



**IMPLEMENTASI DESAIN *BUCK CONVERTER*
DENGAN *PID CONTROLLER* MENGGUNAKAN
METODE *TUNING GENETIC ALGORITHM (GA)***

SKRIPSI

**diajukan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana
Pendidikan Program Studi Pendidikan Teknik Elektro**

Oleh

Bella Risky Ananda

NIM. 5301415050

**PENDIDIKAN TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS NEGERI SEMARANG
2020**

PERSETUJUAN PEMBIMBING

Nama : Bella Risky Ananda
NIM : 5301415050
Program Studi : S-1 Pendidikan Teknik Elektro
Judul : Implementasi Desain *Buck Converter* dengan *PID Controller*
Menggunakan Metode *Tuning Genetic Algorithm (GA)*

Skripsi ini telah disetujui oleh pembimbing untuk diajukan ke sidang panitia ujian Skripsi Program Studi Pendidikan Teknik Elektro Fakultas Teknik Universitas Negeri Semarang.

Semarang, 12 Desember 2019

Pembimbing,



Ir. Ulfah Mediaty Arief, M.T, IPM

NIP. 196605051998022001

PENGESAHAN

Skripsi dengan judul “Implementasi Desain *Buck Converter* dengan *PID Controller* Menggunakan Metode *Tuning Genetic Algorithm (GA)*” telah dipertahankan di-depan sidang Panitia Ujian Skripsi Fakultas Teknik UNNES pada tanggal 27 Desember 2019.

Oleh

Nama : Bella Risky Ananda
NIM : 5301415050

Panitia

Ketua Jurusan TE

Ir. Ulfah Mediaty Arief, M.T, IPM
NIP. 196605051978022001

Sekretaris Jurusan TE

Drs. Sri Sukamta, M.Si, IPM
NIP.196505081991031003

Penguji 1

Dr. Hari Wibawanto, M.T.
NIP. 196501071991021001

Penguji 2

Aryo Baskoro Utomo, S.T., M.T.
NIP. 1984090920121002

Penguji3/ Pembimbing

Ir. Ulfah Mediaty Arief, M.T, IPM
NIP.196605051998022001

Mengetahui
Dekan Fakultas Teknik UNNES



Dr. Nur Qudus, M.T
NIP.196911301994031001

PERNYATAAN KEASLIAN

Dengan ini saya menyatakan bahwa

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Sarjana, Magister, dan /atau Doktor), baik di Universitas Negeri Semarang (UNNES) maupun di Perguruan Tinggi lain.
2. Skripsi ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan Pembimbing dan masukan Tim Penguji.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya ini, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi ini.

Semarang, 13 Desember 2019

Yang membuat pernyataan,



Bella Risky Ananda

NIM. 5301415050

MOTO DAN PERSEMBAHAN

Motto:

Menjadi orang hebat tidaklah dimulai dari pemikiran dan hasil yang besar tetapi dari hal terkecil yang dapat memberikan manfaat dan senyuman bagi orang lain.

Persembahan:

Dengan mengucapkan syukur atas limpahan rahmat dan innayah Allah SWT, saya persembahkan skripsi ini untuk:

- ❖ Orang tua tercinta, Ibu Nani Lestari dan Bapak Kustiyono atas segala doa, dukungan moral dan moril yang tak pernah lelah dan terhenti.
- ❖ Kedua saudara kandungku, Eriskha Rahesti dan Dian Rizky Aniendhita yang selalu memberikan motivasi dan semangat.
- ❖ Sahabat-sahabatku tersayang yang selalu mendengarkan keluh kesahku dan memberikan semangat.
- ❖ Teman-teman seperjuangan Pendidikan Teknik Elektro 2015, khususnya Rombel 3.

ABSTRAK

Bella Risky Ananda. 2019. Implementasi Desain *Buck Converter* dengan PID *Controller* Menggunakan Metode *Tuning Genetic Algorithm* (GA)

Perancangan *Buck Converter* dengan performa yang baik dan efisiensi daya yang tinggi terus dilakukan para peneliti sebelumnya. *Buck Converter* sebagai sistem orde dua dan *nonlinear* membutuhkan kontroler sebagai penstabil performa. Penelitian ini bertujuan mengimplementasikan rancangan DC-DC *Buck Converter* *synchronous mode Countinuous Conductor Mode* (CCM) dalam bentuk *hardware* dengan menambahkan *Proportional Integral Derivative* (PID) *Controller* sebagai penstabil tegangan *output* berdasarkan nilai *set point* yang ditetapkan. Spesifikasi rancangan *Buck Converter* bekerja pada frekuensi tinggi sebesar 25kHz, tegangan *input* 12V hingga 24 V, dan *set point* tegangan *output* sebesar 5 V diujikan dengan beban dan tegangan masukan bervariasi. *Genetic Algorithm* (GA) digunakan untuk membantu menentukan nilai *gain* atau parameter PID yang tepat agar menghasilkan performa tinggi. Proses *tuning* dengan GA menggunakan “Toolbox Optimization” yang tersedia pada aplikasi MATLAB. Implementasi *hardware Buck Converter* dengan PID *Controller* menggunakan *platform* Arduino ATmega250 dengan *compiler* pemrograman yang digunakan adalah “Simulink Support Arduino Package”, yaitu fasilitas *library* Arduino yang tersedia di Simulink MATLAB. Fasilitas tersebut digunakan sebagai konfigurasi dan komunikasikan pemrograman dengan mikrokontroler Arduino. Hasil performa terbaik *Buck Converter* menggunakan pengendalian *PI Controller* dengan nilai $K_p = 0,003$ dan $K_i = 7$. Berdasarkan pengujian variasi beban diperoleh rata-rata nilai *rise time* 0,22 detik, *settling time* 0,029 detik, *overshoot* 1,05%, dan *error steady state* 3,20% serta diperoleh efisiensi daya sebesar 82,29%. Sedangkan pengujian variasi tegangan pada kontrol terbaik *PI Controller* menghasilkan rata-rata *rise time* 0,033 detik, *settling time* 0,041 detik, *overshoot* 1,35%, *error steady state* 8,40%, dan efisiensi daya sebesar 77,35%.

Kata Kunci: *Buck Converter*, *Hardware*, *Controller*, Performa, Simulink, Efisiensi.

PRAKATA

Segala puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah melimpahkan Rahmat, Karunia, dan Innayah-Nya sehingga penulis dapat menyelesaikan Skripsi dengan lancar. Skripsi yang berjudul “Implementasi Desain *Buck Converter* dengan *PID Controller* Menggunakan Metode *Tuning Genetic Algorithm (GA)*”, disusun sebagai salah satu persyaratan meraih gelar Sarjana Pendidikan pada Program Studi S1 Pendidikan Teknik Elektro Universitas Negeri Semarang. Shalawat dan salam tak lupa dipanjatkan kepada Nabi Muhammad SAW, semoga kita semua mendapatkan syafaatnya di Yaumul Akhir nanti, Aamiin.

Penyelesaian skripsi ini tidak lepas dari bantuan berbagai pihak, oleh karena itu pada kesempatan ini penulis ingin menyampaikan ucapan terimakasih serta penghargaan kepada:

1. Bapak Kustiyono dan Ibu Nani Lestari, yang selalu memberikan dukungan dan motivasi bagi penulis serta doa yang tak henti-hentinya dipanjatkan.
2. Prof. Dr. Fathur Rokhman, M.Hum, Rektor Universitas Negeri Semarang atas kesempatan yang diberikan kepada penulis untuk menempuh studi di Universitas Negeri Semarang.
3. Dr. Nur Qudus, M.T., Dekan Fakultas Teknik, atas fasilitas yang disediakan bagi mahasiswa.
4. Ir. Ulfah Mediaty Arief, M.T., IPM, selaku Ketua Jurusan Teknik Elektro dan Koordinator Program Studi Pendidikan Elektro, sekaligus menjadi Dosen Pembimbing yang memberikan fasilitas menempuh ilmu dan dengan sabar memberikan arahan serta bimbingan bagi penulis.
5. Dr. Hari Wibawanto, M.T. dan Aryo Baskoro Utomo, S.T., M.T. selaku penguji yang telah memberikan masukan dan saran demi tercapainya hasil yang maksimal dalam karya tulis ini.
6. Semua dosen Jurusan Teknik Elektro FT UNNES yang telah memberi bekal pengetahuan yang berharga.
7. Teman-teman yang selalu memberikan semangat dan yang selalu menemani dan menyumbangkan tenaga maupun pikiran.

8. Berbagai pihak yang telah memberikan bantuan untuk karya tulis ini yang tidak dapat disebutkan satu persatu.

Penulis berharap semoga Skripsi ini memberikan manfaat untuk lembaga pendidikan dan masyarakat pada umumnya.

Semarang, 13 Desember 2019

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN KEASLIAN	iv
LEMBAR PERSEMBAHAN DAN MOTTO	v
ABSTRAK	vi
PRAKATA	vii
DAFTAR ISI.....	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiv
DAFTAR LAMPIRAN	xvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Identifikasi Masalah	5
1.3. Batasan Masalah	5
1.4. Rumusan Masalah	6
1.5. Tujuan Masalah	7
1.6. Manfaat Penelitian.....	7
1.7. Penegasan Istilah	8
BAB II KAJIAN PUSTAKA DAN LANDASAN TEORI	10
2.1. Kajian Pustaka	10
2.2. Landasan Teori	13

2.2.1. <i>Buck Converter</i>	13
2.2.2. <i>PID Controller</i>	27
2.2.3. Optimasi Sistem Kendali	31
2.2.4. <i>Genetic Algorithm (GA)</i>	32
2.2.5. Analisa Respon Transien	34
2.2.6. Komponen <i>Hardware Buck Converter</i>	36
2.2.7. MATLAB	38
2.2.8. Simulink	39
BAB III METODOLOGI PENELITIAN	41
3.1. Waktu dan Tempat Penelitian	41
3.2. Alat dan Bahan Penelitian	42
3.3. Metode Penelitian	42
3.4. Langkah Penelitian	43
3.5. Pembuatan Desain	44
3.5.1. <i>Buck Converter</i> dengan <i>PID Controller</i>	44
3.5.2. Perancangan <i>Hardware Buck Converter</i>	51
3.5.3. Implementasi <i>PID Controller</i> dengan <i>Genetic Algorithm (GA)</i>	53
3.6. Teknik Pengumpulan Data	65
3.7. Uji Respon <i>Buck Converter</i>	66
3.8. Teknik Analisa Data	67
BAB IV HASIL DAN PEMBAHASAN	68
4.1. Hasil Simulasi <i>Buck Converter</i> dengan <i>Genetic Algorithm (GA)</i>	68
4.2. Hasil Simulasi Arduino – Simulink	79

4.3. Hasil Pengujian <i>Hardware Buck Converter</i>	82
4.4. Pembahasan	90
BAB V PENUTUP	96
5.1. Kesimpulan.....	96
5.2. Saran	98
DAFTAR PUSTAKA	99
LAMPIRAN.....	102

DAFTAR TABEL

Tabel 2.1	Pengaruh Peningkatan Parameter PID	27
Tabel 3.1	Jadwal Kegiatan Penelitian	41
Tabel 3.2	Spesifikasi pada Objek Penelitian	42
Tabel 3.3	Parameter Desain <i>Buck Converter</i>	45
Tabel 3.4	Komponen <i>Hardware Buck Converter</i>	51
Tabel 3.5	Parameter GA Sebagai Proses <i>Tuning</i> PID pada <i>Buck Converter</i> ...	55
Tabel 3.6	Instrumen Hasil <i>Tuning</i> dengan GA.....	65
Tabel 3.7	Instrumentasi Pengujian Effisiensi <i>Buck Converter</i> dengan Variasi Beban.....	65
Tabel 3.8	Analisa Transien <i>Controller</i> dengan Variasi Beban	66
Tabel 3.9	Instrumentasi Pengujian Effisiensi <i>Buck Converter</i> dengan Variasi Tegangan.....	66
Tabel 3.10	Analisa Transien Kontroler dengan Variasi Tegangan	66
Table 4.1	Parameter Eksperimen yang di- <i>input</i> dalam Proses GA	70
Table 4.2	Hasil Eksperimen 1 dalam Proses <i>Tuning</i> PI dengan GA.....	73
Tabel 4.3	Hasil Eksperimen 1 dalam Proses <i>Tuning</i> PI dengan GA.....	74
Tabel 4.4	Hasil Eksperimen 1 dalam Proses <i>Tuning</i> PID dengan GA	76
Tabel 4.5	Hasil Eksperimen 2 dalam Proses <i>Tuning</i> PID dengan GA	78
Tabel 4.6	Pengujian PI <i>Controller Buck Converter</i> dengan Variasi Beban	84
Tabel 4.7	Analisis Transien PI <i>Controller</i> dengan Variasi Beban	85
Tabel 4.8	Pengujian PI <i>Controller Buck Converter</i> dengan Variasi Tegangan	86

Tabel 4.9	Analisis Transien PI <i>Controller</i> dengan Variasi Tegangan	87
Tabel 4.10	Pengujian PID <i>Controller Buck Converter</i> dengan Variasi Beban..	88
Tabel 4.11	Analisis Transien PID <i>Controller</i> dengan Variasi Beban	88
Tabel 4.12	Pengujian PID <i>Controller Buck Converter</i> dengan Variasi Tegangan	89
Tabel 4.13	Analisis Transien PID <i>Controller</i> dengan Variasi Tegangan.....	90
Tabel 4.14	Performa PI dan PID pada Simulasi <i>Tuning</i> PID dengan GA.....	91
Tabel 4.15	Performa PI dan PID <i>Controller</i> pada <i>Hardware Buck Converter</i> ..	92
Tabel 4.16	Perbandingan Penelitian Terdahulu	94

