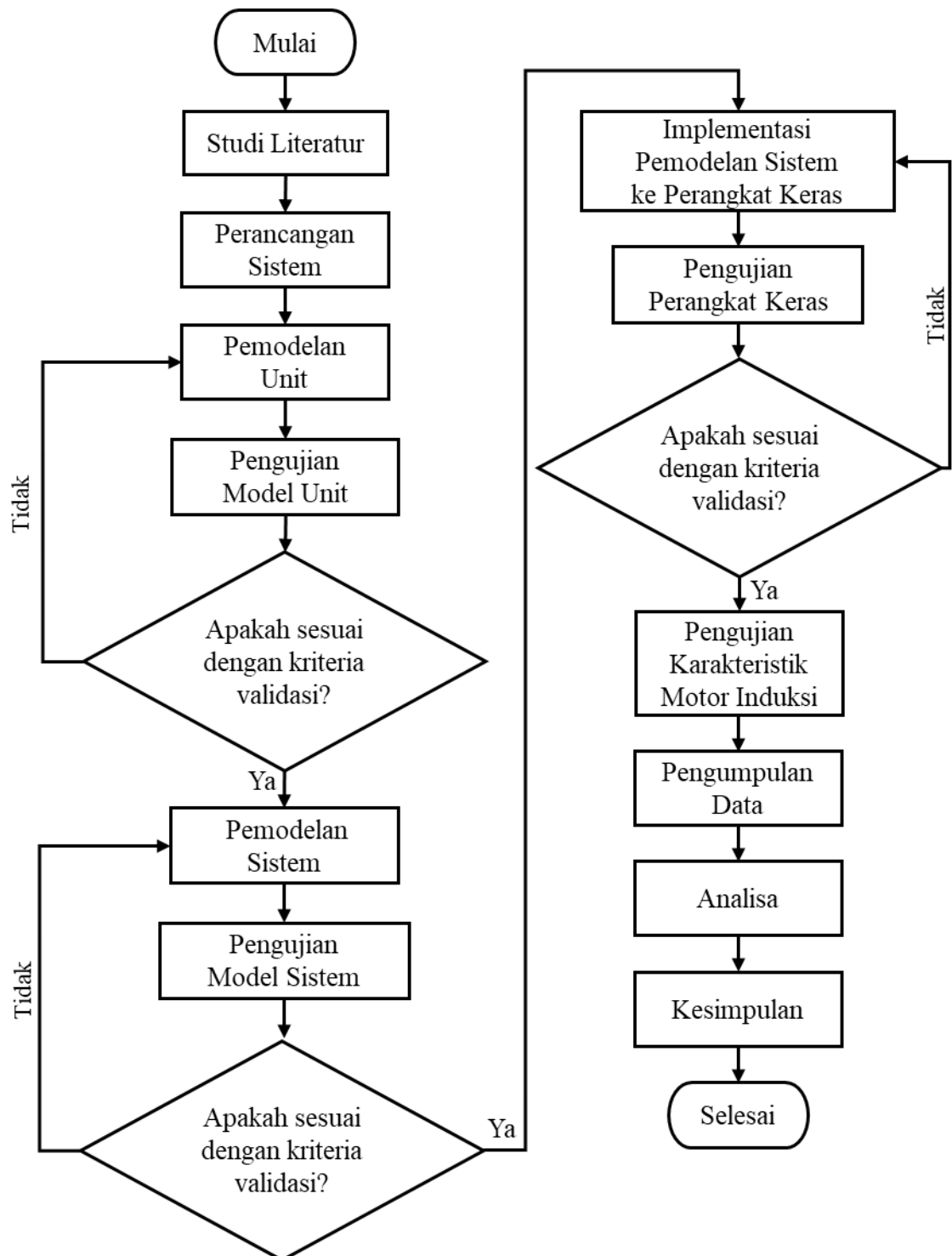


BAB III

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian



Gambar 3.1 Diagram alir penelitian

Berdasarkan diagram alir pada Gambar 3.1, tahapan penelitian ini dapat diuraikan sebagai berikut:

- 1. Mulai:** Tahap awal di mana topik dan tujuan penelitian ditetapkan
 - 2. Studi Literatur:** Melakukan kajian pustaka untuk mengumpulkan landasan teori yang relevan, mencakup motor induksi, metode Field Oriented Control (FOC), SVPWM, inverter, serta pengoperasian perangkat lunak MATLAB/Simulink dan perangkat keras Lucas-Nuelle. Referensi yang digunakan adalah artikel ilmiah berupa jurnal nasional maupun internasional. Referensi juga diambil dari buku yang berhubungan dengan teori-teori yang digunakan dalam penelitian
 - 3. Perancangan Sistem:** Merancang arsitektur sistem kontrol IFOC secara konseptual, termasuk mendefinisikan parameter-parameter utama motor dan sistem kontrol yang akan digunakan
 - 4. Pemodelan Unit:** Menerapkan rancangan setiap komponen sistem (seperti model motor induksi dan model SVPWM) secara terpisah ke dalam blok-blok unit pada perangkat lunak MATLAB/Simulink
 - 5. Pengujian Model Unit:** Setiap unit yang telah dimodelkan (Model Motor Induksi dan Model SVPWM) dijalankan melalui serangkaian pengujian individual. Pengujian ini dirancang untuk menyimulasikan operasi mereka secara terisolasi dan mengumpulkan data kinerja, sesuai dengan prosedur yang dirinci dalam Sub-bab 3.4
- Untuk Model Motor Induksi, pengujian dilakukan dengan memberikan suplai tegangan tiga fasa dan menerapkan kondisi beban yang berbeda (tanpa beban dan berbeban)

- Untuk Model SVPWM, pengujian dilakukan dalam kondisi statis (efisiensi penggunaan tegangan dan THD) dan kondisi dinamis (untuk memvalidasi respons terhadap perubahan referensi)

6. Validasi Model Unit: Data hasil dari "Pengujian Model Unit" dievaluasi dan dibandingkan dengan kriteria validasi teoretis untuk memastikan kinerja dan fungsi setiap model unit sudah sesuai

- Respons Model Motor Induksi dievaluasi berdasarkan kriteria transien dan steady-state yang ditetapkan pada Tabel 3.3
- Respons Model SVPWM dievaluasi berdasarkan kriteria pengujian statis (efisiensi penggunaan tegangan dan THD) dan dinamis (akurasi respons) yang diuraikan dalam Sub-bab 3.4.2

Jika hasil tidak sesuai dengan kriteria tersebut, maka dilakukan perbaikan pada tahap "Pemodelan Unit". Jika hasil sudah valid, penelitian berlanjut ke tahap berikutnya.

7. Pemodelan Sistem: Mengintegrasikan semua unit yang telah tervalidasi menjadi satu model sistem IFOC yang utuh dan fungsional di Simulink

8. Pengujian Model Sistem: Melakukan serangkaian pengujian simulasi pada model sistem IFOC yang telah terintegrasi. Pengujian ini bertujuan untuk memvalidasi kinerja sistem secara keseluruhan berdasarkan skenario-skenario yang telah ditentukan (dirinci pada Sub-bab 3.6). Skenario ini secara spesifik dirancang untuk menguji:

- Skenario A (Respons pelacakan kecepatan)
- Skenario B (Respons terhadap perubahan arus medan)

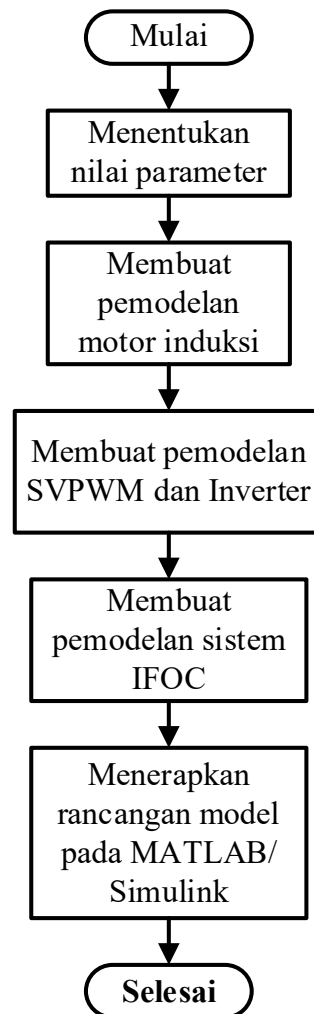
Tujuan utamanya adalah untuk memastikan model sistem mampu memenuhi kriteria validasi pada Tabel 3.4, terutama kemampuan pelacakan kecepatan dan efektivitas *decoupling* sebelum dilanjutkan ke tahap implementasi perangkat keras.

9. **Validasi Model Sistem:** Hasil pengujian sistem simulasi dari Skenario A dan Skenario B (dari tahap "Pengujian Model Sistem") dievaluasi secara menyeluruh. Kinerja model dibandingkan dengan kriteria validasi yang telah ditetapkan pada Tabel 3.4 (dirinci di Sub-bab 3.6), yang secara spesifik menguji kemampuan pelacakan kecepatan, efektivitas *decoupling*, dan respons torsi dan arus. Jika hasil tidak sesuai dengan kriteria tersebut, maka dilakukan perbaikan pada "Pemodelan Sistem". Jika hasil sudah valid, model dianggap siap untuk diimplementasikan.
10. **Implementasi Pemodelan Sistem ke Perangkat Keras:** Model sistem IFOC yang telah tervalidasi disesuaikan dan di-deploy ke perangkat keras modul kontroler Lucas-Nuelle
11. **Pengujian Perangkat Keras:** Melakukan pengujian pada perangkat keras fisik (modul Lucas-Nuelle) yang menjalankan model IFOC yang telah di-deploy. Pengujian ini mengikuti prosedur yang diuraikan dalam Sub-bab 3.8, dengan menjalankan skenario yang sama persis seperti pada tahap "Pengujian Model Sistem". Tujuan dari tahap ini adalah untuk mengumpulkan data kinerja empiris (real-time) dari sistem fisik sebagai persiapan untuk validasi akhir
12. **Validasi Pengujian Perangkat Keras:** Data hasil dari tahap "Pengujian Perangkat Keras" (Skenario A dan B) dievaluasi. Data empiris ini dibandingkan secara langsung dengan kriteria validasi sistem IFOC pada Tabel

3.4. Jika hasil sudah valid dan sesuai (Ya), penelitian berlanjut ke tahap “Pengujian Karakteristik Motor Induksi”. Jika hasil tidak sesuai (Tidak), maka dilakukan evaluasi dan perbaikan model pada tahap "Implementasi Pemodelan Sistem ke Perangkat Keras" dan dilakukan pengulangan "Pengujian Perangkat Keras”.

13. **Pengujian Karakteristik Motor Induksi:** Setelah sistem perangkat keras divalidasi, pengujian eksperimental akhir dilakukan sesuai dengan prosedur pada Sub-bab 3.9. Tahap ini bertujuan untuk menganalisis dan membandingkan secara langsung karakteristik kondisi tunak (steady-state) motor sebelum dan sesudah menggunakan FOC. Pengujian ini melibatkan dua metode untuk perbandingan karakteristik motor sebelum menggunakan FOC (menggunakan metode DOL) dan sesudah menggunakan metode FOC. Data kondisi tunak (seperti kecepatan, torsi, slip, arus, dan faktor daya) dicatat selama pengujian ini untuk menghasilkan kurva karakteristik komparatif.
14. **Pengumpulan Data:** Mengumpulkan dan merekam data hasil dari "Pengujian Model Unit" dan "Pengujian Model Sistem" pada tahap simulasi, "Pengujian Perangkat Keras", dan data komparatif utama dari "Pengujian Karakteristik Motor Induksi”
15. **Analisa:** Menganalisis dan membandingkan data hasil simulasi dengan data hasil pengujian alat untuk memvalidasi model dan mengevaluasi kinerja sistem
16. **Kesimpulan:** Menarik kesimpulan dari keseluruhan hasil analisa dan memberikan saran untuk pengembangan di masa depan
17. **Selesai:** Penelitian dianggap selesai

3.2 Perancangan sistem



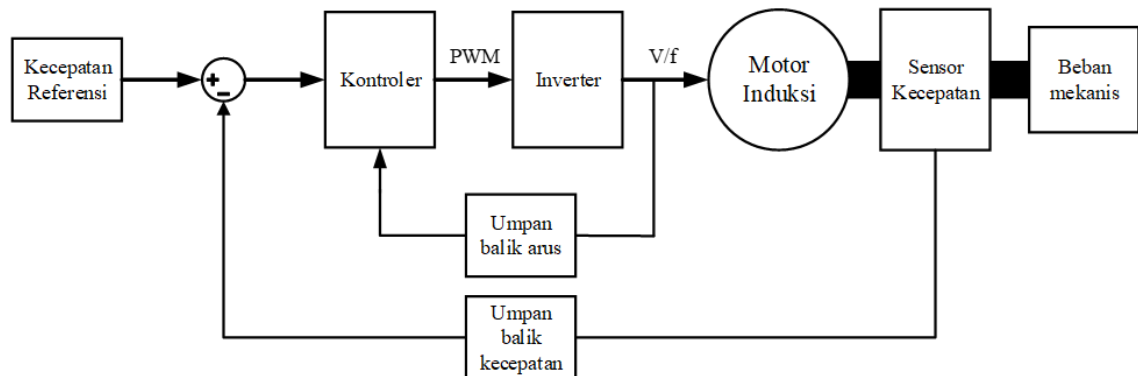
Gambar 3.2 Diagram alir perancangan pemodelan sistem

Diagram alir perancangan sistem pada Gambar 3.2 diatas menjelaskan tahap-tahap perancangan pemodelan sistem *Indirect Field Oriented Control* sebagai berikut:

3.2.1 Perancangan Sistem Kendali

Perancangan sistem kendali pada penelitian ini menerapkan dua *loop* kendali yang diimplementasikan pada *Digital Signal Processor* (DSP) modul Lucas-Nuelle. Sistem ini terdiri dari dua *loop* kendali utama, yaitu *loop* untuk pengaturan kecepatan dan *loop* untuk pengaturan arus (*current loop*).

Secara garis besar, blok diagram perancangan sistem kendali perangkat keras beserta jalur umpan baliknya ditunjukkan pada Gambar 3.3.



Gambar 3.3 Blok diagram perangkat keras

Berdasarkan Gambar 3.3, mekanisme pengendalian motor dapat dijelaskan sebagai berikut:

1. **Input dan pembeding:** Proses dimulai dengan memberikan kecepatan referensi (*setpoint*) yang diinginkan. Nilai ini kemudian dibandingkan dengan kecepatan aktual motor yang dibaca oleh sensor kecepatan. Selisih antara kedua nilai ini menghasilkan sinyal *error* kecepatan
2. **Blok kontroler:** Sinyal *error* kecepatan masuk ke blok kontroler. Di dalam blok ini, diterapkan metode *Field Oriented Control* (FOC) yang memproses sinyal kesalahan tersebut bersama data dari umpan balik arus. Metode FOC bekerja dengan mentransformasikan arus AC tiga fasa menjadi komponen vektor terpisah (d-q), sehingga kontroler dapat mengatur fluks dan torsi secara independen (*decoupling*) untuk menghasilkan sinyal kendali yang presisi. Keluaran dari blok ini adalah *Pulse Width Modulation* (PWM) yang mengatur inverter
3. **Inverter (aktuator):** Unit ini berfungsi sebagai eksekutor daya yang menjembatani sinyal kendali digital dengan sistem fisik motor. Inverter

menerima sinyal PWM dari kontroler dan mengubah tegangan DC (*DC Link*) menjadi tegangan AC tiga fasa variabel (V/f) untuk disuplai ke stator motor. Dalam konteks aliran daya yang telah dibahas pada Sub-bab 2.1.4, tegangan keluaran inverter ini berperan sebagai tegangan terminal stator. Sesuai dengan Persamaan 2.3 ($P_{in} = \sqrt{3}V I_1 \cos \varphi$), pengaturan amplitudo dan frekuensi tegangan oleh inverter secara langsung menentukan besarnya daya input (P_{in}) yang disuntikkan ke dalam belitan stator.

Dengan mengatur tegangan ini melalui algoritma SVPWM, sistem kendali secara efektif mengatur laju energi listrik yang masuk untuk kemudian dikonversi menjadi daya celah udara (P_g) dan akhirnya menjadi daya mekanik untuk memutar beban, sebagaimana diilustrasikan pada diagram aliran daya Gambar 2.6.

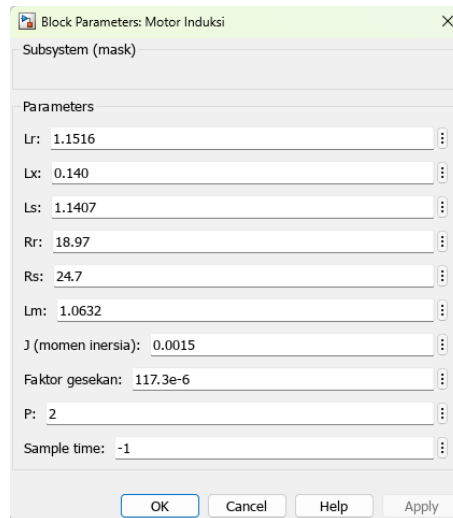
4. **Beban mekanis:** Motor menghasilkan torsi untuk memutar beban mekanis. Jika beban berubah, sensor kecepatan akan mendeteksi perubahan putaran dan mengirimkan sinyal umpan balik, sehingga siklus pengendalian berulang secara *real-time* untuk menjaga kestabilan sistem.

Blok diagram pada Gambar 3.3 di atas menggambarkan arsitektur fisik dan aliran sinyal antar komponen secara umum. Adapun rincian lengkap mengenai algoritma internal dan blok diagram *Field Oriented Control* (FOC) yang diterapkan di dalam kontroler akan dijelaskan secara mendalam pada Sub-bab 3.2.5.

