12-2012

Topics on Light-Emitting-Diode Driver Research

Jaber Hasan
University of Arkansas, Fayetteville

TOPICS ON LIGHT-EMITTING-DIODE DRIVER RESEARCH

TOPICS ON LIGHT-EMITTING-DIODE DRIVER RESEARCH

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Electrical Engineering

By

Jaber Hasan
University of Arkansas
Bachelor of Science in Electrical Engineering, 2003
University of Arkansas
Master of Science in Electrical Engineering, 2006

December 2012
University of Arkansas

ABSTRACT

In this dissertation, light-emitting-diode (LED) drivers are investigated for efficiency issues related to driving Red-Green-Blue (RGB) pixels and multiple LED strings in parallel. A high-efficiency digitally controlled RGB LED driver was designed for driving a 3x3 RGB LED display panel. A multiplexer was used to sense the voltage drop across the current controllers. This driver maintained a minimum drive voltage across the RGB LED pixels required to keep it in regulation leading to a reduction of unwanted power losses in the RGB LED pixels by selecting the minimum drop across the current-controllers as the reference voltage of the digital controller. Additionally, analog dimming was implemented to dim each individual LED in a 3x3 RGB LED display panel. Efficiencies of 85.6 %, 93.3 %, and 91.1 % were experimentally obtained for red, green, and blue LEDs, respectively at the rated output current. For comparison, efficiencies of 38.3% for the red LED, 66.2 % for the green LED, and 64.5 % for the blue LED at the nominal current of 20mA using a 5 V supply were achieved in conventional LED drivers. For parallel connected LED strings, a current controller is required for each string to maintain the desired current with the drive voltage provided by a switch-mode power converter (SMPC). A novel analog/digital LED driver controller was designed for driving a two-string LED load with three white LEDs in each string for backlight application in liquid-crystal-displays (LCDs). In this implementation, an analog controller was used to control the output voltage of the SMPC while a digital controller was used to achieve the minimum drive voltage across the output LED strings, leading to an efficiency of 89 % at the rated output current. A phase shifted pulse width modulation (PSPWM) dimming was implemented to reduce load stresses, improve electromagnetic-interferences (EMI) and increase system efficiency. A mathematical model, based on multirate simulation technique, for merging the analog and digital controllers was

proposed. Finally, a LED driver chip was designed and fabricated using a 0.5 micron CMOS process to reduce size of the overall LED driver system. The integrated circuit consisted of a 16-channel analog multiplexers, five current controllers, and buffer circuits. The LED driver chip function was experimentally verified using a SMPC to drive a five-string LED load in parallel with a single green LED in each string for display panel applications by maintaining a minimum drive voltage across the LED strings, thus leading to an efficiency of 75 % at the rated output current.

This dissertation is approved for recommendation
to the Graduate Council.

Dissertation Director:

_____

(Dr. Simon S. Ang)

Dissertation Committee:

_____

(Dr. Randy L. Brown)

_____

(Dr. Scott Smith)

_____

(Dr. H. Alan Mantooth)

_____

(Dr. Jia Di)

DISSERTATION DUPLICATION RELEASE

I hereby authorize the University of Arkansas Libraries to duplicate this thesis when needed for research and/or scholarship.

Agreed _____
(Jaber Hasan)

Refused _____

ACKNOWLEDGEMENTS

I would like to thank Dr. Ang for his encouragement and support towards completion of my work. I enjoyed working for him. I believe we did some good and interesting work related to LED drivers. I also like to thank Dr. Brown, Dr. Smith, Dr. Mantooth, and Dr. Di for taking the time to serve in my committee.

I would like to thank my father and mother for their support and love. I am grateful to them for their encouragements and their faith in me. I would also like to thank my elder sisters Ishaya and Sarah for their support.

DEDICATIONS

Dedicated to

my parents

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

In 2006, the residential and commercial sectors in the US consumed 2,652 billion kWh of which 39 % was for lighting loads [1]-[2]. It is clear from the data collected in 2006 that the lighting energy consumption is expected to steadily increase in the future. Thus, there is a trend towards developing and promoting energy efficient alternatives for lighting applications. Research is being conducted in developing new high efficiency lighting sources such as LEDs and organic LEDs.

In recent years, LEDs are being introduced in liquid crystal displays (LCDs) backlight, display panels, signage, streetlights, automobiles, traffic lights, and general-purpose lighting due to advancement in LED technology and reduced energy consumption [3]-[8]. In comparison to traditional light sources such as incandescent lamps and cold cathode fluorescent lamps (CCFLs), LEDs have longer lifetime, in the range of 80,000 – 100,000 hours [3]-[4]. LEDs are encapsulated from its operating environment, making it more reliable [5]. The absence of mercury makes LEDs environmentally friendly as they can be disposed of safely [3]-[4]. Additionally, LEDs have wider color gamut, higher luminance, and smooth-dimming properties [3]-[8].

LEDs are current-driven devices, i.e. their brightness is proportional to its current [3]-[7]. Due to limitation in LEDs packaging, the output power of each individual LED is less than 20 W [5]. Therefore, in order to obtain the same luminance as other traditional light sources, many LEDs has to be connected in parallel or series. But there is a limit on the number of LEDs that can be in

series due to limitations on its drive voltage. Similarly, the drawback of LED strings connected in parallel is its current sharing between the strings leading to lower lifetime for the LEDs due to its exponential voltage-current relationship [3].

There are several methods of driving LEDs connected in parallel. The straightforward approach is to have individual constant current drivers for each string in parallel as shown in Fig. 1.1 [3]-[7]. The constant current driver can either be a SMPC or linear voltage regulator. A linear constant current driver is cost effective but it suffers from power losses in its series-pass transistors [9]. However, SMPC constant current drivers are much more efficient but they require storage devices such as capacitors, and inductors for voltage conversion leading to higher parts count and increased cost. Moreover, this approach of employing individual constant current driver for each LED string in parallel is not cost effective for drivers that have large number of LED strings in parallel as in LCD backlight, display panel, and general-purpose lighting applications.

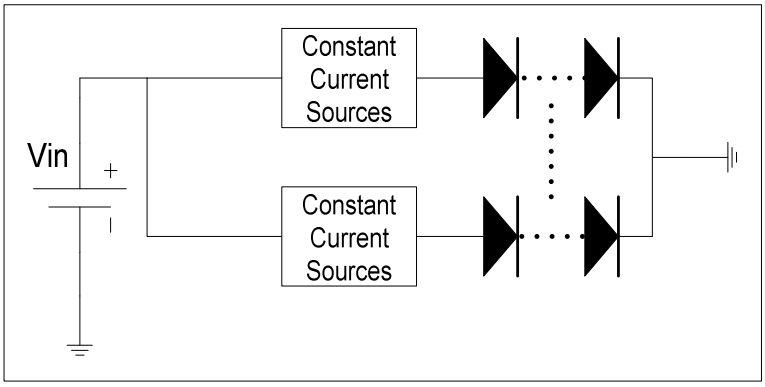


Figure 1.1 A LED driver with individual constant current driver for each LED strings [3]-[8].

For driving LEDs strings in parallel, a different approach is usually employed which uses current controllers for each string in parallel for maintaining the current in the strings and the drive

voltage is provided by the SMPC as shown in Fig. 1.2. In this implementation, the output voltage of the driver is set at its worst-case by the feedback voltage in order to maintain all the current controllers in regulation [3]-[7]. Therefore, in order to improve the efficiency of the output LED strings in this approach, matched forward voltage LEDs are used which adds to the cost of the driver [3]-[7].

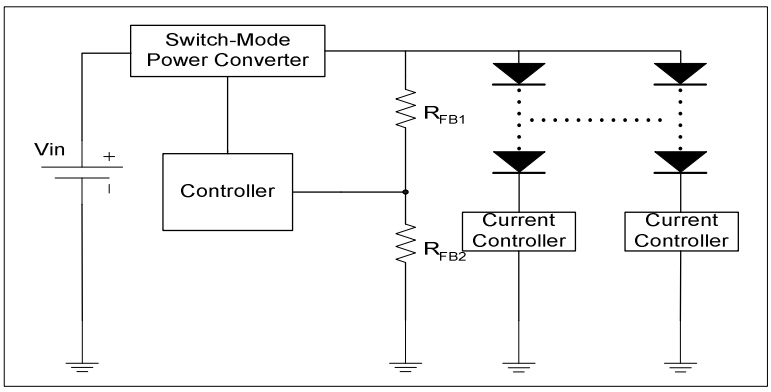


Figure 1.2 A LED driver with SMPC using individual current controllers for each LED strings in parallel.

The main objective of this dissertation research is to design efficient LED drivers to drive multiple LED strings in parallel. A high-efficiency digitally controlled Red-Green-Blue (RGB) LED driver was designed to drive a 3x3 RGB LED display panel to demonstrate the process of maintaining minimum drive voltage across RGB LED strings to reduce unwanted power losses in the LED strings. Additionally, analog dimming capabilities are implemented for each individual LED in a 3x3 RGB LED display panel. Efficiencies of 85.6 %, 93.3 %, and 91.1 % were experimentally verified for the red, green, and blue LEDs, respectively at the rated output current. An analog/digital LED driver controller was designed to drive a two-string LED load with three white LEDs in each string for backlight application in LCDs. In this implementation, an analog/digital controller was designed to maintain a minimum drive voltage across the output LED strings, leading to an efficiency of 89 % at the rated output current. A phase shifted pulse

width modulation (PSPWM) dimming was implemented to reduce load stresses, improve EMI and increase system efficiency. A LED driver chip was designed and fabricated using a 0.5 micron CMOS process to reduce the size of the overall LED driver system. The integrated circuit consisted of a 16-channel analog multiplexers, five current controllers, and buffer circuits. The LED driver chip function was experimentally verified using a SMPC to drive a five-string LED load in parallel with a single green LED in each string for display panel applications by maintaining minimum drive voltage across the LED strings, thus leading to an efficiency of 75 % at the rated output current.

1.2 RESEARCH CONTRIBUTION

The contributions of this dissertation are:

1) A new digitally controlled RGB driver was developed for display panel applications. In this driver, sensing of the gate and drain voltages of the MOSFETs of the current controllers are performed using multiplexers instead of sensing diodes as in [6]-[7] leading to a better efficiency of the driver. The proposed driver has two modes of operation, efficiency optimization mode and operation mode in order to maintain a minimum drive voltage across the LED strings to reduce unwanted power losses in the current controllers of the output LED strings. The fast transient response of LED currents, dynamic adjusting of drive voltage, good configurability, high efficiency, analog dimming, and low cost makes this an attractive solution for driving RGB LEDs in display panels.

2) A new LED driver using an analog/digital controller was developed for LCD backlight applications. In this driver, the power dissipation in current controllers was reduced using minimum drive voltage for the LED string. An improvement of 21 % in efficiency in the output

LED strings was observed by implementing a dynamic adjustment of the drive voltage. Additionally, PSPWM dimming was implemented to reduce load stresses, improve EMI, and increase system efficiency.

1.3 DISSERTATION OVERVIEW

The dissertation is divided into six chapters and the overview of each chapter is summarized below:

Chapter 2 presents the motivation for this dissertation. A brief summary of LED structures is presented. Some of the commonly used power supply drivers and its control circuitry are described. A review of available LED drivers is also included.

Chapter 3 provides the design process and implementation of a RGB LED driver for display panel applications. An efficiency improvement based on dynamic adjustment of drive voltage is proposed in this chapter. The adjustment of light output for each LED pixel using analog dimming with daisy chained digital potentiometers is also presented. A MATLAB SIMULINK model is designed to verify the performance of this proposed driver. Experimental results are provided for a 3x3 RGB panel.

Chapter 4 introduces an analog/digital LED driver controller for LCD backlight applications. The proposed driver maintains minimum drive voltage across the LED strings, leading to higher efficiency in the output LED strings. A dimming approach based on PSPWM is implemented. A mathematical model for merging the analog and digital controller is introduced. A MATLAB SIMULINK model is designed to verify the performance of this proposed driver. Experimental results are provided for an analog/digital driver for a two-string LED load.

Chapter 5 presents the architecture for a LED driver IC in display panel application. A detailed description of the circuits operation and the layout of these circuits will be provided. Experimental results obtained for a complete LED driver system using the fabricated IC to drive a five-string LED load will be presented.

Chapter 6 concludes the dissertation with major contributions and suggestions for future work.

CHAPTER 2

BACKGROUND

2.1 INTRODUCTION

A LED module consists of the LED load for generating the light and a compatible power supply unit for driving the LED load. A regulated SMPC is required in order to maintain a stable operation of the LED load over varying input voltage and temperature. So, in order to maximize the efficiency of the LED module, it is necessary to optimize the design of the required SMPC and current controller circuits.

This chapter is divided into three sections. The first section discusses the physical structure and operation of the LED load. Some of the commonly used power supply drivers and its control architecture are then introduced. The final section summarizes the most relevant research related to achieving high efficiency at the parallel connected LED load, which is the motivation of this dissertation.

2.2 LED LOAD

LEDs are semiconductor devices that emit light [10]. Currently, LEDs produce more lumens per watt and are more efficient than other traditional light sources such as incandescent lamps. Additionally, the faster response time, cost-effectiveness, low power consumption, and longer lifetime have made LEDs an attractive lighting solution.

2.2.1 STRUCTURE AND OPERATION

Figure 2.1 shows the structure of a GaAsP LED [10]. The N region of the LED is formed by the $GaAs_{0.6}P_{0.4}$ layer which is deposited on a GaAsP substrate. The diffusion of zinc into the GaAsP

material forms the P region. When LEDs are forward biased, electrons injected from the N region into the P side of the junction recombine with holes in the P region leading to generation of photons which produces light. Similarly, recombination occurs on the N side of the junction with electrons and excess holes injected from P region causing radiative recombination.

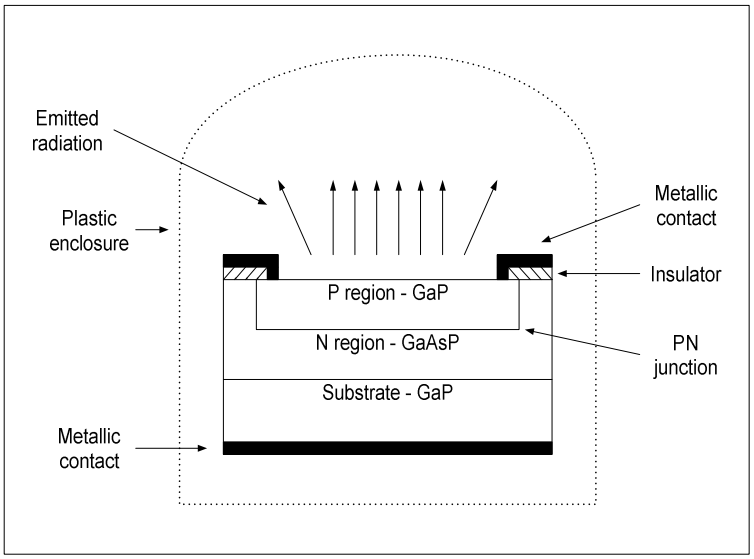


Figure 2.1 Physical structure of a LED [10].

Radiative recombination occurs in direct bandgap compound semiconductors such as GaAs. In this semiconductor, recombination of electrons and holes occurs in the lower valence bands causing energy to be released in the form of light [10]. In indirect bandgap semiconductors, such as Ge and Si, recombination of electrons and holes occurs in conduction bands causing energy to be released in the form of heat [10]. So, in order to produce light output from indirect bandgap compound semiconductors certain dopants must be added, for example nitrogen is added to gallium phosphide (GaP) for the generation of green color.

Electromagnetic radiations occur in LEDs because of the difference in the energies of the bottom of the conduction band and the top of the valence bonds, i.e., the wavelength of radiation is given

by:

$$\lambda = \frac{hc}{E_g} \tag{2.1}$$

where h is Planck's constant in J-s, c is the velocity of light in m/s, E_g is the bandgap energy in J, and λ in μm [10]. Typically, E_g is in the range of 1.8 and 2.8eV so that the emitted radiation is in the visible spectra [10]. Spectral response of various semiconductor compounds is shown in Fig. 2.2.

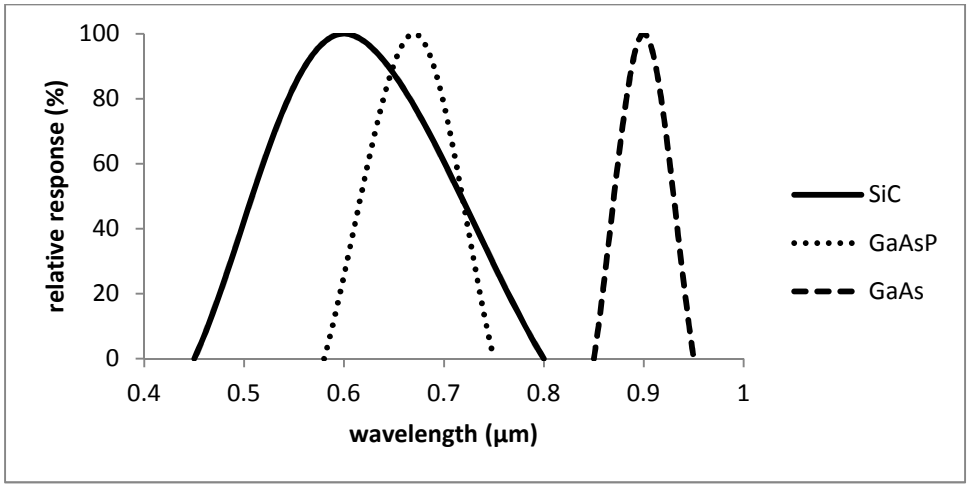


Figure 2.2 Spectral responses of various compounds [10].

2.2.2 V-I CHARACTERISTICS

The terminal characteristics of a LED can be defined by:

$$I_D = I_S e^{\frac{V_f - I_D R_S}{nV_T}} \tag{2.2}$$

where I_D is the LED forward current, I_S is the saturation current, V_f is the forward voltage drop across the LED terminals, R_S is the series resistance, and V_T is a constant called the thermal voltage [1].

Typical V-I characteristics of a white LED is shown in Fig 2.3. Because of the material, the V-I characteristics of LEDs created in the same batch have large variations, thus LEDs are separated according to wavelength, luminous intensity and forward voltage [1]. The variation in LED forward voltage is one of the important factors to be considered in the design of high efficiency drivers for LED strings. This is because some LEDs in a string require lower forward voltages to maintain the desired current in the string. Furthermore, variations in luminous intensity of LEDs affect the white color point in a RGB LEDs; this requires an additional feedback circuitry. Additionally, the luminous intensity and forward voltage of LEDs decrease with increasing junction temperature and over time.

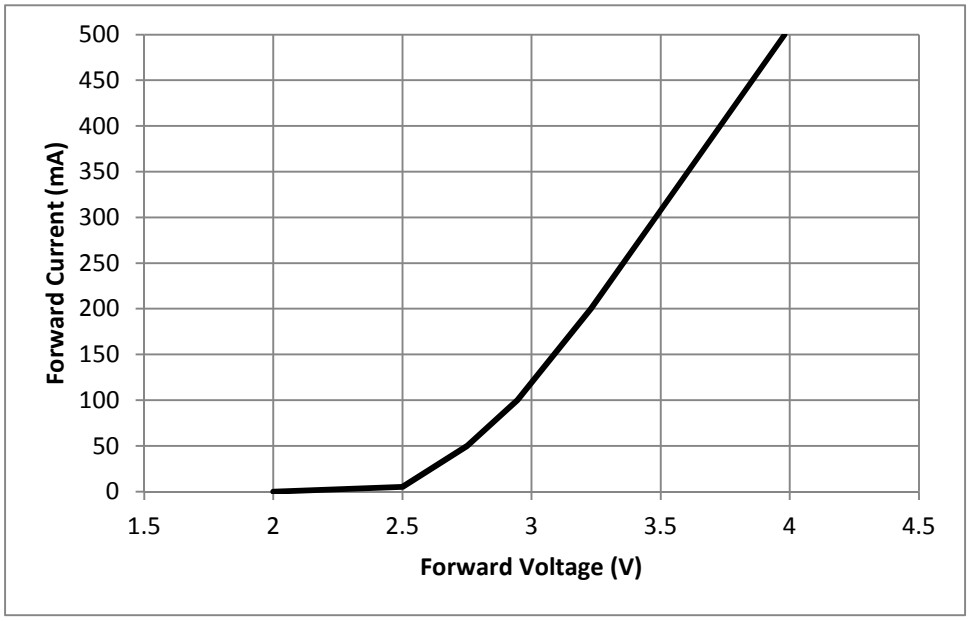


Figure 2.3 V-I characteristics of a white LED [11].

2.3 POWER SUPPLIES

The simplest way to drive a LED string from a DC voltage is by using a current limiting resistor as shown in Fig. 2.4. In the circuit shown in the figure, the efficiency is very poor due to the

power dissipation in the series connected resistor. Additionally, since the forward voltage of the LED changes with temperature and over time, it is difficult to maintain a constant current in the LED strings using this architecture. Thus, other alternatives like the SMPC and linear regulators are considered in this section for driving the LED strings.

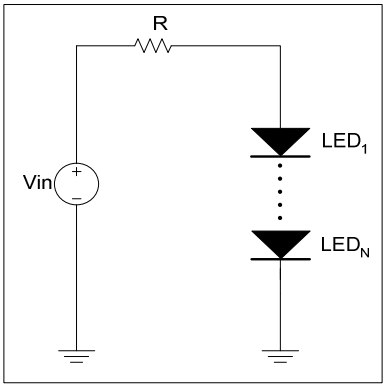


Figure 2.4 A simple LED driver using a series connected resistor.

2.3.1 LINEAR REGULATORS

A linear low-side driver using either a NPN or NMOS transistor LED driver circuit is shown in Fig. 2.5. In this circuit, a constant current is maintained in the LED strings irrespective of LED temperature and usage, by sensing the voltage across the sense resistor, R_{SENSE}, and comparing the sensed voltage with the reference, V_{REF}. The error amplifier (EA) outputs the error voltage between the sensed voltage across R_{SENSE} and the reference (V_{REF}) so that the output of the EA regulates the transistor (Q_1) to the level necessary to maintain the desired current in the LED strings.

Linear regulators are cost effective solutions for driving LED strings since they require no additional storage elements such as inductors or capacitors. However, for parallel connected LED strings, each string would require its individual linear regulator adding to the cost of the

overall system. Additionally, the loss across the transistor of the linear regulator would also result in low efficiency drivers. Thus, SMPC drivers are used for high efficiency implementation.

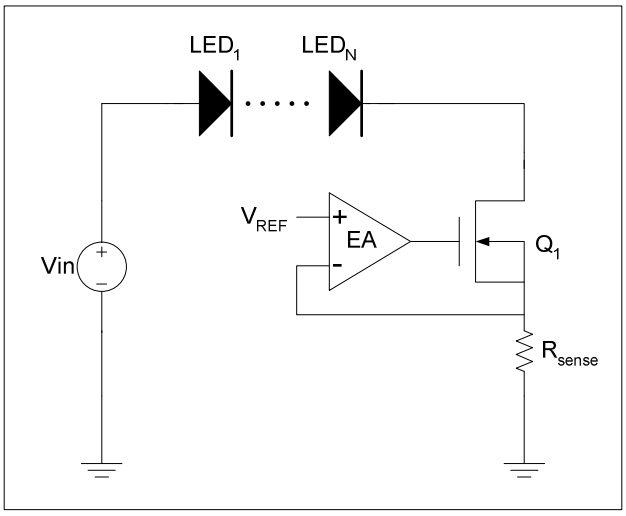


Figure 2.5 A linear regulator driving a LED string.

2.3.2 SMPC

A DC-DC SMPC is a power electronics system that converts an unregulated input into a regulated output [12]. This is done by controlling the duration of ON and OFF times of the switches of the SMPC to produce a desired conversion at the output. Additionally, storage elements like capacitors and inductors are connected in such a way for both energy transfer and as a low-pass filter in order to reduce switching ripples. The buck and boost converters are the two most common SMPC.

The configuration of a buck LED driver for driving a single LED string in series is shown in Fig. 2.6 [13]. A buck converter maintains constant current in the LED strings by regulating the voltage across the current sensing resistor [13]. The voltage across the sensing resistor is compared with a reference voltage in order to generate the error voltage which is then processed

either by a proportional-integral-derivative (PID) or hysteretic controller to produce the pulse-width-modulation (PWM) signal for the transistors of the SMPC. By using negative feedback, the duty cycle of the PWM signal is modulated in order to regulate the voltage across the sensing resistor. Additionally, since the transistor of the buck converter is operating in the linear region, power losses are reduced leading to higher efficiencies than linear regulators.

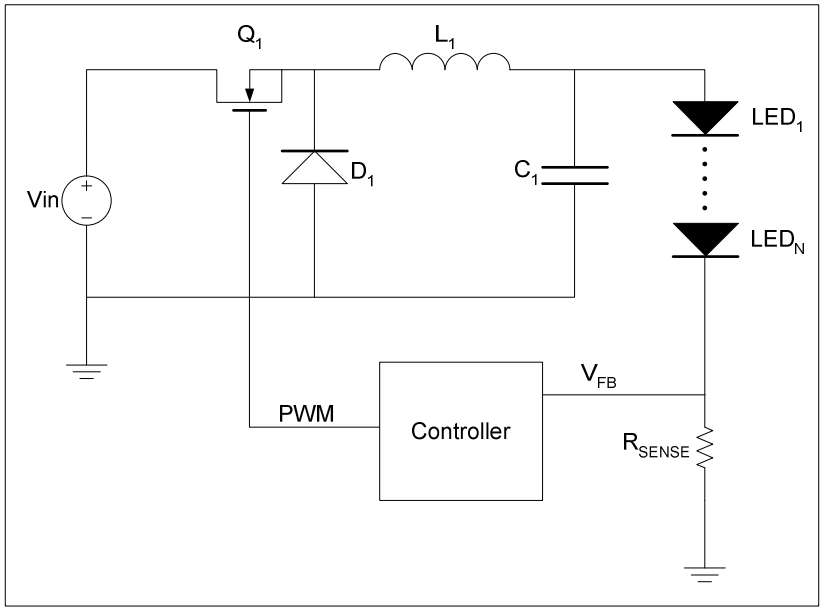


Figure 2.6 A buck SMPC driving a LED string.

2.3.2.1 OPERATION

The operation of a buck converter can be divided into two stages depending on the switching action of its transistors. Assuming that the LED load is a completely resistive load, i.e.,

$$R_{OUT} = \frac{V_{OUT}}{I_{LED}} \tag{2.3}$$

during stage 1 of this converter, the switching transistor turns ON, and the diode is switched OFF and its equivalent circuit is shown in Fig. 2.7. This stage is characterized by the charging of the

inductor as the input voltage is greater than the output voltage [14]. The voltage across the inductor is given by,

$$v_L = L\frac{di_L}{dt} \tag{2.4}$$

Therefore from (2.4) [9],

$$V_{IN} - V_O = L\frac{I_2 - I_1}{t_{ON}} = L\frac{\Delta I}{t_{ON}} \tag{2.5}$$

$$\Delta I = \frac{(V_{IN} - V_O)t_{ON}}{L} \tag{2.6}$$

where I_1 is the minimum inductor current, I_2 is the maximum inductor current, V_{IN} is the input voltage, V_O is the average output voltage, L is the inductance, and ΔI is the peak to peak inductor ripple current.

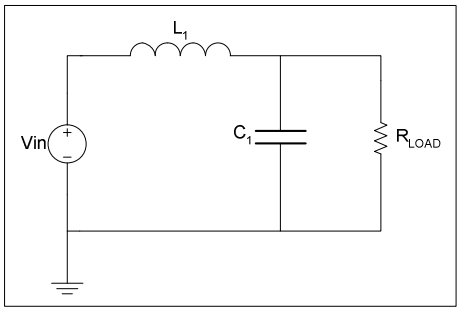


Figure 2.7 Stage 1 equivalent circuit configuration of a buck converter.

Stage 2 is characterized by the power switching transistor turning OFF and the diode being forward biased and its equivalent circuit is shown in Fig. 2.8. In this stage, the energy to the load is provided from the stored energy in the inductor during stage 1 leading to inductor current decreasing. Therefore [9],

$$-V_O = L\frac{I_1 - I_2}{t_{OFF}} = L\frac{\Delta I}{t_{OFF}} \tag{2.7}$$

$$\Delta I = \frac{V_O t_{OFF}}{L} \tag{2.8}$$

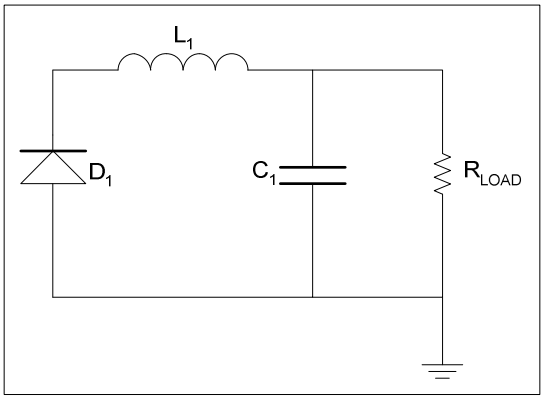


Figure 2.8 Stage 2 equivalent circuit configuration of a buck converter.

For steady state operation, the peak-to-peak inductor current has to be the same during both the ON time (t_{ON}) and OFF time (t_{OFF}) cycle. Therefore [9],

$$\frac{(V_{IN} - V_O)t_{ON}}{L} = \frac{V_O t_{OFF}}{L} \tag{2.9}$$

$$t_{ON} = DT_S \tag{2.10}$$

$$t_{OFF} = (1 - D)T_S \tag{2.11}$$

where D is the duty cycle and T_S is the switching period. Thus [9],

$$(V_{IN} - V_O)DT_S = V_O(1 - D)T_S \tag{2.12}$$

$$V_O = V_{IN}D \tag{2.13}$$

It can be seen that the average output voltage is equal to the product of the input voltage and the duty cycle [14]. Thus, the drive voltage provided by the buck converter to the LED load can only be lower than the input voltage. In order to keep the drive voltage in regulation, the duty cycle of the PWM signal changes when its input and load changes.

2.3.2.3 SYCHRONOUS RECTIFIER

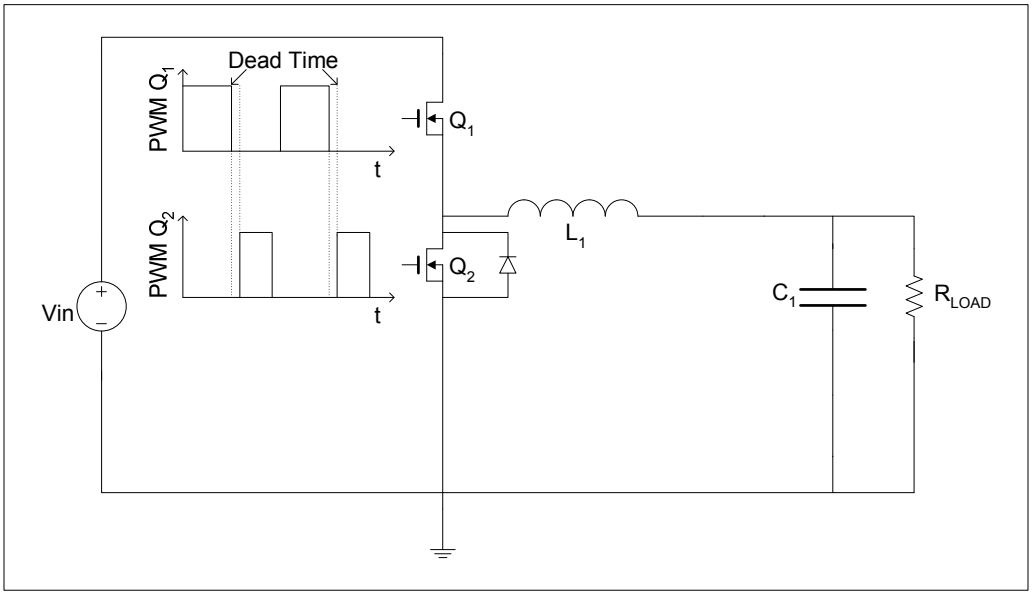


Figure 2.9 A synchronous buck converter.

In order to improve the efficiency of the buck converter based LED driver, a synchronous buck converter as shown in Fig. 2.9 is implemented [8]-[9]. A synchronous buck converter is basically a modified version of the buck converter, which leads to increased efficiency and cost, where the freewheeling diode, D_1, is replaced by a switch, Q_2 [14]. The voltage drop across the free-wheeling diode in a conventional buck converter presents an important limitation in converter efficiency [9]. The output voltage for the buck converter is given as

$$V_{OUT} = DV_{IN} - (1-D)V_{DIODE} \tag{2.14}$$

where D refers to the duty cycle of the SMPC, and V_{DIODE} is the voltage drop across the free-wheeling diode [9].

Thus, the higher the voltage drops across the diode, the less efficient the buck converter [8]-[9]. A MOSFET can be used in place of the free-wheeling diode since the voltage drop across its drain and source is smaller than that of a diode. The gate signal for the two MOSFETs in a synchronous configuration has to be complimentary so that the synchronous buck converter can have its two modes of operation similar to a conventional buck converter. In addition, a dead time is required to ensure that neither of the MOSFETs conducts at the same time or it would lead to a large shoot-through current as evident in Fig. 2.9.

2.3.3 CONTROL ARCHITECTURE

Several control schemes can be used to control the DC-DC SMPC but the most popular control scheme is the PWM control [14]. The duration of ON and OFF pulses of the transistor of the converter are being modulated in a PWM controller in order to maintain regulation at the output. There are two types of PWM control, fixed and variable frequency control [9],[14]. The most common PWM control technique is the fixed frequency PWM method because the variable frequency PWM leads to unwanted EMI due to changing of switching frequency. Fixed frequency PWM technique is achieved by maintaining a constant frequency while modulating the ON and OFF times.

There are two different modes of operation for PWM control depending on the control signals required to maintain regulation: voltage-mode control and current-mode control. Voltage-mode control is implemented in LED drivers for a single LED string by sensing the voltage across a current sensing resistor (V_{FB}) and comparing the voltage against a reference (V_{REF}) as shown in

Fig. 2.6. This error voltage is then processed by a controller to generate the PWM signal for the switches of the SMPC. Typical controllers used in voltage-mode controllers are PID or hysteresis based and is implemented using operational-amplifiers (opamps) in analog controllers or as control algorithms in digital controllers. The output voltage of the controller is then compared to a sawtooth signal by a comparator to generate the necessary PWM signal for the transistors of the SMPC as shown in Fig. 2.10. In this dissertation, voltage-mode control is used for driving LED strings for ease of implementation.

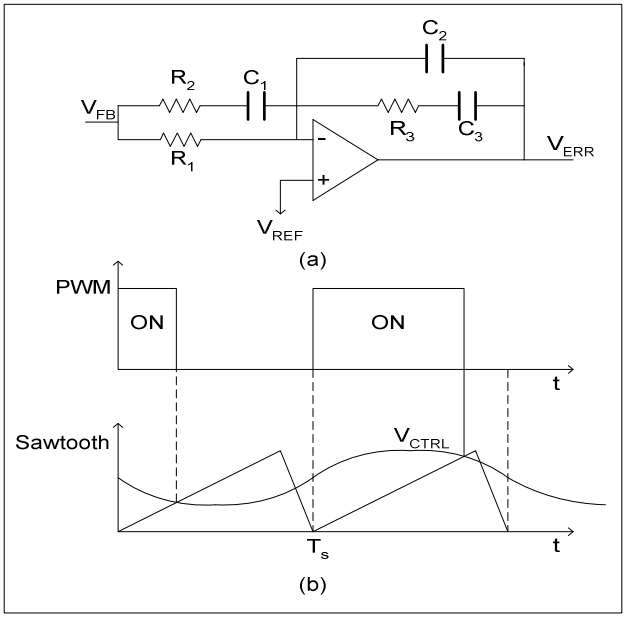


Figure 2.10 Voltage-mode PWM control. (a) A two zero and three pole PID controller implemented using opamps in analog controller. (b) Waveforms for a fixed frequency PWM controller.

Current-mode controllers have an extra loop in addition to the voltage feedback loop as in Fig. 2.11 for driving a single LED string. The ON and OFF times of the PWM signal are determined by the time at which the sensed inductor current (V_{CS}) reaches the reference value (V_{REF}) for the inner loop. This reference value is obtained from the processing of the voltage difference between the voltage feedback and the reference of the controller. Current-mode control has

several advantages over the traditional voltage-mode control, over current protection due to switching OFF of transistors of the SMPC when the inner current feedback loop reaches its threshold value, simpler controller design due to reduction of the SMPC system order, and no load sharing problem for SMPC operating in parallel [9]. However, the major drawback of this mode of control is its instability whenever the duty cycle is greater than 50%. By introducing slope compensation technique, this instability can be compensated.

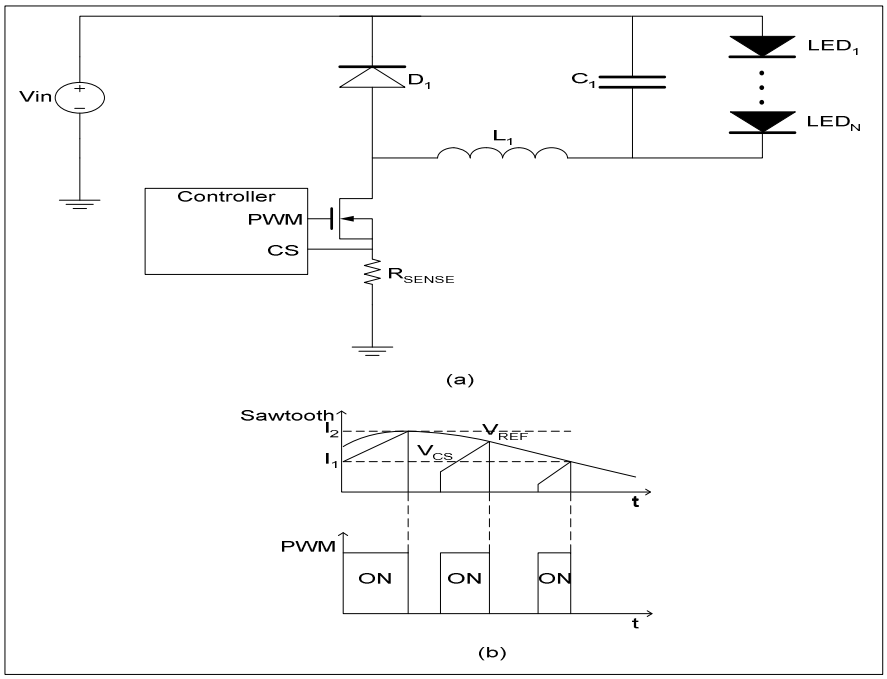


Figure 2.11 Current-mode control: (a) Buck converter driving a single LED string using current-mode controller. (b) Waveforms for a fixed frequency PWM controller.

2.4 LITERATURE REVIEW

The research problem investigated in this dissertation is the design of a high-efficiency power electronics driver circuit for driving multiple strings of LEDs in parallel configuration. The main focus of this research is to develop a driver circuit which drives the LED strings with a minimum drive voltage leading to high efficiency across the strings and reduction of unwanted power

losses in the current controller circuits while maintaining good current regulation in the strings. Prototypes of three different LED drivers were designed and simulated using MATLAB SIMULINK and results are included.

Recently, there has been a push towards introducing LEDs in new applications due to its various advantages over other traditional lighting sources. Efforts are being made to improve the efficiency, reliability, and cost of the LED driver circuits. Ways of driving parallel connected LED strings are already introduced and discussed as in Fig. 1.1 and 1.2. But these configurations do not lead to high efficiency or cost-effective approaches. Thus, new techniques are currently being introduced that will adjust the drive voltage of the LED strings to its minimum leading to high efficiency. In addition, the use of a single SMPC to drive multiple LEDs in parallel configuration leads to reduction of cost of the overall system.

An approach of driving multiple LED strings for parallel configuration is shown in Fig 2.12 and is being implemented in [15]-[16] using a digital controller for driving LED backlights for LCD panels [4]-[7]. In this implementation, current controllers are used in series with the individual LED strings to maintain the desired current across the strings [4]. In order to increase the efficiency, the minimum drop across the current-controllers (V_{MIN}) is sensed using diodes and compared against the reference (V_{REF1}), thus, the drive voltage provided by the SMPC is adjusted to its minimum value leading to voltage drop across the current controllers being set to:

$$V_{MIN} = V_{REF1} - V_F \tag{2.15}$$

where V_F is the drop across the sensing diodes [4]-[7].

In this approach, V_{REF1} has to be selected higher than the worst-case condition, leading to the LED strings being driven by higher voltage than is necessary in order to maintain the desired

current, and thus, this results in a higher power dissipation in the current controllers in the LED strings [4]-[7]. In addition, the fluctuations of voltage drop across the sensing diodes due to temperature leads to variations in power losses across the current controllers [4]-[7].

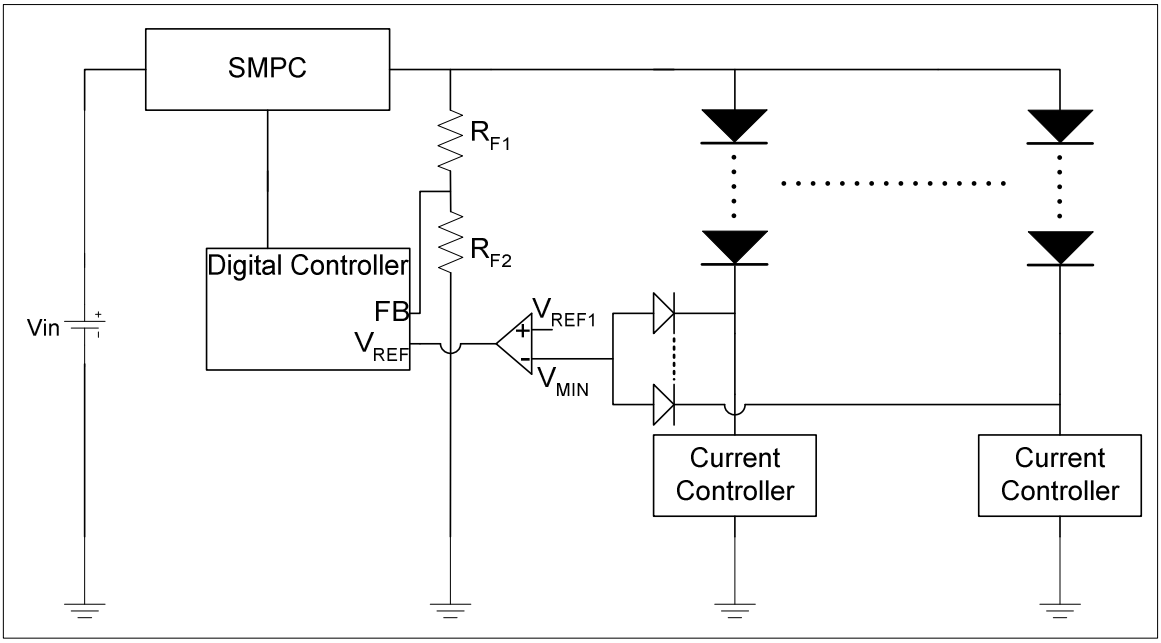


Figure 2.12 A LED driver as implemented in [15]-[16].

Another approach for increasing the efficiency of the driver for multi string LEDs is implemented in [17] as shown in Fig. 2.13 for low power lighting applications [3]. In this implementation, a STMicroelectronics L6561 controller was used, the lowest voltage drop across the current controllers are sensed using sensing diodes (V_{MIN}) and compared against a reference (V_{REF1}) to generate the proper feedback voltage for the SMPC. The lowest voltage drop across the current controllers are sensed so that the driver provides the minimum drive voltage necessary for the LED strings to maintain regulation leading to higher efficiency at the LED load. Similar to [15]-[16], the use of sensing diodes leads to variations in sensing voltage due to temperature dependence of diodes and also since V_{REF1} has to be selected higher than the worst-

case condition in order to maintain the LED strings in regulation, leading to lower efficiency at the LED load.

