VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

Self Tuning Buck Converter

Niels Clayton

Supervisors: Daniel Burmester and Ramesh Rayudu

Submitted in partial fulfilment of the requirements for
Bachelor of Engineering with Honours.

Abstract

Switch-mode power supplies are commonly used in a wide variety of consumer and professional appliances to transform DC voltages with high efficiency. One such switch-mode supply is the buck converter, which steps down a DC voltage. The current buck converter design process requires that a specific output filter be designed around the switching frequency of the converter, the required output voltage, and the desired inductor ripple. This filter design process often results in the selection of non-standards components that are difficult to purchase or manufacture, increasing costs and leading to design compromises. This project will implement a control system to actively control the inductor current ripple by modulating the switching frequency of the converter. This will allow engineers to design buck converters directly for the inductor current ripple they wish to tolerate, eliminating the issues of designing this output filter.

Contents

Chapter 1: Introduction

Historically power distribution has been primarily in the form of AC (Alternating current). This is credited to the fact that AC power is more efficient to transmit over long distances and made it easy to step up and down the voltages efficiently with transformers [1]. However, with the invention of solid-state electronics such as the metal–oxide–semiconductor field-effect transistor or MOSFET for short, it has become possible to efficiently step up and step down direct current (DC) voltages. This has been achieved by the invention of the switch-mode power supply, which has facilitated the continued reduction of size and increase in efficiency of electronics [2].
Today, switch-mode power supplies can be found in a wide variety of consumer and professional electronics, with some examples being laptops, phones, and any form of DC charger. Their widespread usage when compared to other DC-DC converters such as linear regulators can be attributed to their far greater efficiency. One such switch-mode power supply is the buck converter, which will step down a DC input voltage to a lower DC output voltage.

1.1 Project Motivation

Although buck converters are a widespread technology, they are not without their limitations and drawbacks. The current buck converter design process requires that a specific output filter be designed around the switching frequency of the converter, and the desired inductor ripple. This filter design process will often result in the converter requiring discrete passive components that are non-standard and hard to source. This will usually result in the designer having to make compromises in their design for either the cost or the performance of the converter.
Another drawback of this design process is the static nature of both the filter and the switching components once they have been selected. This results in the buck converters desired inductor current ripple only being achieved at a very specific designed output voltage or load. This means that with current buck converter designs varying the desired output voltage or varying the output load will cause the inductor current ripple to vary. This is an issue, as very few loads are static and will not change during their operation.

1.2 Project Goals

This project aims to eliminate the need to design the output stage of a buck converter. By implementing a control system that varies the switching frequency of the converter, we will be able to directly manipulate the inductor current ripple. This project aims to produce a proof of concept buck converter that is capable of operating at 12V, with an output range of 3-10V and precision of $\pm 5\%$. The converter will also be able vary its switching frequency between $1kHz$ and $100kHz$, allowing for selection of inductor current ripple between 20% and 50% with precision of $\pm 5\%$. All of this must be implemented while maintaining the standard functionality of the converter. For full system requirements please refer to Appendix A.

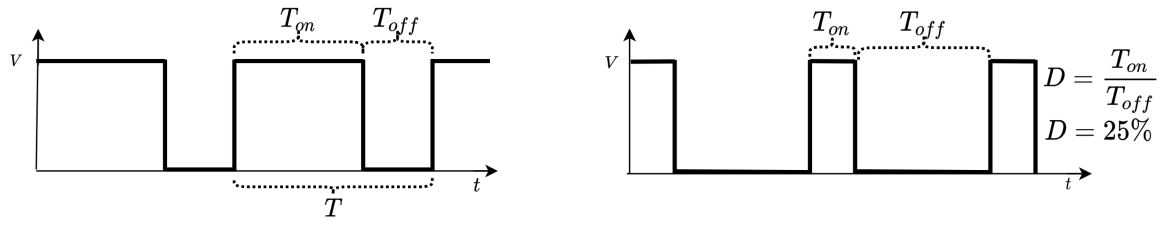
Chapter 2: Background

A literature research was performed to inform design decision made in this project, and to evaluate any existing research. It will discuss buck converter design factors and topologies, as well as the various different methods of PWM generation. In performing this literature research, we searched Google Scholar, Engineering Village, and Te Waharoa to find designs that utilised variable frequency PWM.
These searches returned no research relevant to the designs of this project, with the only related work focusing on the electromagnetic noise reduction using randomised frequency modulation [3, 4]. Because of this, research was instead performed to inform the design of the buck converter and the generation of PWM signals.

2.1 Pulse Width Modulated Signal Generation

Pulse width modulation (PWM) is a digital signal generation technique shown in Figure 2.1a, in which the Period T of the signal is held constant, while the ratio of its logic high period T_{on} to logic low period T_{off} is modulated. This ratio of high period to the low period is referred to as the duty cycle of the PWM signal and is often expressed as a percentage, this can be seen in Figure 2.1b.

PWM signals are used in a wide variety of applications for both digital and analogue electronics. PWM is often used to generate analogue signals from digital components by varying the average voltage of the digital PWM signal over time [5]. PWM is also used to control the switching elements contained within switch mode power supplies using this same principle, as discussed in Section 2.2. With regard to this project, we will be looking to generate a PWM signal that can be modulated in both duty cycle and frequency.



(a) Pulse width modulated signal

(b) Duty cycle calculation

Figure 2.1: Pulse width modulated signal characteristics

2.1.1 Analogue PWM Signal Generation

Designing a PWM signal generator using analogue components has three distinct stages required to generate the signal. These stages can be seen in Figure 2.2, and include clock

generation, triangle wave generation, and signal comparator stages [6].

The clock generation stage generates a square wave clock signal at a set frequency. This is usually done using a quartz crystal oscillator, or another form of resonating oscillator circuit. The triangle wave generating state must take the clock signal from the previous stage, and produce a triangle wave of the same frequency. This stage is most often done using a standard op-amp integrating circuit with unity gain at the resonating frequency of the clock source. The final signal comparator stage will convert this triangle wave into a PWM signal. Using a comparator, a refrence voltage can be applied to the non-inverting input, and then the triangle wave can be applied to the inverting input. This will produce a pulse train with the same frequency as the clock source, where the period of T_{on} and T_{off} is set by the refrence voltage.

