 25 Km/h

Pada pengujian ini mengambil nilai kecepatan pada tiap *set point* kecepatan seperti pengujian sebelumnya pada kondisi *steady-state* agar kecepatan motor stabil. Untuk sensor kecepatan yang dipakai pada sistem yakni *hall* sensor digital bawaan dari motor BLDC yang digunakan. Data hasil pembacaan sensor dikirim lewat uart. Pembacaan rpm oleh sensor dan selisih nya dengan tachometer dapat dilihat pada tabel 4.18 berikut.

Tabel 4. 18 Data Pengujian sensor kecepatan

<i>Set point</i>	Sensor Kecepatan (rpm)	Alat Ukur Kecepatan (rpm)	Selisih (rpm)	Error (%)
10 Km/h	124	120	4	3.33%
15 Km/h	180	183	3	1.64%
20 Km/h	245	244	1	0.4%
25 Km/h	313	310	3	0.97%

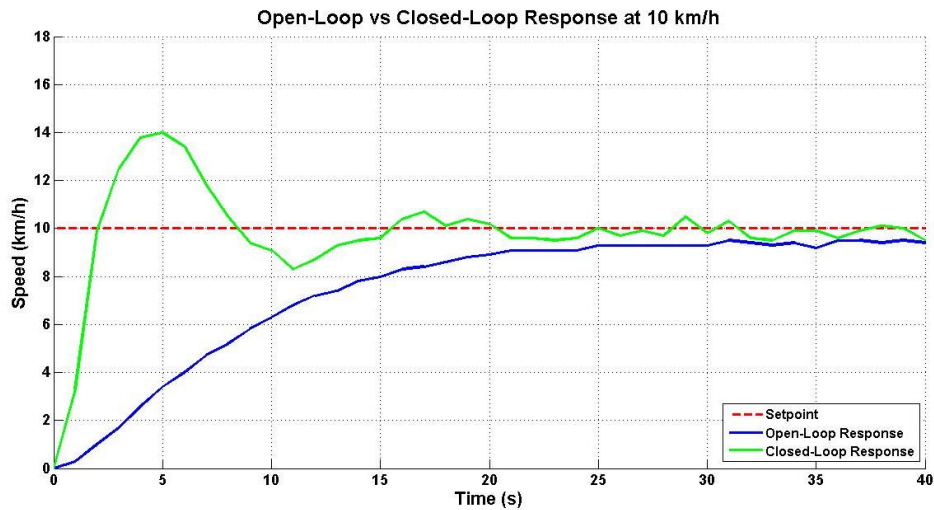
Berdasarkan tabel 4.18, hasil pengujian menunjukkan tingkat akurasi dari *hall* sensor yang digunakan cukup presisi terhadap alat ukur tachometer yang digunakan. Pengukuran yang didapati oleh sensor memiliki selisih rpm 1 hingga 4 rpm dengan nilai error maksimal yang tercatat sebesar 3,33% pada kondisi *steady-state*, angka tersebut tergolong cukup rendah karena selisihnya tidak jauh dengan pembacaan tachometer. Pada *sampling time* 100 ms dengan resolusi 24 *tick* per rotasi, satu selisih pembacaan *tick* secara diskrit setara dengan deviasi sebesar 25 RPM. Keberhasilan sistem dalam menekan deviasi pembacaan hingga di bawah 5 RPM membuktikan keberhasilan dari pengaplikasian filter EMA. Filter ini terbukti mampu menanggulangi lonjakan pembacaan RPM sehingga hasil pembacaan menjadi lebih akurat.

4.9 Perbandingan *Open-loop* dan *Close-loop*

Pada Percobaan kali ini, penguji membandingkan kurva respon sistem *Open-loop* dan *Close-loop* untuk melihat sistem mana yang lebih baik meliputi parameter respon waktu seperti *overshoot*, *rise time*, *settling time*, dan *steady-state error*. Pengujian perbandingan dilakukan berdasarkan *set point* kecepatan dari 10 Km/h, 15 Km/h, 20 Km.h, dan 25 Km/h. Pada pengujian ini parameter PID yang digunakan adalah $K_p = 0,05$ $K_i = 0,006$ dan $K_d = 0,016$ yang di dapat dari hasil *fine tuning* pada percobaan sebelumnya. Hasil pengujian akan di tampilkan pada beberapa point berikut.

