

## BAB II

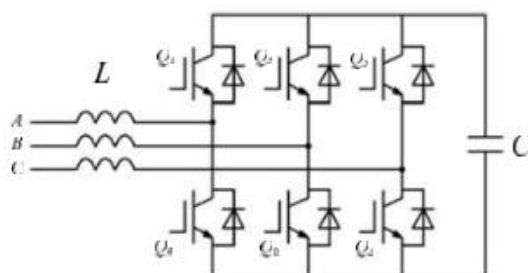
### LANDASAN TEORI

#### 2.1 Filter Aktif

Filter aktif merupakan rangkaian elektronika daya berupa PWM konverter yang di rancang untuk mengurangi harmonisa dengan cara menyuntikkan arus untuk mengkompensasi harmonisa yang di timbulkan oleh beban nonlinier (Sibuea & Thayib, 2014). Komponen utama pada filter aktif adalah konverter dan controller (Odinanto et al., 2008), sehingga kinerja filter aktif ini sangat bergantung pada jenis kontrol pembangkit arus referensi yang digunakan. Konverter dalam filter aktif dihubungkan dengan sumber tegangan beban untuk mengkompensasi harmonisa.

Penentuan pola pensaklaran pada konverter pada filter aktif yaitu bekerja dengan cara membandingkan arus kompensasi ( $I_{ref}$ ) dengan arus feedback ( $I_{abc}$ ) yang merupakan output dari konverter. Keluaran dari konverter digunakan untuk mengkompensasi bagian gelombang sinusoidal yang hilang pada saat arus mengalir menuju ke beban non linier dengan cara membangkitkan arus yang berlawanan fasa dengan arus harmonisa.

Kontroller digunakan untuk memperkecil sinyal *error* pada sumber tegangan beban yang terdapat harmonisa. Keluaran dari kontroller akan digunakan untuk mentrigger inverter. Gambar 2.1 merupakan skema komponen *switching* dari filter aktif.



Gambar 2.1 Filter Aktif

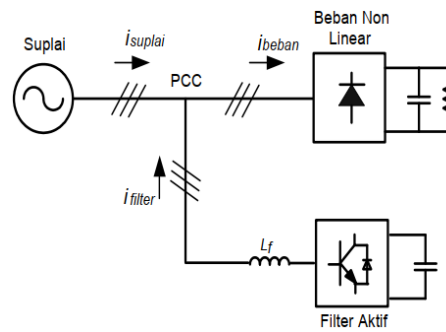
(Assaffat et al., 2013)

Filter aktif dibagi menjadi dua jenis berdasarkan caranya dihubungkan dengan sistem, yaitu filter aktif seri (*series*) dan filter aktif paralel (*shunt*). Filter yang paling sering dipakai adalah jenis filter aktif shunt, hal ini disebabkan karena filter aktif *shunt* tidak membutuhkan tambahan transformator sebagaimana filter aktif *series* (Sibuea & Thayib, 2014).

### 2.1.1 Filter Aktif *Shunt*

Gambar 2.2 menunjukkan konfigurasi sistem filter aktif *shunt* tiga fasa. Filter aktif *shunt* pada dasarnya menggunakan konverter jenis *voltage source inverter* (VSI) untuk mengkompensasi harmonisa arus beban (A. K. Gautam et al., 2016).

Saat fasa arus filter aktif *shunt* dan fasa arus beban mempunyai fasa yang sama ataupun fasanya berlawanan pada frekuensi harmonisa maka kedua fasa akan saling menghilangkan sehingga jumlah vektor arus menjadi nol pada arus sumber ( $i_{suplai}$ ) di *Point of Common Coupling* (PCC) sehingga arus suplai mendekati sinusoidal.

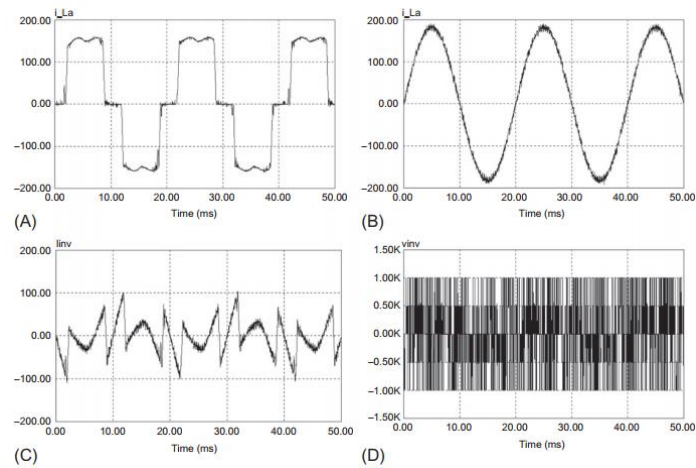


Gambar 2.2 Skema Filter Aktif Shunt

(Meliala, 2011)

Filter aktif *shunt* terdiri dari rangkaian konverter, keluaran dari konverter dihubungkan dengan L ataupun LC yang dipasang secara paralel dengan beban yang memiliki peran penting dalam mengoperasikan filter aktif secara stabil dan benar agar terjadi kompensasi arus (Meliala, 2011).

Prinsip kerja dari filter aktif *shunt* yaitu dengan mendapatkan inputan dari sumber tegangan dan arus beban sesaat ( $i_{beban}$ ) yang digunakan untuk memperoleh arus referensi harmonisa pada sistem kontrol. Arus referensi tersebut akan digunakan untuk referensi pembangkit lebar pulsa (PWM) yang akan dibandingkan dengan sinyal *carrier* yang akan menghasilkan pulsa. Pulsa tersebut digunakan untuk pemicu konverter yang kemudian digunakan untuk membangkitkan gelombang arus filter untuk diinjeksikan ke dalam sistem (Prabowo et al., 2015). Bentuk gelombang dari filter aktif shunt dapat dilihat pada Gambar 2.3.

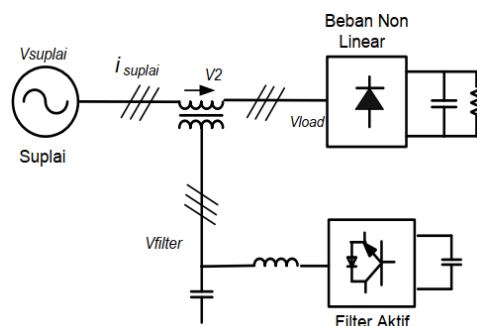


Gambar 2.3 Bentuk gelombang pada filter aktif *shunt* (A) Arus beban, (B) Arus sumber setelah pemasangan filter, (C) Arus kompensasi, (D) Tegangan output komponen aktif.

(Rashid, 2018)

### 2.1.2 Filter Aktif Seri

Berbeda dengan filter aktif shunt, filter aktif seri lebih banyak digunakan sebagai isolator harmonisa dan mengkompensasi distorsi tegangan seperti tegangan kedip, dan tegangan tidak pada level sistem tegangan tinggi dan tegangan rendah. Pada filter aktif seri, keluaran dari konverter dihubungkan dengan filter L atau LC kemudian dikopling dengan transformator (Meliala, 2011). Filter aktif seri dihubungkan secara seri diantara suplai dengan beban seperti diperlihatkan pada Gambar 2.4.

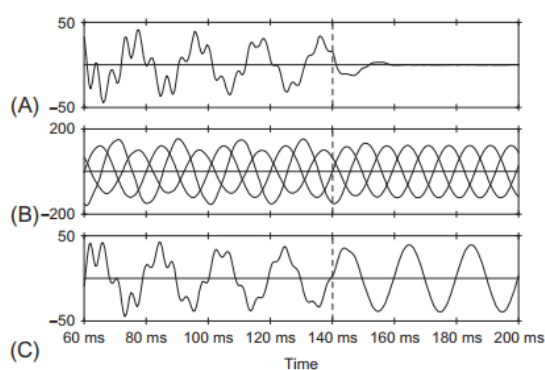


Gambar 2.4 Skema Filter Aktif Seri

(Meliala, 2011)

Prinsip kerja dari filter aktif seri yaitu konverter menghasilkan tegangan keluaran ( $v_{filter}$ ) yang nilainya sebanding dengan arus harmonisa sumber ( $i_{suplai}$ ). Pada tegangan keluaran ( $v_2$ ) kopling transformator sisi sekunder, nilainya sebanding terhadap arus mengalir melalui transformator kopling atau disebut dengan tahanan aktif pada frekuensi harmonisa. Arus harmonisa pada sumber akan berkurang seiring dengan naiknya impedansi frekuensi harmonisa sumber yang disebabkan oleh tahanan aktif (Meliala, 2011).

Seperti telah disebutkan di atas, filter aktif seri juga digunakan untuk memfilter harmonisa tegangan dan kompensasi kedip tegangan. Bentuk gelombang arus dan tegangan dari filter aktif seri ditunjukkan pada Gambar 2.5.