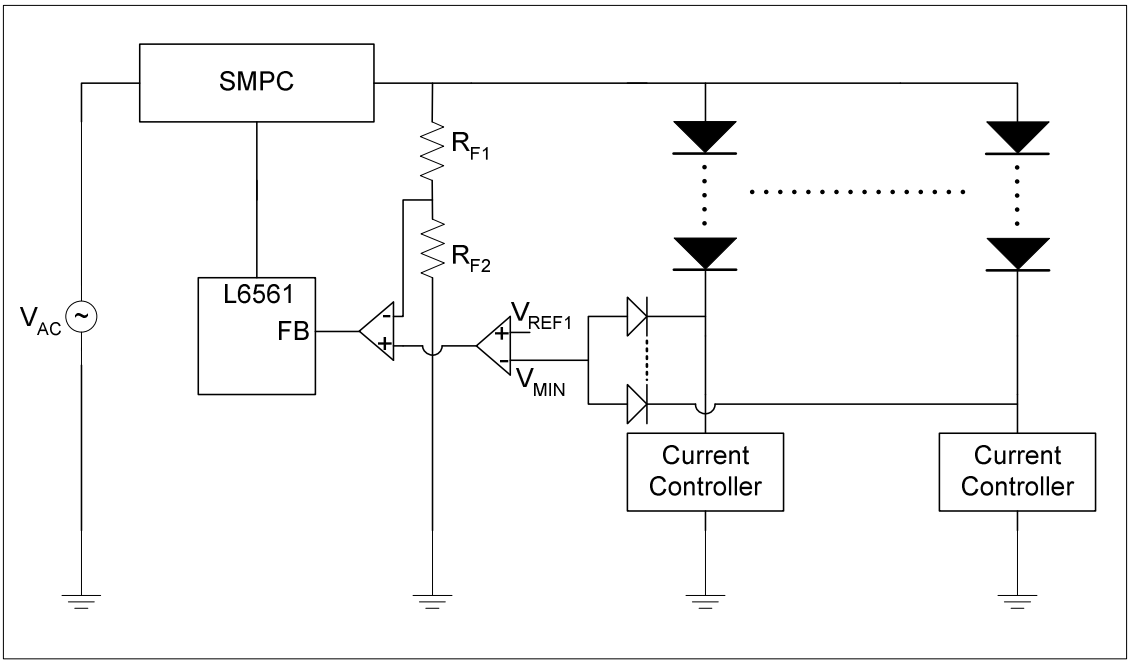


Figure 2.13 A LED driver as implemented in [17].

Similarly, in [6]-[7], the sensing diodes are used to sense the maximum gate-source voltage (V_{MAX}) of the MOSFETs of the current controllers circuit as shown in Fig. 2.15 in order to adjust the drive voltage of the driver to the optimal level [3],[4],[6],[7]. In this implementation, a Texas Instruments' UC3843 PWM controller was used, by exploiting the inverse relationship between the gate-source and drain-source voltage of a MOSFET in the linear region for proper adjustment of the drive voltage, i.e., the current controllers with the minimum drain-source voltage will have the maximum gate-source voltage for the desired current. In this case, V_{MAX} was the feedback voltage for the controller which would be used by the controller to generate the proper PWM signal for the switching transistor in the SMPC. Similar to other approaches explained above, the sensing diodes causes variations in sensed voltages due to temperature dependence of diodes,

and, in addition, the modulator gain of the controller has to be bounded between an upper and lower limit in order for proper operation of the driver [3]-[4],[6]-[7].

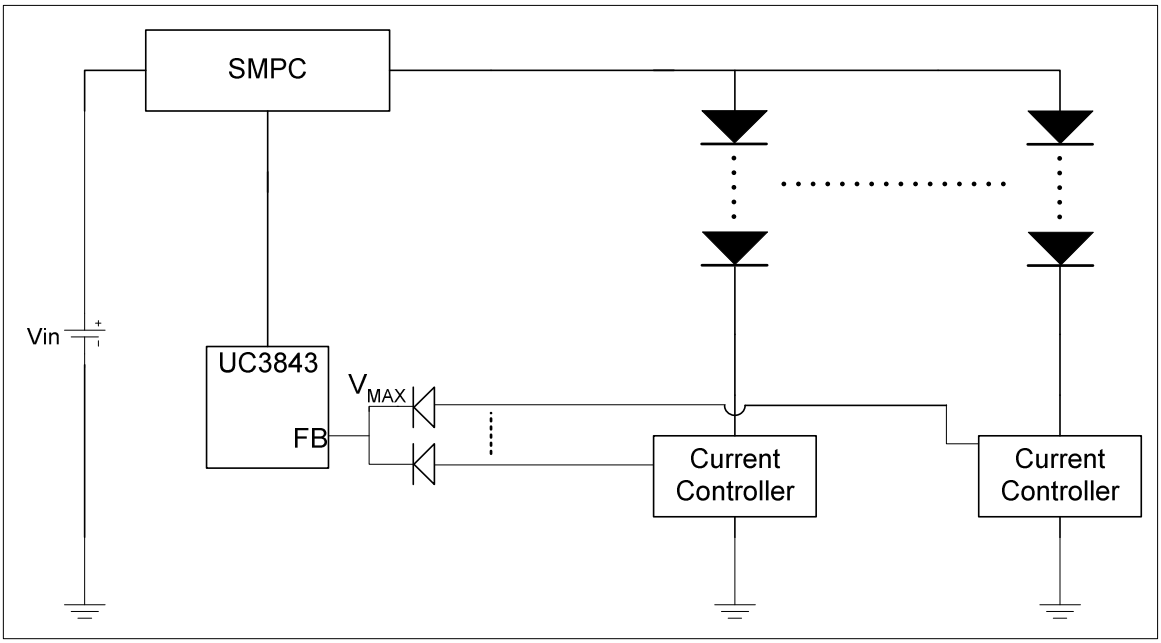


Figure 2.14 A LED driver as implemented in [6]-[7].

Therefore, the focus of this dissertation is to design new drivers for multi string LEDs using current controller circuits in order to reduce the power dissipation in the current controllers circuit by driving the LED load using a minimum drive voltage. Chapters 3, 4 and 5 show the design and description of proposed LED drivers to meet the above requirement.

CHAPTER 3

A HIGH-EFFICIENCY DIGITALLY CONTROLLED RGB DRIVER FOR LED PIXELS

3.1 INTRODUCTION

Introduced in this chapter is a high-efficiency digitally controlled RGB LED driver for LED display panels. Recent advances in LED technology, has led to RGB LEDs finding new applications for its use. A digitally controlled RGB driver was designed to drive a 3x3 RGB display panel. The proposed driver maintains high efficiency at the LED load by maintaining minimum voltage across the LEDs and the current controllers necessary to keep it in regulation by selecting the minimum drain voltage of the MOSFETs of the current controllers as the feedback voltage for the SMPC driver. In the proposed driver, a dimming controller circuit is also able to dim each individual LED in the pixel. A MATLAB SIMULINK model of the proposed driver is also shown. At the maximum rated current of 180 mA for the RGB-driver efficiencies of 85.6 %, 93.3 %, and 91.1 % for the red, green, and blue LEDs, respectively, were experimentally verified [4].

Section 3.2 provides a detailed description of the RGB-driver for driving 3x3 display panels. It also explains the two different modes of operation of the digital controller in order to maintain a minimum drive voltage across the LED strings. Section 3.3 of this chapter introduces different types of dimming control implemented in LEDs. An analog dimming was implemented for this design using a daisy chaining multiple digital potentiometers to dim each individual LED in a 3x3 RGB display panel. The design procedure of a synchronous buck converter is introduced in section 3.4. The closed loop modeling of a synchronous buck converter using a PID voltage-mode feedback controller in MATLAB is also shown. Section 3.5 shows the simulation of this

A journal paper based on this chapter is published in IEEE Transactions on Industrial Applications.

driver in MATLAB SIMULINK. Finally, section 3.6 shows the experiment results of the RGB LED driver for driving a 3x3 display panel.

3.2 CIRCUIT DESCRIPTION

The proposed LED driver for the red color is shown in Fig. 3.1. It consists of multiplexers, current-controllers, a dc-dc SMPC, and a microcontroller for a 3x3 RGB display panel [4]. Similar arrangements are used for the green and blue color LED drivers. Due to large number of control nodes and RGB LEDs required – the driver driving RGB-based LEDs for display panel applications is quiet complex. The current controller is used to maintain the current across the individual LED string by sensing the voltage drop across the sensing resistor and comparing it to the dimming voltage in the EA. The EA processes this error voltage to regulate the corresponding MOSFET in the current controller to maintain the desired LED string current [6]-[7].

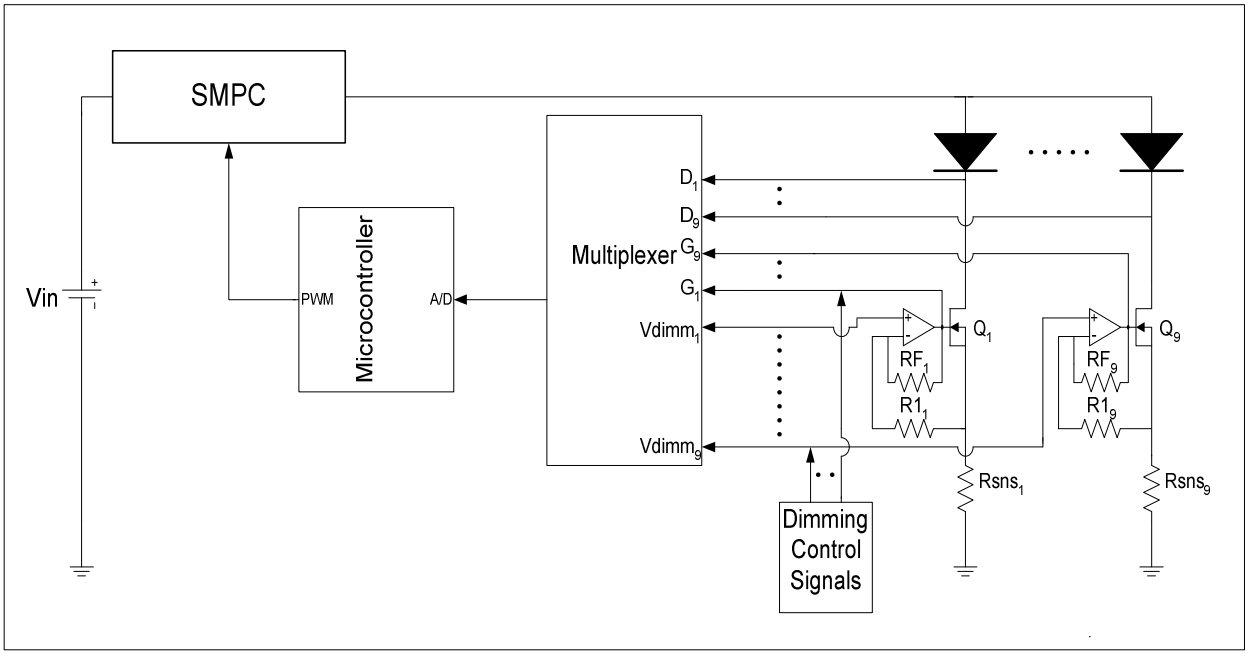


Figure 3.1 Proposed red color LED driver.

As mentioned in Section 2.4, the use of a sensing diodes leads to fluctuation of sensing voltages due to temperature dependence of its voltage, thus a multiplexer is used to sense the gate (V_G), drain (V_D), and dimming control voltages (V_{DIMM}). The output of the multiplexer is connected to the analog-to-digital (A/D) converter inputs of the microcontroller, which compares the reference voltage with the sensed minimum V_D of the MOSFETs of the current controllers in order to generate the error voltage during the operation mode of the driver. This error voltage is processed by a PID controller inside the microcontroller to generate the duty cycles for the PWM signal for the switches of the SMPC.

The drive voltage for the LED strings is provided by the SMPC. The SMPC can be either an isolated or non-isolated topology [4]-[7]. In this implementation, the SMPC is a non-isolated synchronous buck converter since the drive voltages for the red, green, and blue LEDs are 2.3 V, 3.66 V, and 3.72 V, respectively.

One of the significant disadvantages of using current controller is its power losses in the MOSFETs of the current controllers, since the MOSFETs operate in the saturation region of operation. Therefore, in order to reduce losses in the MOSFETs and increase efficiency at the LED load, the MOSFETs are maintained with minimum V_{DS} required for the desired current setting [4]. For example, if a single 5 V power supply was used to drive a RGB LED pixel, it would yield efficiencies of 38.3% for the red LED, 66.2 % for the green LED, and 64.5 % for the blue LED at the nominal current of 20mA. Thus, in order to maintain a minimum voltage drop across the MOSFETs of the current controllers, the proposed LED driver operates in two modes of operation: efficiency optimization and operation mode [4]. In the efficiency optimization mode, it involves finding the minimum V_{DS} indirectly in order to maintain the desired LED string current by reducing the duty cycle of the PWM signal until the sensed maximum V_G is

greater than the threshold value [3]. This is done by exploiting the inverse relationship between drain-source (V_{DS}) and gate-source (V_{GS}) voltage of a MOSFET for a set drain current in the linear region, i.e., when V_{DS} decreases V_{GS} increases and vice versa [6]-[7]. Fig. 3.2 shows the flowchart of this mode. In the operation mode, the control algorithms are processed to generate the corresponding duty cycles for the PWM signal for the desired LED string current and input voltage settings. Fig. 3.3 shows the flowchart of this mode.

EFFICIENCY OPTIMIZATION MODE:

1) Initially at startup, the microcontroller would output a PWM signal so the SMPC output voltage would be 5V, which is high enough to maintain the MOSFETs of the current controllers in the saturation region and the LED strings are at the desired current setting [4],[8].
2) For each color, the gate voltages, V_G, of the MOSFETs are sensed by the multiplexer and are sent to the microcontroller for processing through the A/D input pin while the current controllers maintain the desired current settings in the LED strings. The maximum V_G is found from these sensed voltages for each color.
3) The sensed maximum V_G is compared against a threshold value for each color of LEDs in order to determine the proper operation of LED strings. This threshold value is the maximum V_G in order to operate the MOSFETs of the current controllers with minimum V_{DS} required to maintain the desired current in the LED strings.
4) If the sensed maximum V_G of the MOSFETs for each color of LEDs is below this threshold value, the duty cycle of the PWM signal for the switches of the SMPC is decreased leading to reduction of drive voltage provided by the SMPC to the LED strings. After the reduction of duty cycle of the PWM signal, the microcontroller senses

the V_G of the MOSFETs again. The above process continues until the V_G of the MOSFETs are above the threshold, indicating the LED strings are no longer able to maintain the desired current setting.

5) Therefore, the duty cycle of the PWM signal is increased by the value it was reduced by in the previous step in order to maintain the desired current in the LED strings at this time and then enters the operation mode.

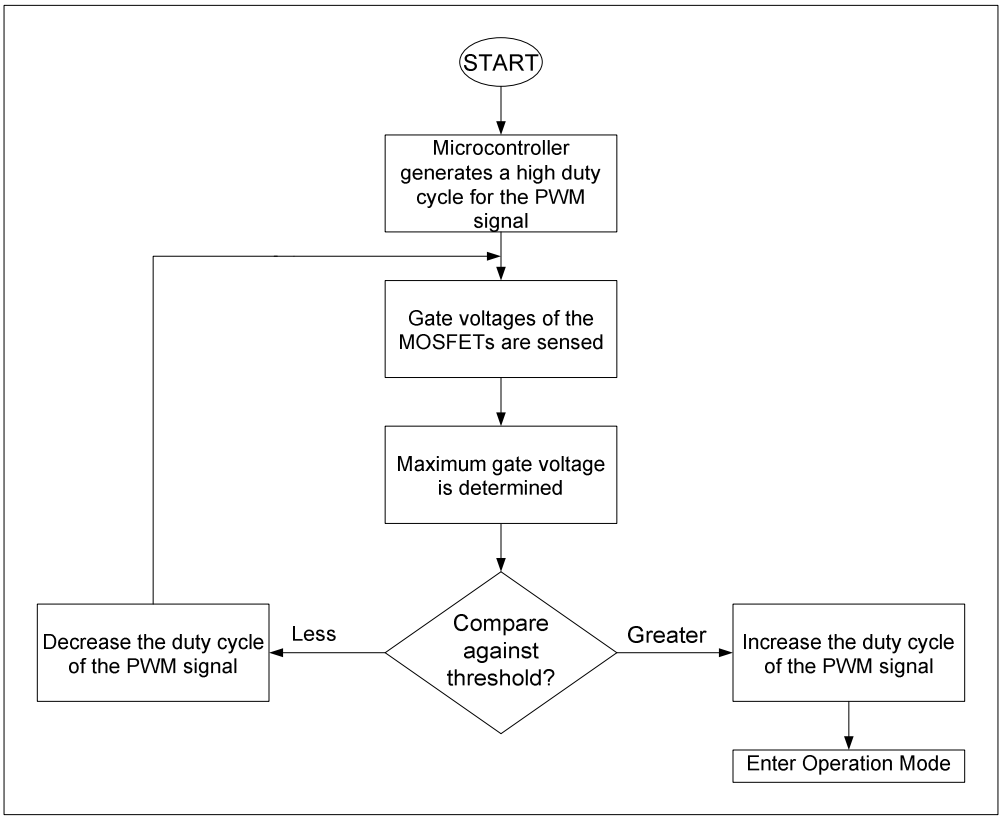


Figure 3.2 Flowchart of efficiency optimization mode.

OPERATION MODE:

1) The multiplexer senses the V_D and V_{DIMM} of each color and send it to the microcontroller through the A/D input pin.

2) At the beginning of this mode, the minimum V_D and maximum V_{DIMM} are found from the sensed V_D and V_{DIMM}, respectively, and stored in memory. The minimum V_D is set as the reference voltage at this time for the set current level. Thus, the SMPC is adjusted according to this reference voltage leading to the LED strings being driven by the minimum drive voltage, and hence, reducing unwanted power losses in the MOSFETs of the current controllers.
3) Then, the error voltage is found by comparing this reference voltage with the sensed minimum V_D of the MOSFETs, i.e., $V_{ERROR} = V_{REF} - V_{MIN(D_1,.....,D_9)}$. This error voltage is processed by the PID controller in order to generate the corresponding duty cycle of the PWM signal for the switches of the SMPC.
4) If the maximum V_{DIMM} is changed, the microcontroller will enter the efficiency optimization mode in order to find the minimum V_{DS} for the new desired current setting. Thus, the SMPC drives the LED strings with a minimum drive voltage at all current setting leading to increased efficiency in the LED load.

3.3 DIMMING CONTROL

Dimming is an essential feature in LED display panels due to the requirement of displaying various color contrasts. Additionally, dimming capabilities are implemented for energy conservation purposes in general-purpose lighting [18]. For example, a person for reading and writing requires 300 1x (30 fc) whereas for computer uses requires only 30 1x (3 fc) [19]. Dimming is implemented in LEDs by changing its forward current, as LEDs light output is proportional to the current flowing through them. There are two primary types of dimming schemes in the LED industry, Analog or PWM dimming [18],[20]. The choice of dimming depends on the types of LEDs, dimming range, and the application it is being implemented [1].

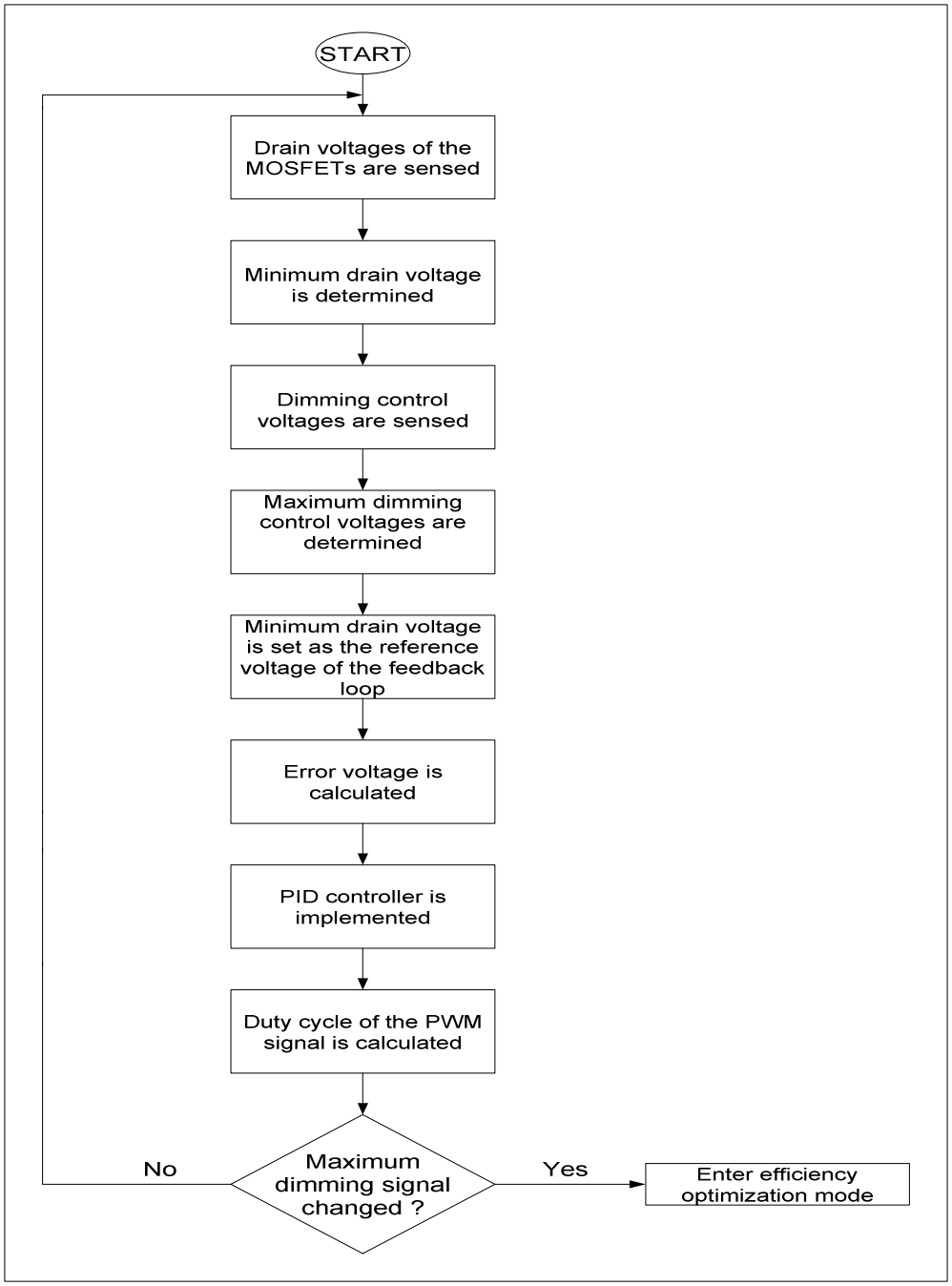


Figure 3.3 Flowchart of operation mode.

Analog dimming is implemented by adjusting the current flowing through the linear region of LED V-I curve leading to generation of proportional light output, i.e., 50% of the maximum LED string current generates 50% brightness in the LED string as shown in Fig. 3.4 [4],[8],[18],[21]. For a LED string with individual current controller, analog dimming is

implemented by changing the dc voltage of the non-inverting input of the EA of the current controller. Dimming range of 10 % to 100 % is achievable using this dimming method [18]. Due to non-linear I-V curve of LEDs at current levels below 10%, this method suffers from linear change of brightness at these lower currents. Therefore, analog dimming is implemented for up to 10% of the nominal LED current, and then, changes to PWM dimming below 10% for improved linear brightness [18],[22].

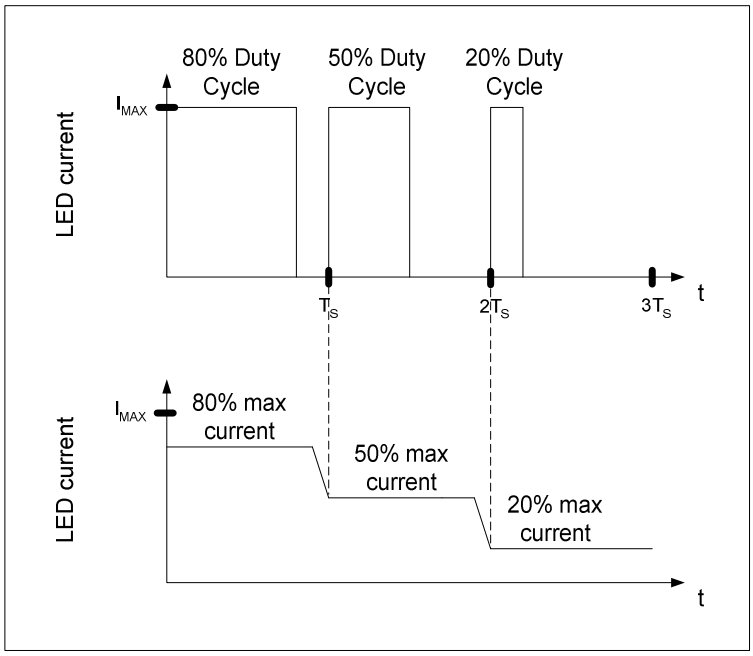


Figure 3.4 LED current with analog and PWM dimming [22].

PWM dimming is achieved by changing the duty cycle of the PWM signal. This is usually performed using a microcontroller or a digital signal processor (DSP). For example, LED string current at 50% duty cycle would result in 50% brightness in the LED string as shown in Fig. 3.4 [4],[8],[21]-[22]. Similar to analog dimming, in LED string with individual current controllers, this type of dimming method is implemented by changing the PWM signal of the non-inverting input of the EA of the current controller. PWM dimming method can achieve linear dimming

ranges between 0 % to 100 %. Table 3.1 summarizes the advantages and disadvantages of these two types of dimming methods [23].

Table 3.1 – Analog versus PWM dimming [18]-[23].

PWM Dimming	Analog Dimming
LED current is adjusted by modulating the LED peak current.	LED current is adjusted by changing the LED current.
Wider dimming range.	Dimming range – 10 % to 100 %.
Low chromaticity shift.	Chromaticity shift due to changing of LED current.
Lower optical-to-electrical efficiency.	Higher optical-to-electrical efficiency, i.e., more lumens for the same power.
Audible ringing due to output capacitor of SMPC and operating range above 100 Hz in order to reduce flickering.	No audible noise or flickering.

In this proposed driver, individual dimming of each color of each pixel is required [4],[8]. In this implementation, 27 different dimming control signals are required to be generated, and thus, for ease of implementation – analog dimming is implemented. The dimming signals are generated externally and are fed to the non-inverting input of the EAs of the current controllers.

A 3-wire serial-peripheral-interface (SPI) (active-low Chip Select (CS), Clock, and Data) is used by the Microchip's MCP42100 digital potentiometers (DPs) to communicate with Microchip's PIC18F4331 microcontroller to generate the analog dimming signals for the LED strings [4],[8],[24]-[26]. In this implementation of driving a 3x3 RGB display panel, 27 different dimming signals would be needed to be generated in order for each string to have its individual dimming signals [4]. Thus, using a 3-wire SPI would require the PIC18F4331 microcontroller three individual input-output (I/O) pins to control each DP. Therefore, the DPs needs to be daisy chained to provide the dimming signals as shown in Fig. 3.5, by connecting the serial-data-in

(SI) of one DP to the serial-data-out (SO) of the other DP on a single 3-wire SPI bus [4],[8],[24]-[26]. By using a daisy chain, multiple slave devices can be connected in a serial manner using only three I/O pins from the master for communication [4],[8],[24]-[27].

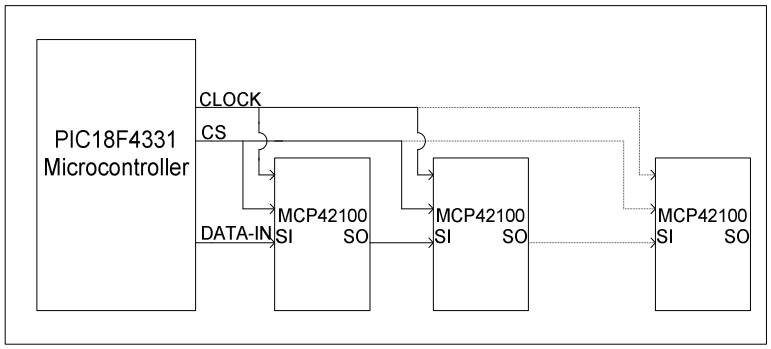


Figure 3.5 Daisy chained MCP42100 digital potentiometers [4],[8],[23]-[26].

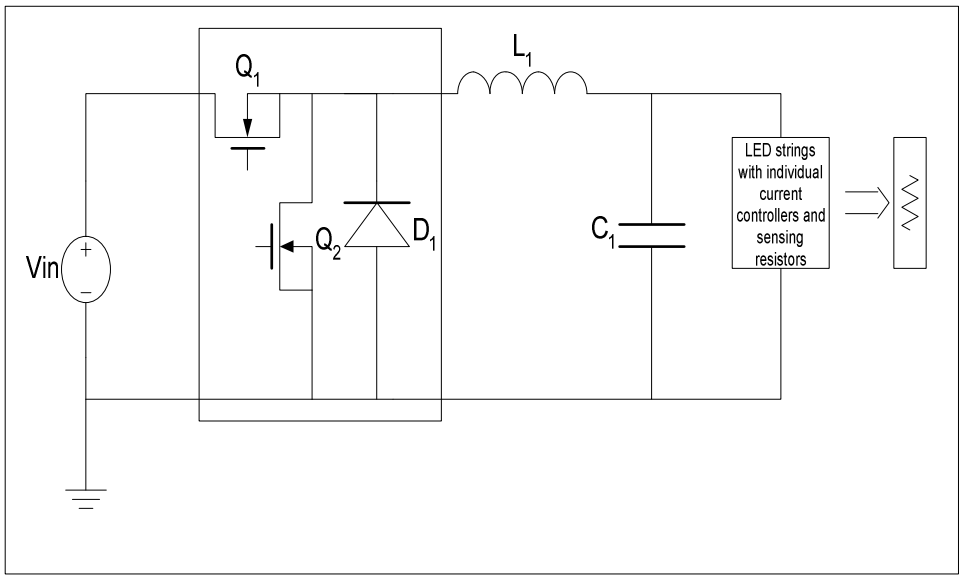
In this implementation, serial communication is established using Maxim IC's MAX232 rs-232 drivers/ receivers between a personal computer (PC) and the PIC18F4331 dimming controller in order to be able to individually dim each LEDs of RGB display panel [4],[8],[28]. Communication between daisy chained DPs and PIC18F4331 starts as soon as the active-low CS pin goes low [4],[8],[24]. The command and data bits are clocked into DP1's shift register and propagate to SO pin of DP1 [4],[8],[24]-[29]. Since the slave DP's are connected serially, the command and data bit are clocked into DP2's shift register and appears at SO pin of DP2. This process continues as long as active-low CS pin remains low until all the DPs receive its appropriate dimming signals. The command and data bits are executed only by the DPs on the rising edge of the active-low CS pin [29].

3.4 DESIGN OF DC-DC SYNCHRONOUS BUCK CONVERTER

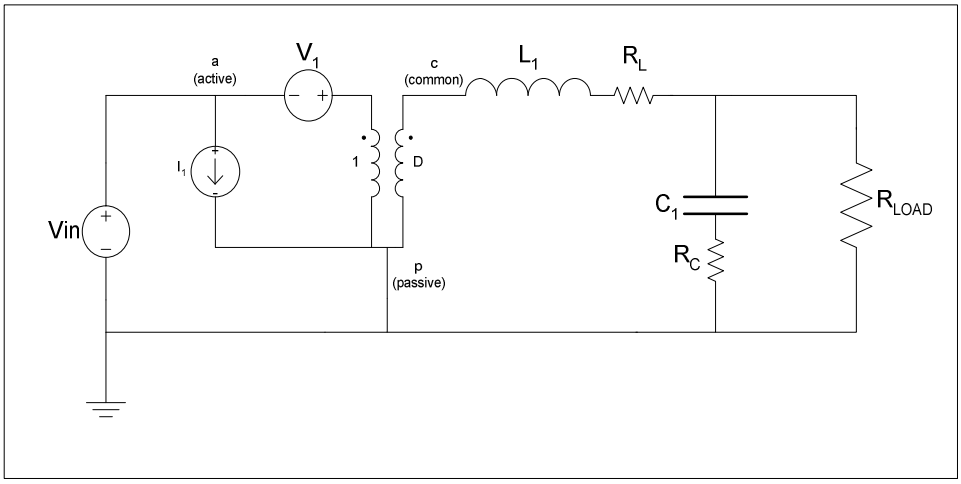
A DC-DC synchronous buck converter discussed in section 2.2.2.2 is used to provide the drive voltages for each red, green, and blue LEDs in the 3x3 RGB pixels. The synchronous buck

converter was chosen for this application due to the requirement of drive voltage of the red, green, and blue LEDs being lower than the input voltage. Additionally, synchronous buck converter provides better efficiencies than conventional buck at these lower drive voltages [8]-[9].

The multi-string LED loads with its individual current controllers, and sensing resistors are modeled as resistive loads as shown in Fig. 3.6 (a). This approximation does not affect the dynamic response of the converter. Thus, the dynamic behavior of the synchronous buck converter can be developed using the Vorpèrian averaged-switch model, which replaces the non-linear components of the converter with its simple equivalent circuit model as shown in Fig. 3.6 (b) [9],[30].



(a)



(b)

Figure 3.6 (a) DC-DC buck converter with LEDs modeled as resistive load. (b) Small-signal model of DC-DC buck converter based on Vorpèrian averaged-switch model [9],[30].

Using the small-signal model shown in Fig. 3.6 (b) and state space averaging method, the control-to-output transfer function can be found to be [30]-[34],

$$\frac{\widehat{v_O}}{\hat{d}} = \left(\frac{V_O}{D}\right)\left[\frac{1+sR_CC}{1+s\left(R_CC+(R_{LOAD}//R_L)C+\frac{L}{R_{LOAD}+R_L}\right)+s^2LC\left(\frac{R_{LOAD}+R_C}{R_{LOAD}+R_L}\right)}\right] \tag{3.1}$$

In the transfer function above, R_C and R_L are the parasitic resistances associated with capacitors and inductors. It has two poles, and one left half plane zero due to the ESR of the output capacitor [30]-[32]. This transfer function is used in MATLAB to design the appropriate controller for the DC-DC synchronous buck converter.

3.4.1 PARAMETER CALCULATIONS

The buck converter output filter storage elements are selected in order to satisfy the allowed ripple magnitudes of the inductor current and capacitor voltage [1],[3],[16]. The inductor current

ripple is found using equations (2.6) and (2.8) [9],

$$t_{ON} = \frac{L\Delta I}{V_{IN}-V_O}, t_{OFF} = \frac{L\Delta I}{V_O} \tag{3.2}$$

Therefore, switching frequency is $f_S = \frac{1}{T_S} = \frac{1}{t_{ON}+t_{OFF}}$ (3.3). The nominal input voltage of the buck converter was chosen to be 12 V to yield drive voltages of 2.3 V, 3.66 V, and 3.72 V for red, green, and blue LEDs, respectively. From [9],

$$L = \frac{DV_{IN}(1-D)}{f_S\Delta I} = 75\mu H \tag{3.4}$$

The value of the capacitor is depended on the allowed output ripple voltage since the capacitor of the output filter in the buck converter is connected directly across the load. Therefore, the output capacitor is found by taking into account that the charging and discharging capacitor current are the same during the ON mode and OFF mode of a synchronous buck converter [9]. Hence [9],

$$C = \frac{V_{IN}D(1-D)}{8f_S^2 L\Delta v_C} = 100\mu F \tag{3.5}$$

The performance of the synchronous buck drivers for a 3x3 RGB LED pixel was experimentally verified using a PIC18F4431 controller with a switching frequency of 100 kHz [4],[7]-[8]. Each string of LEDs are rated at 20 mA and , thus, the synchronous buck converter needs to deliver a total output current of 180 mA since there are nine strings of red, green and blue LEDs. The key components and parameters for the design of synchronous buck driver are shown in Table 3.2 – (a) and (b).

3.4.2 DESIGN OF CLOSED LOOP DIGITAL VOLTAGE-MODE FEEDBACK

Recently, there has been a trend towards introducing digital controllers in SMPC due to

Table 3.2 – (a) Key parameter list, (b) Key component list.

(a)

Parameter	Value
Input Voltage (V_{IN})	12 V
Output Voltage (V_O)	RED=2.3 V, GREEN=3.66 V, BLUE=3.72 V
Output Current (I_O)	180 mA
Frequency	100 kHz
Parasitic	R_C=150 m [39], R_L=70m [40]
Ripple Inductor Current (ΔI)	RED=0.25 A, GREEN=0.34 A, BLUE=0.34 A
Ripple Output Voltage (ΔV)	RED=3 mV, GREEN=4 mV, BLUE=4 mV

(b)

Components	Value
Input Capacitor	10 µF, 25 V
MOSFETs of the Synchronous Buck Converter	IRF9630, IRFD9120
Output Inductor	75 µH, 2.5 A
Output Capacitor	100 µF, 35 V
MOSFETs of the Current Controllers	Q2N7000
Error Amplifiers of the Current Controllers	TLC272
Sensing Multiplexers	ADG706B
PWM Controller	PIC18F4431

advancement in very-large-scale-integration (VLSI), availability of low cost, high performance DSPs and microcontrollers with enhanced features like A/D and PWM [4],[8][12],[35]-[38]. Digital controllers have certain advantages over analog controllers such as less susceptible to aging and temperature, flexibility as control algorithm can be changed in software, and ability to

add more features without the need for designing complex circuitry [12],[35]-[38]. Thus, in this RGB LED driver a PIC18F4431 is used to control the synchronous buck converter for each color. In this implementation, the microcontroller senses the minimum drain voltage of the MOSFETs of the current controllers and compares with the reference voltage as discussed in section 3.2 in order to regulate the drive voltage of the LED strings of each color. In this approach, a digital redesign approach using frequency response technique is implemented. In digital redesign approach, the controller is initially designed in the s-domain, and then, digitized into its equivalent z-domain.

The design of a digital voltage mode feedback controller is based on equation (3.1) and on the parameters described in Table (3.2). The load of the synchronous buck converter is assumed to be resistive (R_{LOAD}), i.e.,

$$V_{OUT} = I_{OUT} * R_{OUT} \tag{3.6}$$

In this implementation, R_{LOAD} is equivalent to parallel combination of nine LEDs. Therefore, the effective resistance of each string, R_{SINGLE_STRING}, is series combination of equivalent DC resistance of the LED, i.e., $R_{EQU} = V_{FWD}/I_{LED\_STRING}$(3.7), the on-resistance of the MOSFETs, and the sensing resistor [4],[41]. For the case of red LEDs in a RGB LED, the equivalent DC resistance is obtained at their typical forward voltage, i.e., 1.9 V [42]. Therefore [4],

$$R_{EQU} = \frac{1.9}{0.020} = 95\ \Omega \tag{3.8}$$

$$R_{SINGLE_{STRING}} = R_{EQU} + R_{MOSFET} + R_{SENSE} = 95 + 1.8 + 10 = 106.8\ \Omega \tag{3.9}$$

$$R_{LOAD} = R_{SINGLE\_STRING\_1} // \ldots\ldots // R_{SINGLE\_STRING\_9} = 11.9\ \Omega \tag{3.10}$$

Substituting the values of R_{LOAD} from equation (3.10) and the parameter values from Table (3.2), the transfer function for the red synchronous buck driver is,

$$G_P(s) = \frac{0.00018s+12}{7.55*10^{-9}s^2+2.822*10^{-5}s+1} \tag{3.11}$$

Using MATLAB, the open loop Bode plot of the above transfer function is shown in Fig. 3.7. The transfer function has two conjugate poles at – (1.87±j11.36)*10^3 radian/sec , and left half plane zero at -6.7*10^4 radian/sec. Based on these results, a suitable PID controller is designed in MATLAB to meet the requirement for gain, crossover frequency and phase margins. The PID controller is defined using the following difference equation [43],

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau)d\tau + K_D \frac{de(t)}{dt} \tag{3.12}$$

where e(t) is the input to the controller and is found by comparing the reference voltage with the feedback voltage, K_P is the proportional gain, K_I is the integral gain, and K_D is the derivative gain. By implementing Laplace transform on equation (3.12), the transfer function of a PID is [12],[44],

$$G_C(s) = K_P + \frac{K_I}{s} + K_D s \tag{3.13}$$

Using MATLAB, the PID controller was designed with a pole in the origin, and two zeros at -3.42*10^3 radian/sec and -2.91*10^5 radian/sec, respectively. The closed Bode plot of the red color synchronous buck driver is shown in Fig. 3.8 and the model used for Bode plot is shown in Appendix A. From Fig. 3.8, it can be seen that the system bandwidth is about 6.5 kHz, and the phase margin is about 47°. The transfer function of the designed PID controller using equation (3.13) is,

$$G_C(s) = 2 + \frac{6.74*10^3}{s} + 6.78 * 10^{-6}s \tag{3.14}$$

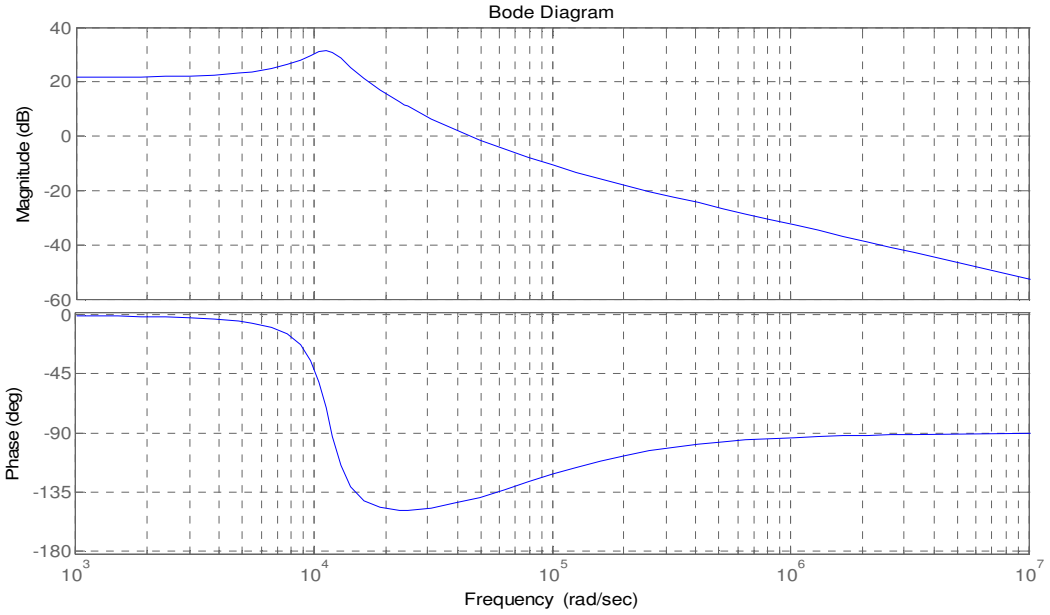


Figure 3.7 Open loop Bode plot of the red color synchronous buck driver.

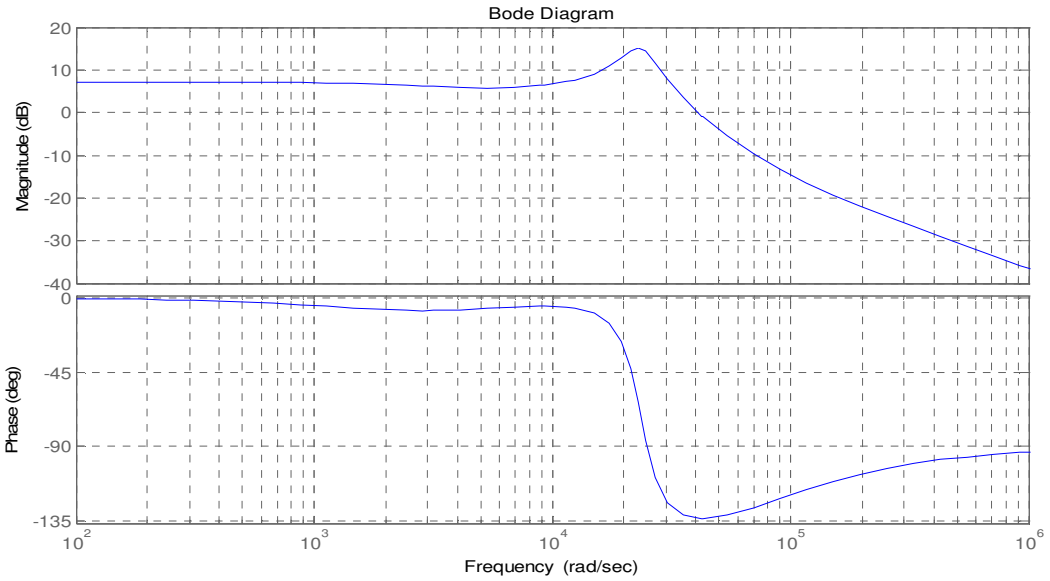


Figure 3.8 Closed loop Bode plot of the red color synchronous buck driver.