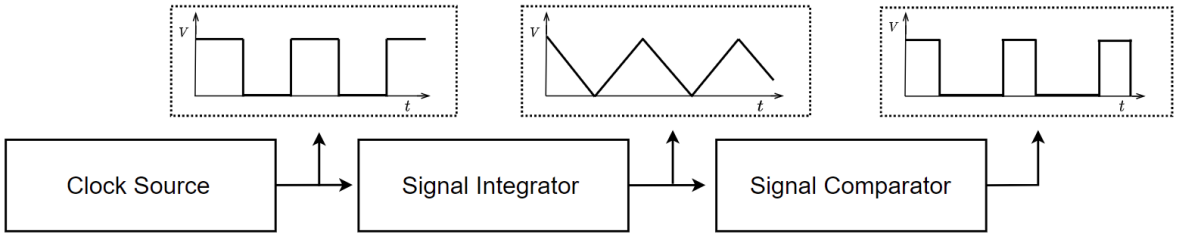


Figure 2.2: Stages of analogue PWM generation

2.1.2 Digital PWM Signal Generation

Designing a PWM signal generator with digital components is far more simple than the method described in Section 2.1.1, and can be done using either a microcontroller or a Field Programable Gate Array (FPGA). By using an internal timer that is continually incrementing at a known period we can set a period for our PWM. Then by toggling a digital I/O when a compare variable is equal to the value of the timer. we are able to generate a PWM signal with a variable duty cycle [7]. This can be achieved on most microcontrollers, however the maximum frequency and duty cycle accuracy will be dependant on individual clock speed a and internal register sizes.

2.2 Buck Converters

The buck converters is a variant of a switch mode power supply that steps down a DC input voltage to a DC output voltage. They are commonly used in a wide variety of consumer and professional appliances such as laptops, phones, and chargers due to their high efficiency compared to other DC-to-DC step down converters such as linear regulators [8].

The basic operational components of a buck converter can be seen below in Figure 2.3. From this we see that a buck converter has three main elements, the input voltage source, two switching components, and an output filter across the load. In the case of Figure 2.3, the first switching component is an actively controlled switch such as a MOSFET or transistor, and the second a passive switching diode. This configuration of an active and a passive switch is known as the non-synchronous buck converter topology, if the passive diode were to be replaced with a second active switch the topology would be considered synchronous. Although both topologies function under the same fundamental principles, the non-synchronous topology is easier to implement with the drawback of higher losses and

therefor lower efficiency.

It can also be seen from Figure 2.3 that a buck converter has two operating states that are controlled through the activation of these switching components. By toggling these switching components at high speed though the use of PWM, we can control the current flowing through the inductor of the output filter. By controlling this current we are also able to directly control the current through, and voltage across the output load of the converter. Using this, buck converters will often have a feedback control system in their design to be able to actively control and regulate the output voltage during usage. This controller will vary the duty cycle of the the switching PWM signal, thereby varying the output voltage of the buck converter as shown in Equation (2.1).

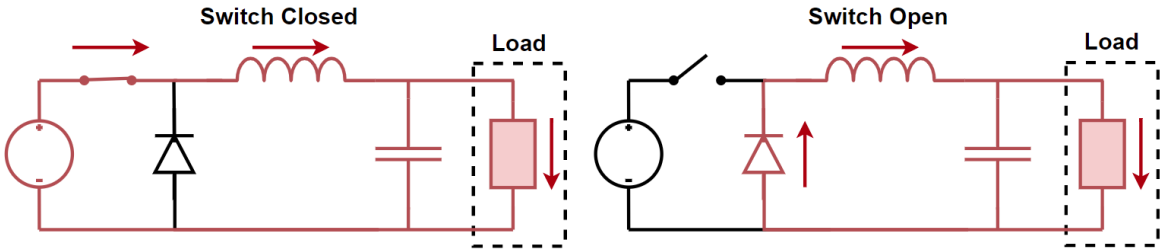


Figure 2.3: Operating states of a buck converter

2.2.1 Buck Converter Design

The design of a common buck converter has two primary considerations, the output voltage of the converter V_0, and the inductor current ripple of the converter Δi_L. These considerations can be specified by designing the buck converter using Equation (2.1) & Equation (2.2) [9, 10].

When designing a buck converter the first design specification that must be met is the output voltage. In Equation (2.1) the output voltage can be directly related to the input voltage V_{in} and the switching duty cycle D. Using this equation it is possible to directly set the output voltage of the buck converter by varying this duty cycle.

$$V_o = D \cdot V_{in} \tag{2.1}$$

Once the output voltage has been specified, the inductor current ripple can be calculated and specified with Equation (2.2). This equation allows for the inductor current ripple to be directly related to the inductor size L, and the PWM switching frequency f_s. This allows the the specification of the inductor current ripple through the varying of these two values.

$$\Delta i_L = \frac{V_o \cdot (1 - D)}{L \cdot f_s} \tag{2.2}$$

These two equations will be used to inform the designs and specifications of this project, and will be discussed in detail in Section 3.1.

Chapter 3: Work Completed

This chapter will discuss the work that has been completed so far on this project. It will begin by discussing the projects requirements', using them to design and justify the final specifications of the system. Once the specifications are outlined, the architecture and design of the final system will be discussed. Finally the design and current implementation of the PWM signal generator will be specified.