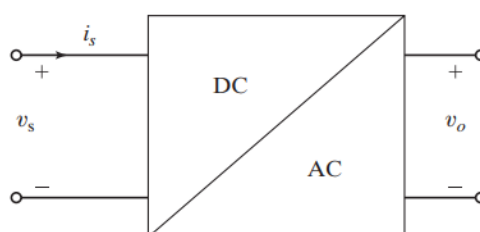


Gambar 2.5 Bentuk gelombang sebelum dan sesudah pemasangan filter aktif seri,

(a) Arus netral sistem, (b) Tegangan Fasa-Netral, (c) Arus sumber tegangan

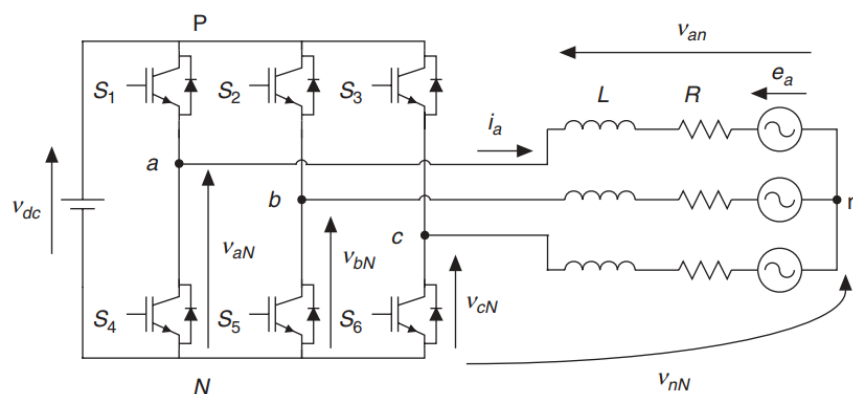
## 2.2 Inverter Tiga Fasa Sumber Tegangan

Konverter yang digunakan pada filter aktif adalah jenis konverter DC ke AC atau dikenal sebagai inverter. Tegangan atau arus yang diinputkan ke inverter adalah DC dan output yang dihasilkan adalah AC seperti yang ditunjukkan pada Gambar 2.6. Output idealnya harus berupa AC gelombang sinus murni, tetapi tegangan output dari inverter praktis mengandung harmonisa.



Gambar 2.6 Blok Diagram Inverter

(Rashid, 2014)



Gambar 2.7 Rangkaian Inverter Tiga Fasa Sumber Tegangan

(Rodriguez & Cortes, 2012)

Pada Gambar 2.7 dapat diketahui bahwa rangkaian inverter tiga fasa sumber tegangan terdiri dari 6 buah komponen pensaklaran yang beroperasi dalam mode komplementer untuk menghindari adanya hubung singkat sumber DC ( $V_{dc}$ ).

Kondisi pensaklaran  $S_x$ , dengan  $x=1,\dots,6$ , diwakili dengan sinyal pensaklaran  $S_a$ ,  $S_b$ , dan  $S_c$  yang di definisikan pada Tabel 2.1 Kondisi Sinyal Pensaklaran.

Tabel 2.1 Kondisi Sinyal Pensaklaran

Sumber: (V. K. Singh et al., 2018)

| <b>Sa</b>                      | <b>Sb</b>                      | <b>Sc</b>                      |
|--------------------------------|--------------------------------|--------------------------------|
| $S_1$ On (1) dan $S_4$ Off (0) | $S_2$ On (1) dan $S_5$ Off (0) | $S_3$ On (1) dan $S_6$ Off (0) |
| $S_1$ Off (0) dan $S_4$ On (1) | $S_2$ Off (0) dan $S_5$ On (1) | $S_3$ Off (1) dan $S_6$ On (0) |

Sinyal pensaklaran ini menentukan nilai tegangan output yang di nyatakan dalam persamaan (2.1a).

$$v_{aN} = S_a V_{dc} \quad (2.1a)$$

$$v_{bN} = S_b V_{dc} \quad (2.1b)$$

$$v_{cN} = S_c V_{dc} \quad (2.1c)$$

Dimana  $V_{dc}$  adalah tegangan sumber DC.

Mengingat vektor kesatuan  $\mathbf{a} = e^{j\frac{2\pi}{3}} = -\frac{1}{2} + j\frac{\sqrt{3}}{2}$ , yang merepresentasikan perpindahan antar fasa sebesar  $120^\circ$ . Vektor tegangan keluaran dapat didefinisikan pada persamaan (2.2).

$$v = \frac{2}{3} (v_{aN} + \mathbf{a}v_{bN} + \mathbf{a}^2v_{cN}) \quad (2.2)$$

Pada kondisi pensaklaran  $(S_a, S_b, S_c) = (0, 0, 0)$  menghasilkan nilai vektor tegangan pada persamaan (2.3).

$$v = \frac{2}{3} (0 + \mathbf{a}0 + \mathbf{a}^20) = 0 \quad (2.3)$$

Pada kondisi pensaklaran  $(1, 0, 0)$  menghasilkan nilai vektor tegangan pada persamaan (2.4)

$$v = \frac{2}{3} (V_{dc} + \mathbf{a}0 + \mathbf{a}^20) = \frac{2}{3} V_{dc} \quad (2.4)$$

Pada kondisi pensaklaran (1,1,0) menghasilkan nilai vektor tegangan pada persamaan (2.5).

$$\begin{aligned}
 v &= \frac{2}{3}(V_{dc} + \mathbf{a}V_{dc} + \mathbf{a}^2 0) \\
 &= \frac{2}{3}\left(V_{dc} + \left(-\frac{1}{2} + j\frac{\sqrt{3}}{2}\right)V_{dc}\right) \\
 &= \frac{V_{dc}}{3} + j\frac{\sqrt{3}}{2}V_{dc}
 \end{aligned} \tag{2.5}$$

Pada kondisi pensaklaran (1,1,1) menghasilkan nilai vektor tegangan pada persamaan (2.6).

$$v = \frac{2}{3}(V_{dc} + \mathbf{a}V_{dc} + \mathbf{a}^2V_{dc}) = \frac{2}{3}V_{dc}(1 + \mathbf{a} + \mathbf{a}^2) = 0 \tag{2.6}$$

Sinyal pensaklaran  $S_a, S_b, S_c$  menghasilkan 8 kombinasi kondisi pensaklaran dengan nilai vektor tegangan seperti pada Tabel 2.2 Kombinasi Pensaklaran dan Vektor Tegangan.

Tabel 2.2 Kombinasi Pensaklaran dan Vektor Tegangan

Sumber: (Rodriguez & Cortes, 2012)

| $S_a$ | $S_b$ | $S_c$ | Vektor Tegangan V                                      |
|-------|-------|-------|--|
| 1     | 1     | 1     | $V_0 = 0$  |
| 0     | 0     | 0     | $V_1 = 0$  |
| 1     | 0     | 0     | $V_2 = \frac{2}{3}V_{dc}$                              |
| 1     | 1     | 0     | $V_3 = \frac{1}{3}V_{dc} + j\frac{\sqrt{3}}{3}V_{dc}$  |
| 0     | 1     | 0     | $V_4 = -\frac{1}{3}V_{dc} + j\frac{\sqrt{3}}{3}V_{dc}$ |
| 0     | 1     | 1     | $V_5 = -\frac{2}{3}V_{dc}$                             |
| 0     | 0     | 1     | $V_6 = -\frac{1}{3}V_{dc} - j\frac{\sqrt{3}}{3}V_{dc}$ |
| 1     | 0     | 1     | $V_7 = \frac{1}{3}V_{dc} - j\frac{\sqrt{3}}{3}V_{dc}$  |



Konverter jenis ini menggunakan kapasitor sebagai sumber tegangan DC (Prabowo et al., 2015). Nilai tegangan DC yang dibutuhkan untuk perancangan filter aktif dapat dipahami pada persamaan (2.10):

$$V_{DC} = 2\sqrt{2}V_{s,max} \quad (2.7)$$

dengan,

$V_{DC}$  = Tegangan DC yang dibutuhkan

$V_{s,max}$  = Tegangan maksimal sumber

Dengan diketahuinya nilai  $V_{DC}$ , nilai kapasitor sebagai sumber tegangan DC didapatkan dengan persamaan (2.11) (B. Singh et al., 2015).

$$C_{DC} = \frac{I_d}{2\omega V_{DC,ripple}} \quad (2.8)$$

dengan,

$C_{DC}$  = DC link kapasitor untuk aktif filter (Farad)

$I_d$  = Arus yang dihasilkan filter aktif (Ampere)

$\omega$  = Frekuensi fundamental dari sumber (rad/s)

$V_{DC,ripple}$  = Nilai maksimum dari *ripple* pada DC link (Volt)

Nilai  $V_{DC,ripple}$  dipertimbangkan 1-3% dari nilai  $V_{DC}$  filter aktif . Nilai arus yang dihasilkan oleh filter aktif dapat dipahami sebagai berikut :

$$I_d = \frac{S_{filter}}{V_{DC}} \quad (2.9)$$

dengan,

$I_d$  = Arus yang dihasilkan filter aktif (Ampere)

$S_{filter}$  = rating harmonik dari filter aktif (VA)

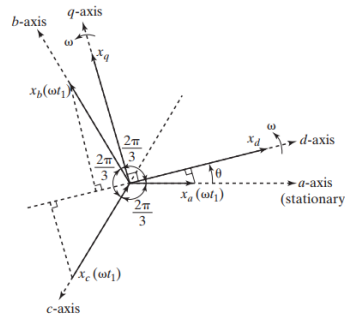
$V_{DC}$  = tegangan DC yang dibutuhkan (Volt)

Rating harmonik dari filter aktif dapat dipahami sebagai perkalian dari tegangan fundamental sumber dengan komponen arus harmonisa.

