Adaptive Primary Side Control for a Wireless Power Transfer Optimization

by

Thilani Imanthika Dissanayake Bogoda

Submitted to the Department of Electrical Engineering and Computer Science in Partial Fulfillment of the Requirements for the Degree of Master of Engineering in Electrical Engineering and Computer Science at the Massachusetts Institute of Technology

May 25, 2012

Copyright 2012 Thilani Imanthika Dissanayake Bogoda. All rights reserved.

The author hereby grants to M.I.T. permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole and in part in any medium now known or hereafter created.

Author: 

Department of Electrical Engineering and Computer Science
May 25, 2012

Certified by: 

Eko T. Lisuwandi
Senior Design Engineer, Linear Technology
VI-A Thesis Supervisor

Certified by: 

David J. Perreault
Professor of Electrical Engineering
M.I.T. Thesis Supervisor

Accepted by: 

Prof. Dennis M. Freeman
Chairman, Masters of Engineering Thesis Committee
Adaptive Primary Side Control for a Wireless Power Transfer Optimization

by

Thilani Imanthika Dissanayake Bogoda

Submitted to the Department of Electrical Engineering and Computer Science

May 25, 2012

In Partial Fulfillment of the Requirements for the Degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

A resonant inductive wireless power transfer system, consisting of a primary (transmitter) circuit and secondary (receiver) circuit, was designed and implemented. This document also contains a novel indirect feedback method to optimize the power efficiency of a wireless transfer system. The indirect feedback method presented allows the primary circuit to adapt its power delivery to the power requirements of the secondary circuit without requiring a direct feedback signal from the secondary. Also presented are the results of the implementation of the indirect feedback method.

VI-A Thesis Supervisor: Eko T. Lisuwandi
Title: Senior Design Engineer, Linear Technology

Thesis Supervisor: David J. Perreault
Title: Professor of Electrical Engineering
Acknowledgements

I would like to thank the VI-A program and Sam Nork, Manager of Linear Technology Design Center, for giving me the opportunity to work on this project. I would further like to thank Trevor Barcelo for all his help given throughout this project (including his lab bench!). I would also like to acknowledge Rick Brewster for his ideas and suggestions which were a critical part of this thesis.

I would like to express my sincere gratitude to Eko Lisuwandi, my thesis advisor at Linear. This thesis would have died in its sleep if it weren’t for his brilliant advice and support (and his never-ending supply of humor). I enjoyed working with you, Eko.

I would also like to thank Professor David Perreault, my on-campus thesis advisor, for his suggestions and for helping me to review this document.

Joe Sousa for his wealth of knowledge and for the box of inductors and loops (a.k.a the box of goodies) he gave me on the 2\textsuperscript{nd} day of my internship. I got a head start on my project because of that box of goodies.

Ron Swinnich for making the impossible, possible. You have yet to disappoint me Ron.

Wendi Li and Taylor Barton for their constant support and encouragement. I was really lucky to have both of you as my TAs at MIT.

Deepali Ravel and Aseema Mohanty: meeting you both was the best part about coming to MIT. I cry myself to sleep sometimes when I think about the time we wasted watching all those chipmunk youtube videos. Despite my everyday statements to the contrary, you two are (seriously) very cool.

Frank Yaul for putting up with me for 5 years of classes at MIT. I’m going to miss working with you.

My parents, Sanath and Shirani Bogoda, for never getting mad at me for my first electrical engineering experience (you know what I’m talking about). Thank you for being there for me even though you both live many thousands of miles away from me. It’s a privilege to be your kid.

My sister, Imanthi Bogoda, for being my best friend, advisor, dictator (this list goes on). I have yet to meet someone who is as awesome as you are.
## Table of Contents

**Introduction** ........................................................................................................................... 11

**Chapter 1: Inductive Power Transfer System** ........................................................................... 14
  1.1 Inductive Power Transfer ........................................................................................................ 14
  1.2 Resonant Inductive Coupling: Transformer Model ................................................................. 16
  1.3 Transmitter and Receiver Coil Magnetics .............................................................................. 18
    1.3.1 Quality Factor .................................................................................................................... 18
    1.3.2 Optimum Coil Geometry .................................................................................................. 19
    1.3.3 Addition of Ferrite Material ............................................................................................. 20

**Chapter 2: Receiver Architecture** ............................................................................................ 23
  2.1 Receiver with Tuning Controller ............................................................................................ 26
    2.1.1 Different Topologies .......................................................................................................... 26
      2.1.1.1 Parallel LC Tank – Tuned to Detuned ....................................................................... 26
      2.1.1.2 Parallel-Series LC Tank – Detuned to Tuned ............................................................. 29
    2.1.2 Switching Techniques for the Tuning Controller .............................................................. 31
      2.1.2.1 Switching techniques: Linear ....................................................................................... 31
      2.1.2.2 Improved Switching Technique: Hysteresis ............................................................... 36
  2.2 Future work: Receiver with Buck Converter ......................................................................... 41

**Chapter 3: Transmitter Architecture** ..................................................................................... 42
  3.1 Cross-coupled Current Mode Class-D Power Amplifier ......................................................... 42
    3.1.1 Gate Drive ............................................................................................................................ 44
    3.1.2 Startup Requirements of the cross-coupled class-D amplifier ....................................... 52
  3.2 Royer Oscillator ...................................................................................................................... 54
    3.2.1 Base Drive .......................................................................................................................... 55
    3.2.2 Power Dissipation ............................................................................................................. 56
List of Figures

Figure 1-1: Typical Inductive Power Transfer System ............................................................... 14
Figure 1-2: Resonant Inductive Power Transfer System .......................................................... 16
Figure 1-3: Transformer Model ................................................................................................. 16
Figure 1-4: Simplified Transformer Model ................................................................................ 17
Figure 1-5: Transmitter and Receiver coil ................................................................................ 19
Figure 1-6: Maximum pickup vs number of loops .................................................................... 19
Figure 1-7: Transmitter (or Receiver) coil .................................................................................. 20
Figure 1-8: Magnetic field of a single loop with no ferrite material ......................................... 21
Figure 1-9: Magnetic field of a single loop with ferrite material ............................................... 21
Figure 1-10: Magnetic field of a single loop with a ferrite block larger than the loop ............... 21
Figure 2-1: Basic Receiver Circuit (with transmitter coil coupled) .......................................... 23
Figure 2-2: Complete Transformer Model .................................................................................. 24
Figure 2-3: Primary Circuit replaced with Thevenin Equivalent .............................................. 24
Figure 2-4: Simplified Receiver Circuit (with transmitter coil coupled) .................................... 25
Figure 2-5: Parallel LC Tank Receiver Circuit (with no transmitter coil coupled) ..................... 26
Figure 2-6: Impedance vs Frequency Curve .............................................................................. 27
Figure 2-7: Parallel Series LC tank receiver (with transmitter coil coupled) .............................. 29
Figure 2-8: Simplified Parallel Series LC tank receiver ............................................................ 29
Figure 2-9: Parallel Series LC tank (with detuning capacitor across tank capacitor) ................. 29
Figure 2-10: Tuning controller - linear switching ...................................................................... 31
Figure 2-11: Gate voltage of the detuning MOSFET ............................................................... 32
Figure 2-12: Receiver board with linear switching ...................................................................... 32
Figure 2-13: Tuning controller with stabilized linear switching .................................................. 33
Figure 2-14: Gate voltage of detuning MOSFET (stabilized linear switching) ......................... 34
Figure 2-15: Gain curve ............................................................................................................ 35
Figure 2-16: Gain Curve .......................................................................................................... 35
Figure 2-17: Gate voltage of detuning MOSFET ..................................................................... 36
Figure 2-18: Output ripple and the gate voltage superimposed .................................................... 37
Figure 2-19: Tuning controller with hysteretic switching ............................................................ 37
Figure 2-20: The test setup to compare power efficiencies of different tuning controllers ........... 38
Figure 2-21: Receiver coil (5.6uH) placed at the center of transmitter coil (22uH) ..................... 38
Figure 2-22: Modified hysteretic tuning controller with PWM .................................................. 39
Figure 2-23: Receiver circuit with hysteretic tuning controller (with PWM) ............................... 40
Figure 2-24: Gate voltage of the detuning MOSFET for hysteretic tuning controller with PWM (for 2 different loads) ......................................................................................................... 40
Figure 2-25: Receiver circuit with buck converter ..................................................................... 41
Figure 3-1: A typical transmitter circuit (with the receiver coupled) ......................................... 42
Figure 3-2: A Simplified Current Mode Class-D amplifier circuit .............................................. 43
Figure 3-3: Cross-coupled class-D amplifier (with zener and resistor) – version 1 .................... 43

8
Figure 3-4: Cross-coupled class-D amplifier (with capacitive divider) – version 2 ........................................... 62
Figure 3-5: Voltage on sides 'acn' & 'acp' of the primary LC tank (with version 2) ........................................ 46
Figure 3-6: Current in the primary coil (with version 2 circuit) ................................................................. 47
Figure 3-7: Actual setup of the cross-coupled class-D amplifier (version 3) ............................................. 47
Figure 3-8: Cross-coupled class-D amplifier with capacitive divider and level shifter (version 3) .............. 48
Figure 3-9: Voltage on the zener (version 3 circuit) ..................................................................................... 49
Figure 3-10: Gate voltage of M1 (blue) & drain voltage of M1 (red) ......................................................... 50
Figure 3-11: Gate Voltage of M2 (black) and drain voltage of M2 (purple) .............................................. 50
Figure 3-12: Gate voltage of M1 and M2 ................................................................................................. 51
Figure 3-13: Change in tank voltage's overlap time by varying V_ref ....................................................... 52
Figure 3-14: Resistance of a differential cross-coupled NMOS pair .......................................................... 53
Figure 3-15: Modified Royer ................................................................................................................. 54
Figure 3-16: Test setup to investigate the base drive .............................................................................. 56
Figure 3-17: Royer – board (primary coil not shown) .............................................................................. 57
Figure 3-18: Royer - voltage across the primary LC tank with no secondary circuit coupled (Vdc=10V, $V_{ac,primary} = 62V_{p-p}, f_0 = 106kHz$) .................................................................................. 57
Figure 3-19: Collector and Base voltage of left and right transistors in roayer (dark blue = base voltage of left NPN, light blue = collector voltage of left NPN, green = collector voltage of right NPN, pink = base voltage of right NPN) .................................................................................................................. 58
Figure 4-1: Overall Power Efficiency of the open loop system ................................................................. 60
Figure 4-2: Direct Feedback from Secondary to Primary .......................................................................... 61
Figure 4-3: bqTESLA - Resistive Modulation ........................................................................................ 62
Figure 4-4: bqTESLA - Capacitive Modulation ...................................................................................... 62
Figure 4-5: Supply Current vs. Supply Voltage when a resistor R1 is connected across a voltage source 64
Figure 4-6: Supply Current vs Supply Voltage Curve when load R2 is regulated by DC/DC converter 64
Figure 4-7: Supply Current vs Supply Voltage with dissipative resistor R2 ........................................... 65
Figure 4-8: Model of Resonant Inductive System (including parasitics) .................................................. 66
Figure 4-9: Primary Supply Current (rms) vs. Primary Supply Voltage (rms) ......................................... 67
Figure 4-10: Primary Supply Current vs. Primary Supply Voltage ......................................................... 68
Figure 5-1: Closed Loop Transmitter Setup ............................................................................................. 70
Figure 5-2: Closed Loop Transmitter with OPTR in more detail ............................................................. 71
Figure 5-3: Optimal Power Transfer Regulator - Circuit block diagram ............................................... 73
Figure 5-4: Simple Slope Detection Algorithm - initial sweeping ............................................................. 76
Figure 5-5: Simple Slope Detection Algorithm - dithering ................................................................. 76
Figure 5-6: Simple Slope Detection for multiple loads ................................................................. 76
Figure 5-7: Overview of State Diagrams ............................................................................................... 78
Figure 5-8: Control SM ......................................................................................................................... 79
Figure 5-9: Sweep/Dither SM .............................................................................................................. 80
Figure 5-10: D/A Converter SM ........................................................................................................... 81
Figure 5-11: Delay A/D Converter ....................................................................................................... 82
Figure 5-12: A/D Converter SM .......................................................................................................... 83
Figure 6-1: Closed Loop Transmitter - OPTR and DC/DC Converter ....................................................... 85
List of Tables

Table 1: Comparison of power efficiencies with different tuning controllers .................. 37

Table 2: Effect of Capacitive Divider Ratio.................................................................. 43

Table 3: Effect of Base Drive on Power Efficiency and maximum secondary load .......... 53
Power and signals can be transferred electromagnetically either via direct electrical connections in a closed circuit or wirelessly. Since the early days of electromagnetism, methods to transfer signals wirelessly over large distance have evolved rapidly. However, development of wireless power transmission systems (across large air gaps) is lagging behind the progress in wireless communication even though Nikola Tesla demonstrated as early as 1900s that energy can be transferred wirelessly [1]. This is mostly due to the huge power losses associated with transporting energy over long distances.