Since the controller is implemented using the PIC18F4431 microcontroller, the analog controller shown in equation (3.14) needs to be transformed into its digital equivalent using matched pole-zero method with a sampling frequency of 20 µs. Therefore, the difference equation of (3.12) takes the form of,

$$u(k) = K_1 e(k) + K_2 \sum_{i=0}^{k} e[i] + K_3\{e[k] - e[k-1]\} \tag{3.15}$$

where u(k) is the controllers output, e(k) is the controllers input, and K_1, K_2, and K_3 are the gains of the PID controller in z-domain [43]. This difference equation is implemented in the PIC18F4431 microcontroller. Thus, using MATLAB command 'c2d' $G_C(z)$ becomes,

$$G_C(z) = \frac{2.04z^2 - 1.91z + 5.7*10^{-3}}{z-1} \tag{3.16}$$

$$u(k) = 2.04e(k) - 1.91 \sum_{i=0}^{k} e[i] + 5.7 * 10^{-3}\{e[k] - e[k-1]\} \tag{3.17}$$

For green and blue LEDs, whose forward voltages are different from red LEDs, R_{LOAD} of the synchronous buck converter for these colors are found to be,

$$R_{EQU} = \frac{3.2}{0.020} = 160\ \Omega \tag{3.18}$$

$$R_{SINGLE_{STRING}} = R_{EQU} + R_{MOSFET} + R_{SENSE} = 160 + 1.8 + 10 = 171.8\ \Omega \tag{3.19}$$

$$R_{LOAD} = R_{SINGLE\_STRING\_1} // \ldots\ldots // R_{SINGLE\_STRING\_9} = 19.1\ \Omega \tag{3.20}$$

Substituting the values of R_{LOAD} from equation (3.20) and the parameter values from table (3.2), the transfer function for the green and blue synchronous buck driver is,

$$G_P(s) = \frac{0.00018s + 12}{7.53*10^{-9}s^2 + 2.589*10^{-5}s + 1} \tag{3.21}$$

Using MATLAB, the open loop Bode plot of the above transfer function is shown in Fig. 3.9. The transfer function has two conjugate poles at – (1.72±j11.40)*10^3 radian/sec, and left half plane zero at -6.7*10^4 radian/sec. Based on these results, a suitable PID controller is designed in MATLAB,

$$G_C(s) = 7.4 + \frac{1.7*10^4}{s} + 2.96 * 10^{-6}s \tag{3.22}$$

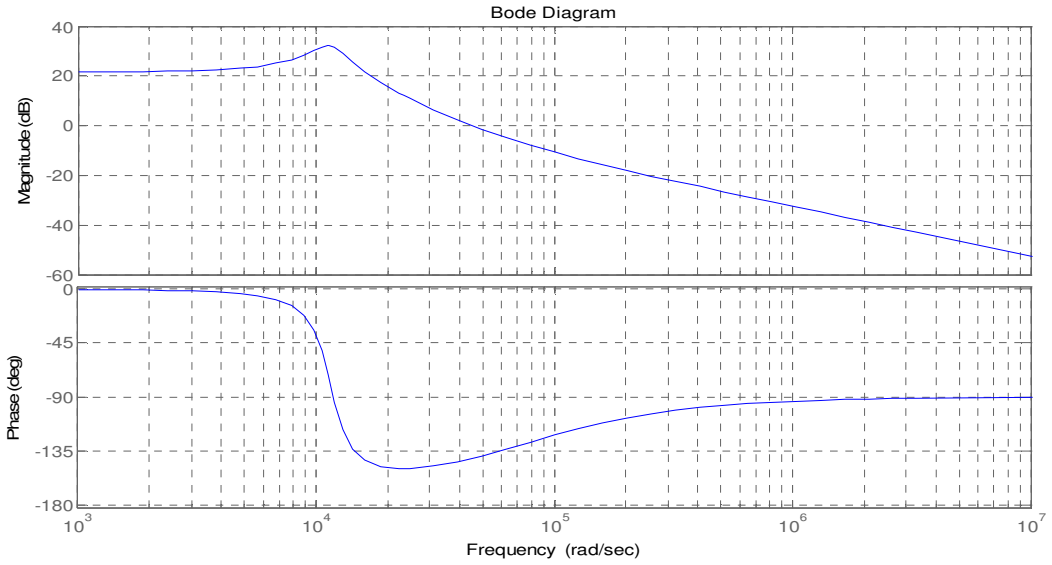


Figure 3.9 Open loop Bode plot of the green and blue color synchronous buck driver.

Using MATLAB, the PID controller was designed with a pole in the origin, and two zeros at -2.29*10^3 radian/sec and -2.50*10^6 radian/sec, respectively. The Bode plot of the green and blue color closed loop synchronous buck driver is shown in Fig. 3.10 and the model used for Bode plot is shown in Appendix A. From Fig. 3.10, it can be seen that the system bandwidth is about 13 kHz, and the phase margin is about 64°.

Similar to the red color LED, the analog controller shown in equation (3.22) needs to be transformed into its digital equivalent using the matched pole-zero method with a sampling

frequency of 20 µs. Thus, using MATLAB command 'c2d' $G_C(z)$ becomes,

$$G_C(z) = \frac{7.55-7.21z}{z-1} \tag{3.23}$$

$$u(k) = 7.55e(k) - 7.21\sum_{i=0}^{k} e[i] \tag{3.24}$$

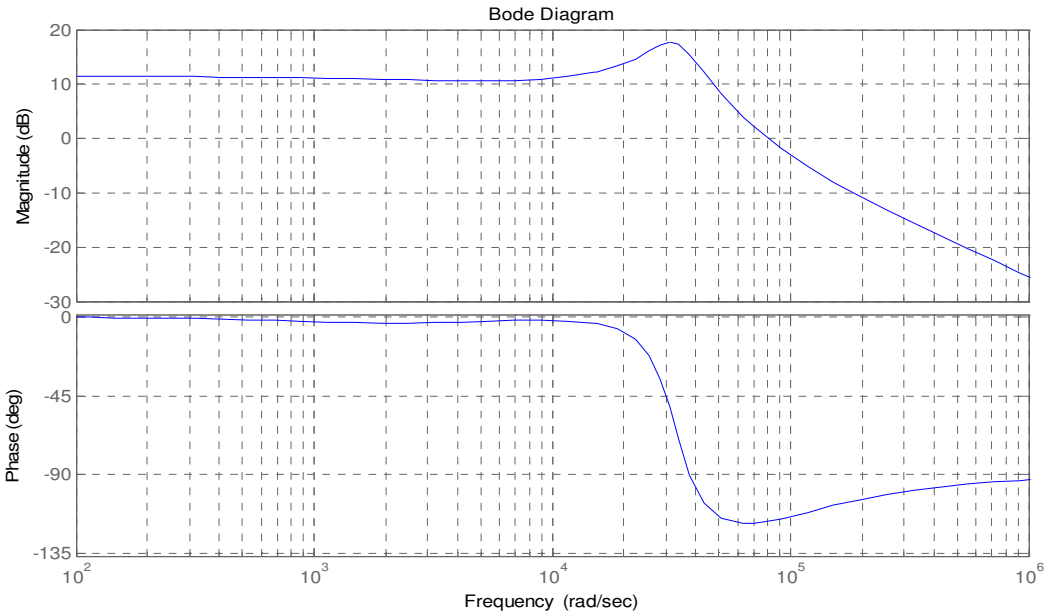


Figure 3.10 Closed loop Bode plot of the green and blue color synchronous buck driver.

3.5 SIMULATION RESULTS

The performance characteristics of red, green, and blue LED drivers was verified using MATLAB SIMULINK during both steady state and transient operation [1],[6]. State space equations are used to linearize the synchronous buck converter as shown in Fig. 3.11 [45],[46],

$$\frac{di_L}{dt} = \frac{1}{L}(V_{IN} * d - i_L * R_{LOAD} - V_O) \tag{3.25}$$

$$\frac{dv_C}{dt} = \frac{1}{C}(i_L - I_O) \tag{3.26}$$

$$V_O = v_C + R_C(i_L - I_O) \tag{3.27}$$

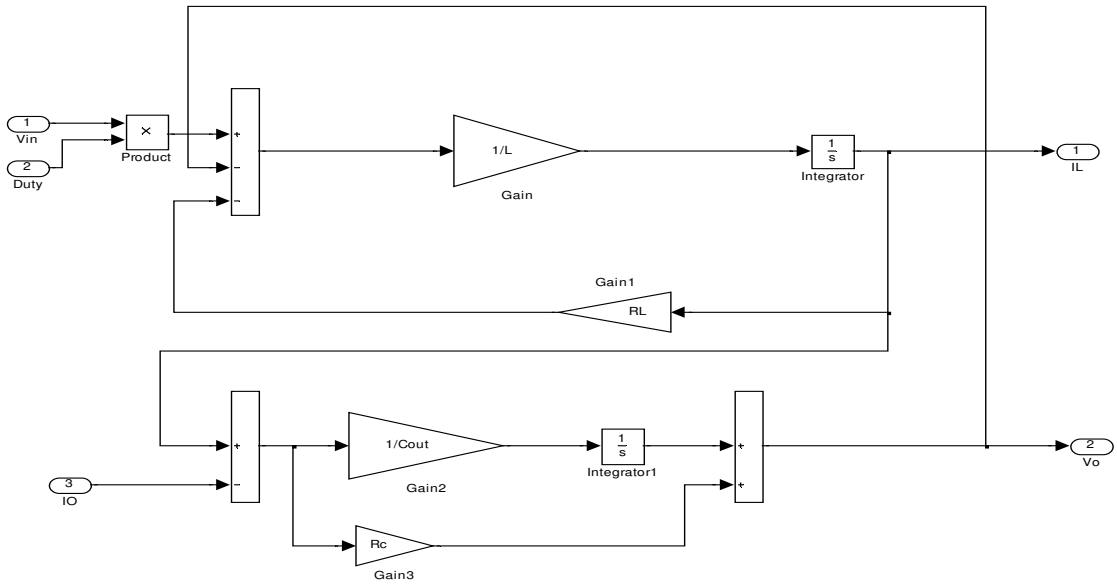


Figure 3.11 State space model for the synchronous buck converter.

The complete red color LED driver model implemented in MATLAB SIMULINK is shown in Fig. 3.12. Similar models are implemented for green and blue color drivers. LED loads are modeled as resistive loads as shown in equations (3.10) and (3.20). Since the DC equivalent resistance of the LED string reduces when the current across it reduces, thus the drive voltage of the LED strings needs to be adjusted accordingly. The drive voltage of the LED strings needs to be reduced in order to maintain a minimum voltage drop across the current controllers of the LED strings, leading to high efficiency at the LED load during load transient. Thus, this can be modeled by increasing the feedback voltage during load transients as shown in Fig. 3.12, resulting in decrease of drive voltage due to the difference between the reference voltage and feedback voltage, i.e.,

$$V_{ERROR} = V_{REF} - V_{FB} \tag{3.26}$$

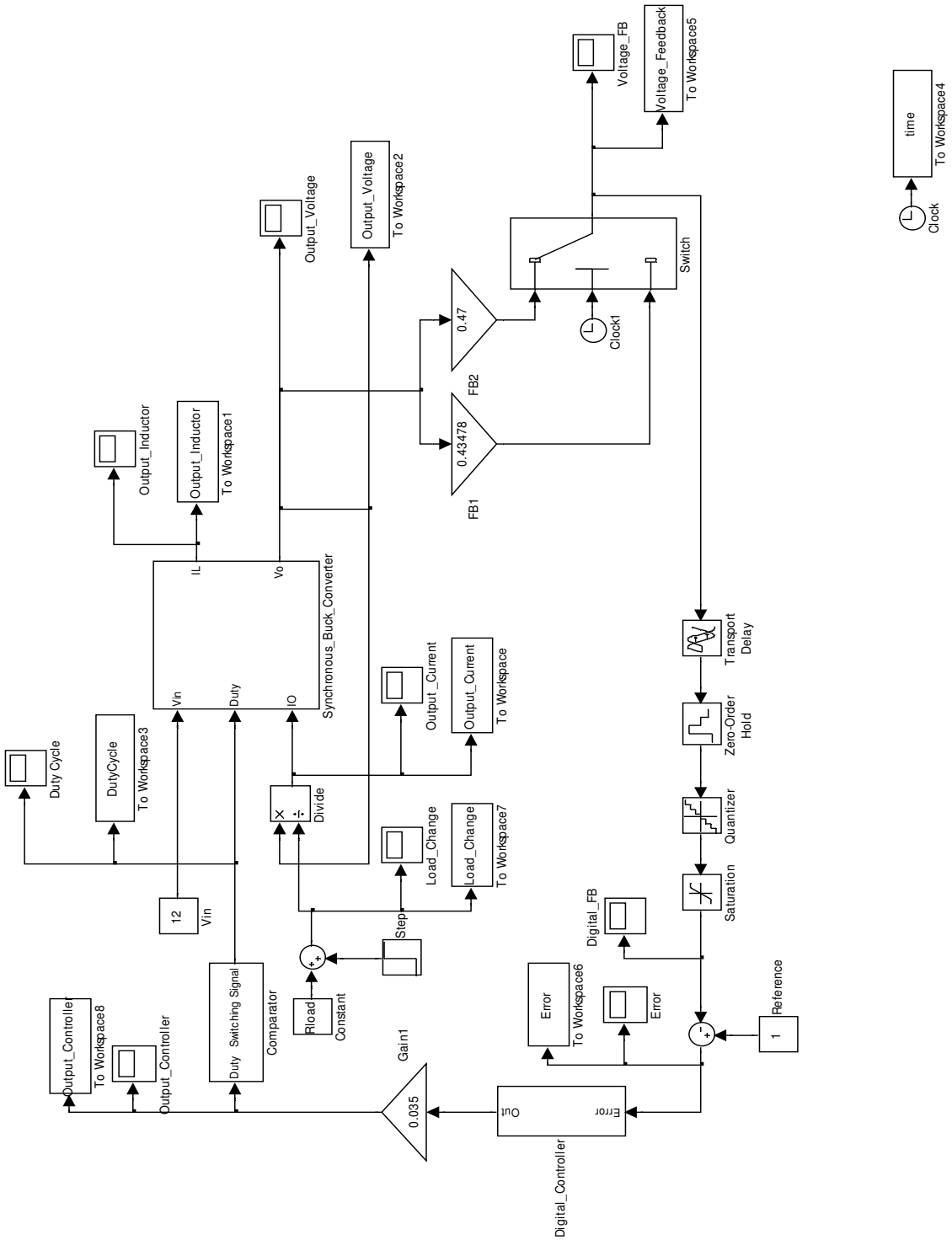
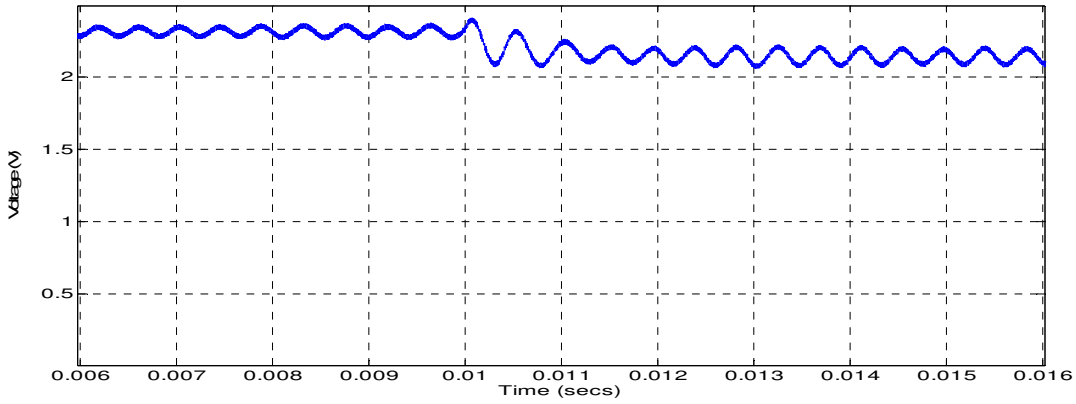


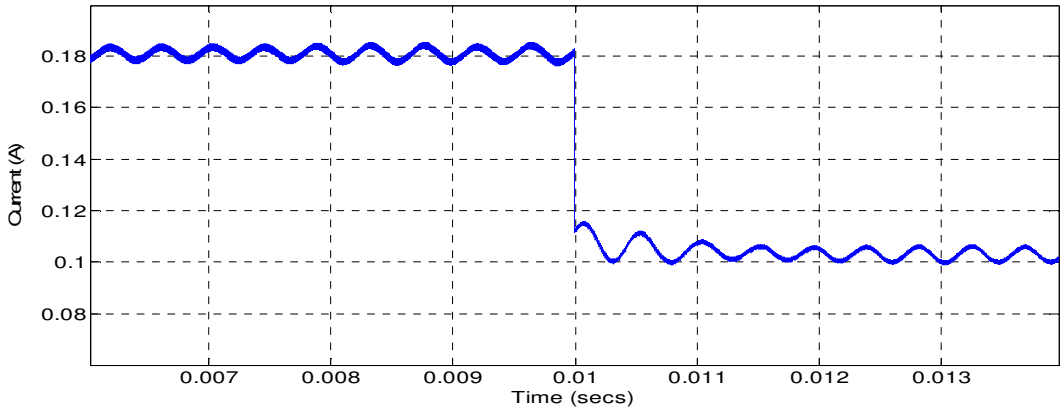
Figure 3.12 Red color LED driver model implemented in MATLAB SIMULINK.

The ADC is modeled using zero-order hold and quantizer blocks from SIMULINK library. The digital controller are modeled according to equation (3.15), and the comparator is used to compare the output of the controller with a sawtooth signal in order to generate the proper PWM signals for the synchronous buck converter [45]-[48]. The model was simulated using ODE3 (Bogacki-Shampine) differential equations. A fixed step size of $0.1*10^{-6}$ seconds was used to obtain accurate results.

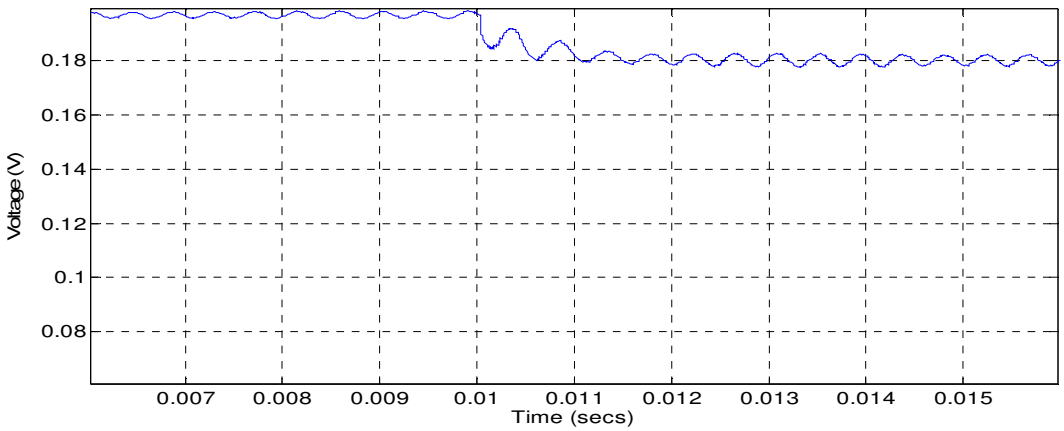
The simulation result for the red color LED driver during a load transient from 180 mA to 104 mA is shown in Fig. 3.13. Fig. 3.13 (a) shows the output voltage being adjusted to a minimum value within 2ms from 2.3 V to 2.15 V after transient in order to maintain the desired current, leading to lower power losses in the current controllers of the LED strings. The ripple output voltage is ± 50 mV which is above the specified limit. Fig. 3.13 (b) shows the output current reaches the new steady state value within 1ms after transient, since refresh rate of 1000 Hz or greater are required for LEDs used in display panel applications. The ripple output current is ± 5 mA which is below the specified limit. Fig. 3.13 (c) shows the output of the digital controller being changed from 0.197 V to 0.18 V during transient to adjust the output voltage to a minimum value. Fig. 3.13 (d) shows the inductor current indicating that the synchronous buck converter is operating in continuous conduction mode. Note that a synchronous converter only operates in continuous conduction mode.


Voltage (V)
2
1.5
1
0.5
0.006
0.007
0.008
0.009
0.01
0.011
0.012
0.013
0.014
0.015
0.016
Time (secs)

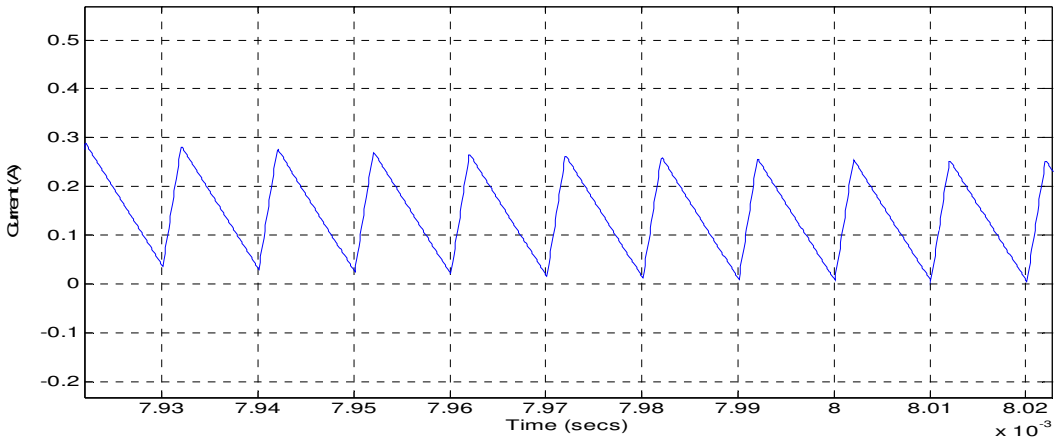
(a)


Current (A)
0.18
0.16
0.14
0.12
0.1
0.08
0.007
0.008
0.009
0.01
0.011
0.012
0.013
Time (secs)

(b)


Voltage (V)
0.18
0.16
0.14
0.12
0.1
0.08
0.007
0.008
0.009
0.01
0.011
0.012
0.013
0.014
0.015
Time (secs)

(c)

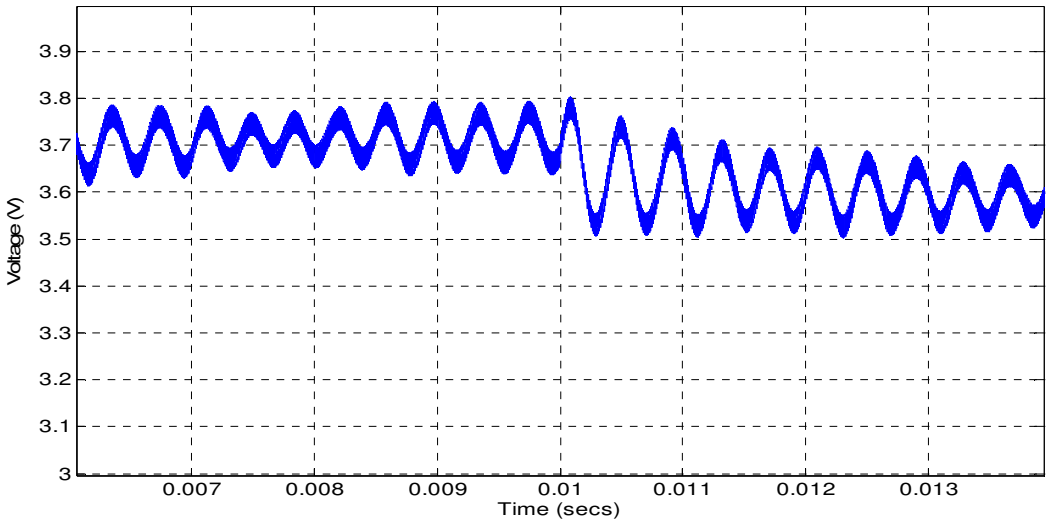


(d)

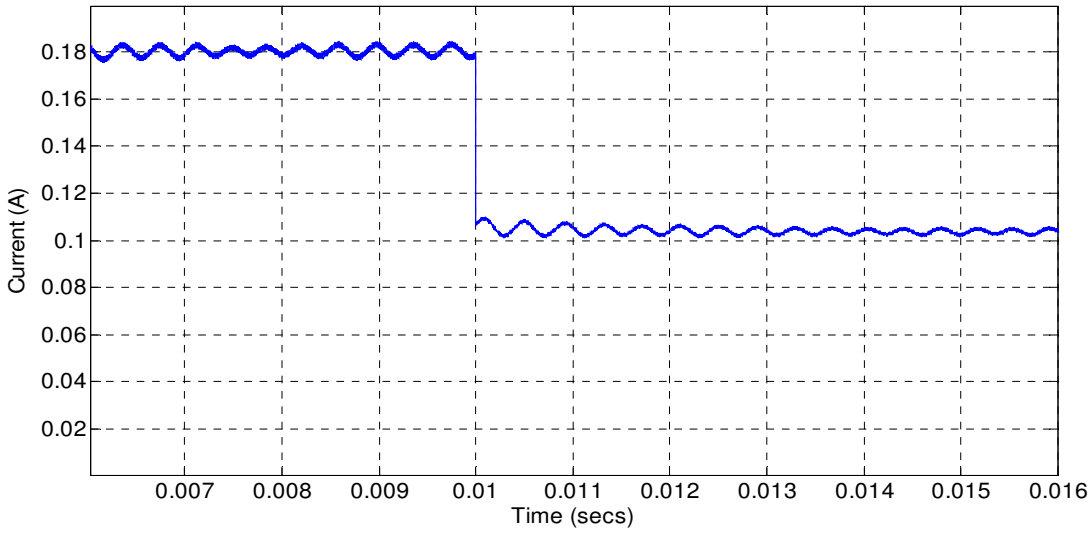
Figure 3.13 (a) Output voltage, (b) Output Current, (c) Output of the digital controller, and (d) Inductor Current of the red color LED driver.

Similarly, the simulation results for the green and blue color LED drivers during a load transient from 180 mA to 104 mA are shown in Fig. 3.14. Since the typical forward voltages of the green and blue LEDs are equal, the synchronous buck converter transfer function shown in equation (3.21) applies to both colors, thus only one model for green and blue color LED driver was designed in MATLAB SIMULINK. Fig. 3.14 (a) shows the output voltage being adjusted to a minimum value within 2ms from 3.7 V to 3.56 V after transient in order to maintain the desired current, leading to lower power losses in the current controllers of the LED strings. The ripple output voltage is ± 80 mV which is above the specified limit. Fig. 3.14 (b) shows the output current reaches the new steady state value within 1ms after transient, since refresh rate of 1000 Hz or greater are required for LEDs used in display panel applications. The ripple output current is ± 5 mA which is below the specified limit. Fig. 3.14 (c) shows the output of the digital controller being changed from 0.307 V to 0.298 V during transient to adjust the output voltage to

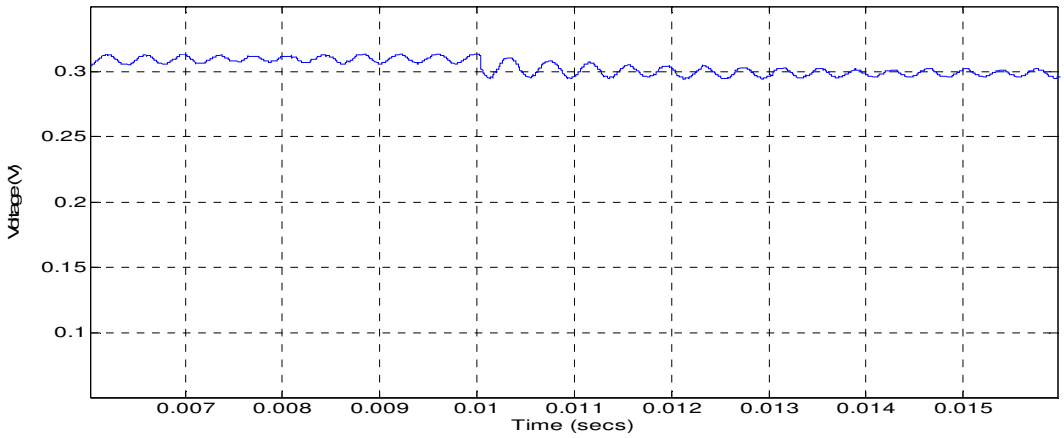
a minimum value. Fig. 3.14 (d) shows the inductor current indicating that the synchronous buck converter is operating in continuous conduction mode.



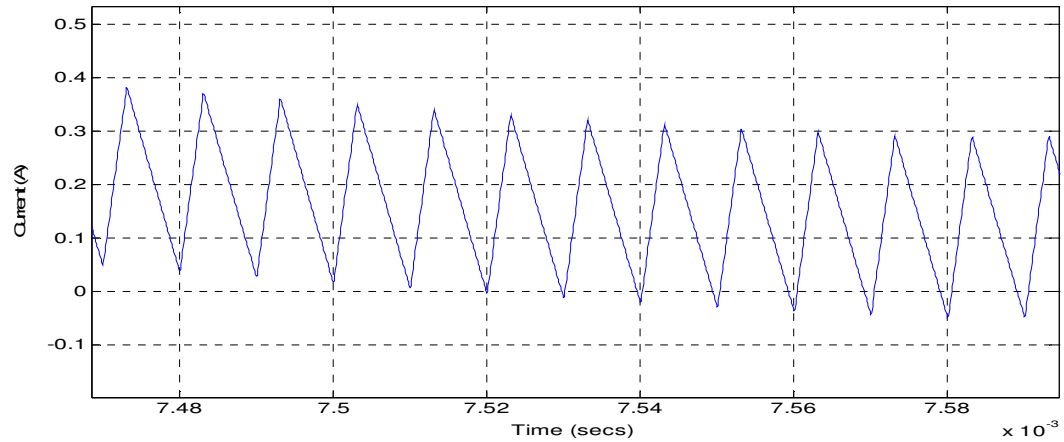
(a)



(b)



(c)



(d)

Figure 3.14 (a) Output voltage, (b) Output Current, (c) Output of the digital controller, and (d) Inductor Current of the green and blue color LED driver.

3.6 EXPERIMENTAL RESULTS

A prototype was built to verify the performance characteristics of the red, green, and blue LEDs drivers for a 3x3 RGB pixel. The synchronous buck converter provides the drive voltage for the

LED strings for each color, and its key parameters and components are listed in Table 3.2. A photograph of the prototype RGB driver for a 3x3 RGB pixel is shown in Fig. 3.15.

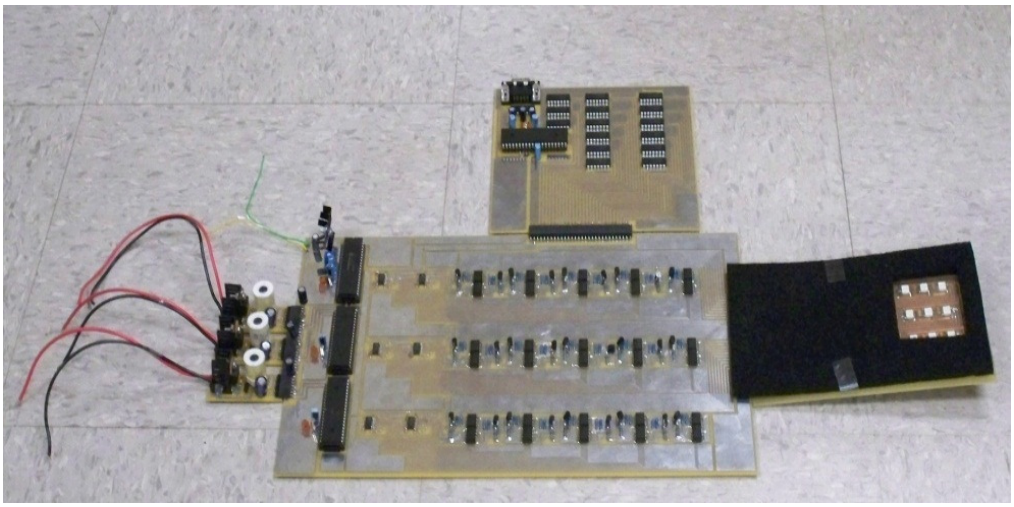


Figure 3.15 Photograph of the prototype RGB driver for 3x3 RGB pixels.

In order to verify the mode of operation of the synchronous buck driver for the red LED, the inductor current is measured at a maximum driver current of 180 mA, and is shown in Fig. 3.16. The upper waveform is the measured high side PWM signal (Channel 1), the second waveform is the measured low side PWM signal (Channel 2), the third waveform is the measured output voltage (Channel 3), and the bottom waveform is the measured inductor current (Channel 4) of the synchronous buck driver for the red color. From the measured output voltage, the peak to peak ripple voltage is ± 80 mV which is above the specified limit. Additionally, from the figure, the measured inductor current implies the synchronous buck converter is operating in continuous conduction mode.

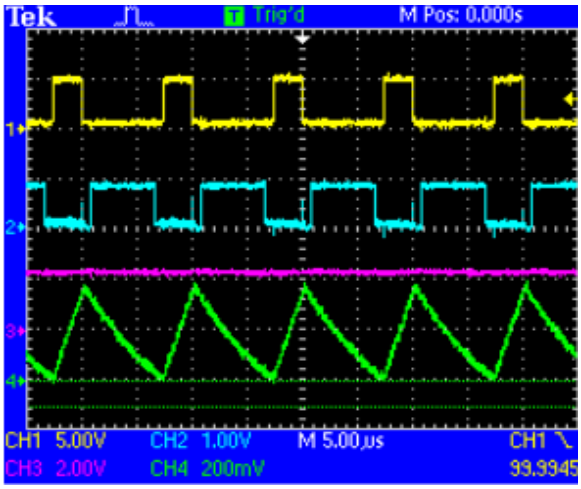


Figure 3.16 Measured PWM signal, output voltage, and inductor current of the synchronous buck driver for red LEDs.

In order to maintain a minimum voltage drop across the current controllers of the LED strings for the desired load current, an efficiency optimization mode is implemented in the PIC18F4431 as explained in section 3.2. Fig. 3.17 shows the measured output voltage of the synchronous buck driver for the red LEDs (upper waveform, Channel 1) and the measured duration of the efficiency optimization mode (bottom waveform, Channel 2) at startup. It can be observed that initially, the output voltage of the driver is about 5 V, thus the MOSFETs of the current controllers are in saturation region of operation leading to unwanted power losses for the set current in the LED strings. Then, the efficiency optimization mode is enabled for duration of 6ms in order to reduce the output voltage to its minimum value necessary to maintain the set current, thus forcing the MOSFETs of the current controllers to operate with minimum V_{DS}, leading to increased efficiency at the LED load.

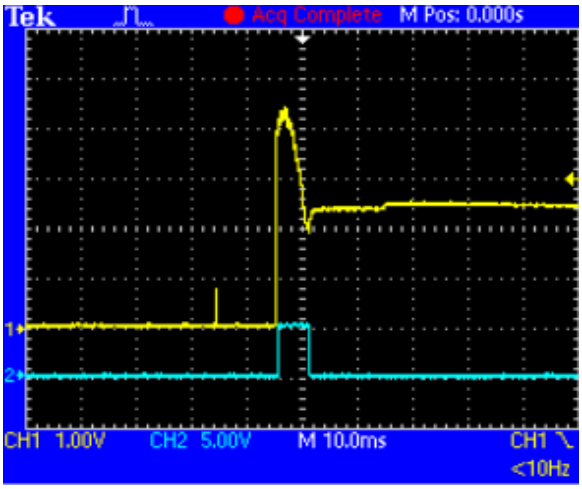


Figure 3.17 Measured output voltage and duration of efficiency optimization mode for synchronous buck driver for red LEDs.

The load transient from 180 mA to 104 mA is shown in Fig. 3.18 for the red LEDs. The upper waveform (Channel 1) is the measured output voltage, and the lower waveform (Channel 2) is the measured output current of the synchronous buck driver for the red LEDs. As can be seen from the figure, that the output current transients to its value rapidly, thus the refresh rate of the red LEDs is about 1000 Hz. Additionally, the decrease in I_D leads to an increase in V_{DS} in the MOSFETs of the current controllers, leading to an increase in output voltage of the driver as shown in the figure. Thus, the efficiency optimization mode is enabled, to find the minimum drive voltage for the LED strings for the new set current as the output voltage transitions from 2.36 V to 2.2 V. The ripple output current of the driver is ± 60 mA which is below the specified limit. Fig. 3.19 shows the load transient from 180 mA to 0 mA. The upper waveform (Channel 1) is the output voltage, and the lower waveform (Channel 2) is the output current of the synchronous buck driver for the red LEDs. As can be seen in the figure, zero output current of the driver results in zero output drive voltage, thus the driver is turned off in this case.

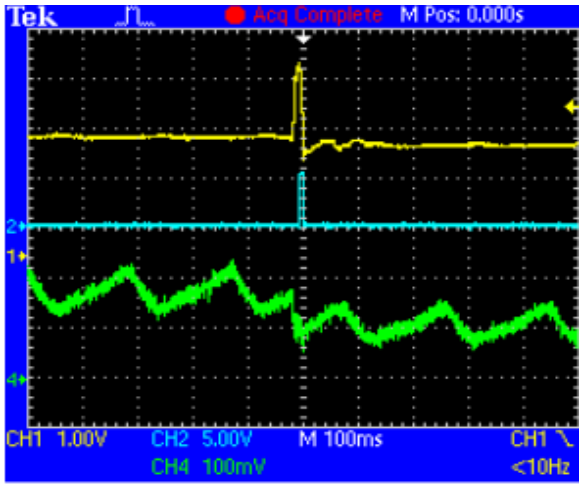


Figure 3.18 Measured output voltage, duration of efficiency optimization mode, and output current during load transients from 180 mA to 104 mA.

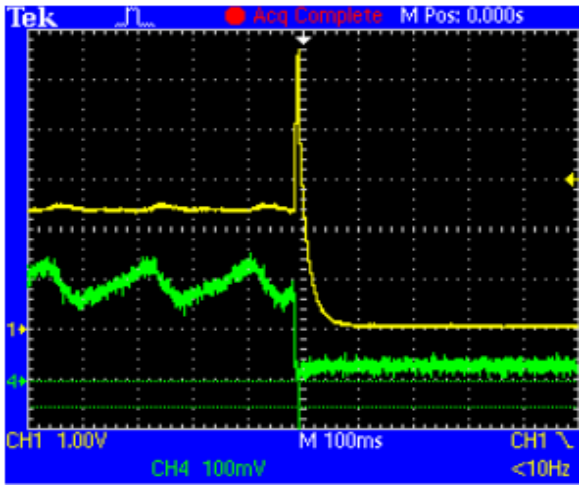


Figure 3.19 Measured output voltage, and output current during load transients from 180 mA to 0 mA.

In Fig. 3.20, the measured output current and voltage as a function of input voltage is shown for the red LED driver at maximum driver current of 180 mA. As observed in the figure, the variation is within 1 % for output voltage and current, respectively.

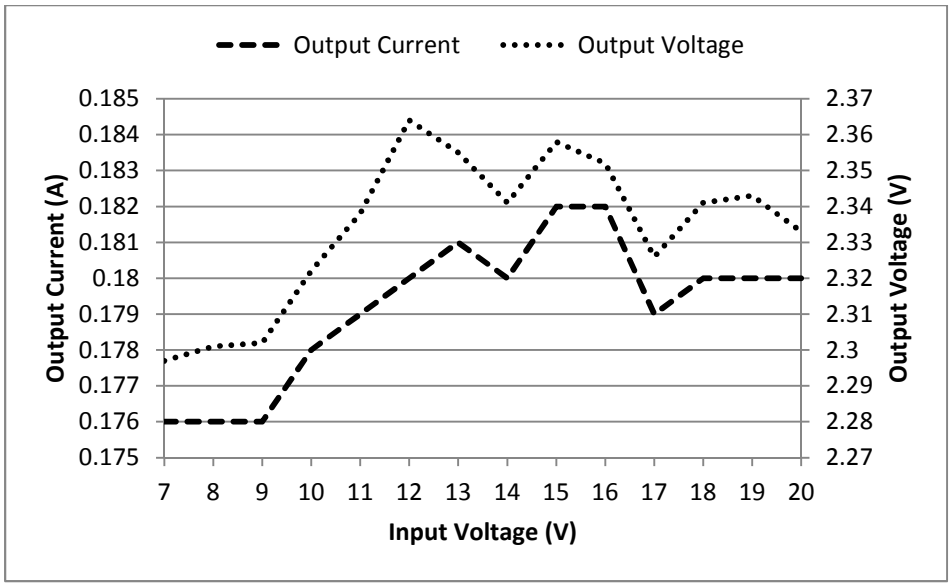


Figure 3.20 Measured output voltage and current for different input voltages.

The ratio of the power dissipated by the red LED to the total power dissipated by the LED string is defined as the efficiency of the LED string as in equation (3.27) [4]. The total power dissipated by the LED, MOSFETs of the current controllers, and the sensing resistor is defined as the total power loss in the LED string [4]. The efficiency and the forward LED voltage drop of a single red LED string for various dimming signals are shown in Fig. 3.21. From the figure, we can deduce that as the voltage drop across the red LED decreases, the efficiency of the LED string increases.

$$Efficiency = \frac{P_{LED}}{P_{STRING}} x\ 100 \tag{3.27}$$

In order to verify the mode of operation of the synchronous buck driver for the green LED, the inductor current is measured at a maximum driver current of 180 mA and is shown in Fig. 3.22. The upper waveform is the measured high side PWM signal (Channel 1), the second waveform is the measured low side PWM signal (Channel 2), the third waveform is the measured output

voltage (Channel 3), and the bottom waveform is the measured inductor current (Channel 4) of the synchronous buck driver for the green color. From the measured output voltage, the peak to peak ripple voltage is ± 60 mV which is above the specified limit. Additionally, from the figure, the measured inductor current implies that the synchronous buck converter is operating in continuous conduction mode.

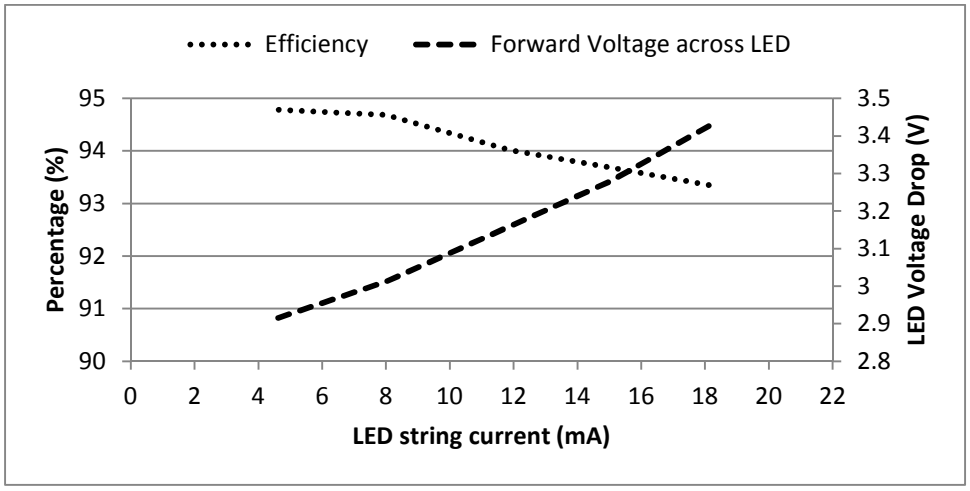


Figure 3.21 Measured efficiency, and LED voltage drop for various dimming signals for a red LED string.