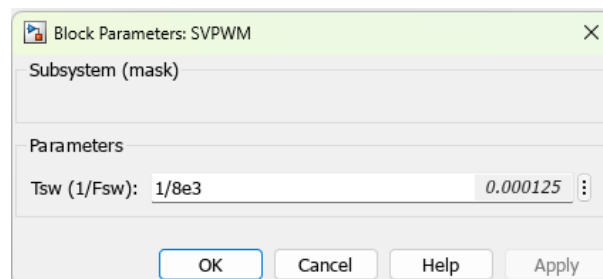
3.2.2 Menentukan nilai parameter

Pada tahap ini, parameter yang dibutuhkan dalam pemodelan didefinisikan pada setiap blok sub-sistem (model unit) menggunakan mode “*Mask editor*” dari

Simulink. Berikut merupakan nilai-nilai parameter yang digunakan pada model unit motor induksi dan SVPWM:



Gambar 3.4 Parameter model motor induksi



Gambar 3.5 Parameter model SVPWM

Data parameter motor induksi yang digunakan pada pemodelan disesuaikan dengan data motor induksi yang tersedia pada modul praktikum *Power Electronics and Drives* Lucas-Nulle. Kemudian untuk parameter periode *switching* (T_{sw}) disesuaikan dengan Modul Percobaan *Power Electronics and Drives* 300 W Lucas-Nulle yang menggunakan frekuensi *switching* inverter (f_{sw}) sebesar 8 kHz.

3.2.3 Perancangan model motor induksi

Pada tahap ini, pemodelan motor induksi dimulai dengan melakukan tinjauan ulang terhadap model matematis motor induksi yang relevan dari Bab II sub-Bab 2.3. Model dinamis motor induksi. Kemudian dilakukan proses penurunan persamaan agar sesuai dengan kebutuhan sistem.

Persamaan arus stator didefinisikan sebagai berikut:

$$\begin{aligned} \frac{di_{s\alpha}}{dt} = & \frac{1}{\sigma \cdot L_s} v_{s\alpha} - \frac{1}{\sigma \cdot L_s} \left(R_s + \frac{L_m^2}{L_r \cdot T_r} \right) \cdot i_{s\alpha} + \left(\frac{L_m}{\sigma \cdot L_s \cdot L_r \cdot T_r} \right) \cdot \lambda_{ra} \\ & + \left(\frac{L_m}{\sigma \cdot L_s \cdot L_r} \right) \cdot \omega_r \cdot \lambda_{r\beta} \end{aligned} \quad (3.1)$$

$$\begin{aligned} \frac{di_{s\beta}}{dt} = & \frac{1}{\sigma \cdot L_s} \cdot v_{s\beta} - \frac{1}{\sigma \cdot L_s} \cdot \left(R_s + \frac{L_m^2}{L_r \cdot T_r} \right) \cdot i_{s\beta} + \left(\frac{L_m}{\sigma \cdot L_s \cdot L_r \cdot T_r} \right) \cdot \lambda_{r\beta} \\ & - \left(\frac{L_m}{\sigma \cdot L_s \cdot L_r} \right) \cdot \omega_r \cdot \lambda_{r\alpha} \end{aligned} \quad (3.2)$$

Dimana:

$v_{s\alpha}, v_{s\beta}$: Tegangan stator sumbu $\alpha - \beta$ (V)

$i_{s\alpha}, i_{s\beta}$: Arus stator sumbu $\alpha - \beta$ (A)

$\lambda_{s\alpha}, \lambda_{s\beta}$: Fluks stator sumbu $\alpha - \beta$ (Wb)

$\lambda_{r\alpha}, \lambda_{r\beta}$: Fluks rotor sumbu $\alpha - \beta$ (Wb)

R_s : Resistansi stator (Ω)

R_r : Resistansi rotor (Ω)

L_s : Induktansi stator (H)

L_r : Induktansi rotor (H)

L_m : Induktansi mutual (H)

T_r : konstanta waktu rotor ($T_r = \frac{L_r}{R_r}$) (s)

σ : Koefisien kebocoran induktansi ($\sigma = L_s - \frac{L_m^2}{L_r}$)

ω_r : Kecepatan sudut rotor (rad/s)

Arus stator $i_{s\alpha}$ dan $i_{s\beta}$ diuraikan dari persamaan tegangan stator (Persamaan 2.17-2.20), persamaan fluks stator (Persamaan 2.21-2.22) dan fluks rotor (Persamaan 2.23-2.24) dalam kerangka acuan stasioner.

Persamaan fluks rotor didefinisikan sebagai berikut:

$$\frac{d\lambda_{r\alpha}}{dt} = -\frac{1}{T_r} \cdot \lambda_{r\alpha} + \frac{L_m}{T_r} \cdot i_{s\alpha} - \omega_r \cdot \lambda_{r\beta} \quad (3.3)$$

$$\frac{d\lambda_{r\beta}}{dt} = -\frac{1}{T_r} \cdot \lambda_{r\beta} + \frac{L_m}{T_r} \cdot i_{s\beta} + \omega_r \cdot \lambda_{r\alpha} \quad (3.4)$$

Dimana:

$v_{s\alpha}, v_{s\beta}$: Tegangan stator sumbu $\alpha - \beta$ (V)

- $i_{s\alpha}, i_{s\beta}$: Arus stator sumbu $\alpha - \beta$ (A)
 $\lambda_{s\alpha}, \lambda_{s\beta}$: Fluks stator sumbu $\alpha - \beta$ (Wb)
 $\lambda_{r\alpha}, \lambda_{r\beta}$: Fluks rotor sumbu $\alpha - \beta$ (Wb)
 R_s : Resistansi stator (Ω)
 R_r : Resistansi rotor (Ω)
 L_s : Induktansi stator (H)
 L_r : Induktansi rotor (H)
 L_m : Induktansi mutual (H)
 T_r : konstanta waktu rotor ($T_r = \frac{L_r}{R_r}$) (s)
 σ : Koefisien kebocoran induktansi ($\sigma = L_s - \frac{L_m^2}{L_r}$)
 ω_r : Kecepatan sudut rotor (rad/s)

Fluks rotor λ_{ra} dan λ_{rb} diuraikan dari persamaan tegangan rotor (Persamaan 2.19-2.20) dan fluks rotor (Persamaan 2.23-2.24) dalam kerangka acuan stasioner.

Kemudian, untuk model mekanis motor yang memuat nilai kecepatan dan torsi di uraikan dari Persamaan 2.35 menjadi persamaan berikut:

$$\omega_r = \int \frac{T_m - T_L - \text{fric} \cdot \omega_r}{J} \quad (3.5)$$

Dimana:

- ω_r : Kecepatan sudut rotor (rad/s)
 T_m : Torsi mekanis (N.m)
 T_L : Beban mekanis (N.m)
 fric : Koefisien gesekan ($\frac{N.m.s}{rad}$)
 J : Momen inersia ($Kg.m^2$)

Kecepatan sudut rotor (ω_r) akan dikonversi menjadi kecepatan putar rotor per menit (n_r) menggunakan persamaan berikut:

$$n_r = \omega_r \cdot \left(\frac{60}{2 \cdot \pi} \right) \quad (3.6)$$

Dimana:

- n_r : Kecepatan putar rotor permenit (RPM)
 ω_r : Kecepatan sudut rotor (rad/s)

Untuk kebutuhan pengujian model unit, digunakan persamaan tegangan tiga fasa sebagai input utama yang memberikan tegangan input ke model unit motor

induksi. Persamaan sumber tegangan tiga fasa yang digunakan adalah sebagai berikut:

$$\begin{aligned} V_a(t) &= V_m \sin(\omega t) \\ V_a(t) &= V_m \sin\left(\omega t - \frac{2\pi}{3}\right) \\ V_a(t) &= V_m \sin\left(\omega t + \frac{2\pi}{3}\right) \end{aligned} \quad (3.7)$$

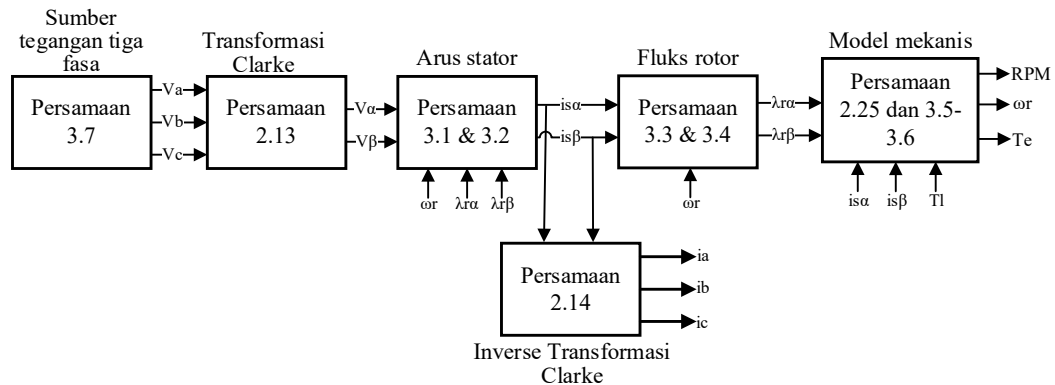
Dengan:

$$\omega = 2\pi f$$

Dimana:

V_a, V_b, V_c : Tegangan tiga fasa (V)
 V_m : Tegangan puncak/amplitudo (V)
 ω : Kecepatan sudut ($\frac{rad}{s}$)
 t : Waktu (s)
 f : Frekuensi (Hz)

Berdasarkan persamaan-persamaan model dinamis motor induksi di atas, disusun blok diagram pemodelan motor induksi yang ditunjukkan pada Gambar 3.6 berikut:



Gambar 3.6 Blok diagram pemodelan motor induksi

Blok diagram pemodelan motor induksi pada Gambar 3.6 menunjukkan model motor induksi tersusun atas beberapa blok yang memiliki fungsi sebagai berikut:

1. Blok "Sumber tegangan tiga fasa" merupakan input utama yang memberikan tegangan listrik tiga fasa (V_a, V_b, V_c) ke model motor induksi. Pembuatan blok ini didasarkan pada Persamaan 3.7 untuk menghasilkan tegangan tiga fasa (V_a, V_b, V_c)
2. Blok "Transformasi Clarke" berfungsi untuk mengubah sinyal tegangan tiga fasa (V_a, V_b, V_c) menjadi sinyal tegangan dua fasa (V_α, V_β) dalam kerangka acuan stasioner, sesuai dengan Persamaan 2.13
3. Blok "Arus stator" berfungsi untuk menghitung komponen arus stator pada sumbu α ($i_{s\alpha}$) dan β ($i_{s\beta}$) sesuai dengan persamaan Persamaan 3.1 dan 3.2
4. Blok "Fluks rotor" berfungsi untuk menghitung komponen fluks rotor pada sumbu alpha ($\lambda_{r\alpha}$) dan beta ($\lambda_{r\beta}$) sesuai dengan Persamaan 3.3 dan 3.4
5. Blok "Model Mekanis" berfungsi untuk menghitung torsi elektromagnetik (T_e) yang dihasilkan oleh motor sesuai dengan Persamaan 2.25, dan menghitung kecepatan sudut rotor (ω_r) sesuai dengan Persamaan 3.5 dan dikonversi menjadi kecepatan putar per menit (RPM) menggunakan Persamaan 3.6
6. Blok "Inverse Transformasi Clarke" berfungsi untuk mengubah kembali sinyal arus stator dua fasa ($i_{s\alpha}, i_{s\beta}$) yang dihasilkan model motor induksi ke arus tiga fasa ($i_{sab}, i_{sbc}, i_{sca}$), sesuai dengan Persamaan 2.14

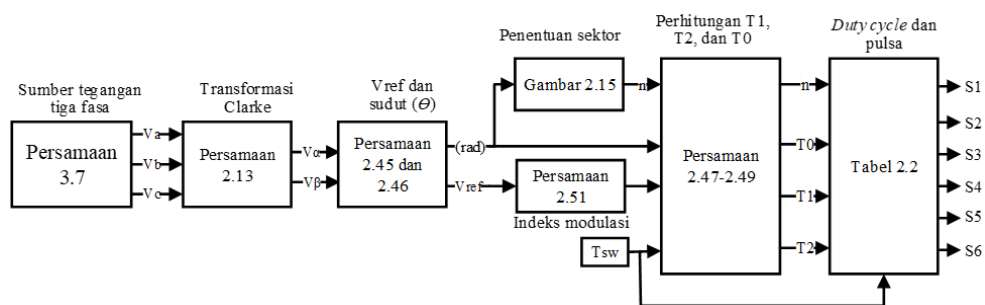
3.2.4 Perancangan model SVPWM

Pada sistem *Field Oriented Control* (FOC), unit SVPWM dan Inverter bekerja sebagai satu kesatuan fungsional yang bertugas untuk mengubah perintah tegangan dari algoritma kontrol menjadi tegangan AC tiga fasa yang akan disuplai ke motor induksi. Perancangan unit ini bertujuan untuk menciptakan sebuah *power stage* yang mampu menghasilkan tegangan dengan amplitudo dan frekuensi yang

dapat diatur secara presisi dan dinamis. Perancangan ini menjembatani antara keluaran *Inverse Park Transformation* ($V_{a_{ref}}$ dan $V_{\beta_{ref}}$) dengan suplai daya aktual untuk motor.

1. Model SVPWM

Pada tahap ini, pemodelan SVPWM didasarkan pada langkah-langkah implementasi yang dijelaskan pada Bab II Sub-Bab “2.6.2 *Space Vector Pulse Width Modulator* (SVPWM)”. Berdasarkan langkah-langkah implementasi tersebut, disusun blok diagram pemodelan SVPWM yang ditunjukkan pada Gambar 3.7 berikut:



Gambar 3.7 Blok diagram pemodelan SVPWM

Blok diagram pemodelan SVPWM yang ditunjukkan pada Gambar 3.7 menunjukan model SVPWM tersusun atas beberapa blok yang memiliki fungsi sebagai berikut:

- Blok "Sumber tegangan tiga fasa" merupakan input tegangan tiga fasa ke model SVPWM. Dalam konteks pengujian model SVPWM, blok ini menghasilkan sumber tegangan sinusoidal ideal yang akan menjadi vektor tegangan referensi SVPWM
- Blok "Transformasi Clarke" berfungsi untuk mengubah sinyal tegangan tiga fasa (V_a, V_b, V_c) menjadi sinyal tegangan dua fasa (V_α, V_β) dalam kerangka

acuan stasioner, sesuai dengan Persamaan 2.13. Dalam implementasi FOC, input V_α dan V_β ini langsung dihasilkan oleh IFOC

- c. Blok “ V_{ref} dan sudut (θ)” berfungsi untuk menerima komponen tegangan V_α dan V_β dari Transformasi Clarke, kemudian menghitung nilai magnitudo tegangan referensi (V_{ref}) menggunakan Persamaan 2.46, serta menghitung sudut (θ) antara V_α dan V_β menggunakan Persamaan 2.45. amplitudo tegangan referensi (V_{ref}) dan sudut (θ) ini adalah input utama untuk blok penentuan sektor dan perhitungan waktu aktif, karena keduanya mendefinisikan posisi dan panjang vektor tegangan referensi di bidang $\alpha - \beta$
- d. Blok “Penentuan sektor” berfungsi untuk menentukan posisi sektor vektor tegangan referensi berdasarkan nilai sudut (θ) yang dihasilkan pada blok sebelumnya. Penentuan sektor sangat krusial karena setiap sektor memiliki kombinasi switching aktif yang berbeda tergantung pada posisi vektor tegangan referensi berada
- e. Blok “Indeks modulasi” berfungsi untuk menghitung indeks modulasi (a) berdasarkan magnitudo vektor tegangan referensi (V_{ref}) dan tegangan DC Link (V_{DC}) menggunakan Persamaan 2.51
- f. Blok “Perhitungan T_1, T_2 , dan T_0 ” berfungsi untuk menghitung durasi waktu aktif untuk dua vektor aktif terdekat (T_1, T_2) dan durasi waktu untuk vektor nol (T_0) dalam satu periode switching (T_s) menggunakan Persamaan 2.47 sampai Persamaan 2.49. Nilai T_1, T_2 , dan T_0 secara langsung menentukan siklus kerja (duty cycle) sesuai dengan proses pembentukan V_{ref} yang ditunjukkan pada Gambar 2.17

- g. Blok “Duty cycle dan pulsa”, berdasarkan nilai T_1 , T_2 , dan T_0 (serta periode *switching* T_s), blok ini menghasilkan sinyal PWM akhir untuk setiap fasa (pulsa gerbang untuk sakelar atas dan bawah pada setiap kaki inverter). Pada blok ini dihasilkan output utama dari model SVPWM yang akan mengendalikan sakelar inverter sesuai dengan tegangan inputnya (tegangan referensi $V_{\alpha_{ref}}$ dan $V_{\beta_{ref}}$)

2. Model inverter

Model inverter terdiri dari enam saklar (misalnya IGBT) yang operasinya dikendalikan langsung oleh sinyal pulsa (*Pulse Width Modulation*) yang dihasilkan oleh blok SVPWM. Perancangan inverter ini sederhana namun fungsional, di mana setiap kaki inverter akan menghasilkan tegangan keluaran berdasarkan status ON/OFF saklar atas dan bawahnya. Keluaran mentah dari inverter bukanlah gelombang sinus murni, melainkan gelombang tegangan AC berbentuk pulsa dengan frekuensi tinggi yang mengandung banyak harmonisa. Untuk mengatasi hal ini, sebuah filter LC dirancang untuk ditempatkan setelah inverter. Filter ini berfungsi untuk meredam komponen harmonisa frekuensi tinggi dan hanya meloloskan komponen frekuensi fundamentalnya.