Wireless transfer of energy can be achieved by non-radiative methods (near-field) or by radiative methods (far-field) such as omni-directional antennas and lasers. There are several drawbacks in using radiative methods to transfer energy: most of the energy is lost in free space and/or requires direct line of sight with the object. However, with non-radiative modes such as inductive power transfer or capacitive power transfer, radiative losses are reduced because the magnetic field or the electric field is localized.

The basic concept behind inductive power transfer is equivalent to the working of a traditional transformer except for the difference in coupling factor. In a transformer, the primary and secondary coils are tightly coupled with the coupling factor approximately equal to 1. But in an inductive power transfer system, the coupling factors can range from 0.1 to 0.6 depending on the separation between the primary and secondary coils. Resonant inductive coupling, which is inductive coupling with resonance added to the equation, is more suitable for transferring energy in loosely coupled systems [2], [3]. Inductive coupling and resonant inductively coupling are explained in more detail in chapter 1.

One main drawback with inductive coupling is the inability to transfer power through metal objects: when metal objects are placed in a magnetic field, significant amount of power is dissipated due to the formation of eddy currents in these objects. However, in capacitive power transfer this drawback is not present because instead of a distributed magnetic field, a
distributed electric field links the plates of a capacitor which are separated by the air gap [4]. One of the capacitor plates is free and can be used to transfer power to a movable load. However, sufficiently high voltage across the capacitor plates can exceed the electric breakdown voltage of air and cause capacitive power transfer to fail.

With the elimination of cables, wireless energy can make life truly mobile by charging consumer electronics like mobile phones to providing power to movable loads such as motors at the end of a crane. Implementation of wireless energy can also charge medical implants without requiring frequent surgeries to replace batteries. Also, wireless power transfer systems can be scaled to provide a broad range of power levels which support a wide variety of products and systems.

In this thesis, a medium range (5-10 cm) wireless power transfer system that can provide 0.5W to 5W of power is built and tested. The system employs resonant inductive coupling to transfer power. Later in the thesis, the power efficiency of the system is optimized by the implementation of Optimal Power Transfer Regulator which uses indirect feedback from the secondary to ensure that the primary (or the transmitter) provides just enough power to regulate secondary (or receiver) load and minimizes standby power losses [5].

Chapter organization of this thesis is as follows: Chapter 1 discusses inductive and resonant inductive coupling in more details; Chapter 2 and 3 discusses the receiver and transmitter architecture of the open loop system respectively; Chapter 4 discusses the problems with the current open loop setup and introduces a novel indirect feedback method to implement the Optimal Power Transfer Regulator: Chapter 5 discusses the implementation of the Optimal Power Transfer Regulator (OPTR); in Chapter 6 the new closed loop system with OPTR is tested by evaluating the improvement in the power efficiency in comparison to the previous open loop system and concludes with a summary and suggestions for future work.
References


Chapter 1
Inductive Power Transfer System

In an Inductive Power Transfer System, energy is transferred across a physical gap between the primary and the secondary by electromagnetic induction. The basic structure and operation of an Inductive Power Transfer system is discussed in this section.

1.1 Inductive Power Transfer

A typical setup of an Inductive Power Transfer system is shown in figure 1-1. The system consists of a primary unit and a secondary unit which are electrically isolated. The transmitter circuit and transmitter coil are contained in the primary unit. The transmitter circuit, powered by supply voltage ($V_{supply}$), is a DC/AC converter which creates an alternating current in the transmitter inductor loop (TX coil). The secondary unit contains a pick up coil (or receiver [RX] coil) connected to some regulator circuitry. When the receiver coil is placed in the magnetic field created by the alternating current in the transmitter coil, an A/C voltage is induced across
the receiver coil due to the inductive coupling between the two coils. The A/C voltage induced across the receiver coil serves as the supply voltage to the secondary unit. Inside the secondary unit, the A/C voltage is rectified and fed as the input to the regulator circuitry.

Depending on the distance and the alignment of the receiver coil with respect to the transmitter coil, the receiver coil absorbs only a part of the magnetic flux generated by the transmitter coil. Coupling factor, $k$, is a measure of the amount of flux generated by the transmitter coil linking the receiver coil. The value of $k$ ranges from 1 to 0: coupling factor of 1 represents that all the flux generated by the primary coil is linking the secondary coil, therefore the two coils are perfectly coupled. A coupling factor of 0 means that none of the flux generated by the primary coil reaches the secondary coil therefore the two coils are not magnetically coupled. In the Inductive Power Transfer Systems considered in this thesis, the coupling factor can range from 0.1-0.6 and therefore are loosely coupled.

In inductive coupling, the energy is transferred due to the coupling of the transmitter and receiver coils. For efficient energy transfer, the receiver coil needs to be tightly coupled to the transmitter coil such that the receiver coil can pick up as much of the magnetic flux generated as is possible. Since magnetic field density weakens with distance, the smaller the separation between the two coils, the stronger their coupling tends to be.

The concept of inductive coupling is similar to the operation of a traditional transformer. However, traditional transformer methods are not generally feasible method to transmit over larger air gaps (for medium range power transmission) [1]. The power transmission range can be improved by using resonance on the secondary unit. Resonance effect can be easily incorporated to the receiver circuit by forming a LC tank on the secondary side. The primary side can also be made resonant by forming a LC tank. The energy coupling is strongest when the primary and secondary LC tanks are tuned to the same resonant frequency and there is weak energy coupling between off-resonant objects. The LC tanks are tuned by adjusting the capacitor and inductor values.

From Coupled Mode Theory [2], it can be shown that resonant inductive coupling delivers more power in comparison to traditional non-resonant inductively coupling when the geometry
and supply voltage is fixed [3]. A more detailed explanation is given using the transformer model in the next section 2.2.

Figure 1-2 shows basic structure of a Resonant Inductive Power Transfer System. The LC tank topology does not necessarily have to be in parallel configuration. The transmitter and receiver circuits are discussed in more detail in chapter 3 and chapter 2 respectively.

![Figure 1-2: Resonant Inductive Power Transfer System](image)

1.2 Resonant Inductive Coupling: Transformer Model

The transformer model shown in figure 1-3 can be used explain the operation of resonant inductively-coupled systems.

![Figure 1-3: Transformer Model](image)
$V_{primary}$ is the independent A/C voltage across the primary LC tank which supplies power to the system. The A/C voltage, $V_{secondary}$, is voltage developed at the secondary side due to the magnetic coupling between transmitter and receiver coils. $C_s$ and $C_p$ are the capacitors on the primary and secondary side LC tank respectively. The mutual inductance ($L_m$) between the primary and secondary coils due to magnetic coupling is given by,

$$L_m = k\sqrt{L_1 L_2}$$

where $L_1$ and $L_2$ are the self-inductances of the primary and secondary respectively. The self-inductance of the primary ($L_1$) and secondary ($L_2$) are given by,

$$L_1 = L_{t1} + \frac{L_m}{n} \text{ or } (L_1 = L_{t1} + L_u)$$

$$L_2 = L_{t2} + n L_m$$

where,

$L_{t1}$ = leakage inductance of the primary

$L_u$ = magnetizing inductance of the primary

$n$ = turns ratio

$L_{t2}$ = leakage inductance of the secondary

The secondary side from figure 1-3 can be reflected to the primary resulting in a simple circuit shown in figure 1-4.

![Figure 1-4: Simplified Transformer Model](image-url)
C_r is the reflected capacitance of the secondary and L_r is the reflected leakage inductance of the secondary. When the transmitter and receiver coils are loosely coupled, the secondary leakage inductance is significantly larger than the magnetizing inductance. Therefore, in an inductively coupled system if C_r is set to 0, most of the primary current $i_p$ will flow through the magnetizing inductance ($i_m >> i_s'$) and back to the source. However, in resonant inductively coupled systems, the addition of the secondary capacitor lowers the secondary impedance at resonance. Hence, resonant inductive coupling transfers power across larger air gaps much more efficiently [3].

When the transmitter and receiver coils are tightly coupled in a resonant inductive system, mutual inductance, $L_m$, increases and the secondary leakage inductance, $L_{12}$, drops. The resonant frequency of the secondary LC tank shifts away from the primary resonant frequency. Therefore, the tightly coupled receiver does not pick up as much energy as the loosely coupled receiver coil. When the receiver coil is tightly coupled, $L_r$ in figure 1-4, drops significantly and reflected secondary impedance is dominated by the reflected secondary LC tank capacitance, $C_r$. At the primary resonant frequency, secondary impedance is not lowered (unlike in the loosely coupled system). For tightly coupled systems, the traditional non-resonant coupling method transfers power more efficiently. The resonant inductive coupling method is more effective at enhancing energy transfer for lower coupling factors. Hence, when the geometry of the transmitter and receiver coils is fixed, resonant inductively coupling has an optimum coupling factor when the system is not too loosely coupled or too tightly coupled.

1.3 Transmitter and Receiver Coil Magnetics

1.3.1 Quality Factor

The transmitter and receiver coils need to possess high quality factors, $Q$, to minimize power losses in the system. The quality factor for a loop with inductance $L$ and a series parasitic resistance $R$ at a frequency of $f$ is given by,
Depending on the amount of power transfer, the current flowing through the transmitter loop can easily vary over a large range (for example from 1A to 10A in the prototype system described in this thesis). To reduce standby power losses, the series resistance of the transmitter and receiver coils must be minimized.

### 1.3.2 Optimum Coil Geometry

For the receiver coil to pick up as much magnetic flux as possible and develop that flux into a large voltage, both the number of loops and the area of each loop need to be large. In one coil configuration, the area of each loop is fixed and the number of loops can be stacked vertically (see figure 1-5). However, loops that are away from the transmitter’s magnetic field will pick up smaller amounts of magnetic flux, while still contributing loss. After a certain (optimum) number of loops, the losses due to the series resistance start to dominate which will reduce the amount of power transfer to the secondary (see figure 1-6).

![Figure 1-5: Transmitter and Receiver coil](image)

![Figure 1-6: Maximum pickup vs number of loops](image)
Instead of stacking the loops vertically, the loops can be wound on the same plane so that all the loops pick up the optimum magnetic flux and the area of each loop keeps increasing as loops are added (see figure 1-7). The same coil configuration can be used for the transmitter coil.

![Figure 1-7: Transmitter (or Receiver) coil](image)

1.3.3 Addition of Ferrite Material

A ferrite block placed underneath the transmitter loop improves the pickup by the receiver dramatically. The ferrite block needs to be placed underneath the loop (not in the center of the transmitter coil). When the ferrite block is placed underneath the loop the magnetic field of the current loop is more vertical and concentrated inside the loop. The ferrite block is acting as 'magnetic flux guide' by guiding the magnetic field underneath the loop more into the center of the loop and strengthening the field inside the loop. Figure 1-8 and 1-9 show the loop without and with the ferrite material respectively.
The ferrite block has to fit the size of the loop. If the ferrite block is bigger than the loop itself, the receiver pickup decreases. The magnetic field in the center of the loop is probably less vertical and uniform. If the ferrite block is placed inside the loop (i.e. between the primary and secondary, the ferrite tends to short out the magnetic coupling which makes it harder for the receiver to pick up any magnetic flux. The magnetic flux pattern for a loop with a ferrite block larger than the size of the coil is shown in figure 1-10.