DAFTAR GAMBAR

Gambar 2.1	Rangkaian <i>Buck Converter</i> Konvensional.....	14
Gambar 2.2	Pengontrolan Lebar Pulsa <i>Duty Circle</i>	15
Gambar 2.3	Kondisi <i>Mode 1 Buck Converter</i> pada saat <i>Switch</i> Aktif ON	16
Gambar 2.4	Kondisi <i>Mode 2 Buck Converter</i> pada saat <i>Switch</i> Aktif OFF	17
Gambar 2.5	Gelombang <i>Buck Converter mode CCM</i> : (a) Arus Induktor IL; (b) Tegangan Induktor (VL); (c) Arus Masukan (Vs)	18
Gambar 2.6	Grafik Respon Arus Induktor	22
Gambar 2.7	Arus Masukan <i>Buck Converter</i>	23
Gambar 2.8	Gelombang Arus yang Melalui MOSFET	24
Gambar 2.9	Riak Tegangan Keluaran <i>Buck Converter</i>	25
Gambar 2.10	Diagram Blok Sistem Umpan Balik <i>PID Controller</i>	27
Gambar 2.11	Proses Optimasi Menggunakan Algoritma Genetika	32
Gambar 2.12	Analisis Respon Transien dan Keadaan Tunak (Stabil)	34
Gambar 2.13	Papan Mikrokontroler Arduino ATmega2560.....	36
Gambar 2.14	(a) Bentuk Fisik TLP 250. (b) Konfigurasi Pin TLP250	37
Gambar 2.15	(a) Skematik Rangkaian Sensor Tegangan dan Arduino Uno. (b) Rangkaian Pembagi tegangan pada Sensor Tegangan.....	38
Gambar 2.16	<i>Toolbox Library</i> Arduino	40
Gambar 3.1	Diagram Alir Alur Penelitian	44
Gambar 3.2	Rangkaian Kontrol <i>Buck Converter</i> TLP 250	48
Gambar 3.3	Desain Rangkaian Simulasi <i>Close Loop Buck Converter</i> dengan Kontroler PID pada Simulink-MATLAB	49

Gambar 3.4	Diagram Blok Sistem Umpan Balik <i>Buck Converter</i>	49
Gambar 3.5	Skema Kontroler PID pada Sistem <i>Buck Converter</i>	50
Gambar 3.6	Skema Perancangan <i>Hardware</i> Rangkaian <i>Buck Converter</i>	52
Gambar 3.7	Tampilan <i>Toolbox Optimization Genetic Algorithm</i>	53
Gambar 3.8	Parameter <i>Toolbox</i> Type dari Populasi.....	56
Gambar 3.9	<i>Toolbox</i> Parameter GA Ukuran Populasi	57
Gambar 3.10	<i>Toolbox</i> Parameter <i>Creation Function</i>	58
Gambar 3.11	<i>Toolbox</i> Parameter <i>Fitness Scalling</i>	59
Gambar 3.12	<i>Toolbox</i> Parameter <i>Selection</i>	60
Gambar 3.13	<i>Toolbox</i> Parameter Reproduksi	62
Gambar 3.14	<i>Toolbox</i> Parameter <i>Mutation</i>	62
Gambar 3.15	<i>Toolbox</i> Parameter <i>Crossover</i>	63
Gambar 3.16	<i>Toolbox</i> Parameter <i>Migration</i>	64
Gambar 4.1	Persebaran Pencarian Solusi Eksplorasi GA	69
Gambar 4.2	Respon <i>Buck Converter</i> tanpa <i>Controller</i>	72
Gambar 4.3	Hasil Respon PI Terbaik pada Eksperimen 1	74
Gambar 4.4	Respon Terbaik <i>Buck Converter</i> dengan PID <i>Controller</i>	77
Gambar 4.5	<i>Toolbox</i> Blok Parameter PID <i>Controller</i>	80
Gambar 4.6	Implementasi Arduino Simulink Pemrograman <i>Buck Converter</i>	80
Gambar 4.7	“S-Function Builder” untuk Mengubah Frekuensi PWM Arduino.....	81
Gambar 4.8	<i>Icon</i> Konfigurasi Simulink ke <i>Hardware</i>	82

Gambar 4.9	Pengujian <i>Hardware Buck Converter</i>	82
Gambar 4.10	Sinyal PWM dari (a) Pin PWM Arduino; (b) <i>Switching</i> MOSFET	83
Gambar 4.11	Respon <i>Buck Converter</i> pada Beban 30Ω	85
Gambar 4.12	Respon <i>Buck Converter</i> pada Tegangan 12V	86
Gambar 4.13	Respon <i>PID Controller Buck Converter</i> pada Beban 30Ω	88
Gambar 4.14	Respon <i>Buck Converter</i> pada Tegangan 12V	89
Gambar 4.15	Performa <i>Steady State</i> : (a) Variasi Beban, (b) Variasi Tegangan	92
Gambar 4.16	Analisis <i>Pole</i> pada <i>Plan Buck Converter</i>	93

DAFTAR LAMPIRAN

Lampiran 1	Usulan Pembimbing.....	102
Lampiran 2	Surat Penetapan Dosen Pembimbing Skripsi.....	103
Lampiran 3	Surat Tugas Penguji Sidang Skripsi.....	104
Lampiran 4	Berita Acara Sidang Skripsi.....	105
Lampiran 5	<i>Syntax Function Tuning Genetic Algorithm</i>	106
Lampiran 6	<i>Toolbox Optimizatiion GA</i>	107
Lampiran 7	Grafik Respon Pengujian <i>Hardware</i>	108
Lampiran 8	Grafik Respon <i>Tuning</i> dengan <i>Genetic Algorithm</i>	111
Lampiran 9	<i>Datasheet</i> Buffer 74HC541	124
Lampiran 10	<i>Datasheet</i> TLP250	125
Lampiran 11	<i>Datasheet</i> MOSFET IR250	126

BAB I

PENDAHULUAN

1.1. Latar Belakang

DC-DC *Converter* merupakan rangkaian elektronika yang mengubah sumber arus searah DC dari satu *level* tegangan ke yang lain, memberikan *output* tegangan yang diatur (J.B.Fermeiro, *et al.* 2017). Sirkuit elektronik, baik analog maupun digital membutuhkan catu daya dengan besaran berbeda (Jagdish Kurnar, *et al.* 2018), contohnya pada perangkat pendukung dalam sumber energi terbarukan dalam hal ini adalah penstabil tegangan *output* dari perangkat energi terbarukan seperti *Photovoltage* (PV). Secara global, beberapa peneliti telah membahas desain dan mengkaji DC-DC *Converter* untuk mendapatkan efisiensi daya yang tinggi. Efisiensi daya dapat dihitung sebagai rasio antara energi rata-rata dikirim ke beban dan energi rata-rata diambil dari sumber tegangan *input* (Ahmed G. Radwan, *et al.* 2018). Oleh sebab itu sistem catu daya terus dioptimalkan untuk mendapatkan kinerja yang efisien pada hasil keluaran catu daya.

Buck converter merupakan salah satu topologi DC-DC *Converter* yang menurunkan tegangan input dengan metode *regulator switching* (A.W. Cristri & Iskandar R.F, 2017). Pengoperasian *converter* secara *regulator linear* sudah mulai ditinggalkan, hal tersebut dikarenakan *regulator linear* tidaklah efektif, menurut pemaparan Sanjay Murmu dan M. B. Sharma (2015), *regulator linear* hanya mampu menghasilkan efisiensi daya keluaran sebesar 66,67% dan sisanya 33,33% terbuang menjadi panas, sedangkan *regulator switching* jauh lebih efisien

dibanding *regulator linear* yang mencapai efisiensi daya sebesar 80% hingga 95% (Sanjay Murmu dan M. B. Sharma, 2015).

Buck converter memiliki desain yang sederhana, terdiri dari komponen *switching* semikonduktor seperti MOSFET dan dioda, komponen pasif terdiri dari induktor dan kapasitor, serta resistor sebagai beban (Nilesh Shinde, *et al.* 2018). Adanya komponen *switching* membuat *Buck Converter* menjadi sistem orde dua dan menghasilkan keluaran bersifat *nonlinear* terhadap domain waktu. Untuk menghasilkan keluaran *Buck Converter* yang stabil dengan respon yang cepat, maka diperlukan kontroler untuk mengontrol peralihan *switching* MOSFET dengan menerapkan *Pulse Width Modulation* (PWM) dengan mengatur *duty cycle*. Menurut H. Abderrezek, *et al.* (2018) kontrol konvensional konverter seperti *Sliding Mode* (K. Kannabiran dan S. Alagarsamy, 2017) gagal mencapai kekokohan dalam *nonlinear*, variasi parameter, gangguan beban, variasi tegangan *input*, serta memiliki kalkulasi kontrol yang kompleks dan sulit untuk diterapkan. Desain dan implementasi sistem kontrol menuntut penggunaan teknik yang efektif yang menawarkan solusi sederhana dan pragmatis sehingga mendapatkan performa kinerja yang baik terlepas dari gangguan dan ketidakpastian sebuah sistem (Magdy Saoudi, *et al.* 2017).

Diantara kontroler *modern* yang seringkali digunakan di industri adalah *Proportional Integral Derivative* (PID) *Controller* (M. F. Aranza, *et al.* 2016). Menurut Mirzaei dan Mojallali (2015) *PID Controller* telah menjadi standar industri karena kontroler ini memiliki desain yang sederhana, memiliki performansi yang baik dan mudah untuk diimplementasikan. Performa *PID Controller*

dipengaruhi oleh penentuan nilai *gain*, yaitu *Gain Proportional* (K_p), *Gain Intergral* (K_i), dan *Gain Derivative* (K_d) yang diperoleh melalui proses *tuning*. Beberapa metode *tuning* konvensional yang sering digunakan dalam menentukan parameter PID, diantaranya: *Ziegler-Nichols* (ZN), *Gain-Phase Margin*, *Root Locus*, *Cohen-Coon*, *Gain Scheduling*, dan *Trial-and-Error* (M. F. Aranza *et al.*, 2016; Sowjanya dan Srinivas, 2015). Menurut M. F. Aranza *et al.* (2016), metode *tuning* konvensional PID tersebut dianggap kurang optimal untuk sistem yang *nonlinear* dan memiliki orde tinggi serta perhitungan menggunakan metode tersebut relatif sulit. Penentuan nilai *gain* menjadi hal yang penting dalam kinerja *PID Controller*, karena ketika nilai *gain* tidak tepat akan menghasilkan respon yang tidak optimal.

Beberapa peneliti telah mengabungkan *Evolutionary Algorithm* seperti *Genetic Algorithm* (GA) melalui kecerdasan optimasi untuk membantu menyelesaikan permasalahan *tuning PID Controller* dan memperoleh performansi terbaik pada sistem (P. Ouyang dan V. Pano, 2015). GA diperkenalkan oleh John H. Holland pada tahun 1960-an yang sebelumnya dikenal dengan "*Genetic Reproductive Plans*" (S. Sumanthi, *et al.* 2016). Peneliti mengimplementasikan GA dalam metode *tuning* PID, seperti Charanjeet Singh (2015) serta D.C. Meena dan Ambrish Devanshu (2017) mengkomparasikan GA dengan metode *tuning* konvensional. Hasil penelitian tersebut menunjukkan bahwa dengan metode GA menghasilkan performa yang lebih baik dibanding ZN. Tujuan utama GA digunakan untuk menentukan nilai *gain* PID yang terbaik melalui proses evolusi dan eksplorasi dalam metode GA.

Permasalahan untuk mendapatkan performa *Buck Converter* terbaik dengan *PID Controller* dibutuhkan nilai parameter *gain* yang tepat. Dalam penelitian ini algoritma GA diusulkan untuk membantu mendapatkan parameter *gain* PID. Perancangan dan pengujian secara komprehensif *Buck Converter* menggunakan *PID Controller* dengan metode *tuning* GA yang disimulasikan menggunakan Simulink MATLAB untuk mengetahui performansi pada sistem *Buck Converter*.

Hasil simulasi *Buck Converter* berbasis *PID Controller* diimplementasikan dalam bentuk *hardware* sesuai perancangan desain yang telah diusulkan dengan menggunakan mikrokontroler ATmega2560 pada *board* Arduino Mega2560 sebagai sistem akuisi data dan pengendali. Pemrograman pengendali *hardware* dalam *board* Arduino secara umum menggunakan pemrograman bahasa C dengan *compiler* Arduino IDE, akan tetapi penelitian ini mengusung pemrograman Arduino dengan Simulink.

Simulink merupakan *platform* didalam aplikasi MATLAB (Matrix Laboratory) yang dapat mensimulasikan sistem dinamik secara *real time* dalam bentuk diagram blok (Mhamed Fannakh, *et al.* 2018). Selain sebagai simulasi desain, perkembangannya MATLAB telah terintegrasi dengan Arduino yang dapat digunakan pemrograman kontroler dengan menggunakan permodelan Simulink, yaitu dengan menambahkan fasilitas “Simulink Support Arduino Package” dan “Matlab Support Arduino Package” sebagai jembatan komunikasi antara MATLAB dan Arduino (R.Barber, *et al.* 2013). Beberapa jurnal perancangan konverter menggunakan fasilitas ini untuk pemrograman dalam *hardware* yang

telah dirancang, seperti dalam penelitian Akarsh Sinha, *et al.* (2013) dan Mhmed Fannakh, *et al.* (2018).

1.2. Identifikasi Masalah

Berdasarkan latar belakang, adapun identifikasi masalah sebagai berikut:

1. Pentingnya peran *Buck Converter* dalam perangkat elektronik sehingga dibutuhkan performa yang efisien dan memiliki performa yang baik.
2. *Buck Converter* sebagai sistem orde dua dan *nonlinear* diperlukan *controller* untuk menjaga kestabilan dan mempercepat respon keluaran.
3. *PID Controller* menjadi kontrol yang memiliki desain dan perhitungan yang sederhana, menghasilkan performa yang baik dan mudah diimplementasikan memiliki kekurangan dalam permasalahan menentukan nilai *gain* PID.
4. Dibutuhkan metode *tuning* yang mudah dan memperoleh nilai yang tepat seperti GA.