3.2.5 Perancangan model sistem IFOC

Pada tahap ini, pemodelan sistem *Indirect Field Oriented Control* (IFOC) secara garis besar dibangun berdasarkan blok diagram yang disajikan pada Gambar 2.11 dan arsitektur perangkat keras pada Gambar 3.3. Berdasarkan pada blok diagram tersebut, “Estimator Sudut Fluks Rotor (*Angle*)” diperlukan beberapa proses perhitungan nilai parameter fluks rotor dan kecepatan sudut

1. Estimator fluks rotor (λ_{dr})

Untuk mendapatkan sudut fluks rotor, diperlukan nilai fluks rotor pada kondisi decoupling $\lambda_{qr} = 0$, sesuai dengan pembahasan Sub-Bab 2.4.2, Persamaan 2.33 diturunkan sehingga didapatkan Persamaan 2.41 yang merepresentasikan perubahan fluks rotor berdasarkan arus medan stator (i_{ds})

2. Estimator posisi sudut fluks rotor (θ_e)

Untuk mendapat nilai posisi sudut fluks rotor (θ_e), digunakan Persamaan 2.38 yang membutuhkan nilai kecepatan sudut sinkron (ω_e), sehingga dilakukan proses penguraian Persamaan 2.28 dan 2.32 pada kondisi *decoupling* ($\lambda_{qr} = 0$), sehingga didapatkan Persamaan berikut:

Dengan:

$$\omega_e = \omega_{slip} + \omega_r$$

$$\omega_{slip} = \frac{L_m}{T_r} \cdot \frac{i_{qs}}{\lambda_{dr}}$$

Maka:

$$\omega_e = \frac{L_m}{T_r} \cdot \frac{i_{qs}}{\lambda_{dr}} + \omega_r \quad (3.8)$$

Dimana:

- ω_e : Kecepatan sudut sinkron (rad/s)
- ω_{slip} : Kecepatan sudut *slip* (rad/s)
- ω_r : Kecepatan sudut rotor (rad/s)
- L_m : Induktansi mutual (H)
- T_r : Konstanta waktu rotor ($T_r = \frac{L_r}{R_r}$) (s)
- i_{qs} : Arus beban (A)
- λ_{dr} : Fluks rotor sumbu d (Wb)

Kemudian pada penelitian ini terdapat proses *decoupling*. Proses *decoupling* ini berfungsi untuk mengkompensasi tegangan keluaran dari pengontrol arus (PI). Proses ini dibutuhkan karena, meskipun asumsi $\lambda_{qr} = 0$ pada proses estimasi fluks rotor telah memberikan *decoupling* pada sisi fluks, efek kopling silang yang tidak

diinginkan masih muncul pada tegangan stator (V_{ds} dan V_{qs}) mengacu pada Persamaan 2.26 dan 2.27. Kopling silang ini diakibatkan oleh variabel kecepatan sinkron (ω_e), sehingga akan menyebabkan interaksi antara sumbu d dan q dan menyebabkan kondisi pengendalian medan dan torsi motor induksi yang terpisah tidak akan tercapai.

Secara spesifik, suku-suku yang mengandung perkalian ω_e dengan variabel dari sumbu ortogonalnya (misalnya, $\omega_e \cdot i_{ds}$ atau $\omega_e \cdot i_{qs}$) adalah penyebab utama kopling silang pada tegangan V_{ds} dan V_{qs} .

Untuk mengatasi masalah tersebut dan memastikan kontrol fluks (i_{ds}) dan torsi (i_{qs}) yang sepenuhnya independen, diperlukan sebuah proses decoupling pada keluaran kontroler arus PI. Proses decoupling ini menghitung komponen tegangan kompensasi yang kemudian ditambahkan atau dikurangkan ke output kontroler PI.

Komponen kompensasi tegangan yang pada proses decoupling ini didasarkan pada penurunan persamaan tegangan stator (Persamaan 2.26 dan 2.27). Persamaan untuk komponen kompensasi tegangan adalah sebagai berikut:

Persamaan tegangan kompensasi V_{ds}

$$V_{ds_{kompensasi}} = -\omega_e \cdot \sigma \cdot L_s \cdot i_{qs} + \frac{L_m}{L_r} \cdot p \lambda_{dr} \quad (3.9)$$

Sehingga didapatkan tegangan V_{ds} yang terkompensasi sebagai berikut:

$$V_{ds_{ref}} = V_{ds_{PI}} - V_{ds_{kompensasi}} \quad (3.10)$$

Persamaan tegangan kompensasi V_{qs}

$$V_{qs_{kompensasi}} = \omega_e \cdot \left(\sigma \cdot i_{ds} + \frac{L_m}{L_r} \cdot \lambda_{dr} \right) \quad (3.11)$$

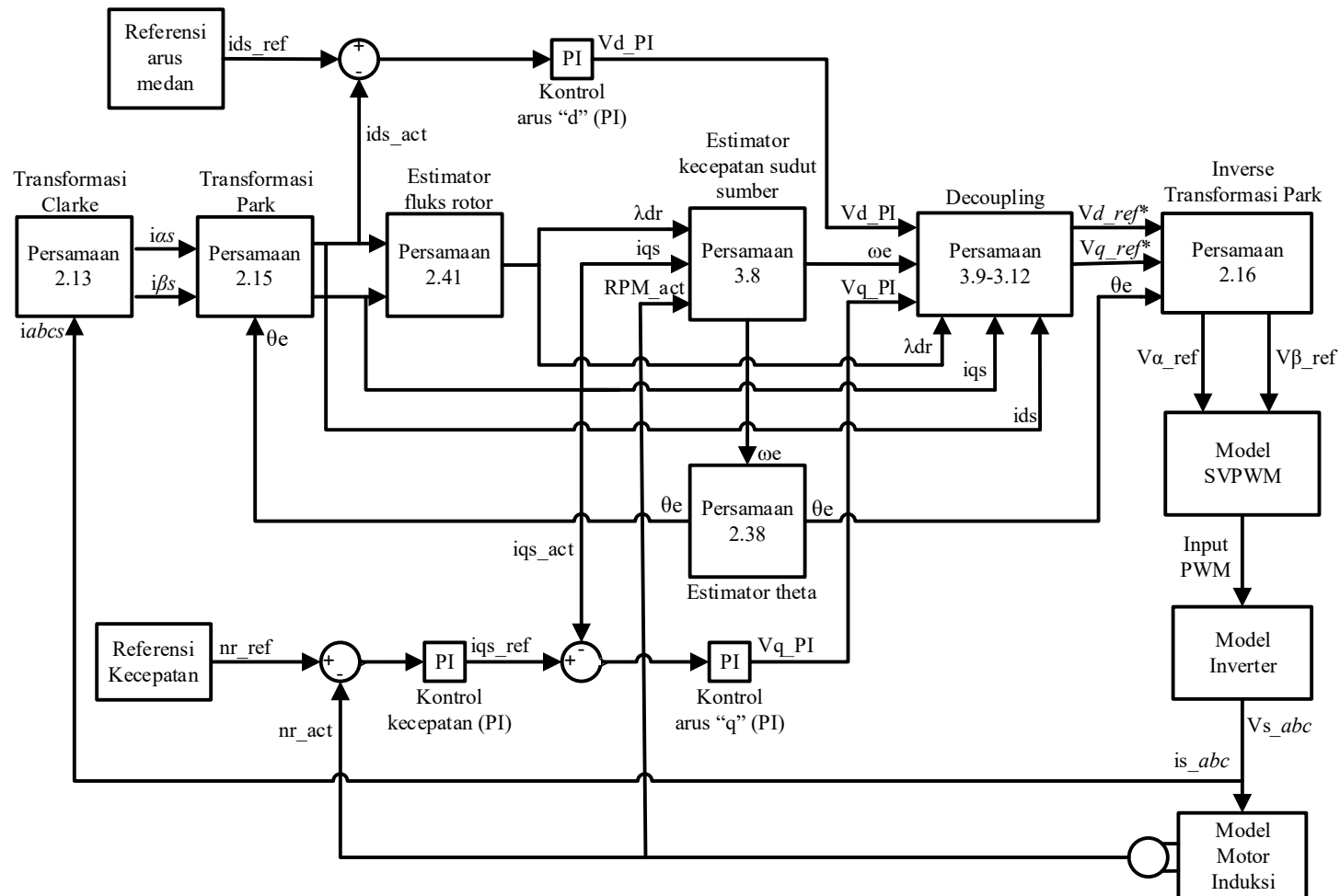
Sehingga didapatkan tegangan V_{qs} yang terkompensasi sebagai berikut:

$$V_{qsref} = V_{qsPI} + V_{qs kompensasi} \quad (3.12)$$

Dimana:

$V_{ds kompensasi}$	: Tegangan kompensasi <i>decoupling</i> sumbu d (V)
$V_{qs kompensasi}$	: Tegangan kompensasi <i>decoupling</i> sumbu q (V)
V_{dsPI}, V_{qsPI}	: Tegangan output dari kontroler PI arus (V)
V_{dsref}, V_{qsref}	: Tegangan referensi sumbu $d - q$ final (terkompensasi) (V)
i_{qs}	: Arus beban (A)
i_{ds}	: Arus medan (A)
ω_e	: Kecepatan sudut fluks (rad/s)
σ	: Koefisien kebocoran induktansi $\left(\sigma = L_s - \frac{L_m^2}{L_r}\right)$
L_s	: Induktansi stator (H)
L_r	: Induktansi rotor (H)
L_m	: Induktansi mutual (H)
λ_{dr}	: Fluks rotor sumbu d (Wb)

Berdasarkan pembahasan di atas, maka didapatkan blok diagram pemodelan sistem IFOC, ditunjukkan pada Gambar 3.8 berikut:



Gambar 3.8 blok diagram pemodelan sistem IFOC

Blok diagram pemodelan sistem IFOC yang ditunjukkan pada Gambar 3.8 menunjukan beberapa blok yang memiliki fungsi sebagai berikut:

1. Blok “Referensi Kecepatan” merupakan *setpoint* kecepatan yang diinginkan untuk motor induksi
2. Blok “Kontrol Kecepatan” berfungsi untuk membandingkan kecepatan referensi (nr_{ref}) dengan kecepatan aktual motor (nr_{act}). Berdasarkan error kecepatan, pengontrol PI ini menghasilkan referensi arus torsi (i_{qsref}) untuk mengendalikan torsi yang dibutuhkan motor
3. Blok “Kontrol Arus q (PI)” berfungsi untuk membandingkan referensi arus torsi (i_{qsref}) dengan arus torsi aktual (i_{qsact}) yang diukur. Outputnya adalah referensi tegangan sumbu q ($V_q - PI$)
4. Blok “Referensi Arus Medan” berfungsi untuk menghasilkan nilai referensi untuk arus medan (i_{dsref}) yang diperlukan untuk mempertahankan fluks rotor pada nilai yang diinginkan. Dalam operasi di bawah kecepatan nominal, nilai ini biasanya dijaga konstan untuk memaksimalkan fluks dan torsi. Blok ini mengimplementasikan prinsip yang ditunjukkan oleh Persamaan 2.43 di kondisi steady-state
5. Blok “Kontrol Arus d (PI)” berfungsi untuk membandingkan referensi arus medan (i_{dsref}) dengan arus medan aktual (i_{dsact}) yang diukur. Outputnya adalah referensi tegangan sumbu d ($V_d - PI$)
6. Blok “Transformasi Clarke” berfungsi untuk menerima arus fasa aktual dari motor (i_{as}, i_{bs}, i_{cs}) dan mengubah arus tiga fasa ini menjadi dua komponen arus ortogonal dalam kerangka acuan stasioner ($i_{\alpha s}$ dan $i_{\beta s}$). Transformasi ini didasarkan pada Persamaan 2.13

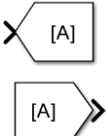
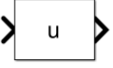
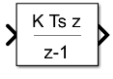
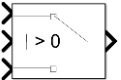
7. Blok “Transformasi Park” berfungsi untuk menerima komponen arus $i_{\alpha s}$ dan $i_{\beta s}$ dari Transformasi Clarke, kemudian mengubahnya menjadi komponen arus dalam kerangka acuan berputar sinkron (i_{ds} dan i_{qs}). Transformasi ini memerlukan informasi posisi sudut fluks rotor (θ_e) untuk menyelaraskan sumbu d dengan vektor fluks rotor. Transformasi ini didasarkan pada Persamaan 2.15
8. Blok “Estimator Fluks Rotor” berfungsi untuk menghitung estimasi fluks rotor sumbu d (λ_{dr}) berdasarkan arus medan stator i_{ds} . Implementasi blok ini didasarkan pada Persamaan 2.41, yang merepresentasikan dinamika fluks rotor setelah kondisi decoupling fluks rotor sumbu q ($\lambda_{qr} = 0$) terpenuhi
9. Blok “Estimator Sudut Fluks Rotor” berfungsi untuk menghitung nilai kecepatan sinkron (ω_e) dan kemudian mengintegrasikannya untuk mendapatkan nilai posisi sudut fluks rotor (θ_e). Perhitungan ini didasarkan pada Persamaan 3.8 dan Persamaan 2.38. Nilai sudut θ_e ini sangat krusial untuk proses Transformasi Park dan Inverse Transformasi Park, memastikan bahwa sumbu d selalu selaras dengan fluks medan magnet yang sebenarnya
10. Blok “Decoupling” berfungsi untuk membatalkan efek kopling silang yang melekat pada persamaan tegangan stator di kerangka acuan d-q. Blok ini menerima input berupa fluks rotor (λ_{dr}), kecepatan sinkron (ω_e), serta komponen arus i_{ds} dan i_{qs} . Dengan mengimplementasikan Persamaan 3.9 sampai Persamaan 3.12, blok ini menghitung komponen tegangan kompensasi dan menambah/mengurangnya dari output PI controller arus ($V_d - PI$ dan $V_q - PI$) untuk menghasilkan tegangan V_{dref} dan V_{qref} yang telah terdecoupling

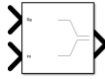
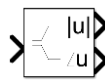
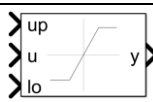
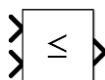
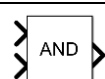
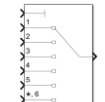
11. Blok “Inverse Transformasi Park” berfungsi untuk menerima referensi tegangan d-q yang telah ter-*decoupling* ($V_{d_{ref}}$ dan $V_{q_{ref}}$) dari Blok “Decoupling”, blok ini mengubahnya kembali ke komponen tegangan dalam kerangka acuan stasioner ($V_{\alpha_{ref}}$ dan $V_{\beta_{ref}}$). Transformasi ini menggunakan informasi sudut θ_e dari estimator fluks dan didasarkan pada Persamaan 2.16. Tegangan inilah yang kemudian menjadi input untuk blok SVPWM
12. Blok “Model SVPWM” berfungsi untuk menghasilkan sinyal Pulse Width Modulation yang kompleks dan efisien untuk mengendalikan sakelar-sakelar pada inverter tiga fasa, berdasarkan tegangan referensi $V_{\alpha_{ref}}$ dan $V_{\beta_{ref}}$
13. Blok “Model Inverter” merupakan komponen elektronika daya yang mengubah tegangan DC menjadi tegangan AC tiga fasa variabel frekuensi dan amplitudo yang diperlukan untuk mengendalikan motor, dikendalikan oleh sinyal PWM dari blok SVPWM
14. Blok “Model Motor Induksi” merupakan objek kontrol utama, yaitu motor induksi tiga fasa. Motor menerima tegangan AC dari inverter dan memberikan umpan balik berupa arus tiga fasa (i_{as}, i_{bs}, i_{cs}) dan kecepatan rotor ($n_{r_{act}}$) kepada sistem kontrol. Torsi elektromagnetik (T_e) juga merupakan output kunci dari blok motor ini

3.2.6 Penerapan rancangan model pada MATLAB/Simulink

Setelah semua langkah – langkah di atas dilakukan, maka pemodelan dapat diterapkan langsung pada MATLAB/Simulink. Pada Tabel 3.1 di bawah ini dijelaskan berbagai komponen yang digunakan pada tahap penerapan ini.

Tabel 3.1 Komponen-komponen Simulink/Matlab

No.	Komponen	Gambar	Fungsi
1.	<i>Mux</i>		Menggabungkan beberapa sinyal input menjadi satu sinyal output berbentuk vektor.
2.	<i>Scope</i>		Menampilkan sinyal atau data dalam bentuk grafik selama simulasi berlangsung, mirip dengan fungsi osiloskop.
3.	<i>Constant</i>		Menghasilkan nilai konstanta yang tidak berubah selama simulasi.
4.	<i>Go to dan From</i>		Digunakan untuk menghubungkan sinyal antara blok-blok tanpa perlu membuat garis penghubung yang panjang, sehingga model menjadi lebih rapi.
5.	<i>PI</i>		Merepresentasikan blok kontroler Proporsional-Integral (PI), yang digunakan untuk mengontrol suatu sistem.
6.	<i>Sum</i>		Melakukan operasi penjumlahan atau pengurangan pada sinyal input.
7.	<i>Fcn</i>		Menerapkan fungsi matematika yang didefinisikan pengguna pada sinyal input.
8.	<i>Gain</i>		Mengalikan sinyal input dengan nilai konstanta yang ditentukan (gain).
9.	<i>Inport-Output</i>		Digunakan untuk menentukan input dan output dari sebuah subsistem atau model utama.
10.	<i>Subsystem</i>		Mengelompokkan beberapa blok ke dalam satu blok tunggal untuk menyederhanakan dan mengatur model.
11.	<i>Product</i>		Melakukan operasi perkalian pada sinyal input.
12.	<i>Unit delay</i>		Menunda sinyal input sebanyak satu periode sampel. Ini berfungsi sebagai operator diskrit z^{-1} .
13.	<i>Discrete-Time Integrator</i>		Melakukan operasi integrasi sinyal secara diskrit.
14.	<i>Add</i>		Melakukan operasi penjumlahan pada sinyal input.
15.	<i>Switch</i>		Meneruskan salah satu dari dua sinyal input ke output berdasarkan nilai sinyal kontrol.
16.	<i>Divide</i>		Melakukan operasi pembagian pada sinyal input.