When more than one loop is used as the transmitter (for example having two loops next to each other) the ferrite block helps to guide the magnetic flux between the loops. Hence, the
effect of having the second loop is more pronounced. The ferrite block helps to direct and strengthen the magnetic field density of a loop at certain locations by placing loops with current flowing in the same or opposite direction in different orientations.

References


Chapter 2
Receiver Architecture

The basic receiver circuit introduced in the previous section is explored in more detail in this chapter. The typical structure of a receiver, as shown in Figure 2-1, consists of a LC tank formed by receiver coil $L_{t-rx}$ and capacitor $C_{t-rx}$ which is tuned to the same resonant frequency as the primary side LC tank formed by $L_{t-tx}$ and $C_{t-tx}$. The $L_{t-rx}$ is the inductance of the receiver coil when the transmitter coil is not magnetically coupled to it (and $L_{t-tx}$ is inductance of the transmitter coil when the receiver coil is not magnetically coupled to it). The output voltage of the LC tank is then rectified and connected to a regulator which can be a step down DC/DC converter or a tuning controller.

![Figure 2-1: Basic Receiver Circuit (with transmitter coil coupled)](image)

When the primary (transmitter) circuit is coupled to the secondary (receiver) circuit, the system can represented by the transformer model in Figure 2-2 (the transformer model is
explained in more detail in chapter 1). The complete transformer model can further simplified by replacing the primary circuit with its Thevenin equivalent when looking back from the secondary tank capacitor ($C_{t-rx}$) [1]. The simplified circuit is shown in .

![Figure 2-2: Complete Transformer Model](image)

![Figure 2-3: Primary Circuit replaced with Thevenin Equivalent](image)

In Error! Reference source not found., $L_{eq}$ is given by,

$$L_{eq} = L_{t2} + (nL_m || n^2L_{t1})$$

where,

$L_{t1} =$ leakage inductance of the primary coil
L_{12} = \text{leakage inductance of the secondary coil}

L_m = \text{mutual inductance between the primary and secondary coil due to magnetic coupling}

n = \text{number of turns from the primary to the secondary}

and \( V_{eq} \) is given by,

\[
V_{eq} = k \left( \frac{L_2}{L_1} \right)^{V_{primary}}
\]

where,

L_1 = \text{self-inductance of the primary (see chapter 1.2)}

L_2 = \text{self-inductance of the secondary (see chapter 1.2)}

k = \text{coupling factor between the primary and secondary coils}

V_{primary} = \text{AC voltage applied across the primary LC tank}

The receiver circuit in Error! Reference source not found. is modified to take into account the coupling between the transmitter coil and the receiver coil to produce the circuit in Error! Reference source not found..

![Figure 2-4: Simplified Receiver Circuit (with transmitter coil coupled)](image-url)
2.1 Receiver with Tuning Controller

Various receiver circuits can be designed by changing the LC tank topology (e.g. parallel, series and parallel-series) and adjusting the tuning controller to regulate the output voltage accordingly. The tuning controller uses negative feedback to regulate the output voltage by changing the resonant frequency of the secondary LC tank [2].

2.1.1 Different Topologies

2.1.1.1 Parallel LC Tank – Tuned to Detuned

Figure 2-5 shows a receiver circuit with the LC tank in parallel configuration. The impedance, $Z$, of a parallel RLC circuit is given by,

$$ Z(s) = \frac{sLR}{s^2LCR + sL + R} $$

**Figure 2-5: Parallel LC Tank Receiver Circuit (with no transmitter coil coupled)**

Quality factor, $Q$, of a parallel RLC circuit is,

$$ Q = R \sqrt{\frac{C}{L}} $$
When the tuning controller is switched off (i.e. when MOSFET M_d is turned off), the secondary LC tank operates at the same resonant frequency as the primary LC tank and absorbs the maximum possible amount of energy from the transmitter coil. When the output voltage exceeds the regulated output level (which is set internally in the tuning controller), the tuning controller turns on the MOSFET M_d and adds the capacitor C_d to the secondary LC tank. The resonant frequency of the secondary LC tank is changed to a lower value ('detuned') due to the addition of capacitor C_d. When detuned, the secondary LC tank absorbs less energy from transmitter. The receiver achieves regulation by controlling the amount of energy picked up by the secondary LC tank using the tuning controller. The impedance curves when the secondary LC tank is in tune with the primary and when detuned are plotted in Figure 2-6.

![Figure 2-6: Impedance vs Frequency Curve](image)
In Figure 2-6, the green curve is the impedance of the LC tank when the detuning capacitor \( C_d \) is switched off. The impedance curve shifts to the left when \( C_d \) is switched on (blue curve).

The upper limit of the load regulation range (that is the upper limit of the load current for when the output voltage level is in regulation) is determined by the maximum amount of energy picked by the receiver coil. The lower limit of regulation is set by the value of the detuning capacitor. Ideally, the receiver should be in regulation when the load current is 0 A. This condition requires the detuning capacitor to be replaced by a wire to essentially short the secondary LC tank and prevent the receiver from absorbing any energy from the primary. However, then the detuning switch, \( M_d \), must be able to handle a large amount of current to completely detune the secondary LC tank. Therefore, the practical approach to extend the lower limit of regulation is to place a big enough detuning capacitor (in this setup about 1uF). The value of this detuning capacitor will determine how much the impedance curve in Figure 2-6 will shift to the left and thus the regulation range.

One caveat is presented with this parallel LC tank receiver topology: when the receiver coil is tightly coupled to the transmitter coil, large voltages can be generated in the receiver circuit because the secondary LC tank starts off being in tune with the primary. This problem can be alleviated by setting the LC tank to be detuned in the first place and capacitor added to the network to make it in tune with the primary. An example of such method is shown with a parallel-series LC tank method shown in the next section.

### 2.1.1.2 Parallel-Series LC Tank – Detuned to Tuned

The parallel-series topology shown in Figure 2-7 is initially detuned and is tuned to the primary resonant frequency when the tuning controller switches on the MOSFET \( M_d \). Parallel-series receiver circuit has the tank capacitor, \( C_t \), in series with receiver coil, \( L_t \), and the detuning or rather the tuning capacitor, \( C_d \), in parallel with receiver coil. When \( M_d \) is switched on, the total tank capacitance is \( C_d + C_t \). The resonant frequency \( (f_0) \) of the parallel-series LC tank shifts from \( 1/(2\pi\sqrt{L_tC_t}) \) to \( 1/(2\pi\sqrt{L_t(C_d + C_t)}) \) where the latter is equal the primary resonant frequency.
Capacitor, $C_t$, controls the minimum load required to maintain regulation. Therefore, by decreasing the value of $C_t$, the receiver can regulate smaller minimum loads. However, the same capacitor $C_t$ also controls the amount of power delivered to the load therefore $C_t$ needs to be sufficiently large to power the tuning circuitry and achieve regulation in the first place.

Capacitor, $C_d$, controls the upper limit in the load regulation range. Bigger value for $C_d$ increases the maximum amount of load current the receiver circuit can have while maintaining voltage regulation at the output. However, two factors constrict the value of $C_d$.

1. Since the receiver is shifting from detuned to tuned by adding $C_d$ to the tank, $L_t$ and $C_t+C_d$ combination need to match the primary resonant frequency. Unlike in the parallel
LC tank receiver, $C_d$ cannot be set to an arbitrarily large value: the primary resonant frequency sets a limit on $C_d$.

2. The amount of power delivered to the load is still controlled by capacitor $C_t$. In order to increase the upper limit in the current load regulation range, $C_t$ and $C_d$ both must be increased.

Though the parallel series receiver prevents any large voltages build up in the receiver circuit during power up, the load regulation range is severely limited in comparison to the parallel receiver. The ideal receiver configuration, which a combination of both parallel and parallel-series receivers, is shown in Figure 2-9.

![Figure 2-9: Parallel Series LC tank (with detuning capacitor across tank capacitor)](image)

The receiver is initially detuned and tuned to the primary LC tank with addition of capacitor $C_d$. Since detuning/tuning capacitor $C_d$ is added across the $C_t$, the receiver can regulate higher current loads by delivering more power to the load. However, implementing the gate drive for MOSFET $M_d$ is rather challenging. For this thesis, the parallel LC tank receiver topology was implemented.
2.1.2 Switching Techniques for the Tuning Controller

The tuning controller consists of an operational amplifier or a comparator which compares the rectified output voltage to a previously set reference voltage and produces a voltage that controls the gate of the detuning MOSFET. Different tuning controller circuits can be implemented mainly by employing different methods to switch the detuning MOSFET. The two such techniques are discussed in the following section.

2.1.2.1 Switching technique: Linear

The receiver circuit in Figure 2-10 uses a tuning controller which drives the gate of the detuning MOSFET, $M_d$, linearly to regulate the output voltage. The negative feedback is formed as follows. The operational amplifier LT1018 compares the output voltage, which is divided down by the resistor divider set by resistors R3 and R4, to the reference voltage set by the two diodes D2 and D3. The output of the operational amplifier is filtered by the RC low pass filter set by C3 and R1 ($f_{3dB} = 72.3\, \text{kHz}$) and drives the gate of the detuning MOSFET $M_d$. The output voltage of this receiver circuit is regulated at about 10V.

Figure 2-10: Tuning controller - linear switching
Figure 2-11 shows the gate drive of $M_d$. The gate voltage switches linearly but is not driven all the way to the output voltage. The negative feedback loop is unstable therefore the gate voltage, as seen in Figure 2-11, is oscillating. Some higher frequency components observed on the gate waveform are attributed to the oscillating waveform at resonant frequency coupled from the drain of $M_d$ to the gate. Other high frequency content is due to the parasitic inductances present on the circuit board.

![Figure 2-11: Gate voltage of the detuning MOSFET](image)

**Figure 2-11: Gate voltage of the detuning MOSFET**

![Figure 2-12: Receiver board with linear switching](image)

**Figure 2-12: Receiver board with linear switching**
Linear switching is not the optimum method to drive detuning MOSFET: a significant amount of power is dissipated in the MOSFET $M_d$.

The tuning controller in Figure 2-13 stabilizes the gate voltage of $M_d$. $V_{ref2}$ was set to about 1.3-1.4V. This approach is even more inefficient at driving the gate of $M_d$.

![Figure 2-13: Tuning controller with stabilized linear switching](image)

The gate voltage for the stabilized tuning controller is shown in Figure 2-14.
During certain points of operation the circuit briefly becomes unstable. This observation can probably be explained by the gain curve. The gain curve is a graph of gate voltage of $M_d (V_g)$ plotted against output voltage ($V_{out}$) when the circuit is driven in open loop. The transfer function is given by,

$$\frac{V_{out}}{V_g} = G_m \cdot \frac{R_{load}}{R_s + R_{load}} \cdot \frac{1}{1 + s \cdot C_{out} \cdot R_{load}}$$

where $R_s$ (source impedance) was measured to be 16.8Ω, $C_{out}$ is the output capacitor and $G_m$ is simply a constant (refer to circuit in Figure 2-13). As explained previously, the gain curves were plotted by setting $R_{load}$ to a fixed value and driving the circuit open loop by applying a DC voltage at the gate of $M_d$. Therefore, the slope of gain curve is simplified to,

$$slope = G_m \cdot \frac{R_{load}}{R_s + R_{load}}$$

Gain plots for two different loads are plotted in Figure 2-15 and Figure 2-16. The circuit became unstable probably due to the non-linear regions of the gain curves.
Figure 2-15: Gain curve

Figure 2-16: Gain Curve
Power efficiency of the receiver can be increased by improving the gate drive of the detuning MOSFET using hysteresis. The hysteresis switching method is discussed in next section.

2.1.2.2 Improved Switching Technique: Hysteresis

In this switching mode, the gate of the detuning MOSFET is switched harder by driving $M_d$ all the way to the output voltage using hysteresis. Hence, the gate is either on or completely off as seen in Figure 2-17. Some ripple is introduced at the output voltage due to hysteresis (see Figure 2-18). The receiver circuit with hysteresis switching is shown in Figure 2-19.