1.3. Batasan Masalah

Berdasarkan indentifikasi masalah diatas maka penelitian ini dibatasi permasalahan dengan asumsi sebagai berikut:

1. Topologi *converter* yang digunakan adalah *Buck Converter* dengan mode *Continous Current Mode* (CCM).
2. Nilai tegangan *input* *Buck Converter* sebesar 12-24 VDC dan tegangan *output* sebesar 5 VDC dengan frekuensi *switching* tetap sebesar 25 kHz.

3. Kontroler yang digunakan *Buck Converter* adalah *Proportional Integral* (PI) *Controller* dan *Proportional Integral Derivative* (PID) *Controller*.
4. Permodelan *Buck Converter* disimulasikan menggunakan *software* MATLAB versi R2018a.
5. Proses *tuning* menggunakan metode GA yang diproses melalui “Toolbox Optimization” yang tersedia di MATLAB, dengan batasan nilai parameter yang akan dibahas dalam Bab IV.
6. Pembuatan *hardware Buck Converter* digunakan sebagai realisasi hasil perancangan menggunakan mikrokontroler ATmega 2560 pada *platform* Arduino Mega 2560.
7. Tidak menggunakan *compiler* Arduino IDE, melainkan menggunakan program Simulink yang terintegrasi dengan Arduino.
8. Pengujian performa *hardware Buck Converter* dilakukan dengan menguji menggunakan variasi beban dan variasi tegangan dengan nilai yang telah ditentukan di Bab III.
9. Analisis transien *hardware Buck Converter* dengan melihat grafik respon pada *Scope time* dalam program Simulink MATLAB.

1.4. Rumusan Masalah

Berdasarkan identifikasi masalah diatas, permasalahan yang akan dibahas dalam penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana medesain dan merancang *Buck Converter* dengan PID *Controller*?

2. Bagaimana menentukan parameter *tuning* PID dengan *Genetic Algorithm* (GA) untuk mengoptimalkan hasil respon *Buck Converter*?
3. Bagaimana mengimplentasikan dalam bentuk *hardware* rangkaian *Buck Converter* dengan *PID Controller* dengan Simulink MATLAB?
4. Bagaimana performansi *Buck Converter* pada *hardware* yang telah dibuat?

1.5. Tujuan Penelitian

Penelitian dalam skripsi ini memiliki tujuan untuk menjawab perumusan masalah yang dimiliki dalam penelitian antara lain:

1. Menghasilkan *Buck Converter* yang sesuai dengan perancangan.
2. Mendapatkan parameter *tuning* PID pada rangkaian *Buck Converter* melalui proses optimisasi dengan GA.
3. Merealisasikan *hardware Buck Converter* sesuai dengan perancangan dengan menggunakan Simulink MATLAB.
4. Mengetahui pengaruh performa kontroler terhadap efisiensi *Buck Converter*.

1.6. Manfaat Penelitian

Berdasarkan tujuan penelitian, manfaat yang diharapkan dalam penelitian ini adalah:

1. Menghasilkan permodelan simulasi dan *hardware Buck Converter* yang dengan efisiensi tinggi.

2. Hasil penelitian ini dapat memberikan pengetahuan mengenai *Buck Converter* dengan *PID Controller* beserta cara *tuning* PID yang menggunakan GA.
3. Menjadi referensi *Buck Converter* dengan kontroler PID berbasis algoritma bagi penelitian dimasa mendatang.
4. Menjadi sebuah referensi untuk mahasiswa yang akan mengerjakan penelitian dengan topik energi terbarukan khususnya mengenai *Power Electronics* seperti *Buck Converter*.

1.7. Penegasan Istilah

Agar tidak terjadi penafsiran yang berbeda pada penelitian yang berjudul “Implementasi Desain *Buck Converter* dengan *PID Controller* Menggunakan Metode *Tuning Genetic Algorithm* (GA)” memiliki definisi sebagai berikut:

1. Arti dari implementasi yaitu pelaksanaan atau penerapan. Dalam penelitian ini implementasi yang dimaksud adalah penerapan dan pelaksanaan dari desain *Buck Converter* yang direalisasikan dalam bentuk *hardware*.
2. *Buck Converter* merupakan salah satu topologi *converter* yang berfungsi sebagai penurun tegangan *input* menjadi besaran nilai tegangan yang lebih rendah. Rangkaian *Buck Converter* terdiri dari komponen: *switching aktif*, *switching passive*, induktor, kapasitor, dan resistor sebagai beban. Berdasarkan komponen *switching* yang digunakan, *Buck Converter* merupakan sistem orde dua dan *nonlinear*.

3. *PID Controller* adalah kontroler yang telah banyak diimplementasikan dalam dunia industri yang memiliki desain dan perhitungan yang sederhana, memiliki performa yang baik dalam menjaga kestabilan sistem, mempercepat respon dan mengurangi lonjakan yang dihasilkan sistem terlebih pada sistem orde tinggi dan *nonlinear*. Terdiri dari kalkulasi tiga *gain*, yaitu: *Gain Proportional* (K_p), *Gain Integral* (K_i), dan *Gain Derivative* (K_d). Nilai *gain* tersebut sangat penting untuk menentukan performansi dari sistem.
4. Metode *tuning* adalah cara dan proses yang digunakan untuk menentukan nilai *gain* PID (K_p , K_i , dan K_d).
5. *Genetic Algorithm* (GA) adalah metode *metaheuristic* yang dikembangkan berdasarkan prinsip genetika dan proses seleksi alamiah berdasarkan Teori “Evolusi Darwin”. GA digunakan sebagai metode *tuning* PID yang digunakan untuk mengeksplorasi nilai *gain* yang tepat.

BAB II

KAJIAN PUSTAKA DAN LANDASAN TEORI

Bab ini membahas mengenai kajian pustaka yang terkait dalam penelitian dan kajian teoritis dan referensi yang mendukung penelitian. Adapun bahasan dalam bab ini meliputi penelitian sebelumnya yang diulas dalam sub bab kajian pustaka serta teori tentang *Buck Converter*, *PID Controller*, *Genetic Algorithm*, pemrogramman, dan komponen inti perangkat keras (*hardware*) yang digunakan.

2.1. Kajian Pustaka

Dalam suatu penelitian, diperlukan adanya kajian dari penelitian yang sudah ada untuk dipelajari metode, cara, dan hasil penelitian yang kemudian memberikan wacana dalam bentuk perbaikan atau pembaharuan yang lebih inovatif dan mencegah plagiat. Berikut ini adalah penelitian terdahulu yang berkaitan dengan penelitian ini, antara lain:

1. Penelitian Singh Charanjeet (2015) yang berjudul “Genetic Algorithms Based PID Controller Design”. Peneliti membahas *tuning* PID menggunakan metode Ziegler-Nicholas (ZN) dan *Genetic Algorithm* (GA) yang disimulasikan dalam MATLAB 2013a. Peneliti mengungkapkan penggunaan *PID Controller* berbasis GA dapat meningkatkan kinerja indeks sistem, menghasilkan *overshoot* yang lebih kecil, dan waktu mencapai keadaan tunak lebih cepat dibanding penggunaan metode konvensional ZN.
2. Penelitian H. Kovacevic dan Z. Stojanovic (2016) yang berjudul “Buck Converter Controlled by Arduino Uno”. Hasil penelitian berupa realisasi *hardware Buck Converter* dengan *PI Controller* menggunakan Arduino Uno

dengan pemrograman menggunakan Arduino IDE. Diawali dengan permodelan sistem yang disimulasikan di Simulink MATLAB, menghasilkan respon pada simulasi sebesar *overshoot* 0,65 V, *peak time* 0,0015 detik dan *rise time* 0,009 detik.

3. Penelitian Irma Husnaini dan Krismadinata (2017) dengan judul “Komparasi Pengendali PI dan PID untuk Tegangan Keluaran Konverter Buck”. Hasil penelitian berupa perancangan dan simulasi tegangan keluaran *Buck Converter* dengan MATLAB. Hasil pengujian mengungkapkan pengendali PI mampu meminimumkan *overshoot* tetapi respon sistem menjadi lebih lambat dengan terjadinya peningkatan pada *rise time* sebesar 0,091 detik dan *settling time* 0,0515 detik. Sedangkan pengendali PID mampu memperbaiki respon transien sistem dengan menghilangkan osilasi dan meningkatkan kecepatan respon sistem akan tetapi masih menimbulkan *overshoot* sebesar 26%. Dari dua pengontrolan tersebut, pengendali PID memiliki performansi lebih baik dibandingkan pengendali PI pada pengendalian tegangan keluaran *Buck Converter* yang telah dirancang. Penentuan parameter PID dalam penelitian ini menggunakan teknik polynomial dengan menghitung frekuensi dan sudut fasa (*Bode Diagram*), perhitungan ini dirasa kurang efektif dan perhitungan kurang sulit untuk digunakan.
4. Penelitian D.C. Meena dan Ambrish Devanshu (2017) yang berjudul “Genetic Algorithm Tuned PID Controller for Process Control”. Peneliti mengkaji GA dengan metode tuning klasik yaitu ZN dalam *PID Controller*. Peneliti mengungkapkan bahwa metode penyetelan konvensional memiliki

kemampuan yang sangat terbatas yang semakin berkurang jika model matematika eksplisit tidak tersedia. Namun, keuntungan yang diperoleh dari metode klasik ini dapat digunakan sebagai nilai awal untuk metode penyetelan lanjutan. Di sini, telah ditunjukkan bahwa pengontrol PID yang disetel GA memiliki karakteristik kinerja yang jauh lebih baik dibandingkan dengan *tuning* PID secara klasik.

5. Penelitian D. Sivamani, *et al.* (2018) yang berjudul “Genetic Algorithm based PI Controller for DC-DC Converter Applied to Renewable Energy Applications”. Topologi *converter* yang digunakan dalam penelitian tersebut adalah *Buck Boost Converter* yang diterapkan pada Generator dalam *Wind Energy Conversion System* (WECS). Topologi *converter* ini menggunakan kontroler PI berbasis GA yang disimulasikan di Simulink MATLAB untuk mengatur *Pulse Width Modulation* (PWM). Berdasarkan respon sistem *converter* yang dirancang peneliti mengamati dan menganalisa nilai *rise time*, *settling time*, dan *overshoot* sistem. Hasil simulasi menunjukkan *open loop Buck Boost Converter*, *closed loop* GA berbasis PI Controller untuk pulsa PWM, berhasil meminimalkan gangguan amplitudo tinggi (*overshoot*).
6. Penelitian Mhamed Fannakh, *et al.* (2018) yang berjudul “Hardware Implementation of Fuzzy Logic MPPT in an Arduino Card Using a Simulink Support Package for PV Application”. Penelitian tersebut merancang *hardware Boost Converter* yang diterapkan pada sistem *Maximum Power Point Tracking* (MPPT) dengan menggunakan *Fuzzy Logic* untuk penstabil tegangan dan arus yang dihasilkan *Photovoltage Panel* (PV). Pemrograman menggunakan

Simulink yang ter-*support board* ARDUINO Mega 2560. Dengan frekuensi kerja PWM sebesar 3900Hz menghasilkan efisiensi *converter* sebesar 80% dengan respon *time* 0,035 detik. Adapun pengembangan yang disarankan penelitian ini adalah penelitian ke depan dapat menyajikan solusi untuk meningkatkan frekuensi PWM untuk konverter dengan nilai statis.

Dari beberapa penelitian di atas dapat ditarik kesimpulan bahwa pengendali PID dapat diterapkan dalam *Buck Converter* untuk meningkatkan efisiensi *Buck Converter*. Permasalahan *PID Controller* adalah mendapatkan nilai parameter yang tepat dalam sistem yang dirancang, sehingga diperlukan algoritma optimasi untuk membantu mendapatkan nilai parameter tersebut selain metode *tuning* konvensional. Berdasarkan kajian pustaka di atas, GA telah banyak diterapkan dalam *PID Controller* dan mendapatkan performansi yang baik. Pada penelitian ini akan mendesain, mensimulasikan, dan merealisasikan *Buck Converter* dengan *PID Controller* dengan metode *tuning* GA.

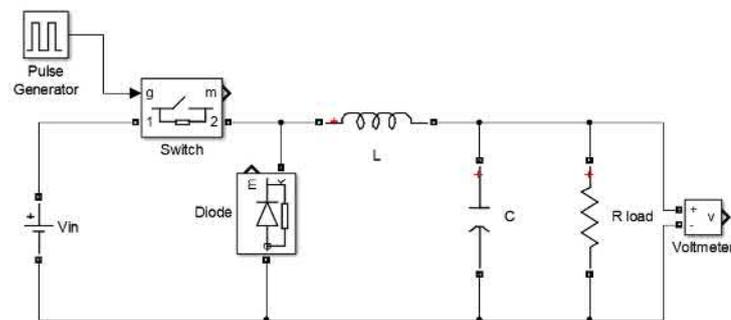
2.2. Landasan Teori

2.2.1. *Buck Converter*

Buck Converter merupakan salah satu topologi fundamental DC-DC *converter* yang memiliki fungsi menurunkan tegangan *input* (V_i) dari tegangan DC yang lebih besar (Cristri A.W & Iskandar R.F, 2017). *Buck Converter* memiliki efisiensi yang tinggi dan memiliki desain yang sederhana, yang terdiri dari komponen *switching* semikonduktor seperti MOSFET dan dioda, komponen pasif terdiri dari induktor dan kapasitor, serta resistor sebagai beban (Nilesh

Shinde, *et al.* 2018). Adanya komponen *switching* membuat *Buck Converter* menjadi sistem orde dua dan menghasilkan keluaran bersifat *nonlinear* terhadap domain waktu.

Buck Converter sebagai *regulator switching* memiliki efisiensi daya yang lebih besar, yaitu sebesar 80% hingga 95% dibandingkan dengan pengoperasian *regulator linear* yang hanya mampu menghasilkan efisiensi daya sebesar 66,67% sedangkan sisanya yaitu 33,33% terbuang menjadi panas. Gambar 2.1 merupakan rangkaian konvensional *regulator switching* dalam *Buck Converter*.