### 2.3 Perhitungan Arus Referensi

#### 2.3.1 Transformasi Park

Dalam sebuah kendali filter aktif, diperlukan ekstraksi arus referensi dan kontrol arus (Sathiyarayanan & Mishra, 2016). Salah satu metode yang populer digunakan adalah berbasis *Synchronous Reference Frame Theory* (SRFT) atau teori d-q. Dalam teori kompensasi SRFT, arus beban di konversi ke koordinat *dq0* menggunakan transformasi Park. Tegangan sumber digunakan untuk menghitung sudut sinus dan sudut cosinus menggunakan *Phase Locked Loop* (PLL), dimana nilai sudut tersebut digunakan di dalam teori SRF untuk memisahkan harmonisa dari fundamentalnya (Babu P et al., 2017). Pada Gambar 2.8 menampilkan transformasi koordinat *abc* ke koordinat *dq*.



Gambar 2.8 Proyeksi koordinat dq terhadap koordinat abc

(Rashid, 2014)

$$\begin{bmatrix} i_{ld} \\ i_{lq} \\ i_{l0} \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin(\theta) & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} i_{la} \\ i_{lb} \\ i_{lc} \end{bmatrix} \quad (2.10)$$

Pada persamaan (2.10) merupakan transformasi Park, ' $\theta$ ' adalah sudut fasa urutan positif dari sistem tegangan yang dihasilkan oleh unit *phase locked loop* (PLL). PLL membangkitkan fungsi  $\sin \theta$  dan  $\cos \theta$  pada frekuensi fundamental, kemudian disinkronkan dengan komponen fundamental tegangan (Setiyono & Dwinanto, 2021).

Setelah arus beban  $i_{ld}$  dan  $i_{lq}$  pada koordinat  $dq0$  ditentukan, selanjutnya diperlukan *low pass filter* untuk mengekstraksi komponen DC dan komponen AC pada komponen aktif  $i_{ld}$  yang ditulis dalam persamaan (2.11).

$$i_{ld} = \bar{i}_{ld} + \tilde{i}_{ld} \quad (2.11)$$

Komponen DC  $\bar{i}_{ld}$  adalah nilai arus fundamental yang disuplai sumber, sedangkan komponen AC  $\tilde{i}_{ld}$  adalah komponen harmonisa dari arus beban (Ali et al., 2016). Nilai arus referensi pada koordinat  $dq0$  diperoleh dengan persamaan (2.12).

$$\begin{bmatrix} i_{cd}^* \\ i_{cq}^* \end{bmatrix} = \begin{bmatrix} \tilde{i}_{ld} \\ i_{lq} \end{bmatrix} \quad (2.12)$$

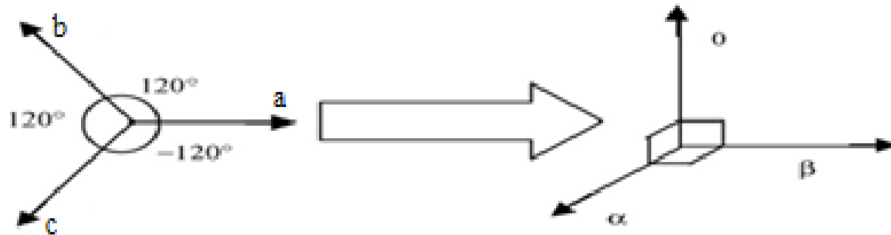
Langkah selanjutnya adalah transformasi balik ke koordinat  $abc$ . Arus referensi pada koordinat  $abc$  dapat ditentukan dengan transformasi balik Park yang ditulis dalam persamaan (2.13).

$$\begin{bmatrix} i_{fa} \\ i_{fb} \\ i_{fc} \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos \theta & -\sin(\theta) \\ \cos\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta - \frac{2\pi}{3}\right) \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix} \begin{bmatrix} i_{cd}^* \\ i_{cq}^* \end{bmatrix} \quad (2.13)$$

Nilai arus referensi dari hasil perhitungan tersebut kemudian akan di olah oleh sistem kendali pada filter aktif.

### 2.3.2 Transformasi Clarke

Setelah mendapat nilai arus referensi pada koordinat abc, selanjutnya di gunakan teori transformasi clarke yaitu mengubah koordinat abc arus dan tegangan pada sistem tiga fasa menjadi koordinat  $\alpha\beta 0$  seperti pada Gambar 2.9.



Gambar 2.9 Transformasi Sistem abc menjadi  $\alpha\beta 0$

(Talagalamani & P, 2017)

Rumus untuk transformasi clarke seperti pada persamaan (2.14):

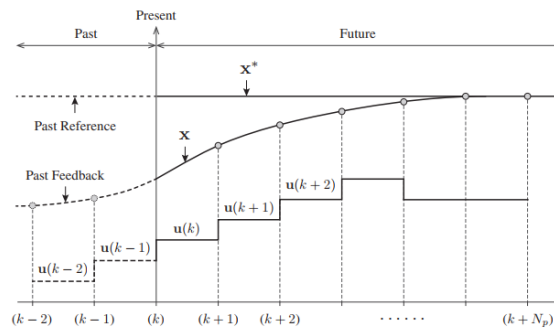
$$\begin{bmatrix} x_0 \\ x_\alpha \\ x_\beta \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} \quad (2.14)$$

Dengan “x” untuk nilai tegangan ataupun nilai arus. Pada penelitian ini, teori transformasi clarke digunakan dalam perhitungan MPC dalam menghasilkan switching.

## 2.4 Model Predictive Control (MPC)

*Model Predictive Control* (MPC) merupakan teknik pengendalian yang menggunakan model proses pada suatu sistem untuk memprediksi perilaku masa depan dari variabel yang dikendalikan. MPC menggunakan pendekatan diskrit

untuk optimalisasi dengan cara meminimalisasi nilai *cost function* (Rodriguez & Cortes, 2012).



Gambar 2.10 Prinsip Kerja *Model Predictive Control*

(Yaramasu & Wu, 2017)

Prinsip kerja MPC ditunjukkan pada Gambar 2.10. Lintasan referensi kendali ( $x^*$ ) diatur konstan untuk menyederhanakan analisis. MPC dirumuskan dalam diskrit dengan memungkinkan variabel mengubah nilainya sendiri ketika berjalan tiap langkah sampling. Jika dibandingkan dengan kendali histeresis klasik dan teknik kendali linier, yang hanya menggunakan nilai variabel masa lalu (*past*) dan sekarang (*present*) sebagai acuan kendalinya, MPC menggunakan nilai variabel masa lalu (*past*), sekarang (*present*), dan masa depan (*future*) sebagai acuan kendalinya. Dengan kata lain, MPC memprediksi kesalahan di masa depan dan mengambil tindakan pengendalian preventif sehingga sistem tidak akan mengalami *error* yang besar, sehingga membuat sistem secara keseluruhan menjadi kokoh (Yaramasu & Wu, 2017).

#### 2.4.1 Model Waktu Diskrit

Dalam metodenya, MPC menggunakan model waktu diskrit yang digunakan untuk prediksi yang dapat dinyatakan sebagai model ruang keadaan (*state space*) seperti pada persamaan (2.15) dan (2.16):

$$x(k + 1) = Ax(k) + Bu(k) \quad (2.15)$$

$$y(k) = Cx(k) + Du(k) \quad (2.16)$$

Pemodelan proses diskritisasi pada MPC terjadi pada perhitungan arus beban dinamis yang dapat digambarkan dengan persamaan diferensial vektor pada persamaan (2.17)

$$v = Ri + L \frac{di}{dt} + e \quad (2.17)$$

Dimana  $v$  adalah vektor tegangan yang dihasilkan oleh inverter,  $R$  adalah resistansi,  $i$  adalah vektor arus,  $L$  induktansi dan  $e$  adalah vektor *back-emf*.

Model waktu diskrit pada persamaan (2.17) merupakan model prediktif untuk periode waktu sampling  $T_s$  tertentu. Model waktu diskrit ini nantinya akan digunakan untuk memprediksi nilai arus beban masa depan dengan mempertimbangkan vektor tegangan dan arus beban terukur pada saat pengambilan sampel  $k$ . Dari persamaan diatas, turunan arus beban  $di/dt$  dapat direpresentasikan dengan pendekatan forward Euler sebagai persamaan (2.18):

$$\frac{di}{dt} \approx \frac{i(k + 1) - i(k)}{T_s} \quad (2.18)$$

Jika disubstitusikan kedalam persamaan (2.17) maka akan mendapatkan persamaan (2.19) prediksi arus beban pada waktu  $k+1$ , untuk setiap nilai vektor tegangan  $v(k)$  yang dihasilkan oleh inverter.

$$i^p(k + 1) = \left(1 - \frac{RT_s}{L}\right) i(k) + \frac{T_s}{L} (v(k) - e(k)) \quad (2.19)$$

Dimana  $e(k)$  menunjukkan estimasi back-emf yang ditulis dengan persamaan (2.20).

$$e(k) = v(k-1) - \frac{L}{T_s} i(k) - \left(R - \frac{L}{T_s}\right) i(k-1) \quad (2.20)$$

### 2.4.2 Cost Function

Kunci dari model kendali prediktif ini terletak pada *cost function* ( $g$ ) untuk dapat memilih keadaan *switching* yang optimal dengan nilai *error cost function* terkecil.