3.1 Defining & Justifying System Specifications

Based on the system requirements that have been outlined in Chapter 1, a set of system specifications can be created to inform the design. Current work on this project has been towards the implementation of the PWM signal generator, and as such all specifications outlined will pertain to this work.
The final system is required to be able to select both the output voltage and inductor current ripple of the buck converter. From this requirement we specify that our PWM generator must vary both the duty cycle and the switching frequency of the buck converter, based on Equation (2.1) & Equation (2.2). The requirements also specify the level of precision that will be required for these selections, allowing us to specify the tolerable error. Using these same equations, the minimum duty cycle step size in Equation (3.3), the inductor minimum and maximum values in Equation (3.5) & Equation (3.4), and the frequency step size in Equation (3.7) have all been derived. For the derivation of these values see Appendix B.

Duty cycle step calculation:

$$V_{error} = V_{min} \cdot error = 0.15V \tag{3.1}$$

$$D_{step} = \frac{V_{error}}{V_{in}} = 0.0125 \tag{3.2}$$

$$N_{step} = \frac{1}{D_{step}} = 80 \tag{3.3}$$

Inductor Sizing Calculations:

$$L_{max} = \frac{V_{max} \cdot (1 - D_{max})}{f_{min} \cdot I_{min}} = 27.7mH \tag{3.4}$$

$$L_{min} = \frac{\frac{V_{in}}{2} \cdot (1 - 0.5)}{f_{max} \cdot I_{min}} = 0.5mH \tag{3.5}$$

Frequency step calculation:

$$f_{step} = \frac{V_{max} \cdot (1 - D_{max})}{(I_{min} - I_{Error}) \cdot L_{max}} - f_{min} = 52Hz \tag{3.6}$$

$$N_{steps} = \frac{(f_{max} - f_{min})}{f_{step}} = 1881 \tag{3.7}$$

From these equations we can build a list of final specifications to inform the design of our PWM generator and buck converter. The PWM generator must provide a minimum voltage step size of $0.0125V$, for a resolution of 80 voltage steps between 3V and 10V. The PWM generator must also be able to provide a minimum frequency step size of $52Hz$, for a resolution of 1881 frequency steps between $1KHz$ & $100kHz$. Finally we can also specify that the buck converter must be capable of functioning with inductor values between $0.5mH$ & $27.7mH$.

By designing the PWM generator and the buck converter to these specifications, we are able to guarantee that we can always achieve the requirements outlined in Chapter 1.

3.2 System Architecture & Design

To achieve the specifications that have been outlined in Section 3.1, it is important to design the system architecture around them. In Figure 3.1 an overview of the system architecture can be seen, with three main design sections outlined. These sections each represent a significant segment of work that must be completed for the final artefact of this project to be achieved.

The first section of work that must be completed is the design of the PWM generation, denoted 1 in Figure 3.1. This PWM generator will be used to control both the output voltage and the inductor current ripple, and as such must be able to modulate both the duty cycle and the frequency of the PWM to the precisions required.

The second section of work is the design of the sensing elements required by the system, denoted 2 in Figure 3.1. These elements will be used to measure both the output voltage and the inductor current ripple, and therefor must be able to achieve the required precisions and sampling rates.

Finally the third section of work is the design and implementation of the two control systems, denoted 3 in Figure 3.1. These control systems will be responsible for maintaining the desired output voltage and inductor current ripple of the buck converter. This system will therefor be responsible for facilitating the final functionality of the project, combining sections 1 & 2.

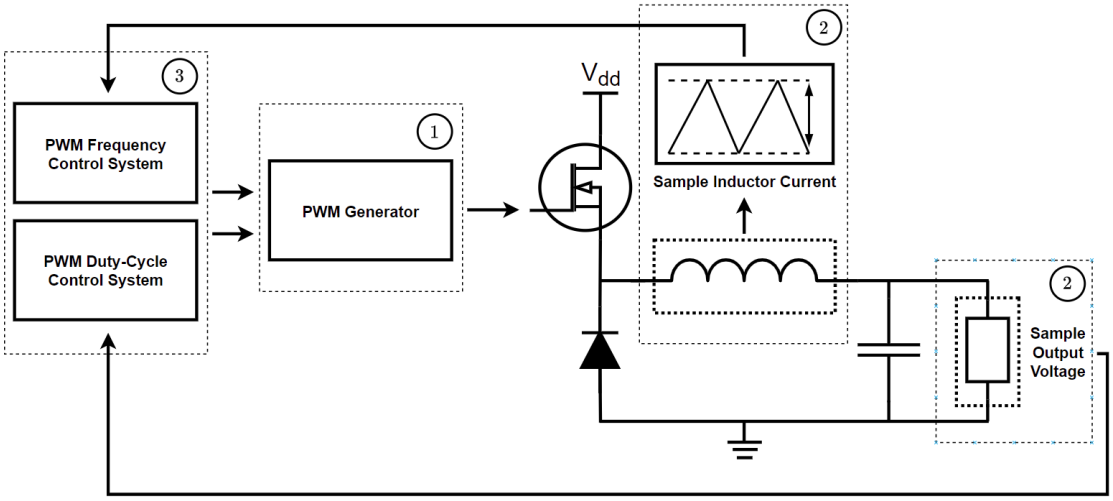


Figure 3.1: High level system overview

3.3 PWM Generation Design

In the design of the PWM generator, both the analogue and digital designs discussed in Section 2.1 were considered, designed, and tested for. Each of these designs presented pros and cons that would affects the overall design of the system architecture. This section will discuss these designs and finalise the design of the PWM generator for this system.

3.3.1 Analogue PWM Generator Design

As discussed in Section 2.1.1, the design of the analogue PWM signal generator requires three stages, each of which will have its own design requirements based on the specifications outlined in Section 3.1.

The clock generation stage will be responsible for setting the frequency of the final PWM signal. This specifies that the clock source have a variable frequency output range between $1kHz$ and $100kHz$, with a minimum step size of $52Hz$. Research was done on a variety of clock sources, looking at voltage-controlled oscillators (VCO's), signal generator IC's, and even the basic 555 timer. From this VCO's were identified to operate at much higher frequencies than those used in this project. It was also identified that signal generator IC's often require selections of passive components to operate effectively, increasing their complexity. For this reason the variable frequency 555 timer circuit was selected, as it provided the required specifications.