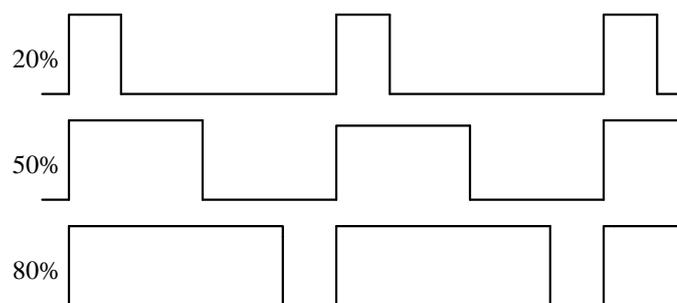
Gambar 2.1 Rangkaian *Buck Converter* Konvensional

Maksud dan fungsi setiap komponen dalam rangkaian *Buck Converter* pada gambar 2.1 adalah:

1. Sumber tegangan (V_i) merupakan sumber tegangan DC yang mensuplai rangkaian *Buck Converter*.
2. Semikonduktor, sebagai *active switch* (saklar aktif) berupa komponen semikonduktor, seperti: *Insulated Gate Bipolar Transistor* (IGBT), *Metal Oxide Semiconductor Field Effect Transistor* (MOSFET), *Thyristor*, maupun jenis semikonduktor lainnya. Saklar aktif berfungsi mencacah tegangan masukan dengan frekuensi tinggi dan mengubah tegangan masukan dengan amplitude konstan menjadi bentuk gelombang kotak. Semikonduktor

digunakan untuk modulasi sinyal *Pulse Width Modulation* (PWM) dengan periode (T) dan *duty circle* (Trzynadlowski, 2003). Semikonduktor yang digunakan dalam *switching* catu daya harus memiliki waktu *switching* yang cepat dan mampu menahan lonjakan tegangan yang dihasilkan oleh induktor.

- PWM adalah sebuah metode untuk mengontrol sirkuit elektronika daya modern yang digunakan adalah mengontrol lebar pulsa *duty circle* (D) dengan nilai amplitudo dan frekuensi *switching* yang tetap. *Duty circle* merupakan ratio dari periode (T) pada saat semikonduktor atau saklar aktif dalam kondisi *ON* untuk mencapai satu siklus periode yaitu pada saat T_{on} hingga T_{off} . Lebar pulsa PWM berbanding lurus dengan tegangan keluaran (V_{out}), semakin besar lebar pulsa maka tegangan keluaran juga semakin besar. Tampak Gambar 2.2 lebar pulsa *duty circle* yang dikontrol dengan metode PWM.



Gambar 2.2 Pengontrolan Lebar Pulsa *Duty Circle*

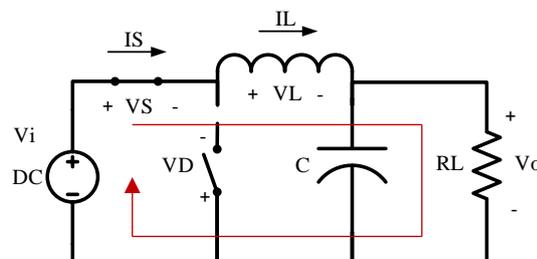
- Dioda sebagai *passive switch* (saklar pasif), merupakan *uncontrollable semiconductor switching device*, karena membuka atau menutupnya dioda tergantung dari tegangan dan arus yang mengalir pada rangkaian. Dioda akan terbuka jika dalam keadaan *forward bias*, yaitu tegangan anoda lebih tinggi daripada tegangan katoda, sebaliknya dioda akan menutup jika dalam

keadaan *reverse bias*, yaitu tegangan anoda lebih rendah dari pada tegangan katoda.

5. Induktor (L), merupakan komponen yang berfungsi untuk menyimpan arus.
6. Kapasitor (C), merupakan komponen yang berfungsi sebagai *low pass filter*.
7. Resistor (R) adalah beban sistem dan berfungsi sebagai bagian dari *low pass filter*.

Komponen semikonduktor MOSFET sebagai saklar aktif telah banyak diaplikasikan dalam topologi konverter, karena dirasa lebih efisien dibanding dengan semikonduktor lainnya. Hipotesa ini dikuatkan pada penelitian Lukman Wira Cahyadi, *et al.* (2018) hasil penelitian tersebut menunjukkan bahwa DC-DC *Buck Converter* penggunaan MOSFET sinkron nilai efisiensi cenderung stabil pada lebar pulsa *duty circle* berapapun dibanding tanpa penggunaan MOSFET.

Buck Converter menurut kinerjanya secara dinamik terbagi menjadi dua mode (Ahmed G. Radwan, *et al.* 2018), yaitu *Countinous Conductor Mode* (CCM) dan *Discontionus Conductor Mode* (DCM). Prinsip kerja *Buck Converter* dengan komponen yang diasumsikan ideal dan konverter beroperasi dalam kondisi CCM dapat dibagi menjadi 2 mode operasi yaitu mode 1 ketika saklar aktif *ON* dan mode 2 ketika saklar aktif *OFF*.



Gambar 2.3 Kondisi Mode 1 *Buck Converter* Saklar Aktif *ON*

Mode 1 yang ditunjukkan pada Gambar 2.3, ketika saklar aktif *ON* (tertutup) dan dioda *OFF* (terbuka). Arus akan mengalir dari sumber menuju ke induktor (L) sehingga terjadi pengisian arus pada induktor sedangkan ditapis dengan kapasitor (C), kemudian melewati beban (R) dan kembali ke sumber. Berdasarkan aliran arus pada Gambar 2.2 diperoleh $V_L = V_i - V_o$ sehingga diperoleh persamaan 2.1 dan 2.2.

$$\frac{\partial i_L}{\partial t} = \frac{V_i}{L} - I_L \frac{1}{L} \left(RL + \frac{R \times Rc}{R+Rc} \right) - V_C \frac{1}{L} \left(1 - \frac{Rc}{R+Rc} \right) \quad (2.1)$$

$$\frac{\partial v_C}{\partial t} = I_L \frac{1}{C} \left(\frac{R}{R+Rc} \right) - V_C \frac{1}{L} \left(\frac{1}{R+Rc} \right) \quad (2.2)$$

Keterangan:

$\frac{\partial i_L}{\partial t}$ = arus induktor saat t mendekati 0 dan t mendekati tak terhingga (A)

$\frac{\partial v_C}{\partial t}$ = tegangan kapasitor saat t mendekati 0 dan t mendekati tak terhingga (V)

V_i = tegangan *input* (V)

V_L = tegangan induktor (V)

V_C = tegangan kapasitor (V)

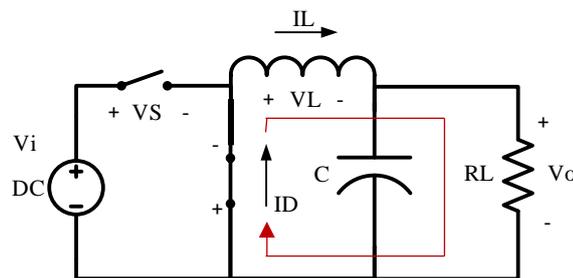
I_L = arus induktor (A)

L = nilai induktansi (H)

R = nilai resistansi sebagai beban (Ω)

Rc = rugi-rugi resistansi kapasitor diabaikan atau sama dengan nol (Ω)

C = nilai kapasitansi (F)



Gambar 2.4 Kondisi Mode 2 *Buck Converter* Saklar Aktif *OFF*

Sedangkan pada saat mode 2, yaitu saklar aktif *OFF* (terbuka) tampak pada Gambar 2.4 menyebabkan dioda dalam keadaan bias maju, arus yang disimpan induktor dikeluarkan menuju beban lalu ke dioda *freewheeling* dan kembali lagi

ke induktor. Dalam kondisi diperoleh tegangan induktor $V_L = -V_o$, harga negatif disebabkan karena arus induktor I_L berkurang. Persamaan 2.3 dan 2.4 merupakan persamaan matematis kondisi arus pada komponen induktor dan kapasitor. Sinyal arus induktor pada saat mode 1 dan mode 2 tampak pada Gambar 2.5.

$$\frac{\partial i_L}{\partial t} = -I_L \frac{1}{L} \left(RL + \frac{R \times R_C}{R+R_C} \right) - V_C \frac{1}{L} \left(1 - \frac{R_C}{R+R_C} \right) \quad (2.3)$$

$$\frac{\partial v_C}{\partial t} = I_L \frac{1}{C} \left(\frac{R}{R+R_C} \right) - V_C \frac{1}{C} \left(\frac{1}{R+R_C} \right) \quad (2.4)$$

Keterangan:

$\frac{\partial i_L}{\partial t}$ = arus induktor saat t mendekati 0 dan t mendekati tak terhingga (A)

$\frac{\partial v_C}{\partial t}$ = tegangan kapasitor saat t mendekati 0 dan t mendekati tak terhingga (V)

I_L = arus induktor (A)

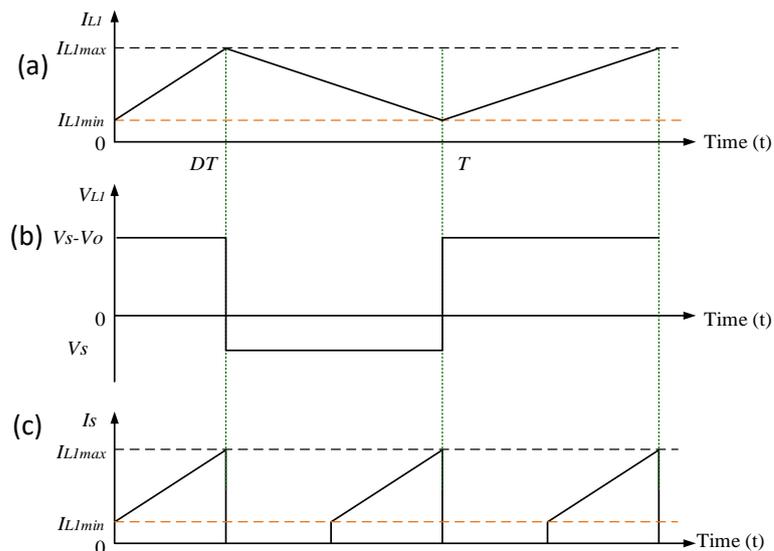
V_C = tegangan kapasitor (V)

L = nilai induktansi (H)

R = nilai resistansi sebagai beban (Ω)

R_C = rugi-rugi resistansi kapasitor diabaikan atau sama dengan nol (Ω)

C = nilai kapasitansi (F)



Gambar 2.5 Gelombang *Buck Converter* mode CCM: (a) Arus Induktor I_L ; (b) Tegangan Induktor (V_L); (c) Tegangan Input ($V_S=V_{in}$)

Sumber: (A.M Trzynadlowski, 2003, p.193)

Energi yang tersimpan dalam L meningkat selama waktu (t) pada mode saklar aktif *OFF* dan I_L menurun selama *off-state* pada mode saklar aktif *ON*. Perubahan i_L dapat dihitung dengan persamaan 2.5.

$$-V_L = L \frac{\partial i_L}{\partial t} \quad (2.5)$$

Peningkatan arus induktor pada perubahan waktu (t) *ON* dan *OFF* ketika saklar aktif *ON* diperoleh persamaan 2.6.

$$\Delta I_{Lon} = \int_{t_{off}}^{t_{on}} \frac{V_L}{L} dt = \frac{(V_i - V_o)}{L} t_{on} \quad (2.6)$$

Dengan nilai $t_{on} = D \times T$

Sebaliknya ketika saklar aktif *OFF*, karena terjadi penyimpanan energi induktif, I_L terus mengalir seperti tampak pada Gambar 2.5 (b) bentuk gelombang arus DC yang beroperasi mengulang dari satu periode waktu ke periode berikutnya, diperoleh persamaan 2.7.

$$\Delta I_{Loft} = \int_{t_{on}}^{T=t_{on}+t_{off}} \frac{V_L}{L} dt = -\frac{V_o}{L} t_{off} \quad (2.7)$$

Dengan nilai $t_{off} = (1 - D) \times T$

Keterangan:

$\frac{\partial i_L}{\partial t}$ = arus induktor saat t mendekati 0 dan t mendekati tak terhingga (A)

ΔI_{Lon} = arus induktor pada saat saklar aktif *ON* (A)

ΔI_{Loft} = arus induktor pada saat saklar aktif *OFF* (A)

L = nilai induktansi (H: Henry)

V_L = tegangan induktor (V)

$-V_L$ = polaritas negatif tegangan induktor (V)

V_i = tegangan *input* (V)

V_o = tegangan *output* (V)

t_{on} = waktu saklar aktif *ON* (detik)

t_{off} = waktu saklar aktif *OFF* (detik)

D = *duty circle*

T = periode (detik)

Diasumsikan konverter beroperasi dalam kondisi ideal energi yang tersimpan disetiap komponen pada akhir siklus 1 periode nilai energi akan sama dengan awal siklus, yang berarti bahwa i_L bernilai sama pada waktu $t = 0$ dan pada waktu $t = T$, sehingga menghasilkan persamaan 2.8.

$$\Delta I_{Lon} + \Delta I_{LoFF} = 0 \quad (2.8)$$

Dengan mensubsitusikan nilai ΔI_{Lon} dan ΔI_{LoFF} diperoleh persamaan 2.9.

$$(V_i - V_o)DT - V_o(1 - D)T = 0 \quad (2.9)$$

$$DV_i - V_o = 0 \quad (2.10)$$

$$\text{Berdasarkan persamaan 2.10, diperoleh } D = \frac{V_o}{V_i} \quad (2.11)$$

Keterangan:

ΔI_{Lon} = arus induktor pada saat saklar aktif *ON* (A)

ΔI_{LoFF} = arus induktor pada saat saklar aktif *OFF* (A)

V_i = tegangan *input* (V)

V_o = tegangan *output* (V)

D = *duty circle*

T = periode (detik)

Dari persamaan 2.11, tegangan *ouput* bervariasi secara linier berdasarkan besarnya nilai *duty circle* yang diberikan. Nilai *duty circle* adalah perbandingan t_{on} dan periode (T), sehingga nilainya tidak akan lebih dari 1. Hal tersebut menghasilkan nilai $V_o \leq V_i$, dan disebut dengan konverter *step down* atau disebut *Buck Converter*.