17.	<i>Real image to complex</i>		Menggabungkan komponen sinyal real dan imajiner untuk membentuk satu sinyal kompleks.
18.	<i>Complex to Magnitude-Angle</i>		Mengubah sinyal kompleks menjadi nilai magnitudo dan sudutnya.
19.	<i>Saturation Dynamic</i>		Membatasi sinyal input agar tidak melebihi batas atas atau bawah yang ditentukan.
20.	<i>Radians to Degrees</i>		Mengonversi nilai dari satuan radian ke derajat.
21.	<i>Relational Operator</i>		Membandingkan dua sinyal input dan menghasilkan output boolean (true atau false) berdasarkan perbandingan tersebut.
22.	<i>Logical Operator</i>		Melakukan operasi logika seperti AND, OR, atau NOT pada sinyal input.
23.	<i>Multiport Switch</i>		Meneruskan salah satu dari beberapa sinyal input ke output berdasarkan nilai dari sinyal kontrol.

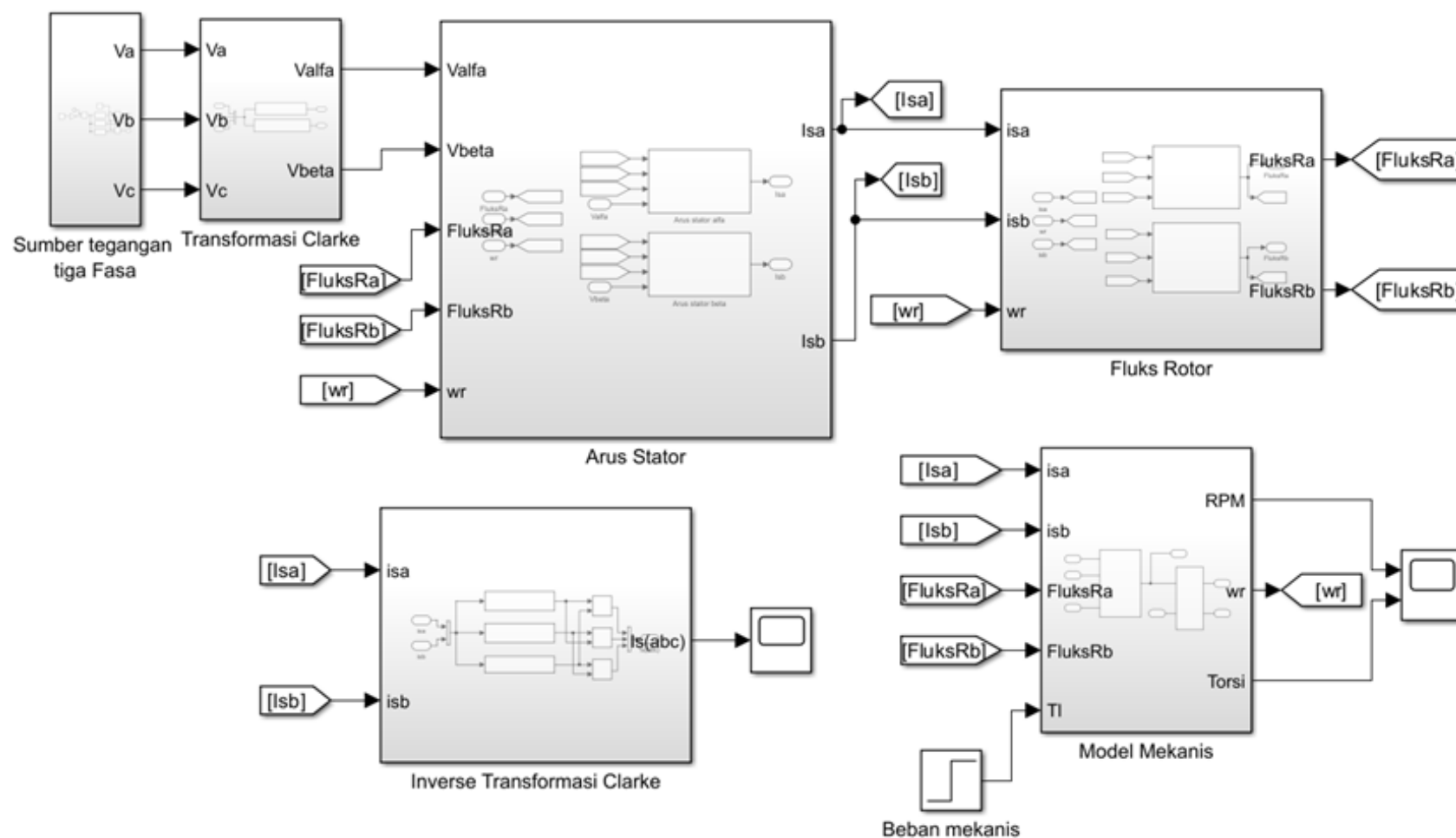
3.3 Pemodelan Unit

Pada tahap ini, dilakukan penerapan model setiap unit yang dijelaskan pada

Sub-Bab 3.2 Perancangan Sistem menggunakan Matlab/Simulink

3.3.1 Model motor induksi

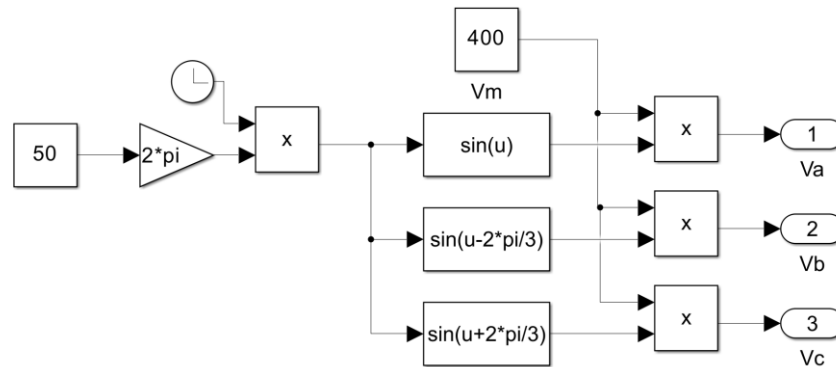
Pemodelan motor induksi dibuat sesuai dengan blok diagram pemodelan motor induksi yang ditunjukkan pada Gambar 3.6. Penerapan model unit pada Matlab/Simulink ditunjukkan pada Gambar 3.9 berikut:



Gambar 3.9 Model unit motor induksi

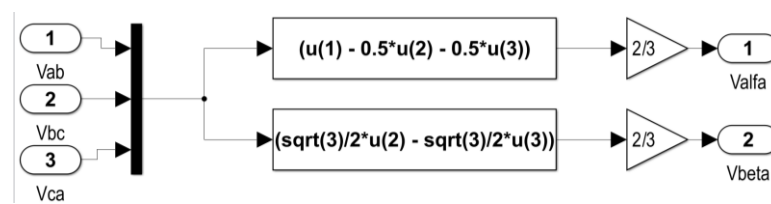
Model unit motor induksi yang ditunjukkan pada Gambar 3.9 disusun dengan beberapa sub-sistem sebagai berikut:

1. Sub-sistem "Sumber tegangan tiga fasa"



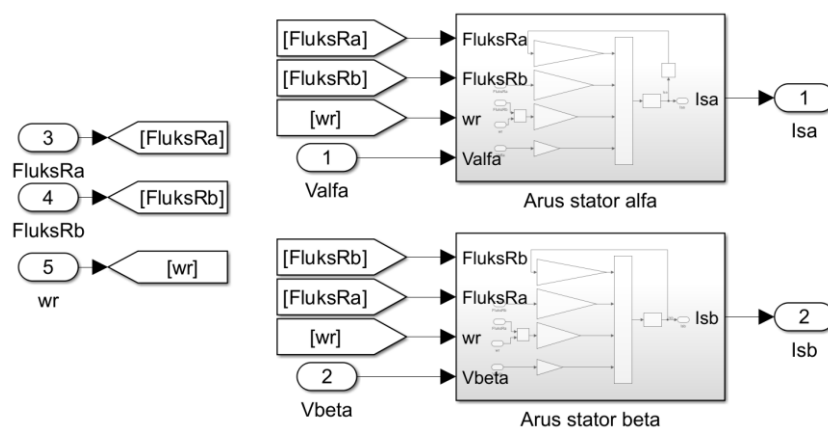
Gambar 3.10 Sub-sistem "Sumber tegangan tiga fasa"

2. Sub-sistem "Transformasi Clarke"

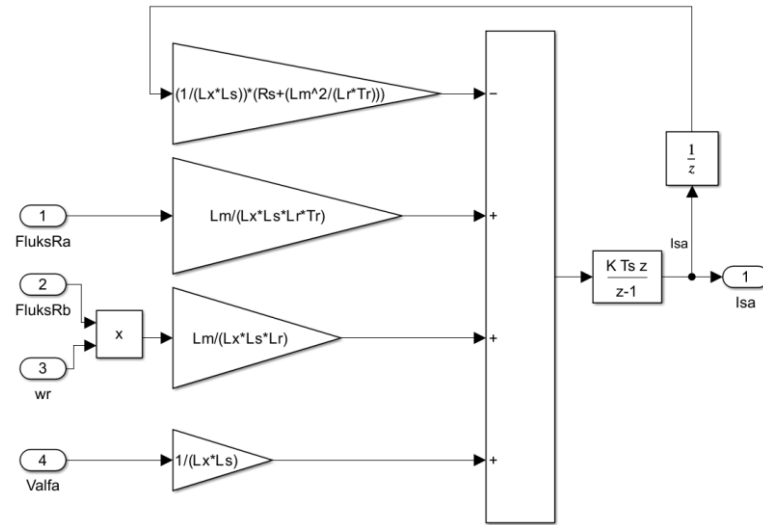


Gambar 3.11 Sub-sistem "Transformasi Clarke"

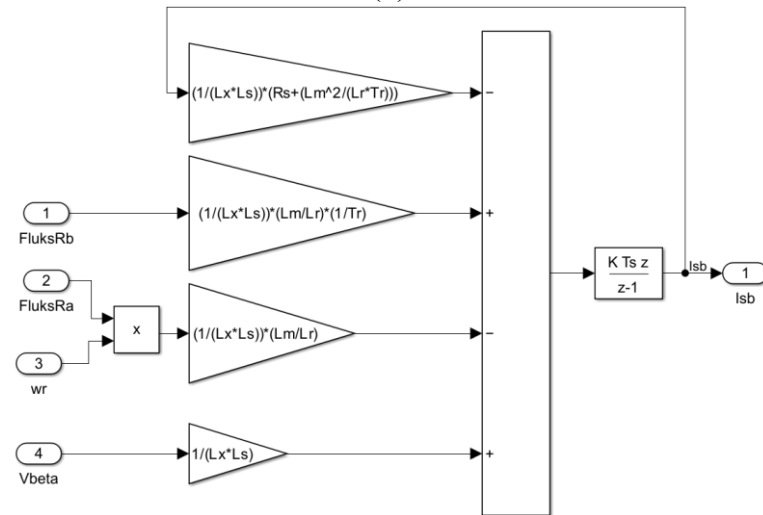
3. Sub-sistem "Arus stator"



(a)



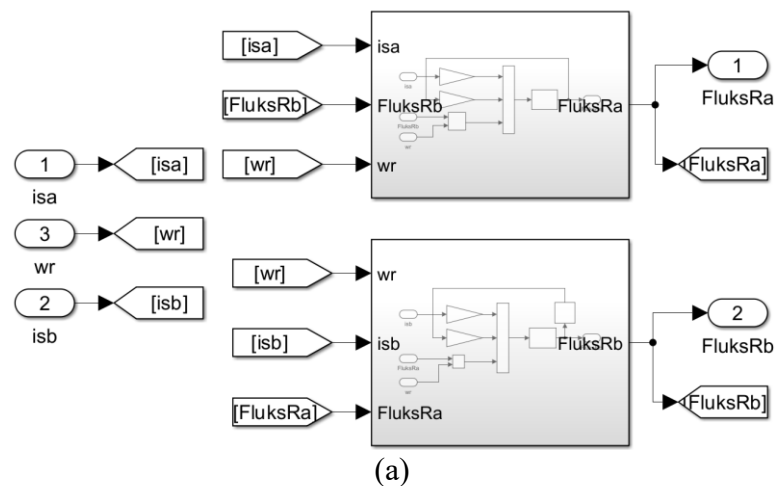
(b)



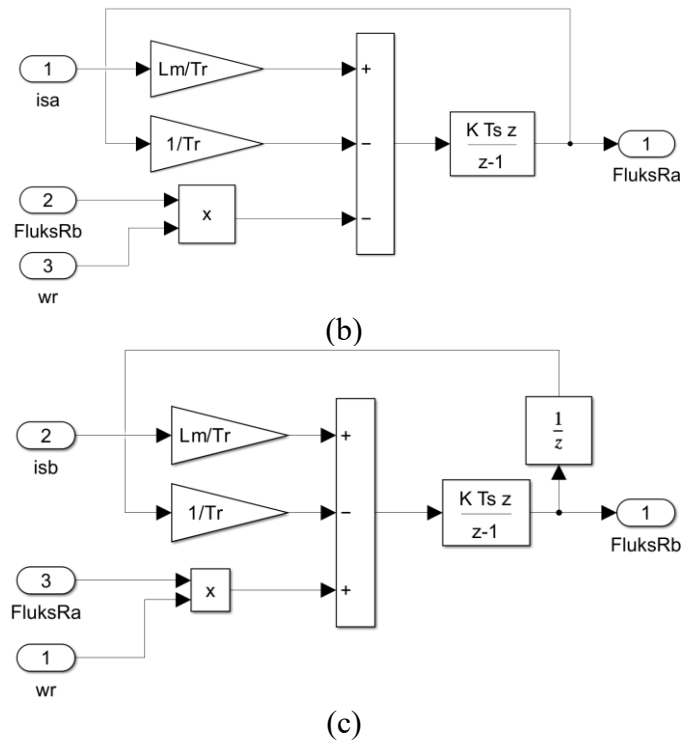
(c)

Gambar 3.12 (a) Sub-sistem "Arus stator" (b) Sub-sistem perhitungan arus stator sumbu alfa ($i_{s\alpha}$). (b) Sub-sistem perhitungan arus stator sumbu beta ($i_{s\beta}$)

4. Sub-sistem "Fluks rotor"

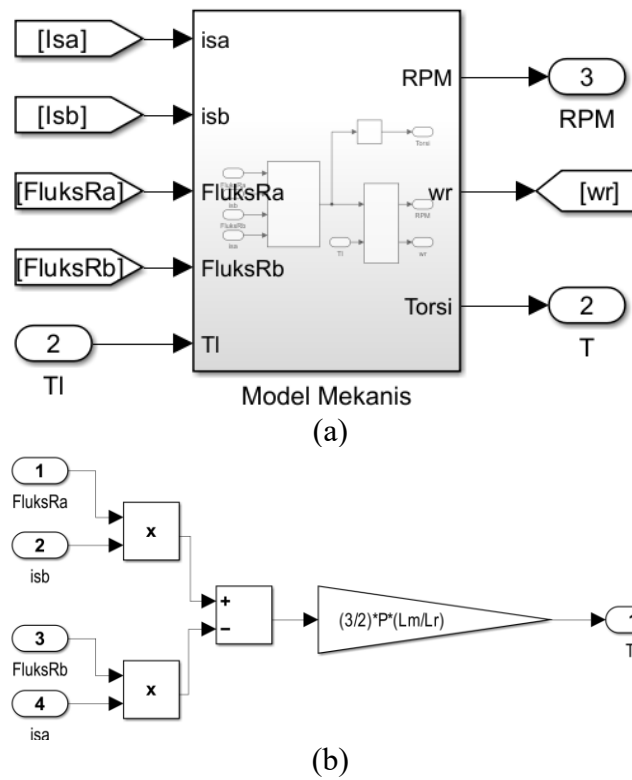


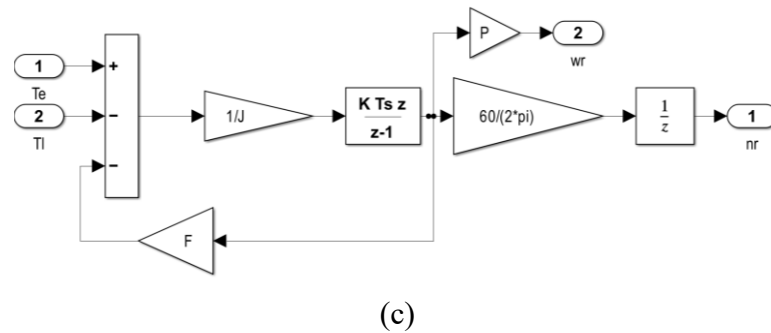
(a)



Gambar 3.13 (a) Sub-sistem “Fluks rotor” (b) Sub-sistem perhitungan fluks rotor sumbu alfa ($\lambda_{r\alpha}$). (c) Sub-sistem perhitungan fluks rotor sumbu beta ($\lambda_{r\beta}$)

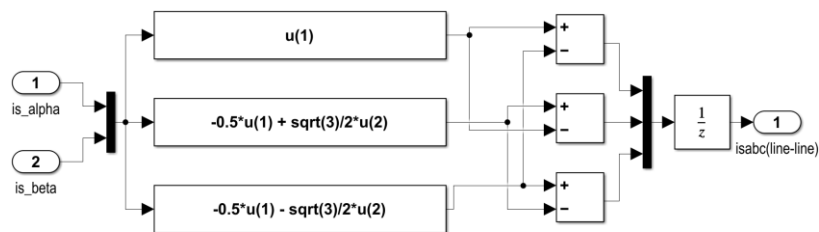
5. Sub-sistem "Model Mekanis"





Gambar 3.14 (a) Sub-sistem “Model mekanis” (b) Sub-sistem perhitungan torsi (T_e). (c) Sub-sistem perhitungan kecepatan putaran rotor per menit (n_r) dan kecepatan sudut rotor (ω_r)