![Graph showing gate voltage of detuning MOSFET over time](image)

**Figure 2-17: Gate voltage of detuning MOSFET**
The comparator LTC1440 has programmable hysteresis and a built-in reference which was used to set $V_{ref}$.

The overall power efficiency was measured for the three switching techniques. The test setup is shown in Figure 2-20. Class-D power amplifier (see Chapter 3 for more details) was
used as the DC/AC converter to generate an AC voltage of 140Vpk-pk across the primary LC tank. The resonant frequency of the primary LC tank was 108 kHz. The DC supply ($V_{\text{supply}}$) to DC/AC converter was fixed at 20V. The output voltage of the receiver ($V_{\text{out}}$) was regulated at 10V. The receiver coil position was fixed with respect to the transmitter (see Figure 2-21). The measurements were taken for the three different tuning controllers. The results are shown in table 1.

![Diagram](image1.png)

**Figure 2-20:** The test setup to compare power efficiencies of different tuning controllers

![Diagram](image2.png)

**Figure 2-21:** Receiver coil (5.6uH) placed at the center of transmitter coil (22uH)
Table 1: Comparison of power efficiencies with different tuning controllers

<table>
<thead>
<tr>
<th></th>
<th>Linear and Stable</th>
<th>Linear and Unstable</th>
<th>Hysteresis</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{\text{supply}}$</td>
<td>20V</td>
<td>20V</td>
<td>20V</td>
</tr>
<tr>
<td>$I_{\text{supply}}$</td>
<td>1.09 A</td>
<td>1.1 A</td>
<td>1.15 A</td>
</tr>
<tr>
<td>$V_{\text{out}}$</td>
<td>10 V</td>
<td>10 V</td>
<td>10 V</td>
</tr>
<tr>
<td>$I_{\text{load}}$</td>
<td>800 mA</td>
<td>830 mA</td>
<td>1 A</td>
</tr>
<tr>
<td><strong>Efficiency</strong></td>
<td><strong>36.70 %</strong></td>
<td><strong>37.73 %</strong></td>
<td><strong>43.48 %</strong></td>
</tr>
</tbody>
</table>

There is a significant improvement in overall power efficiency by driving the gate of detuning MOSFET with hysteresis. However, the switching of detuning capacitors is more audible with the hysteresis gate drive. The tuning controller implemented in Figure 2-22 uses pulse width modulation to fix the switching frequency of the tuning capacitor at (non-audible) frequency of 30 kHz.

![Figure 2-22: Modified hysteretic tuning controller with PWM](image-url)
Figure 2-23: Receiver circuit with hysteretic tuning controller (with PWM)

Figure 2-24: Gate voltage of the detuning MOSFET for hysteretic tuning controller with PWM (for 2 different loads)
2.2 Future work: Receiver with Buck Converter

The receiver setup in Figure 2-25 replaces the tuning controller with a buck converter to step down secondary LC tank voltage and regulate the output voltage. This receiver configuration is much more power efficient due several reasons.

1. Buck converters can be designed to have high power efficiencies which in turn improves the overall power efficiency of the secondary unit.

2. A significant amount of power is dissipated in the ESR of the detuning capacitor and the detuning MOSFET during switching. Using a buck converter that can withstand a high voltage input will not require any detuning circuit to pre-regulate the output of the receiver’s rectifier.

![Figure 2-25: Receiver circuit with buck converter](image)

References


Chapter 3
Transmitter Architecture

The transmitter circuit, which comprises of a DC/AC converter, produces the AC voltage across the primary LC tank to generate a magnetic field. Figure 3-1 shows a typical setup for a transmitter circuit with the receiver (secondary) circuit coupled. The power converter does not necessarily have to be DC/AC converter: it can be AC/AC or AC/DC/AC converter. For this thesis, DC/AC converter is used for power conversion.

![Figure 3-1: A typical transmitter circuit (with the receiver coupled)](image)

In this chapter, two different topologies of DC/AC converters for a Resonant Inductive Power Transfer system are discussed.

3.1 Cross-coupled Current Mode Class-D Power Amplifier

The gate drive of a current mode class-D power amplifier can be modified to design a resonant oscillator which then can be used as a DC/AC converter in the transmitter circuit. An ideal current mode class-D (CMCD) amplifier is shown in Figure 3-2 on the right side. The two
transistors M1 and M2, which control the two current sources, are driven 180° out-of-phase [1]. Ideally, the transistors have zero voltage across at the time of switching caused by the resonance filter (formed by the parallel LC tank - \(L_{\text{tank-tx}}\) and \(C_{\text{tank-tx}}\)). Therefore, the ideal operation achieves zero-voltage-switching (ZVS). The circuit to the left in Figure 3-2 replaces the two current sources with two large inductors \(L_{s1}\) and \(L_{s2}\). \(L_{s1}\) and \(L_{s2}\) are wound on the same core therefore are tightly coupled. It is important to note where the polarities of \(L_{s1}\) and \(L_{s2}\) are when connected in the circuit: this is to ensure that the transistors are driven 180° out-of-phase. Also the output capacitances of the two transistors become a part of the parallel LC tank.

The gate drives of the two transistors M1 and M2 can be cross-coupled (i.e. gate of M1 can be connected to the drain of M2 and vice versa) to design a resonant oscillator. The frequency of such an oscillator will be set by the resonant frequency of the parallel LC tank. In a resonant inductive power transfer system, \(L_{\text{tank-tx}}\) will be the inductance of the transmitter (primary) coil.

Depending on the amount of power delivered by the primary circuit, the voltage across the primary LC tank can be large. Therefore in a practical circuit, the gates of the transistors cannot be directly cross-coupled to the respective drains of the transistors. The drain voltage is divided down before driving the gate of the transistor. In the next section, implementation of the gate drive is discussed in more detail.
3.1.1 Gate Drive

Three different versions of gate drive implementations were tested. The first version of the gate drive implemented with a resistor and zener is shown in Figure 3-3. This is an inefficient way to drive the two MOSFETs because a significant amount of power dissipated in the 1kΩ resistors and two zeners. The output waveform across the LC tank is distorted and becomes even more distorted when the supply voltage is increased. This transmitter cannot operate at higher voltages (more than 10V) as the resistor and zener gets too hot to function.

![Figure 3-3: Cross-coupled class-D amplifier (with zener and resistor) – version 1](image)

In the second version, the resistor and zener was replaced with a capacitive divider (Figure 3-4). The output of the capacitor divider is then sent to a comparator (followed by an inverter) to drive the gate of the MOSFET harder to minimize power dissipation in the MOSFETs. In this setup, C5 (47nF) and L3 (22µH) form the LC tank of the primary. L3 is the inductance of the primary coil.
To find the optimum operating point for the transmitter, different ratios for the capacitor divider were tested (Table 2). The ratio of the capacitive divider is given by the following expression,

\[
\text{ratio} = \frac{\frac{C_2}{C_2 + C_3}}{\frac{C_1}{C_1 + C_4}}
\]

![Cross-coupled class-D amplifier (with capacitive divider) - version 2](image)

**Figure 3-4: Cross-coupled class-D amplifier (with capacitive divider) – version 2**

**Table 2: Effect of Capacitive Divider ratio**

<table>
<thead>
<tr>
<th>Ratio</th>
<th>C1,C2</th>
<th>C3,C4</th>
<th>Max. primary loop current (I_{lp}/A_{pp})</th>
<th>Max. Tank Voltage ((V_{acp} - V_{acp})/V_p)</th>
<th>Max. Supply Voltage (V_{supply}/V)</th>
<th>Max. Supply Current/A</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.68</td>
<td>47n</td>
<td>22n</td>
<td>3.7</td>
<td>55.6</td>
<td>9.5</td>
<td>0.7</td>
</tr>
<tr>
<td>0.5</td>
<td>47n</td>
<td>47n</td>
<td>4.1</td>
<td>59.6</td>
<td>10</td>
<td>0.8</td>
</tr>
<tr>
<td>0.5</td>
<td>100n</td>
<td>100n</td>
<td>3.1</td>
<td>46.8</td>
<td>8.6</td>
<td>0.9</td>
</tr>
<tr>
<td>0.4</td>
<td>100n</td>
<td>150n</td>
<td>4.1</td>
<td>52.8</td>
<td>9</td>
<td>0.9</td>
</tr>
<tr>
<td>0.32</td>
<td>47n</td>
<td>100n</td>
<td>5.37</td>
<td>78.4</td>
<td>13.5</td>
<td>0.8</td>
</tr>
<tr>
<td>0.24</td>
<td>47n</td>
<td>150n</td>
<td>6.8</td>
<td>111</td>
<td>20</td>
<td>0.8</td>
</tr>
<tr>
<td>0.24</td>
<td>68n</td>
<td>220n</td>
<td>7.2</td>
<td>113</td>
<td>20</td>
<td>0.7</td>
</tr>
<tr>
<td>0.23</td>
<td>100n</td>
<td>330n</td>
<td>7.7</td>
<td>115</td>
<td>20</td>
<td>0.56</td>
</tr>
</tbody>
</table>

45
Table 2 shows the maximum current through the primary coil and LC tank voltage that can be reached without heating up the capacitor divider and the zener too much for different capacitor ratios. For higher ratios, the maximum supply voltage that can be reached is about 10V. At higher ratios, the voltage across the zener is clipping therefore power is dissipated in both the zener and ESR of the capacitors. At a ratio of 0.24, the capacitor divider and the zener were getting hot but comparatively less to the original setup in Figure 3-3.

Figure 3-5 shows the voltage at ACP and ACN nodes of the transmitter. The voltage across the LC tank would be ACN – ACP. The voltage output across the tank is not sinusoidal. The overlap between two waveforms can be minimized by bringing down the reference voltage, $V_{ref}$, to about 0V on the LT1720 comparator.

The current through the primary coil is shown in Figure 3-6.
This circuit has a startup issue. Reference voltage, $V_{ref}$, on the comparator can be set to zero during operation but when circuit is starting up, $V_{ref}$ need to be at a higher value (about 3V) and then slowly decreased to 0V. This is to ensure that one of the MOSFETs switches to deliver power to the LC tank. If $V_{ref}$ is set to zero at startup, the supply gets shorted as both MOSFETs (M1 and M2) turns on caused by the initial rising edge on ACP and CAN when the supply is first turned on. In the third version of the cross-coupled class-D amplifier, the startup issue is solved by an RC network to slowly bring $V_{ref}$ from 5V ($V_{cc}$) to 0V with a slow time constant (see figure 10). In the third version of transmitter, the gate drive is further improved by adding a level shifter to drive the gates of M1 and M2 harder. In the third version, all passive components are replaced by the surface mount typess to reduce energy dissipated due to ESR (see Figure 3-7).

Figure 3-6: Current in the primary coil (with version 2 circuit)

Figure 3-7: Actual setup of the cross-coupled class-D amplifier (version 3)
Figure 3-8: Cross-coupled class-D amplifier with capacitive divider and level shifter (version 3)
Figure 3-8 shows the full schematic for the version 3 of cross-coupled amplifier. The LC tank values are $C_5 = 47\,\text{nF}$ and $L_3 = 22\,\text{uH}$. The capacitor ($47\,\text{nF}$) in the LC tank is made up by ten of $4.7\,\text{nF}$ in parallel to distribute the heat dissipated more efficiently. Supply voltage to comparator, $V_{cc}$, is $5\,\text{V}$. Supply voltage to the level shifter, $V_{cc2}$, can be $5\,\text{V}$ or greater (up to $V_{gs,\text{max}}$ of IRF3710 which is $20\,\text{V}$) However, the MOSFETs (Si3552) used in the inverter, which drive the gates of M1 and M2, can handle only up to $2\,\text{A}$. Therefore the maximum value we set $V_{cc2}$ is about $7-10\,\text{V}$. The ratio of the capacitor divider is reduced to 0.043. The voltage across the zener is not clipping anymore (see Figure 3-9) therefore it is not getting heated up.

![Figure 3-9: Voltage on the zener (version 3 circuit)](image)

The capacitors C15 and C16 (in Figure 3-7) eliminate the ringing on the output waveform. The ringing was probably caused by inductance from the leads of the two MOSFETs M1 and M2. Note however that C15 and C16 are now part of the LC tank. Therefore, the effective capacitance of the LC tank is $C_{\text{eff}} = C_5 + 0.5(0.1\,\text{uF}) = 97\,\text{nF}$. The resonant frequency of the transmitter is $108\,\text{kHz}$.