The next section designed was the signal integrator stage. This stage consisted of a basic op-amp integrator circuit, with a design requirement that it be able to integrate the clock signal across the frequency range required. This circuit was designed and implemented in an LTSpice simulation to evaluate it's performance, and can be seen in Appendix C. From this simulation it was noted that the integrator's frequency response was similar to that of a first order low pass filter, and greatly attenuated the integrated signal. For this reason it was decided that analogue PWM generation would not be implemented in this system, as it presented many issues.

3.3.2 Digital PWM Generator Design

As discussed in Section 2.1.2, The design of the digital PWM generator is far simpler than that of the analogue, and can be implemented in a wide variety of methods. In this project microcontrollers and FPGA's have been considered.
Based solely on the capabilities of the platform, the PWM generator would be best designed and implemented on an FPGA as it would allow for superior speed and precision. However FGPA design brings a lot of difficulties, primarily in the prototyping and testing stages. Because of this, due to the limited time from of this project, we have decided to implement this PWM design using a microcontroller.
The selection of the microcontroller is highly dependant on the clock frequency and design of the PWM peripherals, as it must be capable of achieving the specifications outlined in Section 3.1. A large selection of microcontroller datasheets were reviewed to identify their specifications, including AVR, STM8, Espressif, and teensy based microcontrollers. From this review it was decided that the ESP32 microcontroller would be best suited to this project [11]. This microcontroller is capable of outputting a maximum PWM frequency $125kHz$ with a duty cycle resolution of 9 bits (512 voltage steps). From here a short C program was written to test the PWM functionality of the ESP32, and it was confirmed that it met the required specifications. The source code and images of this PWM signal can be found in Appendix D.

Chapter 4: Future Plan

This final chapter will outline the work that is yet to be completed for this project, and how we plan complete it. It will also discuss how we plan to evaluate this system once this future work has been completed, outlining the changes to the evaluation that have come about since the project proposal. Finally this chapter will then present the project timeline for the remaining work and evaluation, breaking down this work into discrete blocks of time.

4.1 Work to be Completed

As discussed in Section 3.2, there are 3 main segments of work in this project. The first segment of work was the PWM generator, which has been outlined in Section 3.3 and has already been completed. The next two section were the specification and selection of the required sensing elements, and then the implementation of the control system that will make this project functional.

To complete the specification and selection of the sensing elements, a large amount of background research will need to be completed. This research will include the identifying of various ways that the inductor current ripple can be measured, as well as defining the required specifications of the sensors based on the system requirements. Once this research has been completed and components have been selected, it will be important to evaluate the functionality of the components, to verify that they do meet the required specifications. Finally once the sensing elements have been selected and tested, the creation of a printed circuit board (PCB) to hold the PWM generator, sensing elements, and buck converter will be completed. This PCB will be the final physical artefact that is created from this project.

Once this physical artefact has been generated, the only section left to be completed is the projects control system. This control system will be responsible for regulating both the inductor ripple current and the buck converter output voltage. To be able to successfully implement this section of the project I will again need to perform a section of background research. This research will inform the design and implementation of this controller within the microcontroller, helping to select the controller architecture and structure. Once the controller design has been specified, it will need to be implemented in software on the microcontroller, allowing for time to test the implementation.

4.2 System Evaluation

The evaluation of this system has not changed substantially since the project proposal, and is still based upon meeting the following selection of specifications.
However, the evaluation of the system will now be conducted using a range of load resistances, evaluating its performance with 10Ω, 15Ω, and 20Ω output loads, using a constant

supply voltage of 12V DC.

- The buck converter will be able to take input voltages up to 12V DC
- The buck converter must maintain the basic functionality outlined in eq. (2.1)
- The buck converter will have an output voltage range between 3V and 10V DC
- The output voltage accuracy will be within $\pm 5\%$ of the target output voltage
- The user will be able to define the inductor ripple between 20% and 50%
- The inductor ripple accuracy will be within $\pm 5\%$ of the defined inductor ripple
- The buck converter will have a switching frequency range of 1kHz to 100kHz
- The control system will have no steady state error

Since all of these evaluation metrics are highly qualitative, the evaluation of the final artefact will consist of a series of tests that will look to confirm the systems ability to achieve each of theses requirements.

4.3 Project Timeline

The Gantt chart in Figure 4.1 directly relates the remaining work set out in Section 4.1 to the time remaining for the project. For both the sensing element design and the control system, we believe that a large amount of background research will need to be completed before work is able to commence. This background research will inform and help reduce the time require to effectively design these sections. we plan to have all of the implementation and evaluation of the system completed before the end of the mid trimester break, as this will allow for a large amount of report writing time.

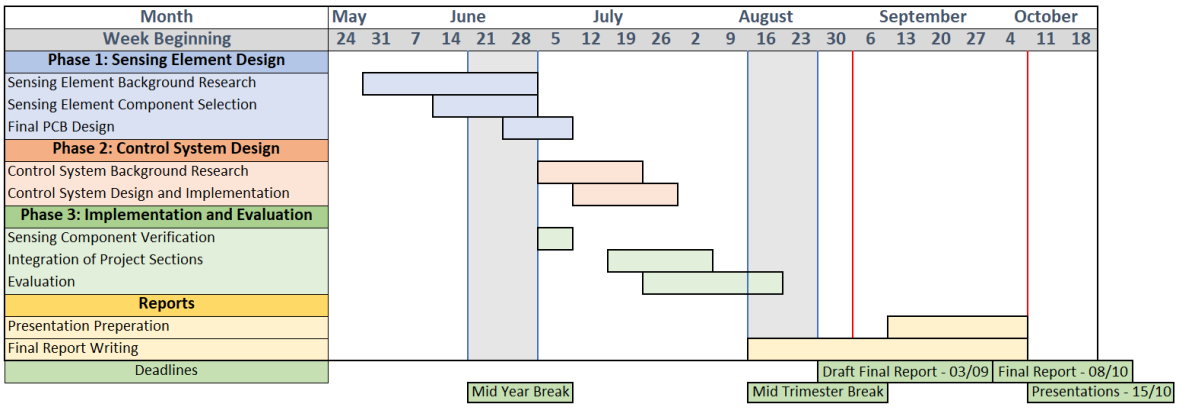


Figure 4.1: Proposed project timeline

Feedback

At this stage in the project I have a very clear idea as to what I am planning to, and how I am planning to get there. I would also greatly appreciate all feedback that can be provided on this report, so that I can implement the changes in my final report. I would greatly appreciate it if It would be possible to organise an in person meeting with my markers to discuss this report and project.