1. Kalkulasi Nilai Induktor

Diperoleh persamaan 2.12 diferensial yang disederhanakan berdasarkan arus melalui induktor ketika saklar aktif *ON*, dengan asumsi pada kondisi ideal.

$$\frac{i_L(t)}{dt} = V_d - V_o \quad (2.12)$$

Diasumsikan rangkaian dalam keadaan tunak maka memungkinkan adanya arus yang sangat kecil dalam induktor (I_{Lmin}) pada peralihan *ON* dan *OFF* dalam saklar aktif. Oleh karena itu nilai interval waktu $0 \leq t \leq T_{on} = DT$, menghasilkan persamaan 2.13.

$$i_L(t) = \frac{V_d - V_o}{L} t + I_{Lmin} \quad (2.13)$$

Arus induktor meningkat secara linier seiring waktu dan mencapai nilai maksimumnya I_{Lmax} dengan $t \rightarrow T_{on} = DT$

$$I_{Lmax} = \frac{V_d - V_o}{L} DT + I_{Lmin} \quad (2.14)$$

Mendefinisikan perubahan arus dari nilai minimum ke maksimum sebagai arus puncak ke puncak riak ΔI_{Lmin} , persamaan 2.11-2.14 menghasilkan ekspresi untuk ΔI_L pada persamaan 2.15.

$$\Delta I_{Lmax} = I_{Lmax} - I_{Lmin} = \frac{V_d - V_o}{L} DT \quad (2.15)$$

Arus induktor melalui MOSFET sisi bawah ketika saklar aktif *OFF* diperoleh $0 \leq t \leq T_{off}$ menghasilkan persamaan 2.16.

$$L \frac{di_L(t)}{dt} = -V_o \quad (2.16)$$

Keterangan:

$\frac{i_L(t)}{dt}$ = arus induktor saat t mendekati 0 dan t mendekati tak terhingga (A)

V_d = tegangan pada dioda (V)

V_o = tegangan *output* (V)

$i_L(t)$ = arus induktor dalam satuan waktu tertentu (A)

I_{Lmin} = arus terkecil yang melewati induktor (A)

I_{Lmax} = arus induktor maksimum (A)

L = nilai induktansi (H)

D = *duty circle*

T = periode (detik)

Dari persamaan differensial orde pertama 2.16, diperoleh:

$$i_L(t) = -\frac{V_o}{L}t + I_{Lmax} \quad (2.17)$$

Dimana nilai $t \rightarrow T_{off} = (1 - D)T$, arus induktor berkurang ke nilai minimum I_{Lmin} sehingga diperoleh persamaan 2.18.

$$i_L(t) = -\frac{V_o}{L}(1 - D) + I_{Lmax} \quad (2.18)$$

Persamaan 2.18 menghasilkan ekspresi lain untuk riak arus puncak ke puncak sebagai berikut:

$$\Delta I_L = I_{Lmax} - I_{Lmin} = \frac{V_d - V_o}{L} (1 - D)T \quad (2.19)$$

Keterangan:

$i_L(t)$ = arus induktor dalam satuan waktu tertentu (A)

V_o = tegangan *output* (V)

V_d = tegangan dioda (V)

I_{Lmin} = arus terkecil yang melewati induktor (A)

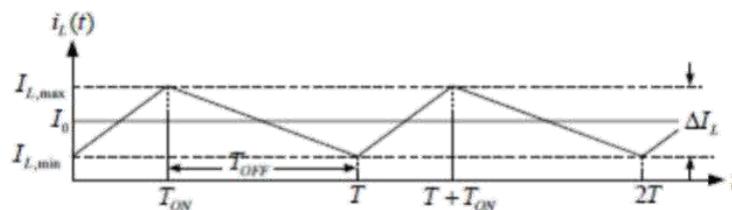
I_{Lmax} = arus induktor maksimum (A)

ΔI_L = *ripple* arus induktor (A)

L = nilai induktansi (H)

D = *duty circle*

T = periode (detik)



Gambar 2.6 Grafik Respon Arus Induktor

Sumber: (Sanjay Murmu dan M. B. Sharma, 2015)

Arus melalui induktor pada persamaan 2.18 selama (t) waktu dan persamaan 2.19 selama waktu tunda digambarkan pada Gambar 2.6. Arus rata-rata dalam induktor sama dengan arus DC melalui beban sehingga diperoleh

persamaan 2.20. Sedangkan ekspresi untuk arus maksimum dan minimum melalui induktor dituliskan dalam persamaan 2.21 dan persamaan 2.22.

$$I_{L,avg} = I_o = \frac{V_o}{R} \quad (2.20)$$

$$I_{Lmax} = I_{L,avg} + \frac{\Delta I_L}{2} = \frac{V_o}{R} + \frac{V_o}{2L}(1 - D)T \quad (2.21)$$

$$I_{Lmin} = I_{L,avg} - \frac{\Delta I_L}{2} = \frac{V_o}{R} - \frac{V_o}{2L}(1 - D)T \quad (2.22)$$

Keterangan:

$I_{L,avg}$ = arus rata-rata induktor (A)

I_o = arus *output* (A)

$-V_o$ = tegangan *output* polaritas negatif (V)

R = resistansi beban (Ω : ohm)

I_{Lmax} = arus maksimal yang melewati induktor (A)

I_{Lmin} = arus maksimal yang melewati induktor (A)

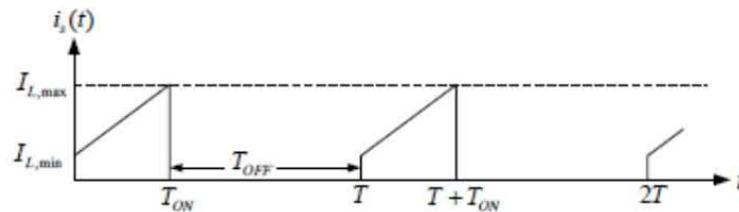
L = nilai induktansi (H)

ΔI_L = *ripple* arus induktor (A)

D = *duty circle*

T = periode (detik)

Arus yang dipasok oleh sumber *input* bervariasi dari I_{Lmin} hingga I_{Lmax} selama saklar *ON* dan pada waktu sama dengan nol tampak pada Gambar 2.7.

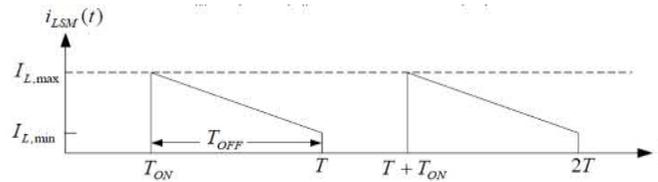


Gambar 2.7 Arus Masukan *Buck Converter*
Sumber: (Sanjay Murmu dan M. B. Sharma, 2015)

Ketika saklar aktif, induktor, dan kapasitor dalam kondisi ideal, diperoleh daya rata-rata yang dikeluarkan sama dengan nol. Sedangkan, daya rata-rata yang disuplai oleh sumber harus sama dengan daya rata-rata yang dikirim ke beban sehingga diperoleh persamaan 2.3.

$$V_d I_d = V_o I_o = DV_s I_s \quad (2.23)$$

Persamaan 2.23 dapat diekspresikan rata-rata arus sumber dengan rata-rata arus beban $I_s = DI_o$, arus melalui sisi bawah MOSFET ditunjukkan pada Gambar 2.8.



Gambar 2.8 Gelombang Arus yang Melalui MOSFET
Sumber: (Sanjay Murmu dan M. B. Sharma, 2015)

Buck Converter beroperasi pada mode CCM, selalu ada arus induktor. Arus minimum dalam mode CCM dapat menjadi nol. Akibatnya, terdapat nilai minimum induktor yang memastikan konverter dalam mode CCM. Sehingga diperoleh dari persamaan 2.22 dengan mengatur I_{Lmin} sama dengan nol.

$$\frac{V_o}{R} - \frac{V_o}{2L_{min}} (1 - D)T = 0 \quad (2.24)$$

$$L_{min} = \frac{1-D}{2} RT = \frac{1-D}{2f_s} R \quad (2.25)$$

Berdasarkan persamaan 2.25 diperoleh nilai induktansi pada persamaan 2.26.

$$L = \frac{V_o \cdot (1-D)}{f_s \cdot \Delta I_L} \quad (2.26)$$

Keterangan:

V_d = tegangan pada dioda (V)

V_o = tegangan *output* (V)

I_d = arus yang melewati dioda (A)

V_s = tegangan saklar aktif MOSFET (V)

I_s = arus yang melewati saklar aktif MOSFET (V)

R = resistansi beban (Ω : ohm)

L_{min} = nilai induktansi minimum (H)

L = nilai induktansi (H)

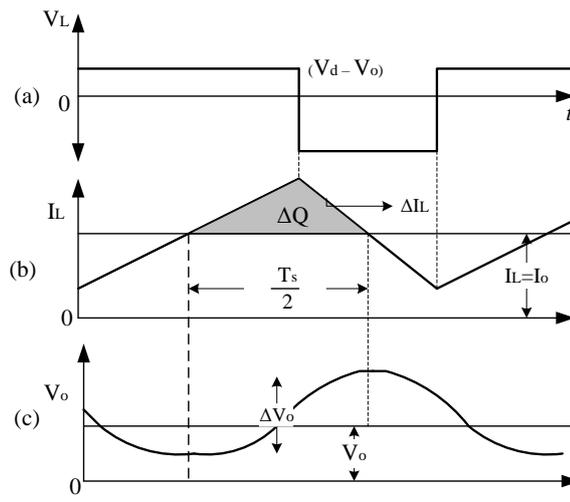
D = *duty circle*

T = periode (detik)

f_s = *frekuensi switching* (Hz)

2. Kalkulasi Nilai Konduktor

Tegangan keluaran sistem pada kenyataannya tidak dapat bernilai konstan secara sempurna, hal ini dikarenakan kapasitor bekerja terus menerus dan melakukan pengisian serta pelepasan muatan pada waktu saklar *ON* dan *OFF*.



Gambar 2.9 Riak Tegangan Keluaran *Buck Converter*
Sumber: (Sanjay Murmu dan M. B. Sharma, 2015)

Tegangan keluaran sistem V_o akan menghasilkan *ripple* ΔV_o pada Gambar 2.9 (c) sedangkan Gambar 2.9 (b) menunjukkan karakteristik arus dari kapasitor. Riak tegangan puncak-ke-puncak ΔV_o dituliskan dalam persamaan 2.27.

$$\Delta V_o = \frac{\Delta Q}{C} = \frac{1}{C} \left(\frac{1}{2} \right) \left(\frac{\nabla I_L}{2} \right) \left(\frac{T_s}{2} \right) \quad (2.27)$$

Berdasarkan gambar 2.8 pada saat T_{off} diperoleh persamaan 2.28.

$$\Delta I_L = \frac{V_o}{L} (1 - D) T_s \quad (2.28)$$

Keterangan:

ΔV_o = tegangan *output peak to peak* (V)

ΔQ = muatan kapasitor (Coulomb)

∇I_L = *ripple* arus induktor (A)

V_o = tegangan *output* (V)

L = nilai induktansi (H)

C = nilai kapasitansi (F)

D = *duty circle*

T_s = periode *switching* (detik)

Mensubstitusikan ΔI_L dari persamaan 2.28 ke persamaan 2.27, diperoleh persamaan 2.29.

$$\Delta V_o = \frac{T_s}{8C} \left(\frac{V_o}{L} \right) (1 - D) T_s \quad (2.29)$$

Dengan mensubstitusikan $T_s = \left(\frac{f_c}{f_s} \right)^2$, dihasilkan persamaan 2.30.

$$\frac{\Delta V_o}{V_o} = \frac{1}{8} \frac{T_s^2 (1-D)}{LC} = \frac{\pi^2}{2} (1 - D) \left(\frac{f_c}{f_s} \right)^2 \quad (2.30)$$

Dengan demikian besarnya kapasitor dapat dihitung dengan mensubstitusikan nilai frekuensi, ditulis dalam persamaan 2.31.

$$\frac{\Delta V_o}{V_o} = \frac{(1-D)V_o}{8Lfs^2\Delta V_o} \quad (2.31)$$

Dimana frekuensi *switching* $f_s = \frac{1}{T}$.

Keterangan:

ΔV_o = tegangan *output peak to peak* (V)

∇I_L = *ripple* arus induktor (A)

V_o = tegangan *output* (V)

L = nilai induktansi (H)

C = nilai kapasitansi (F)

D = *duty circle*

T_s = periode *switching* (detik)

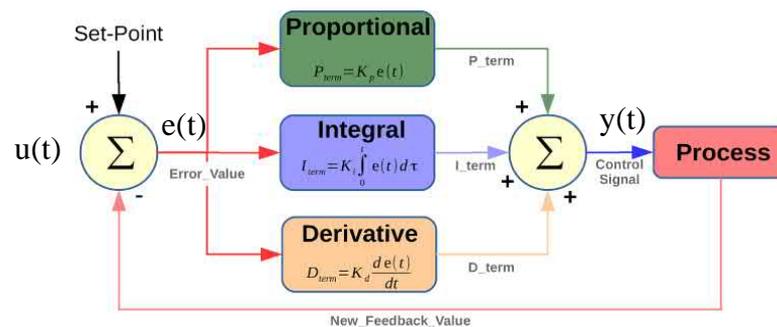
f_c = frekuensi *cut off* (Hz)

f_s = frekuensi *switching* (Hz)

Persamaan 2.30 menunjukkan bahwa riak atau *ripple* tegangan dapat diminimalkan dengan memilih frekuensi sudut f_c dari *Low Pass Filter* pada *output* sedemikian rupa sehingga $f_c < f_s$. Selain itu, riak tidak tergantung pada daya beban keluaran, asalkan konverter beroperasi dalam mode konduksi kontinu. Persentase riak pada tegangan keluaran konverter biasanya ditetapkan sangat kecil misalnya 1% (Sanjay Murmu dan M. B. Sharma, 2015).