Figure 3-10 shows the voltage at node 'ACN' (red) and gate voltage of M1 (blue). Similarly, Figure 3-11 shows the voltage at node 'ACP' (purple) and the gate voltage of M2 (black). In both ACP and ACN waveforms the falling edge is slower compared to the rising edge. At the end of the falling edge of ACP and ACN, both MOSFETs M1 and M2 are switched off which opens the
current source (refer to Figure 3-2). This can be seen in figure 3-12 where both gate voltages are plotted on the same graph.

Figure 3-10: Gate voltage of M1 (blue) & drain voltage of M1 (red)

Figure 3-11: Gate Voltage of M2 (black) and drain voltage of M2 (purple)
(The gate voltage of M2 has some ringing on the rising edge of the waveform. This is due to parasitic inductance on the path connecting the inverter to the gate of M2.)

The time during which both MOSFETs are switched off can be reduced by decreasing the reference voltage of the comparator, \( V_{\text{ref}} \), to a negative value (about -0.2 to -0.3 V) which switches on the MOSFET a little early. Another solution would be to increase the ground nodes of the zeners to a more positive voltage (about 0.2-0.4V). By reducing the two MOSFETs switching off time, the overlap between the two waveforms ACP and ACN can be reduced (see Figure 3-13).

The caveat with this transmitter board is that the supply voltage to the comparator (\( V_{\text{cc}} \)) needs to switched on after switching on the \( V_{\text{supply}} \). \( V_{\text{supply}} \) needs to be greater than about 12V for the transmitter to work due to the smaller capacitor divider ratio.

The startup issue is not completely resolved even in the third version of the cross-coupled class-d amplifier with the RC network to bring \( V_{\text{ref}} \) slowly down from 3V to 0. Startup requirements for this circuit are discussed in the next section.
3.1.1 Startup Requirements of the Cross-coupled Class-D amplifier

For cross-coupled class-D oscillator to start oscillating, minimum amount of loop gain or similarly minimum amount of negative resistance is required [2]. To evaluate the startup conditions of the circuit, transconductance is set to its small signal value because the circuit operates in the small signal region before the oscillations have built up in the circuit [3].

Figure 3-14 shows the resistance $R_s$ looking in to the differential cross-coupled NMOS pair. Let $R_P$ be the equivalent parallel resistance of the primary LC tank that is in parallel with $R_s$. $g_m$ is the small signal transconductance of the NMOS device.
Figure 3-14: Resistance of a differential cross-coupled NMOS pair

From small signal analysis of the differential cross-coupled NMOS pair, $R_a$ is given by,

$$R_a = -\frac{2}{g_m}$$

Parallel equivalent resistance of the LC tank, $R_T$, is a positive value therefore for the overall resistance to be negative,

$$\frac{1}{R_a} + \frac{1}{R_T} < 0$$

After simplifying the above expression to ensure the cross-coupled class-D oscillator to start,

$$g_m > \frac{2}{R_T}$$

When the secondary (receiver) circuit is not coupled to the primary circuit, $R_T$ is the parasitic parallel resistance of the LC tank. When the secondary circuit is coupled to the primary, the secondary load is reflected to the primary and will be included in $R_T$ in addition to the parasitic resistance of the LC tank. The startup of the oscillator is sensitive to the secondary load conditions and the parasitics of the LC tank. Therefore, the startup of the oscillator is not guaranteed. The cross-coupled class-D amplifier is not a practical solution for resonant inductive power transfer systems. In the next section, Royer oscillator is introduced and discussed which is guaranteed to startup.
3.2 Royer Oscillator

Modified version of the original royer circuit is shown in Figure 3-15 [4],[5]. The original royer circuit used the saturation of the transformer to define the switching frequency which produced a square wave output. In the modified royer, the switching occurs due to the saturation of the transistors. The switching frequency is set by the LC tank formed by $C_{t-tx}$ and $L_{t-tx}$ which is the primary LC tank. $L_{t-tx}$ is the inductance of the primary (or transmitter) coil. The resonant frequency circuit in Figure 3-15 is set about 103kHz. Unlike the original royer, the circuit is forced to run sinusoidally by the LC tank.

Figure 3-15: Modified Royer
The startup in royer is guaranteed: when the DC supply, \( V_{DC} \), is switched on, the transistor on the left (Figure 3-15) is biased which ensure self-starting of oscillations. The switching occurs when the on (left) transistor comes out of saturation and the spike in collector current is picked up by the base windings (\( L_b \)) and switches on the right transistor.

The voltage across the primary LC tank is given by,

\[
V_{ac,primary} = 2\pi V_{DC} \quad \text{(peak-to-peak)}
\]

The maximum supply voltage, \( V_{DC} \), which can be applied is limited by the base to emitter reverse breakdown of the NPN which is typically around -6V. The maximum reverse voltage is given by,

\[
V_{reverse,\text{max}} = 0.5 \pi V_{DC} \frac{L_b}{L_{s1}}
\]

By adjusting the turn ratio between the transformer’s primary winding (\( L_{s1} \)) and the base winding (\( L_b \)), the maximum supply voltage can be increased.

### 3.2.1 Base Drive

The base drive of the royer in Figure 3-15 is adjusted by varying the base potentiometer. Using the test setup shown in Figure 3-16, the effect of base drive on the overall power efficiency, incremental power efficiency and maximum secondary load current was tested when the secondary output voltage was regulated at 8.1V. The supply voltage to royer was fixed at 20V.

**Table 3: Effect of Base Drive on Power Efficiency and Maximum Secondary Load**

<table>
<thead>
<tr>
<th>Base Resistor ( R_b/\Omega )</th>
<th>Supply Current of royer with no secondary load ( I_{STBY}/mA )</th>
<th>Maximum Secondary Load ( I_{load,max}/mA )</th>
<th>Supply Current of royer at max. secondary load ( I_s/mA )</th>
<th>Incremental Efficiency %</th>
<th>Overall Efficiency %</th>
</tr>
</thead>
<tbody>
<tr>
<td>575</td>
<td>383</td>
<td>943</td>
<td>986</td>
<td>63.34</td>
<td>38.73</td>
</tr>
<tr>
<td>1.07k</td>
<td>343</td>
<td>923</td>
<td>935</td>
<td>63.14</td>
<td>39.98</td>
</tr>
<tr>
<td>2.48k</td>
<td>320</td>
<td>908</td>
<td>902</td>
<td>63.19</td>
<td>40.77</td>
</tr>
<tr>
<td>5k</td>
<td>310</td>
<td>890</td>
<td>878</td>
<td>63.46</td>
<td>41.05</td>
</tr>
<tr>
<td>10k</td>
<td>303</td>
<td>880</td>
<td>866</td>
<td>63.3</td>
<td>41.15</td>
</tr>
</tbody>
</table>
The base drive does not change the incremental or the overall power efficiency significantly. However, the base drive does change the maximum secondary load current when the secondary output is fixed: the maximum secondary load is more with a bigger base drive.

### 3.2.2 Power Dissipation

Most of the power is dissipated in the ESR of the capacitors in the primary LC tank. When X7R capacitors were replaced by COG capacitors, which have lower ESR than X7R (but lower energy density), the power consumption dropped almost by half.

Figure 3-17 shows the actual board of royer circuit (primary coil is not shown in the figure).
Figure 3-17: Royer – board (primary coil not shown)

Figure 3-18: Royer - voltage across the primary LC tank with no secondary circuit coupled
\[(V_{DC}=10V, V_{ac,primary} = 62V_{p-p}, f_0 = 106kHz)\]
Figure 3-19: Collector and Base voltage of left and right transistors in royer (dark blue = base voltage of left NPN, light blue = collector voltage of left NPN, green = collector voltage of right NPN, pink = base voltage of right NPN)

For the implementation of the closed loop system, royer was used as the DC/AC converter.

References


Chapter 4
Power Transfer Optimization

This section discusses the power inefficiencies in open loop wireless power transfer systems and existing methods that use direct feedback from the receiver to improve overall power efficiency. A new indirect feedback method is introduced later in the chapter. This new method is implemented and tested in the following chapters.

4.1 Power Dissipation in open loop wireless energy system

The removal of a direct electrical contact in a wireless power system has many benefits but also causes its own predicament: there is no direct feedback from the receiver so the receiver is unable to relay information regarding its power needs to the transmitter. The magnetic field produced by the transmitter coil needs to be strong enough to energize the receiver under worst case operating condition. Worst case operating condition occurs at maximum receiver load current with the lowest coupling between the primary and the secondary. A strong magnetic field requires large amount of current on the primary coil. Therefore, the transmitter coil will have maximum current flowing through it at all times which leads to large power dissipation in the transmitter, most significantly in the parasitic resistance of the primary coil and primary resonant capacitor.

Figure 4-1 shows the overall power efficiency of an open loop wireless energy system vs. secondary load current when the supply voltage to the transmitter (in this case a Royer circuit) is fixed at 20V. See Appendix A for the full schematic of the test setup.

The power efficiency increases to a maximum value (≈ 60-65%) with the increasing load current. For smaller load currents, the power efficiency is significantly lower because the
transmitter is providing more than enough power: the transmitter is powered to support the worst case operating condition. When there is no secondary load present, the power efficiency is zero. For instance, if the secondary load is a cellphone battery, the power efficiency of the wireless charging system is zero when the battery is fully charged.

In order to optimize the overall power efficiency of the system, the primary unit should transmit just enough energy to satisfy the power needs of the secondary. Hence, the primary circuit will need to adjust the current flowing in the transmitter coil depending on the secondary load (or several secondary loads when more than one secondary coils are coupled to the primary coil) and the coupling factor between primary and the secondary coils. A wireless feedback link needs to be established between the primary and the secondary to implement this power adaptive primary circuit.

![Overall Efficiency vs Load Current](image)

**Figure 4-1: Overall Power Efficiency of the open loop system**
4.2 Methods to improve power efficiency

The importance of feedback to adjust the power delivered by the primary to match secondary power requirements is highlighted in Figure 4-1. There are multiple ways to establish a wireless feedback link.

4.2.1 Direct Feedback

In direct feedback, some control circuitry in the receiver provides direct feedback information to the transmitter, and the transmitter adapts to the required power level (see Figure 4-2). One method to achieve direct feedback is by using a dedicated radio link between the primary and the secondary.

![Figure 4-2: Direct Feedback from Secondary to Primary](image)

The bqTESLA wireless power solution from Texas Instruments controls the amount of power transmitted depending on a feedback signal from the secondary [1]. The receiver communicates with the transmitter with two different methods: Resistive Modulation or Capacitive Modulation.

- Resistive Modulation (see Figure 4-3): The load seen by the primary side is changed by adding a communication resistor at the rectified output of the receiver. The voltage across the transmitter coil changes as a result.
Capacitive Modulation (see Figure 4-4): The voltage across the transmitter coil is modulated by changing the resonant frequency of the receiver. The resonant frequency of the receiver is changed by adding a communication capacitor to the receiver coil.

The transmitter IC detects the change in voltage across the transmitter coil, de-modulate the communication signal, and adjusts the power delivered accordingly.

Figure 4-3: bqTESLA - Resistive Modulation

Figure 4-4: bqTESLA - Capacitive Modulation
4.2.2 Indirect Feedback

In direct feedback methods described in the previous section, feedback information provided to the transmitter control circuitry relies heavily on the communication circuit used in the secondary side. A better approach to the closed loop wireless transfer system would be to implement the control circuitry in the primary side without any direct feedback communication from the secondary side. One such indirect feedback method is described in the next section.

4.2.2.1 Detecting change in gradient

In a wireless power system, all the power consumed in the secondary is provided by the primary: that is there is no power gain from the primary to the secondary. If the output voltage of the secondary is regulated efficiently, the primary should be able to detect the point when it has provided just enough power for the secondary to achieve regulation: this is the optimum regulation point for a particular secondary load [2].