Bibliography

[1] E. Earley, "What's the difference between ac and dc?." Online, Sept. 2013.

[2] G. Bocock, "History of switch mode power supplies (smps)." Online.

[3] J. R. Roman, "PWM regulator with varying operating frequency for reduced EMI," Mar. 2001.

[4] Y. L. Familiant and A. Ruderman, "A Variable Switching Frequency PWM Technique for Induction Motor Drive to Spread Acoustic Noise Spectrum With Reduced Current Ripple," *IEEE transactions on industry applications*, vol. 52, no. 6, pp. 5355–5355, 2016.

[5] W. Tareen, M. Aamir, S. Mekhilef, M. Nakaoka, M. Seyedmahmoudian, B. Horan, M. A. Memon, and N. A. Baig, "Mitigation of power quality issues due to highpenetration of renewable energy sources in electricgrid systems using three-phase apf/statcomtechnologies: A review," in *Power Electronics in Renewable Energy Systems* (T. Suntio and T. Messo, eds.), 2019.

[6] J. Caldwell, *Analog Pulse Width Modulation*. texas instruments, June 2013.

[7] S. Colley, "Pulse-width modulation (pwm) timers in microcontrollers." Online, Feb. 2020.

[8] N. Mohan, *Power Electronics: A First Course*. Don Fowley, Oct. 2011.

[9] N. Mohan, *Power Electronics a First Course*, ch. 4: Switch Mode DC-DC Converters: Switching Analysis, Topology Selection and Design, pp. 38–68. Don Fowley, 2012.

[10] B. Hauke, "Basic Calculation of a Buck Converter's Power Stage," tech. rep., Texas Instruments, Aug. 2015.

[11] Espressif Systems, *ESP32: Technical Reference Manual*, 4.4 ed., 2021.

Appendix A: Project Proposal

VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

Self Tuning Buck Converter

Niels Daniel Clayton

Supervisor: Daniel Burmester, Ramesh Rayudu

Submitted in partial fulfilment of the requirements for
Honours of Electronic and Computer System Engineering.

Abstract

Switch-mode power supplies are commonly used in a wide variety of consumer and professional appliances to transform DC voltages with high efficiency. One such converter is the buck converter. The design of the buck converter requires specific components to design the output filter. These components be difficult to purchase or accurately manufacture. This project will implement a control system to actively control the switching frequency and eliminate the need design the output filter.

Contents

1 Introduction

1.1 Switch Mode Power Supplies

Switch-mode power supplies convert a DC input voltage to another DC output voltage. They are commonly used in a wide variety of consumer and professional appliances such as laptops and chargers due to their high efficiency compared to other DC-to-DC converters.

The step-down switch mode power supply also known as a buck converter, is a common DC-to-DC power converter that steps down an input voltage to the desired output efficiently. Currently, the design of these converters for specific applications requires a specifically designed output filter, designed around the switching speed of the converter. This filter will smooth the converter output voltage and maintain the inductor current ripple at the designed values. However this design process is not always applicable, if the specified components for the filter are not available there can be lead times, cost implications and delays.

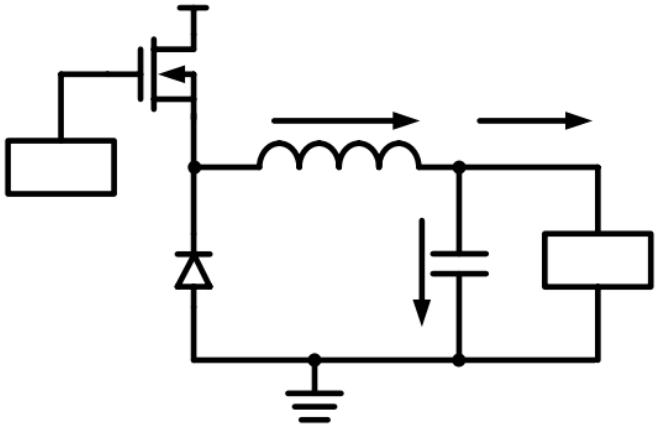


Figure 1: Asynchronous buck converter topology [1]

This project aims to remove the need for designing the output stage of the buck converter. Using a control system, the switching frequency and duty cycle of the converter can be modulated to meet a selected inductor ripple, while maintaining a selected output voltage. This project will focus on the design and implementation of this control system on a basic asynchronous buck converter.

2 The Problem

The current buck converter design requires that a specific output filter be designed around the switching frequency of the converter, and the desired inductor ripple. This filter design process will often result in the designer requiring discrete passive components that are not easily available. This requirement for precise discrete components can greatly increase the cost of the converter, and can also cause delays in manufacturing due to the lead time of components.

Another side effect of this design process is that this desired inductor ripple will only be achieved at the designed output voltage or load. This means that in current buck converter designs, varying the output voltage, or output load will cause the inductor ripple to vary. This can lead to the buck converter no-longer meeting the required specifications, making it hard to design a converter for a complex load.

This project aims to eliminate the need to design the output stage of a buck converter. This will be achieved by implementing a control system to vary both the switching frequency and duty cycle, to achieve the desired inductor ripple and output voltage. This will allow the buck converter to actively adjust the switching frequency to match the currently designed output filter at all times.