6. Sub-sistem "Inverse Transformasi Clarke"

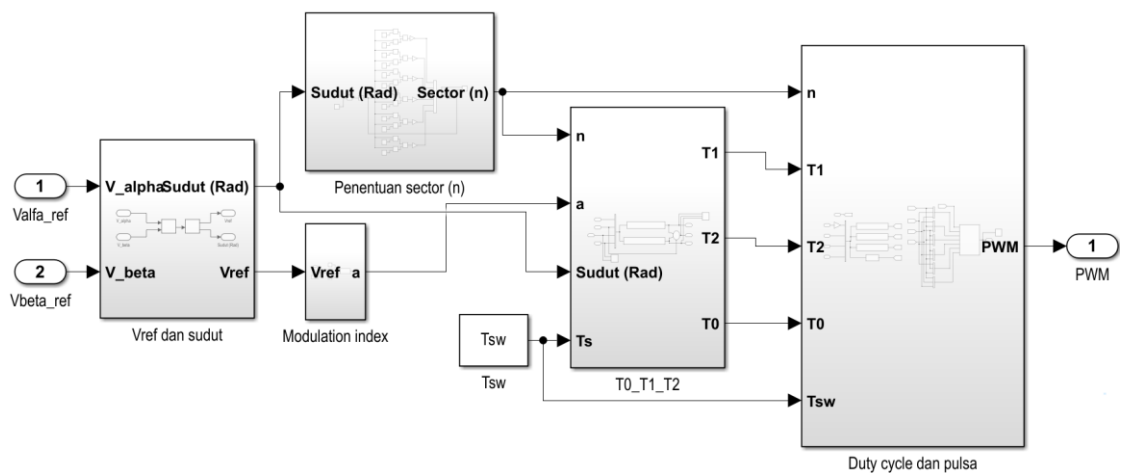


Gambar 3.15 Sub-sistem "Inverse Transformasi Clarke"

3.3.2 Model Space Vector Pulse Width Modulation (SVPWM)

1. Pemodelan SVPWM

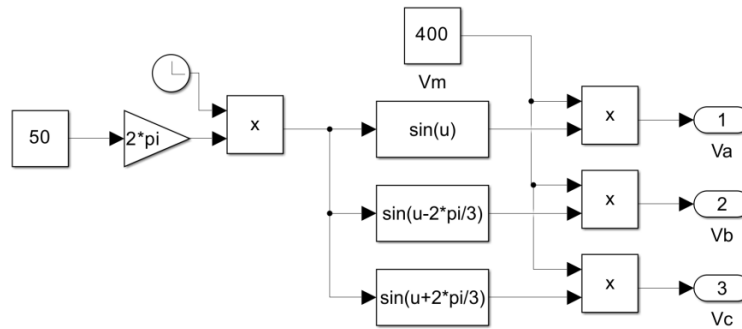
Pemodelan SVPWM dibuat sesuai dengan blok diagram pemodelan SVPWM yang ditunjukkan pada Gambar 3.7. Penerapan model unit pada Matlab/Simulink ditunjukkan pada Gambar 3.16 berikut:



Gambar 3.16 Model unit SVPWM

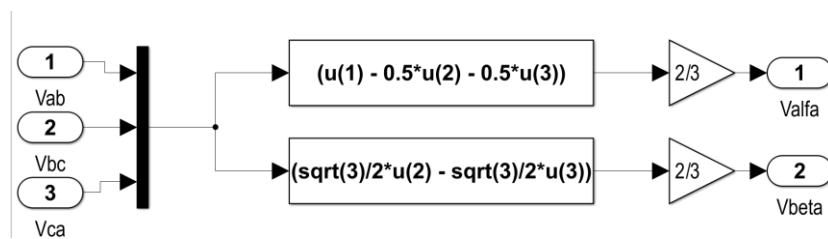
Model unit SVPWM yang ditunjukkan pada Gambar 3.15 disusun dengan beberapa sub-sistem sebagai berikut:

- a. Sub-sistem "Sumber tegangan tiga fasa"



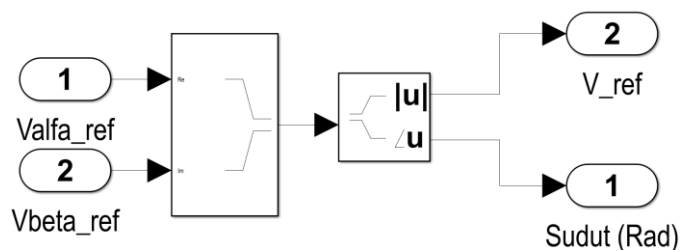
Gambar 3.17 Sub-sistem "Sumber tegangan tiga fasa"

- b. Sub-sistem "Transformasi Clarke"



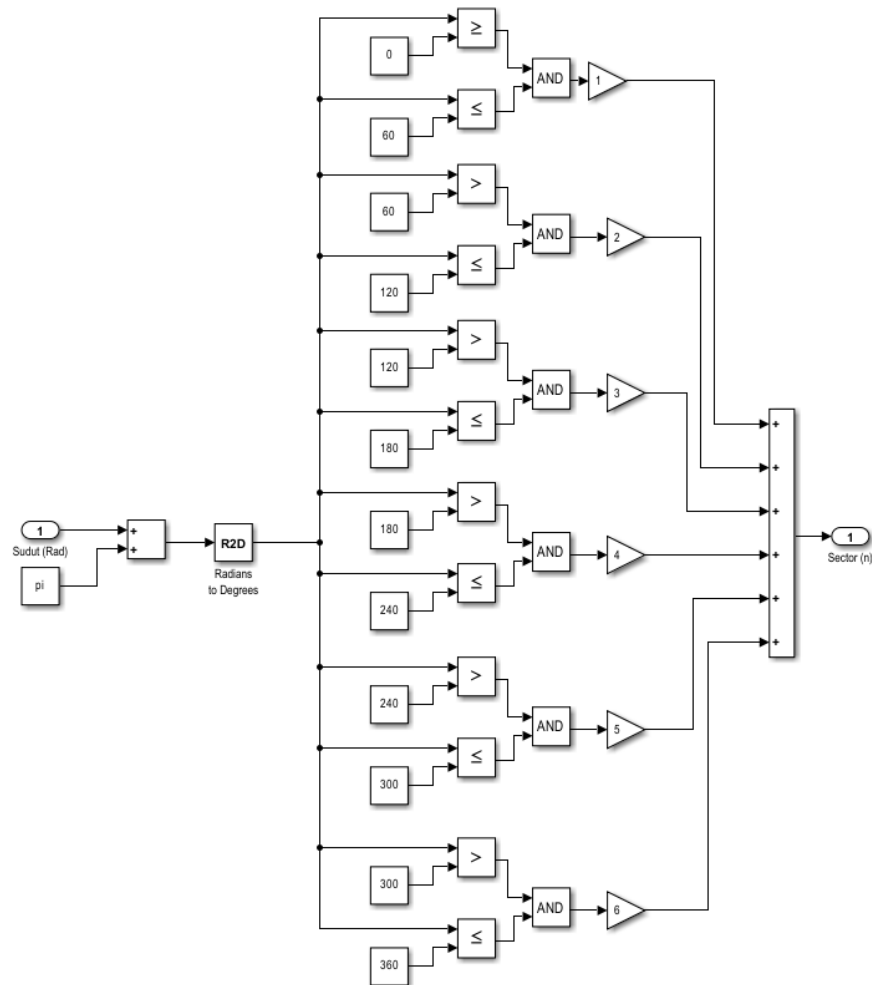
Gambar 3.18 Sub-sistem "Transformasi Clarke"

- c. Sub-sistem " V_{ref} dan sudut (θ)"



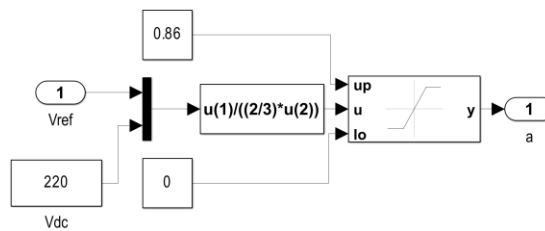
Gambar 3.19 Sub-sistem " V_{ref} dan sudut (θ)"

- d. Sub-sistem "Penentuan sektor"



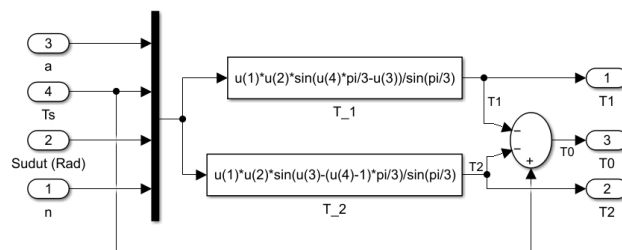
Gambar 3.20 Sub-sistem “Penentuan sektor”

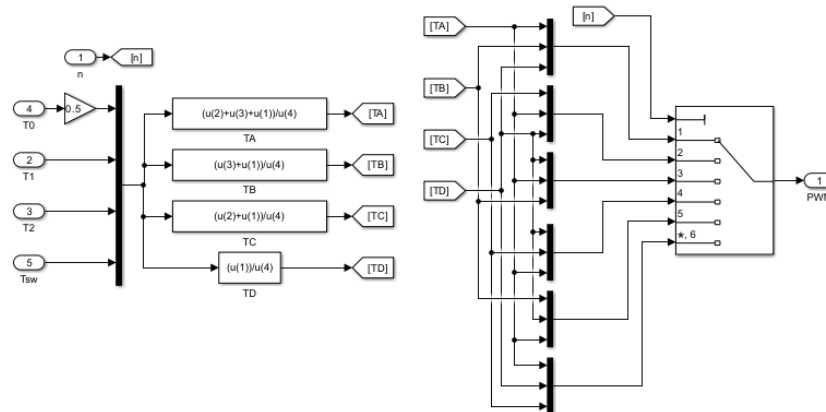
e. Sub-sistem “Indeks modulasi”



Gambar 3.21 Sub-sistem “Indeks modulasi”

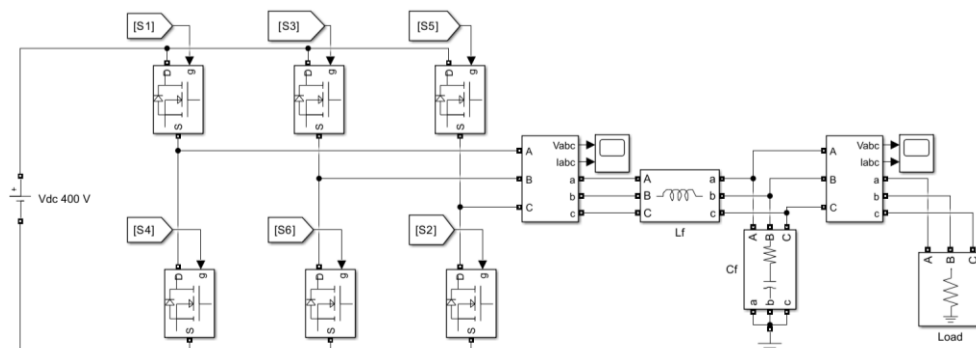
f. Sub-sistem “Perhitungan T_1 , T_2 , dan T_0 ”

Gambar 3.22 Sub-sistem “Perhitungan T_1 , T_2 , dan T_0 ”

g. Sub-sistem “*Duty cycle* dan pulsa”Gambar 3.23 Sub-sistem “*Duty cycle* dan pulsa”

2. Model Inverter

Setelah model SVPWM menghasilkan enam sinyal pulsa kendali, sinyal tersebut digunakan untuk menggerakkan model inverter tiga fasa. Model ini dirancang untuk merepresentasikan rangkaian elektronika daya yang sesungguhnya. Keluaran dari model inverter dihubungkan ke model filter LC. Filter ini dimodelkan menggunakan komponen pasif standar dari *library* Simulink. Model inverter yang dibuat menggunakan sumber tegangan DC sebesar 400 V dengan nilai komponen filter yang digunakan pada pemodelan ini adalah $L_f = 1\text{ mH}$, $C_f = 20\text{ }\mu\text{F}$, dan sebuah resistor redaman $R_d = 1\text{ }\Omega$ untuk mencegah osilasi. Gabungan dari model inverter dan filter LC ditunjukkan pada Gambar 3.24 di bawah ini:



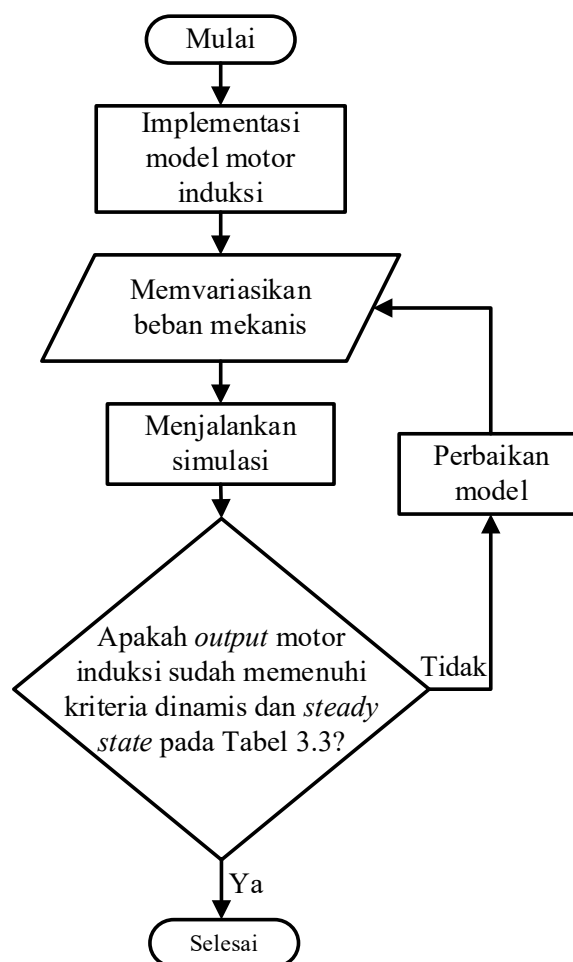
Gambar 3.24 Model inverter dan filter LC

3.4 Pengujian Model Unit

Pengujian unit dilakukan untuk memvalidasi apakah unit yang dimodelkan sesuai dengan spesifikasi yang diinginkan.

3.4.1 Pengujian Model Motor induksi

Pengujian model motor induksi bertujuan untuk memvalidasi model yang sudah dibuat mampu merepresentasikan unjuk kerja dinamis motor induksi. Diagram alir pengujian model motor induksi ditunjukkan pada Gambar 3.25 berikut:



Gambar 3.25 Diagram alir pengujian model motor induksi

Berdasarkan diagram alir pengujian model motor induksi pada Gambar 3.25 di atas, pada proses pengujian model motor induksi dilakukan variasi beban

mekanis yang bertujuan untuk mengetahui unjuk kerja dinamis model pada kondisi beban yang berbeda. Variasi beban mekanis yang akan dilakukan pada proses pengujian ditunjukkan Tabel 3.2 berikut:

Tabel 3.2 Variasi beban mekanis pengujian model motor induksi

No	Kondisi	Nilai
1	Tanpa beban	$T_L = 0 \text{ N.m}$
2	Beban konstan	$T_L = 2 \text{ N.m}$

Setelah dilakukan proses pengujian dengan kondisi beban mekanis yang berbeda. Dilakukan proses validasi terhadap arus stator (i_s), torsi (T_e) dan kecepatan rotor motor induksi (n_r). Model dianggap berhasil ketika memenuhi kriteria pada Tabel 3.3 berikut:

Tabel 3.3 Kriteria validasi pengujian model motor induksi

Parameter	Kriteria validasi kondisi transien	Kriteria validasi kondisi <i>steady-state</i>
Arus Stator (i_s)	Terjadi lonjakan arus pada saat <i>starting</i>	Harus stabil pada kondisi setelah <i>starting</i> dan beban konstan
Torsi Elektromagnetik (T_e)	Wajib menunjukkan lonjakan torsi (<i>overshoot</i>) awal untuk mengatasi inersia	Harus stabil pada nilai hampir nol (untuk pengujian tanpa beban) atau setara dengan beban (T_L) (untuk pengujian dengan beban)
Kecepatan Rotor (n_r)	Kecepatan meningkat dari nol menuju kecepatan tunak dengan akselerasi yang wajar	Kecepatan harus stabil pada nilai yang lebih rendah dari kecepatan sinkron (n_s), yang membuktikan adanya slip ($s > 0$)

3.4.2 Pengujian SVPWM

Pengujian model SVPWM (*Space Vector Pulse Width Modulation*) bertujuan untuk memvalidasi algoritma modulasi yang digunakan dalam mengendalikan inverter tiga fasa. Validasi ini penting untuk memastikan model menghasilkan tegangan output dengan kualitas daya terbaik dan mampu merespons perintah kontrol secara dinamis.