To find the optimum regulation point, the supply voltage to the primary is modulated and the resulting primary current consumption is monitored. There is a change in incremental current consumption (a change in gradient) when the secondary has reached its desired regulation point. Any further increase in primary supply voltage after the receiver has achieved regulation will cause the incremental rise in primary current consumption to decrease.
Figure 4-5: Supply Current vs. Supply Voltage when a resistor $R_1$ is connected across a voltage source

In Figure 4-5 shows the I-V curve for a simple circuit with a resistor $R_1$ connected across the supply voltage $V_s$. Current $I_s$ increases linearly when the $V_s$ is increased. The incremental change in current (or the gradient of I-V curve) provided by the supply $V_s$ is constant and is equal to the reciprocal of $R_1$.

Figure 4-6: Supply Current vs Supply Voltage Curve when load $R_1$ is regulated by DC/DC converter

100% Efficiency
0V dropout

DC/DC Converter

Output regulation
point of DC/DC converter

$1/R_1$
In Figure 4-6, the load $R_1$ is now connected to the output of a DC/DC converter. For simplicity, let the DC/DC converter be 100% efficient with 0V dropout regulated at $V_{\text{reg}}$. When $V_s$ is increased, $I_s$ increases linearly till $V_s$ reaches the regulation point $V_{\text{reg}}$. After the DC/DC converter achieves regulation, any further increase in $V_s$ will cause $I_s$ to drop (Figure 4-6) because the output power of the converter is constant for a given resistive load.

$$P_{\text{out}} = \frac{V_{\text{reg}}^2}{R_1}$$

Before regulation, gradient of the graph was equal to $1/R_1$. After the regulation point, the gradient is negative. There is a change in gradient at the regulation point.

Figure 4-7: Supply Current vs Supply Voltage with dissipative resistor $R_2$

The circuit in Figure 4-7 is similar to the one in Figure 4-6 except dissipative element $R_2$ is included in addition to load resistor $R_1$. The current $I_s$ increases linearly with increasing $V_s$ till load $R_1$ reaches regulation. However, the gradient before the regulation point changed when this regulation point is reached.
After the desired regulation point is achieved, incremental power consumption is determined by the dissipative resistor $R_2$ as the power output of the DC/DC converter is fixed for load $R_1$. At higher voltages ($V_s$), gradient of the I-V curve is equal to $1/R_2$ as seen in Figure 4-7. As long as the output load (which is $R_1$ in this case) is regulated, there is always a change in gradient.

![Figure 4-8: Model of Resonant Inductive System (including parasitics)]](image)

The circuit in Figure 4-7 can be used to model the wireless power transfer system shown in Figure 4-8. $L_{tank-tx}$ is the transmitting coil and $L_{tank-rx}$ is the receiver coil. The transmitter coil and receiver coil are coupled by some coupling factor $k$. $L_{tank-tx}$ and $C_{tank-tx}$ form the LC tank on the primary side, and $L_{tank-rx}$ and $C_{tank-rx}$ form the LC tank on the secondary. $R_{p1}$-$R_{p4}$ represent the parasitic resistances associated with each capacitor and inductor.

Parasitic resistances $R_{p1}$-$R_{p4}$ can be modeled by the dissipative element $R_2$ in Figure 4-7. The load resistance $R_{load}$ of the secondary is equivalent to $R_1$ in the circuit model. The graph in Figure 4-9 shows the change in the RMS supply current ($I_{ac, rms}$) when the RMS primary supply voltage ($V_{ac, rms}$) is modulated.
In Figure 4-9, the secondary tries absorbs as much as possible from the primary coil to regulate its load for smaller primary supply voltages (refer to slope 1 on the graph). When the primary supply voltage reaches \( V_{\text{reg}} \), the primary delivers enough power for the secondary load to reach regulation. If the primary supply voltage is increased further, any incremental rise in primary current consumption (slope 2) is due to the power dissipated in the parasitic components present on both primary and the secondary. Hence, there is always a detectable change in gradient in the primary current consumption provided the secondary load is efficiently regulated (slope 2 < slope 1).

The graph in Figure 4-10 shows the variation in primary current when the primary d.c supply voltage is modulated for different secondary resistive loads. The measurements were taken with an actual open loop setup with the Royer circuit as the transmitter and the parallel-hysteresis receiver coupled to it. The coupling factor between the primary and secondary coils was fixed to about 0.3.
The shape of the graph from the experimental data matches the curve in Figure 4-9. The dots in Figure 4-10 indicate the points when the secondary load is in regulation. As predicted, there is a change in the incremental rise in primary current when primary supply voltage is increased after the secondary achieves regulation. If this change in slope can be detected, the optimum operating point of the transmitter can be determined as the change in slope occurs when the primary delivers just enough power for the secondary to reach regulation. The primary supply voltage can be periodically modulated to detect the change in gradient which allows the transmitter to sense the power requirements of the receiver without any direct feedback signal from the receiver itself. This indirect method will improve the overall power efficiency by reducing standing losses in the system.

The indirect change-in-gradient method will be implemented and tested in the following chapters.
References


Chapter 5
Implementation

This section describes the design and construction of the primary unit with the indirect feedback method introduced in chapter 4. The indirect feedback method periodically modulates the supply voltage of the primary unit to find the optimum regulation point. Algorithm that performs the change in gradient detection and supply voltage modulation is also discussed in this chapter.

5.1 Setup

Figure 5-1 shows a setup implementing the indirect feedback solution. Supply voltage \( V_{supply} \) to the DC/AC converter (transmitter circuit) is provided by a DC/DC converter. The DC/DC converter’s output voltage is controlled by the Optimum Power Transfer Regulator (OPTR). OPTR modulates \( V_{supply} \) by varying the current through feedback resistors \( R_{FB1} \) and \( R_{FB2} \) and performs gradient detection by measuring the current through the sense resistor. Figure 5-2 provides a more detailed diagram of the OPTR.
A current sense amplifier in the OPTR measures the current ($I_{\text{supply}}$) consumed by the transmitter through the sense resistor $R_{\text{sense}}$. The output of the current sense amplifier is converted to a voltage by a resistor and sampled by the A/D converter. A Digital Logic and Memory unit stores the sampled values and calculates the gradient. The Digital Logic block contains the algorithm to detect the change in gradient and modulates the supply voltage ($V_{\text{supply}}$) accordingly.

![Figure 5-2: Closed Loop Transmitter with OPTR in more detail](image)

A D/A converter, connected at the output of digital logic unit, translates the digital representation of supply voltage modulation to a current which is injected to feedback resistors network to change the DC/DC converter’s output.
5.2 OPTR - Full Implementation

In the actual design of the primary unit, the digital logic and memory block was implemented using a complex programmable logic device (CPLD). The CPLD and the other analog blocks (DC/DC converter, current sensor, ADC & DAC) are combined to form the Optimal Power Transfer Regulator. Figure 5-3 shows how the signals to/from the CPLD interact with the rest of the analog circuitry. The signals will be explained in more detail in chapter 5.2.2. The CPLD sets RUN_BOOST to high to switch on the DC/DC Converter during power up. One end of the sense resistor (Rsense) is connected to the output of the DC/DC converter and the other end is connected to the supply nodes of the DC/AC converter in the transmitter circuit (i.e. Royer circuit). Current sensor measures the current through the sense resistor and converts the sensed current to a voltage. This voltage is fed to the positive analog input of the A/D Converter. The negative analog input of the A/D Converter is connected to ground. The A/D Converter used is a Linear Technology 12 bit ADC with 2.5V internal reference.

The CPLD will read the sampled signal from the A/D converter and use the Simple Slope Detection algorithm to set output current (I_{out}) of D/A converter. I_{out} is injected to the feedback resistors (R_{FB1} and R_{FB2}) and the output of the DC/DC Converter is modified. The process repeats.
5.2.1 Simple Slope Detection Algorithm

There are many ways to implement the slope detection algorithm. In the simplest algorithm, the supply voltage of the transmitter is periodically modulated to calculate the change in current consumption [1]. The calculated gradient is compared against a pre-programmed reference gradient to determine the optimum regulation point. Other advanced algorithms (e.g. binary search) can reach the optimal point more efficiently. For this thesis, the simple slope detection algorithm was implemented using MAX V CPLD from Altera Corporation and is described in detail in chapter 5.2.2. The detailed description of this simple slope detection algorithm follows.

Definitions:

- \( V_{\text{supply}} \) - Supply Voltage
- $V_{s,\text{max}}$ – Maximum Supply Voltage of Transmitter
- $V_{s,\text{min}}$ – Minimum Supply Voltage of Transmitter
- $\text{slope}_{\text{ref}}$ – Pre-programmed Reference Slope
- $\text{slope}_{\text{current}}$ – Variable

Initialization:

- The system is pre-calibrated and $\text{slope}_{\text{ref}}$ is pre-programmed into the CPLD. The minimum value for $\text{slope}_{\text{ref}}$ is the incremental change in current when the transmitter supply voltage is modulated with no load on the secondary side. $\text{slope}_{\text{ref}}$ is usually set to a value higher than the minimum to account for system random variability and non-idealities.
- $T_d$ is pre-programmed into the CPLD.

Sequence:

1. Set the supply voltage ($V_{\text{supply}}$) to $V_{s,\text{min}}$. Measure and store the current ($I_0$). Increase $V_{\text{supply}}$ by 1V. Measure and store the current ($I_1$).

2. Calculate the gradient at $V_{s,\text{min}}$ which is equal to $[I_1-I_0]$. Store this value in $\text{slope}_{\text{current}}$.

3. Compare $\text{slope}_{\text{current}}$ against a pre-programmed value $\text{slope}_{\text{ref}}$.
   
   - If $\text{slope}_{\text{current}}$ is smaller than $\text{slope}_{\text{ref}}$, the optimum supply is reached.
   - Otherwise, step $V_{\text{supply}}$ and calculate a new value for $\text{slope}_{\text{current}}$ till $\text{slope}_{\text{current}}$ is less than $\text{slope}_{\text{ref}}$. Store the current at new supply voltage [$V_{\text{supply}}$] in $I_1$ and the current at previous step [$V_{\text{supply}} - 1$] in $I_0$. Update $\text{slope}_{\text{current}}$ (which is equal to $I_1-I_0$)
If $\text{slope}_{\text{current}}$ is still greater than $\text{slope}_{\text{ref}}$ at $V_{\text{supply}} = V_{s_{\text{max}}}$, the optimum supply voltage is $V_{s_{\text{max}}}$ (the transmitter cannot deliver the power required by the load).

4. Once the optimum supply voltage is reached, the voltage stepping is paused for a pre-programmed time ($T_d$).

5. Once $T_d$ expires, the supply voltage is continuously stepped down, with the same step size as before (1V), until the gradient value ($\text{slope}_{\text{current}}$) increases above $\text{slope}_{\text{ref}}$.

6. The voltage is then stepped up again one or two steps until the gradient is less than $\text{slope}_{\text{ref}}$ again.

7. The cycle repeats with step 4.

Several factors need to be considered when choosing a value for $T_d$: settling time for both transmitter and receiver circuit after changing the transmitter supply voltage and the frequency at which receiver load changes (a factor specific to the application).

Figure 5-4 shows the sequence steps 1-4 during which the supply voltage is increased in fixed steps (1V) to determine the initial optimum point. Figure 5-5 shows the sequence steps 5-6 when the supply voltage is dithered about the previous optimum to adjust regulation point to any changes in receiver load.
Simple slope detection algorithm can also be used to detect the optimum regulation point for multiple loads. As seen in Figure 5-6, the algorithm will produce multiple gradient changes with the final gradient change corresponding to the optimum supply voltage for all loads.
5.2.2 CPLD Implementation

This section provides more details on the digital implementation of the simple slope algorithm and how the CPLD interacts with the analog blocks (DC/DC converter, A/D converter, D/A converter) on the primary side.

5.2.2.1 Overview of the State Machines

Figure 5-7 shows an overview of the individual state machines that constitute the simple slope detection algorithm. The diagram shows input/output signals of the CPLD that control the analog blocks, and the main signals inside the CPLD responsible for the state machines’ event changes.

Input signal CLK is provided by the 5MHz internal clock of MAX V CPLD. The entire system can be reset by the RESET input signal using a push button.