Pada penelitian ini, parameter yang digunakan pada persamaan *cost function* adalah parameter arus, seperti pada persamaan (2.21) berikut :

$$g = |i_{\alpha}^* - i_{\alpha}^p| + |i_{\beta}^* - i_{\beta}^p| \quad (2.21)$$

Dimana  $i_{\alpha}^*$  dan  $i_{\beta}^*$  merupakan bilangan *real* dan bilangan *imaginer* dari arus referensi, sedangkan  $i_{\alpha}^p$  dan  $i_{\beta}^p$  merupakan bilangan *real* dan bilangan *imaginer* dari arus prediksi.

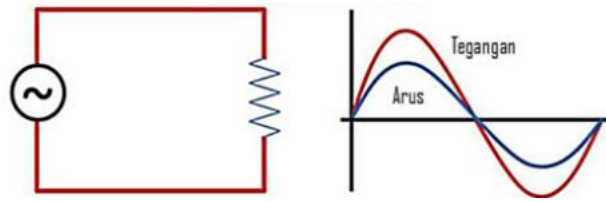
Pada penelitian ini, keluaran dari metode kendali MPC digunakan sebagai pengatur teknik pensaklaran konverter pada filter aktif.

## 2.5 Beban

### 2.5.1 Beban Resistif

Beban resistif adalah beban yang memiliki sifat resistif murni (resistor) seperti pada elemen pemanas dan lampu pijar. Resistor bersifat menghalangi aliran elektron yang melewatinya sehingga mengakibatkan terkonversinya energi listrik menjadi panas. Gelombang arus dan tegangan listrik yang melewati resistor akan selalu bersamaan saat mencapai maksimum maupun pada saat mencapai minimum, sehingga tidak mengakibatkan pergeseran fasa arus maupun tegangan listrik

jaringan. Untuk rangkaian dan bentuk gelombang dari beban resistif dapat dilihat pada Gambar 2.11.



Gambar 2.11 Rangkaian dan Bentuk Gelombang Beban Resistif

Persamaan impedansi beban resistif dapat dilihat pada persamaan (2.22):

$$Z = R \quad (2.22)$$

Dengan :

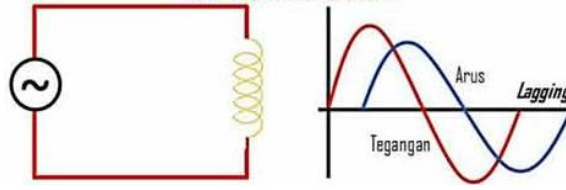
$Z$  = Impedansi ( $\Omega$ )

$R$  = Resistansi ( $\Omega$ )

### 2.5.2 Beban Induktif

Beban Induktif adalah beban yang memiliki lilitan kawat (kumparan) yang terdapat di berbagai alat-alat listrik seperti motor, trafo, dan relay. Kumparan dibutuhkan oleh alat-alat listrik tersebut untuk menciptakan medan magnet sebagai komponen kerjanya. Pembangkitan medan magnet pada kumparan inilah yang menjadi beban induktif pada rangkaian arus listrik AC. Selain itu, kumparan tersebut juga menyebabkan terhambatnya laju arus, sehingga terjadi pergeseran posisi gelombang arus menjadi tertinggal (*lagging*) sebesar  $90^\circ$  dari gelombang tegangan. Untuk rangkaian dan bentuk gelombang dari beban induktif dapat dilihat pada Gambar 2.12.





Gambar 2.12 Rangkaian dan Bentuk Gelombang Beban Induktif  
 Persamaan impedansi beban induktif dapat dilihat pada persamaan (2.23):

$$Z = X_L = \omega L \quad (2.23)$$

Dengan :

$Z$  = Impedansi ( $\Omega$ )

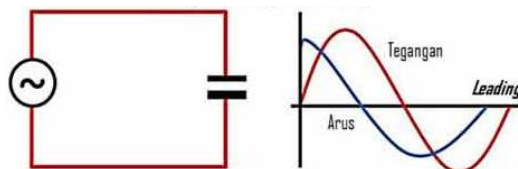
$X_L$  = Reaktansi Induktansi ( $\Omega$ )

$\omega$  = frekuensi angular ( $2\pi f$ )

$L$  = Induktansi (H)

### 2.5.3 Beban Kapasitif

Beban kapasitif merupakan kebalikan dari beban induktif. Beban kapasitif memiliki sifat menghalangi terjadinya perubahan nilai tegangan listrik. Dapat dikatakan bahwa kapasitor seakan-akan menyimpan tegangan sementara waktu. Karena hal tersebut, maka beban kapasitif dapat mengakibatkan gelombang arus akan mendahului (*Leading*) tegangannya sebesar  $90^\circ$ . Untuk rangkaian dan bentuk gelombang dari beban kapasitif dapat dilihat pada Gambar 2.11 Gambar 2.13



Gambar 2.13 Rangkaian dan Bentuk Gelombang Beban Kapasitif  
 Persamaan impedansi beban induktif dapat dilihat pada persamaan (2.24):

$$Z = X_C = \frac{1}{\omega C} \quad (2.24)$$

Dengan :

$Z$  = Impedansi ( $\Omega$ )

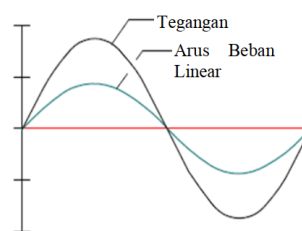
$X_C$  = Reaktansi Kapasitansi ( $\Omega$ )

$\omega$  = frekuensi angular ( $2\pi f$ )

$C$  = Kapasitansi (F)

#### 2.5.4 Beban Linier

Beban linier adalah beban dengan impedansi yang selalu konstan, sehingga arus selalu sebanding dengan tegangan setiap waktunya seperti pada Gambar 2.14. Karakteristik ini mengikuti hukum Ohm yang menyatakan bahwa arus akan berbanding lurus dengan tegangan. Apabila beban diberi tegangan sinusoidal, maka arus yang mengalir pun berbentuk sinusoidal dan tidak mengalami distorsi gelombang yang menyebabkan adanya harmonisa. Beberapa contoh beban linear adalah lampu pijar, pemanas dan beban lainnya selama tidak dipasok oleh perangkat konverter daya.

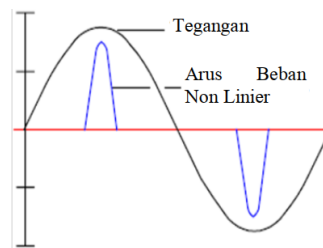


Gambar 2.14 Bentuk Gelombang Tegangan dan Arus Beban Linier

#### 2.5.5 Beban Non Linier

Beban non linier adalah kebalikan dari beban linier, dimana nilai impedansinya tidak konstan dalam setiap periode tegangan masukan. Oleh karena

itu, maka arus yang dihasilkan tidak berbanding lurus dengan tegangan yang diberikan, sehingga beban non linier tidaklah mematuhi Hukum Ohm yang menyatakan arus berbanding lurus dengan tegangan. Gelombang arus yang dihasilkan oleh beban non linier tidak sama dengan bentuk gelombang tegangan karena terjadi cacat (distorsi) seperti pada Gambar 2.15. Beban nonlinier yang umumnya merupakan peralatan elektronik yang didalamnya banyak terdapat komponen semikonduktor, dalam proses kerjanya berlaku sebagai saklar yang bekerja pada setiap siklus gelombang dari sumber tegangan. Proses kerja ini akan menghasilkan gangguan atau distorsi gelombang arus yang tidak sinusoidal.

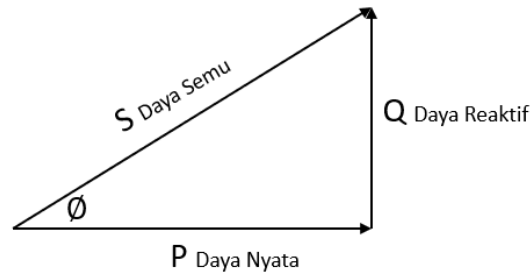


Gambar 2.15 Bentuk Gelombang Tegangan dan Arus pada Beban Non Linier

Beban non linier dapat menghasilkan gelombang-gelombang arus yang berbentuk sinusoidal berfrekuensi tinggi yang disebut dengan arus harmonisa.

## 2.6 Daya Listrik

Daya listrik adalah besarnya energi listrik yang dikonsumsi atau dihasilkan oleh suatu rangkaian setiap satuan waktu. Sumber energi seperti tegangan listrik akan menghasilkan daya listrik sedangkan beban yang terhubung dengannya akan menyerap daya listrik tersebut. Daya listrik dapat dibagi menjadi 3 yaitu daya nyata (P), daya reaktif (Q) dan daya semu (S).



Gambar 2.16 Segitiga Daya

Gambar 2.16 merupakan hubungan antara daya aktif (P), daya reaktif (Q), dan daya semu (S), yang disebut dengan segitiga daya.

### 2.6.1 Daya Aktif (P)

Secara sederhana, daya aktif adalah daya yang dibutuhkan oleh beban resistif. Daya menunjukkan aliran energi listrik dari sumber ke jaringan beban untuk diubah menjadi energi lain seperti cahaya, gerak maupun bunyi.