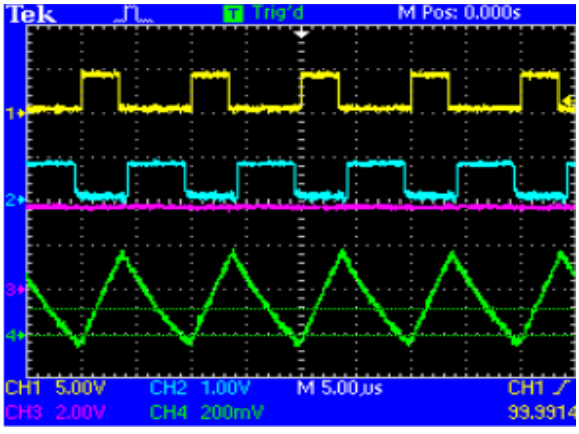


Figure 3.22 Measured PWM signal, output voltage, and inductor current of the synchronous buck driver for green LEDs.

In order to maintain a minimum voltage drop across the current controllers of the LED strings for the set current, an efficiency optimization mode is implemented in PIC18F4431 as explained in section 3.2. Fig. 3.23 shows the measured output voltage of the synchronous buck driver for the green LEDs (upper waveform, Channel 1) and the measured duration of the efficiency optimization mode (bottom waveform, Channel 2) at startup. It can be observed that initially, the output voltage of the driver is about 5 V, thus the MOSFETs of the current controllers are in saturation region of operation leading to unwanted power losses for the set current in the LED strings. Then, the efficiency optimization mode is enabled for a duration of 6ms to reduce the output voltage to its minimum value necessary to maintain the set current, thus forcing the MOSFETs of the current controllers to operate with minimum V_{DS}, leading to increased efficiency at the LED load.

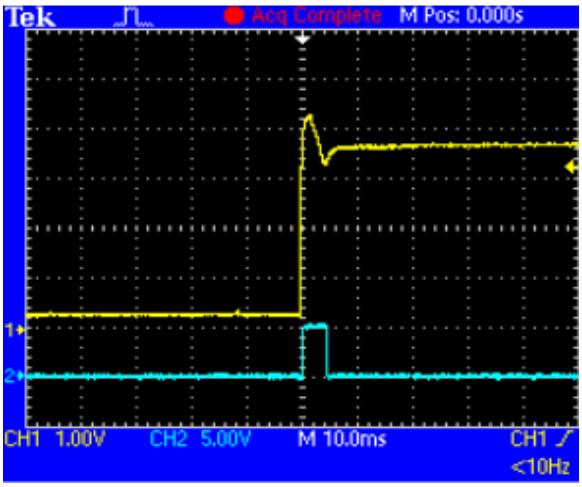


Figure 3.23 Measured output voltage and duration of efficiency optimization mode for synchronous buck driver for green LEDs.

The load transient from 180 mA to 104 mA is shown in Fig. 3.24 for the green LEDs. The upper waveform (Channel 1) is the measured output voltage, and the lower waveform (Channel 2) is the measured output current of the synchronous buck driver for the green LEDs. As can be seen

from the figure, the output current transients to its value rapidly, thus the refresh rate of the green LEDs is about 1000 Hz. Additionally, the decrease in I_D leads to increase in V_{DS} in the MOSFETs of the current controllers, leading to increase in output voltage of the driver as shown in the figure. Thus, the efficiency optimization mode is enabled, to find the minimum drive voltage for the LED strings for the new set current as the output voltage transitions from 3.66 V to 3.37 V. The ripple output current of the driver is ± 50 mA which is below the specified limit. Fig. 3.25 shows the load transient from 180 mA to 0 mA. The upper waveform (Channel 1) is the output voltage, and the lower waveform (Channel 2) is the output current of the synchronous buck driver for the green LEDs. As can be noticed in the figure, zero output current of the driver results in zero output drive voltage, thus the driver is turned off in this case.

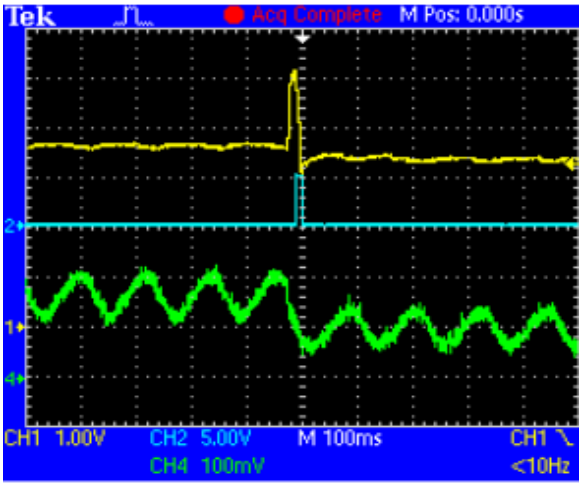


Figure 3.24 Measured output voltage, duration of efficiency optimization mode, and output current during load transients from 180 mA to 104 mA.

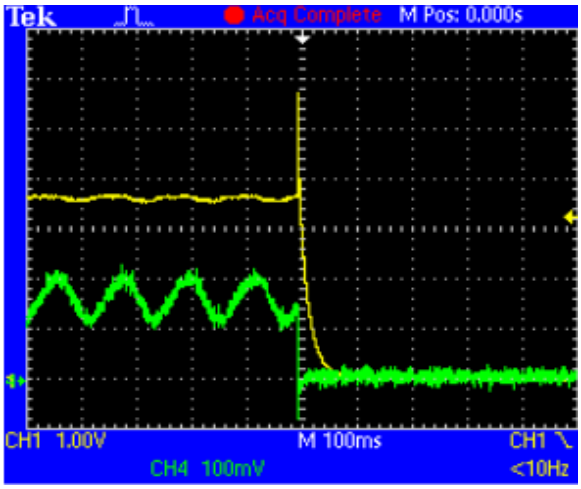


Figure 3.25 Measured output voltage, and output current during load transients from 180 mA to 0 mA.

In Fig. 3.26, the measured output current and voltage as a function of input voltage is shown for the green LED driver at maximum driver current of 180 mA. As observed in the figure, the variation is within 1% and 2% for output voltage and current, respectively. The efficiency and forward LED voltage drop of a single green LED string for various dimming signals are shown in Fig. 3.27. From the figure, we can deduce that as the voltage drop across the green LED decreases, the efficiency of the LED string increases.

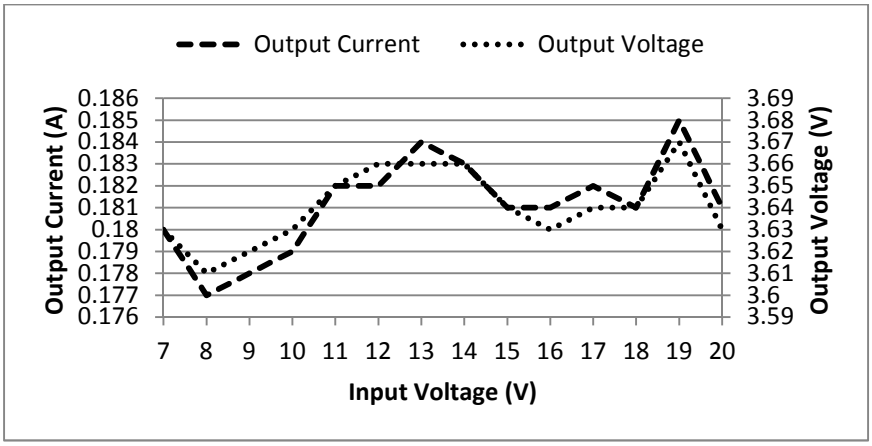


Figure 3.26 Measured output voltage and current for different input voltages.

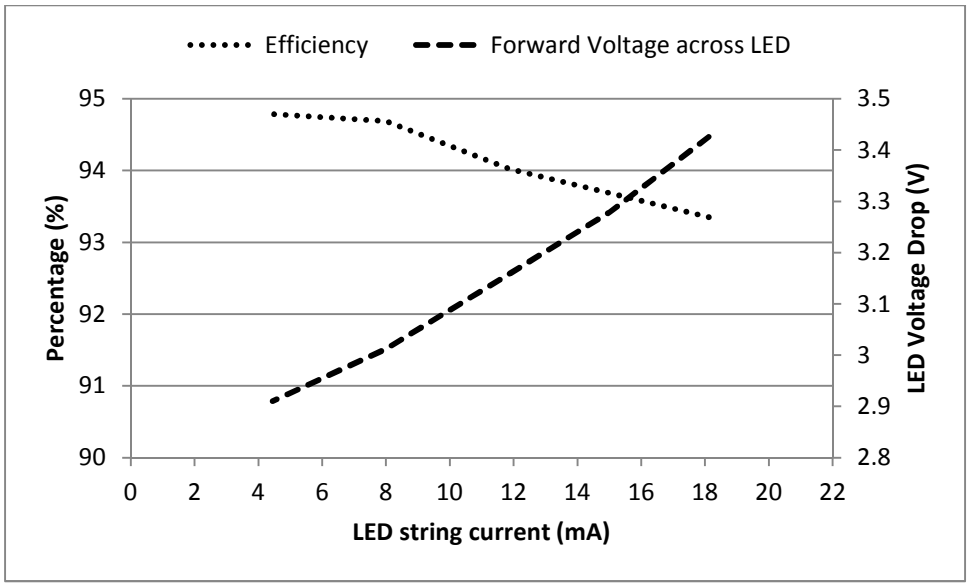


Figure 3.27 Measured efficiency, and LED voltage drop for various dimming signals for a green LED string.

Similar to red and green LEDs, in order to verify the mode of operation of the synchronous buck driver for the blue LED, the inductor current is measured at a maximum driver current of 180 mA, and is shown in Fig. 3.28. The upper waveform is the measured high side PWM signal (Channel 1), the second waveform is the measured low side PWM signal (Channel 2), the third waveform is the measured output voltage (Channel 3), and the bottom waveform is the measured inductor current (Channel 4) of the synchronous buck driver for the blue color. From the measured output voltage, the peak to peak ripple voltage is ± 40 mV which is above the specified limit. Additionally, from the figure, the measured inductor current implies that the synchronous buck converter is operating in continuous conduction mode.

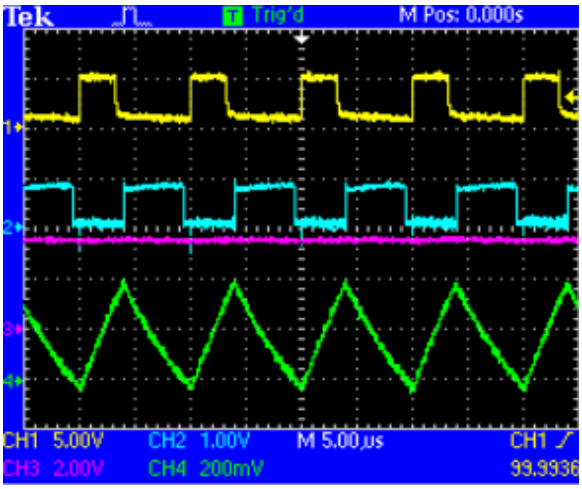


Figure 3.28 Measured PWM signal, output voltage, and inductor current of the synchronous buck driver for blue LEDs.

In order to maintain a minimum voltage drop across the current controllers of the LED strings for the set current, an efficiency optimization mode is implemented in the PIC18F4431 as explained in section 3.2. Fig. 3.29 shows the measured output voltage of the synchronous buck driver for the blue LEDs (upper waveform, Channel 1) and the measured duration of the efficiency optimization mode (bottom waveform, Channel 2) at startup. It can be observed that initially the output voltage of the driver is about 5 V. Thus, the MOSFETs of the current controllers are in the saturation region of operation leading to unwanted power losses for the set current in the LED strings. Then, the efficiency optimization mode is enabled for a duration of 6ms to reduce the output voltage to its minimum value necessary to maintain the set current, thus forcing the MOSFETs of the current controllers to operate with minimum V_{DS}, leading to increased efficiency at the LED load.

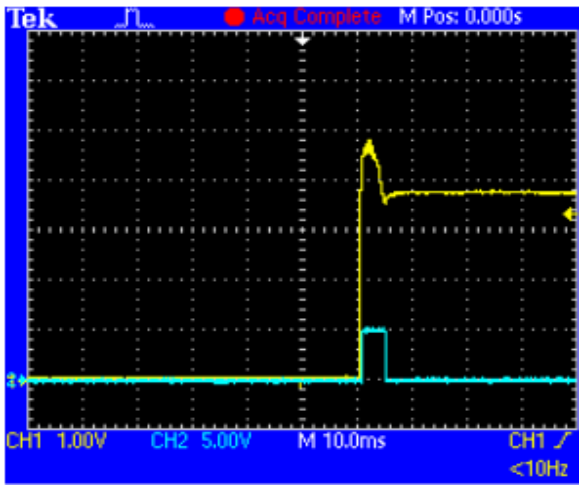


Figure 3.29 Measured output voltage and duration of efficiency optimization mode for synchronous buck driver for blue LEDs.

The load transient from 180 mA to 104 mA is shown in Fig. 3.30 for the blue LEDs. The upper waveform (Channel 1) is the measured output voltage, and the lower waveform (Channel 2) is the measured output current of the synchronous buck driver for the blue LEDs. As can be seen from the figure, that the output current transients to its value rapidly, thus the refresh rate of the blue LEDs is about 1000 Hz. Additionally, the decrease in I_D leads to an increase in V_{DS} in the MOSFETs of the current controllers, leading to an increase in output voltage of the driver as shown in the figure. Thus, the efficiency optimization mode is enabled, in order to find the minimum drive voltage for the LED strings for the new set current as the output voltage transitions from 3.7 V to 3.4 V. The ripple output current of the driver is ± 60 mA which is below the specified limit. Fig. 3.31 shows the load transient from 180 mA to 0 mA. The upper waveform (Channel 1) is the output voltage, and the lower waveform (Channel 2) is the output current of the synchronous buck driver for the blue LEDs. As can be seen from the figure, a zero output current of the driver results in zero output drive voltage, thus the driver is turned off in this case.

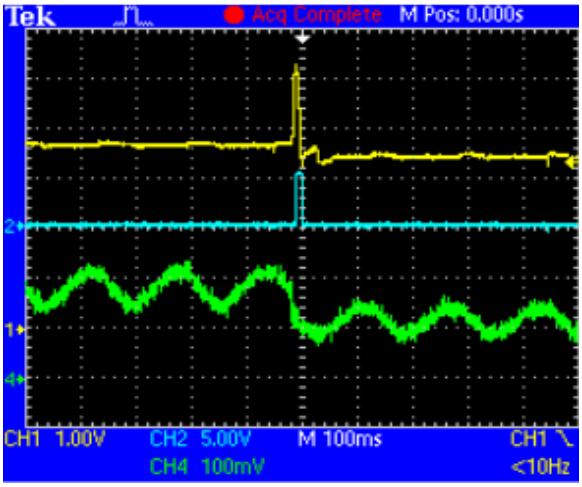


Figure 3.30 Measured output voltage, duration of efficiency optimization mode, and output current during load transients from 180 mA to 104 mA.

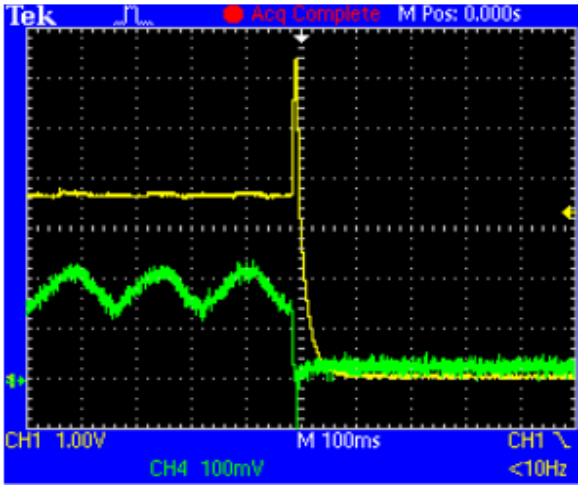


Figure 3.31 Measured output voltage, and output current during load transients from 180 mA to 0 mA.

In Fig. 3.32, the measured output current and voltage as a function of input voltage is shown for the blue LED driver at maximum driver current of 180 mA. As observed in the figure, the variation is within 2% and 3% for output voltage and current, respectively. The efficiency and the forward LED voltage drop of a single blue LED string for various dimming signals are shown in Fig. 3.33. From the figure, we can deduce as the voltage drop across the blue LED

decreases, the efficiency of the LED string increases. Fig. 3.34 shows the 3x3 RGB display panel when all the LED strings are set at its maximum current of 20 mA.

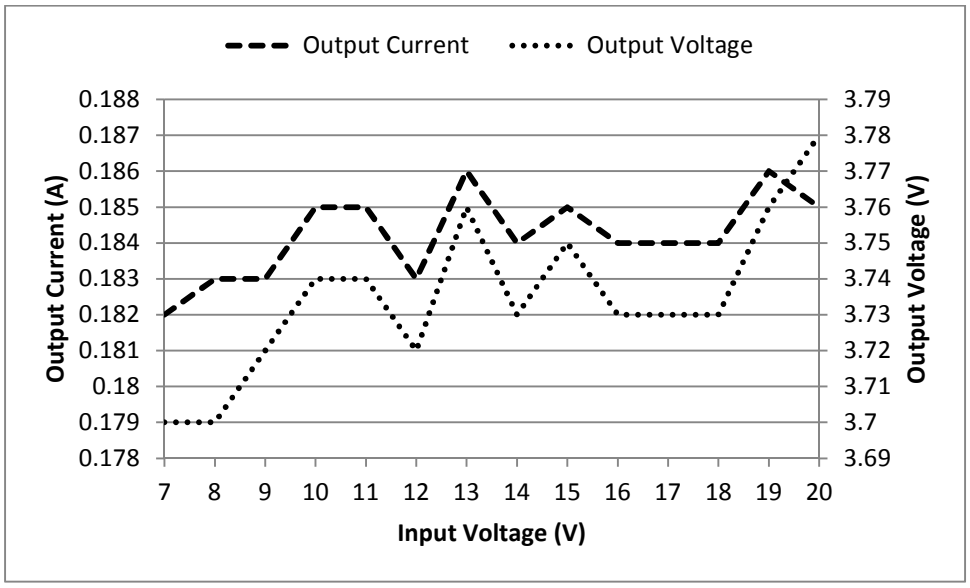


Figure 3.32 Measured output voltage and current for different input voltages.

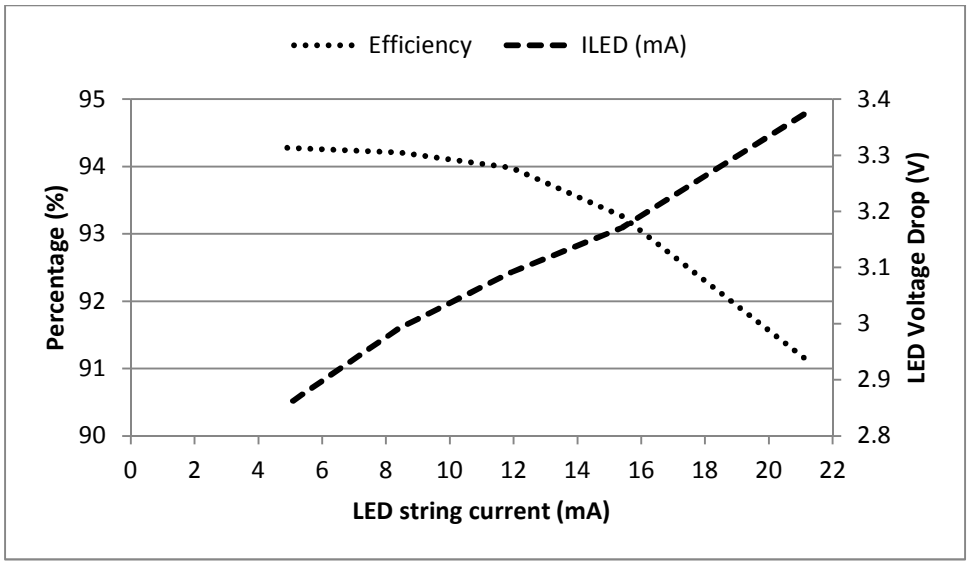


Figure 3.33 Measured efficiency, and LED voltage drop for various dimming signals for a blue LED string.

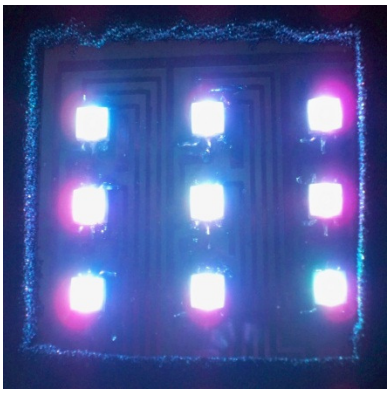


Figure 3.34 3x3 RGB pixels.

Table 3.3 Comparison of RGB driver between the proposed design and the design in [49].

Parameter	RED		GREEN		BLUE	
	Our Work	[49]	Our Work	[49]	Our Work	[49]
Efficiency (%)	>86%	>90%	>90%	>90%	>90%	>90%
Input Voltage (V)	12	240	12	240	12	240
Output Voltage (V)	2.3	82	3.66	130	3.72	120
Max Output Current (mA)	180	500	180	375	180	330
Frequency (kHz)	100	50	100	50	100	50

The designed RGB driver was compared with the RGB drivers designed in [49] and major performance characteristics are summarized in Table 3.3 [4]. The LED load in [49] consisted of three LED strings in parallel, with each strings having 36 RGB LEDs for backlight applications, whereas in the proposed RGB driver there are nine parallel strings for each color of LED drivers with each strings having a single LED for display panel applications. Comparing the efficiencies of the LEDs, green and blue LEDs efficiencies are similar in both the designs but the efficiency of red LED driver is lower in the proposed design than the red LED driver in [49] due to the effect of V_{DS} of the current controllers, i.e., the lower the drive voltage of the LED string, the

more significant the effect of V_{DS} on the efficiency of the LED load. Since the drive voltage for the proposed red LED driver is 2.3 V, thus in order to obtain efficiency of 90% only a V_{DS}=0.24 V is allowed, whereas in the red LED driver in [49] that has a drive voltage of 82 V in which case a V_{DS}=8 V is allowed for an efficiency of 90% [4].

3.7 SUMMARY

The proposed driver for driving a 3x3 RGB display panel using a digital controller is demonstrated in this chapter. In this implementation, the efficiency at the LED load is maximized by driving the LED strings with a minimum drive voltage, thus reducing unwanted power losses in the current controllers. Additionally, an external dimming controller interface is able to dim each LEDs individually in a 3x3 RGB display panel leading to high color contrast. Using the current controllers to maintain current in the LED strings, and synchronous buck converter to provide the drive voltage leads to better system stability since the current and voltage are controlled independently. Efficiency of 85.6 %, 93.3%, and 91.1 % at the LED load was experimentally verified for red, green, and blue LEDs at maximum output current of 180 mA. For comparison, efficiencies of 38.3% for the red LED, 66.2 % for the green LED, and 64.5 % for the blue LED at the nominal current of 20mA using a 5 V supply were achieved in conventional LED drivers.

In order to project the savings in energy cost due to our efficiency improvement in LED strings in the proposed driver. A LED display with a pixel pitch of P16mm which consists of two red, one green, and one blue LEDs per pixel with a resolution of 3906 pixels per square meter [49] was considered. Assuming a single 5 V supply was used to drive each pixel of P16mm LED display panel, a total power consumption of $2 * 5 * 20 * 10^{-3} + 5 * 20 * 10^{-3} + 5 * 20 *$

$10^{-3} = 0.4$ W, and the power consumption per square meter is $3906 * 0.4 = 1.5624$ kW. Additionally, assuming an energy use cost per kWhr = $ 0.20, and the display is running 24 hours a day and 365 days a year, thus, the total energy cost for a P16mm LED display per square meter per year is $ 2,701 per year. In the case of using the proposed LED driver for driving the P16mm LED display, a total power consumption of each pixel to be $2 * 2.36 * 20 * 10^{-3} + 3.66 * 20 * 10^{-3} + 3.72 * 20 * 10^{-3} = 0.242$ W, and the power consumption per square meter is 0.945 kW. Therefore, the total energy cost for a P16mm LED display panel per square meter per year is $ 1,656 per year. Thus, by driving the LED strings of each color with their minimum drive voltage for the desired current settings, the proposed LED driver can save a total of $ 1,045 per year for each square meter, which is a significant saving in the electrical energy cost.

CHAPTER 4

A NOVEL ANALOG/DIGITAL LED DRIVER CONTROLLER

4.1 INTRODUCTION

Introduced in this chapter is a LED driver that uses an analog/digital controller to drive LED strings in parallel for LCD backlight applications. The proposed driver maintains high efficiency at the LED load by maintaining a minimum voltage across the LED strings to keep it in regulation using a digital feedback loop. In the proposed driver, phase-shifted PWM (PSPWM) dimming is implemented to reduce load current variations, improve EMI, and increase system efficiency [1],[3],[5]-[7],[15]-[16],[50]-[52]. Additionally, transformation of the digital controller to its equivalent analog controller using a mathematical model based on multirate simulation technique is demonstrated. A MATLAB SIMULINK model of the proposed driver is also shown. At the maximum rated current of 400 mA for the proposed driver, an efficiency improvement of 21 % was experimentally verified by adding the digital control loop for a two-string LED load with three white LEDs in each string [6].

Section 4.2 provides a detailed description of the proposed driver for driving multiple LED loads in parallel in backlight applications. It also explains the two different modes of operation of the digital controller in order to maintain a minimum drive voltage across the LED strings. Implementation of PSPWM dimming is described in section 4.3. It is shown that PSPWM dimming results in reduce load current variations, improved EMI, and an increase system efficiency than the conventional PWM dimming [3],[5]-[7],[15]-[16],[50]-[52]. Section 4.4 of this chapter introduces the mathematical model of the proposed two loop LED driver. A complete mathematical model of the output voltage of the LED driver with respect to reference

voltage of the analog/digital controller is developed using block diagrams. The design procedure of the power stage is introduced in section 4.5. The closed loop design and modeling of a buck converter using the Texas Instruments' TL594 PWM controller in MATLAB is also shown. The design of the digital controller and its transformation to its analog equivalent is developed in section 4.6. Section 4.7 shows the implementation of this driver in MATLAB SIMULINK along with simulation results. Finally, section 4.8 shows the experiment results of the proposed driver for driving a two-string LED load with three white LEDs in each string.

4.2 CIRCUIT DESCRIPTION

The proposed driver is shown in Fig. 4.1. It consists consisting of a dc-dc buck converter with its analog controller, a digital controller, a comparator, and two strings of white LEDs in parallel for driving LEDs in backlight applications. Current controller is used to maintain the current across the individual LED string by sensing the voltage drop across the sensing resistor and comparing it to the dimming voltage in the EA. The EA processes this error voltage to regulate the corresponding MOSFET to maintain the desired LED string current [6]-[7]. A PSPWM dimming is implemented to change the current in the LED strings to achieve dimming function. The necessary drive voltage for the LED strings is provided by the dc-dc buck converter. In order to maintain a minimum V_{DS} for the MOSFETs of the current controllers of the two LED strings, a comparator is used by comparing the inverting and non-inverting voltages of the EAs of the current controllers.

The SMPC provides the fixed drive voltage for the LED strings in order to maintain the desired current by comparing the feedback voltage from the resistive divider with a reference voltage in the EA of the analog controller. This is the maximum (worst-case) drive voltage provided by the

SMPC, and thus, yielding a low efficiency at the LED loads due to losses in the MOSFETs of the current controllers. So, it is required to change the drive voltage of the SMPC to a minimum value necessary to maintain the desired current in the LED strings. Thus, a digital loop is added to the proposed driver as shown in Fig. 4.1. The digital loop is implemented using a PIC18F4431 microcontroller which injects voltage at the feedback summing node of the SMPC in order to maintain minimum drive voltage for the LED strings. The digital controller achieves this by sensing the outputs of the comparator, drain voltages, and reference voltages of the current controllers.

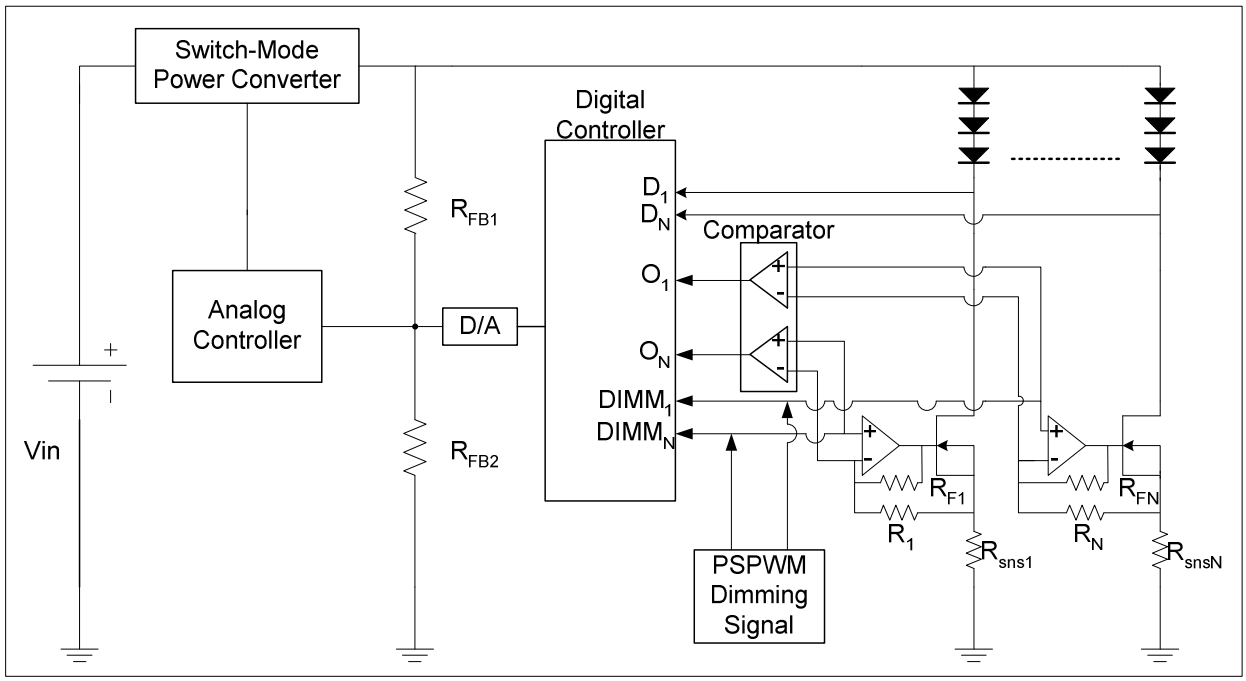


Figure 4.1 Proposed LED driver.

The analog loop in the proposed driver is operating at a higher frequency than the digital loop due to slow change of LED forward voltage and the operating frequency, 100-400 Hz, of PSPWM dimming [6]-[15]. Since the output of the digital controller (G_{C2}) is a PWM signal, it needs to be converted into its equivalent analog voltage before being fed at the feedback

summing node. Thus, by adding the extra digital loop, the proposed driver can significantly increase efficiency at the LED load.

Therefore, in order to reduce losses in MOSFETs of the current controllers and increase efficiency at the LED load, the second loop is added to maintain the MOSFETs at minimum V_{DS} required to maintain the desired LED string current [3]-[4],[8]. The second loop has two modes of operation: efficiency optimization and operation modes [3]-[4],[8]. For example, if a single 13 V power supply was used to drive a string of white LED arrays; it would yield an efficiency of 68 %. The efficiency optimization mode involves finding the minimum V_{DS} indirectly to maintain the desired LED string current by increasing the output of the second controller until the sensed maximum output of the comparators is greater than the threshold value [3]. Fig. 4.2 shows the flowchart of this mode. In the operation mode, the control algorithms are processed to generate the corresponding PWM signal from the second controller for the desired LED string current, to maintain minimum drive voltage for the LED strings. Fig. 3.3 shows the flowchart of this mode.

EFFICIENCY OPTIMIZATION MODE:

1) Initially, the closed-loop SMPC provides the worst-case drive voltage to the LED load at the desired current settings. This voltage is high enough to maintain the MOSFETs of the current controllers in the saturation region leading to undesired power dissipation in the current controllers [3]-[4],[8].
2) Once the output drive voltage of the SMPC reaches zero steady state error, the digital controller starts to sense the output of the comparators to determine the minimum drive voltage required for the LED strings at the desired current settings [3]-[4],[8].

3) Maximum voltage of the sensed output of the comparators are then determined. This voltage is compared to a threshold value. The threshold value is the maximum voltage required to maintain a minimum V_{DS} across the current controllers of the LED strings for the set current. The duty cycle of the PWM signal of the second controller is increased, if the sensed maximum output voltage of the comparators is below the threshold value. This step continues until the sensed maximum output voltage of the comparators is greater than the threshold value, indicating that the LED strings are no longer able to maintain the set current [3]-[4],[8].

4) At this moment, the second controller reduces its duty cycle of the PWM signal to maintain regulation in the LED strings and exits the efficiency optimization mode. The second controller enters the operation mode at this time [3]-[4],[8].

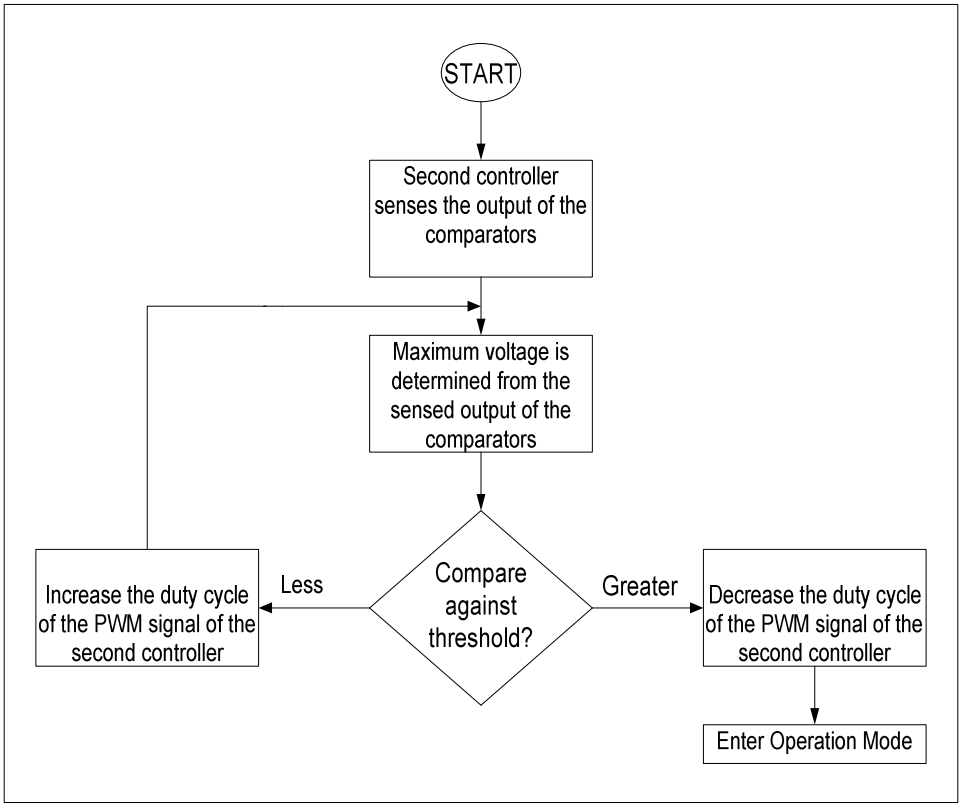


Figure 4.2 Flowchart of efficiency optimization mode.

OPERATION MODE:

1) The second controller determines the minimum drain voltage of the MOSFETs of the current controllers from its sensed value and stores this in its memory as the reference voltage of the controller at the beginning of this mode. Additionally, at the beginning of this mode, the dimming signals of the error amplifiers are also sensed and maximum dimming signal is determined and stored in memory [3]-[4],[8].
2) The stored reference voltage of the second controller is compared against the sensed minimum drain voltage of the MOSFETs of the current controllers to generate the error voltage for the controller, i.e., $V_{ERROR} = V_{REF} - V_{MIN(D_1,D_2)}$. This error voltage is processed by the PID controller to generate the corresponding duty cycle of the PWM signal for the second controller [3]-[4],[8].
3) Since the first controller is analog, the output of the second controller needs to be filtered by a low-pass active filter to generate the equivalent dc signal before being applied at the feedback summing node [3].
4) Since V_{DS} of the MOSFETs of the current controllers increases when the current in the LED string reduces, thus if the dimming signals of the error amplifiers changes, the second controller exits the operation mode, and enters the efficiency optimization mode to find the minimum V_{DS} for the new current settings [3]-[4],[8].

4.3 PSPWM DIMMING

There are two primary types of dimming schemes in the LED industry, Analog or PWM dimming. The major advantages and disadvantages of both types of dimming are clearly explained in section 3.3. Typically, in applications requiring high resolutions such as in display

panels, PSPWM dimming is implemented. PSPWM dimming frequency is between 100-400 Hz for a LED driver used in LCD backlight [7].

Fig. 4.3 (a) shows the implementation of a conventional PWM dimming in LED drivers driving two LED strings in parallel. It can be observed from the waveform that the output current (I_O) is its peak is equal to N times one LED string current (I_{LED}) during the ON period and zero during the OFF period, where N is the number of LED strings [7],[15]-[16],[50]-[52]. Thus, conventional PWM dimming suffers from large load variations, higher output voltage ripple, lower system efficiency, and EMI issues [3],[5],[7],[15]-[16],[50]-[52].

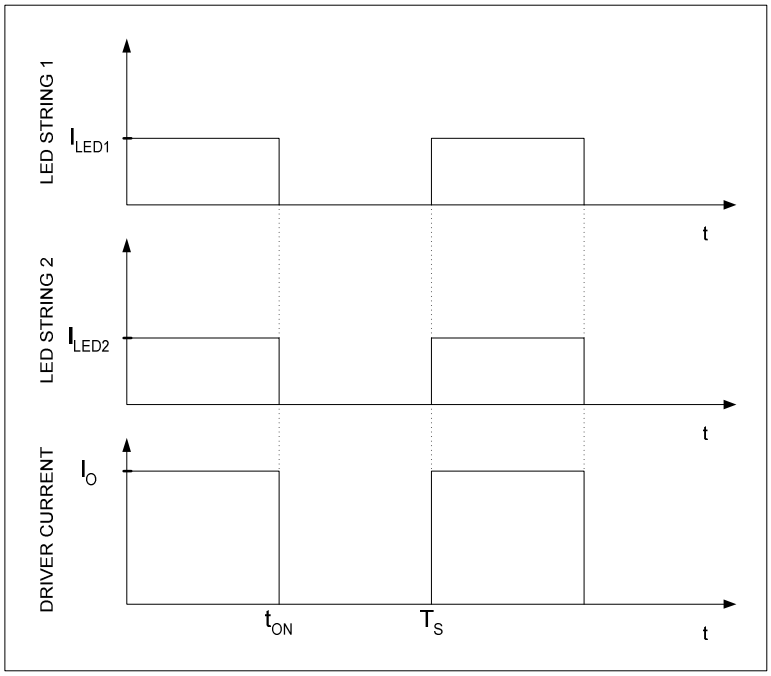
In order to address with the problems associated with the conventional PWM dimming, a PSPWM dimming as described in [1],[3],[5],[7],[15]-[16],[50]-[52] is implemented in the proposed driver. Fig. 4.3 (b) shows the PSPWM dimming implemented in a LED driver with two LED strings. It can be deduced from the waveform that the output current (I_O) is its peak to peak difference is less than or equal to the current in a single LED string (I_{LED}) [16],[52]. This type of dimming is very similar to PWM dimming except the current is drawn sequentially with a certain amount of phase delay [7]. Major advantages of this PSPWM dimming scheme are reduced output current and voltage ripple leading to reduction in sizes of the components of output filter of the driver, improve transient response, linear change of LED color, higher color resolution, better EMI performance, and improve system efficiency [1], [3],[5],[7],[15]-[16],[50]-[52].

The phase delay in the PSPWM dimming is calculated using

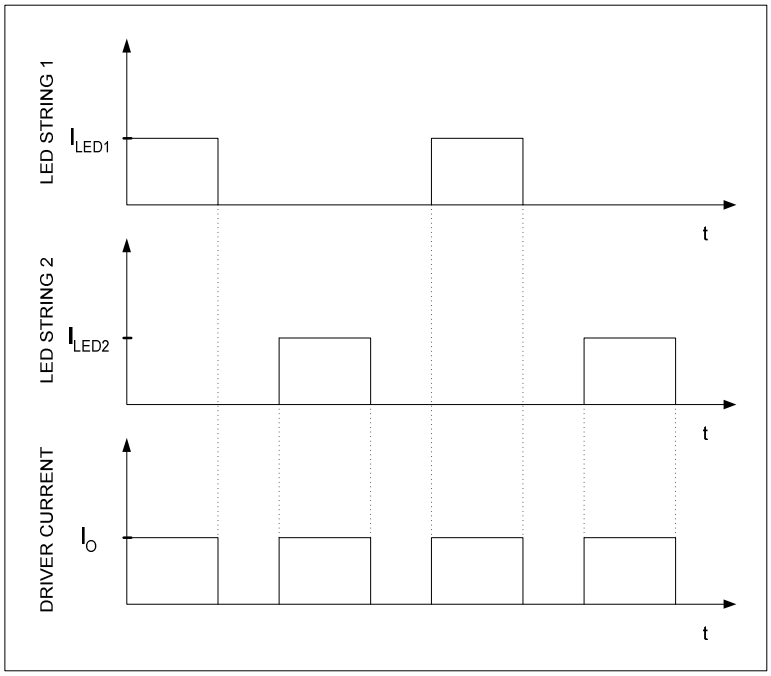
$$\theta = \frac{360^{o} * (k-1)}{N} \tag{4.1}$$

where k is the selected LED string and N is the total number of LED strings in parallel in the

driver, assuming a complete period is 360°, and the phase shift for LED string 1 is 0° [3],[50]. Therefore, the delay between the dimming signal of LED string 1 and LED string 2 is 180° for this proposed implementation [3],[50]. Thus, the average LED string current is $I_{LED,AVG} = \frac{\theta^\circ * I_{LED}}{360^\circ}$ (4.2) [50]. Fig. 4.4 shows the PSPWM dimming method implemented in the proposed driver for 20 %, 50 %, and 80 % duty cycles. The upper waveform (Channel 1) is the PSPWM dimming signal for LED string 1, the middle waveform (Channel 2) is the PSPWM dimming signal for LED string 2, and the lower waveform (Channel 4) is the output current of the LED driver for 20 %, 50 %, and 80 %, respectively. As can be noticed from the figure, that the phase delay between the consecutive two signals is 180°. Only one LED string is ON at any time, when the duty cycle of the PWM dimming signal is less than 1/N, and overlaps occurs between the PWM dimming signal when the duty cycle is greater than 1/N.