Control, which is the main state machine, triggers other state machines accordingly. When the system is powered up, control state machine powers up the DC/DC converter and after a delay triggers the Sweep/Dither state machine to find the optimum regulation point. The Sweep/Dither state machine performs the slope detection algorithm and when the optimum point is reached sends a done signal (regulation_done) to the Control state machine. Upon receiving the done signal, Control will start the \( T_d \) counter and pause the voltage stepping. After \( T_d \) seconds, Control will trigger the Sweep/Dither state machine and the process repeats.

The Sweep/Dither machine will change the supply voltage by triggering the D/A Converter state machine (trigger_dac signal). Some of the outputs of the D/A Converter machine are connected to the D/A Converter IC (LTC 1329). The D/A Converter state machine will also trigger the Delay A/D Converter state machine using the trigger_delay signal. This Delay A/D Converter state machine is simply a delay block. The delay block lets the modified supply voltage settle before triggering the A/D Converter state machine (Sample signal) to sample the new current consumption of the transmitter circuit. After sampling the current, the A/D
Converter machine triggers the Counter state machine, which processes the binary representation of the current and displays the value on a 4 digit 7 segment LED display.

The Sweep/Dither machine can also trigger the A/D converter machine without going through the D/A machine (trigger\_io).

![State Diagram](image)

**Figure 5-7: Overview of State Diagrams**

### 5.2.2.2 Description of the State Machines

The state machines introduced in the previous section (5.2.2.1) are explained in more detail in this section.
Control State Machine

Figure 5-8: Control SM

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power up</td>
<td>The system begins with this state. All blocks are switched on (A/D, D/A, CPLD and display unit) except for the DC/DC converter (LTC3789). All counters are reset. If the system reset push button is on at any point in this state machine, the current state will be updated to <strong>Power up</strong> state.</td>
</tr>
<tr>
<td>Power boost</td>
<td>DC/DC converter is powered up.</td>
</tr>
<tr>
<td>Regulate</td>
<td>This state outputs a signal that triggers the Sweep/Dither State Machine. When the Sweep/Dither State Machine is done (that is after optimum regulation point is achieved), it updates the variable <strong>regulation_done</strong> from low to high which in turn causes a transition in the Control State Machine to the <strong>Busy</strong> state.</td>
</tr>
<tr>
<td>Busy</td>
<td>The Control State Machine will stay in this state for $T_d$ seconds and go back to the <strong>Regulate</strong> state to check for a new optimum point. $T_d$ is a predetermined delay.</td>
</tr>
</tbody>
</table>
**Sweep/Dither State Machine**

![State Machine Diagram]

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ready</td>
<td>Sweep/Dither machine begins with this state. trigger_sd is the signal from the Control State Machine to trigger the Sweep/Dither State Machine.</td>
</tr>
<tr>
<td>setup</td>
<td>Initial $I_0$ and $I_1$ are read.</td>
</tr>
<tr>
<td>calculate slope</td>
<td>Slope is calculated depending whether voltage was increased or decreased to read $I_1$. The dither_on signal is set to low at the Ready state. Therefore, the first time the state machine goes through this state, the next state is Sweep. After the first optimum voltage is achieved, dither_on is set to high. Therefore the second time onwards, the state machine will go from this state to the Dither state.</td>
</tr>
<tr>
<td>sweep</td>
<td>Voltage is increased in pre-determined steps depending on the slope. max_supply is a signal from the DAC State Machine that lets this Sweep/Dither State Machine knows whether maximum supply voltage is reached. Sets regulation variable to high if optimum point is reached. After reaching regulation, the supply voltage is set to a value little higher than the optimum</td>
</tr>
<tr>
<td>(Blank)</td>
<td>voltage to account for system random variability and non-idealities. Therefore, the next state is Trigger D/A converter unless maximum supply voltage is reached.</td>
</tr>
<tr>
<td>(Blank)</td>
<td>dither</td>
</tr>
<tr>
<td>(Blank)</td>
<td>trigger d/a converter</td>
</tr>
<tr>
<td>(Blank)</td>
<td>read I_supply</td>
</tr>
<tr>
<td>(Blank)</td>
<td>done control</td>
</tr>
</tbody>
</table>

**D/A Converter State Machine**

![D/A Converter State Machine Diagram]

Figure 5-10: D/A Converter SM
<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>reset dac</td>
<td>DAC counter is reset. CS pin is set high.</td>
</tr>
<tr>
<td>setup</td>
<td>Delay block</td>
</tr>
<tr>
<td>output din</td>
<td>Outputs DIN serially in 8 clock cycles</td>
</tr>
<tr>
<td>busy</td>
<td>Delay block to satisfy Tcsl0 timing requirement. CS pin has to be pulled low for at least Tcsl0 seconds before setting it high in the next state. (Refer to LTC1329 datasheet)</td>
</tr>
<tr>
<td>cs high</td>
<td>CS pin has been pulled high for a minimum time before next conversion</td>
</tr>
<tr>
<td>wait</td>
<td>Ensures previous trigger (trigger_dac) to the DAC is low</td>
</tr>
<tr>
<td>ready</td>
<td>Sweep/Dither State Machine sends the input signal trigger_dac to change the dac counter</td>
</tr>
</tbody>
</table>

**Delay A/D Converter**

![Diagram of Delay A/D Converter](image)

**Figure 5-11: Delay A/D Converter**

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ready</td>
<td>Initial state. There are two input signals. trigger_delay is an output from DAC machine and trigger_to is an output from sweep/dither machine.</td>
</tr>
<tr>
<td>trigger adc</td>
<td>Outputs sample signal for ADC machine</td>
</tr>
</tbody>
</table>
**A/D Converter State Machine**

**Figure 5-12: A/D Converter SM**

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ready</td>
<td>Initial state. Input signal sample is the output of the delay ADC state machine</td>
</tr>
<tr>
<td>conversion high</td>
<td>CONV pin to ADC is set high</td>
</tr>
<tr>
<td>conversion busy</td>
<td>CONV pin to ADC is pulled low</td>
</tr>
<tr>
<td>read data</td>
<td>DOUT pin read serially. ADC outputs data sampled from the previous conversion. Therefore, CONV pin of the ADC needs to be triggered twice to read current data. When read_ready='0', CONV pin is already triggered once so the next state is set to busy and read_ready is pulled high. During the second cycle, read_ready is set to low and the next state is set to ready (after T seconds delay to ensure that the previous sample signal is low)</td>
</tr>
<tr>
<td>busy</td>
<td>read_ready is set high the next state is conversion high</td>
</tr>
</tbody>
</table>
The closed loop transmitter board with the OPTR was built and tested (refer to Appendix C for complete schematic of the board and Appendix B for the code for CPLD). This setup is discussed in more detail in chapter 6.

References

Chapter 6
Results

The closed loop transmitter implemented in chapter 5 was built and tested using the Royer circuit (described in chapter 3) and the parallel-hysteresis receiver (described in chapter 2). This section describes the new setup for the closed loop system and discusses the power efficiency measurements obtained from the new adaptive transmitter. This section also summarizes the project and presents suggestions for future work.

6.1 Closed Loop Transmitter

![Diagram of Closed Loop Transmitter - OPTR and DC/DC Converter](image)

**Figure 6-1: Closed Loop Transmitter - OPTR and DC/DC Converter**
Figure 6-1 shows the new board that contains the closed power control circuitry (Optimal Power Transfer Regulator implemented in chapter 5) and the DC/DC converter (The full schematic is available in Appendix C). The DC/DC converter used is a Linear Technology buck-boost converter (LTC3789). The buck-boost and the OPTR module is powered by an external supply which is set to 10V. The external power supply is the main and only supply to the transmitter. The output of the board (labeled to TX circuit Royer in Figure 6-1), which is the output of the buck-boost converter (LTC3789), is connected to the input supply nodes of the royer circuit. The current consumption of the royer circuit can be read off from the 4 digit display (which is in mA).

The $T_d$ timer in the simple slope detection algorithm implemented in the CPLD checks for any changes in secondary load every 2 seconds. The $T_d$ delay can be set for longer but in this implementation is limited by the number of logic cells in CPLD. The range of the voltage sweep of the transmitter is set from 5 to 20V. The minimum value of the range is 5V because the current sense amplifier requires at least 4.5V to operate. The voltage step size for the initial voltage sweep and later dithering is set to 0.5V. The actual supply voltage is set to a 1V more than the optimum regulation point determined by the algorithm to ensure that any slight variations in the system does not cause the secondary load to go out of regulation.

$slope_{ref}$ variable in simple slope detection algorithm is programmed through the CPLD simply by changing code. $slope_{ref}$ can also be programmed by adjusting the 21 turn potentiometer labeled in Figure 6-1. The output of the potentiometer is sent to an A/D converter connected to the CPLD.

The schematics for the closed loop transmitter with the receiver circuit (are shown in detail in Appendix C. The transmitter and receiver coils are positioned as shown in figure 6-2 to keep the coupling factor constant. Sheets of non-magnetic material are placed between the two coils to keep the distance in between the coils constant during testing.
The detuning-hysteresis (with PWM) receiver board from chapter 2 is used in the setup. One external power supply (V_{ref} in Figure C-1 in Appendix C) provides the 1V reference voltage required by the receiver board. Second external supply (V_{ext} in Figure C-1 in Appendix C) powers the operational amplifier, comparator and buffer stage leading to the gate of the MOSFET that switches the detuning capacitor. These components are connected to a separate supply in this setup but can be connected to the rectified output of the receiver. Connecting a separate supply ensures each individual supply node is free from the switching noise caused by detuning, which eases testing and debugging this prototype. The function generator (*label) provides the 30 kHz saw-tooth triangle wave to the hysteretic comparator for pulse width modulation.

When the above setup is run, the closed loop transmitter performs the initial voltage sweep and dithers responding to changes in secondary load. When there is no load at the receiver, the optimum supply voltage to the Royer circuit is set to the minimum value (5V) because the calculated slope in the Simple Slope Detection algorithm is less than \text{sloperef}. The optimum supply voltage is set to the maximum value (20V) when the transmitter is unable to provide the power required by large current loads to stay in regulation. However, when the current load is
too big (that is for extremely small resistive loads), the output of the receiver is similar to being shorted to ground. Therefore, the $Q$ of the receiver LC tank is much smaller and the receiver does not absorb enough power from the transmitter. Hence, if the current load is increased beyond a maximum value, the transmitter’s optimum regulation point dithers around the minimum supply voltage of 5V instead of the maximum supply value (20V). However, the system can be calibrated to determine this maximum secondary load level.

6.2 Power Efficiency Comparison

The overall power efficiency (i.e $P_{out,secondary}/P_{in,primary}$) was calculated for different loads using the setup described in the previous section. The data from the closed loop system was plotted against data from the open loop system in Figure 6-3 and Figure 6-4. Data for the open loop system was obtained by setting the supply voltage of the primary (which is the output of buck-boost) to the maximum voltage (20V). The secondary regulation voltage was at 11.5V.

![Overall Power Efficiency vs Secondary Load](image)

*Figure 6-3: Overall Power Efficiency vs Secondary Load*
As seen in Figure 6-3, both open and closed loop transmitter systems have an overall efficiency of 0% when there is no load at the receiver. When the receiver load current is increased, eventually the two curves converge. The area enclosed between the two curves shows the improvement in power efficiency obtained with the closed loop transmitter by operating at the optimum regulation point.

Figure 6-4 shows the power output of the primary for different receiver loads. Again, the two curves will converge for higher current loads. The area in between the two plots corresponds to the power wasted by running the transmitter open loop.

![Power Output of the Primary vs Secondary Load](image)

**Figure 6-4: Power Output of the Primary vs. Secondary Load**

From these results, it is evident that the closed loop system yields a significant reduction in power losses in comparison to the open loop system by adapting to the power requirements of the secondary. Hence, the overall power efficiency has improved across the entire range of power delivered to the secondary.
6.3 Summary

When the power delivered by primary is fixed (the system is running open loop) standby loses in the system degrade the power efficiency. The primary needs to continuously adjust the power delivery to match the requirements of the secondary to minimize standby loses. For this, implementation of a feedback mechanism between the secondary and primary is inevitable. The objective of this thesis was to improve the power efficiency of a wireless energy system using feedback.