Persamaan dari daya aktif dapat dilihat pada persamaan (2.25):

$$P = V \times I \times \cos \varphi \quad (2.25)$$

Keterangan :

P = Daya Aktif (W)

V = Tegangan (V)

I = Arus Listrik (A)

$\cos \varphi$  = Faktor Daya

### 2.6.2 Daya Reaktif (Q)

Secara sederhana, daya reaktif adalah daya yang dibutuhkan untuk membangkitkan medan magnet pada kumparan beban induktif. Seperti pada motor induksi, medan magnet yang dihasilkan oleh daya reaktif pada kumparan stator

berfungsi untuk menginduksi rotor sehingga tercipta medan magnet induksi pada komponen rotor. Persamaan dari daya reaktif dapat dilihat pada persamaan (2.26):

$$Q = V \times I \times \sin \varphi \quad (2.26)$$

Keterangan :

Q = Daya Reaktif (VAR)

V = Tegangan (V)

I = Arus Listrik (A)

Sin  $\varphi$  = Faktor Reaktif

### 2.6.3 Daya Semu (S)

Daya Semu adalah daya yang dihasilkan oleh perkalian antara tegangan dan arus dalam suatu jaringan atau daya yang merupakan hasil penjumlahan trigonometri daya aktif dan daya reaktif. Daya semu ialah daya yang dikeluarkan sumber alternation current (AC) atau dikonsumsi oleh beban. Persamaan dari daya semu dapat dilihat pada persamaan (2.27):

$$S = V \times I \quad (2.27)$$

Keterangan :

S = Daya Semu (VA)

V = Tegangan (V)

I = Arus Listrik (A)

## 2.7 Harmonisa

Harmonisa adalah gangguan yang terjadi dalam sistem distribusi tenaga listrik yang disebabkan adanya beban nonlinier dalam jumlah besar sehingga menarik arus dalam bentuk non-sinusoidal, walaupun sebenarnya beban tersebut

disuplai dari tegangan sinusoidal. Berbeda dengan beban linier yang menarik arus sinusoidal sesuai dengan tegangan yang menyuplainya (Sibuea & Thayib, 2014). Harmonisa ini merupakan gelombang sinusoidal yang memiliki frekuensi tinggi yang sama dengan kelipatan bilangan bulat dari frekuensi fundamental.

Harmonisa muncul karena adanya beban nonlinier yang terhubung pada sistem distribusi tenaga listrik. Beban nonlinier ini umumnya adalah peralatan elektronik yang di dalamnya banyak terdapat komponen semi konduktor, yang dalam proses kerjanya berlaku sebagai saklar yang bekerja pada setiap siklus gelombang dari sumber tegangan. Beban nonlinier dalam sistem tenaga adalah semua jenis perangkat penyearah, seperti yang ditemukan di konverter daya, *uninterruptible power supply* (UPS), dan perangkat busur seperti tungku listrik dan lampu *fluorescent*.

Frekuensi fundamental pada sistem tenaga listrik di Indonesia ialah 50 Hz. Jika terdapat harmonisa ke-2 maka gelombang tersebut mempunyai frekuensi 100 Hz, harmonisa ke-3 mempunyai frekuensi 150 Hz, harmonisa ke-4 mempunyai frekuensi 200 Hz dan seterusnya sehingga menjadi persamaan (2.28):

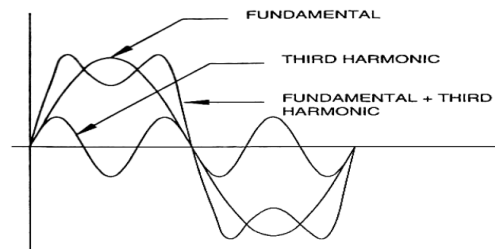
$$f_h = (h) \times f_0 \quad (2.28)$$

dengan  $f_h$  merupakan frekuensi harmonisa,  $h$  bilangan bulat dan  $f_0$  merupakan frekuensi fundamental.

Gelombang harmonisa ini kemudian menumpang pada gelombang murni sehingga menimbulkan cacat gelombang, ini merupakan penjumlahan antara gelombang fundamental dengan gelombang harmonisanya.

Pada Gambar 2.17 menunjukkan gelombang fundamental, gelombang harmonisa orde ke-3 yang mengalami tiga kali siklus penuh ( $3 \times f_0$ ) dalam satu kali

siklus gelombang fundamentalnya, serta penjumlahan gelombang fundamental dan harmonisa orde ke-3.



Gambar 2.17 Penjumlahan Gelombang Fundamental dan Harmonisa orde 3  
(Sugiarto, 2012)

Salah satu contoh dari dampak adanya harmonisa pada jaringan adalah terjadinya kesalahan pengukuran energi listrik pada kwh meter analog/digital seperti yang dilakukan oleh (Yani, 2019), dimana penelitian dilakukan dengan cara mengukur tegangan dan arus pada beban terpasang. Tegangan yang digunakan adalah sebesar 220 V, dan beban yang dipasang adalah lampu pijar (LP) 100 W dan lampu hemat energi (LHE) 20 W. Adapun hasil penelitian pada percobaan 1 LP+5LHE, maka arus masuk =  $1(0,45A) + 5(0,090A) = 0,9A$ , tetapi hasil pengukuran arus bernilai 0,95; 0,96; 0,98; 0,99; 1,01; 1,02; 1,04 yang berarti bahwa terdapat perbedaan nilai arus dan kesalahan ini dipengaruhi oleh adanya harmonisa.

### 2.7.1 Distorsi Harmonisa

Distorsi harmonisa adalah setiap perubahan dalam bentuk sinyal listrik yang tidak disengaja karena adanya faktor tertentu dan secara umum keberadaanya tidak diinginkan pada sistem (Dermawan & Rahman, 2018). Distorsi harmonisa merupakan suatu distorsi dari gelombang arus dan tegangan di jaringan yang tidak

lagi sinusoidal, hal tersebut akan menyebabkan timbulnya arus, tegangan dan daya harmonik di dalam jaringan yang terdapat beban-beban nonlinier.

Ada dua jenis distorsi harmonisa, yaitu:

1. *Individual Harmonic Distortion* (IHD)
2. *Total Harmonic Distortion* (THD)

### 2.7.1.1 *Individual Harmonic Distortion* (IHD)

IHD merupakan rasio antara nilai *root mean square* (RMS) dari harmonisa individual terhadap nilai RMS dari fundamental (Sugiarto, 2012). IHD berlaku pada arus maupun tegangan. IHD ditulis dalam persamaan (2.29).

$$IHD = \frac{I_n}{I_1} \times 100 \% \quad (2.29)$$

Keterangan:

$I_n$  adalah Arus harmonisa pada orde ke-n (A)

$I_1$  adalah Arus fundamental (A)

### 2.7.1.2 *Total Harmonic Distortion* (THD)

THD merupakan rasio antara nilai RMS dari seluruh komponen harmonisa terhadap nilai RMS dari fundamental. Hubungan antara THD dengan IHD ditulis dalam persamaan (2.30).

$$THD = \sqrt{(IHD_2^2 + IHD_3^2 + IHD_4^2 + \dots + IHD_n^2)} \quad (2.30)$$

Total distorsi harmonisa dapat dinyatakan sebagai berikut:

- THD Tegangan

THD tegangan ditulis dalam persamaan (2.31).



$$V_{THD} = \frac{\sqrt{\sum_{n=2}^{n=\infty} V_n^2}}{V_1} \times 100\% \quad (2.31)$$

Keterangan:

$V_h$  adalah tegangan harmonisa pada orde ke  $n$  (V),

$V_1$  adalah tegangan fundamental (V), dan

$n$  adalah komponen harmonisa maksimum yang diamati.

- THD Arus

THD arus ditulis dalam persamaan (2.32).

$$I_{THD} = \frac{\sqrt{\sum_{n=2}^{n=\infty} I_n^2}}{I_1} \times 100\% \quad (2.32)$$

Keterangan:

$I_n$  adalah nilai arus harmonisa pada orde ke  $n$  (A),

$I_1$  adalah nilai arus fundamental (A), dan

$n$  adalah komponen harmonisa maksimum yang diamati.

### 2.7.2 Standar Harmonisa

Standar harmonisa yang sering dipakai adalah standar dari IEEE 519-2014. Standar IEEE 519-2014 ini merupakan revisi dari standar IEEE 519-1992 (Adi Wirajaya et al., 2019). Standar ini berisi tentang rekomendasi penerapan dan persyaratan pengaturan harmonisa pada sistem tenaga listrik baik itu harmonisa tegangan ataupun harmonisa arus.

Standar IEEE 519-2014 menetapkan untuk harmonisa arus ditentukan oleh perbandingan  $I_{sc}/I_L$ .  $I_{sc}$  merupakan arus hubung singkat yang ada pada PCC (*Point of Common Coupling*), sedangkan untuk  $I_L$  merupakan arus beban fundamental. Sedangkan untuk batas harmonisa tegangan ditentukan oleh besarnya tegangan

sistem yang terpasang. Standar harmonisa yang diizinkan untuk arus berdasarkan standar IEEE 519-2014 dapat dilihat pada Tabel 2.3.