Pengujian model SVPWM dilakukan dalam dua kondisi utama untuk memverifikasi akurasi dan kinerja modulasi:

1. Validasi kondisi statis: bertujuan memvalidasi kualitas daya keluaran inverter saat diberikan perintah tegangan referensi yang konstan. Sesuai dengan perancangan yang menyertakan filter LC, kriteria validasi didasarkan pada dua parameter utama. Pengujian dilakukan secara komparatif dengan metode PWM konvensional (SPWM) pada kondisi Indeks Modulasi (M) dan frekuensi konstan. Kriteria kunci validasi statis didasarkan pada dua parameter utama:

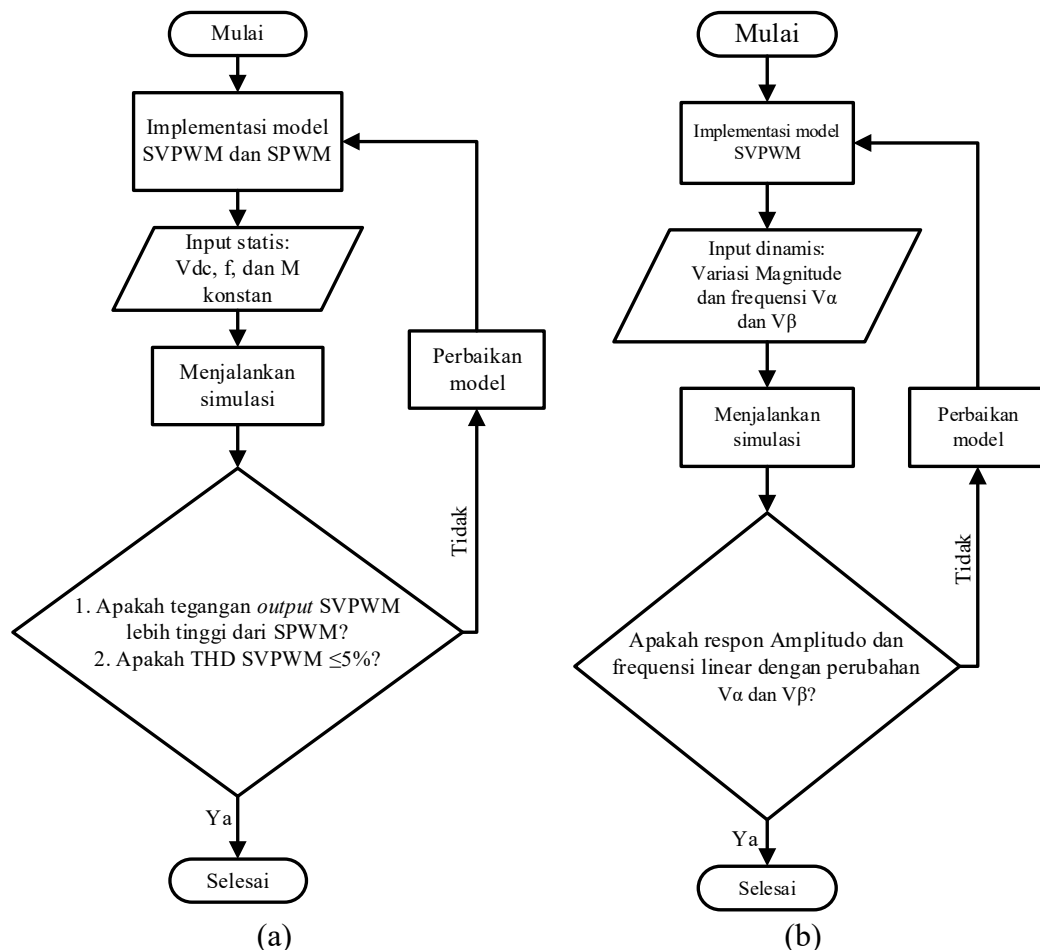
- Efisiensi pemanfaatan tegangan (*voltage utilization*). SVPWM harus menghasilkan tegangan *output* fasa-ke-fasa maksimum sekitar 15% lebih tinggi daripada SPWM, membuktikan efisiensi pemanfaatan tegangan DC
- Kualitas gelombang diukur melalui *Total Harmonic Distortion* (THD). Mengukur tingkat distorsi harmonisa pada tegangan keluaran, baik sebelum maupun sesudah melewati filter LC. Nilai THD *output* SVPWM harus minimal (idealnya $\leq 5\%$ setelah filter), yang menunjukkan keberhasilan dalam menghilangkan harmonik orde rendah

2. Kondisi dinamis: bertujuan memvalidasi respons model terhadap perintah kontrol yang berubah, mereplika sinyal dari *controller* FOC. Pengujian dilakukan dengan memberikan perubahan langkah (*step change*) pada sinyal tegangan referensi V_α dan V_β (magnitudo dan sudut). Model dianggap berhasil jika:

- Perubahan amplitudo tegangan *output* merespons perubahan magnitudo vektor V_{ref} secara cepat, stabil, dan linier.

- Perubahan frekuensi *output* merespons perubahan kecepatan putar vektor secara akurat dan tanpa *overshoot* yang signifikan.

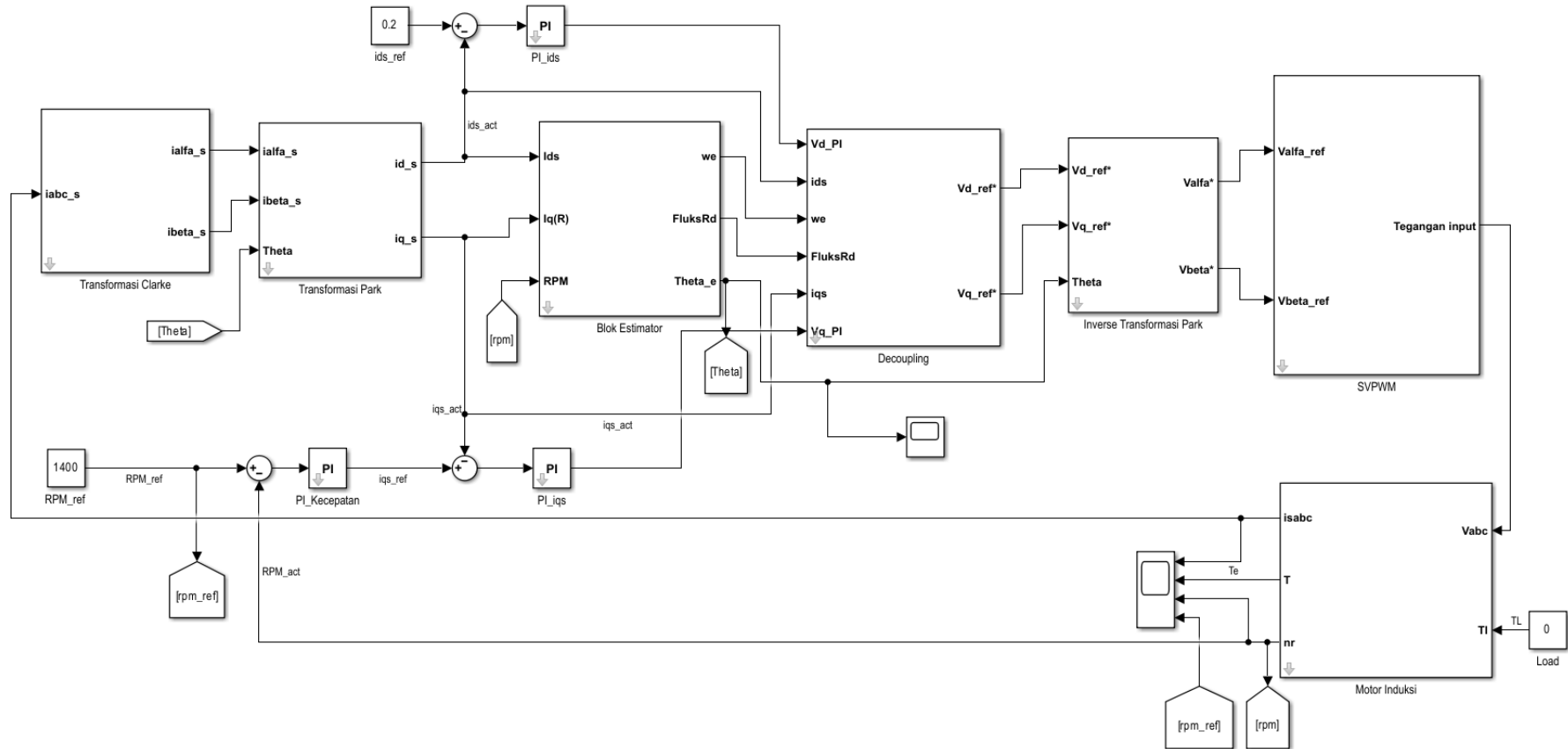
Diagram alir pengujian model SVPWM yang mencakup kedua validasi kondisi tersebut, ditunjukkan pada Gambar 3.26 berikut:



Gambar 3.26 Diagram alir pengujian model SVPWM (a) pengujian statis, (b) pengujian dinamis

3.5 Pemodelan Sistem *Indirect Field Oriented Control* (IFOC)

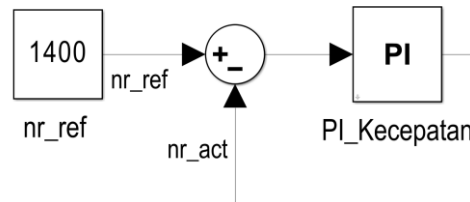
Pemodelan sistem *Indirect Field Oriented Control* (IFOC) dibuat sesuai dengan blok diagram pemodelan sistem *Indirect Field Oriented Control* (IFOC) yang ditunjukkan pada Gambar 3.8. Penerapan pemodelan pada Matlab/Simulink ditunjukkan pada Gambar 3.27 berikut:



Gambar 3.27 Indirect Field Oriented Control (IFOC)

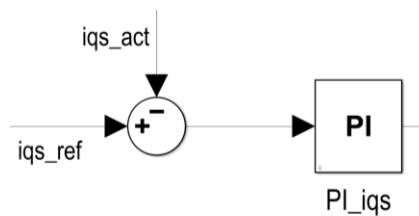
Model *Indirect Field Oriented Control* (IFOC) yang ditunjukkan pada Gambar 3.27 disusun dengan beberapa blok sub-sistem sebagai berikut:

1. Blok “Referensi Kecepatan dan blok “Kontrol Kecepatan (PI)”



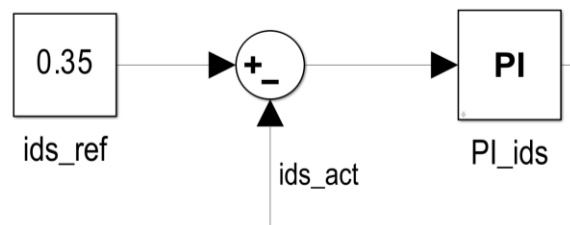
Gambar 3.28 Blok “Referensi Kecepatan dan blok “Kontrol Kecepatan (PI)”

2. Blok “Kontrol Arus q (PI)”



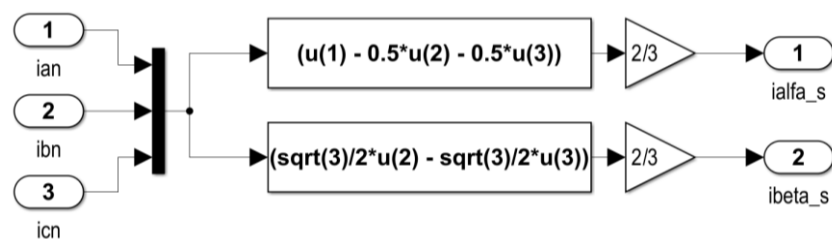
Gambar 3.29 Blok “Kontrol Arus beban i_{qs} (PI)”

3. Blok “Referensi Arus Medan” dan blok “Kontrol Arus d (PI)”



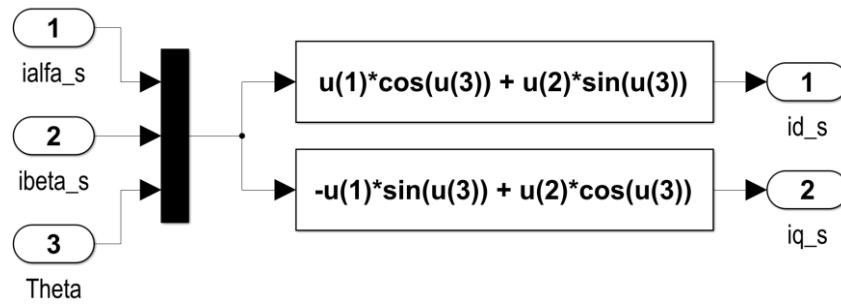
Gambar 3.30 Blok “Referensi Arus Medan” dan blok “Kontrol Arus d (PI)”

4. Sub-Sistem “Transformasi Clarke”



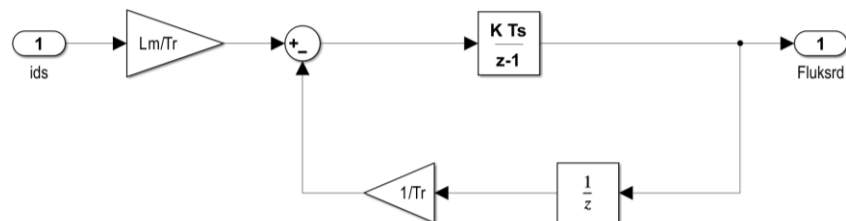
Gambar 3.31 Sub-Sistem “Transformasi Clarke”

5. Sub-Sistem “Transformasi Park”



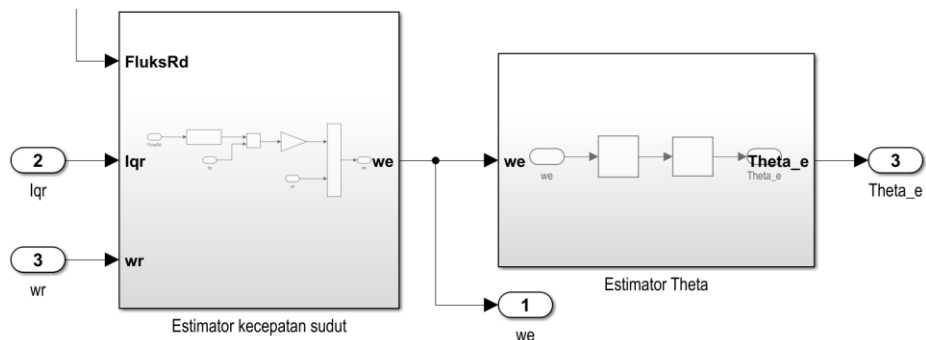
Gambar 3.32 Sub-Sistem “Transformasi Park”

6. Sub-Sistem “Estimator Fluks Rotor”

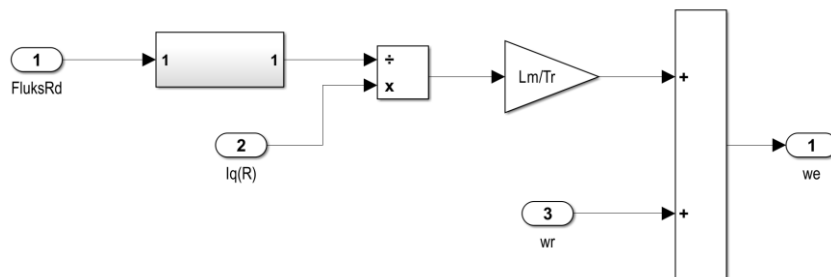


Gambar 3.33 Sub-Sistem “Estimator Fluks Rotor”

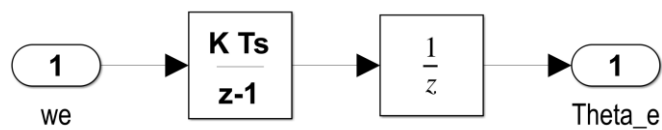
7. Sub-Sistem “Estimator Sudut Fluks Rotor”



(a)

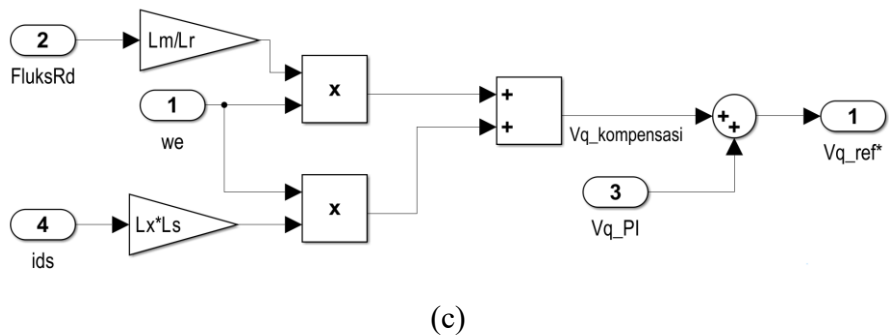
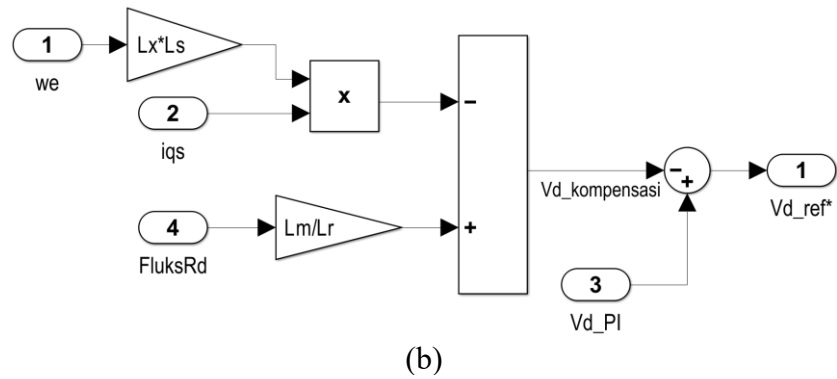
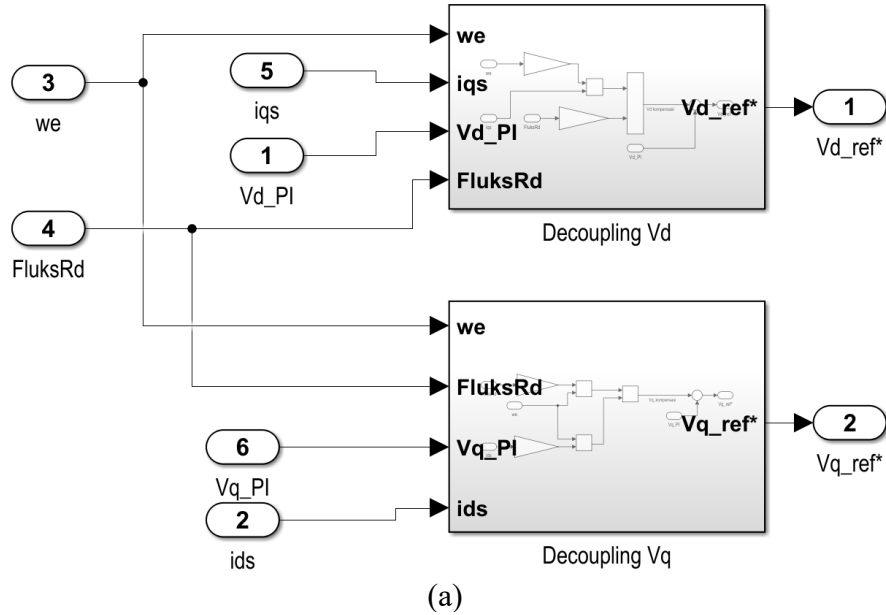