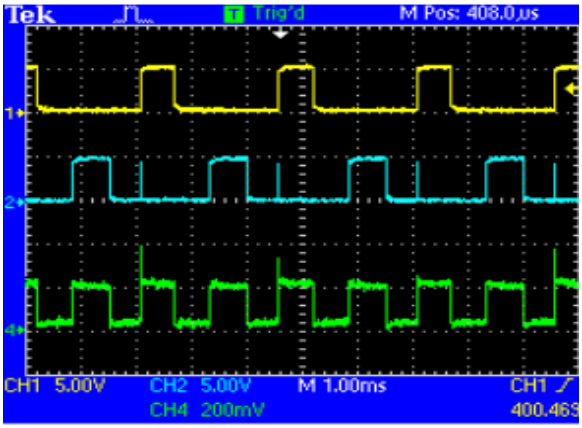


(a)

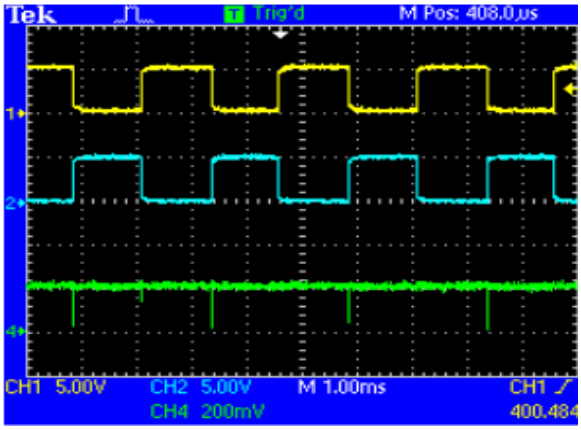


(b)

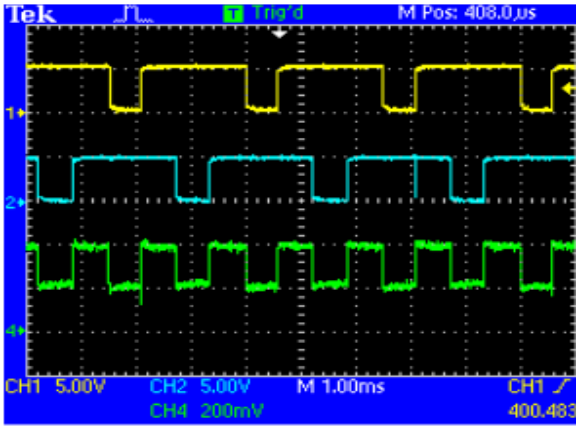
Figure 4.3 (a) Conventional PWM Dimming, (b) PSPWM dimming.



(a)



(b)



(c)

Figure 4.4 PSPWM dimming signal and output current of the driver for duty cycles of (a) 20 %, (b) 50 %, and (c) 80 %.

In this proposed driver for driving two-string LED load with three white LEDs in each string for backlight applications, individual dimming of each string is required. In this implementation, two different PSPWM dimming control signals are required to be generated using Microchip's PIC18F4431 microcontroller which is 180° out of phase with each other. The dimming signals are generated externally and are fed to the non-inverting input of the error amplifiers of the current controllers.

4.4 MATHEMATICAL MODEL

4.4.1 CONTROL OF CONSTANT-CURRENT LOADS

Constant-current loads like LEDs are driven by SMPC which are controlled by either an analog or digital controller as found in literature [4]-[8],[15]-[17],[37],[50]-[52]. In such implementations, feedback voltage of the controller would be compared against a reference voltage to generate the error voltage. This error voltage is processed by a proportional, proportional-integral, or PID controller to generate the corresponding duty cycles of the PWM signal for the switches of the SMPC. However in this proposed implementation, which uses an analog/digital controller to control the LED loads, requires development of a model that would merge the analog and digital domains. The analog controller provides the maximum output voltage of the SMPC. The digital controller is added to the system to minimize V_{DS} of the MOSFETs of the current controllers in order to increase efficiency at the LED load. Therefore, the digital controller is first designed in the z-domain and then transformed into its equivalent s-domain according to [53]-[54].

4.4.2 MODEL OF ANALOG/DIGITAL CONTROLLED LED DRIVER

The second loop of the proposed LED driver is in the digital domain as shown in Fig. 4.5. Since digital controllers have higher bandwidth, reduced parts count, and are not susceptible to temperature and aging of circuit components [12],[35]-[38]. Digital controllers are implemented using either a microcontroller or a DSP. In this implementation, a Microchip's PIC18F4431 microcontroller is used as the controller for the digital loop. In Fig. 4.5, H_1 and H_2 are the feedback gains, and V_{REF1} and V_{REF2} are the references, of the first controller ($G_{C1}(s)$) and the second controller ($G_{C2}(z)$), respectively. The LEDs, current controllers, and sensing resistors are

assumed to be purely resistive loads, and this approximation does not affect the dynamics characteristics of the LED driver system. G_{ADC} and G_{DAC} are the transfer functions of the A/D and D/A converters for $G_{C2}(z)$.

Since there are two references for the system (V_{REF1} and V_{REF2}), the superposition principle as explained in [55] is implemented in Fig. 4.5, to obtain the relationship of the overall output voltage ($V_O(s)$) of the system. As explained in section 3.3, SMPC are non-linear systems and so are the LED loads. Since superposition principle could only be implemented on linear systems, the linearization technique shown in Fig. 3.6 (b) is used for linearizing SMPC and the LED loads by assuming they are resistive loads. Additionally, the slow changing nature of LED forward voltage means the LED loads are operating at a frequency of 400 Hz. However, the analog loop is operating at a frequency of 200 kHz, therefore the driver output changes only when the output of $G_{C2}(z)$ changes. So, the overall system can be assumed to be a linear system and superposition principle can be applied.

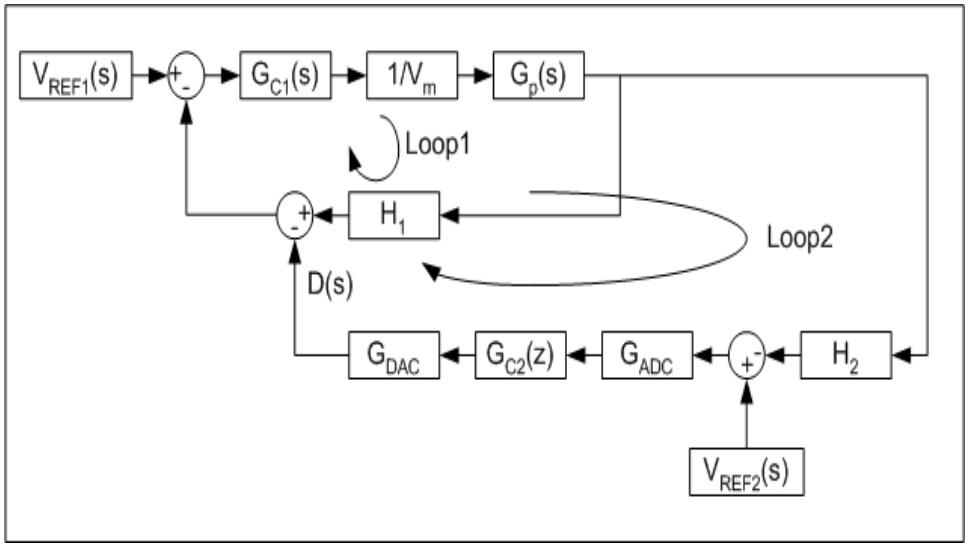


Figure 4.5 Mathematical model of the proposed LED driver.

Using the superposition principle, setting the signal coming from the second loop (D(s)) to zero, the block diagram in Fig. 4.5 reduces to the block diagram shown in Fig. 4.6. Thus, from Fig. 4.6, $V_{O1}(s)$ is defined as

$$V_{O1}(s) = \frac{\frac{G_{C1}(s)G_P(s)}{V_M}}{1+\frac{H_1 G_{C1}(s)G_P(s)}{V_M}} V_{REF1}(s) \tag{4.3}$$

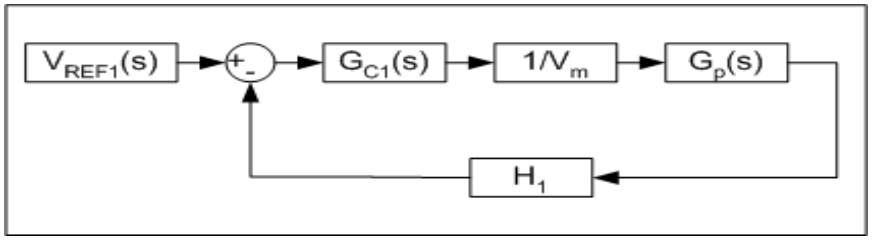


Figure 4.6 Block diagram when D(s) reduces to zero.

Similarly, setting $V_{REF1}(s)$ to zero as shown in Fig. 4.5, the block diagram reduces to Fig. 4.7. Thus, from Fig. 4.7, the relationship of $V_{O2}(s)$ can be obtained as,

$$V_{O2}(s) = \frac{\frac{G_{C1}(s)G_P(s)G_{C2}(z)G_{ADC}G_{DAC}}{V_M}}{1+\frac{H_2 G_{C1}(s)G_P(s)G_{C2}(z)G_{ADC}G_{DAC}}{V_M}} V_{REF2}(s) \tag{4.4}$$

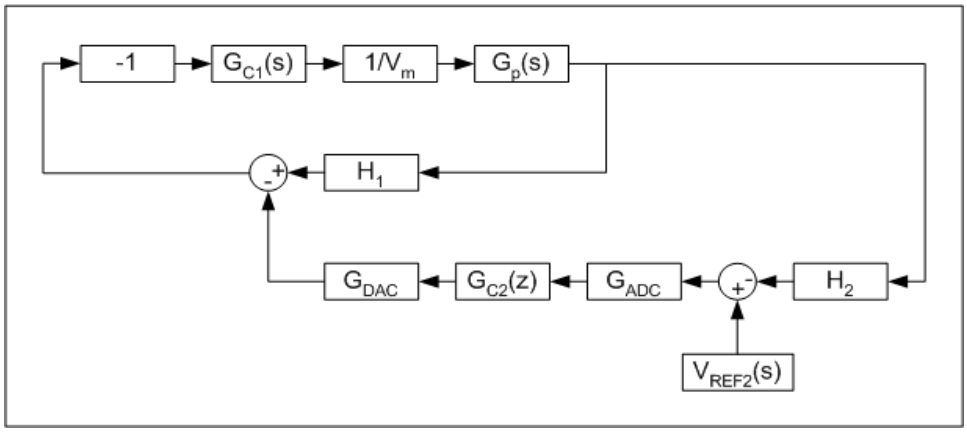


Figure 4.7 Block diagram when $V_{REF1}(s)$ is zero.

Evaluating equations (4.3) and (4.4), and knowing $V_O(s) = V_{O1}(s) + V_{O2}(s)$ (4.5), the relationship of $V_O(s)$ is obtained as,

$$V_O(s) = \frac{\frac{G_{C1}(s)G_P(s)}{V_M}}{1+\frac{H_1 G_{C1}(s)G_P(s)}{V_M}} V_{REF1}(s) + \frac{\frac{G_{C1}(s)G_P(s)G_{C2}(z)G_{ADC}G_{DAC}}{V_M}}{1+\frac{H_2 G_{C1}(s)G_P(s)G_{C2}(z)G_{ADC}G_{DAC}}{V_M}} V_{REF2}(s) \qquad (4.6)$$

4.5 DESIGN OF DC-DC POWER STAGE

A DC-DC buck converter as explained in section 2.2.2 is required to provide the drive voltage for the two-string white LED loads. The buck converter was chosen for this application due to the availability of the input voltage and number of LED strings driven. In this proposed driver, the buck converter is controlled by Texas Instruments' TL594 PWM controller [56]-[57].

The multi-string LED loads with its individual current controllers and sensing resistors are modeled as resistive loads. As explained in section 4.4.2, the dynamic behavior of the buck converter can be developed using Vorpèrian averaged-switch model, which replaces the non-linear components of the converter with its simple equivalent circuit model as shown in Fig. 3.6 (b), and Fig. 4.6 [9],[30]. Therefore, the control-to-output transfer function is similar to equation (3.1) [30]-[34]. This transfer function is used in MATLAB to design the appropriate analog controller for the DC-DC buck converter.

$$\frac{\widehat{v_O}}{\hat{d}} = \left(\frac{V_O}{D}\right)\left[\frac{1+sR_C C}{1+s\left(R_C C+(R_{LOAD}//R_L)C+\frac{L}{R_{LOAD}+R_L}\right)+s^2 LC\left(\frac{R_{LOAD}+R_C}{R_{LOAD}+R_L}\right)}\right] \qquad (4.7)$$

4.5.1 PARAMETER CALCULATIONS

The buck converter output filter storage elements are selected to satisfy the allowed ripple magnitudes of the inductor current and capacitor voltage [1],[3],[16]. The inductor value

is found using equations (3.4) [9],

$$L = \frac{DV_{IN}(1-D)}{f_S \Delta I} = 150\mu H \tag{4.8}$$

The value of the capacitor is dependent on the allowed output ripple voltage, since the capacitor of the output filter in a buck converter is connected directly across the load. Therefore, the output capacitor is found using equation (3.5) [9],

$$C = \frac{V_{IN} D(1-D)}{8 f_S^2 L \Delta v_C} = 100\mu F \tag{4.9}$$

The performance of the buck converter for driving two string of white LED load was experimentally verified using a Texas Instruments TL594 controller with a switching frequency of 200 kHz [4],[7]-[8]. Each string of LEDs is rated at 200 mA, and thus, buck converter needs to deliver a total output current of 400 mA. The key components and parameters for the design of buck driver are shown in Table 4.1 (a) and (b).

Table 4.1 – (a) Key parameter list, and (b) Key component list of the DC-DC power stage.

(a)

Parameter	Value
Input Voltage (Vin)	24 V
Output Voltage (Vo)	13 V
Output Current (Io)	400 mA
Frequency	200 kHz
Parasitic	RC=150 mΩ [39], RL=370 mΩ [58]
Ripple Inductor Current (ΔI)	0.2 A
Ripple Voltage (ΔV)	1.2 mV

(b)

Components	Value
Input Capacitor	10 µF, 25 V
MOSFET	IRF9630
Diode	MUR420
Output Inductor	75 µH, 2.5 A
Output Capacitor	100 µF, 35 V
MOSFETs of the Current Controllers	Q2N7000
Error Amplifiers of the Current Controllers	TLC272
PWM Controller	TL594

4.5.2 DESIGN OF CLOSED LOOP VOLTAGE-MODE FEEBACK

In this proposed driver, a Texas Instruments' TL594 PWM controller is used to control the output voltage of the buck converter. The PWM controller senses the feedback voltage from the resistive divider network and compares with the reference voltage of the PWM controller to regulate the maximum (worst-case) drive voltage of the LED strings. The drawback of using a single controller to provide the drive voltage of the LED strings is that the MOSFETs of the current controllers are operating in the saturation region, and thus, leading to unwanted power losses in the LED strings. In this approach, a type III compensation (PID) network consisting of two zeros and three poles as shown in Fig. 2.10 is implemented in this system [59]-[60]. A block diagram of voltage-mode feedback for this proposed driver is shown in Fig. 4.6 when D(s) is set to zero. The transfer function of this type of compensation network is,

$$G_C(s) = \frac{R_1+R_2}{R_1R_2C_2}\frac{\left(s+\frac{1}{R_3C_3}\right)\left(s+\frac{1}{(R_1+R_2)C_1}\right)}{s\left(s+\frac{C_2+C_3}{R_3C_2C_3}\right)\left(s+\frac{1}{R_2C_1}\right)} \tag{4.10}$$

The design of the voltage mode feedback controller for the buck converter is based on equation

(4.7) and on the parameters described in Table (4.1). The load of the buck converter is assumed to be resistive (R_{LOAD}), i.e.,

$$V_{OUT} = I_{OUT} * R_{OUT} \tag{4.11}$$

In this implementation, R_{LOAD} is equivalent to the parallel combination of two LED strings. Therefore, the effective resistance of each string, R_{SINGLE_STRING}, is the series combination of equivalent DC resistance of the LED, i.e., $R_{EQU} = V_{FWD}/I_{LED\_STRING}$(4.12), the on-resistance of the MOSFETs, and the sensing resistor [4],[41]. For this case, since there are three LEDs per string and the typical forward voltage of the chosen LED is 3.5 V, thus the equivalent DC resistance of the LED strings is [3],[11],

$$R_{EQU} = \frac{3.5*3}{0.200} = 52.5\,\Omega \tag{4.13}$$

$$R_{SINGLE_{STRING}} = R_{EQU} + R_{MOSFET} + R_{SENSE} = 52.5 + 1.8 + 0.68 = 54.98\,\Omega \tag{4.14}$$

$$R_{LOAD} = R_{SINGLE\_STRING_1} // R_{SINGLE\_STRING_2} = 27.49\,\Omega \tag{4.15}$$

Substituting the values of R_{LOAD} from equation (4.15) and the parameter values from Table (4.1), the transfer function for the buck driver is,

$$G_P(s) = \frac{2.4*10^{-5}s+24}{7.495*10^{-9}s^2+6.343*10^{-5}s+1} \tag{4.16}$$

Using MATLAB, the open loop Bode plot of the above transfer function is shown in Fig. 4.8. The transfer function has two conjugate poles at – (4.23±j10.75)*10^3 radian/sec , and a left half plane zero at -1*10^6 radian/sec. Based on these results, a suitable type III compensation (PID) network is designed in MATLAB to meet the requirement for gain, crossover frequency and phase margins.

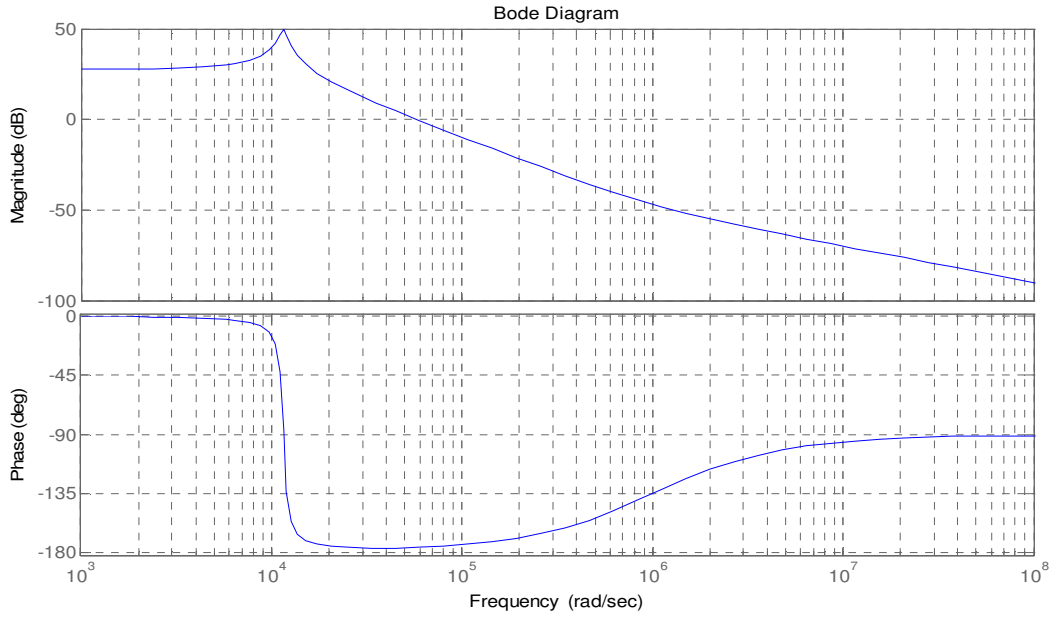


Figure 4.8 Open loop Bode plot of the buck driver.

Using MATLAB, the PID controller is designed with poles at the origin, and at –(9.79±j11.74)*10^4 and zeros at -968 radian/sec, and at -6.83*10^3 radian/sec, respectively. The closed loop Bode plot of the driver is shown in Fig. 4.9 and the model used for Bode plot is shown in Appendix B. From Fig. 4.9, it can be seen that the system bandwidth is about 32 kHz, and the phase margin is about 35°. The transfer function of the designed PID controller using equation (4.10) is,

$$G_C(s) = 3.26 * 10^6 \frac{s^2+7.81*10^3 s+6.62*10^6}{s^3+1.96*10^5 s^2+9.58*10^9 s} \tag{4.17}$$

From equation (4.17), we can deduce that R_1=20 kΩ, R_2=1.5 kΩ, R_3=47 kΩ, C_1=6800 pF, C_2=0.22 nF, and C_3=0.022 µF for equation (4.10).

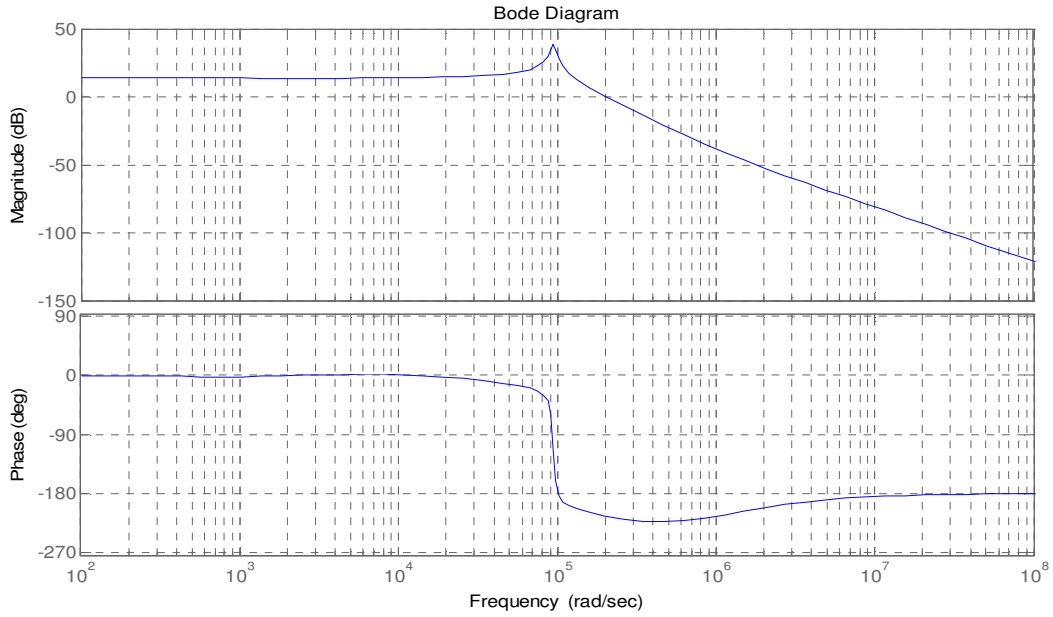


Figure 4.9 Closed loop Bode plot of the buck driver.

4.5.3 SIMULATION RESULTS

The performance characteristics of the buck LED driver with only the analog control loop was verified using MATLAB SIMULINK during both steady state and transient operation [1],[6]. State space equations are used to linearize the buck converter as shown in Fig. 3.11 and in equations (3.25), (3.26) and (3.27) [45],[46]. The complete model implemented in MATLAB SIMULINK is shown in Fig. 4.10. LED loads are modeled as resistive loads as shown in equation (4.15). The analog controller is modeled according to equation (4.17), and the comparator is used to compare the output of the controller with a sawtooth signal to generate the proper PWM signals for the buck converter [45]-[48]. The model was simulated using ODE3 (Bogacki-Shampine) differential equations. A fixed step size of $0.1*10^{-6}$ seconds was used to obtain accurate results.

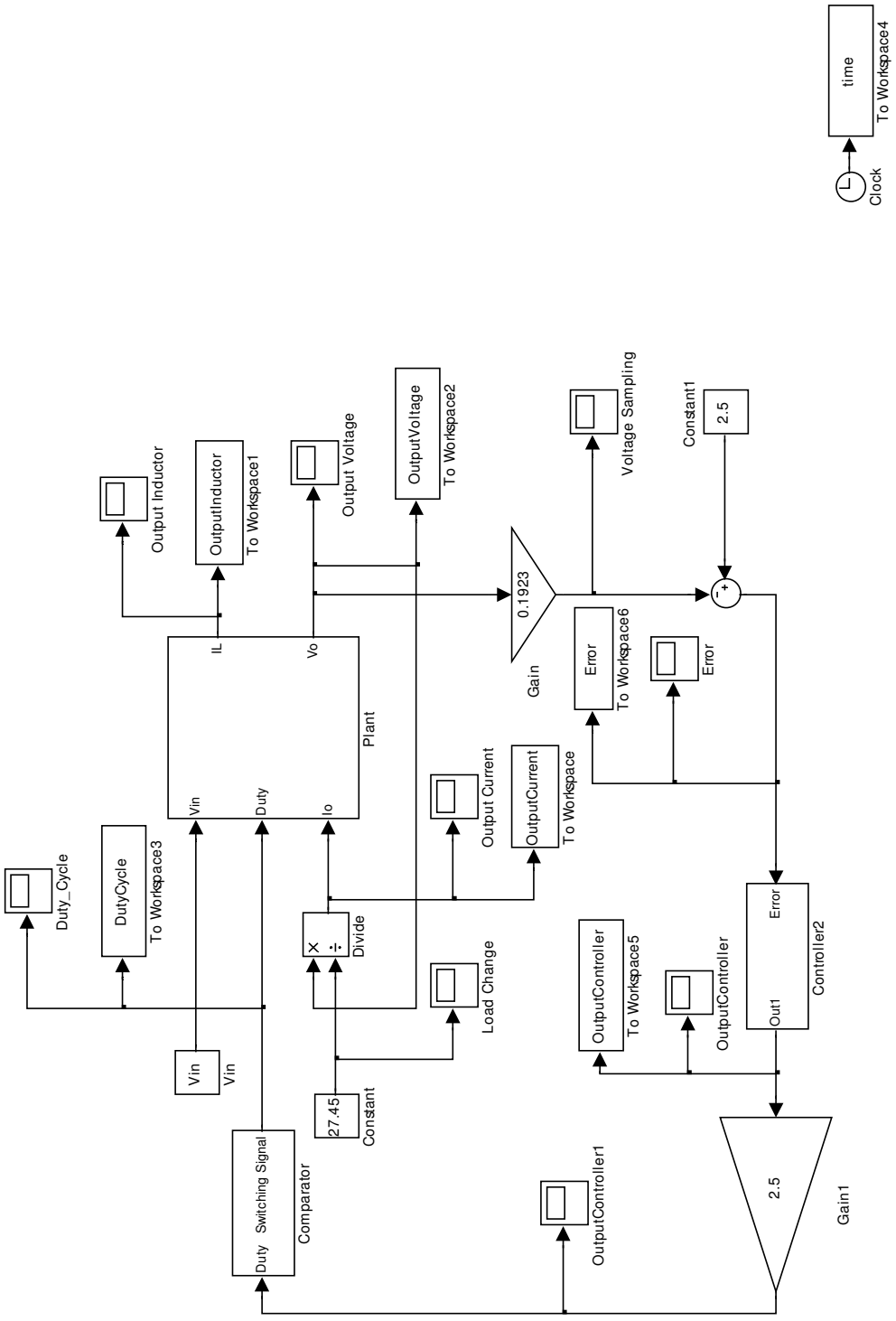
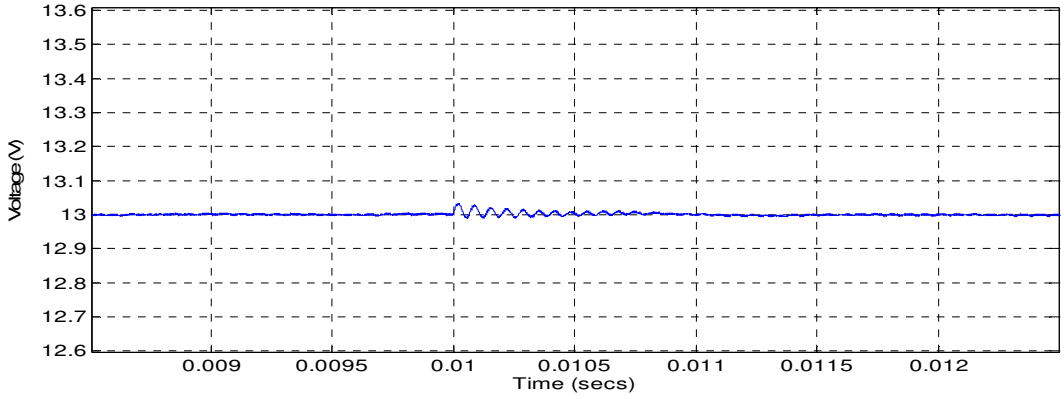
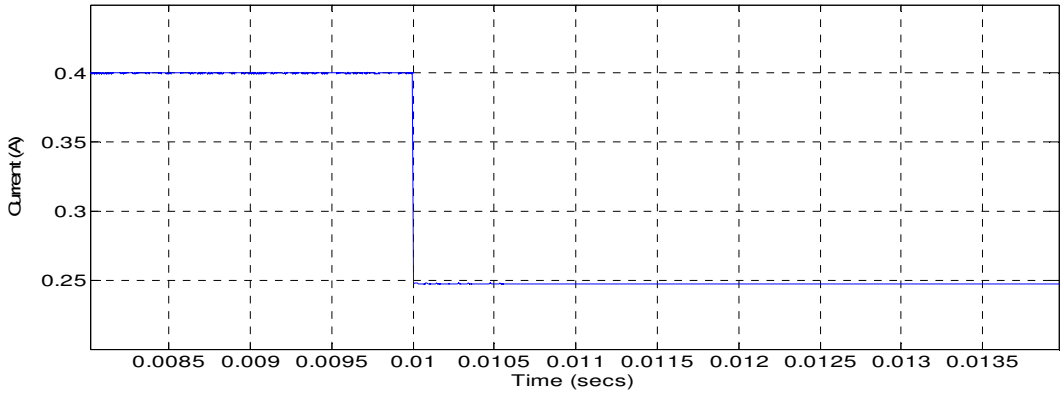


Figure 4.10 Buck driver model implemented in MATLAB SIMULINK.

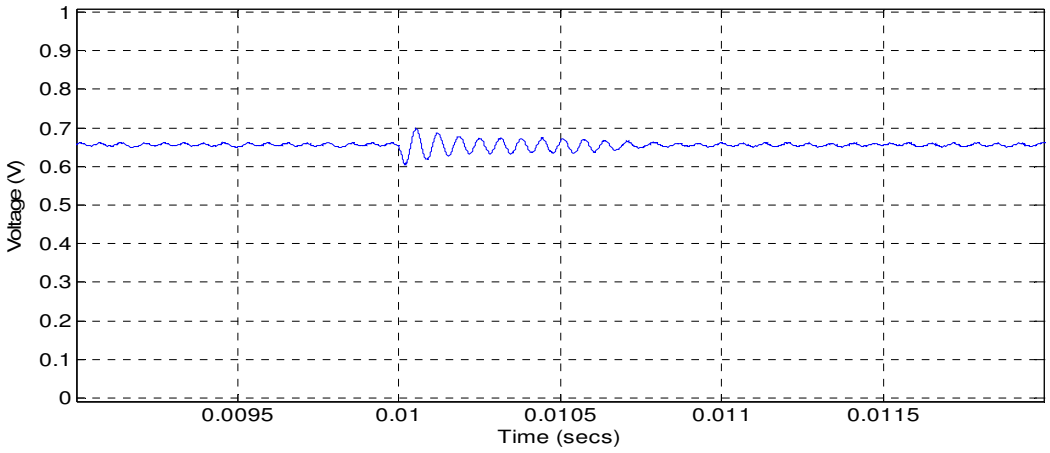
The simulation result of the output voltage for the buck driver for a load transient from 400 mA to 248 mA at 10 ms is shown in Fig. 4.11 (a). The ripple output voltage is ± 3 mV which is above the specified limit. Fig. 4.11 (b) shows that the output current reaches the new steady state value within 10µs after transient, giving a refresh rate of the LED strings greater than 1000 Hz. The ripple output current is ± 2 mA which is below the specified limit. Fig. 4.11 (c) shows the output of the analog controller is maintained at 0.66 V during transient in order to maintain the output voltage of the buck driver in regulation. Fig. 4.11 (d) shows the inductor current indicating that the buck converter is operating in continuous conduction mode.



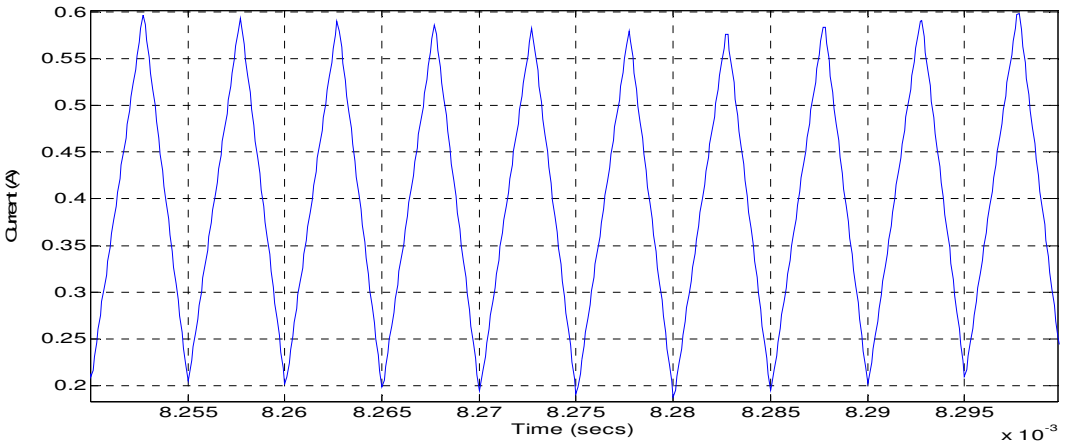
(a)



(b)



(c)



(d)

Figure 4.11 (a) Output voltage, (b) Output Current, (c) Output of the analog controller, and (d) Inductor Current of the buck driver.

4.6 DESIGN OF CONTROLLER FOR THE SECOND LOOP

In the proposed driver, the second controller is designed in the digital domain due to its higher bandwidth, reduced parts count, and are not susceptible to temperature and aging of circuit components [12],[35]-[38]. Thus, in this implementation, a PIC18F4431 is used as the controller

for the second loop. The microcontroller senses the minimum drain voltage of the MOSFETs in the current controllers and compares it with the reference voltage as discussed in section 4.2. This is done to regulate the drive voltage of the LED strings to its minimum. This causes the MOSFETs of the current controllers to operate with minimum V_{DS}, and thus, yielding a high efficiency at the LED loads. In this approach, a digital redesign approach using frequency response technique is implemented. In this digital redesign approach, the controller is initially designed in the s-domain, and then, digitized into its equivalent z-domain.

The block diagram of the complete LED driver system with both the controllers being in s-domain is shown in Fig. 4.12. Using MATLAB, the bode plot of the drive voltage with respect to $V_{REF1}(s)$ is given by equation (4.3) as shown in Fig. 4.13 (a) and the model used for Bode plot is shown in Appendix B, when $G_{C2}(s)$ is equal to 1. As can be found from Fig. 4.13 (a), the crossover frequency is 24.5 kHz and the phase margin is 28°. Similarly, using MATLAB, the bode plot of the drive voltage with respect to $V_{REF2}(s)$ given by equation (4.4) as shown in Fig. 4.13 (b), when $G_{C2}(s)$ is equal to 1. As can be found from Fig. 4.13 (b), the crossover frequency is 24.5 kHz and the phase margin is 28°. Finally, using MATLAB, the bode plot of the drive voltage with respect to both the references ($V_{REF1}(s)$ and $V_{REF2}(s)$) is given by equation (4.5) as shown in Fig. 4.13 (c), when $G_{C2}(s)$ is equal to 1. As can be found from Fig. 4.13 (c), the crossover frequency is 30 kHz and the phase margin is 33°, so the LED driver system is stable.

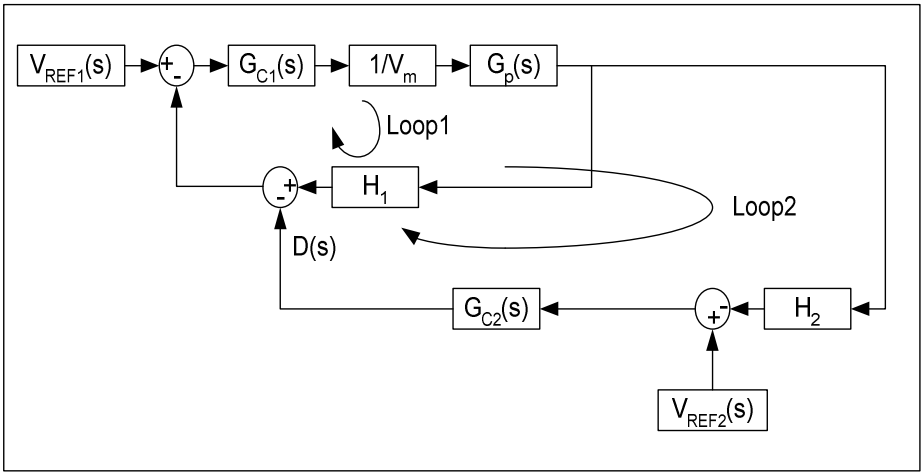
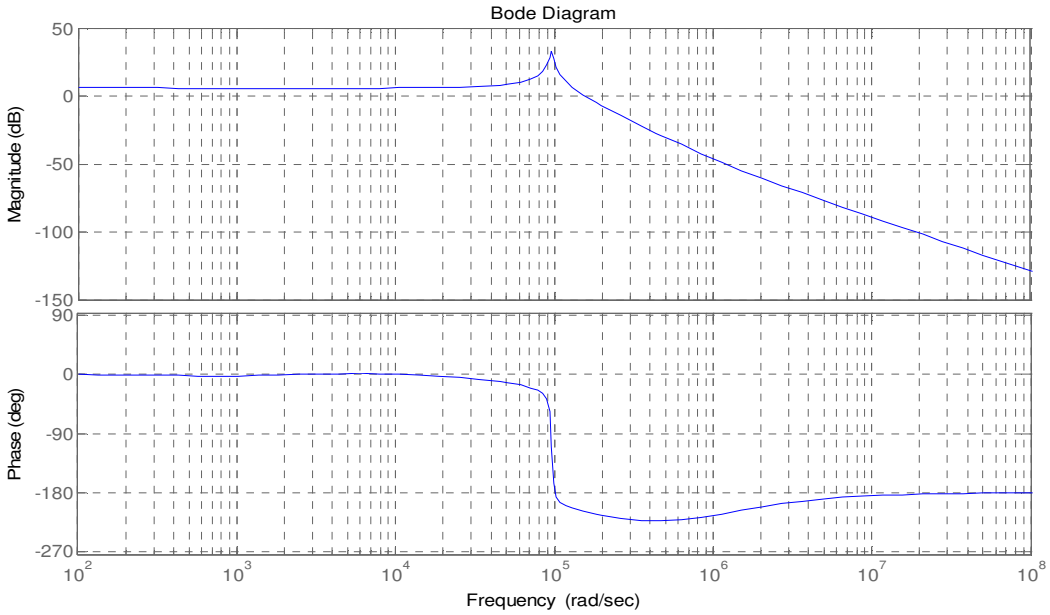
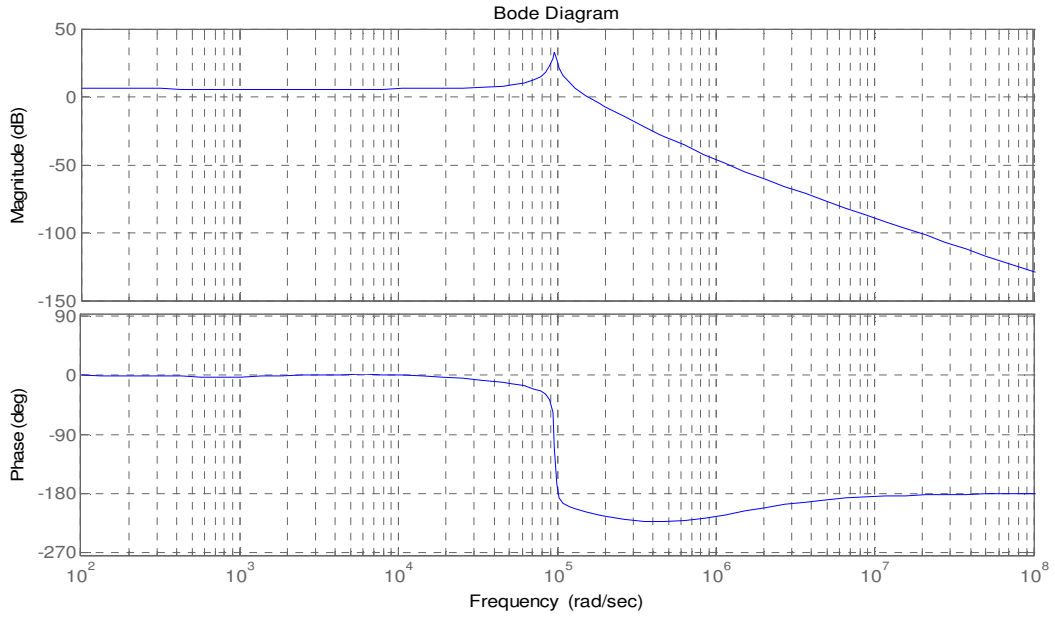


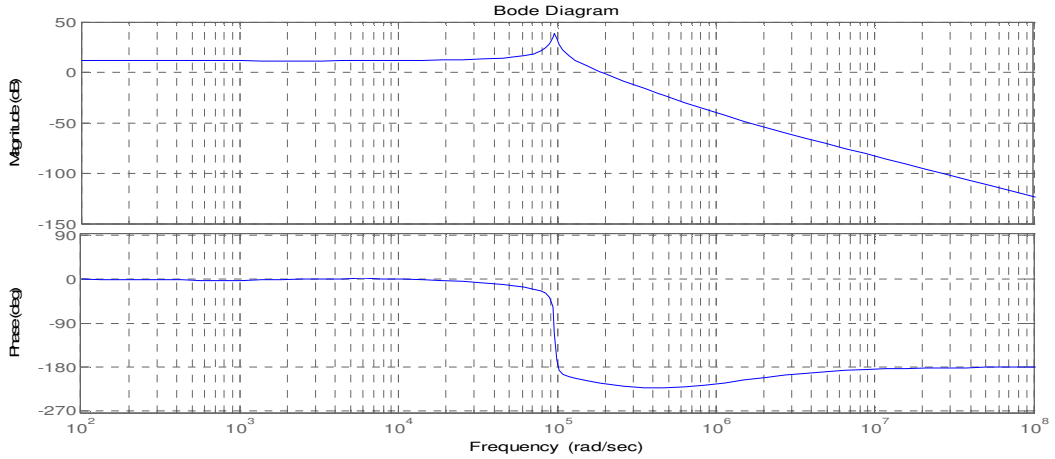
Figure 4.12 Mathematical model of the proposed LED driver when both the controllers are in s domain.



(a)



(b)



(c)

Figure 4.13 Bode plot of the (a) drive voltage with respect to $V_{REF1}(s)$, (b) drive voltage with respect to the $V_{REF2}(s)$, and (c) drive voltage with respect to both the references – when $G_{C2}(s)$ is equal to 1.