4.9.1 Perbandingan *Open-loop* VS *Close-loop* pada 10 Km/h

Pengujian dengan *set point* 10 Km/h merepresentasikan respon sistem pada kecepatan rendah. Perbandingan kurva pada kecepatan ini bertujuan untuk membandingkan kedua respon sistem tersebut. Berikut adalah respon kurva parameter respons kecepatan untuk target 10 km/jam.



Gambar 4. 39 Kurva *Open-loop* VS *Close-loop* Set point 10 Km/h

Dari gambar 4.32 dapat dilihat kalau respon sistem *Close-loop* lebih cepat dibandingkan dengan *Open-loop*. Meskipun kontroler PID mampu memaksa sistem mencapai kecepatan target secara jauh lebih responsif dibandingkan metode *Open-loop*, injeksi sinyal PWM yang berlebih pada fase awal mengakibatkan persentase *overshoot* tertinggi di antara semua pengujian.

Tabel 4. 19 Perbandingan *Open-loop* vs *Close-loop* 10 km/h

Sistem	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)	Arus (A)	Daya (Watt)
<i>Open-loop</i>	15	20	0	0,6	2,92	154.94
<i>Close-loop</i>	2	18	40	0	3,07	162.89
Selisih	13	2	40	0.6	0,15	7,95

Berdasarkan grafik respons kecepatan 4.32 dan komparasi parameter pada Tabel 4.19, penerapan kontroler PID pada *setpoint* 10 km/jam menghasilkan perubahan respon sistem yang sangat drastis. Sistem *Close-loop* memiliki keunggulan yang cukup jauh dibanding dengan sistem *Open-loop*. Dengan *Rise Time* 13 detik lebih cepat, *Settling Time* 2 detik lebih cepat, serta *Steady-State Error* yang dapat di hilangkan berkat adanya kontrol integral. Kekurangan dari sistem *Close-loop* yakni *overshoot* yang dihasilkan mencapai 40% yang mana masih cukup tinggi, adanya *overshoot* merupakan konsekuensi dari penerapan PID yang mana membuat akselerasi awal menjadi jauh lebih cepat untuk mencapai *set point* namun menimbulkan lonjakan kecepatan yang cukup tinggi sebelum di redam oleh sistem. Dari segi konsumsi daya, sistem *Close-loop* mengonsumsi daya lebih tinggi sekitar 7,95 Watt dibanding sistem *Open-loop*.

4.9.2 Perbandingan *Open-loop* VS *Close-loop* pada 15 Km/h

Pengujian dengan *set point* 15 Km/h merepresentasikan respon sistem pada kecepatan rendah. Perbandingan kurva pada kecepatan ini bertujuan untuk membandingkan kedua respon sistem tersebut. Berikut adalah respon kurva parameter respons kecepatan untuk target 15 km/jam.



Gambar 4. 40 Kurva *Open-loop* VS *Close-loop* Set point 15 Km/h

Dari gambar 4.33 dapat dilihat kalau respon sistem *Close-loop* lebih cepat dibandingkan dengan *Open-loop*. Meskipun kontroler PID mampu memaksa sistem mencapai kecepatan target secara jauh lebih responsif, sama seperti percobaan sebelumnya, respon kecepatan masih mengalami *overshoot*.