(b)



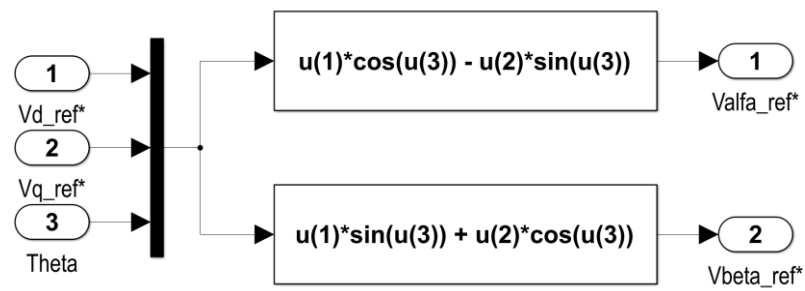
(c)

Gambar 3.34 (a) Sub-Sistem “Estimator Sudut Fluks Rotor” (b) Sub-Sistem “Estimator kecepatan sinkron (ω_e)” (c) Sub-Sistem “Estimator theta (θ_e)”
8. Sub-Sistem “Decoupling”



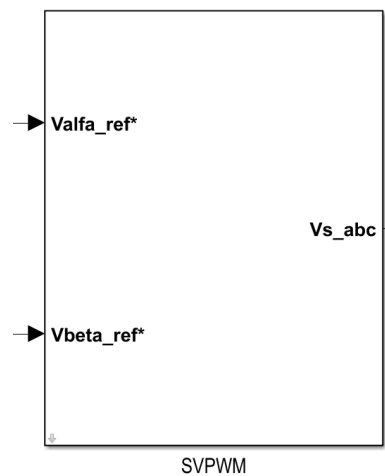
Gambar 3.35 (a) Sub-Sistem “Decoupling” (b) Sub-Sistem “Decoupling V_d ” (c) Sub-Sistem “Decoupling V_q ”

9. Sub-Sistem “Inverse Transformasi Park”



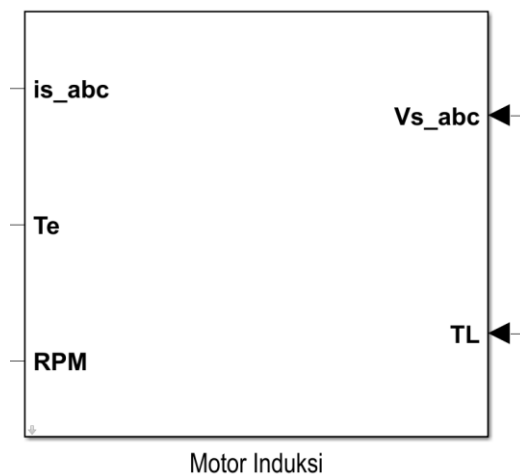
Gambar 3.36 Sub-Sistem “Inverse Transformasi Park”

10. Blok “Model SVPWM”



Gambar 3.37 Blok “Model SVPWM”

11. Blok “Model Motor Induksi”



Gambar 3.38 Blok “Model Motor Induksi”

3.6 Pengujian Model Sistem *Indirect Field Oriented Control (IFOC)*

Setelah unit-unit dimodelkan dan diuji secara terpisah, pada tahap ini model unit telah diintegrasikan membentuk satu model sistem IFOC yang utuh di

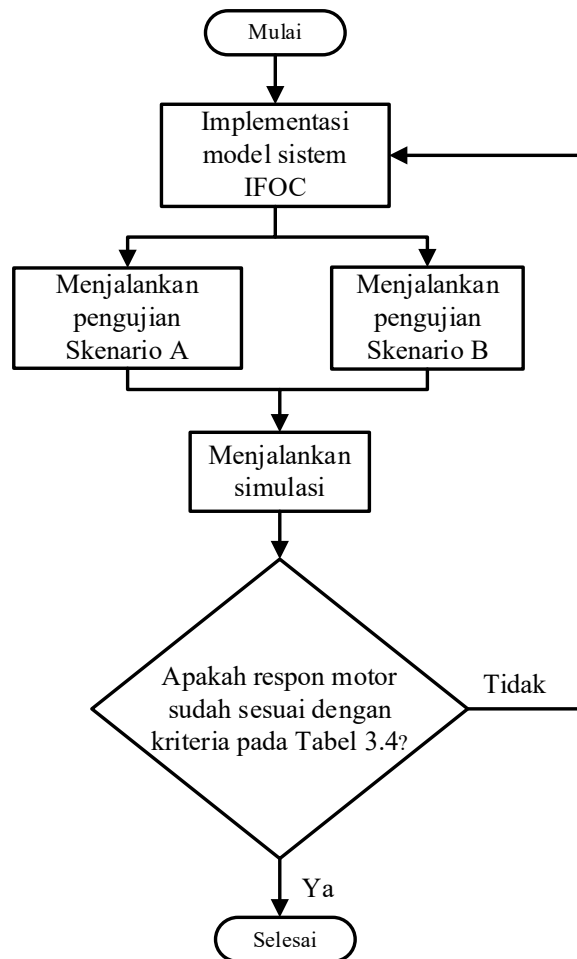
Simulink. Kemudian pada sub-bab ini dilakukan pengujian untuk memvalidasi model sistem secara keseluruhan sebelum diimplementasikan ke perangkat keras.

Validasi ini berfokus pada pembuktian bahwa model sistem IFOC yang dibangun berjalan dengan benar dan mampu mengendalikan model motor induksi sesuai dengan landasan teori pada sub-bab 2.4. Model sistem dianggap tervalidasi jika unjuk kerjanya memenuhi kriteria-kriteria kunci yang dirangkum pada Tabel 3.4 berikut:

Tabel 3.4 Kriteria validasi sistem IFOC

No	Parameter	Kriteria validasi sistem IFOC
1	Kecepatan aktual (nr_{act})	1. Kecepatan aktual (nr_{act}) mampu mencapai kecepatan referensi dengan <i>steady state error</i> yang minimal terlepas dari perubahan beban mekanis (T_L) 2. Kecepatan aktual (nr_{act}) tetap stabil terlepas dari perubahan arus medan referensi (i_{dsref}) dan beban mekanis (T_L)
2	Arus medan aktual (i_{dsact})	1. Arus medan aktual (i_{dsact}) mampu mencapai arus medan referensi 2. Arus medan aktual (nr_{act}) tetap stabil terlepas dari perubahan kecepatan dan beban mekanis (T_L)
3	Arus beban aktual (i_{qsact}) dan torsi (T_e)	1. Pada kondisi tanpa beban ($T_L = 0$), arus beban aktual (i_{qsact}) dan torsi (T_e) mendekati nol 2. Pada kondisi dengan beban ($T_L > 0$), arus beban aktual (i_{qsact}) harus meningkat secara proporsional untuk menghasilkan torsi (T_e) yang sesuai dengan kebutuhan beban mekanis

Untuk melakukan validasi model sistem IFOC sesuai dengan kriteria pada Tabel 3.4, pengujian model sistem akan dibagi menjadi dua skenario utama, diilustrasikan pada diagram alir pengujian sistem IFOC pada Gambar 3.39 berikut:



Gambar 3.39 Diagram alir pengujian model sistem IFOC (simulasi)

Skenario pengujian (A dan B) pada diagram alir pengujian model sistem IFOC (Gambar 3.39) adalah sebagai berikut:

1. Skenario A: Respons pelacakan kecepatan

Skenario ini bertujuan untuk menganalisis kemampuan sistem dalam mengikuti perubahan kecepatan referensi (nr_{ref}) dan memvalidasinya berdasarkan kriteria pada Tabel 3.4. Prosedur pengujiannya adalah dengan mengatur nilai nr_{ref} mulai dari 200 RPM hingga 1400 RPM dengan interval 100 RPM pada setiap pengujian, sementara arus medan ($i_{ds_{ref}}$) dijaga konstan pada 0,35 A. Kemudian pengujian dilakukan pada kondisi tanpa beban ($T_L = 0$ N.m) dan dengan beban konstan ($T_L = 2$ N.m)

2. Skenario B: Respons terhadap perubahan arus medan

Skenario ini bertujuan untuk memvalidasi efektivitas *decoupling* IFOC, yang sesuai dengan kriteria validasi kecepatan aktual (nr_{act}) dan arus medan aktual ($i_{ds_{act}}$) pada Tabel 3.4. Prosedur pengujiannya adalah dengan menjaga referensi kecepatan (nr_{ref}) konstan pada 1000 RPM, sementara perubahan arus medan referensi ($i_{ds_{ref}}$). diatur mulai dari 0,2 A hingga 0,7 A, dengan interval 0,05 A pada setiap pengujian, dan dilakukan pada kondisi tanpa beban ($T_L = 0$ N.m) dan dengan beban konstan ($T_L = 2$ N.m).

Jika respons sistem pada kedua skenario pengujian sesuai dengan kriteria validasi pada Tabel 3.4 maka model dianggap tervalidasi dan dapat dilanjutkan ke tahap implementasi perangkat keras (sub-bab 3.7).

3.7 Implementasi Model Sistem Ke Perangkat Keras

Setelah model sistem IFOC divalidasi melalui simulasi, tahap selanjutnya adalah mengimplementasikan model tersebut ke perangkat keras. Proses ini melibatkan penyesuaian model Simulink agar kompatibel dengan modul percobaan *Power Electronics and Drives* 300 W Lucas-Nuelle. Penyesuaian ini dilakukan dengan mengganti beberapa blok standar Simulink dengan blok-blok spesifik dari *library* Lucas-Nuelle (LN) yang telah terintegrasi dengan MATLAB. Blok-blok ini berfungsi sebagai antarmuka langsung antara algoritma kontrol yang dirancang dengan perangkat keras seperti sensor, aktuator, dan prosesor *Digital Signal Processing* (DSP) pada modul.

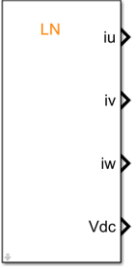
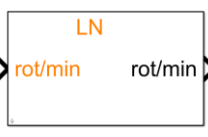
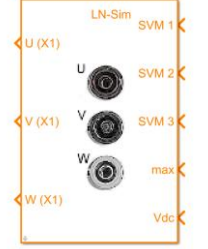
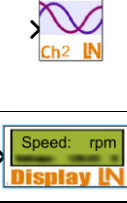
3.7.1 Penyesuaian model sistem *Indirect Field Oriented Control* (IFOC)

Model sistem IFOC yang semula murni untuk simulasi dimodifikasi dengan menambahkan dan mengganti beberapa blok fungsional. Tujuan utama dari

penyesuaian ini adalah untuk memastikan model dapat dikompilasi dan diunggah (melalui proses *Build and Deploy*) ke *real-time controller* pada perangkat keras. Gambar 3.40 di bawah ini menunjukkan model IFOC yang telah disesuaikan menggunakan blok-blok dari library Lucas-Nuelle.

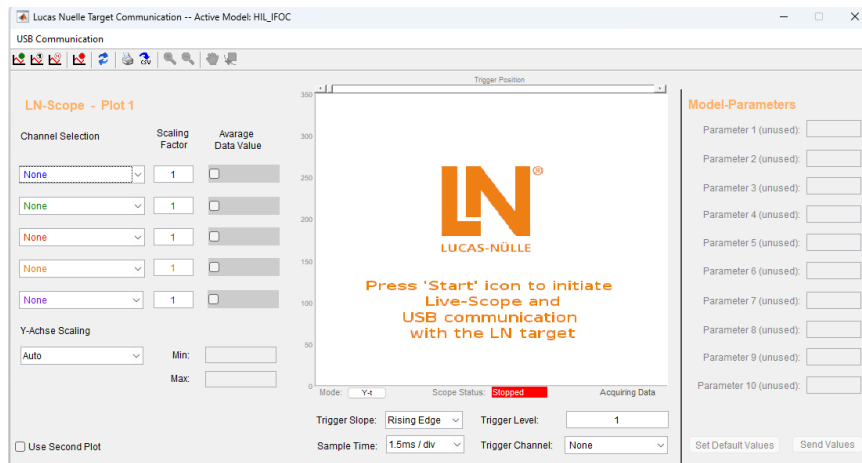
Blok-blok utama dari *library* Lucas-Nuelle yang digunakan dalam model di atas dijelaskan fungsinya pada Tabel 3.5 berikut:

Tabel 3.5 Komponen-komponen Lucas-Nuelle

No.	Komponen	Gambar	Fungsi
1	<i>AD-Values</i>		Menjadi antarmuka ke <i>Analog-to-Digital Converter</i> (ADC) untuk membaca nilai arus tiga fasa (iu, iv, iw) dan tegangan DC-Link (Vdc) secara <i>real-time</i>
2	<i>Clarke-Park Transformation</i>		Menyediakan fungsi transformasi vektor (Clarke dan Park) yang telah dioptimalkan untuk dijalankan pada prosesor DSP perangkat keras
3	<i>PI Controller</i>		Blok kontroler PI diskrit dengan fitur <i>anti-windup</i> yang esensial untuk implementasi pada perangkat keras, menggantikan kontroler PI standar
4	<i>Speed Calculation</i>		Menjadi antarmuka untuk membaca nilai kecepatan motor dari sensor <i>encoder</i> pada perangkat keras
5	<i>Sim Converter</i>		Menjadi antarmuka ke modul inverter saat diimplementasikan dan juga dapat mensimulasikan perilaku inverter untuk pengujian <i>closed-loop</i>
6	<i>Live Scope dan Display LN</i>		Blok komunikasi untuk mengirimkan data dari perangkat keras ke komputer secara <i>real-time</i> untuk monitoring melalui <i>Graphical User Interface</i> (GUI)
7	<i>Software Error</i>		Blok untuk memicu penghentian operasi jika terjadi kondisi berbahaya yang terdeteksi oleh logika proteksi

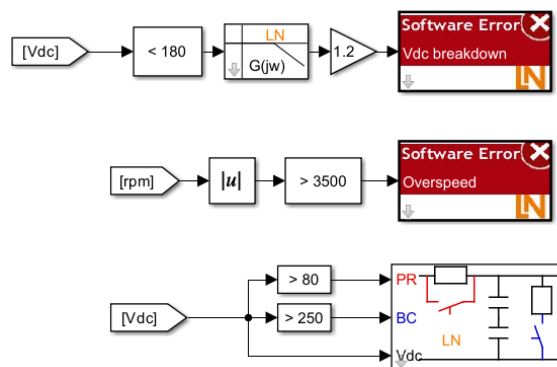
Berdasarkan gambar dan tabel di atas, beberapa penyesuaian kunci yang dilakukan pada model adalah:

1. **Penyesuaian blok matematis:** Blok-blok transformasi Clarke dan Park yang sebelumnya dibuat secara manual, kini digantikan oleh blok LN Clarke-Park yang lebih efisien untuk kompilasi kode
2. **Integrasi sensor dan aktuator:** *Input* dan *output* model yang semula menggunakan sumber virtual, kini dihubungkan ke blok antarmuka perangkat keras seperti AD-Values untuk membaca arus dan Speed Calculation untuk membaca kecepatan
3. **Penggunaan kontroler khusus:** Blok PI standar diganti dengan blok $PI\ U(t)$ dari library LN untuk memastikan kompatibilitas dan kinerja yang optimal
4. **Implementasi sistem komunikasi dan monitoring:** Untuk memungkinkan monitoring secara real-time, model dilengkapi dengan blok-blok komunikasi seperti Live Scope dan Display LN. Blok-blok ini mengirimkan data variabel penting (seperti kecepatan dan arus) dari perangkat keras ke komputer. Data ini kemudian divisualisasikan melalui
5. **Graphical User Interface (GUI)** dari Lucas-Nuelle, seperti yang ditunjukkan pada Gambar 3.41, sehingga unjuk kerja sistem dapat diamati secara langsung.



Gambar 3.41 Antarmuka GUI Lucas-Nuelle untuk Monitoring Real-Time

6. Implementasi sistem proteksi: Aspek krusial dalam transisi dari simulasi ke perangkat keras adalah keamanan. Oleh karena itu, model dilengkapi dengan logika proteksi menggunakan blok Software Error. Seperti yang ditunjukkan pada Gambar 3.42, sistem dirancang untuk secara otomatis menghentikan operasi jika mendeteksi kondisi berbahaya, seperti kecepatan motor melebihi batas aman (3500 RPM) atau tegangan DC-Link turun di bawah ambang batas kritis (180 V).



Gambar 3.42 Implementasi Logika Proteksi pada Model

Setelah proses penyesuaian model selesai, maka dilakukan proses “*Build and Deploy*”. Proses ini merupakan fitur dari MATLAB/Simulink yang secara otomatis mengubah model grafis yang telah dirancang menjadi kode C/C++ yang dapat dieksekusi, kemudian mengompilasi dan mengunggahnya langsung ke

prosesor DSP (*Digital Signal Processing*) pada perangkat keras Lucas-Nuelle. Proses ini memanfaatkan fitur Simulink Coder dan Embedded Coder.