An analog PID controller given by equations (3.12) and (3.13) was designed in MATLAB to satisfy the requirements for gain, crossover frequency and phase margins. The transfer function of the designed PID controller is,

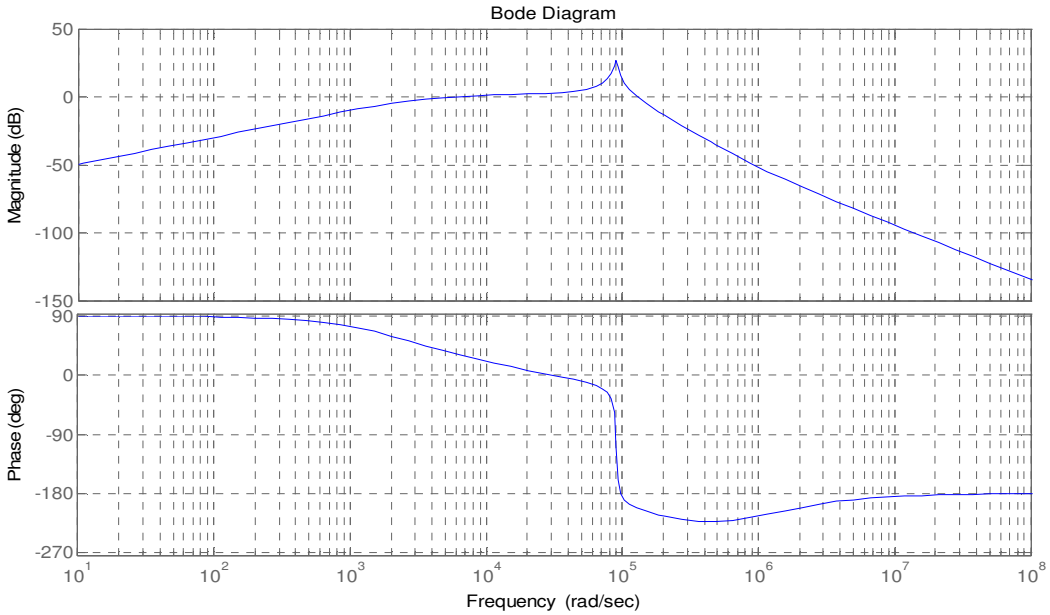
$$G_C(s) = 2 + \frac{1*10^4}{s} + 1.1881 * 10^{-4}s \tag{4.18}$$

The PID controller was designed with a pole in the origin, and two zeros at $-(8.42 \pm j3.65)*10^3$ radian/secs. Using MATLAB, the bode plot of the drive voltage with respect to $V_{REF1}(s)$ is given by equation (4.3) as shown in Fig. 4.14 (a), when $G_{C2}(s)$ is equal to equation (4.18). As can be found from Fig. 4.14 (a), the crossover frequency is 20.5 kHz and the phase margin is 22°. Similarly, using MATLAB, the bode plot of the drive voltage with respect to $V_{REF2}(s)$ given by equation (4.4) as shown in Fig. 4.14 (b), when $G_{C2}(s)$ is equal to equation (4.18). As can be found from Fig. 4.14 (b), the crossover frequency is 25 kHz and the phase margin is 29°. Finally, using MATLAB, the bode plot of the drive voltage with respect to both the references ($V_{REF1}(s)$ and $V_{REF2}(s)$) is given by equation (4.5) and shown in Fig. 4.14 (c), when $G_{C2}(s)$ is equal to equation (4.18). As can be found from Fig. 4.14 (c), the crossover frequency is 28 kHz and the phase margin is 32°, so the LED driver system is stable.

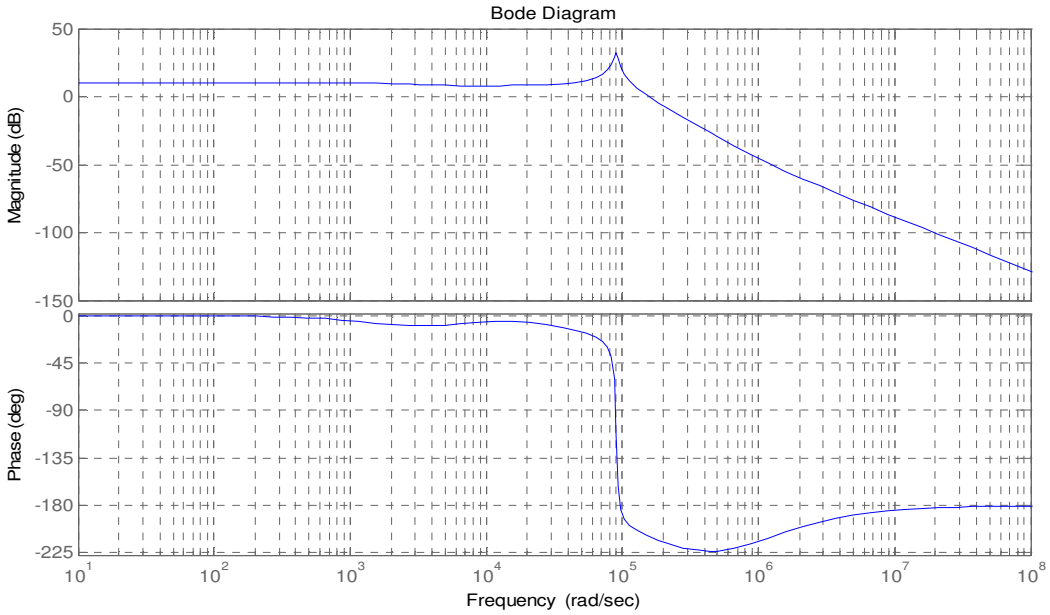
The second controller is implemented using a microcontroller, thus equation (4.18) needs to be transformed into its digital equivalent using the matched pole-zero method with a sampling frequency of 20 µs. Thus, using MATLAB command 'c2d' $G_C(z)$ becomes,

$$G_C(z) = \frac{7.012z^2 - 11.82z + 5.008}{z - 1} \tag{4.19}$$

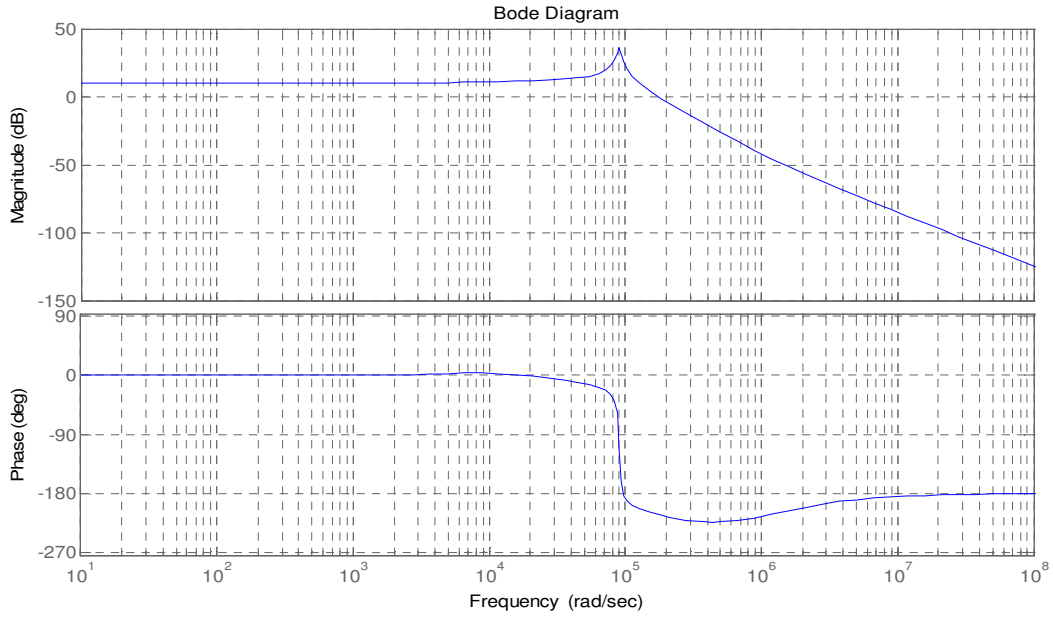
$$u(k) = 7.012e(k) - 11.82\sum_{i=0}^{k} e[i] + 5.008\{e[k] - e[k-1]\} \tag{4.20}$$


Bode Diagram
Magnitude (dB)
50
0
-50
-100
-150
Phase (deg)
90
0
-90
-180
-270
10^1
10^2
10^3
10^4
10^5
10^6
10^7
10^8
Frequency (rad/sec)

(a)


Bode Diagram
Magnitude (dB)
50
0
-50
-100
-150
Phase (deg)
0
-45
-90
-135
-180
-225
10^1
10^2
10^3
10^4
10^5
10^6
10^7
10^8
Frequency (rad/sec)

(b)



(c)

Figure 4.14 Bode plot of the (a) drive voltage with respect to $V_{REF1}(s)$, (b) drive voltage with respect to the $V_{REF2}(s)$, and (c) drive voltage with respect to both the references – when $G_{C2}(s)$ is equal to equation (4.18).

4.6.1 TRANSFORMATION OF DIGITAL TO ANALOG

The first loop and second loop of the proposed driver system is in s-domain and z-domain respectively as seen in equation (4.6). Thus, the second loop needs to be transformed into s-domain as shown in Fig. 4.12. Therefore, equation (4.19) needs to transformed into its equivalent s-domain according to [53]-[54], using multirate simulation technique [53]. An analog signal when sampled into its equivalent digital signal losses some information [53]. So, by satisfying the condition that "the damped natural frequency in each conjugate pole pairs of the second controller is sampled at a period T according to" [3],[53]-[54],

$$\omega_I < \frac{\Pi}{T}, \forall\, i \tag{4.21}$$

then $G_{C2}(s)$ can be obtained from $G_{C2}(z)$.

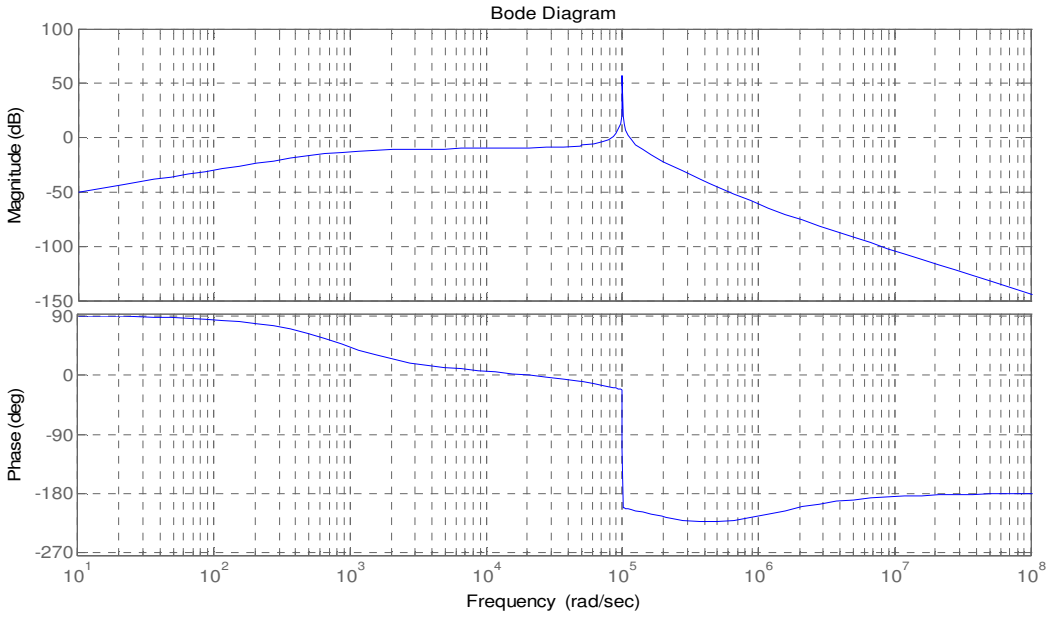
If equation (4.21) is satisfied, poles of the $G_{C2}(s)$ can be obtained using the relationship [3],[53]-[54],

$$p_I = \frac{1}{T}\left[\ln\ p_I + i\beta_I\right] \tag{4.22}$$

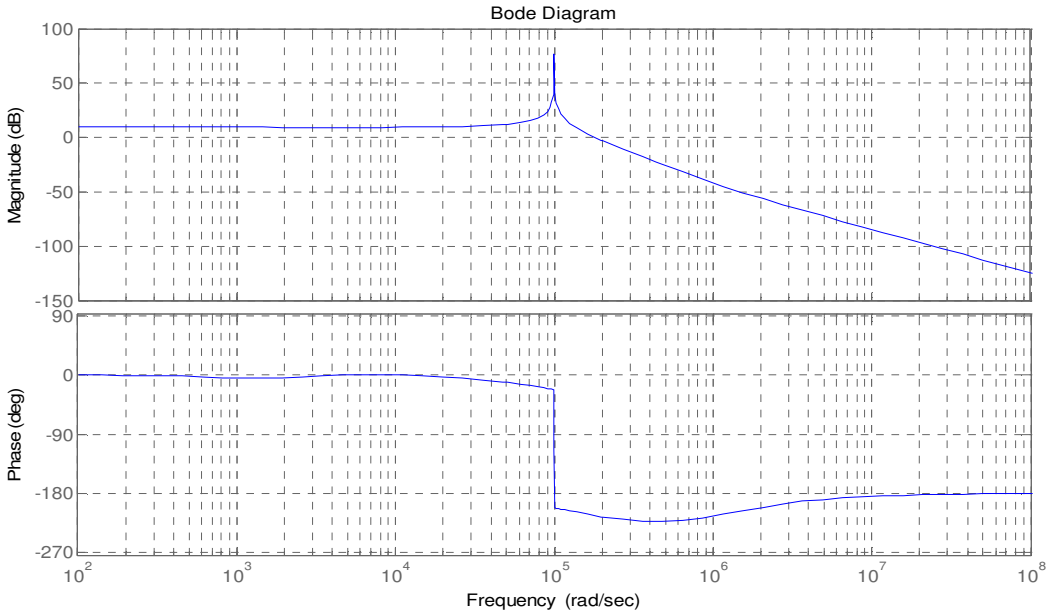
The transformation is done by imposing partial fraction expansion technique and the procedures shown in [53], and the poles for $G_{C2}(s)$ are found according to equation (4.22). The calculation of $G_{C2}(s)$ from $G_{C2}(z)$ is shown in the Appendix B. So,

$$G_{C2}(s) = 9.3 + \frac{1*10^4}{s} \tag{4.23}$$

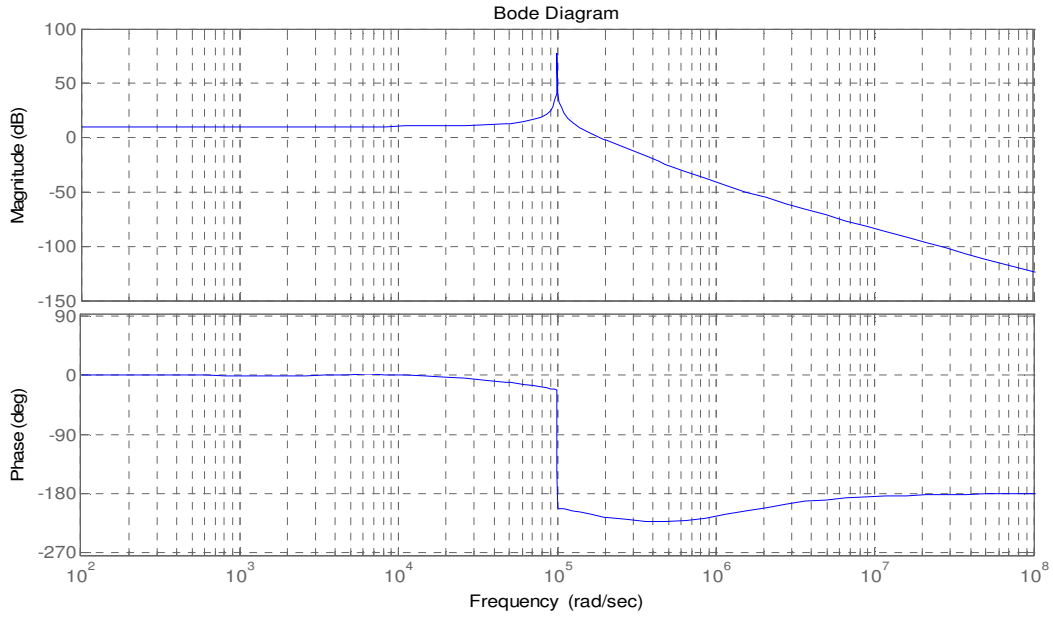
The PI controller was designed with a pole in the origin, and zeros at $-1.08*10^3$ radian/sec. Using MATLAB, the bode plot of the drive voltage with respect to $V_{REF1}(s)$ is given by equation (4.3) as shown in Fig. 4.15 (a), when $G_{C2}(s)$ is equal to equation (4.23). As can be found from Fig. 4.15 (a), the crossover frequency is 28 kHz and the phase margin is 24°. Similarly, using MATLAB, the bode plot of the drive voltage with respect to $V_{REF2}(s)$ is given by equation (4.4) as shown in Fig. 4.15 (b), when $G_{C2}(s)$ is equal to equation (4.23). As can be found from Fig. 4.15 (b), the crossover frequency is 28.7 kHz and the phase margin is 33°. Finally, using MATLAB, the bode plot of the drive voltage with respect to both the references ($V_{REF1}(s)$ and $V_{REF2}(s)$) is given by equation (4.5) as shown in Fig. 4.15 (c), when $G_{C2}(s)$ is equal to equation (4.23). As can be found from Fig. 4.15 (c), the crossover frequency is 30 kHz and the phase margin is 34°, so the LED driver system is stable.


Bode Diagram
Magnitude (dB)
100
50
0
-50
-100
-150
Phase (deg)
90
0
-90
-180
-270
10^1
10^2
10^3
10^4
10^5
10^6
10^7
10^8
Frequency (rad/sec)

(a)


Bode Diagram
Magnitude (dB)
100
50
0
-50
-100
-150
Phase (deg)
90
0
-90
-180
-270
10^2
10^3
10^4
10^5
10^6
10^7
10^8
Frequency (rad/sec)

(b)



(c)

Figure 4.15 Bode plot of the (a) drive voltage with respect to $V_{REF1}(s)$, (b) drive voltage with respect to the $V_{REF2}(s)$, and (c) drive voltage with respect to both the references – when $G_{C2}(s)$ is equal to equation (4.23).

4.7 SIMULATION RESULTS

The performance characteristics of the proposed two-loop LED driver was verified using MATLAB SIMULINK during both steady state and changing of the feedback voltage of the second controller [1],[6]. The complete model is implemented in MATLAB SIMULINK as shown in Fig. 4.16. The analog controller is modeled according to equation (4.17), and the digital controller is modeled according to equation (4.19). The model was simulated using ODE3 (Bogacki-Shampine) differential equations. A fixed step size of $0.1*10^{-6}$ seconds was used to obtain accurate results.

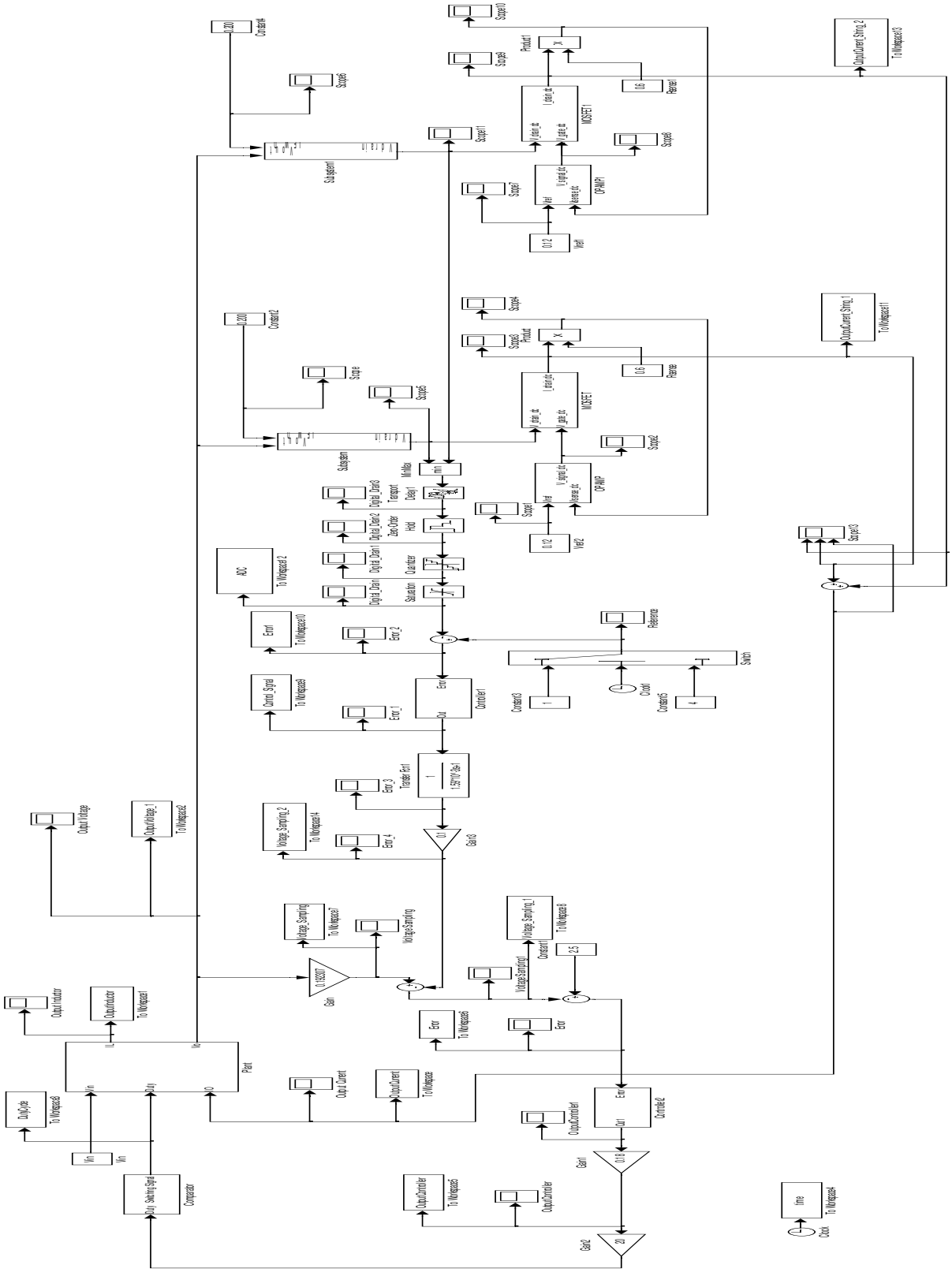


Figure 4.16 Proposed driver model implemented in MATLAB SIMULINK.

The LED loads are modeled as shown in Fig. 4.17. The forward voltage drop across the LEDs are deduced from

$$v = iR_S + nV_T ln\left(\frac{i}{I_S}\right) \tag{4.24}$$

where n is a constant, i is the forward current in the LEDs, V_T is the thermal voltage, and I_S is the saturation current [61]-[62]. V_T, is defined by

$$V_T = \frac{kT}{q} \tag{4.25}$$

where k is the Boltzmann's constant, T is the temperature, and q is the charge of electrons [61]. I_S, is defined by,

$$I_S = Aqn_i{}^2\left(\frac{D_p}{L_pN_D} + \frac{D_n}{L_nN_A}\right) \tag{4.26}$$

where A is the area, n_i is the intrinsic carrier density, D_p and D_n is the diffusion constant of holes and electronics, L_p and L_n is the diffusion length of holes and electrons, and N_A and N_D is the concentration of holes and electrons [61]. A MATLAB m-file for the calculation of the LED forward voltage drop is shown in the Appendix B.

Since the first loop with its SMPC provides the worst case drive voltage to the LED strings leading to unwanted power losses in the LED strings. Thus, the second loop is added in order to adjust the drive voltage of the LED strings to its minimum requirement in order to maintain regulation in the LED strings after the worst-case drive voltage has stabilized. Thus, it is modeled by reducing the reference voltage of the second loop as shown in Fig. 4.16, from 4 V to 1 V. So, the signal output of the second controller changes in order to accommodate the change in reference for the second controller. This voltage gets subtracted from the sensed drive voltage,

and the output is the feedback voltage for the analog controller. Since the feedback voltage of the analog controller increases at this instance, the duty cycle of the PWM signal of this controller reduces, resulting in decrease in drive voltage of the LED strings. The ADC is modeled using zero-order hold and quantizer blocks from the SIMULINK library.

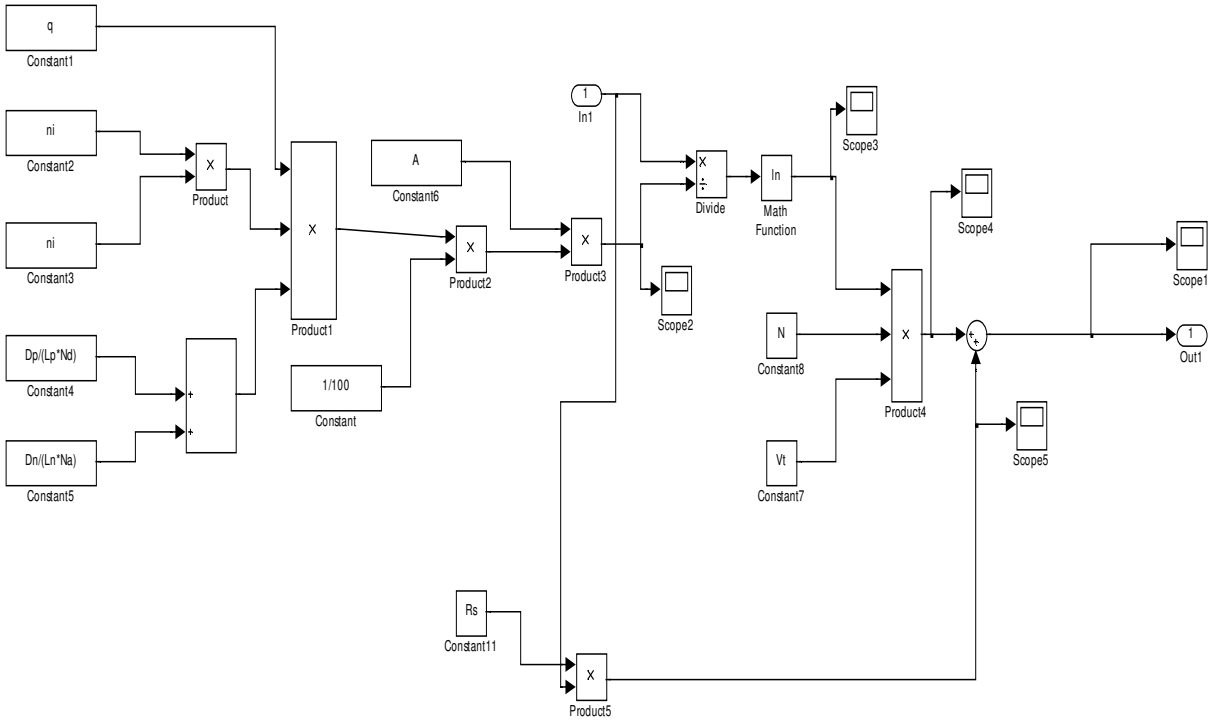
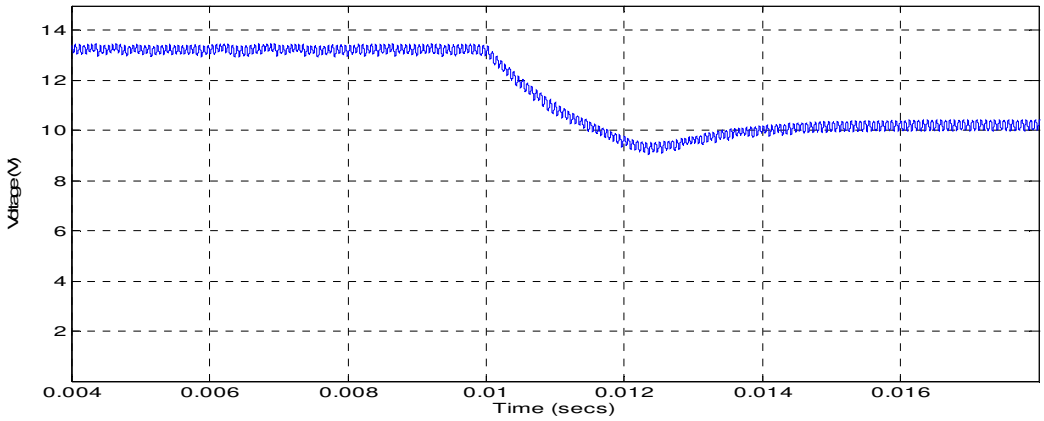


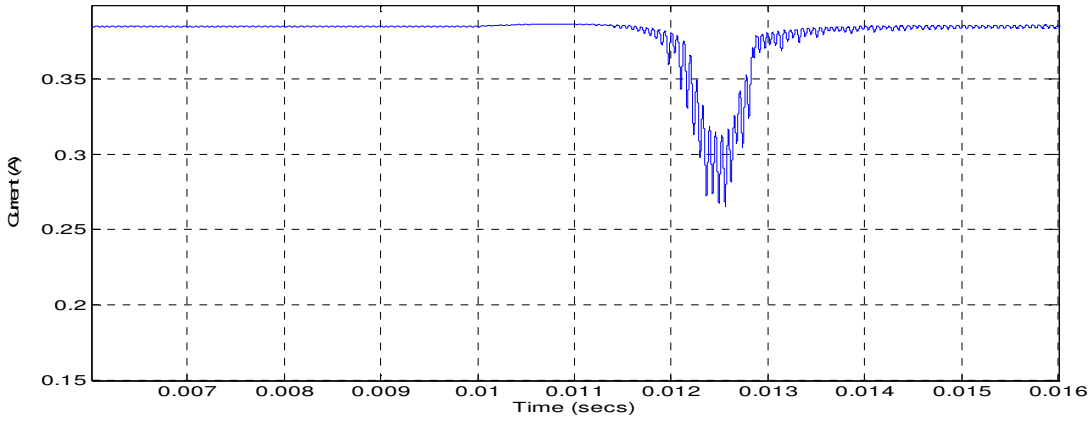
Figure 4.17 LED model implemented in MATLAB SIMULINK.

Fig. 4.18 (a) shows the output drive voltage being adjusted to a minimum value within 4ms from 13 V to 10.3 V, after initial worst-case drive voltage of 13 V has stabilized and the reference of the second controller has changed to 1 V from 4 V at 10ms. The ripple output voltage is ± 200 mV which is above the specified limit. Fig. 4.18 (b) shows the output current during this interval. The ripple output current is ± 0.3 mA which is below the specified limit. As can be seen from the figure that the output current momentarily drops to 270 mA and then recovers back to 400 mA.

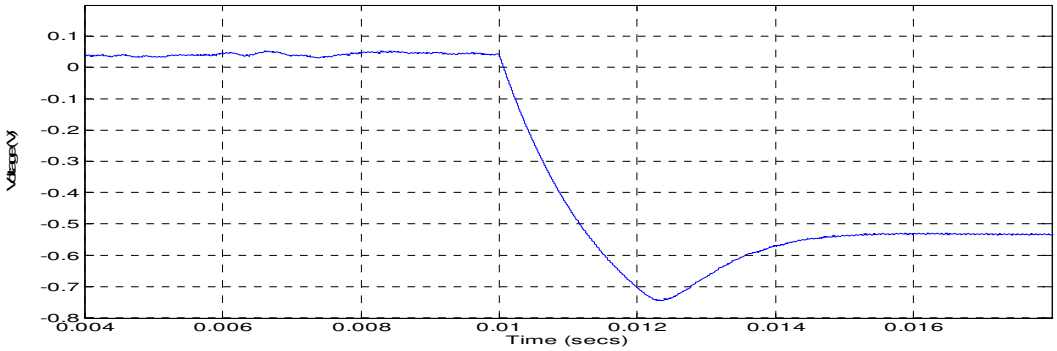
Fig. 4.18 (c) shows the output of the digital controller changing from 0 V to -0.54 V during this transition. As explained earlier, initially the SMPC with its analog controller provides the worst-case drive voltage and then when this voltage stabilizes the digital controller starts to inject voltage to the feedback summing node in order to reduce the drive to its minimum for the set current – this is proved by this figure. The output voltage of the feedback summing node is shown in Fig. 4.18 (d). This voltage is maintained at 2.5 V due to the reference of the analog controller being at 2.5 V.



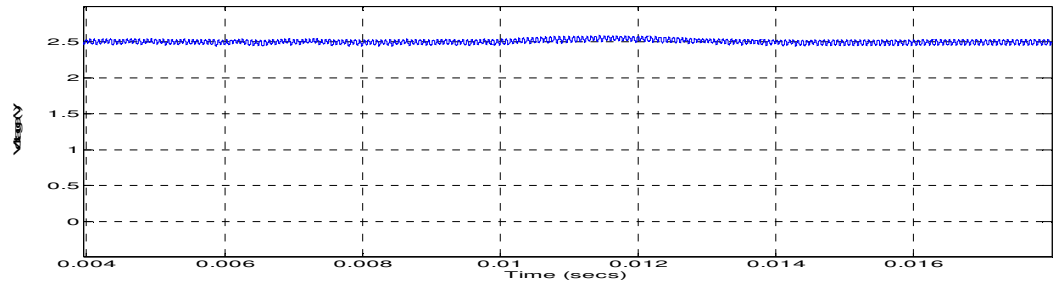
(a)



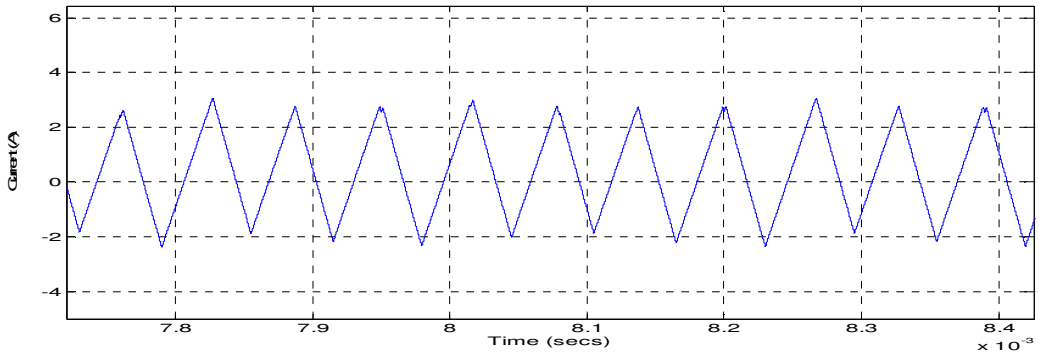
(b)



(c)



(d)



(e)

Figure 4.18 (a) Output voltage, (b) Output Current, (c) Output of the digital controller, (d) Output of the feedback summing point, and (e) Inductor current of the proposed LED driver.

4.8. EXPERIMENTAL RESULTS

A prototype was built to verify the performance characteristics of the proposed LED driver driving two LED strings of white LEDs. The maximum voltage drop of the selected LED is 4 V, and since the output drive voltage is lower than the input voltage, a buck converter with its key parameters and components are listed in Table 4.1 is used to provide the drive voltage to the LED strings. A photograph of the prototype built in lab is shown in Fig. 4.19.

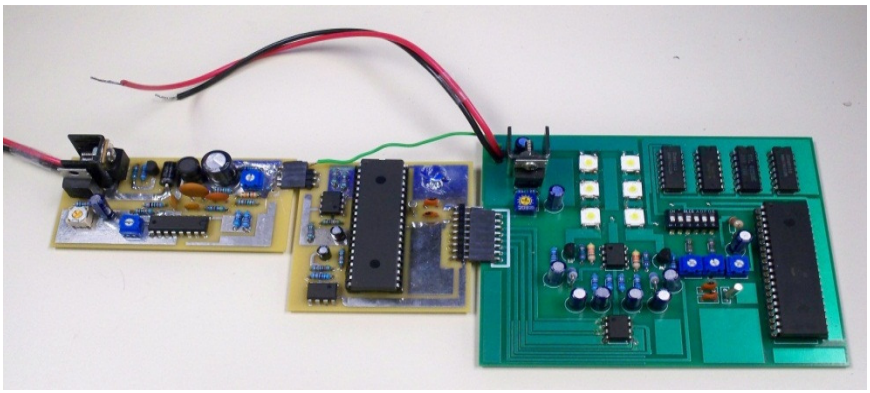


Figure 4.19 Photograph of the prototype LED driver.

The first loop provides the LED strings with worst-case drive voltage of 13 V at 400 mA. But this leads to unwanted power losses in the MOSFETs of the current controllers, yielding a low efficiency at the LED load. Thus, a second controller is added to find the optimum drive voltage for the LED strings at the set current, leading to high efficiency at the LED load. Fig. 4.20 shows when the output voltage transition to its minimum value for the set current. The upper waveform (Channel 1) is the measured output drive voltage of the driver, the second waveform (Channel 2) is the measured output of the second controller, the third waveform (Channel 3) is the measured duration of the efficiency optimization mode, and the lower waveform (Channel 4) is the measured output current of the driver. The output drive voltage adjusts to its minimum value of

10.2 V from initial worst-case value of 13 V, required to maintain 200 mA in each string of LEDs. As can be seen in the figure, the output of the second controller changes from 0 V to -0.54 V, when efficiency optimization mode is enabled, in order to inject voltage at the feedback summing node. Thus, the feedback voltage of the first controller increases, leading it to decrease the duty cycle of the PWM signal for the switch of the SMPC. The efficiency at the LED loads, are calculated according to equation (3.27). Therefore, the efficiency at the LED loads after transition is 89 %. Thus, adding the second controller results in efficiency improvement of about 21 %, clearly demonstrating the advantages of using this approach in driving LED strings.

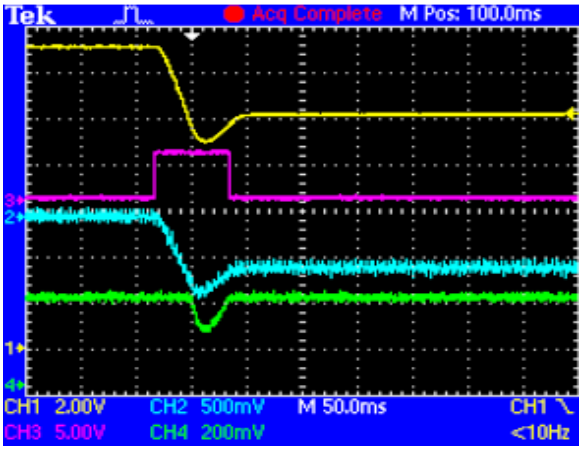


Figure 4.20 Measured output voltage, duration of efficiency optimization mode, output of the second controller and output current of the driver during adjustment of output drive voltage of the LED driver.

In order to verify the mode of operation of the buck driver, the inductor current is measured at maximum driver current of 400 mA after the second controller has minimized the drive voltage of the LED strings, and is shown in Fig. 4.21. The upper waveform is the measured output voltage (Channel 1), the second waveform is the measured PWM signal (Channel 2), the third waveform is the measured output of the feedback summing node (Channel 3), and the bottom waveform is the measured inductor current (Channel 4) of the buck driver. From the measured

output voltage, the peak to peak ripple voltage is ± 50 mV which is above the specified limit. Additionally, from the figure, the measured inductor current implies the buck converter is operating in continuous conduction mode.

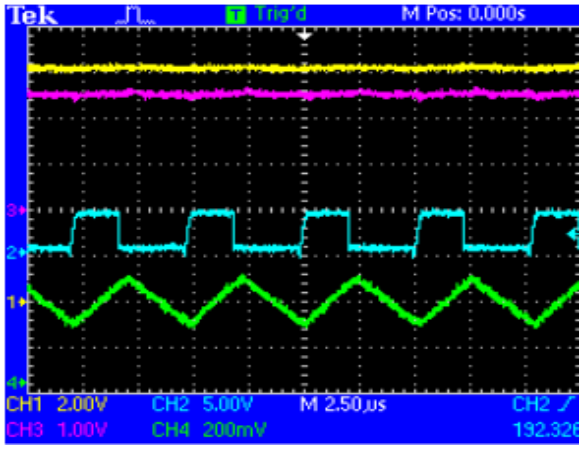


Figure 4.21 Measured output voltage, PWM signal, output of the feedback summing node and inductor current of the buck driver after the second controller has minimized the drive voltage of the LED strings.

The load transients of LED string current from 100 % PSPWM signal to 80 % PSPWM signal is shown in Fig. 4.22. The upper waveform (Channel 1) is the measured output voltage, the middle waveform (Channel 3) is the measured duration of efficiency optimization mode, and the lower waveform (Channel 4) is the measured output drive current. During the transition of PSPWM signal, efficiency optimization mode is enabled to find a minimum drive voltage for the new current setting, leading to improved efficiency.

The self-adjusting nature of the output voltage of the driver due to PSPWM is clearly demonstrated in Fig. 4.23. The upper waveform (Channel 1) is the measured output voltage, the middle waveform (Channel 4) is the output current of the driver, and measured PSPWM signal of the current controllers of the LED strings are also shown (Channel 3 and 4). From the figure,

the adjustment of the output voltage in a step fashion leads to improvement in efficiency at the LED load [63].

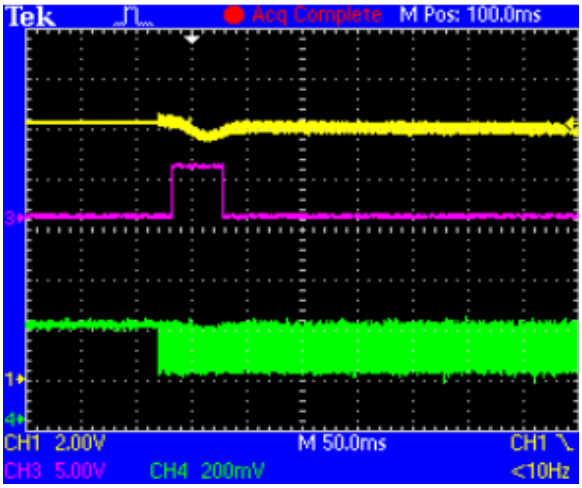


Figure 4.22 Measured output voltage, and output current during load transient from 100 % to 80 % PSPWM signal.

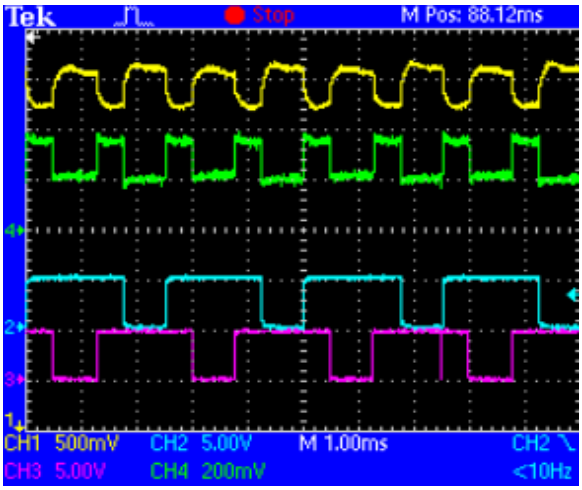


Figure 4.23 Measured output voltage, output current and the PSPWM signals of the current controllers of the LED strings.

In Fig. 4.24, the measured output current and voltage as a function of input voltage is shown for the designed LED driver at maximum driver current of 400 mA after the drive voltage has

transitioned to the minimum drive voltage. As observed in the figure, the variation is within 1 % for output voltage.

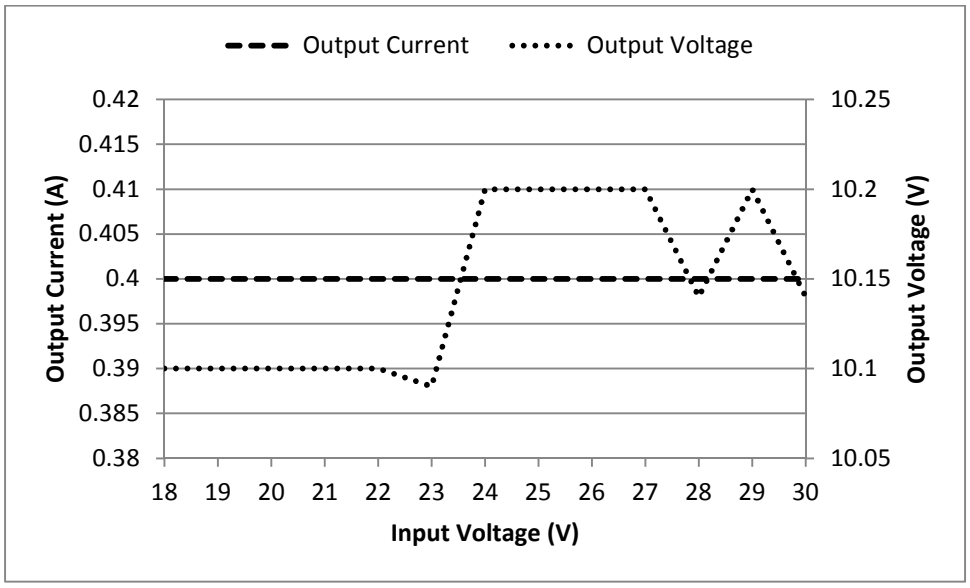


Figure 4.24 Measured output voltage and current for different input voltages.

Table 4.2 Comparison of drivers between the proposed design and the design in [6].