2.2.2. PID Controller

Proportional – Integral – Derivative (PID) Controller memiliki siklus umpan balik dalam kontrol mekanisme, yaitu mengoreksi nilai *error* antara hasil proses kontrol pada keluaran sistem yang terukur dengan nilai *set point* yang telah ditetapkan, selisih nilai yang dihasilkan dikalkulasi untuk memberikan aksi pengontrolan untuk mencapai nilai yang ditetapkan. Tampak pada Gambar 2.10 PID Controller melibatkan tiga parameter terpisah, yaitu *gain Proporsional* (K_p), *gain Integral* (K_i), dan *gain derivative* (K_d). Pengaruh setiap *gain* PID terhadap respon sistem dirangkum Tabel 2.1.



Gambar 2.10 Diagram Blok Sistem Umpan Balik PID Controller

Keterangan:

$e(t)$ = nilai *error* (kesalahan)

$u(t)$ = nilai masukan

$y(t)$ = nilai keluaran

Tabel 2.1 Pengaruh Peningkatan Parameter PID

Gain	Rise time	Overshoot	Settling time	Steady state error	Stability
K_p	Lebih Cepat	Meningkat	Perubahan kecil	Menurun	Menurun
K_i	Lebih Cepat	Meningkat	Meningkat	Menghilang	Menurun
K_d	Perubahan kecil	Menurun	Menurun	Tidak ada efek	Meningkat jika K_d kecil

Berdasarkan Tabel 2.1 terdapat 4 parameter respon dinamis yang menjadi fokus perbaikan yaitu: *rise time*, *overshoot*, *settling time*, dan *steady state error*. Upaya perbaikan parameter tersebut dikenal dengan 3 aksi pengendali.

a. *Proportional Control*

Proportional Control (P) memiliki sinyal keluaran sebanding dengan sinyal *error* sistem. Artinya sinyal kontrol akan langsung mengubah sinyal *error* dikalikan dengan suatu konstanta proporsionalnya. Persamaan kontroler proporsional terhadap domain waktu tampak pada Gambar 2.10 ditulis dalam persamaan 2.32 dan domain *Laplace* ditulis pada persamaan 2.33.

$$\text{Domain waktu: } u(t) = K_p e(t) \quad (2.32)$$

$$\text{Domain } Laplace: U(s) = K_p \quad (2.33)$$

Keterangan:

$u(t)$ = nilai masukan domain waktu

$U(s)$ = nilai masukan domain *Laplace*

K_p = *gain Proportional*

$e(t)$ = nilai *error* (kesalahan)

Kendali proporsional memiliki nilai konstanta proporsional (K_p) mempengaruhi cepatnya respon kontroler untuk mencapai nilai *set point*. Semakin besar nilai K_p maka respon akan semakin cepat, namun jika nilai K_p besar akan menyebabkan respon *overshoot* dan osilasi.

b. *Integral Control*

Integral Control (I) memiliki sinyal keluaran *integral* dari *error* sistem. Kendali *integral* dapat mengeliminasi *steady state error*. Selain itu, *Integral Controller* responnya membutuhkan selang waktu tertentu sehingga terkesan

memperlambat respon. Adapun persamaan domain waktu dan domain *Laplace* kendali *integral* tertulis pada persamaan 2.34 dan 2.35.

$$\text{Domain waktu: } u(t) = K_i \int e(t) dt \quad (2.34)$$

$$\text{Domain } Laplace: U(s) = \frac{K_i}{s} \quad (2.35)$$

Keterangan:

$u(t)$ = nilai masukan domain waktu

$U(s)$ = nilai masukan domain *Laplace*

K_i = *gain Integral*

$e(t)$ = nilai *error* (kesalahan)

s = satuan frekuensi bilangan kompleks

c. *Derivative Control*

Derivative Control (D) adalah kontroler yang tidak menghasilkan sinyal keluaran saat tidak ada sinyal *error*. Selain itu kontroler ini adalah sifatnya yang responsif terhadap perubahan *error*. Hal ini membuat kontroler derivatif dapat menambah kestabilan sistem. Persamaan domain waktu dan domain *Laplace* kendali *integral* tertulis pada persamaan 2.36 dan 2.37.

$$\text{Domain waktu: } u(t) = K_d \frac{de(t)}{dt} \quad (2.36)$$

$$\text{Domain } Laplace: U(s) = K_d s \quad (2.37)$$

Keterangan:

$u(t)$ = nilai masukan domain waktu

$U(s)$ = nilai masukan domain *Laplace*

K_d = *gain Derivative*

$e(t)$ = nilai *error* (kesalahan)

s = frekuensi bilangan kompleks

d. *Proportional Integral Control*

Gabungan aksi kontrol proporsional dan aksi kontrol *integral* membentuk aksi kontrol proporsional ditambah *integral* (kendali PI). Gabungan aksi ini

mempunyai keunggulan dibandingkan dengan masing-masing penyusunnya. Elemen-elemen kontroler P dan I secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem dan mengurangi *steady state error*. Kekurangan kendali PI adalah memberikan nilai deviasi yang maksimum, sehingga memiliki periode osilasi yang lebih lama.

e. *Proportional Derivative Control*

Teknik *proportional-derivatif* (PD) *Controller* adalah pengendali yang merupakan gabungan antara teknik kendali proporsional (P) dengan teknik kendali derivatif (D). Penambahan derivatif dapat menghilangkan osilasi yang berlebihan sistem pengendalian proporsional. Sedangkan kekurangan kendali PD adalah *steady state error* yang tidak dapat hilang.

f. *Proportional Integral Derivative Control*

Pengendali PID merupakan kombinasi dari kendali P, I, dan D. Keuntungan kombinasi ketiga elemen kendali membuat pengendali PID nyaris tidak memiliki *steady state error*, karena adanya pengendali *integral*. Selain itu berkurangnya deviasi maksimum dan waktu osilasi yang merupakan hasil gabungan pengendalian PI dan PD.

Sinyal yang keluar dari kontroler adalah jumlah *gain* proporsional (K_P) dikali besarnya *error*, *gain integral* (K_I) dikali *integral error*, dan *gain* derivatif (K_D) dikali derivatif *error*. Berdasarkan Gambar 2.10 representasi domain waktu dari sinyal $u(t)$ yang diumpankan ke sistem kontroler yang diberikan ditulis dalam persamaan. Persamaan PID *Controller* dituliskan dalam domain waktu pada persamaan 2.38.

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (2.38)$$

Persamaan 2.38 dapat transformasikan dalam domain frekuensi melalui transformasi *Laplace* yaitu $G_c(s)$, diperoleh persamaan 2.39.

$$U(s) = K_p + \frac{K_i}{s} + K_d s \quad (2.39)$$

Keterangan:

$u(t)$ = nilai masukan domain waktu

$U(s)$ = nilai masukan domain *Laplace*

K_p = *gain Proportional*

K_i = *gain Integral*

K_d = *gain Derivative*

$e(t)$ = nilai *error* (kesalahan)

s = frekuensi bilangan kompleks

2.2.3. Optimisasi Sistem Kendali

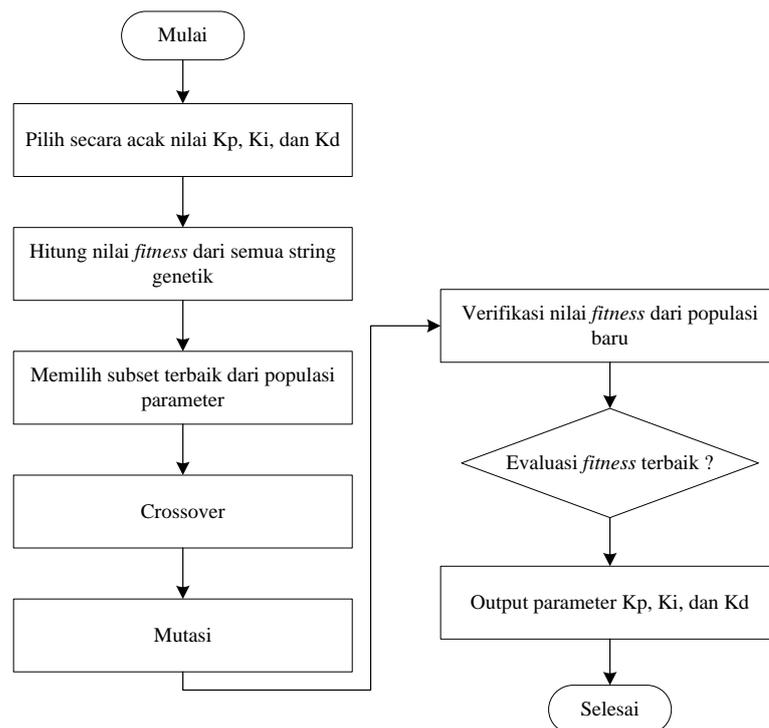
Optimisasi sistem kendali digunakan untuk meminimumkan kesalahan indeks kinerja tertentu. Indeks kerja merupakan bilangan yang menunjukkan “perbaikan” kinerja sistem. Indeks kinerja harus menghasilkan satu nilai positif atau nol, dan akan diperoleh jika dan hanya jika pengukuran penyimpang nol (Ogata,2010). Salah satu kesalahan indeks kerja dalam *system control* adalah *Integral Time-Weighted Absolute Error* (ITAE), yang dapat digunakan untuk memperbaiki nilai PID *Controller*. Menurut Abderrezek, *et al.* (2018) ITAE mampu memperkecil nilai persentase *overshoot* dan nilai *settling time*. Penggunaan ITAE dalam sistem kendali dalam bentuk kontinyu dinyatakan dengan persamaan 2.40.

$$ITAE = \int_0^T t|e(t)|dt \quad (2.40)$$

Keterangan: $e(t)$ = *error* tegangan terhadap waktu; $0 - T$ = batas waktu (detik)

2.2.4. Genetic Algorithm (GA)

GA diperkenalkan oleh John H. Holland pada tahun 1960-an yang sebelumnya dikenal dengan “*Genetic Reproductive Plans*” (S. Sumanthi, *et al.* 2016). GA adalah metode untuk menyelesaikan masalah optimisasi yang didasarkan pada seleksi alam, melalui proses evolusi biologis yang berdasarkan teori evolusi Charles Darwin (Singh Charanjeet, 2015) dan (Dusan Fister, *et al.* 2016). GA menjadi salah satu dari algoritma pertama di algoritma evolusi (EA) yang berdasarkan pada proses seleksi alam dimana individu terkuat yang akan bertahan diakhir.



Gambar 2.11 Proses Optimisasi Menggunakan *Genetic Algorithm*

Tampak pada Gambar 2.11, proses inialisasi GA bekerja dengan satu set solusi kandidat yang disebut populasi GA mendapatkan solusi optimal setelah serangkaian perhitungan berulang. GA menghasilkan populasi solusi pengganti

berurutan yang diwakili oleh kromosom, yaitu solusi untuk masalah, hingga hasil yang dapat diterima diperoleh.

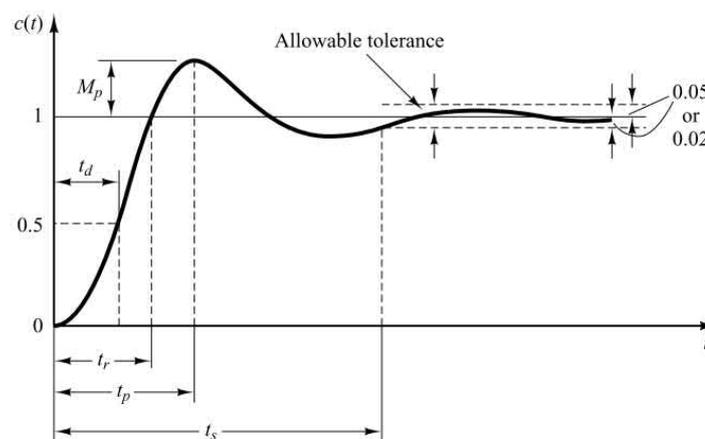
GA biasanya diinisialisasi dengan acak populasi terdiri dari antara 20-100 individu (J. Kumar Ohri & M. Chinda, 2014). Setiap individu dalam populasi terdapat proses *selection*, *crossover*, dan *mutation*. Proses *selection* merupakan keturunan individu yang terbaik akan terpilih lagi menjadi *parent* pada generasi berikutnya. Untuk memberikan solusi yang lebih baik, GA menggunakan operator genetik atau operator evolusi seperti *crossover* dan mutasi untuk penciptaan individu baru dari yang sudah ada dalam populasi. *Output* terbaik dari proses evolusi adalah individu yang baik atau dengan sebutan nilai *fitness organism*. Proses keseluruhan merupakan proses iterasi dari generasi ke generasi, dimana setiap generasi menghasilkan keturunan atau *offspring*, setiap keturunan mempunyai individu yang baik dibandingkan *parent* atau orang tuanya. Terkait dengan karakteristik pencarian eksploitasi dan eksplorasi, GA dapat berurusan dengan ruang pencarian besar secara efisien, dan karenanya memiliki sedikit kesempatan untuk mendapatkan solusi optimal lokal dari pada algoritma lainnya.

Seperti dalam proses evolusi, GA memanipulasi setiap populasi individu dengan nilai *fitness* yang terkait untuk generasi baru individu menggunakan prinsip Darwin reproduksi dan kelangsungan hidup yang terkuat. Setiap individu dari populasi merupakan solusi yang memungkinkan. Oleh karena itu, GA mencari di antara serangkaian solusi di ruang pencarian, selalu menuju performa yang optimal dengan nilai individu terkuat.