Secara konseptual, proses “*Build and Deploy*” yang dilakukan dalam penelitian ini meliputi tiga langkah utama yang dieksekusi secara otomatis oleh Simulink:

1. **Pembangkitan kode (*Code Generation*):** Simulink secara otomatis menghasilkan kode sumber (*source code*) dalam bahasa C. Kode ini merupakan representasi tekstual dari setiap blok dan logika yang ada di dalam model
2. **Kompilasi (*Compilation*):** Kode C yang telah dihasilkan kemudian dikompilasi menggunakan *compiler* yang sesuai untuk arsitektur prosesor DSP pada modul Lucas-Nuelle. Proses ini mengubah kode sumber menjadi file *executable* atau *binary* yang dapat dipahami dan dijalankan oleh mikrokontroler
3. **Pengunggahan (*Deployment/Download*):** File *executable* yang telah berhasil dikompilasi kemudian diunggah ke memori mikrokontroler pada modul Lucas-Nuelle melalui koneksi USB. Setelah proses ini selesai, model IFOC yang tadinya hanya ada di komputer kini telah tertanam di dalam perangkat keras dan siap untuk dijalankan secara real-time untuk mengendalikan motor induksi

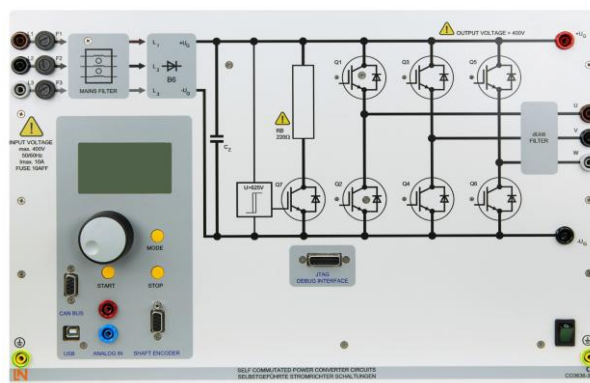
Proses ini secara signifikan mempercepat transisi dari tahap desain berbasis model ke implementasi fisik, memastikan bahwa logika yang telah divalidasi melalui simulasi dapat diterapkan secara akurat pada perangkat keras.

3.7.2 *Power Electronics and Drives* 300 W Lucas-Nuelle (Perangkat Keras)

Pada sub-bab ini akan dijelaskan mengenai konfigurasi dan komponen perangkat keras yang digunakan untuk merealisasikan perancangan sistem kendali yang telah diuraikan pada Sub-bab 3.2.1. Perangkat keras ini berfungsi sebagai *plant* fisik dan unit pemroses untuk menjalankan algoritma IFOC. Seluruh komponen merupakan bagian dari modul percobaan *Power Electronics and Drives* 300 W dari Lucas-Nuelle. Rangkaian perangkat keras ini berfungsi sebagai sistem fisik yang akan dikendalikan oleh model IFOC yang telah di-*deploy*.

Modul perangkat keras *Power Electronics and Drives* 300 W dari Lucas-Nuelle yang digunakan dalam penelitian ini adalah sebagai berikut:

1. Modul kontroler dan inverter



Gambar 3.43 Modul kontroler dan inverter

Modul kontroler dan inverter pada Gambar 3.43 merupakan pusat pemrosesan dan unit daya (inverter). Bagian kontroler dilengkapi prosesor DSP yang menjalankan algoritma IFOC, sementara bagian inverter berbasis IGBT berfungsi untuk mengubah tegangan DC menjadi tegangan AC tiga fasa untuk motor. Spesifikasi dan fitur utama ditunjukkan pada tabel 3.6 berikut:

Tabel 3.6 Fitur dan spesifikasi modul kontroler dan inverter

Fitur/Parameter	Deskripsi/Nilai
Rangkaian	Terdapat rangkaian penyearah, DC <i>link</i> , inverter, rangkaian filter dan unit kontroler

Pengendalian	Unit pemicuan gerbang (<i>gate firing</i>) dan pengukuran yang dikendalikan unit kontroler pada modul
Pengukuran	Pengukuran arus dan tegangan tiga fasa secara <i>real time</i>
Konektivitas	Antarmuka untuk koneksi ke Matlab (komputer) melalui <i>port</i> USB
Frekuensi PWM	Dapat dipilih
Tegangan output	3x47-400 V
Frekuensi output	50-60
Daya output maksimal	1 kVA

Berdasarkan Tabel 3.6, modul ini memegang peranan vital dalam implementasi sistem karena mengintegrasikan dua fungsi utama yang telah dirancang pada Gambar 3.3 ke dalam satu perangkat keras tunggal. Pertama, modul ini berfungsi sebagai blok kontroler melalui unit pemroses sinyal digital (DSP) internal yang bertugas menjalankan algoritma FOC, mengolah sinyal umpan balik, dan menghasilkan sinyal PWM. Kedua, modul ini sekaligus berfungsi sebagai inverter melalui rangkaian daya berbasis IGBT yang mengeksekusi sinyal kendali tersebut untuk menyuplai daya listrik variabel ke motor.

2. Motor induksi tiga fasa



Gambar 3.44 Motor induksi tiga fasa

Motor induksi pada Gambar 3.44 merupakan objek yang dikendalikan. Spesifikasi motor dijelaskan pada Tabel 3.7 berikut:

Tabel 3.7 Spesifikasi motor induksi

Parameter	Nilai	Satuan
Tipe motor	Motor induksi 3 fasa	
Tipe rotor	Sangkar Tupai (<i>Squirrel Cage</i>)	
Jumlah Kutub	4	
Data nominal:		
Daya nominal	0,37	kW
Tegangan nominal (Y/ Δ)	400/230	V
Arus nominal (Y/ Δ)	0,95/1,65	A
Frekuensi	50	Hz
Kecepatan nominal	1400	RPM
Faktor daya	0,76	
Torsi nominal	2,5	N.m

3. Sensor kecepatan



Gambar 3.45 Sensor kecepatan

Sensor kecepatan pada Gambar 3.45 terpasang pada poros motor untuk memberikan umpan balik sinyal kecepatan dan posisi rotor secara presisi ke modul kontroler. Spesifikasi utamanya memiliki resolusi 1024 pulsa/putaran dengan kecepatan maksimum yang dapat diukur sebesar 6000 RPM.

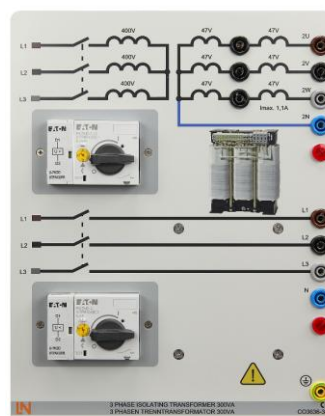
4. Modul beban mekanis terkendali



Gambar 3.46 Modul beban mekanis terkendali

Modul beban mekanis terkendali pada Gambar 3.46 berfungsi sebagai unit pembebanan mekanis terkendali bagi motor induksi. Unit ini terdiri dari *servo-brake* yang dikendalikan oleh unit kontrol digital. Memungkinkan pengaturan torsi yang sangat presisi hingga 12 N.m. Sistem ini beroperasi dalam empat kuadran (*four-quadrant operation*), sehingga mampu menyimulasikan berbagai karakteristik beban dinamis maupun statis, seperti beban konstan, kipas, atau *flywheel*, secara akurat.

5. Modul catu daya tiga fasa



Gambar 3.47 Catu daya tiga fasa

Catu daya tiga fasa pada Gambar 3.47 berfungsi sebagai catu daya utama yang menyediakan tegangan AC yang kemudian akan disearahkan menjadi tegangan bus DC dan digunakan oleh modul inverter. Spesifikasi utama ditunjukkan pada Tabel 3.8 berikut:

Tabel 3.8 Catu daya tiga fasa		
Parameter	Nilai	Satuan
Daya nominal	300	VA
Tegangan input	3x400	V
Frekuensi	50/60	Hz
Tegangan output	3 x 94	V

6. Komputer

Sebuah komputer (PC/laptop) digunakan sebagai stasiun kerja untuk merancang model di Matlab/Simulink, melakukan proses "*Build and Deploy*" ke modul kontroler, dan memonitor data pengujian secara *real-time* melalui antarmuka GUI. Spesifikasi perangkat lunak MATLAB R2022a (atau lebih tinggi) dengan library Simulink, Simulink Coder, Embedded Coder

7. Modul multimeter



Gambar 3.48 Modul multimeter

Modul multimeter pada Gambar 3.48 merupakan instrumen pengukuran digital presisi tinggi yang berfungsi untuk memantau dan menganalisis parameter kelistrikan vital seperti tegangan, arus, berbagai jenis daya (aktif, reaktif,

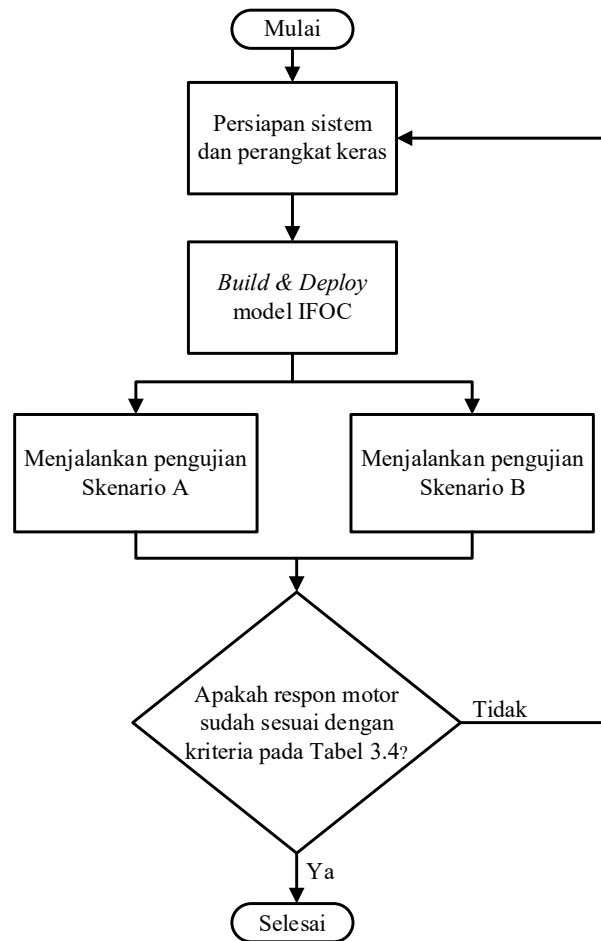
semu), serta faktor daya secara *real-time*. Dalam penelitian ini, modul ini berperan krusial karena kemampuannya mengukur nilai RMS secara independen sehingga sangat efektif untuk mengukur tegangan *output* inverter.

3.8 Pengujian Perangkat Keras

Setelah model IFOC diimplementasikan ke perangkat keras (Sub-bab 3.7), tahap ini bertujuan untuk memvalidasi bahwa sistem pada perangkat keras berjalan dengan benar. Untuk memvalidasi perangkat keras, skenario pengujian yang sama dengan validasi model sistem (Sub-bab 3.6) dijalankan pada perangkat keras Lucas-Nuelle:

1. Skenario A: Respons pelacakan kecepatan
2. Skenario B: Respons terhadap perubahan arus medan

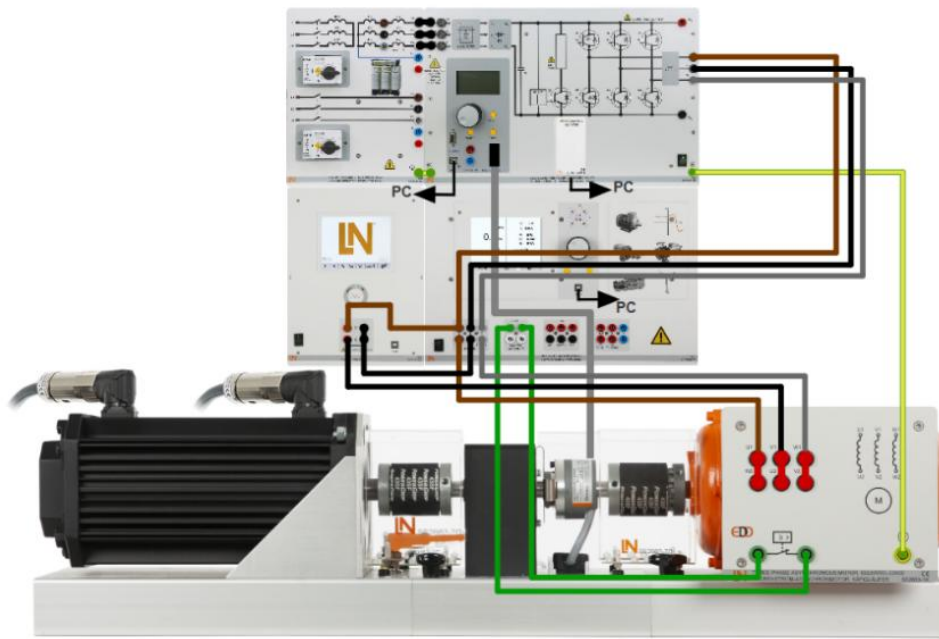
Perangkat keras dianggap tervalidasi jika hasil pengujian Skenario A dan B memenuhi kriteria validasi yang sama pada Tabel 3.4. Jika hasil pengujian perangkat keras sesuai dengan kriteria, maka sistem pada perangkat keras dinyatakan valid dan siap digunakan untuk pengujian karakteristik motor. Proses pengujian ini dirangkum secara visual dalam diagram alir pada Gambar 3.49.



Gambar 3.49 Diagram alir pengujian model sistem IFOC (perangkat keras)

3.8.1 Konfigurasi Perangkat Keras

Konfigurasi modul percobaan *Power Electronics and Drives* 300 W dari Lucas-Nuelle dengan menggunakan unit modul yang dijelaskan pada sub-bab 3.7.2 untuk menguji sistem FOC adalah sebagai berikut:



Gambar 3.50 Diagram pengkabelan untuk pengujian FOC

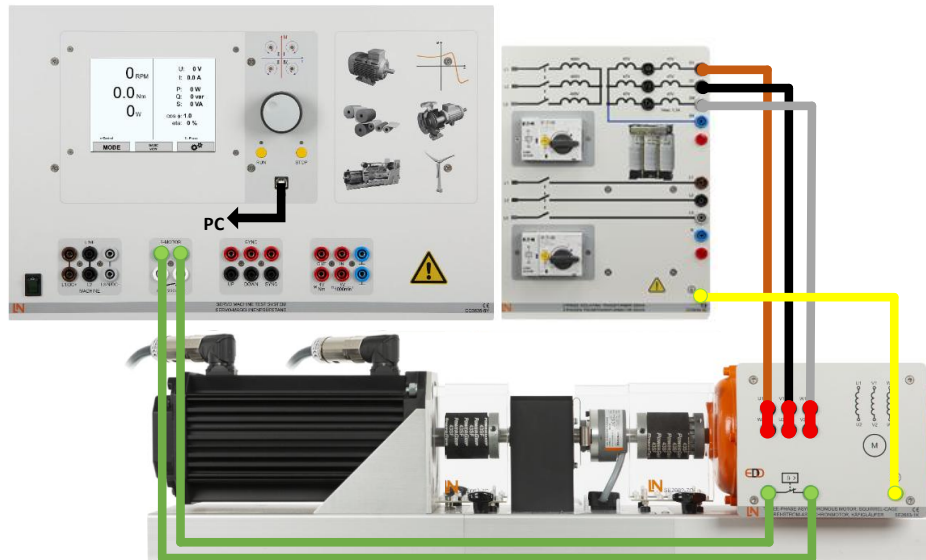
3.9 Pengujian Karakteristik Motor Induksi

Pada sub-bab ini, dilakukan pengujian eksperimental akhir yang bertujuan untuk menganalisis dan membandingkan karakteristik motor induksi sebelum dan sesudah menggunakan metode *Field Oriented Control* (FOC). Pengujian ini berfokus pada perbandingan karakteristik kondisi tunak (*steady-state*) motor saat merespons beban mekanis.

Untuk mendapatkan data perbandingan karakteristik motor sebelum dan sesudah menggunakan metode FOC, dua metode kontrol diuji pada perangkat keras menggunakan prosedur pembebanan mekanis yang sama yaitu beban dinaikkan secara bertahap (*ramp*) dari 0 N.m hingga 3 N.m pada interval 0,2 N.m.

1. Metode sebelum menggunakan FOC yang digunakan adalah *Direct On Line* (DOL) yaitu suplai tegangan dan frekuensi konstan (50 Hz) menggunakan modul catu daya tiga fasa. Pada metode ini kecepatan motor tidak dikendalikan (*open loop*) dan akan turun secara alami seiring kenaikan beban mekanis.

Konfigurasi pengkabelan untuk pengujian DOL ditunjukkan pada Gambar 3.51 berikut:



Gambar 3.51 Diagram pengkabelan untuk pengujian DOL.

2. Metode sesudah menggunakan FOC yang digunakan adalah *Indirect Field Oriented Control* (IFOC) yaitu kecepatan referensi ($n_{r_{ref}}$) diatur sesuai dengan kecepatan maksimal motor sebelum menggunakan FOC yaitu 1486 RPM. Pada metode ini FOC akan secara aktif berupaya mempertahankan kecepatan motor meskipun beban mekanis dinaikkan. Untuk konfigurasi pengkabelan pada pengujian FOC ini, digunakan konfigurasi yang sama seperti pengujian perangkat keras yang dijelaskan pada sub-bab 3.8.1

Ketika beban mekanis dinaikkan secara bertahap (*ramp*) dari 0 N.m hingga 3 N.m, data kondisi tunak akan dicatat untuk menghasilkan kurva karakteristik motor. Data yang direkam meliputi kecepatan, torsi, slip, tegangan *line to line* RMS ($V_{LL_{RMS}}$), arus *line* RMS ($i_{L_{RMS}}$) dan faktor daya. Data tersebut menjadi dasar analisis untuk mengevaluasi karakteristik motor sebelum dan sesudah menggunakan metode FOC.

3.10 Waktu dan Tempat Penelitian

Proses pelaksanaan penelitian dan penyusunan laporan tugas akhir ini dimulai pada bulan September tahun 2024 yang bertempat di Laboratorium Teknik Elektro Fakultas Teknik Universitas Siliwangi Kampus Mugarsari, Kota Tasikmalaya.