Parameter	[3]	[6]
Efficiency (%)	>89%	>87%
Input Voltage (V)	24	24
Output Voltage (V)	10.2	33.28
Max Output Current (mA)	400	1050
Frequency (kHz)	200	120

The designed driver was compared with the drivers designed in [6] and major performance characteristics are summarized in Table 4.2 [4]. The load in [6] consisted of four resistive loads of different values in parallel on order to emulate equivalent LED loads, whereas in the proposed driver there are two parallel strings of LED loads with three LEDs in each string. As can be

seen, efficiencies are very similar in both the drivers for the desired output voltage. However, the efficiency of the proposed LED driver system will be higher than [6] at higher output voltage.

4.9 SUMMARY

A novel LED driver for driving multiple LED strings in backlight application of LCD panel is presented in this chapter. The driver used an analog/digital controller to minimize the drive voltage of the LED strings, leading to improved efficiency at the LED loads and reduction of unwanted power losses in the MOSFET of the current controllers. Additionally, PSPWM dimming is implemented leading to a better overall system performance. The separate control of LED drive voltage and LED current leads to a better system stability [64]. The proposed driver was experimentally verified for a load current of 400 mA leading to improvement of 21 % in efficiency at this current setting.

CHAPTER 5

A LED DRIVER IC FOR LED PIXELS

5.1 INTRODUCTION

Introduced in this chapter is a LED driver IC for LED display panels. The LED driver chip is implemented in a 0.5 micron CMOS process due to cost and available processes. In this implementation, a digitally controlled LED driver using the fabricated IC was designed to drive five strings of green color LEDs. The proposed driver maintains high efficiency at the LED load by maintaining a minimum voltage across the LEDs and the current controllers necessary to keep it in regulation by selecting the minimum drain voltage of the MOSFETs of the current controllers as the feedback voltage for the SMPC driver. In the proposed driver, an external dimming controller circuit is also able to dim all five LEDs in the pixel. At the maximum rated current of 100 mA, efficiency of 75 % for the green LED was experimentally verified.

Section 5.2 provides a detailed description of the LED driver for driving five strings of green LEDs for display panel applications. It also explains the two different modes of operation of the digital controller in order to maintain a minimum drive voltage across the LED strings. Section 5.3 provides a detail description of various blocks of the proposed LED driver IC. Finally, Section 5.4 shows the experiment results of the complete LED driver system using the fabricated IC.

5.2 CIRCUIT DESCRIPTION

The proposed LED driver is shown in Fig. 5.1. It consists of a multiplexer, current-controllers, a dc-dc SMPC, and a TMS320F28335 digital-signal-processor (DSP) to drive five strings of

---

A report based on this chapter is submitted to MOSIS.

individual green LEDs. Similar arrangements are used for red and blue color LED drivers in a RGB LED. Current controller is used to maintain the current across the individual LED string by sensing the voltage drop across the sensing resistor and comparing it with the dimming voltage in the EA. The EA processes this error voltage to regulate the corresponding MOSFET to maintain the desired LED string current [6]-[7].

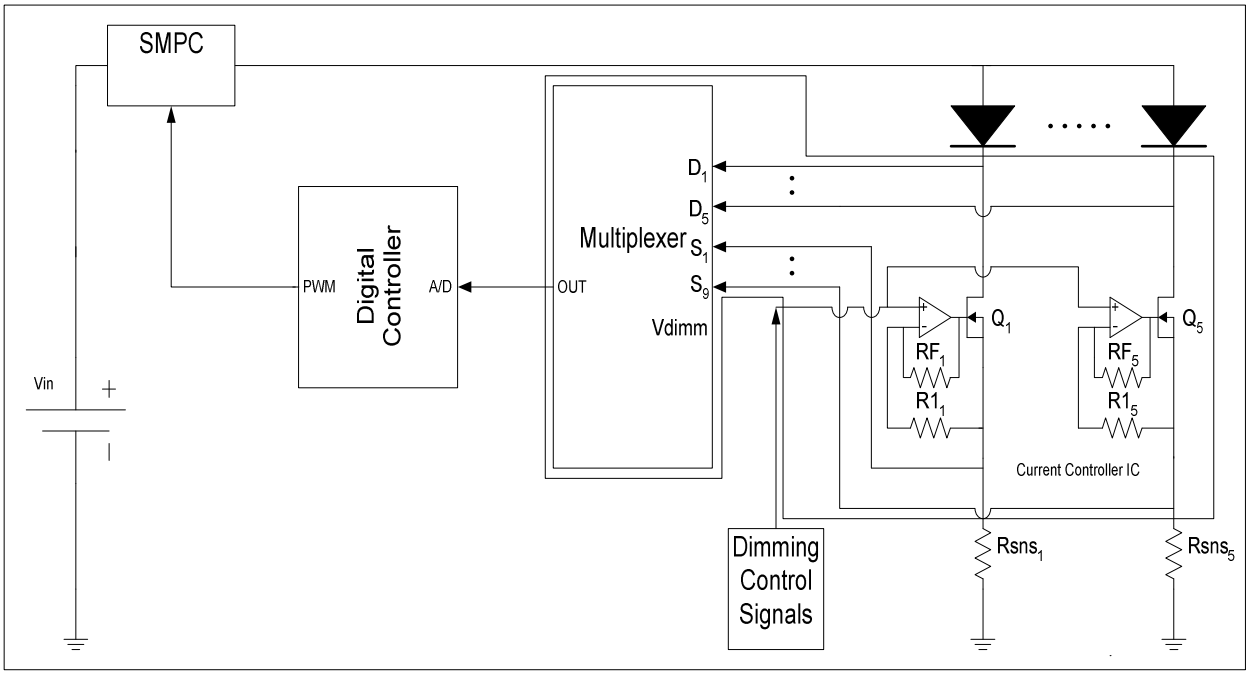


Figure 5.1 Proposed LED driver.

As mentioned in section 2.4, the use of sensing diode leads to fluctuation of sensing voltage due to temperature dependence of diode voltage. Thus, a multiplexer is used to sense the gate (V_G), drain (V_D), and dimming control voltages (V_{DIMM}). The output of the multiplexer is connected to the analog-to-digital (A/D) converter inputs of the DSP, which compares the reference voltage to the sensed minimum V_D of the MOSFETs of the current controllers to generate the error voltage during the operation mode of the driver. This error voltage is processed by a proportional-integral-derivative (PID) controller inside the DSP to generate the correct duty cycles for the

PWM signal for the switches of the SMPC. In this implementation, the current-controllers, and the sensing multiplexers are integrated in the proposed IC, leading to a reduction of size of the overall system. The analog dimming interface is externally generated and the signals are fed to the corresponding I/O pins of the proposed IC. The drive voltage for the LED strings is provided by the non-isolated buck converter since the drive voltage for the green LEDs is 4.5 V to maintain the desired current of 100 mA.

The power losses in the MOSFETs of the current controller are significant disadvantages of using current controllers, thus the MOSFETs of the current controllers are maintained to operate with minimum V_{DS} required for the desired current setting. For example, if a single 5 V power supply was used to drive five strings of green LEDs; it would yield an efficiency of 64 % at the nominal current of 20mA. Thus, in order to maintain minimum voltage drop across the MOSFETs of the current controllers, the proposed LED driver operates in two modes of operation: efficiency optimization and operation mode [4]. In the efficiency optimization mode, the minimum V_{DS} is found indirectly to maintain the desired LED string current by reducing the duty cycle of the PWM signal until the sensed maximum V_S is lower than the threshold value [3]. Fig. 5.2 shows the flowchart of this mode. In the operation mode, the control algorithms are processed in order to generate the corresponding duty cycles for the PWM signal for the desired LED string current and input voltage settings. Fig. 3.3 shows the flowchart of this mode.

EFFICIENCY OPTIMIZATION MODE:

1) Initially at startup, the DSP would output a PWM signal so the SMPC output voltage would be 5V, which is high enough to maintain the MOSFETs of the current controllers in saturation region and the LED strings are at the desired current setting

[4],[8].

2) For each string, V_S of the MOSFETs are sensed by the multiplexer and are sent to the DSP for processing through the A/D input pin while the current controllers are maintaining the desired current settings in the LED strings. The maximum V_S is found from these sensed voltages.
3) The sensed maximum V_S is compared against a threshold value in order to determine the proper operation of LED strings. This threshold value is the maximum V_S in order to operate the MOSFETs of the current controllers with minimum V_{DS} required to maintain the set current in the LED strings.
4) If the sensed maximum V_S of the MOSFETs of the LED strings is above this threshold value, the duty cycle of the PWM signal for the switches of the SMPC is decreased leading to reduction of drive voltage provided by the SMPC to the LED strings. After the reduction of duty cycle of the PWM signal, the microcontroller senses the V_S of the MOSFETs again and above process continues until the V_S of the MOSFETs are below the threshold indicating the LED strings are no longer able to maintain the desired current setting.
5) Therefore, the duty cycle of the PWM signal is increased by a value to maintain the desired current in the LED strings and then enters the operation mode.

OPERATION MODE:

1) The multiplexer senses the V_D and sends it to the microcontroller through the A/D input pin.

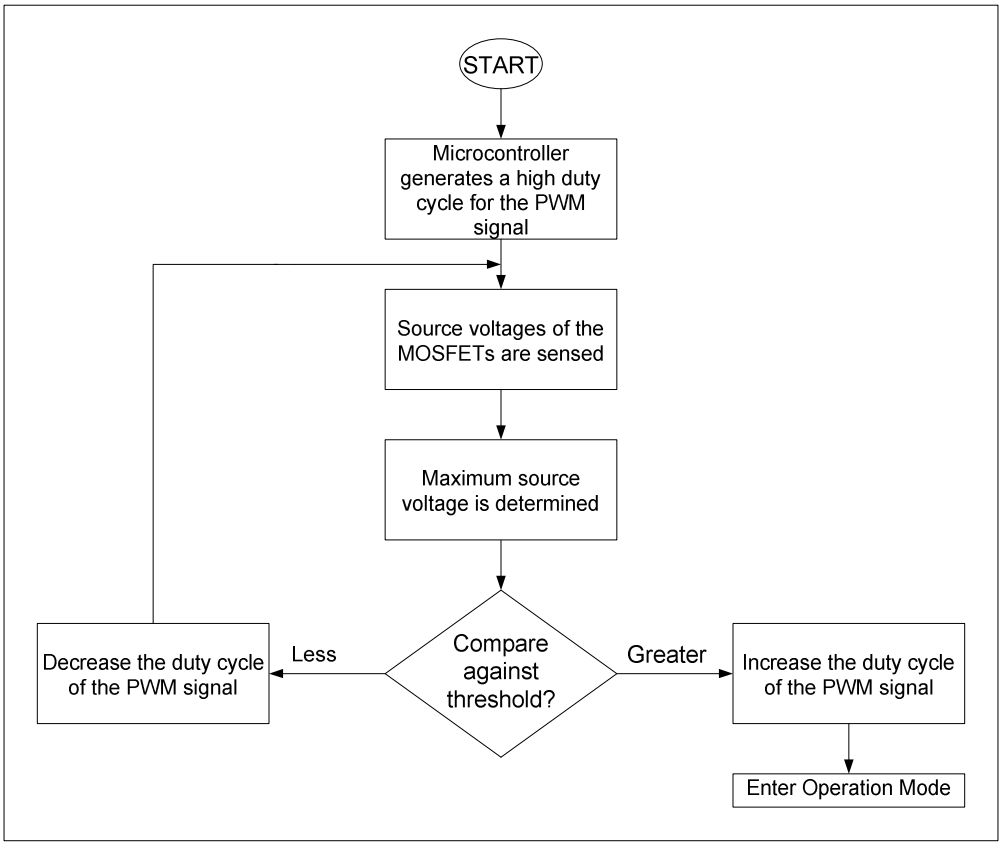


Figure 5.2 Flowchart of efficiency optimization mode.

2) At the beginning of this mode, the minimum V_D is found from the sensed V_D, and stored in memory. The minimum V_D is set as the reference voltage at this time for the set current level. Thus, the SMPC is adjusted according to this reference voltage leading to the LED strings being driven by the minimum drive voltage and hence, reducing unwanted power losses in the MOSFETs of the current controllers.

3) Then, error voltage is found by comparing this reference voltage with the sensed minimum V_D of the MOSFETs, i.e., $V_{ERROR} = V_{REF} - V_{MIN(D_1,.....,D_5)}$. This error voltage is processed by the PID controller in order to generate the corresponding duty cycle of the PWM signal for the switches of the SMPC.

5.3 CIRCUIT IMPLEMENTATION

5.3.1 ERROR AMPLIFIER

The EAs form the heart of the current controllers of the LED strings. The amplifiers regulate the MOSFETs to maintain the desired current in the LED strings. Several design aspects are taken into consideration during the design of this EA. First, the amplifier should not saturate from large input signals, maintain proper voltage swing, and must be biased by low biasing current [65]-[70]. Secondly, proper compensation technique must be implemented to maintain a stable operation with a sufficient phase margin.

Based on the above specifications, a two-stage single-ended CMOS operational amplifier is chosen for this design with p-channel input transistors due to its low noise immunity, and improved input common mode range [67],[70]. Fig 5.3 shows the implemented operational amplifier in the fabricated LED driver IC.

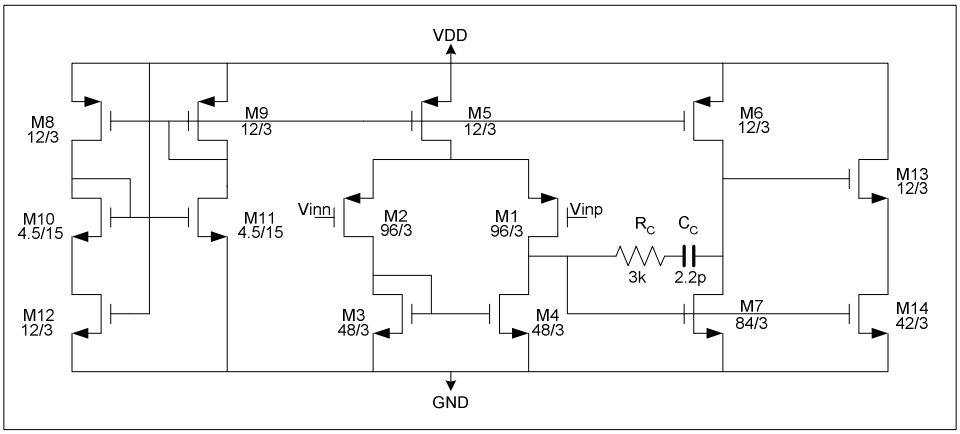


Figure 5.3 Single-ended two-stage operational amplifier.

The biasing stage is implemented using a Wilson current mirror (M8, M9, M10 and M11) with its high output resistance, to provide the biasing current for the differential and common source

amplifier stages. M12 acts as a resistor by operating in triode region of operation to provide the reference current for the current mirror configuration. A NMOS transistor is used instead of a resistor in this implementation to improve the power dissipation of the circuit [67]. A PMOS differential amplifier stage (M1, M2, M3, and M4) is implemented with an active load (M5) to increase the gain. The output stage is implemented with a common source configuration (M6, and M7) to maintain the gain of the overall operational amplifier. Since common source configuration is not very stable, a Miller compensation resistive-capacitive network is added to provide a negative feedback between the input and output to maintain stability. The resistor and capacitor values of the miller compensation network were found to be 3 kΩ, and 2.2 pF, respectively. Additionally, an output buffer stage formed using transistors M13 and M14 is used to drive low resistance output loads. Fig. 5.4 shows the layout of the designed amplifier in cadence virtuoso.

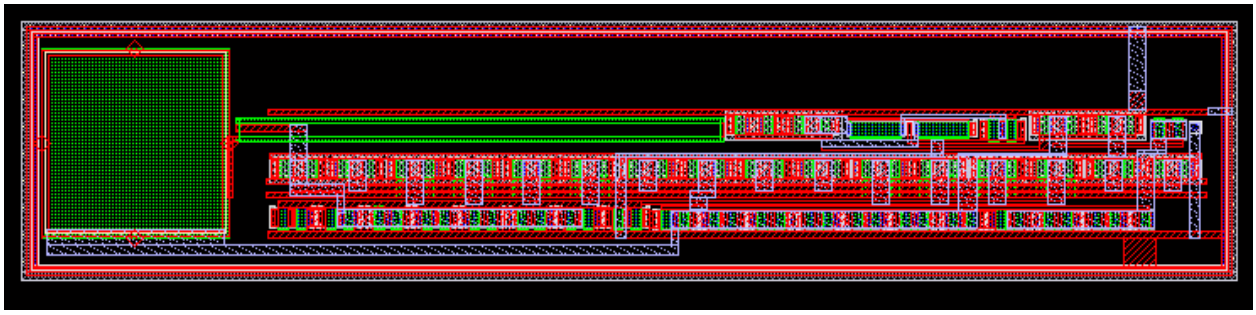


Figure 5.4 Layout of the two-stage CMOS operational amplifier.

The simulation of the above operational amplifier is performed in Orcad PSPICE 16.3 at level 7. The SPICE file for the selected 0.5 micron CMOS process is listed in the Appendix C. In this simulation, in order to account for the capacitance in I/O pads, a 6 pF capacitance was added to the output of the designed operational amplifier. Fig. 5.5 shows the Bode plot of the designed operational amplifier. As can be seen from the figure, the phase margin is 52.5°, and a unity gain

frequency is 10 MHz. Fig. 5.6 shows the output voltage swing of the designed operational amplifier with matching feedback resistors in a non-inverting configuration. It can be observed from the figure that the voltage swing is 2, i.e., $G = \frac{V_O}{V_{IN}} = 1 + \frac{R_2}{R_1} = 2$, where R_2 and R_1 are equal.

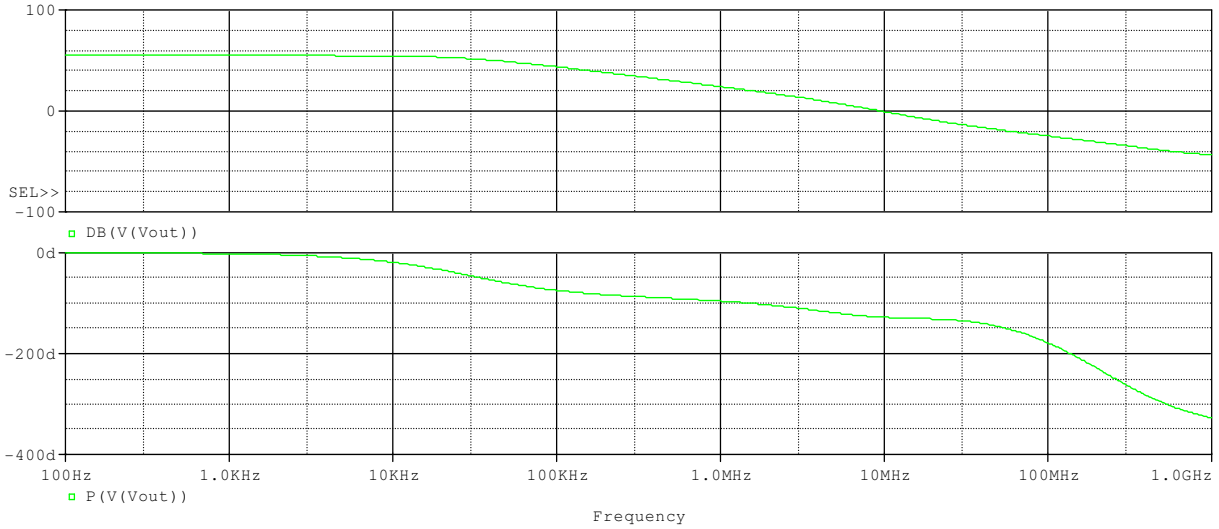


Figure 5.5 Bode plot of the operational amplifier.

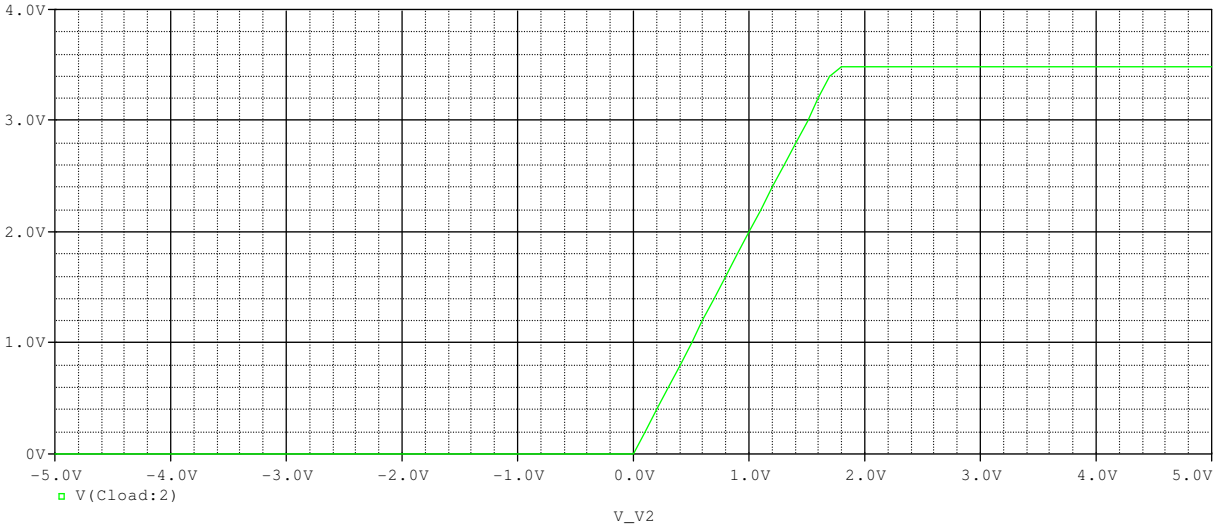


Figure 5.6 Output voltage swing.

5.3.2 ANALOG MULTIPLEXER

A 16 to 1 analog multiplexer as shown in Fig. 5.7 is implemented for sensing the drain and source voltages of the integrated MOSFETs of the current controllers. In this implementation, the outputs of each multiplexer bit are tied together to a common output wire which is connected to an I/O pin of the fabricated IC. In this design, due to reduction of noise and improved coupling between input and output, a T-switch multiplexer is designed instead of the traditional transmission gates based multiplexers. In T-switch multiplexers, signals Phi1 and Phi2 are complementary signals. M1, M3, and M5 are NMOS transistors, and M2, M4, and M6 are PMOS transistors. Therefore, the complementary pair of transistors results in rail-to-rail swing capabilities. Thus, when signal Phi1 is high, making Phi2 low, in this case Analog In is connected to Analog Out since M1, M2, M3, and M4 are all ON. In the other case, when signal is Phi1 is low, making Phi2 high, the top transistors are all off, and M5 and M6 are ON. Additionally, the point X being grounded, leading to a better decoupling of the Analog In and Analog Out. In order to minimize crosstalk, transistors M5 and M6 are laid out to be three times larger than the other transistors, resulting in faster discharge of gate charges [65]. Fig. 5.8 shows the layout of the designed multiplexer in cadence virtuoso.

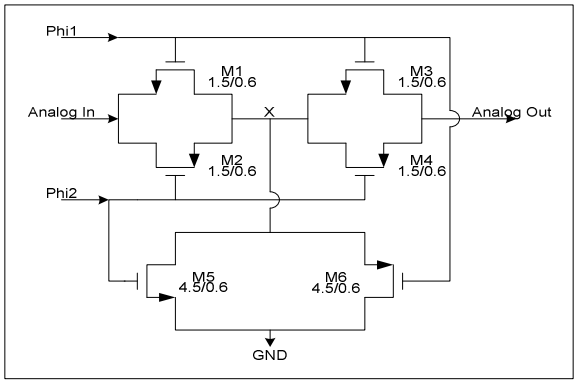


Figure 5.7 A T-switch multiplexer.

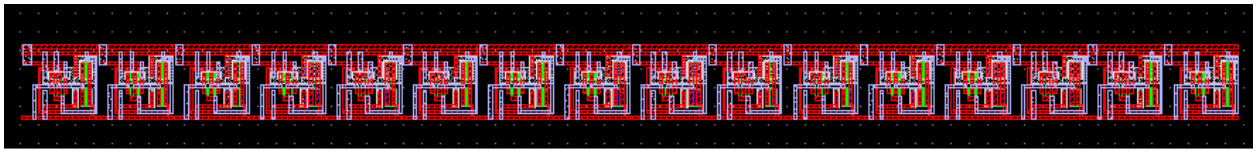


Figure 5.8 Layout of the 16 to 1 multiplexer.

5.3.3 DECODERS

An array of 16 4-input NOR gates as shown in Fig. 5.7 are used as the decoder for the 16 to 1 T-switch multiplexer. The inputs of the NOR gates are connected to the digital I/O pins of the fabricated IC. Buffers are used before and after the decoders to be able to switch the multiplexers in relatively high speed since the synchronous buck driver is operating at 100 kHz. Additionally, buffering is used to turn the decoders and multiplexers ON and OFF and prevent any unwanted spike [65]. Fig. 5.10 shows the layout of the designed decoder in cadence virtuoso. Fig. 5.11 shows the complete layout of the chip in cadence virtuoso.

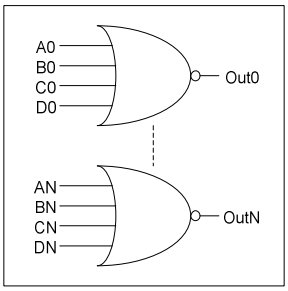


Figure 5.9 Decoders.

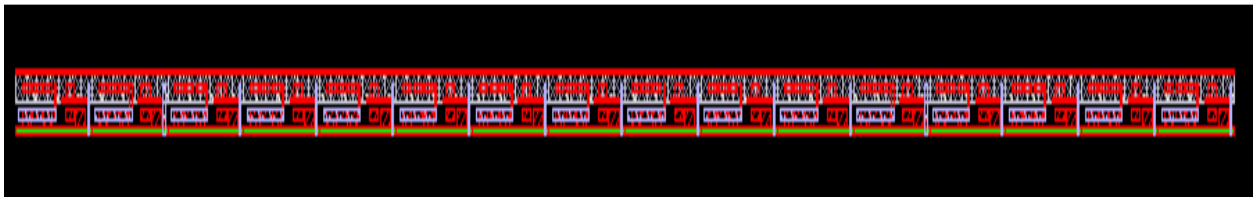


Figure 5.10 16 4-input NOR gate decoders.

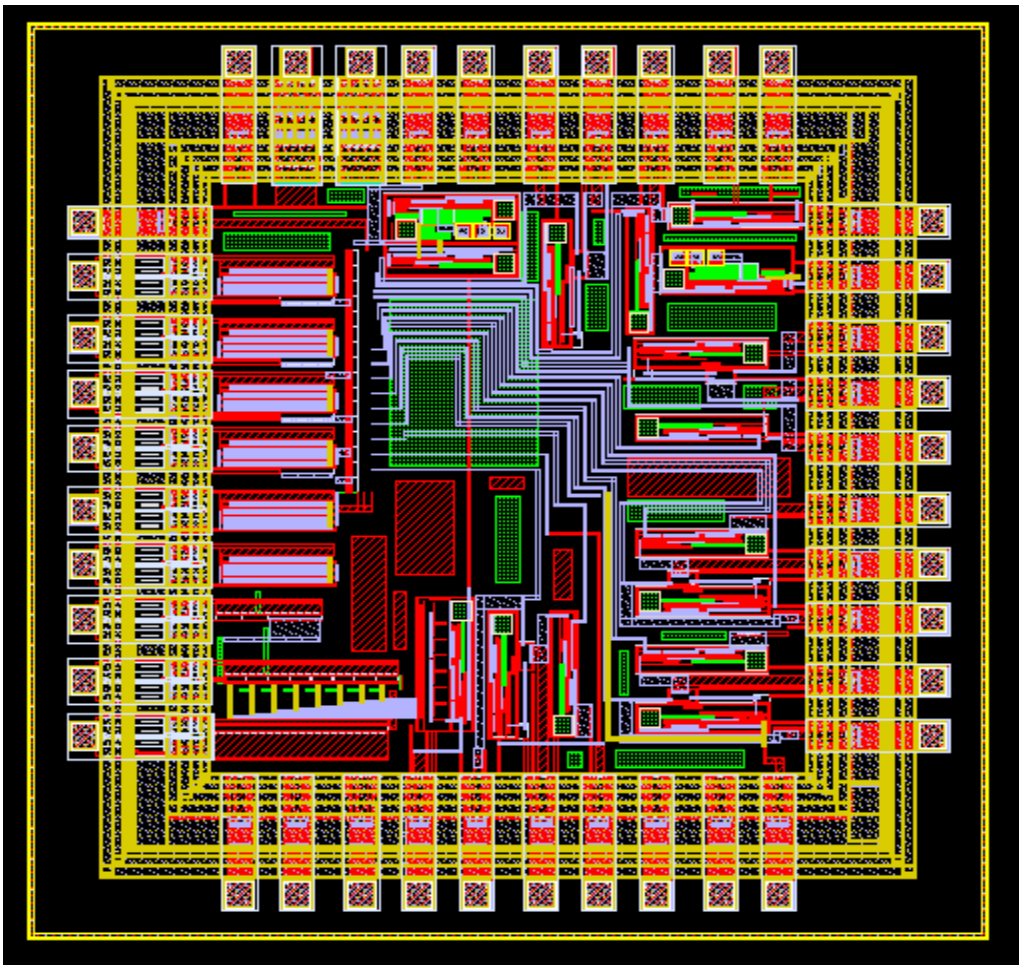


Figure 5.11 Complete layout of the chip.

5.4 EXPERIMENTAL RESULTS

A prototype was built to verify the performance characteristics of the LED driver driving a five string green LEDs. The synchronous buck converter provides the drive voltage for the LED strings, and its key parameters and components are listed in Table 3.2. A photograph of the prototype LED driver is shown in Fig. 5.12.

In order to verify the mode of operation of synchronous buck driver, the inductor current is measured at a maximum driver current of 100 mA, and is shown in Fig. 5.13. The upper

waveform is the measured high side PWM signal (Channel 1), the second waveform is the measured low side PWM signal (Channel 2), the third waveform is the measured output voltage (Channel 3), and the bottom waveform is the measured inductor current (Channel 4) of the synchronous buck driver. From the measured output voltage, the peak to peak ripple voltage is ± 112 mV which is above the specified limit. Additionally, from the figure, the measured inductor current implies the synchronous buck converter is operating in continuous conduction mode.

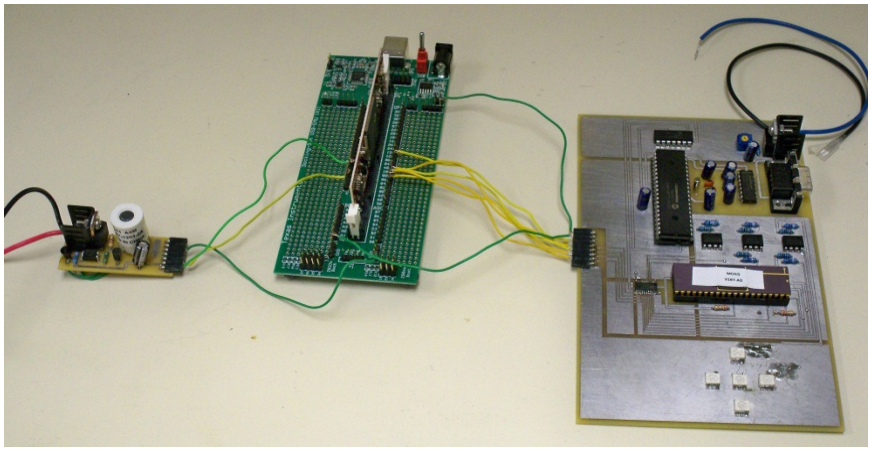


Figure 5.12 Photograph of the prototype LED driver.

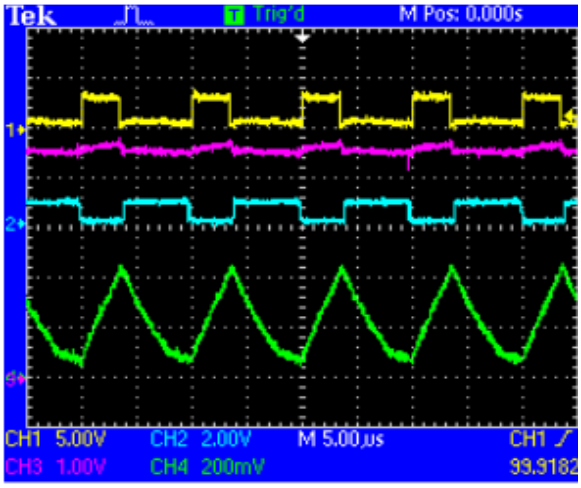


Figure 5.13 Measured PWM signal, output voltage, and inductor current of the synchronous buck driver.

In order to maintain minimum voltage drop across the current controllers of the LED strings for the set current, an efficiency optimization mode is implemented in TMS320F28335 as explained in section 5.2. Fig. 5.14 shows the measured output voltage of the synchronous buck driver (upper waveform, Channel 1) and the measured duration of the efficiency optimization mode (bottom waveform, Channel 2) at startup. It can be observed that initially, the output voltage of the driver is about 5 V, thus the MOSFETs of the current controllers are in saturation region of operation leading to unwanted power losses for the set current in the LED strings. Then, the efficiency optimization mode is enabled for duration of 5ms in order to reduce the output voltage to its minimum value necessary to maintain the set current, thus forcing the MOSFETs of the current controllers to operate with minimum voltage drop, leading to an increased efficiency at the LED load.

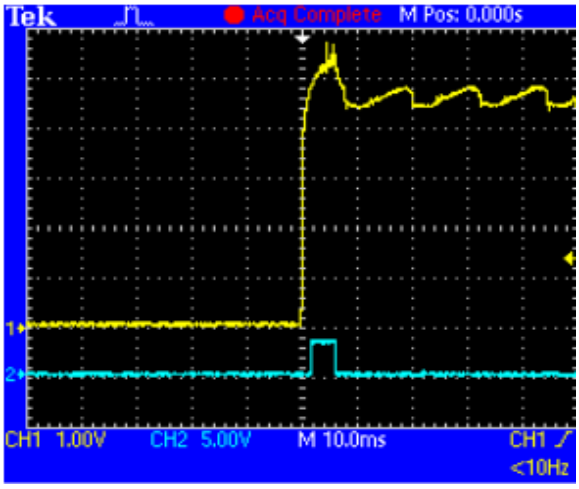


Figure 5.14 Measured output voltage and duration of efficiency optimization mode for synchronous buck driver.

In Fig. 5.15, the measured output current and voltage as a function of input voltage at maximum driver current of 100 mA. As observed in the figure, the variation is within 1 % and 3 % for

output voltage and current, respectively.

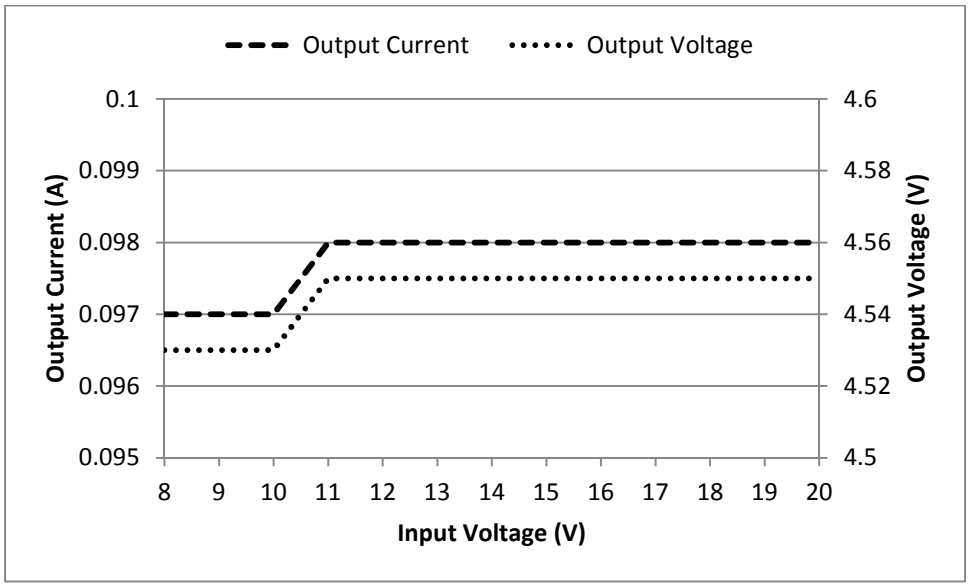


Figure 5.15 Measured output voltage and current for different input voltages.

5.5 SUMMARY

The proposed driver for driving five green LEDs for display panel applications using a digital controller is demonstrated in this chapter. In this implementation, the efficiency at the LED load is maximized by driving the LED strings with the minimum drive voltage, thus reducing unwanted power losses in the current controllers. Additionally, an external dimming controller interface is able to dim each LEDs leading to high color contrast. Using the current controllers to maintain current in the LED strings, and synchronous buck converter for providing the drive voltage leads to a better system stability since the current and voltage are controlled independently. Efficiency of 75 % at the LED load was experimentally verified for green LEDs at maximum output current of 100 mA – an improvement of 11 %.

CHAPTER 6

CONCLUSION

6.1 CONCLUSION

In this dissertation, three different types of LED drivers have been presented. Since LEDs are currently being introduced in various applications due to their advantages over other traditional light sources, there is a need for LED driver with optimal performance and increased efficiency. Current off-the-shelf power supplies do not result in ideal drivers for LEDs since LEDs are current-driven devices. Thus, in this work, a through and complete investigation of drivers in the system and IC level has been evaluated.

Chapter 3 introduces a new LED driver for RGB display panel using a multiplexer for sensing the voltage drop across the current controllers rather than using diodes. A PIC18F4431 microcontroller was used to control the drive voltage of each color of a RGB LED in a 3x3 display panel. An external analog dimming interface was implemented using daisy chaining multiple digital potentiometers to generate the 27 different dimming signals for the 3x3 display panel, enabling individual dimming of LEDs in a 3x3 display panel. The use of a multiplexer instead of diodes result in a better sensing of the voltages since the voltage drop across the diode changes with respect to diode junction temperature. Additionally, since the current and voltages of the LEDs were controlled separately, a better overall system performance and stability was achieved. The efficiency optimization algorithm in the microcontroller maintained a minimum drive voltage for the RGB pixels and yielded efficiencies of 85.6 %, 93.3 %, and 91.1 % for the red, green, and blue LEDs, respectively, at the maximum output drive current of 180 mA. A MATLAB SIMULINK model of the LED driver of each color was designed and results are

shown in the simulation result section of this chapter. The response of drivers for each color due to line voltage and load change, and during startup and turnoff were verified experimentally.

Chapter 4 introduces a new LED driver for LCD backlight using an analog/digital controller to maintain a minimum drive voltage across the LED strings. A Texas Instruments TL594 PWM controller was used to provide the worst-case drive voltage to the LED strings, and a PIC18F4431 microcontroller was used to provide the necessary voltage at the feedback summing point in order to change the drive voltage of the LED strings to its minimum. An external PSPWM dimming interface was implemented using a PIC18F4431 microcontroller to improve the dimming of the LED strings. Additionally, a mathematical model of the complete system was developed using block diagrams. Since there exists an analog and digital control loop, the digital control loop was changed into its analog equivalent using the multirate sampling algorithm [52]-[53]. Additionally, since the current and voltages of the LEDs were controlled separately, it yielded a better overall system performance and stability. An efficiency optimization and an operation routine were implemented in the digital controller to improve the efficiency of the LED drivers by 21 % at the maximum output drive current of 400 mA. A MATLAB SIMULINK model of the LED driver system was designed and results are shown in the simulation result section of this chapter. The response of drivers to line voltage and load change, and initialization of the digital control loop were verified experimentally.

Chapter 5 describes the design of a LED driver IC for driving five green LED strings using a multiplexer to sense the voltage drop across the current controllers. A TMS320F28335 DSP along with the fabricated IC was used to control the drive voltage of the LED strings. An external analog dimming interface was implemented with a digital potentiometer to dim the LED strings. Similar to the design of RGB drivers for LED display panels, a multiplexer is used for

sensing the voltages instead of diodes and separate control circuitry is used for current and drive voltage of the LED strings. An efficiency improvement of 11 %, at the maximum output drive current of 100 mA was demonstrated. The response of the driver due to line voltage, and during startup was verified experimentally.

6.2 CONTRIBUTION OF THIS RESEARCH

The contributions of this dissertation are:

1) A new digitally controlled RGB driver was developed for a 3x3 display panel. Multiplexer instead of diode are used for sensing of the voltages leading to a better efficiency of the driver. In order to maintain minimum drive voltage across the output LED load, the proposed driver implemented an efficiency optimization mode in the digital controller, leading to a reduction of losses in the current controllers of the LED strings. In order to maintain the minimum drive voltage across the LED strings, an operation mode was implemented which processes the error voltage using a PID controller to generate the correct PWM signal for the switches of the SMPC. Additionally, an analog dimming interface is able to dim each LEDs of each color individually leading to generation of variety of colors. Efficiencies of 85.6 %, 93.3 %, and 91.1 % for the red, green, and blue LEDs, respectively, at the maximum output drive current of 180 mA.
2) A novel LED driver using an analog/digital controller was developed for LCD backlight applications. An analog controller was used to control the output voltage of the SMPC while a digital controller was used to achieve the minimum drive voltage across the output LED strings. As such, the LED driver has both an analog controller as well as a

digital controller. The analog controller provides the worst-case drive voltage to the LED strings, and thus, by implementing an efficiency optimization mode in the digital control loop, the LED loads were driven by minimum drive voltage to maintain regulation in the LED strings. In order to maintain the minimum drive voltage, an operation mode is also implemented in the digital control loop to process the error voltage using a PID controller to generate the correct PWM signal for the switches of the SMPC. Additionally, PSPWM dimming was implemented in order to improve the system efficiency. A mathematical model was developed for this two loop system. An efficiency of 89 % at the rated output drive current of 400mA was observed.

6.3 FUTURE WORK

The following are recommendations for future work:

(1) Implement adaptive control for the RGB drivers for 3x3 display panel. This will lead to a faster transient response.

(2) Implement a PWM dimming for the RGB drivers for 3x3 display panel. This will lead to linear change of brightness for the LEDs. However, it would be difficult to generate 27 different PWM signals for the 3x3 display panel.

(3) Implement adaptive control in the digital control loop for the LED drivers for LCD backlight. This will lead to a faster transient response.

(4) Integrate dimming interface in the LED driver IC leading to reduction of size of the PCB of the LED driver system.

REFERENCES

[1] M. Doshi, "Power control architectures for cold cathode fluorescent lamp and light emitting diode based light sources," Ph.D. dissertation, Dept. Elect. Eng., Univ. Colorado, Boulder, 2010.

[2] Annual Energy Review (AER), Energy Inform. Admin. (EIA): Washington DC, USA, DOE/EIA-0384(2008), 2009.

[3] J. Hasan, S. S. Ang, T. Dhayagude, and D. Sangam, "A novel analog/digital LED driver controller," to be submitted for publication.

[4] J. Hasan and S. S. Ang, "A high-efficiency digitally controlled RGB-driver for LED pixels," *IEEE Trans. Ind. Appl.*, Vol. 47, No. 6, pp. 2422-2429, Nov-Dec 2011.

[5] W. Chen and S. Y. R. Hui, "A dimmable light-emitting diode (LED) driver with mag-amp postregulators for multistring applications," *IEEE Trans. Power Electron.,* Vol. 26, No. 6, pp. 1714-1722, Jun 2011.

[6] Y. Hu and M. M. Jovanovic, "A novel LED driver with adaptive drive voltage," *Proc. IEEE Applied Power Electronics Conf.,* 2008, pp. 565-571.

[7] Y. Hu, and M. M. Jovanovic, "LED driver with self-adaptive drive voltage," *IEEE Trans. Power Electron.,* Vol. 23, No. 6, pp. 3116-3125, Nov 2008.

[8] J. Hasan, D. H. Nguyen, and S. S. Ang, "A RGB-driver for LED display panels," *Proc. IEEE Applied Power Electronics Conf.,* 2010, pp. 750-754.

[9] S. S. Ang and A. Oliva, *Power-Switching Converters,* 2nd ed. Boca Raton, FL: CRC Press, 2005.

[10] K. Kano, *Semiconductor Devices,* Upper Saddle River, NJ: Prentice-Hall, 1998.

[11] "Datasheet: 1-Watt SMD 6mm LED (OVSPW1BCR4)," *TT electronics*, 2008.