3 Proposed Solution

The proposed solution to this problem will have 3 phases, the initial research and setup, the design, and the implementation phase. Each phase will have a selection of tasks that must be completed to be able to progress to the following phase. The following subsections will go into detail on the undertakings of each phase. For time estimations of each task please look at the Gantt chart in Figure 2.

3.1 Research and Set-up

The first phase of this project is the research and setup. The major component of this phase is the creation of the literature review that will be the basis of this project. In this literature review, I will compare and contrast a wide range of methods available to produce a frequency variable PWM signal. This review will also be used to compare and contrast varying sensing methods for sampling the inductor ripple and the output voltage.

When the literature review is complete, I will select the methods of PWM generation and output sensing that are most suitable for this project. This will help me identify a range of components that I believe will be feasible for the design. These components will be purchased for used in the design phase.

Finally, once the components have been purchased I will design a selection of development PCB's that will allow for simplistic interfacing with the purchased components and the existing buck converter design. This will help expedite the design phase.

3.2 Design

In the design phase of the project, I will test and evaluate all of the selected PWM generation methods, and sensing methods. This will involve a large amount of prototyping on breadboards, along with possibly embedded programming on microcontrollers, and FPGA (Field Programmable Gate Array) design. The method of evaluation of these designs will be decided upon in the literature review.

Once the evaluation of the designs is complete, I will make the final hardware selection. The components selected at this stage will be used to implement the closed-loop controller for the converter.

3.3 Implementation

Finally, in the implementation phase of the project, the finalised hardware design will be used to implement the closed-loop control system. This system will sample the inductor ripple and vary the switching speed and duty cycle of the buck converter to reach a requested ripple value. A Simulink simulation may be designed to help with the implementation and design of this controller.

As the controller is being implemented, I will also design a final PCB for the buck converter. This PCB will include the entire buck converter design, with female headers for easy filter inductor and capacitor swapping.

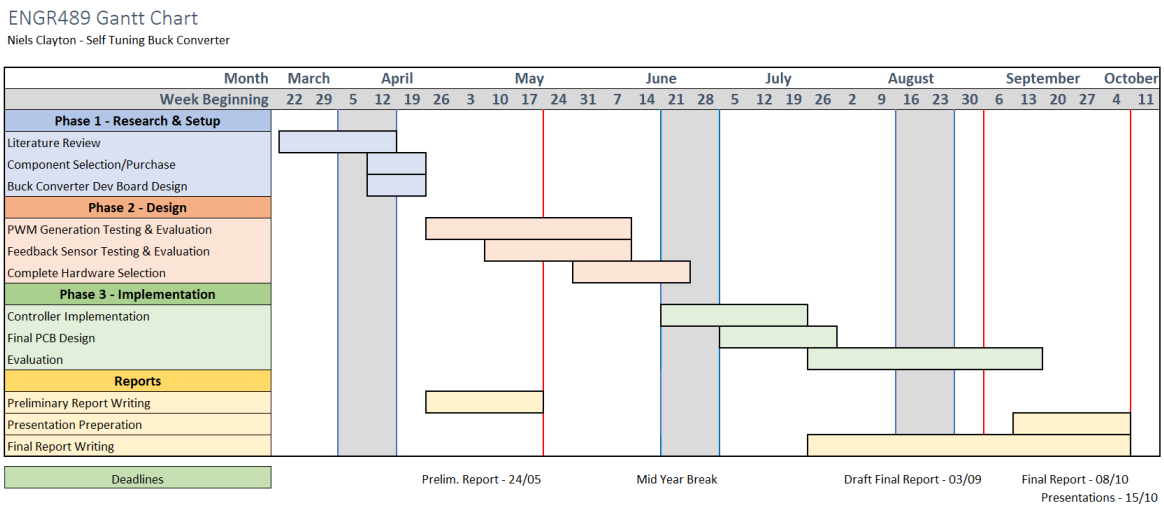


Figure 2: Project Gantt Chart

4 Project Evaluation

The evaluation of this project will be based upon meeting the following selection of specifications. All evaluations of the design will be conducted using a 10Ω output load, with an input voltage of 12V DC.

- The buck converter will be able to take input voltages up to 12V DC
- The buck converter maintains basic functionality of $V_{out} = D \times V_{in} -$ effeciency
- The buck converter will output voltages between 3V and 10V DC
- The output voltage accuracy will be within $\pm$5% of the target output voltage
- The user will be able to define the inductor ripple between 20% and 50%, with increments of 5%.
- The inductor ripple accuracy will be within $\pm$5% of the defined inductor ripple
- The buck converter will have a switching frequency range of 1kHz to 100kHz

5 Safety and Resourcing

5.1 Safety Considerations

Below in Table 1 there is a safety hazard identification, prevention, and mitigation table.

Potential Hazards	Hazard Prevention	Hazard Mitigation
Burns due to resistive elements heating during use.	Ensure that no more than 5W of power is dissipated across the load at any time.	Turn off equipment that is overheating.
Cuts due to misuse of wire cutters	Ensure that there are no distractions when operating potentially hazardous equipment.	Bandage the cut, and fill out an accident/incident report.
Burns due to misuse of soldering iron	Ensure that there are no distractions when operating potentially hazardous equipment. Make sure to correctly store the soldering iron after use.	Apply ice to the burn and fill out an accident/incident report.
Dangerous voltages	Ensure that the power supply never outputs more than the maximum of 50V.	Turn off that is not operating safely

Table 1: Hazard assessment

5.2 Required Equipment

The Following list of equipment is currently available without purchase, but will be required for this project:

- Lab bench power supply
- Lab bench oscilloscope
- Lab bench signal generator
- lab bench multi-meter
- Prototyping breadboards
- Passive components (resistors, inductors, capacitors, MOSFETs)

5.3 Budget

Below in Table 2 is the total estimated cost of the components I will need to purchase for this project.

Component	Description	Cost
PCB Manufacturing	In house or external PCB manufacturing	$50
Assorted components	Parts to be tested and evaluated as discussed in section 3.2	$200

Table 2: Budget

5.4 Covid-19 Considerations

Due to the effects of Covid-19 lock-downs, it is important to be prepared for the possible implication of going to the varying alert levels will have.