Tabel 2.3 Batas Distorsi Arus untuk Sistem dengan Rating 120V - 69kV

Sumber: IEEE 519-2014

| <i>Maximum harmonic current distortion in percent of <math>I_L</math></i> |                 |                  |                  |                  |                     |      |
|---|-----------------|------------------|------------------|------------------|---------------------|------|
| <i>Individual harmonic order (odd harmonic)</i>                           |                 |                  |                  |                  |                     |      |
| $I_{sc}/I_L$  | $3 \leq h < 11$ | $11 \leq h < 17$ | $17 \leq h < 23$ | $23 \leq h < 35$ | $35 \leq h \leq 50$ | TDD  |
| < 20  | 4.0             | 2.0              | 1.5              | 0.6              | 0.3                 | 5.0  |
| 20-50   | 7.0             | 3.5              | 2.5              | 1.0              | 0.5                 | 8.0  |
| 50-100  | 10.0            | 4.5              | 4.0              | 1.5              | 0.7                 | 12.0 |
| 100-1000  | 12.0            | 5.5              | 5.0              | 2.0              | 1.0                 | 15.0 |
| >1000   | 15.0            | 7.0              | 6.0              | 2.5              | 1.4                 | 20.0 |

Menurut (Nugroho & Reza, 2018) dan (Thamizh Thentral et al., 2021), berdasarkan dari standar IEEE 519-2014 menjelaskan bahwa batas THD arus yang baik adalah dibawah 5%.

## 2.8 *Field Programmable Gate Array (FPGA)*

*Field Programmable Gate Array (FPGA)* adalah komponen terpadu elektronika yang terbuat dari semikonduktor dan dirancang untuk dapat diprogram secara berulang-ulang oleh pengguna dalam mengaplikasikan rangkaian digital. Kata "*Field Programmable*" pada FPGA berarti perangkat ini dapat diprogram langsung di lapangan tanpa harus dibawa ke laboratorium menggunakan komputer. Sedangkan "*Gate Array*" artinya bahwa FPGA ini terdiri atas gerbang-gerbang digital dimana interkoneksi masing-masing gerbang tersebut dapat dikonfigurasi antara satu sama lainnya.

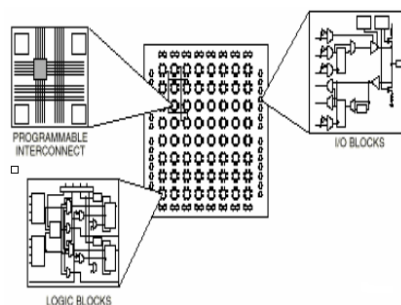
FPGA diprogram dengan menggunakan bahasa *Hardware Description Language (HDL)* yang mirip dengan yang digunakan untuk *Application-Specific*

*Integrated Circuit* (ASIC). FPGA dapat melakukan proses logika dan matematika sesuai dengan program yang ditanamkan.

FPGA mempunyai komponen gerbang terprogram (*programmable logic*) dan sambungan terprogram. Komponen gerbang terprogram yang dimiliki FPGA meliputi jenis gerbang logika biasa (AND, OR, XOR, NOT) maupun jenis fungsi matematis dan kombinatorik yang lebih kompleks (decoder, adder, subtractor, multiplier, dan lain-lain). Blok-blok komponen di dalam FPGA bisa juga mengandung elemen memori (*register*) mulai dari flip-flop sampai pada RAM (*Random Access Memory*).

Secara umum arsitektur bagian dalam dari IC FPGA terdiri atas tiga elemen utama seperti pada Gambar 2.18, dengan penjelasan sebagai berikut:

1. *Configure Logic Blocks* (CLB). Bagian yang akan memproses segala bentuk rangkaian logika yang dibuat oleh pengguna.
2. *I/O Blocks*. Sebagai interface antara *external pin* dari perangkat lain dengan *internal user logic*.
3. *Programmable Interconnect*. Bagian yang berisi *wire segments* dan *programmable switches*, selain itu bagian ini juga akan menghubungkan antara CLB satu dengan CLB lainnya.



Gambar 2.18 Arsitektur *Field Programmable Gate Array*

Saat ini perkembangan FPGA berlangsung dengan cepat dan dewasa ini terdapat bermacam-macam keluarga FPGA dengan kebutuhan perancangan dan perangkat perancangan ( *design tools* ) yang berbeda. Xilinx merupakan salah satu perusahaan yang memproduksi FPGA disamping perusahaan lain yang juga memproduksi FPGA seperti INTEL, ACTEL, PLESSEY Semiconductor, dan lain-lain.

Seperti yang telah dijelaskan sebelumnya, FPGA diprogram dengan menggunakan bahasa *Hardware Description Language* (HDL). Adapun bahasa yang populer digunakan dalam memprogram FPGA adalah Verilog dan VHSIC *Hardware Description Language* (VHDL).

### **2.8.1 Verilog**

Verilog adalah bahasa *Hardware Description Language* (HDL) yang digunakan untuk menggambarkan sistem digital seperti sakelar jaringan, mikroprosesor, memori, atau flip-flop. Artinya, dengan menggunakan HDL kita dapat menggambarkan perangkat keras digital apa pun di tingkat mana pun. Desain yang dijelaskan dalam HDL tidak tergantung pada teknologi, sangat mudah untuk mendesain dan debugging, dan biasanya lebih berguna daripada skema, terutama untuk sirkuit besar. Saat ini, verilog adalah HDL paling populer yang digunakan dan dipraktikkan di seluruh industri semikonduktor.

Semua sistem bahasa pemrograman dalam verilog dibungkus oleh suatu *module*, dimana *module* ini adalah kata kunci yang dicadangkan di verilog yang menunjukkan pembuatan blok kode dengan input dan output yang ditentukan. Sebuah *module* diapit oleh sepasang kata kunci, yaitu *module* dan *endmodule*.

Setelah kata kunci *module*, kita harus memberikan nama modul, namun tidak boleh sama dengan kata kunci (*keyword*) yang dimiliki oleh Verilog. Arah dan tipe *port* didefinisikan di dalam tanda kurung setelah deklarasi nama *module*.

Berdasarkan aturan tersebut, Gambar 2.19 adalah contoh membuat modul dengan verilog untuk memodelkan gerbang AND:

```

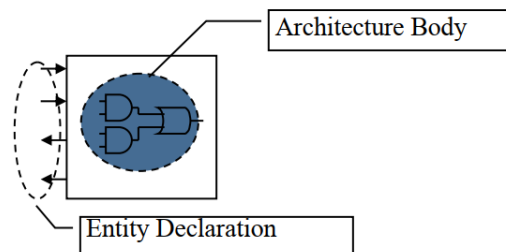
1 |
2 | module gerbangAND(input wire A,B,
3 |                   output wire Y);
4 |     assign Y = A & B;
   | endmodule

```

Gambar 2.19 Contoh Program Verilog

### 2.8.2 VHDL

VHDL merupakan kependekan dari VHSIC (*Very High Speed Integrated Circuits*) *Hardware Description Language*. VHDL dapat digunakan sebagai dokumentasi, pembuktian, dan sintesa pada sebuah perancangan digital. Secara umum struktur dari pemrograman VHDL terdiri atas dua bagian yaitu bagian *entity* dan bagian *architecture* seperti pada Gambar 2.20.



Gambar 2.20 Bagian pada VHDL

Pada bagian *entity* mendeskripsikan spesifikasi pin-pin eksternal yang digunakan dari *circuit* atau rancangan yang akan dibuat. Sedangkan pada bagian *architecture* menjelaskan atau mewakili fungsi sesungguhnya dari circuit atau

rangkaian. Untuk lebih jelasnya, dapat dilihat pada contoh program sederhana menggunakan VHDL pada Gambar 2.21.

```

library ieee;
use ieee.std_logic_1164.all;

entity example_and is
  port (
    input_1    : in  std_logic;
    input_2    : in  std_logic;
    and_result : out std_logic
  );
end example_and;

architecture rtl of example_and is
  signal and_gate : std_logic;
begin
  and_gate  <= input_1 and input_2;
  and_result <= and_gate;
end rtl;

```

Gambar 2.21 Contoh Program VHDL

Menurut (Friendly, 2017) ada tiga cara untuk mendeskripsikan circuit dalam VHDL, yaitu:

1. *Behavioral*, yaitu didesain berdasarkan algoritma.
2. *Dataflow* (RTL), yaitu didesain berdasarkan alur register data.
3. *Structural*, yaitu didesain berdasarkan perkomponen dan “merangkai komponen tersebut”.

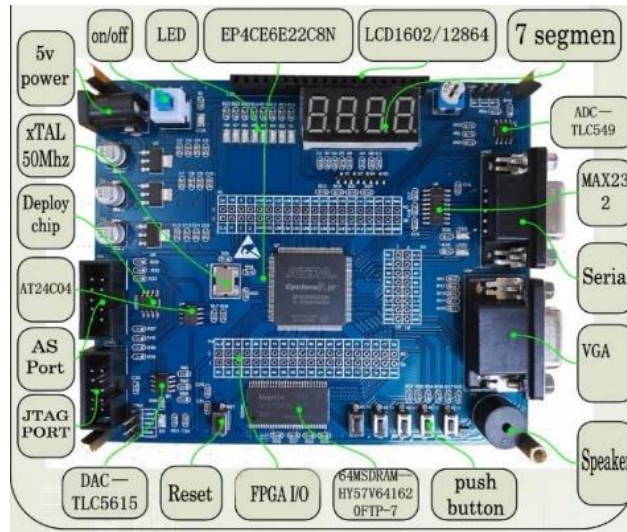
## 2.9 Altera Cyclone IV

FPGA yang digunakan dalam penelitian ini adalah development board dengan chip FPGA Altera Cyclone seri 4 EP4CE6E22C8N yang memiliki fitur antara lain:

- Menggunakan versi chip EP4CE6E22C8N,
- Clock frequency 50MHz,
- Dilengkapi dengan SDRAM sebesar 64Mbit,
- Dilengkapi dengan push button, koneksi PS2, dan koneksi VGA,

- Dilengkapi dengan port komunikasi serial RS-232,
- Dilengkapi dengan EEPROM AT24C08,
- Remote untuk eksperimen wireless berbasis infra merah,

Selain dari fitur yang telah disebutkan diatas, adapun fitur lain dari FPGA Altera Cyclone IV seperti pada Gambar 2.22.