[12] L. Guo, "Design and implementation of digital controllers for buck and boost converters using linear and nonlinear control methods," Ph.D. dissertation, Dept. Elect. Eng., Auburn Univ., Auburn, AL, 2006.

[13] J. Falin. (2008). Compensating and measuring the control loop of a high-power LED driver. *Analog Applications Journal* [Online]. pp.14-17. Available: http://www.focus.ti.com/lit/an/slyt308/slyt308.pdf

[14] J. Hasan, "Investigation of constant-current source," M.S. thesis, Dept. Elect. Eng., Univ. Arkansas, Fayetteville, 2006.

[15] M. Doshi and R. Zane, "Digital architecture for driving large LED arrays with dynamic bus voltage regulation and phase shifted PWM," *Proc. IEEE Applied Power Electronics Conf.*, 2007, pp. 287-293.

[16] M. Doshi and R. Zane, "Control of solid-state lamps using a multiphase pulsewidth modulation technique," *IEEE Trans. on Power Electron.,* Vol. 25, No. 7, pp 1894-1904, Jul 2010.

[17] H. J. Chiu, Y. K. Lo, J. T. Chen, S. J. Cheng, C. Y. Lin, and S. C. Mou, "A high efficiency dimmable LED driver for low-power lighting applications", *IEEE Trans. on Ind. Electron.,* Vol. 57, No. 2, pp. 735-743, Feb 2010.

[18] M. Dyble, N. Narendran, A. Bierman, and T. Klein, "Impact of dimming white LEDs: chromaticity shifts due to different dimming methods", *Proc. Fifth Int. Conf. on Solid State Lighting, 2005.*

[19] *The IESNA lighting handbook*, 9th ed., Illuminating Engineering Society of North America, New York, 2000.

[20] Y. Gu, N. Narendran, T. Dong, and H. Wu, "Spectral and luminous efficacy change of high-power LEDs under different dimming methods", *Proc. Sixth Int. Conf. on Solid State Lighting, 2006.*

[21] M. Day. (2004). LED-driver considerations. *Analog Applications Journal* [Online]. pp. 14-17. Available: http://www.focus.ti.com/lit/ab/slyt084/slyt084.pdf

[22] J. Falin. (2010, Jan.). How to Dim a WLED Backlight Driver. *The Magazine Of Solid-State Lighting*. Available: http//www.ledjoutnal.com/eprints/TI_Jan10.html

[23] "Pulse width modulation (PWM) vs. analog dimming of LEDs," *Application Note*, *aimtec*, 2011.

[24] C. L. King and E. Haile, "Interfacing microchip's MCP41XXX and MCP42XXX digital potentiometers to a PICmicro® microcontroller," *Application Note 746*, *Microchip Technology Incorporated*, 2001.

[25] C. L. King and E. Haile, "Communicating with daisy chained MCP42XXX digital potentiometers," *Application Note 747*, *Microchip Technology Incorporated*, 2001.

[26] "Datasheet: MCP41XXX/42XXX – Single/Dual digital potentiometer with SPI interface," *Microchip Technology Incorporated*, 2003.

[27] "Datasheet: PIC18F2331/2431/4331/4431 – 28/40/44-pin enhanced flash microcontrollers with nanoWatt technology, high performance PWM, and A/D," *Microchip Technology Incorporated*, 2003.

[28] "Datasheet: +5 V-powered, multichannel RS-232 drivers/receivers," *Maxim Integrated Products*.

[29] "Daisy-chaining SPI devices," *Application Note 3947*, *Maxim Integrated Products*, 2006.

[30] V. Vorperian, "Simplified analysis of PWM converters using model of PWM switch. Part I: Continuous conduction mode," *IEEE Trans. Aerosp. Electron. Syst.*, Vol. 26, No. 3, pp. 490-496, May 1990.

[31] L. Guo, J. Y. Hung, and R. M. Nelms, "Evaluation of DSP-based PID and fuzzy controllers for DC-DC converters," *IEEE Trans. Ind. Electron.*, Vol. 56, No. 6, pp. 2237-2248, Jun 2009.

[32] L. Guo, "Implementation of digital PID controllers for DC-DC converters using digital signal processors," *IEEE Int. Electro/Information Technology*, 2007, pp. 306-311.

[33] M. K. Zamierczuk, R. C. Cravens II, and A. Reatti, "Closed-loop impedance of PWM buck-derived DC-DC converters," *IEEE Int. Symp. on Circuits and Systems*, 1994, pp. 61-64.

[34] P. Narendra, "Disturbance accommodation control of a buck converter using a fixed point digital signal processor," M.S. thesis, Dept. Elect. Eng., Univ. Arkansas, Fayetteville, 2007.

[35] Y. Duan and H. Jin, "Digital controller design for switchmode power converters," *Proc. IEEE Applied Power Electronics Conf.*, 1999, pp. 967-973.

[36] T. W. Martin and S. S. Ang, "Digital control for switching converters," *Proc. IEEE Int. Symp. on Industrial Electronics*, 1995, pp. 480-484.

[37] A. Torres, J. Garcia, M.R. Secades, A.J. Calleja, and J. Ribas, "Advancing Towards Digital Control for Low Cost High Power LED Drivers," *Proc. IEEE Int. Symp. on Industrial Electronics,* 2007, pp. 3053-3056.

[38] S. Choudhury, "Digital control design and implementation of a DSP based high-frequency DC-DC switching power converter," *Application Note, Texas Instruments Incorporated*.

[39] "Datasheet: Aluminum electrolytic capacitors/SU (Bipolar)," *Panasonic*, 2008.

[40] "Datasheet: Switchmode/high frequency inductors," *Triad Magnetics*.

[41] "Datasheet: 6.5 A, 200 V, 0.8 Ω, P-channel power MOSFETs," *Fairchild Semiconductor*, 2002.

[42] "Datasheet: 5.0 mm x 6.0 mm surface mount LED lamp," *Kingbright*, 2008.

[43] J. Murphree, B. Brzezinski, and J. Parker, "Using a fixed-point digital signal processor as a PID controller," *Proc. of the American Society for Engineering Education Annual Conf.*, 2002.

[44] R. Dorf and R. Bishop, *Modern Control Systems*, 9th ed. Upper Saddle River, NJ: Prentice-hall, 2000.

[45] R. Erickson, "Converter system modeling via MATLAB/SIMULINK," class notes for ECE 5797, Dept. Elect. Eng., Univ. Colorado, Boulder.

[46] R. Erickson and D. Maksimovic, *Fundamentals of Power Electronics*, 2nd ed. New York, NY: Springer, 2001.

[47] R. R. Boudreaux, R. M. Nelms, and J. Y. Hung, "Simulation and modeling of a DC-DC converter controlled by an 8-bit microcontroller," *Proc. IEEE Applied Power Electronics Conf.*, 1997, pp. 963-969.

[48] M. Shrud, A. Kharaz, A. Ashur, A. Faris, and M. Benamar, "Analysis and simulation of automotive interleaved buck converter," *World Academy of Science, Engineering and Technology*, 2010, pp. 10-17.

[49] "LED screen power consumption explained," *Application Note, Vegas LED Screens.*

[50] Y. K. Lo, K. H. Wu, K. J. Pai, and H. J. Chiu, "Design and implementation of RGB LED drivers for LCD backlight modules", *IEEE Trans. on Ind. Electron.*, Vol. 56, No. 12, pp. 4862-4871, Dec 2009.

[51] C. C. Chen, C. Y. Wu, Y. M. Chen, and T. F. Wu, "Sequential color LED backlight driving system for LCD panels," *IEEE Trans. Power Electron.*, Vol. 22, No. 3, pp. 919-925, May 2007.

[52] M. Doshi and R. Zane, "Reconfigurable and fault tolerant digital phase shifted modulator for luminance control of LED light sources," *IEEE Electronics Specialists Conf.*, 2008, pp. 4185-4191.

[53] V. Feliu, "A transformation algorithm for estimating system Laplace transform from sampled data," *IEEE Trans. on Syst. Man and Cybern.*, Vol. SMC-16, No. 1, pp. 168-173, Feb 1986.

[54] R. Aracil, A. Jimenez, and V. Feliu, "Multirate sampling technique in digital control systems simulation," *IEEE Transactions on Syst. Man Cybern.*, Vol. SMC-14, No. 5, pp. 776-780, Sept 1984.

[55] K. Ogata, *System* Dynamics, 4th ed. Upper Saddle River, NJ: Prentice-hall, 2004.

[56] "Datasheet: TL594 Pulse-width modulation control circuit," *Texas Instruments Incorporated*, 2007.

[57] "Designing switching voltage regulators with the TL494," *Application Note*, *Texas Instruments Incorporated*, 2005.

[58] "Datasheet: RL622 series – radial lead RF choke," *J. W. Miller*, 2003.

[59] D. Mattingly, "Designing stable compensation networks for single phase voltage mode regulators," *Application Note*, *Intersil Incorporated*, 2003.

[60] "Design type III compensation network for voltage mode step-down converters," *Application Note 129*, *Analogic Tech Incorporated*, 2009.

[61] A. Sedra and K. Smith, *Microelectronic Circuits*, 5th ed. New York, NY: Oxford University Press, 2003.

[62] S. Mollet. (2001, Jan). Analyze LED characteristics with PSPICE. *EDN Magazine*. Available: http://www.edn.com/design/analog/4357273/Analyze-LED-characteristics-with-PSpice

[63] Y. Hu, M. M. Jovanovic, and C. Weng, "Driver that efficiently regulates current in a plurality of LED strings," U.S. Patent 2009/0187925 A1, Jul. 23, 2009.

[64] D. H. Nguyen, J. Hasan, and S. S. Ang, "A built-in-self-test high-current LED driver," in *Proc. IEEE Int. Conf. on ASIC*, 2009, pp. 340-343.

[65] B. Beckham, "Design and test of the electronic interfaces for a multi-channel neural microprobe and electrochemical headstages," M.S. thesis, Dept. Elect. Eng., Univ. Arkansas, Fayetteville, 2000.

[66] J. Zhang, "Advanced pulse width modulation controller ICs for buck DC-DC converters," Ph.D. dissertation, Univ. California, Berkeley, 2006.

[67] E. Lau and N. Wang, "Single-ended 2-stage OPAMP," unpublished.

[68] P. Allen and D. Holberg, *CMOS Analog Circuit Design*, 2nd ed. New York, NY: Oxford University Press, 2002.

[69] A. Chan, "Switch capacitor and characterization for neural electrochemistry," M.S. thesis, Dept. Elect. Eng., Univ. Arkansas, Fayetteville, 2001.

[70] M. R. Hoque, "Analog circuit blocks for power management," Ph.D. dissertation, Dept. Elect. Eng., Univ. Arkansas, Fayetteville, 2008.

APPENDIX A

M – File for the Red LEDs

```
Vout=2.3;
Vin=12;
T=1*10^-5;
D=Vout/Vin;
Gain=Vout/D;
Iout=0.180;
L=75*10^-6;
Cout=100*10^-6;
Rc=0.150;
RL=0.07;
Rload=11.9;
num =[Rc*Cout 1];
den=[L*Cout*((Rload+Rc)/(Rload+RL))
Rc*Cout+((Rload*RL)/(Rload+RL))*Cout+L/(Rload+RL)  1];
sys=Gain*tf(num,den)
```

M – File for the Green and Blue LEDs

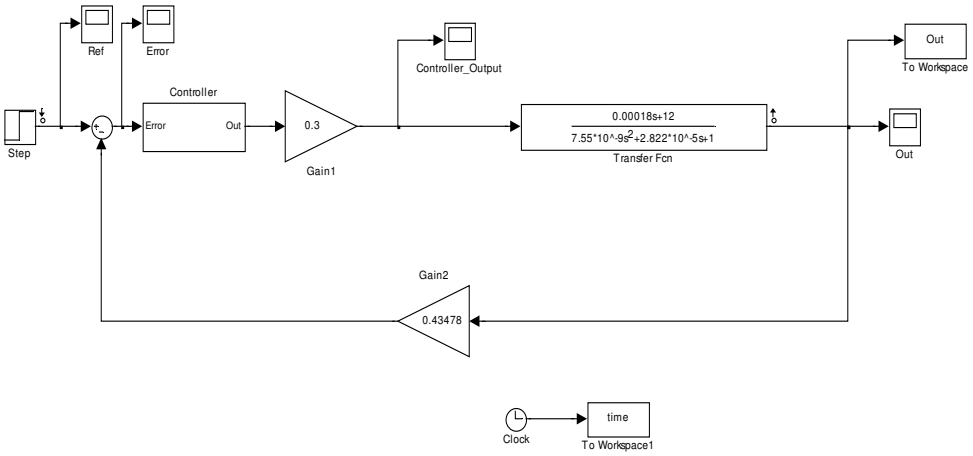
```
Vout=3.7;
Vin=12;
T=1*10^-5;
D=Vout/Vin;
Gain=Vout/D;
Iout=0.180;
L=75*10^-6;
Cout=100*10^-6;
Rc=0.15;
RL=0.07;
Rload=19.1;
num =[Rc*Cout 1];
den=[L*Cout*((Rload+Rc)/(Rload+RL))
Rc*Cout+((Rload*RL)/(Rload+RL))*Cout+L/(Rload+RL)  1];
sys=Gain*tf(num,den)
```

M – File for the Bode Plot Linearization

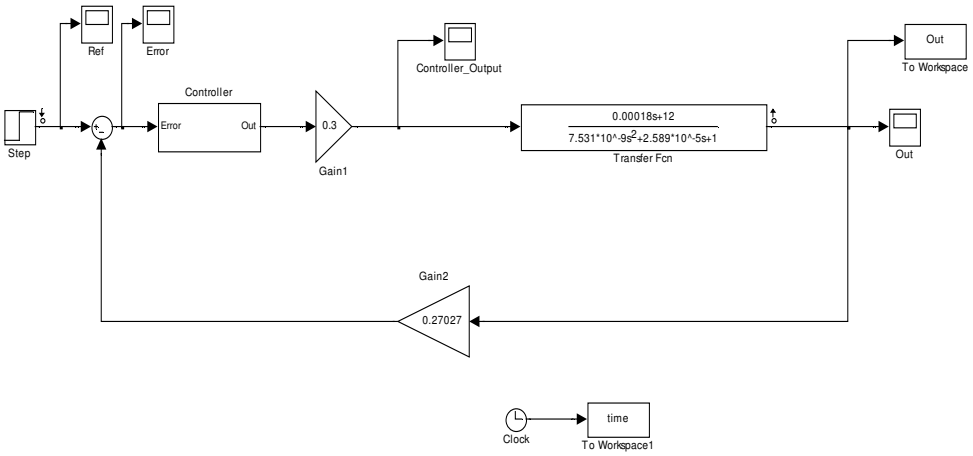
```
mdl = 'MATH_MODEL_Open_Loop_GREEN_LED_Driver';
mdl_1='MATH_MODEL_Closed_Loop_GREEN_LED_Driver';
io = getlinio(mdl); % get i/o signals of mdl?
op = operspec(mdl);
```

```
op = findop(mdl,op); % calculate model operating point?
lin = linearize(mdl,op,io);
io_1 = getlinio(mdl_1); % get i/o signals of mdl?
op_1 = operspec(mdl_1);
op_1 = findop(mdl_1,op_1); % calculate model operating point?
lin_1 = linearize(mdl_1,op_1,io_1);
```

Closed loop model for red LED



Closed loop model for green LED



APPENDIX B

M – File for the White LEDs

```
Vin=24;
Vout=13;
Iout=0.400;
R=27.45;
D=Vout/Vin;
L=75*10^-6;
Cout=100*10^-6;
fs=200*10^3;
Rc=10*10^-3;
Rl=30.4*10^-3;
num=[Rc*Cout 1];
den=[L*Cout*(R+Rc)/(R+Rl) Rc*Cout+((R*Rl)/(R+Rl))*Cout+L/(R+Rl) 1];
Gain=Vout/D;
sys1=Gain*tf(num,den)
```

M- File for the LEDs

```
Dp=10;
Lp=5e-4;
Dn=18;
Ln=1e-3;
Na=1e17;
Nd=1e16;
A=2500e-6;
q=1.60e-19;
ni=1.50e10;
Vt=25e-3;
N=2.3299;
Rs=5.96743;
```

M – File for the Bode Plot Linearization

```
mdl = 'MATH_MODEL_Open_Loop_Buck';
mdl_1 = 'MATH_MODEL_Closed_Loop_Buck';
mdl =
'MATH_MODEL_FUSION_LED_DRIVER_WITH_SECOND_CONTROLLER_EQUAL_1';
mdl_1 =
'MATH_MODEL_FUSION_LED_DRIVER_WITH_SECOND_CONTROLLER_EQUAL_1_A'
;
```

```
mdl='MATH_MODEL_FUSION_LED_DRIVER_WITH_SECOND_CONTROLLER_ANALO
G';
mdl_1='MATH_MODEL_FUSION_LED_DRIVER_WITH_SECOND_CONTROLLER_ANA
LOG_1';
mdl='MATH_MODEL_FUSION_LED_DRIVER_WITH_SECOND_CONTROLLER_MATH';
mdl_1='MATH_MODEL_FUSION_LED_DRIVER_WITH_SECOND_CONTROLLER_MAT
H_1';
io_1 = getlinio(mdl_1);
op_1 = operspec(mdl_1);
op_1 = findop(mdl_1,op_1);
lin_1 = linearize(mdl_1,op_1,io_1);
io = getlinio(mdl);
op = operspec(mdl);
op = findop(mdl,op);
lin = linearize(mdl,op,io);
```

Transformation of Digital to Analog

$$G_C(z) = \frac{7.012z^2 - 11.82z + 5.008}{z - 1} \tag{7.1}$$

$$= \frac{7.012 - 11.82z^{-1} + 5.008z^{-2}}{z^{-1}(1 - z^{-1})} \tag{7.2}$$

Implementing Partial Fraction Expansion:

$$\frac{G_C(z)}{(1 - z^{-1})} = \frac{7.012 - 11.82z^{-1} + 5.008z^{-2}}{z^{-1}(1 - z^{-1})(1 - z^{-1})} \tag{7.3}$$

$$= \frac{K_1}{z^{-1}} + \frac{K_2}{1 - z^{-1}} + \frac{K_3}{(1 - z^{-1})^2} \tag{7.4}$$

From (7.4),

$$K_1 = \infty \tag{7.5}$$

$$K_2 = 9.1 \tag{7.6}$$

$$K_3 = 0.2 \tag{7.7}$$

Applying (7.5), (7.6), and (7.7) into (7.4),

$$= \frac{\infty}{z^{-1}} + \frac{9.1}{1-z^{-1}} + \frac{0.2}{(1-z^{-1})^2} \tag{7.8}$$

Calculating the poles using (4.22),

$$p_1 = 0, m_1 = 2, p_2 = \infty, m_2 = 1 \tag{7.9}$$

Therefore,

$$B_{1,1} = 0.2 + 9.1 = 9.3 \tag{7.10}$$

$$B_{1,2} = \frac{0.2}{20\mu} = 1 * 10^4 \tag{7.11}$$

Since,

$$C = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \tag{7.12}$$

$$B_{i,k} = \frac{(k-1)!}{T^{k-1}} \sum_{l=k}^{m_i} A_{i,l} c'_{k,l} \tag{7.13}$$

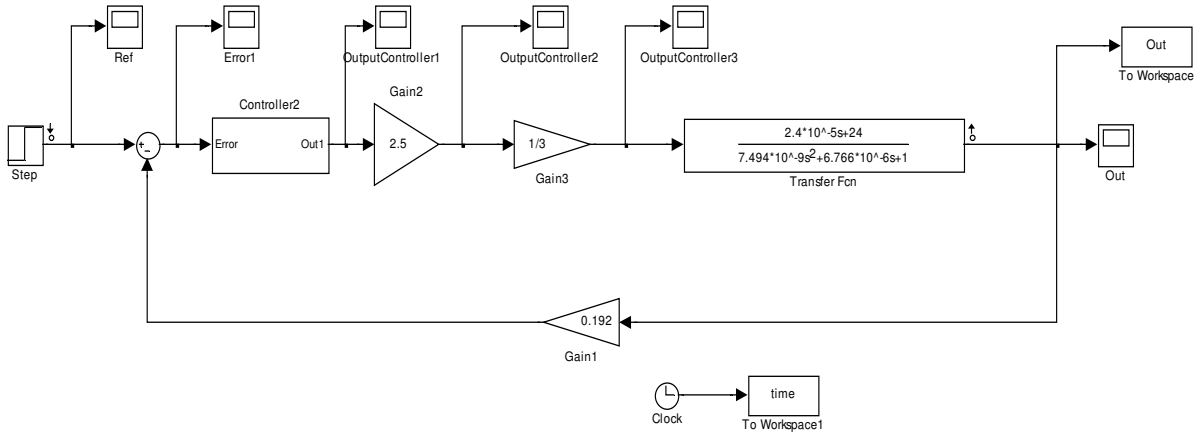
Therefore,

$$G_C(s) = s \sum_{i=1}^{N} \frac{A_{i,1}}{s-p_i} \tag{7.14}$$

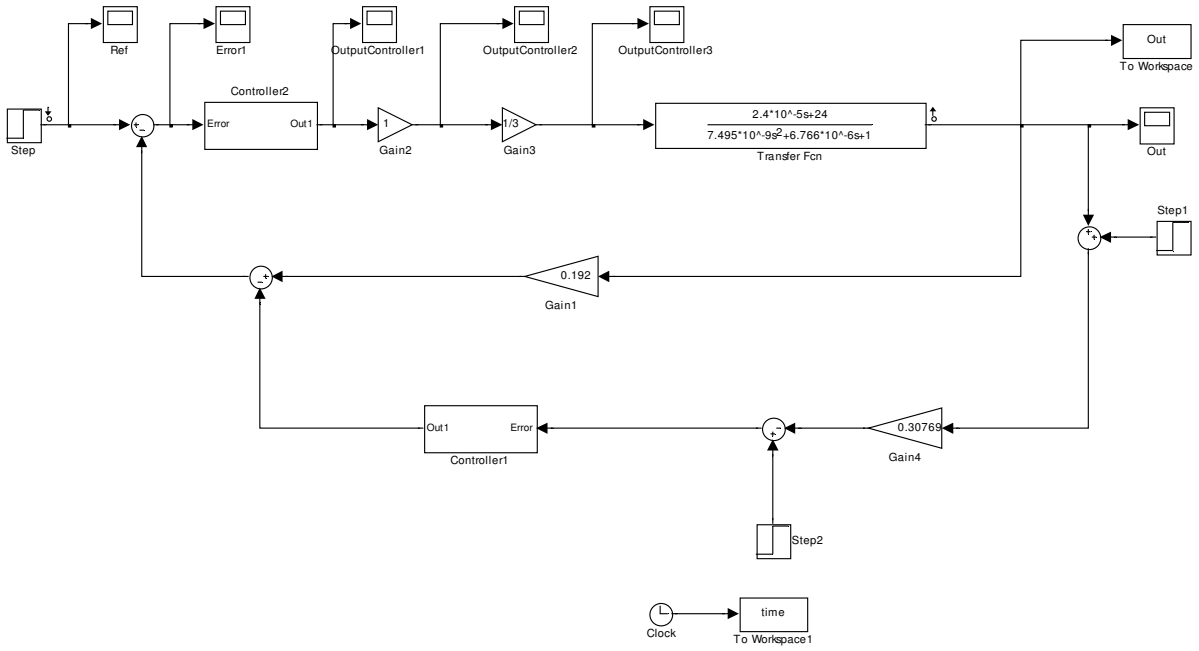
So,

$$G_C(s) = s\left(\frac{9.3}{s} + \frac{1*10^4}{s^2}\right) = 9.3 + \frac{1*10^4}{s} \tag{7.15}$$

Closed loop model for buck converter



Closed loop model for analog/digital controlled LED driver with $G_{C2}(s)=1$



APPENDIX C

ON SEMI C5 0.5µm Process Parameters

MOSIS PARAMETRIC TEST RESULTS

RUN: T06F VENDOR: AMI
TECHNOLOGY: SCN05 FEATURE
SIZE: 0.5 microns

INTRODUCTION: This report contains the lot average results obtained by MOSIS from measurements of MOSIS test structures on each wafer of this fabrication lot. SPICE parameters obtained from similar measurements on a selected wafer are also attached.

COMMENTS: American Microsystems, Inc. C5N.

TRANSISTOR PARAMETERS	W/L	N-CHANNEL	P-CHANNEL	UNITS
MINIMUM	3.0/0.6			
Vth		0.84	-0.93	volts
SHORT	20.0/0.6			
Idss		450	-256	uA/um
Vth		0.71	-0.90	volts
Vpt		10.0	-10.0	volts
WIDE	20.0/0.6			
Ids0		0.2	-0.1	pA/um
LARGE	50.0/50.0			
Vth		0.74	-0.93	volts
Vjbkd		11.1	-11.7	volts
Ijlk		-22.8	-6.0	pA
Gamma		0.46	0.56	V^0.5
K' (Uo*Cox/2)		57.0	-21.1	uA/V^2
Low-field Mobility		455.60	168.65	cm^2/V*s

COMMENTS: Poly bias varies with design technology. To account for mask and etch bias use the appropriate value for the parameter XL in your SPICE model card.

Design Technology	XL
SCN_SUBM (lambda=0.30)	0.00
AMI_C5	0.00
SCN (lambda=0.35)	-0.10

FOX TRANSISTORS	GATE	N+ACTIVE	P+ACTIVE	UNITS
Vth	Poly	>15.0	<-15.0	volts

PROCESS PARAMETERS	N+ACTV	P+ACTV	POLY	POLY2	MTL1	MTL2	MTL3	UNITS
Sheet Resistance	83.7	104.1	24.8	26.1	0.10	0.09	0.05	ohms/sq
Width Variation (measured - drawn)	-0.22	-0.22	-0.17	-0.22	0.04	-0.13	-0.09	microns
Contact Resistance	50.8	97.5	16.3	16.1	1.06	1.42		ohms
Gate Oxide Thickness	138							angstrom

PROCESS PARAMETERS	POLY_HR	N\PLY	N_WELL	UNITS
Sheet Resistance	1198.5	838	834	ohms/sq
Width Variation (measured - drawn)				microns
Contact Resistance				ohms

COMMENTS: N\POLY is N-well under polysilicon.

CAPACITANCE PARAMETERS	N+ACTV	P+ACTV	POLY	POLY2	MTL1	MTL2	MTL3	N_WELL	UNITS
Area (substrate)	425	726	87	40	22	14	39		aF/um^2
Area (N+active)			2504	60	26	18			aF/um^2
Area (P+active)			2441						aF/um^2
Area (poly)				829	55	22	14		aF/um^2
Area (poly2)					58				aF/um^2
Area (metal1)						32	17		aF/um^2
Area (metal2)							33		aF/um^2
Fringe (substrate)	347	249			78	64	47		aF/um
Fringe (poly)					65	45	34		aF/um
Fringe (metal1)						51	39		aF/um
Fringe (metal2)							53		aF/um

Overlap (N+active)		193	aF/um
Overlap (P+active)		220	aF/um

CIRCUIT PARAMETERS			UNITS
Inverters	K		
Vinv	1.0	2.18	volts
Vinv	1.5	2.45	volts
Vol (100 uA)	2.0	0.27	volts
Voh (100 uA)	2.0	4.75	volts
Vinv	2.0	2.62	volts
Gain	2.0	-23.49	
Ring Oscillator Freq.			
DIV256 (31-stg,5.0V)		104.40	MHz
Ring Oscillator Power			
DIV256 (31-stg,5.0V)		0.37	uW/MHz/gate

COMMENTS: SUBMICRON

T06F SPICE BSIM3 VERSION 3.1 PARAMETERS

SPICE 3f5 Level 8, Star-HSPICE Level 49, UTMOST Level 8

```
* DATE: Aug 25/00
* LOT: t06f                WAF: 02
* Temperature_parameters=Default
.MODEL CMOSN NMOS ( LEVEL   = 49
+VERSION = 3.1             TNOM    = 27              TOX     = 1.38E-8
+XJ      = 1.5E-7          NCH     = 1.7E17          VTH0    = 0.6979357
+K1      = 0.83086         K2      = -0.0896917      K3      = 33.4604942
+K3B     = -9.6210089      W0      = 1E-8            NLX     = 1E-9
+DVT0W   = 0               DVT1W   = 0               DVT2W   = 0
+DVT0    = 4.0512525       DVT1    = 0.4240224       DVT2    = -0.0741776
+U0      = 474.2098242     UA      = 1.496746E-11    UB      = 1.728644E-18
+UC      = 1.399262E-11    VSAT    = 1.395377E5      A0      = 0.5106363
+AGS     = 0.1253744       B0      = 2.941185E-6     B1      = 5E-6
+KETA    = -4.167955E-3    A1      = 1.8177E-5       A2      = 0.474759
+RDSW    = 1.697988E3      PRWG    = 9.060644E-3     PRWB    = 0.0463168
+WR      = 1               WINT    = 2.727018E-7     LINT    = 3.801001E-8
+XL      = 0               XW      = 0               DWG     = -2.18053E-8
+DWB     = 4.328818E-8     VOFF    = -0.0167929      NFACTOR = 0.806307
+CIT     = 0               CDSC    = 2.4E-4          CDSCD   = 0
+CDSCB   = 0               ETA0    = 0.01            ETAB    = -1.61673E-3
+DSUB    = 0.2411287       PCLM    = 2.1038693       PDIBLC1 = -0.447298
+PDIBLC2 = 2.445322E-3     PDIBLCB = -0.0554115      DROUT   = 0.6478036
```

```
+PSCBE1  = 5.276894E8      PSCBE2  = 3.152432E-5     PVAG    = 0
+DELTA   = 0.01            RSH     = 83.7           MOBMOD  = 1
+PRT     = 0               UTE     = -1.5           KT1     = -0.11
+KT1L    = 0               KT2     = 0.022          UA1     = 4.31E-9
+UB1     = -7.61E-18       UC1     = -5.6E-11       AT      = 3.3E4
+WL      = 0               WLN     = 1              WW      = 0
+WWN     = 1               WWL     = -6.554E-20     LL      = 0
+LLN     = 1               LW      = 0              LWN     = 1
+LWL     = -9.461E-20      CAPMOD  = 2              XPART   = 0.4
+CGDO    = 1.93E-10        CGSO    = 1.93E-10       CGBO    = 1E-9
+CJ      = 4.222686E-4     PB      = 0.9843774      MJ      = 0.4434249
+CJSW    = 3.588532E-10    PBSW    = 0.1            MJSW    = 0.1198579
+CF      = 0               PVTH0   = -0.1321152     PRDSW   = -36.36825
+PK2     = -2.027638E-3    WKETA   = -0.0214184     LKETA   = 6.0913E-3
)
*
.MODEL CMOSP PMOS ( LEVEL   = 49
+VERSION = 3.1             TNOM    = 27             TOX     = 1.38E-8
+XJ      = 1.5E-7          NCH     = 1.7E17         VTH0    = -0.9152586
+K1      = 0.5392919       K2      = 4.293128E-3     K3      = 5.8480432
+K3B     = -1.6750019      W0      = 1E-8           NLX     = 3.47372E-9
+DVT0W   = 0               DVT1W   = 0              DVT2W   = 0
+DVT0    = 3.5479658       DVT1    = 0.540603       DVT2    = -0.0551904
+U0      = 253.9591841     UA      = 3.811025E-9    UB      = 1E-21
+UC      = -5.63355E-11    VSAT    = 2E5            A0      = 0.7657245
+AGS     = 0.156468        B0      = 1.448763E-6    B1      = 5E-6
+KETA    = -1.771044E-3    A1      = 0              A2      = 0.3151316
+RDSW    = 2.939291E3      PRWG    = -0.0648976     PRWB    = -0.035143
+WR      = 1               WINT    = 3.335179E-7    LINT    = 3.48276E-8
+XL      = 0               XW      = 0              DWG     = -3.1747E-8
+DWB     = 8.766065E-9     VOFF    = -0.0621497     NFACTOR = 1.00210
+CIT     = 0               CDSC    = 2.4E-4         CDSCD   = 0
+CDSCB   = 0               ETA0    = 8.68072E-6     ETAB    = -3.8610E-5
+DSUB    = 0.0174888       PCLM    = 2.249572       PDIBLC1 = 0.193697
+PDIBLC2 = 1.709015E-3     PDIBLCB = -0.1           DROUT   = 0.396097
+PSCBE1  = 5.20778E9       PSCBE2  = 5.104962E-10   PVAG    = 0.9840557
+DELTA   = 0.01            RSH     = 104.1          MOBMOD  = 1
+PRT     = 0               UTE     = -1.5           KT1     = -0.11
+KT1L    = 0               KT2     = 0.022          UA1     = 4.31E-9
+UB1     = -7.61E-18       UC1     = -5.6E-11       AT      = 3.3E4
+WL      = 0               WLN     = 1              WW      = 0
+WWN     = 1               WWL     = -1.205E-20     LL      = 0
+LLN     = 1               LW      = 0              LWN     = 1
+LWL     = 6.268E-21       CAPMOD  = 2              XPART   = 0.4
+CGDO    = 2.2E-10         CGSO    = 2.2E-10        CGBO    = 1E-9
+CJ      = 7.213884E-4     PB      = 0.9586435      MJ      = 0.4967202
```

+CJSW = 2.620315E-10 PBSW = 0.99 MJSW = 0.2948291
+CF = 0 PVTH0 = 5.98016E-3 PRDSW = 14.85984
+PK2 = 3.73981E-3 WKETA = -6.860319E-4 LKETA = 5.737E-4
)
*

Netlist of the designed OPAMP

**** 10/17/12 15:33:09 ******* PSpice 16.3.0 (June 2009) ****** ID# 0 ********

** Profile: "OPAMP_OUTPUTVOLTAGESWING_1-test9" [
C:\Users\jhasan\Desktop\LED_TV_Project_IC\OPAMP\OPAMP_DESIGN\opamp-pspicefiles\opa

**** CIRCUIT DESCRIPTION

****

** Creating circuit file "test9.cir"
** WARNING: THIS AUTOMATICALLY GENERATED FILE MAY BE OVERWRITTEN BY SUBSEQUENT SIMULATIONS

*Libraries:
* Profile Libraries :
* Local Libraries :
.LIB "../../../opamp-pspicefiles/opamp.lib"
* From [PSPICE NETLIST] section of C:\Cadence\SPB_16.3\tools\PSpice\PSpice.ini file:
.lib "C:\Users\jhasan\Desktop\Fusion\Pspice_File\Synchronous_Buck_Converter_PID\TEST-PSpiceFiles\test.lib"
.lib "C:\Cadence\SPB_16.3\tools\pspice\library\nom.lib"

*Analysis directives:
.DC LIN V_V2 -5 5 0.1
.PROBE V(alias(*)) I(alias(*)) W(alias(*)) D(alias(*)) NOISE(alias(*))
.INC "..\OPAMP_OUTPUTVOLTAGESWING_1.net"

**** INCLUDING OPAMP_OUTPUTVOLTAGESWING_1.net ****
* source OPAMP
.EXTERNAL OUTPUT VDD
V_V1 VDD 0 5Vdc
C_Cload 0 VOUT 6pF TC=0,0
M_OPAMP_M10 OPAMP_N55219 OPAMP_N55091 0 0 nmos_transistor
+ L=15u
+ W=4.5u
+ AD=5.4p
+ AS=5.4p

```
+ PD=6.9u
+ PS=6.9u
M_OPAMP_M7        OPAMP_N80679 OPAMP_N55219 VDD VDD pmos_transistor
+ L=3u
+ W=12u
+ AD=14.4p
+ AS=27.6p
+ PD=2.4u
+ PS=4.6u
M_OPAMP_M4        OPAMP_N55841 N56582 OPAMP_N55733 OPAMP_N55733
+  pmos_transistor
+ L=3u
+ W=96u
+ AD=115.2p
+ AS=115.2p
+ PD=2.4u
+ PS=2.4u
M_OPAMP_M11       OPAMP_N55091 OPAMP_N55091 OPAMP_N55095 OPAMP_N55095
+  nmos_transistor
+ L=15u
+ W=4.5u
+ AD=5.4p
+ AS=5.4p
+ PD=6.9u
+ PS=6.9u
M_OPAMP_M14       VOUT OPAMP_N55841 0 0 nmos_transistor
+ L=3u
+ W=42u
+ AD=14.4p
+ AS=27.6p
+ PD=2.4u
+ PS=4.6u
M_OPAMP_M2        OPAMP_N55841 OPAMP_N55597 0 0 nmos_transistor
+ L=3u
+ W=48u
+ AD=57.6p
+ AS=110.4p
+ PD=2.4u
+ PS=4.6u
C_OPAMP_C1        OPAMP_N56035 OPAMP_N80679  2.2pF  TC=0,0
M_OPAMP_M6        OPAMP_N55219 OPAMP_N55219 VDD VDD pmos_transistor
+ L=3u
+ W=12u
+ AD=14.4p
+ AS=27.6p
+ PD=2.4u
```

```
+ PS=4.6u
M_OPAMP_M12        OPAMP_N55095 VDD 0 0 nmos_transistor
+ L=3u
+ W=12u
+ AD=14.4p
+ AS=27.6p
+ PD=2.4u
+ PS=4.6u
M_OPAMP_M3         OPAMP_N55597 N56100 OPAMP_N55733 OPAMP_N55733
+  pmos_transistor
+ L=3u
+ W=96u
+ AD=115.2p
+ AS=115.2p
+ PD=2.4u
+ PS=2.4u
R_OPAMP_R1         OPAMP_N55841 OPAMP_N56035  3k TC=0,0
M_OPAMP_M13        VDD OPAMP_N80679 VOUT 0 nmos_transistor
+ L=3u
+ W=12u
+ AD=14.4p
+ AS=14.4p
+ PD=2.4u
+ PS=2.4u
M_OPAMP_M8         OPAMP_N80679 OPAMP_N55841 0 0 nmos_transistor
+ L=3u
+ W=84u
+ AD=100.8p
+ AS=193.2p
+ PD=2.4u
+ PS=4.6u
M_OPAMP_M9         OPAMP_N55091 OPAMP_N55219 VDD VDD pmos_transistor
+ L=3u
+ W=12u
+ AD=14.4p
+ AS=27.6p
+ PD=2.4u
+ PS=4.6u
M_OPAMP_M1         OPAMP_N55597 OPAMP_N55597 0 0 nmos_transistor
+ L=3u
+ W=48u
+ AD=57.6p
+ AS=110.4p
+ PD=2.4u
+ PS=4.6u
M_OPAMP_M5         OPAMP_N55733 OPAMP_N55219 VDD VDD pmos_transistor
```

```
+ L=3u
+ W=12u
+ AD=14.4p
+ AS=27.6p
+ PD=2.4u
+ PS=4.6u
V_V2         N56582 0 0Vdc
R_R1         0 N56100  1Mega TC=0,0
R_R2         N56100 VOUT  1Mega TC=0,0

**** RESUMING test9.cir ****
.END

Model nmos_transistor: Using BSIM VERSION 3.1 or lower
Model pmos_transistor: Using BSIM VERSION 3.1 or lower**** 10/17/12 15:33:09 *******
PSpice 16.3.0 (June 2009) ****** ID# 0 ********

 ** Profile: "OPAMP_OUTPUTVOLTAGESWING_1-test9"  [
C:\Users\jhasan\Desktop\LED_TV_Project_IC\OPAMP\OPAMP_DESIGN\opamp-
pspicefiles\opa

 ****     MOSFET MODEL PARAMETERS

******************************************************************************

            nmos_transistor pmos_transistor
            NMOS            PMOS
  T_Measured   27              27
   T_Current   27              27
       LEVEL    7              7
           L  100.000000E-06  100.000000E-06
           W  100.000000E-06  100.000000E-06
         VTO     .697936        -.915259
          KP  150.136700E-06  150.136700E-06
       GAMMA    0              0
      LAMBDA    0              0
         RSH   83.7           104.1
          IS    1.000000E-15    1.000000E-15
          JS  100.000000E-06  100.000000E-06
          PB     .984377         .958644
        PBSW     .1              .99
          CJ  422.268600E-06  721.388400E-06
        CJSW  358.853200E-12  262.031500E-12
          MJ     .443425         .49672
        MJSW     .119858         .294829
        CGSO  193.000000E-12  220.000000E-12
```

```
  CGDO  193.000000E-12  220.000000E-12
  CGBO    1.000000E-09    1.000000E-09
   TOX   13.800000E-09   13.800000E-09
    XJ  150.000000E-09  150.000000E-09
 UCRIT   10.000000E+03   10.000000E+03
 DELTA     .01             .01
DIOMOD    2               2
    K1     .83086          .539292
    K2    -.089692         4.293128E-03
  LETA    0               0
  WETA    0               0
    U0  474.2098        253.9592
 XPART     .4              .4
  VTH0     .697936        -.915259
    K3   33.4605          5.848043
    W0   10.000000E-09   10.000000E-09
   NLX    1.000000E-09    3.473727E-09
  DVT0    4.051252        3.547966
  DVT1     .424022         .540603
    UA   14.967460E-12    3.811025E-09
    UB    1.728644E-18    1.000000E-21
    UC   13.992620E-12  -56.335500E-12
  VSAT  139.537700E+03  200.000000E+03
  RDSW    1.697988E+03    2.939291E+03
  VOFF    -.016793        -.06215
NFACTOR     .806307         1.002105
  PCLM    2.103869        2.249572
 PDIBL1    -.447298         .193698
 PDIBL2    2.445322E-03    1.709015E-03
 DROUT     .647804         .396098
 PSCBE1  527.689400E+06    5.207780E+09
 PSCBE2   31.524320E-06  510.496200E-12
    A0     .510636         .765724
    A1   18.177000E-06    0
    A2     .474759         .315132
 NPEAK  170.000000E+15  170.000000E+15
   LDD    0               0
  LITL   78.803550E-09   78.803550E-09
   UA1    4.310000E-09    4.310000E-09
   UB1   -7.610000E-18   -7.610000E-18
   UC1  -56.000000E-12  -56.000000E-12
  PVAG                    .984056
  KETA   -4.167955E-03   -1.771044E-03
  ETA0     .01            8.680720E-06
  ETAB   -1.616730E-03  -38.610980E-06
   K3B   -9.621009        -1.675002
```

```
      DVT2    -.074178        -.05519
      DSUB     .241129         .017489
     MOBMOD    1               1
       AGS     .125374         .156468
     DVT1W     0               0
     DVT2W     0               0
      PRWG     9.060644E-03    -.064898
      PRWB     .046317         -.035143
    PDIBLCB    -.055412        -.1
       DWG  -21.805390E-09  -31.747150E-09
       DWB   43.288180E-09    8.766065E-09
        B0    2.941185E-06    1.448763E-06
        B1    5.000000E-06    5.000000E-06
      LINT   38.010010E-09   34.827690E-09
       LWL  -94.610000E-21    6.268000E-21
      WINT  272.701800E-09  333.517900E-09
       WWL  -65.540000E-21  -12.050000E-21
       DLC   38.010010E-09   34.827690E-09
       DWC  272.701800E-09  333.517900E-09
        CF    0               0
      NOIA  100.000000E+18    9.900001E+18
      NOIB   50.000000E+03    2.400000E+03
      NOIC   -1.400000E-12    1.400000E-12
     LKETA    6.091325E-03  573.759400E-06
     WKETA    -.021418     -686.031900E-06
     PVTH0    -.132115        5.980160E-03
     PRDSW  -36.36826         14.85984
       PK2   -2.027638E-03    3.739810E-03
       VTM     .025864         .025864
    VERSION    3.1             3.1
     PBSWG     .1             .99
     MJSWG     .119858         .294829
     CJSWG  358.853200E-12  262.031500E-12

        JOB CONCLUDED

**** 10/17/12 15:33:09 ******* PSpice 16.3.0 (June 2009) ****** ID# 0 ********

 ** Profile: "OPAMP_OUTPUTVOLTAGESWING_1-test9"  [
C:\Users\jhasan\Desktop\LED_TV_Project_IC\OPAMP\OPAMP_DESIGN\opamp-
pspicefiles\opa

 ****     JOB STATISTICS SUMMARY

*******************************************************************************
  Total job time (using Solver 1)   =         .70
```