In the case of alert level 2, the university will still be open and operating with enforced social distancing and masks. This alert level will have little to no effect on the progress of this project, as the honours lab allows for social distancing.

In the case of alert level 3 & 4, the university will close. This means that I will not be able to access the honours lab to get to my equipment. In the case of alert level 3 or 4, I will have to ask the technicians if I am able to bring home with me the equipment listed in section 5.2 to continue working from home. If this is not possible then I will work to make an accurate simulation of the Project within Simulink.

Bibliography

[1] A. Jain, "Synchronous vs. Aynchronous Buck Regulators," tech. rep., Semtech Corporation, Camarillo, CA.

Appendix B: System Specification Derivation Equations

The system specification derivation equations have been input into the graphing platform `Desmos.com`. This has allowed me to visually inspect these equations and form conclusions. The full working and step by step derivation is also available on `Desmos.com` using the following links.

Duty cycle equation derivation:
`https://www.desmos.com/calculator/8c7wmbyzw4`

Inductor sizing equation derivation:
`https://www.desmos.com/calculator/v7ntrescw5`

PWM frequency equation derivation:
`https://www.desmos.com/calculator/ekjhcrt9zg`

Appendix C: Analogue PWM Generation

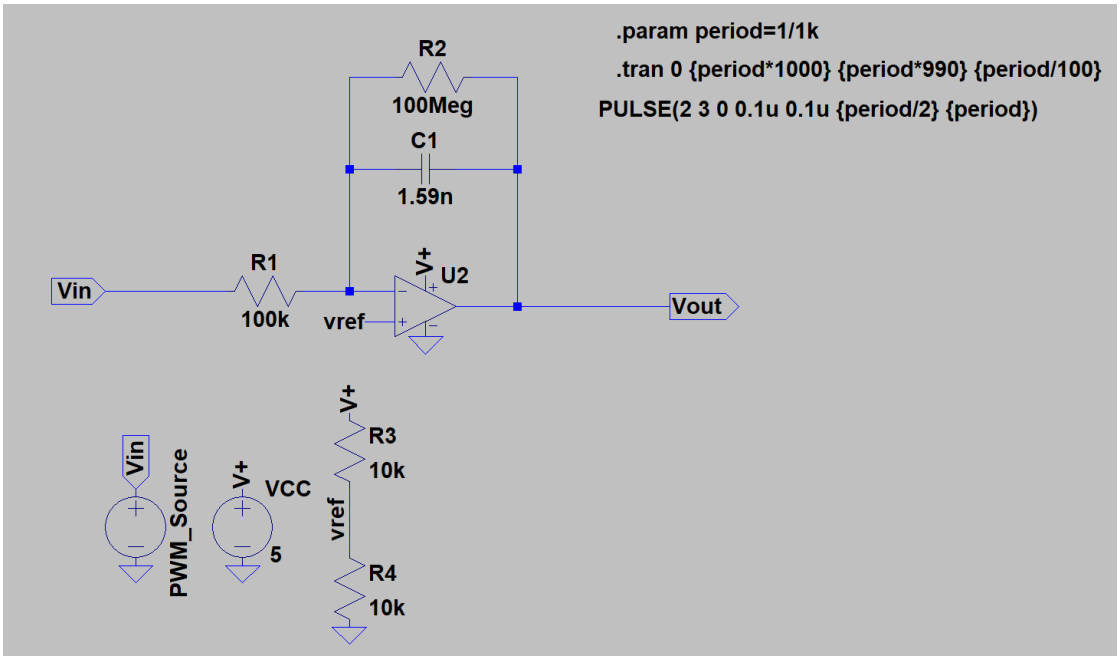


Figure C.1: Analogue PWM LTSpice circuit

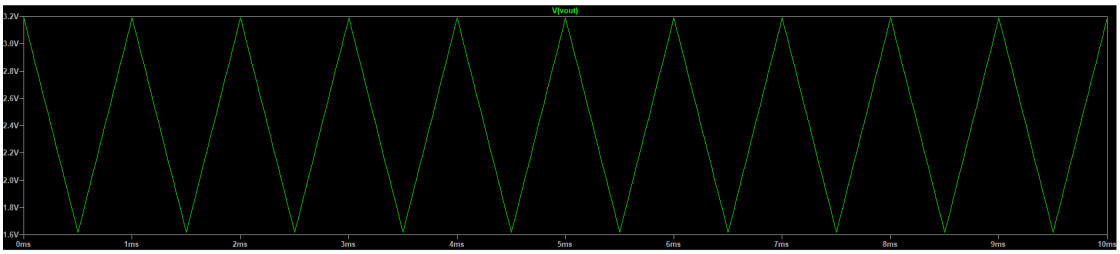


Figure C.2: Analogue PWM LTSpice simulation $1kHz$

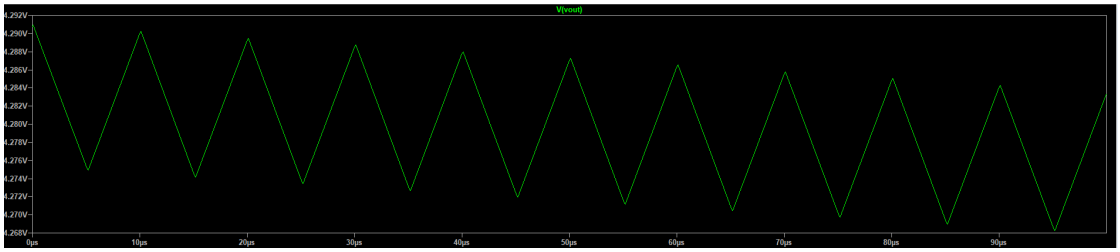
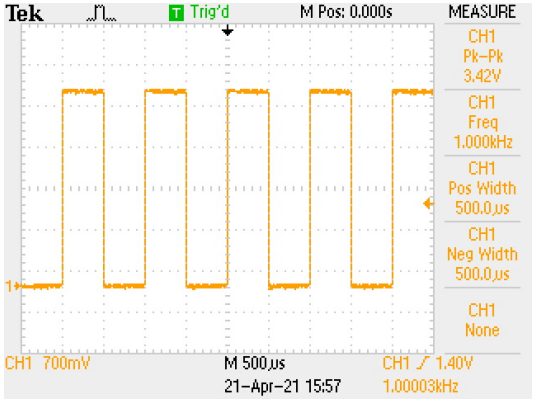