Gambar 2.22 Blok Detail FPGA Altera Cyclone IV

## 2.10 MATLAB dan Simulink

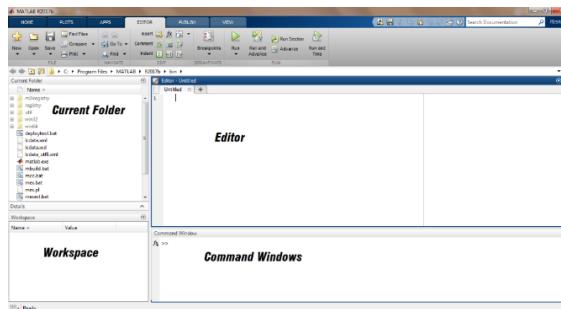
### 2.10.1 MATLAB

MATLAB atau Matrix Laboratory merupakan sebuah platform pemrograman yang dikembangkan oleh perusahaan MathWorks Inc. MATLAB digunakan untuk pemrograman berbasis matriks (Tjolleng, 2017).

MATLAB banyak digunakan pada :

- Analisis data
- Matematika dan Komputasi
- Pengembangan dan Algoritma
- Pemrograman modeling, simulasi, dan pembuatan prototipe

- Pengembangan aplikasi teknik



Gambar 2.23 Tampilan Utama MATLAB

Gambar 2.23 merupakan tampilan utama pada saat membuka aplikasi MATLAB. Secara umum, layar utama MATLAB beberapa bagian penting yaitu:

- *Current Directory*  
Berfungsi untuk mengakses dan menampilkan isi dari direktori kerja saat menggunakan MATLAB.
- *Editor*  
Digunakan untuk membuat M-file yang akan dipakai untuk membuat atau mengedit program yang dijalankan oleh MATLAB.
- *Command Window*  
Digunakan untuk menjalankan barisan-barisan program yang ditulis secara langsung di jendela ini melalui editor.
- *Workspace*  
Berisi informasi tentang variabel yang disimpan berdasarkan program yang dibuat.

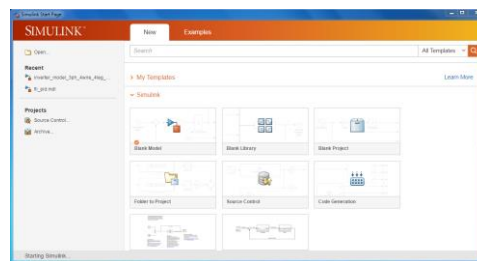
## 2.10.2 Simulink

Simulink adalah sebuah produk tambahan dari MATLAB. Simulink ini merupakan lingkungan pemrograman yang dapat digunakan untuk pemodelan,



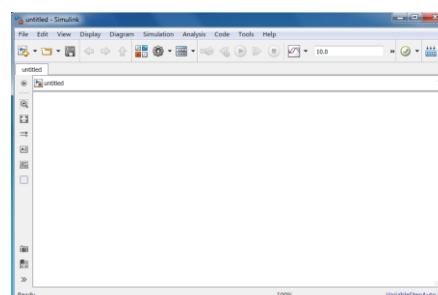
simulasi, dan analisis sistem dinamis menggunakan antarmuka grafis. Dalam Simulink, sistem digambar di layar sebagai diagram blok. Banyak elemen diagram blok tersedia, seperti fungsi transfer, penjumlahan persimpangan, serta perangkat *input* dan *output* virtual seperti generator fungsi dan osiloskop.

Untuk membuka jendela simulink dapat dengan cara mengklik ikon dengan nama simulink dibagian *menu bar Home* atau bisa juga menyetik “simulink” pada *command window*, maka akan muncul jendela awal simulink seperti pada Gambar 2.24.



Gambar 2.24 Jendela Awal Simulink

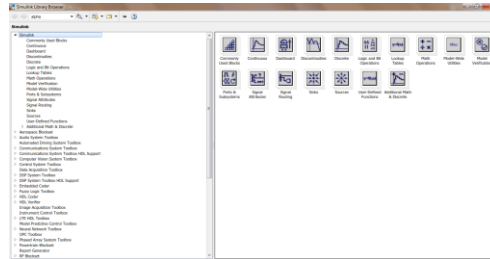
Terdapat beberapa pilihan templates yang ada di simulink. Klik "Open" untuk membuka file yang telah dibuat dan jika ingin membuat model sistem baru dari awal dapat dilakukan dengan cara memilih “Blank Model” pada jendela, maka akan muncul tampilan seperti pada Gambar 2.25.



Gambar 2.25 Tampilan Kerja Simulink

Untuk membuat pemodelan dengan Simulink, tentu perlu adanya blok komponen, maka klik ikon *library browser* pada menu bar Simulink maka akan

muncul tampilan seperti pada Gambar 2.26. Pada gambar tersebut, sisi kiri dikenal dengan *tree pane*, untuk menampilkan semua *library* Simulink yang sudah terinstall. Sisi kanannya dikenal sebagai *contents pane*, yang menampilkan blok-blok yang ada di dalam *library* yang terseleksi pada *tree pane*. Setelah tampil *library browser*, maka pilih blok komponen yang dibutuhkan.

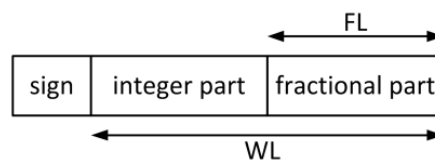


Gambar 2.26 Simulink *Library Browser*

## 2.11 *Fixed-Point*

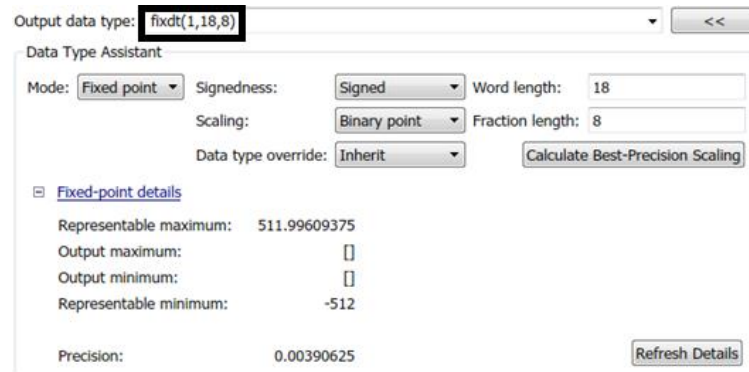
Dalam perangkat keras digital, angka direpresentasikan sebagai tipe data *fixed point*. Dalam pemrograman FPGA, pengaturan nilai *fixed point* berdampak signifikan pada penggunaan sumber daya FPGA, latensi komputasi, serta presisi data (Budiskj, 2022).

Ada tiga bagian dalam representasi *fixed point* yaitu bidang *sign* yang hanya terdiri dari dua bilangan yaitu nol untuk menandakan angka tersebut positif dan satu untuk menandakan angka tersebut negatif, bidang bilangan bulat pada bagian *Word Length (WL)*, dan bidang pecahan yang posisi angkanya berada setelah titik desimal pada bagian *Fraction Length (FL)* seperti pada Gambar 2.27.



Gambar 2.27 Struktur *Fixed Point*

(Kovacs et al., 2017)

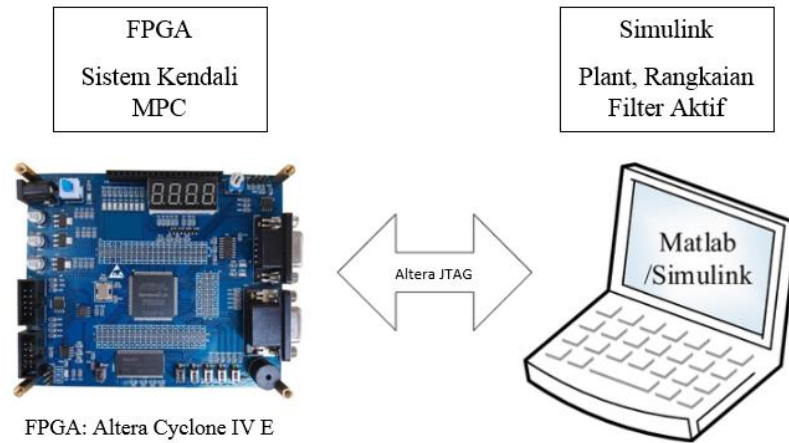


Gambar 2.28 Contoh Pengaturan *Fixed Point* pada Blok Simulink

Pada Gambar 2.28 menunjukkan contoh pengaturan nilai *fixed point* pada salah satu blok yang digunakan dalam simulink. Angka “1” pada bagian yang ditandai merepresentasikan nilai negatif, angka “18” merepresentasikan bagian *word length* yang dapat menampung nilai dari -512 sampai 511.99609375, dan angka “8” merepresentasikan *fraction length* yang digunakan untuk mengatur presisi pada pecahan, semakin besar nilai nya, maka akan semakin bagus presisi nya.