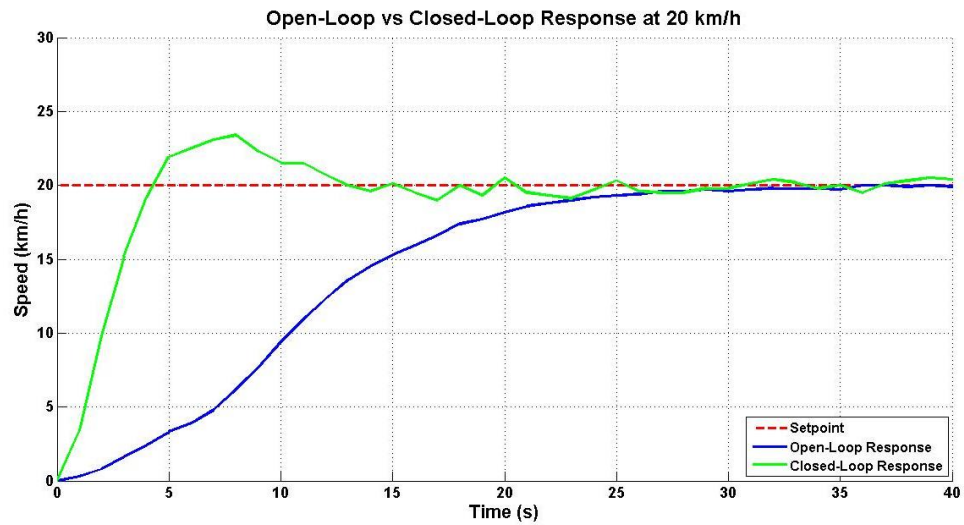
Tabel 4. 20 Perbandingan *Open-loop* vs *Close-loop* 15 km/h

Sistem	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)	Arus (A)	Daya (Watt)
<i>Open-loop</i>	17,5	25	0	0,3	2,99	158,68
<i>Close-loop</i>	2,5	17	32	0	3,17	168,23
Selisih	15	8	32	0.3	0,18	9,55

Berdasarkan grafik respons kecepatan 4.33 dan komparasi parameter pada Tabel 4.20, penerapan kontroler PID pada *setpoint* 15 km/jam menghasilkan perubahan respon sistem yang sangat signifikan. Sistem *Close-loop* memiliki keunggulan yang telak jika dibanding dengan sistem *Open-loop*. Dengan *Rise Time* 15 detik lebih cepat, *Settling Time* 8 detik lebih cepat, serta *Steady-State Error* yang dapat di hilangkan dengan adanya kontrol integral pada PID. Namun sistem *Close-loop* bukan tanpa kekurangan, *overshoot* yang dihasilkan mencapai 32% dikarenakan sistem menaikkan PWM sangat tinggi di awal untuk mencapai *set point* dengan lebih cepat, akan tetapi hal tersebut menimbulkan *overshoot* yang cukup tinggi. Dari segi konsumsi daya, sistem *Close-loop* mengonsumsi daya lebih tinggi sekitar 9,55 Watt dibanding sistem *Open-loop*.

4.9.3 Perbandingan *Open-loop* VS *Close-loop* pada 20 Km/h

Pengujian dengan *set point* 20 Km/h merepresentasikan respon sistem pada kecepatan rendah. Perbandingan kurva pada kecepatan ini bertujuan untuk membandingkan kedua respon sistem tersebut. Berikut adalah respon kurva parameter respons kecepatan untuk target 20 km/jam.



Gambar 4. 41 Kurva *Open-loop* VS *Close-loop* Set point 20 Km/h

Dari gambar 4.33 dapat dilihatkalau respon sistem *Close-loop* lebih cepat dibandingkan dengan *Open-loop*. Meskipun kontroler PID mampu memaksa sistem mencapai kecepatan target secara lebih cepat namun tetap menghasilkan *overshoot*, akan tetapi *overshoot* yang dihasilkan lebih minim jika dibandingkan dengan *set point* sebelumnya.

Tabel 4. 21 Perbandingan *Open-loop* vs *Close-loop* 20 km/h

Sistem	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)	Arus (A)	Daya (Watt)
<i>Open-loop</i>	16,5	22	0	0,2	3,14	166.82
<i>Close-loop</i>	4	13	17	0	3,35	177.99
Selisih	12,5	9	17	0,2	0,21	11,17

Berdasarkan grafik respons kecepatan 4.34 dan komparasi parameter pada Tabel 4.21, penerapan kontroler PID pada *setpoint* 20 km/jam menghasilkan perubahan respon sistem yang lebih baik dibanding *set point* sebelumnya. Keunggulan sistem *Close-loop* dapat dilihat dengan *Rise Time* 12,5 detik lebih cepat, *Settling Time* 9 detik lebih cepat, serta *Steady-State Error* yang dapat di