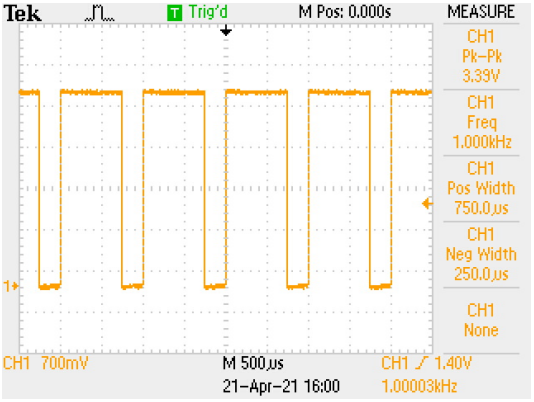
Figure C.3: Analogue PWM LTSpice simulation $100kHz$

Appendix D: Digital PWM Generation

D.1 Digital PWM Generation Figures

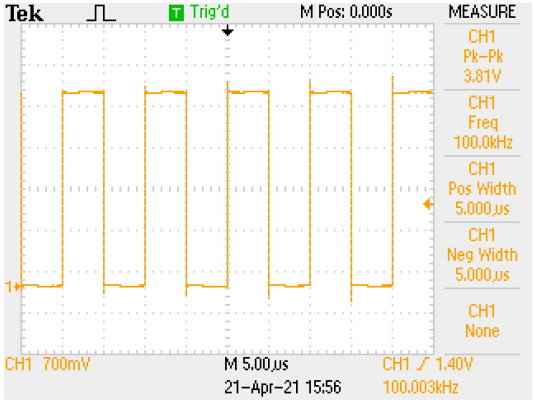


(a) Digital PWM Generation at $1kHz$ and a 50% duty cycle

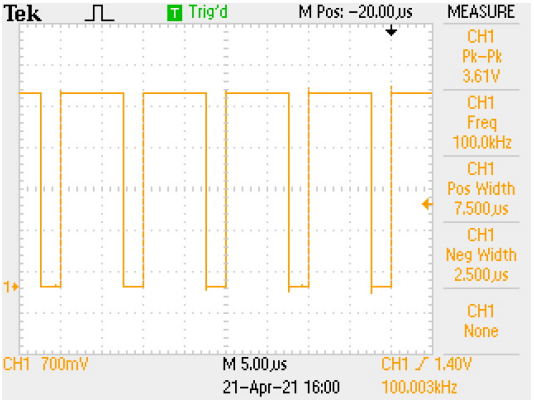


(b) Digital PWM Generation at $1kHz$ and a 75% duty cycle

Figure D.1: Digital PWM Generation at $1kHz$



(a) Digital PWM Generation at $100kHz$ and a 50% duty cycle



(b) Digital PWM Generation at $100kHz$ and a 75% duty cycle

Figure D.2: Digital PWM Generation at $100kHz$

D.2 Digital PWM Generation Code

```
1
2 #include <stdio.h>
3 #include "freertos/FreeRTOS.h"
4 #include "freertos/task.h"
5 #include "driver/ledc.h"
6 #include "esp_err.h"
7
8 // #define DUTY_TEST
9 #define FREQUENCY_TEST
10
11 #define DUTY_RESOLUTION LEDC_TIMER_9_BIT
12 #define FREQUENCY_MIN 1000
13 #define FREQUENCY_MAX 100000
14
15 #define DUTY_STEPS 512
16
17 void app_main()
18 {
19
20     const TickType_t xDelay = (10 / portTICK_PERIOD_MS);
21
22     ledc_timer_config_t ledc_timer = {
23         .duty_resolution = DUTY_RESOLUTION,     // resolution of
              PWM duty
24         .freq_hz = 100000,                      // frequency of PWM
              signal
25         .speed_mode = LEDC_HIGH_SPEED_MODE,     // timer mode
26         .timer_num = LEDC_TIMER_0,              // timer index
27         .clk_cfg = LEDC_AUTO_CLK,               // Auto select the
              source clock
28     };
29
30     ledc_channel_config_t ledc_channel = {
31         .channel    = LEDC_CHANNEL_0,
32         .duty       = 2*DUTY_STEPS/4,
33         .gpio_num   = 23,
34         .speed_mode = LEDC_HIGH_SPEED_MODE,
35         .timer_sel  = LEDC_TIMER_0
36     };
37
38     ledc_timer_config(&ledc_timer);
39     ledc_channel_config(&ledc_channel);
40
41     while (true)
42     {
43         #ifdef DUTY_TEST
44         for(int duty = 0; duty < 255; duty++){
45
```

```
46              ledc_set_duty(ledc_channel.speed_mode,
                    ledc_channel.channel, duty);
47              ledc_update_duty(ledc_channel.speed_mode,
                    ledc_channel.channel);
48              vTaskDelay(xDelay);
49          }
50          #endif
51
52          #ifdef FREQUENCY_TEST
53
54          for(int frequency = FREQUENCY_MIN; frequency <=
                FREQUENCY_MAX; frequency = frequency+10){
55
56              esp_err_t status =
                    ledc_set_freq(ledc_channel.speed_mode,
                    ledc_timer.timer_num, frequency);
57              printf("PWM frequency: %d -> %d\n",
                    ledc_get_freq(ledc_channel.speed_mode,
                    ledc_timer.timer_num), status);
58              // vTaskDelay(xDelay);
59          }
60          #endif
61      }
62  }
```

code/main.c