## 2.12 *FPGA in the Loop (FIL)*

*FPGA in the Loop* adalah fitur yang digunakan untuk membuat simulasi pada Simulink yang melibatkan hardware FPGA secara langsung dan tersinkron. Dengan menggunakan metode FIL model kendali pada *plant* yang dibuat dalam Simulink dapat dihubungkan dengan board FPGA. Keluaran sinyal yang dihasilkan oleh FPGA dapat dilihat melalui Simulink secara realtime. Subsistem pengontrol pada Simulink akan diganti dengan blok khusus yang dapat di upload dan dijalankan oleh papan FPGA. Dalam hal ini, FPGA terhubung ke PC melalui kabel *Joint Test Action Group (JTAG)* (Karimi et al., 2008). *FPGA in the Loop* dapat diilustrasikan pada Gambar 2.29.



Gambar 2.29 Diagram Blok *FPGA in the Loop*

### 2.13 Penelitian Terkait

Penelitian tentang perancangan kendali konverter untuk filter aktif telah banyak dilakukan dan menghasilkan berbagai jurnal sehingga dapat dijadikan sebagai referensi. Maka dari itu, beberapa penelitian terkait yang sebelumnya telah dilakukan ditunjukkan pada Tabel 2.4.

Tabel 2.4 Jurnal Terkait Penelitian

| No | Judul Jurnal                                   | Peneliti, Tahun              | Pembahasan Jurnal  |
|----|--|------------------------------|--|
| 1  | FPGA Controlled 3phase Voltage Source Inverter | (A. A. Gautam & Laxmi, 2020) | Jurnal ini meneliti tentang implementasi dari FPGA Xilinx Spartan untuk mengendalikan inverter sumber tegangan 3-fasa. Implementasi tersebut dilakukan langsung kedalam perangkat keras. Akan tetapi dalam jurnal ini tidak dijelaskan teknik pengendalian yang digunakan. |
| 2  | <i>Active Filter Based on Four-</i>            | (El-Sotouhy et al., 2019)    | Jurnal ini meneliti tentang simulasi filter daya aktif berdasarkan inverter 4-kaki untuk aplikasi 4-kawat.   |

|   |   |                            |  |
|---|---|----------------------------|--|
|   | <i>Leg Inverter and PQ Theory</i>   |                            | Sistem ini mampu menyaring arus harmonisa dari beban nonlinier. Teori pq digunakan untuk mengkompensasi komponen arus yang tidak diinginkan pada sumber suplai. Jurnal ini juga melakukan perbandingan antara teknik <i>Hysteresis Current Controller</i> dan <i>Three-Dimensional Space Vector Modulation</i> .   |
| 3 | <i>Implementation Of Finite Set Model Predictive Current Control for Shunt Active Filter</i>            | (Cherif et al., 2018)      | Jurnal ini meneliti tentang simulasi dan validasi keefektifan metode <i>Finite Set Model Predictive Current Control</i> (FS-MPCC) yang diterapkan pada Shunt Active Filter secara langsung dengan menggunakan <i>Multivariabel Filter</i> (MVF) untuk mengekstrak komponen dasar arus dan tegangan jaringan dalam menghasilkan arus referensi.   |
| 4 | <i>HIL Co-Simulation of Finite Set-Model Predictive Control Using FPGA for a Three-Phase VSI System</i> | (V. K. Singh et al., 2018) | Jurnal ini meneliti tentang implementasi dari metode kendali <i>Finite Set-Model Predictive Control</i> (FS-MPC) untuk kontrol arus sisi beban dari sistem VSI tiga fasa yang dimodelkan menggunakan MATLAB / Simulink dan Xilinx <i>System Generator</i> (XSG), dan diuji melalui simulasi <i>Hardwae-in-the-Loop</i> (HIL) menggunakan FPGA <i>Zed-board Xilinx Zynq-7000 ARM</i> untuk konversi daya terkontrol melalui |

|   |  |                         |  |
|---|--|-------------------------|--|
|   |  |                         | sistem VSI tiga fase yang diimplementasikan dengan kondisi beban motor.  |
| 5 | <i>Shunt Active Filter for Harmonic Compensation Using Fuzzy Logic Technique</i>   | (Chavan et al., 2018)   | Jurnal ini meneliti tentang cara untuk menghilangkan harmonisa arus yang disebabkan oleh beban nonlinier dengan menggunakan filter aktif <i>shunt</i> . Jurnal ini menggunakan teori arus aktif sesaat dan arus reaktif sesaat ( $i_d$ dan $i_q$ ) dalam mengekstraksi arus referensi serta kendali filter aktif nya menggunakan kendali logika fuzzy. |
| 6 | <i>Finite Control Set Model Predictive Control for Dynamic Reactive Power Compensation with Hybrid Active Power Filter</i>   | (Ferreira et al., 2018) | Jurnal ini meneliti tentang penggunaan metode kendali <i>Finite Control Set - Model Predictive Control</i> (FCS-MPC) untuk kompensasi daya reaktif dinamis menggunakan filter aktif <i>hybrid</i> . Penelitian dilakukan berupa simulasi model pada MATLAB/Simulink.   |
| 7 | <i>Automatic Optimization and Control of Power Factor, Reactive Power and Reduction of THD for Linear and Nonlinear Load</i> | (Jarad et al., 2018)    | Jurnal ini meneliti tentang cara untuk melakukan perbaikan kualitas daya mulai dari faktor daya, daya reaktif, dan pengurangan THD dengan menggunakan Arduino UNO. Arduino dibuat sebagai sistem untuk mendeteksi kondisi faktor daya <i>leading/lagging</i> secara otomatis. Peneliti menambahkan kapasitor   |

|   |   |                                  |   |
|---|---|----------------------------------|---|
|   | <i>by using Arduino Uno</i>   |                                  | secara seri untuk perbaikan faktor daya dan filter aktif shunt untuk mengurangi THD dengan hasil yang memuaskan pada berbagai kondisi beban.  |
| 8 | <i>Fundamental Refence Signal Generation for Shunt Active Filter for Harmonic and Reactive Power Mitigation using Xilinx System Generator</i> | (Sriranjani & Jayalalitha, 2017) | Jurnal ini meneliti tentang membuat filter aktif shunt untuk mereduksi harmonisa dan daya reaktif dengan menggunakan metode SRF untuk menghasilkan arus referensinya, dan kendali konverter yang digunakan adalah kendali histeresis yang dibuat kedalam model di MATLAB Xilinx. Model yang telah di buat tersebut diimplementasikan langsung kedalam FPGA Xilinx Spartan 3A, dengan hasil yang ditampilkan kedalam Precision Power Analyzer WT3000 |
| 9 | <i>FPGA Implementation of Open-Loop Controller for Five-Level Three Phase Modular Multilevel Converter</i>                                    | (Fatunmbi et al., 2017)          | Jurnal ini meneliti tentang penggunaan konsep Hardware-in-the-Loop yang dilakukan pada simulasi <i>modular multilevel converter</i> (MMC). MMC disimulasikan didalam MATLAB/Simulink. Kendali diimplementasikan pada FPGA Altera DE2 yang menghasilkan fungsi switching untuk kontrol MMC, sehingga hasil yang ingin diperoleh di dapat secara real-time dan praktis.   |

|    |  |                        |   |
|----|--|------------------------|---|
| 10 | <i>Active Power Filter For Power Quality Improvement</i> | (Saxena & Gupta, 2016) | Jurnal ini meneliti tentang simulasi <i>Shunt Active Power Filter</i> menggunakan inverter 3-fasa 3-kawat untuk memperbaiki kualitas daya dengan cara mereduksi harmonisa dan daya reaktif yang dihasilkan oleh beban nonlinier. Teknik pengendalian yang digunakan adalah teknik kendali PI konvensional dan logika fuzzy. |
|----|--|------------------------|---|

Berdasarkan pembahasan dari jurnal penelitian yang dilakukan sebelumnya pada Tabel 2.4, masing-masing mempunyai cara kerja dan metode yang dipakai yang berbeda-beda, maka dari itu dari jurnal hasil penelitian tersebut dijadikan sebagai referensi pada penelitian ini. Yang berbeda dari penelitian yang dilakukan adalah, pada penelitian ini berfokus pada pengendalian konverter yang digunakan sebagai filter aktif guna mereduksi harmonisa arus sehingga dapat memperbaiki kualitas daya listrik. Metode kendali yang dipakai untuk pensaklaran pada konverter adalah kendali *Model Predictive Control* (MPC) yang diimplementasikan ke dalam perangkat FPGA, MPC dipilih karena metode ini dapat memprediksi kesalahan di masa depan dan mengambil tindakan pengendalian preventif sehingga sistem tidak akan mengalami *error* yang besar. Filter aktif *shunt* dengan inverter 3-fasa 3-kawat digunakan dalam penelitian ini yang dibuat kedalam konsep *FPGA-in-the-Loop* agar lebih praktis dan juga sebagai contoh pengaplikasian dari kendali konverter yang diaplikasikan kedalam prosesor FPGA. FPGA yang digunakan dalam penelitian ini adalah FPGA Altera Cyclone IV dengan frekuensi *Clock* sebesar 50 MHz.