2.2.5. Analisis Respon Transien

Dalam beberapa kasus praktis, karakteristik performansi sistem kontrol yang diinginkan dinyatakan dalam bentuk besaran kawasan waktu. Respon transien suatu sistem terhadap masukan tangga satuan bergantung pada syarat awal. Untuk memudahkan pembandingan respon transien berbagai macam sistem, hal yang dapat dilakukan adalah menggunakan syarat awal standar bahwa sistem awal dalam keadaan diam sehingga keluaran dan semua turunan waktunya pada awal respon sama dengan nol. Selanjutnya karakteristik respon secara mudah dapat dibandingkan.

Respon transien sistem kontrol praktis sering menunjukkan osilasi teredam sebelum mencapai keadaan tunak. *Buck Converter* sebagai sistem berorde 2 memiliki parameter berupa: ω_n adalah frekuensi alami tak teredam (*natural frequency*), ζ adalah konstanta perbandingan peredam (*damping ratio*), ω_d adalah frekuensi alami yang teredam (*damped natural frequency*). Untuk sistem orde 2 dengan $0 < \zeta < 1$ dan $\omega_n > 0$. Tanggapan respon transien ditunjukkan pada gambar 2.12.



Gambar 2.12 Analisis Respon Transien dan Keadaan Tunak (Stabil)
Sumber: (Katsuhiko Ogata, 2010, p. 170)

Buku teknik kontrol *modern* Katsuhiko Ogata, 2010, p. 170) terdapat 5 parameter untuk menentukan karakteristik respon transien, diantaranya:

1. *Delay time*: waktu tunda (t_d), waktu tunda adalah waktu yang diperlukan respon untuk mencapai setengah harga akhir yang pertama kali.
2. *Rise time*: waktu naik (t_r), waktu naik adalah waktu yang diperlukan respon untuk naik dari 10% sampai 90%, 5 sampai 95%, atau 0 sampai 100% dari nilai akhirnya. Untuk sistem orde dua seperti *Buck Converter* digunakan waktu naik 0-100%.
3. *Peak time*: waktu puncak (t_p), merupakan waktu yang diperlukan respon mencapai puncak lewatan pertama kali.
4. *Maximum overshoot*: lewatan maksimum (M_p), adalah harga puncak maksimum dari kurva respon yang diukur dari keadaan tunak (referensi). Besarnya lewatan maksimum dinyatakan dalam presentase, yaitu selisih antara amplitudo terbesar dengan nilai *set point* (nilai yang ditetapkan).
5. *Settling time*: waktu penentapan (t_s), adalah waktu yang diperlukan respon untuk mencapai dan menetap dalam daerah disekitar harga akhir (kondisi tunak) yang ukurannya ditentukan dengan persentase mutlak dari harga akhir. Waktu penentapan ini berkaitan dengan konstanta waktu terbesar dari sistem kontrol, ditulis dalam persamaan 2.41.

$$t_s = \frac{4}{\xi \omega_n}. \text{ Untuk } steady\ state\ 2\% \quad (2.41)$$

Keterangan:

ζ = faktor redaman

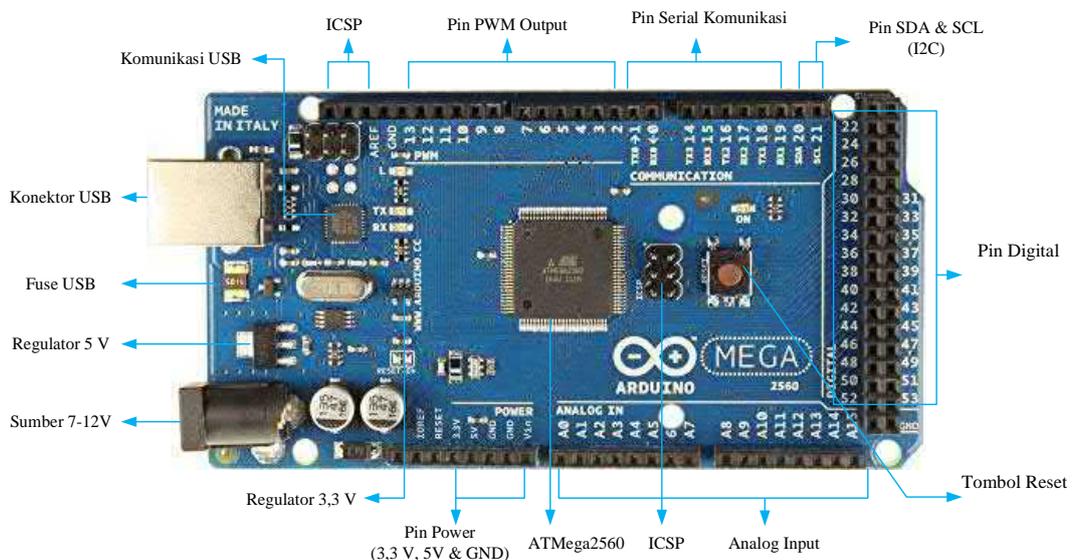
ω_n = frekuensi alami tak teredam (Hz)

Untuk tanggapan yang cepat ω_n harus besar. Jika faktor redaman $\zeta < 0,4$ menyebabkan *overshoot* yang berlebihan pada tanggapan transien dan sistem dengan $\zeta > 0,8$ membuat tanggapan sistem lambat. Sehingga ζ ditetapkan $0,4 < \zeta < 0,8$ maka *overshoot* maksimum (%) untuk *step respons* adalah di antara 25% dan 2,5%.

2.2.6. Komponen *Hardware Buck Converter*

1. Arduino Mega2560

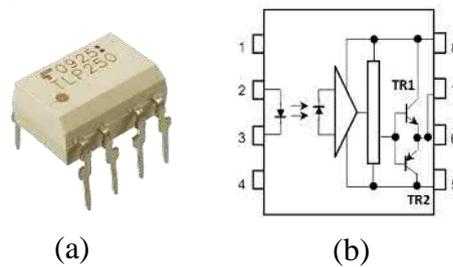
Arduino Mega 2560 adalah *platform* papan mikrokontroler dengan IC ATmega2560, memiliki 54 pin *input-output* digital (14 di antaranya dapat digunakan sebagai *output* PWM), 16 *input* analog, 4 *universal* pemancar penerima sinkron (UART) (*port* serial perangkat keras), osilator kristal 16 MHz, *universal serial bus* (USB) koneksi, *serial programming in-circuit* (ICSP) *header* dan tombol *reset*. Board Arduino Mega 2560 ditunjukkan pada Gambar 2.13.



Gambar 2.13 Papan Mikrokontroler Arduino ATmega2560

2. TLP 250

Rangkaian TLP250 merupakan rangkaian *optocoupler* yang digunakan untuk menguatkan sinyal PWM yang dibangkitkan oleh rangkaian kontrol (*driver*) PWM menjadi *level* tegangan yang lebih tinggi pada rangkaian daya dan mengamankan *Grounding* (GND) dari tegangan rendah dari Arduino dengan tegangan tinggi VCC. Bentuk fisik dan konfigurasi pin kaki dari TLP250 tampak pada Gambar 2.14.



Gambar 2.14 (a) Bentuk Fisik TLP 250. (b) Konfiguasi Pin TLP250

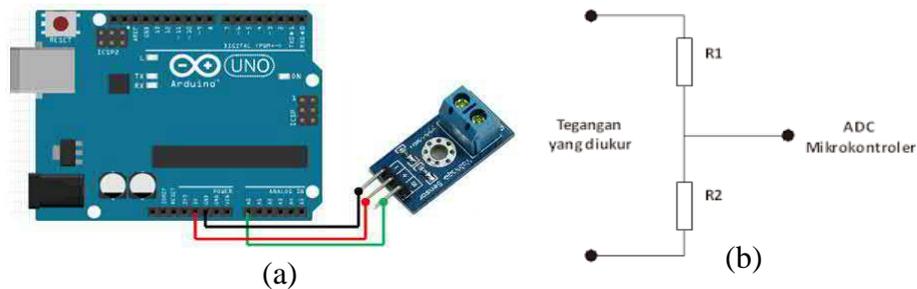
Keterangan Gambar 2.14 pin dalam TLP250:

Pin 1	: <i>Normaly Close (NC)</i>	Pin 5	: GND
Pin 2	: Anoda	Pin 6	: <i>Voutput</i>
Pin 3	: Katoda	Pin 7	: <i>Voutput</i>
Pin 4	: <i>Normaly Close (NC)</i>	Pin 8	: Vcc

3. Sensor Tegangan

Voltage Divider atau Pembagi Tegangan adalah suatu rangkaian sederhana yang mengubah tegangan besar menjadi tegangan yang lebih kecil dengan membagi tegangan *input* menjadi *level* tegangan *output* pada *level* yang ditentukan.

Rangkaian pembagi tegangan terdiri dari 2 nilai resistor tampak pada Gambar 2.15. Terdapat 3 pin *output* sensor yaitu "+" dihubungkan pada 5V/3,3V, "-" terhubung GND, "s" terhubung Arduino pin analog yaitu A0.



Gambar 2.15 (a) Skematik Rangkaian Sensor Tegangan dan Arduino Uno.
(b) Rangkaian Pembagi tegangan pada Sensor Tegangan
Sumber: Arduino.cc

Spesifikasi sensor tegangan pada Gambar 2.15(a), yaitu:

- Variasi Tegangan *input*: DC 0 - 25 V
- Deteksi tegangan dengan jangkauan: DC 0,02445 V - 25 V

Berdasarkan Gambar 2.15 (b) diperoleh persamaan 2.42, hasil nilai tegangan *output* (V_o) dari rangkaian pembagi tegangan::

$$V_{out} = V_{in} \times \left(\frac{R_2}{R_1 + R_2} \right) \quad (2.42)$$

Keterangan:

V_{out} = tegangan keluaran (V)

V_{in} = tegangan masukan (V)

R_1 , R_2 = nilai resistansi pembagi tegangan (Ω)

2.2.7. MATLAB

MATLAB (Matrix Laboratory) adalah bahasa pemrograman tingkat tinggi yang diperuntukkan untuk komputasi teknik. MATLAB mengintegrasikan aspek komputasi, visualisasi, pemrograman serta dapat digunakan untuk aplikasi.

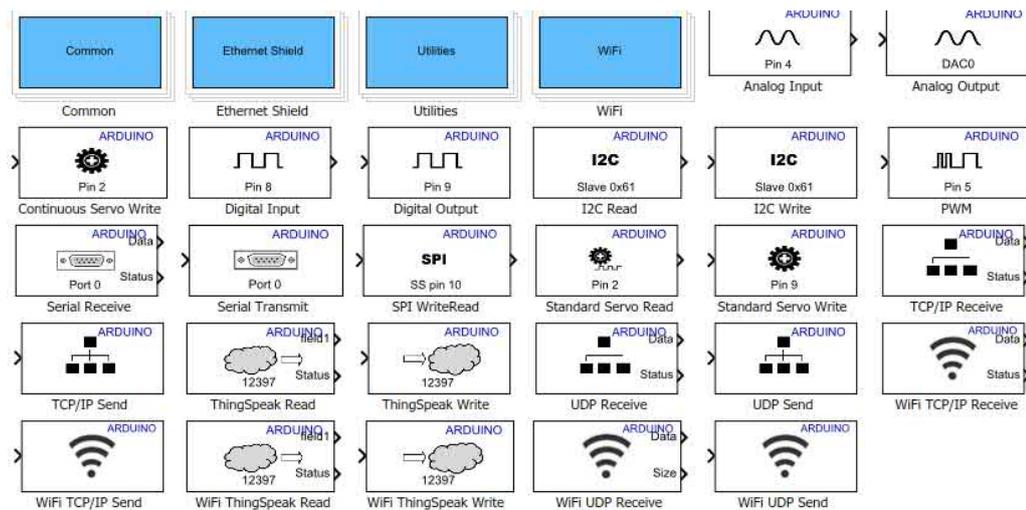
Software ini yang dikembangkan oleh “The Math Work”. MATLAB mampu membantu memecahkan berbagai masalah matematis dalam bidang teknis. *User* dapat memanfaatkan kemampuan MATLAB untuk menganalisa dari berbagai masalah komputasi numerik atau berbasis matrik secara cepat dan tepat.

Dalam penelitian keteknikan, MATLAB digunakan pada pengolahan sinyal, penyelesaian matrik, simulasi kontrol dan sistem, pengajaran keteknikan, berintegrasi dengan aplikasi keteknikan lainnya, dan lain sebagainya. Selain itu, MATLAB memiliki beberapa kelebihan antara lain: bahasa tingkat tinggi (Bahasa C++) untuk matriks pada teknik komputasi, mampu mengolah kode, file, dan data, serta alat interaktif untuk ekspolasi *iterative*, desain, dan pemecahan masalah.

2.2.8. Simulink

Simulink adalah *platform* di dalam MATLAB yang digunakan untuk mensimulasikan sistem dinamik secara *real time*. Simulink MATLAB dapat memudahkan dalam membangun sistem yaitu dari persamaan dinamika dengan berbagai macam fasilitas analisis data (MathWork). Model Simulink hampir sama dengan diagram blok konvensional. Untuk sistem dengan komponen analog, Simulink mewujudkan persamaan aljabar dan diferensial dari model matematika waktu kontinu. Untuk sistem waktu diskrit yang *inheren*, model Simulink merangkum persamaan aljabar dan perbedaan yang mengatur kinerja sistem.

“Simulink Support Package Arduino”, merupakan fasilitas yang tersedia di MATLAB sebagai *interface* dalam menjalankan model Simulink terhadap *hardware board* Arduino. Fasilitas ini didukung *library* dalam blok Simulink untuk mengkonfigurasi dan mengakses sensor Arduino, aktuator, dan antarmuka komunikasi serta memungkinkan untuk menyetel algoritma kontroler yang dikembangkan di Simulink saat dijalankan di Arduino. Untuk mendapatkan *toolbox library* Arduino seperti Gambar 2.16 maka terlebih dahulu menambahkan dan menginstall “Simulink Support Package Arduino” yang diperoleh dari “Add Ons” pada *toolbar* “Environment”.