hilangkan oleh PID. Kontrol PID pada sistem *Close-loop* terlihat lebih baik pada *set point* kecepatan tinggi yang dibuktikan dengan, *overshoot* yang dihasilkan pada 17%, meskipun nilainya tidak kecil, namun angka tersebut hampir setengah kali lebih rendah bila di banding *set point* kecepatan 15 Km/h, penyebabnya masih sama karena sistem menaikkan PWM sangat tinggi di awal untuk mencapai *set point* dengan lebih cepat, akan tetapi pada kecepatan yang lebih tinggi, penurunan kecepatan pada saat *overshoot* untuk mencapai *set point* tidak se jauh pada kecepatan rendah sehingga overshoot yang dihasilkan juga tidak terlalu tinggi. Dari segi konsumsi daya, sistem *Close-loop* mengonsumsi daya lebih tinggi sekitar 11,17 Watt dibanding sistem *Open-loop*.

4.9.4 Perbandingan *Open-loop* VS *Close-loop* pada 25 Km/h

Pengujian dengan *set point* 25 Km/h merepresentasikan respon sistem pada kecepatan rendah. Perbandingan kurva pada kecepatan ini bertujuan untuk membandingkan kedua respon sistem tersebut. Berikut adalah respon kurva parameter respons kecepatan untuk target 25 km/jam.



Gambar 4. 42 Kurva *Open-loop* VS *Close-loop* Set point 25 Km/h

Dari gambar 4.35 dapat dilihat kalau respon sistem *Close-loop* lebih cepat dibandingkan dengan *Open-loop*. Pada *set point* ini, *overshoot* yang di hasilkan jauh lebih minimal dibanding *set point* lainnya, namun *rise time* nya yang paling lama dari *set point* pengujian lainnya.

Tabel 4. 22 Perbandingan *Open-loop* vs *Close-loop* 25 km/h

Sistem	Rise Time (s)	Settling Time (s)	Overshoot (%)	Steady-state Error (Km/h)	Arus (A)	Daya (Watt)
<i>Open-loop</i>	15	23	0	0,8	3,19	169.67
<i>Close-loop</i>	7	13	5,2	0	3,46	183.79
Selisih	8	10	5,2	0,8	0,27	14,12

Berdasarkan grafik respons kecepatan 4.35 dan komparasi parameter pada Tabel 4.22, penerapan kontroler PID pada *setpoint* 25 km/jam menghasilkan perubahan respon sistem yang paling baik dibanding *set point* sebelumnya. Keunggulan sistem *Close-loop* dapat dilihat dengan *Rise Time* 8 detik lebih cepat, *Settling Time* 10 detik lebih cepat, serta *Steady-State Error* yang dapat di hilangkan oleh PID. Pada *set point* ini, dimana menggunakan kecepatan paling tinggi pada pengujian, membuktikan kalau parameter kontrol PID yang digunakan pada sistem *Close-loop* lebih cenderung cocok untuk kecepatan tinggi, hal tersebut dibuktikan dengan *overshoot* yang cukup minim di 5,2% yang mana merupakan nilai terkecil dari berbagai pengujian *set point* yang dilakukan, sistem memaksakan *duty cycle* maksimal pada saat akselerasi yakni di 50% sehingga *set point* dapat di capai dengan cepat. Setelah mencapai *set point* untuk pertama kali, PWM langsung di turunkan namun tidak banyak karena pada *set point* 25 km/h, *duty cycle* yang di pakai agar sistem stabil di sekitar 46% hingga 57%. Oleh sebab itu sistem tidak membutuhkan waktu yang lama untuk menurunkan kecepatan ke *set point* dan *overshoot* jadi lebih minim. Dari segi konsumsi daya, sistem *Close-loop* mengonsumsi daya lebih tinggi sekitar 14,12 Watt dibanding sistem *Open-loop*, yang mana menjadi yang tertinggi jika di banding *set point* yang lain.

Hal tersebut membuktikan kalau semakin tinggi *set point* pengujian, maka sistem *Close-loop* bekerja lebih baik dibanding *Open-loop*, hal tersebut dibuktikan dengan *overshoot* yang semakin mengecil, *rise time* yang lebih rendah, *settling time* yang lebih cepat, serta *steady-state error* yang lebih baik, akan tetapi konsumsi arus dan daya semakin meninggi seiring naiknya *set point* kecepatan pengujian.