Gambar 2.16 *Toolbox Library* Arduino

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

1. Desain dan perencanaan topologi *Buck Converter mode Continuous Current Mode* (CCM) telah diimplementasikan dalam bentuk *hardware* dengan nilai komponen disesuaikan dengan ketersediaan komponen dipasaran yang mendekati nilai sesuai perhitungan dan mempertimbangkan rugi-rugi komponen yang dapat ditimbulkan. Pembangkitan sinyal dan frekuensi *Pulse Width Modulation* (PWM), umpan balik dari pengukuran sensor tegangan, serta sistem *Proportional Integral Derivative* (PID) *Controller* pada *Buck Converter* menggunakan Mikrokontroler *platform* Arduino ATmega 2560.
2. Metode *tuning* PID *Controller* dengan *Genetic Algorithm* (GA) menggunakan bantuan *toolbox optimization* yang tersedia di MATLAB. Proses *tuning* PID menggunakan GA diawali dengan permodelan *Plan System* (*Buck Converter*) dan *Controller System* (PID *Controller*) dalam bentuk domain frekuensi yaitu dengan transformasi *Laplace*. Proses GA selanjutnya adalah menentukan nilai dan tipe dari parameter-parameter yang dibutuhkan, seperti populasi, *crossover*, *lower and upper*, *iterasi/generation*, dan sebagainya. GA akan mengevolusi dan mengeksplorasi berdasarkan Teori Darwin yang akan menghasilkan nilai yang terkuat atau terbaik.

Kontroler yang ditetapkan dalam penelitian ini adalah PI dan PID *Controller*, dengan metode *tuning* GA ini menghasilkan: $K_p = 0,003$ dan $K_i = 7$ pada PI *Controller*, dan $K_p = 0,001$, $K_i = 5$, dan $K_d = 0,001$ pada PID *Controller*.

3. Pemrograman mikrokontroler Arduino ATmega 2560 menggunakan Simulink pada MATLAB yang terintegrasi dan *support* dengan Arduino. Perbedaan Simulink dengan *compiler* Arduino (Arduino IDE) adalah dalam interpretasi pemrogramannya, Model pemrograman Simulink dalam bentuk diagram blok dengan penggunaan *library* yang sesuai dengan alur algoritma pemrograman sedangkan Arduino IDE dalam bentuk bahasa pemrograman C++. Pemrograman Simulink yang telah di *run* dan di *upload* ke mikrokontroler dapat dilihat hasil respon *hardware* dengan memasang *Scope* yang tersedia di *library* Arduino. Grafik *Scope* yang tampil merupakan pembacaan dan pengukuran data dari respon *hardware* Buck Converter secara *real time*, grafik inilah yang digunakan dalam analisis transien Buck Converter. Berdasarkan hasil data nilai *steady state system* Buck Converter salah satunya pada Tabel 4.8 ketika pengujian menggunakan Multimeter dan Tabel 4.9 pengujian berdasarkan respon yang tampak pada *Scope* Simulink terdapat sedikit perbedaan. Hal tersebut tentunya wajar karena hasil pembacaan/pengukuran Multimeter bersifat nilai rata-rata sedangkan pembacaan di *Scope* MATLAB bersifat *continuous* dan dapat membaca *noise* yang timbul pada *hardware* tampak pada grafik respon yang dihasilkan.
4. Performansi terbaik Buck Converter dengan variasi tegangan maupun variasi beban pada penggunaan PI *Controller*, penentuan ini berdasarkan kecepatan

respon (*rise time* dan *settling time*), *overshoot* yang dihasilkan, nilai *steady state* dan *error steady state* serta efisiensi yang dihasilkan *hardware Buck Converter*. Dengan *PI Controller* respon yang dihasilkan *rise time*: 0,0221 detik, *settling time*: 0,0296 detik, *overshoot*: 1,05%, *steady state* sebesar 5,357 V sehingga menghasilkan *error steady state*: 7,15%. Sedangkan *PID Controller* menghasilkan *rise time* sebesar 0,0603 detik, *settling time* 0,0762 detik, *peak time* 0,0837 detik, *overshoot* 3,36% dengan nilai *steady state* sebesar 5,29 Volt dan *error steady state* yang dihasilkan sebesar 5,80%. Keunggulan dari *PID Controller* menghasilkan *noise* yang kecil dibanding *PI Controller* karena sifat *Derivative Controller* adalah meredam *overshoot* tetapi berdampak pada waktu respon yang lebih lama.

5.2. Saran

Berdasarkan keterbatasan penelitian, ada beberapa saran yang dapat dipertimbangkan pada penelitian selanjutnya, diantaranya yaitu:

1. Penelitian dapat dikembangkan dengan membandingkan metode *tuning* lainnya, tidak hanya GA saja, terlebih dengan membandingkan metode *tuning* dengan *Artificial Intelligent (AI)* dan metode *tuning* konvensional lainnya.
2. Implementasi *hardware Buck Converter* berdasarkan desain dan perencanaan hanya menghasilkan efisiensi daya rata-rata sebesar 81,49%, diharapkan pengembangan lainnya dapat meningkatkan efisiensi dan memperbaiki respon transien yang dihasilkan khususnya untuk mengurangi *noise* dan *ripple* yang dihasilkan *Buck Converter* pada frekuensi tinggi yaitu 25kHz.

DAFTAR PUSTAKA

- Abderrezek, H., Ameer, A., & Harmas, M. N. (2018). Stochastic Optimization Methods Based Robust Control for DC/DC Buck Converter. 2018 5th International Conference on Control, Decision and Information Technologies, CoDIT 2018, (1), 315–320. <https://doi.org/10.1109/CoDIT.2018.8394886>
- Aranza, M. F., Kustija, J., Trisno, B., & Hakim, D. L. (2016). Tuning PID Controller Using Particle Swarm Optimization Algorithm On Automatic Voltage Regulator System. IOP Conference Series: Materials Science and Engineering, 128(1). <https://doi.org/10.1088/1757-899X/128/1/012038>
- Deekshitha, C., & Shenoy, K. Latha. (2017). Simulation of Open Loop and Closed Loop Synchronous Buck Converter for LED Applications. Ijireeice, 5(2), 112–115. <https://doi.org/10.17148/ijireeice/ncaee.2017.25>
- Cristri, A. W., & Iskandar, R. F. (2017). Analysis and Design of Dynamic Buck Converter with Change in Value of Load Impedance. Procedia Engineering, 170, 398–403. <https://doi.org/10.1016/j.proeng.2017.03.064>
- Fannakh, Mhmaed., Elhafyani, M. L., & Zouggar, S. (2018). Hardware implementation of the fuzzy logic MPPT in an Arduino card using a Simulink support package for PV application, 510–518. <https://doi.org/10.1049/iet-rpg.2018.5667>
- Ferreiro, J. B. L., Pombo, J. A. N., Calado, M. R. A., & Mariano, S. J. P. S. (2017). A New Controller for DC-DC Converters Based on Particle Swarm Optimization. Applied Soft Computing Journal, 52, 418–434. <https://doi.org/10.1016/j.asoc.2016.10.025>
- Fister, Dusan., Safaric, Riko., & Fister, Iztok. (2016). Parameter Tuning of PID Controller with Reactive Nature-Inspired Algorithms. <https://doi.org/10.1016/j.robot.2016.07.005>
- Husnaini, Irma., & Krismadinata. (2017). Komparasi Pengendali PI dan PID Untuk Tegangan Keluaran Konverter Buck. Jurnal Nasional Teknik Elektro, 6(3). <https://doi.org/10.20449/jnte.v6i3.387>
- Ikhsan, M., & Away, Yuwaldi. (2014). Studi dan Penerapan PID pada Kontrol Buck Converter Berbasis Mikrokontroler ATmega328P. Seminar Nasional dan Expo Teknik Elektro.
- Kannabiran, K., & Alagarsamy, S. (2016). Unified Control of DC-DC Buck Converter using Dynamic Adaptive Controller for Battery Operated Devices. Revista Facultad de Ingeniería, Universidad de Antioquia, No. 81, pp. 35-46. DOI: 10.17533/udea.redin.n81a04 35

- Khandhediya, Kalpana. S., Singh, P. S., & Daxini, P. B. M. (2015). Design and Simulation of Open-Loop Buck Converter with Losses, (2), 3–5.
- Kovacevic, H., & Stojanovic, Z. (2016). Buck Converter Controlled by Arduino Uno. 2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2016 - Proceedings, 1638–1642. <https://doi.org/10.1109/MIPRO.2016.7522401>
- Kurnar, Jagdish., Surjan, Balwinder. Singh., & Omer, Prabhu. (2018). Design of Robust PID Controller for Buck Converter Using Bat Algorithm. In 1st IEEE International Conference on Power Electronic, Intelligent Control and Energy System (ICPEICES).
- Cahyadi, L. W., Andromeda, T., Facta, M., (2017). Kinerja Konverter Arus Searah Tipe *Buck Converter* dengan Umpan Balik Tegangan Berbasis TL494. TRANSIENT, VOL.6, NO. 1, MARET 2017, ISSN: 2302-9927, 162
- Meena, D. C., & Devanshu, Ambrish. (2017). Genetic algorithm tuned PID Controller for Process Control. Proceedings of the International Conference on Inventive Systems and Control, ICISC 2017, 1–6.
- Mirzaei, & Mojallali. (2015). Auto Tuning PID Controller Using Chaotic PSO Algorithm for a Boost Converter. Syrian refugees struggle with TB. Science, 350(6267), 1446–1448.
- Murmu, Sanjay., & Sharma, B. (2015). Study and Design, Simulation of PWM based Buck converter for Low Power Application. IOSR Journal of Electrical and Electronics Engineering Ver. II, 10(4), 2278–1676. <https://doi.org/10.9790/1676-10420117>
- Ningtias,Dieta.2018. Pengaturan Kecepatan Motor DC Menggunakan Kontrol PID Dengan Metode Tuning Ziegler Nichols dan Algoritma Genetika. Fakultas Teknik. Universitas Negeri Semarang:Semarang
- Ogata, Katsuhiko. (2010). Modern Control Engineering. (H. M. J, Ed.) (Fifth Edit). America: Prentice Hall.
- Ohri, J., Kumar, N., & Chinda, M. (2014). an Improved Genetic Algorithm for PID Parameter Tuning. Proc.of the 2014 International Conference on Circuits, Systems, Signal Processing, 191–198. <https://doi.org/nrn2168> [pii]r10.1038/nrn2168
- Ouyang, Puren., & Pano, Vanggjel. (2015). Comparative study of DE, PSO and GA for position domain PID controller tuning. Algorithms, 8(3), 697–711. <https://doi.org/10.3390/a8030697>
- Puchta, Erickson. D. P., Lucas, R., Ferreira, F. R. V, Siqueira, H. V, & Kaster, M. S. (2016). Gaussian Adaptive PID Control Optimized via Genetic Algorithm

- Applied to a Step-down DC-DC Converter. 12th IEEE International Conference on Industry Applications (INDUSCON), 1(2). <https://doi.org/10.1109/induscon.2016.7874509>
- Barber, R., Hora, M., & Cespo, J. (2013). Control Practices using Simulink with Arduino as Low Cost Hardware. IFAC Proceedings Volumes (IFAC-PapersOnline) (Vol. 10). Sheffield, UK: IFAC Symposium Advances in Control Education. <https://doi.org/10.3182/20130828-3-UK-2039.00057>
- Radwan, Ahmed. G., Emira, A. A., AbdelAty, A. M., & Azar, A. T. (2018). Modelling and Analysis of Fractional Order DC-DC Converter. ISA Transactions, 82, 184–199. <https://doi.org/10.1016/j.isatra.2017.06.024>
- Saoudi, Magdy., El-Sayed, Ahmed., & Metwally, Hamid. (2017). Design and Implementation of Closed-Loop Control System for Buck Converter Using Different Techniques. IEEE Aerospace and Electronic Systems Magazine, 32(3), 30–39. <https://doi.org/10.1109/MAES.2017.150261>
- Shinde, N., Sankad, Surgappa., Patil, L.S., (2018). Design and Study Voltage Characteristics of Buck Converter by Matlab Simulink. 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI), p. 680–683.
- Singh, Charanjeet. (2015). Genetic Algorithms Based PID Controller Design. Publisher: Ijedr |, 3(3), 2–5. Retrieved from www.ijedr.org
- Sinha, Akarsh., Pavithra, M., Sutharshan, K. R., & Sahoo, Sarat. Kumar. (2013). Arduino Based Pulse Width Modulated Output Voltage Control Of A Dc-Dc Boost Converter Using Proportional, Integral And Derivative Control Strategy. Publisher: Australian Journal of Basic and Applied Sciences, 7(11), 104–108.
- Sivamani, D., Harikrishnan, R., & Essakiraj, R. (2018). Genetic Algorithm Based PI Controller for DC-DC Converter Applied to Renewable Energy Applications, 118(16), 1053–1071.
- Sowjanya, K. Lakshmi., & Srinivas, L. Ravi. (2015). Tuning of PID Controllers Using Particle Swarm Optimization, (2), International Journal of Industrial Electronics and Electrical Engineering, Vol.3 Issue 2, 17–22.
- Sugiono. (2015). Metode Penelitian Pendidikan (Pendekatan Kuantitatif, Kualitatif, dan R&D). Penerbit: Alfabeta, Bandung.
- Sumanthi, S., Kumar, L. Ashok., & P, Surekha. (2016). Intelligence Paradigms for Optimization Problems Using. London, New York: CRC Press : Taylor & Francis Group.
- Trzynadlowski, A. M. (2003). Power Electronic Converters. Control in Power Electronics. <https://doi.org/10.1016/B978-012402772-5/50002-8>