The existing feedback mechanisms use direct feedback from the secondary to control the power delivered by the primary circuit. This thesis presented a novel feedback method to improve the system performance. The method implemented in this thesis does not require any direct feedback signal from the secondary. The feasibility of this indirect feedback was tested by building a wireless power transfer system with the transmitter circuit from chapter 3 modified to include an additional unit called the Optimal Power Transfer Regulator (OPTR). OPTR, implemented in chapter 5, contains the circuitry and the algorithm for the indirect feedback mechanism. The results obtained from the new closed loop system demonstrate that the novel indirect feedback method significantly improves power efficiency in comparison to the open loop system.

6.4 Future Work

Some further work can be done to increase the power efficiency of the current setup. If the current secondary circuit (parallel-hysteresis receiver) is replaced by the receiver with a buck converter (see section 2.2), the power efficiency can be increased. The buck converter is more efficient at regulating the secondary load compared the tuning controller: a significant amount of power is dissipated in the ESR of the detuning capacitor when the detuning MOSFET is switched on.
The Simple Slope Detection algorithm introduced in this thesis is the most basic algorithm to implement the indirect feedback method. The algorithm can be improved to reach the optimal regulation point faster using binary search instead of sweeping the supply voltage stepwise. In the current setup, the reference gradient ($\text{slope}_{\text{ref}}$) is set by the user either by directly programming the CPLD or using the potentiometer to program the CPLD. This step requires user calibration that is different for different systems. An autonomous reference slope detection algorithm can be included to implement a more robust transmitter.

If any metal object is placed in the transmitting coil’s magnetic field, power will be wasted due to the dissipation in metal and can lead to overheating. By incorporating metal object detection and temperature protection features, the current system can be improved to operate more safely and reliably.
Appendix A
Test Setup for Open Loop System

Figure A-1 show the schematic of the overall setup of the transmitter circuit (Royer) coupled to the receiver circuit (hysteretic tuning controller with PWM) which was used to measure the open loop power efficiencies shown in figure 4-1 (chapter 4). $V_{dc}$ which is the supply voltage to the Royer was fixed at 20V. As seen in figure A-2, the coupling between the primary and secondary coils was fixed by keeping the position constant. The coupling factor is approximately 0.3.

The same setup was used to obtain data for figure 4-10 (chapter 4). The primary supply voltage, which is the DC source $V_{dc}$ in the schematic, was increased from 0 to about 15V. The corresponding current from $V_{dc}$ supply was measured.

![Figure A-1: Full schematic of the test setup](image)

92
Primary Circuit (Royer) Primary LC tank capacitor Ct-tx Primary Coil Lt-tx Receiver circuit with hysteretic tuning controller (& PWM)

Secondary Coil Lt-rx

Secondary LC tank capacitor Ct-rx

Figure A-2: Picture of the setup
Appendix B
State Machines: VHDL Code For CPLD

The VHDL code for the state machines described in chapter 5 is presented in this section.

<table>
<thead>
<tr>
<th>Name of state machine in chapter 5</th>
<th>Name in the code file</th>
</tr>
</thead>
<tbody>
<tr>
<td>control</td>
<td>new_brain</td>
</tr>
<tr>
<td>sweep/dither</td>
<td>sweep_dither2</td>
</tr>
<tr>
<td>D/A converter</td>
<td>dac_counter</td>
</tr>
<tr>
<td>delay A/D converter</td>
<td>delay_adc2</td>
</tr>
<tr>
<td>A/D converter</td>
<td>adc_conv2</td>
</tr>
</tbody>
</table>

Code files counter, counter_top and display_drive3 are used to display the primary supply current on the 7 segment LED display and are not described in chapter 5. sweep_dither_top combines all the state machines and display driver code.
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity sweep_dither_top is
Port (reset: in std_logic;
run_boost : out std_logic;
run_dac : out std_logic;
clk_dac : out std_logic;
run_adc : out std_logic;
shdn : out std_logic;
run_adc : in std_logic;
conv_adc : out std_logic;
clk_adc : out std_logic;
led_out : out STD_LOGIC_VECTOR (6 downto 0);
led_out : out std_logic;
shdn : out std_logic;
din_dac : out std_logic;
clk_adc : out std_logic;
adc : out std_logic;
clk_adc : out std_logic;
led_out : out std_logic;
anl_out : out std_logic;
an2_out : out std_logic;
an3_out : out std_logic;
an4_out : out std_logic
);
end sweep_dither_top;

architecture Behavioral of sweep_dither_top is
signal int_clk : std_logic;
signal trigger_dac : std_logic;
signal sig_in : std_logic;
signal delta_in : std_logic_vector (7 downto 0) := "00000000";
signal trigger_i0 : std_logic := '0';
signal reg_trigger : std_logic := '0';
signal reg_done : std_logic := '0';
signal max_supply : std_logic := '0';
signal min_supply : std_logic := '0';
signal trigger_adc : std_logic := '0';
signal dout_out : std_logic_vector (11 downto 0) := "000000000000";
signal trig_count : std_logic := '0';
signal clk0 : std_logic;
signal clk1 : std_logic;
signal clk2 : std_logic;
signal clk3 : std_logic;
signal do_out : std_logic_vector (3 downto 0);
signal d1_out : std_logic_vector (3 downto 0);
signal d2_out : std_logic_vector (3 downto 0);
signal d3_out : std_logic_vector (3 downto 0);

component int_osc -- internal 5MHz clock
PORT (oscena IN STD_LOGIC ;
osc : OUT STD_LOGIC
);
end component;
component newbrain

Port ( CLK : in std_logic;
        RESET: in std_logic;
        DONE_REQ : IN STD_LOGIC;
        TRIG_REQ : out std_logic;
        RUN : OUT STD_LOGIC
    );
end component;

component sweep dither2

Port ( CLK : in std_logic;
        TRIG DITHER : in std_logic;
        SUPPLY_MIN : IN STD_LOGIC;
        SUPPLY_MAX : IN STD_LOGIC;
        ADC DIN : IN STD_LOGIC VECTOR (11 downto 0);
        TRIG ADC : OUT STD_LOGIC;
        '0' = + '1' =
    );
end component;

component dac counter

Port ( CLK IN : in STD_LOGIC;
        RESET : in STD_LOGIC;
        PB1 : IN STD_LOGIC;
        UP_DOWN : IN STD_LOGIC;
        DELTA: IN STD LOGIC VECTOR (7 DOWNTO 0);
        SUPPLY_MIN : OUT STD LOGIC;
        SUPPLY_MAX : OUT STD_LOGIC;
        CS : OUT STD_LOGIC;
        CLK DAC : OUT STD LOGIC;
        DIN : out STD LOGIC;
        SHDN : OUT STD_LOGIC
    );
end component;

component delay adc2

Port ( CLK : in std_logic;
        TRIG DELAY : In std logic;
        TRIG IO : in std logic;
        TRIG_ADC : out std_logic
    );
end component;

component adc conv2

Port ( RESET : in STD_LOGIC;
        SAMPLE : IN STD_LOGIC;
        CLK IN: IN STD LOGIC;
        CLK OUT : OUT STD LOGIC;
        CONV_OUT : OUT STD LOGIC;
        DIN : IN STD LOGIC;
        DOUT : OUT STD LOGIC VECTOR (11 DOWNTO 0);
        TRIG COUNTER : OUT STD LOGIC
    );
end component;

5MHz clock
reset any counters
run pin to boost

-- output from a/d after changing
-- voltage
-- done signal to brain
-- input to dac
-- increment/decrement dac counter

-- increase or decrease dac counter
-- change in count

the supply voltage

-- trigger dac to sweep supply

-- 5MHz clock
component counter_top --
  Port (CLK_IN : in STD_LOGIC;
    RESET : in STD_LOGIC;
    TRIG_IN : IN STD_LOGIC;
    DIN : IN STD_LOGIC_VECTOR (11 downto 0);
    D0 : out STD_LOGIC_VECTOR (3 downto 0);
    CLK_OUT : out STD_LOGIC);
end component;

component counter
  Port (CLK_IN : in STD_LOGIC;
    RESET : in STD_LOGIC;
    RESET_COUNT : IN STD_LOGIC;
    D0 : out STD_LOGIC_VECTOR (3 downto 0);
    CLK_OUT : out STD_LOGIC);
end component;

component display_drive3 is
  Port (D0 : in STD_LOGIC_VECTOR (3 downto 0);
    D1 : in STD_LOGIC_VECTOR (3 downto 0);
    D2 : in STD_LOGIC_VECTOR (3 downto 0);
    D3 : in STD_LOGIC_VECTOR (3 downto 0);
    CLK_IN : in STD_LOGIC;
    LED : out STD_LOGIC_VECTOR (6 downto 0);
    LED_dot : out STD_LOGIC;
    AN1 : out STD_LOGIC;
    AN2 : out STD_LOGIC;
    AN3 : out STD_LOGIC;
    AN4 : out STD_LOGIC);
end component;

begin
  int_osc_inst : int_osc PORT MAP (oscena => reset,
    osc => int_clk
  );

  newbrain_inst : newbrain PORT MAP (CLK => int_clk,
    RESET => reset,
    counters
    DONE_REG => reg_done,
    TRIG_REG => reg_trigger,
    RUN => run_boost
  );

  sweep_dither2_inst : sweep_dither2 PORT MAP (CLK => int_clk,
    RESET => reset,
    TRIG_DITHER => reg_trigger,
    SUPPLY_MAX => max_supply,
    SUPPLY_MIN => min_supply,
    ADC_DIN => dout_out,
    a/d after changing the supply voltage
    TRIG_DAC => trigger_dac,
    to sweep supply voltage
    TRIG_ADC => trigger_i0,
    DONE => reg_done,
    -- run pin to
    5MHz clock
    -- reset any
    -- run pin to
    -- output from
    -- trigger dac
    -- done signal
to brain
input to dac
increment/decrement dac counter '0' = + '1' = -

dac_counter_inst : dac_counter PORT MAP (CLK_IN => int_clk,
RESET => reset,
PB1 => trigger_dac,
UP_DOWN => sign_in,
);

delay_adc2_inst : delay_adc2 PORT MAP (CLK => int clk,
RESET => reset,
TRIG_DELAY => trigger_dac,
TRIG_I0 => trigger_i0,
TRIG_ADC => trigger_adc
);

adc_conv2_inst : adc_conv2 PORT MAP (RESET => reset,
SAMPLE => trigger_adc,
CLK_IN => int_clk,
CLK_OUT => clk_adc,
CONV_OUT => conv_adc,
DIN => din_adc,
DOUT => dout_out,
TRIG => dout_count,
);

counter_top_inst : counter_top PORT MAP (CLK_IN => int_clk,
RESET => reset,
TRIG_IN => trigger_count,
DIN => dout_out,
CLK_OUT => clk0
);

counter1_inst : counter PORT MAP (CLK_IN => clk0,
RESET => reset,
RESET_COUNT => trigger_count,
D0 => d0_out,
CLK_OUT => clk1
);

counter2_inst : counter PORT MAP (CLK_IN => clk1,
RESET => reset,
RESET_COUNT => trigger_count,
D0 => d2_out,
CLK_OUT => clk2
);

counter3_inst : counter PORT MAP (CLK_IN => clk2,
SUPPLY_MIN => min_supply,
SUPPLY_MAX => max_supply,
CS => cs_out,
CLK_DAC => clk_dac,
DIN => din_dac,
SHDN => shdn
);

delayadc2_inst delayadc2 PORT MAP (CLK => int clk,
RESET => reset,
TRIGDELAY => trigger_dac,
TRIG_10 => trigger_iG,
TRIGADC => trigger_adc
);

counter_top_inst : counter_top PORT MAP (CLK_IN => int_clk,
RESET => reset,
TRIG_IN => trigger_count,
DIN => dout_count,
CLK_OUT => clk0
);

counter1_inst : counter PORT MAP (CLK_IN => clk0,
RESET => reset,
RESET_COUNT => trigger_count,
D0 => d0_out,
CLK_OUT => clk1
);

counter2_inst : counter PORT MAP (CLK_IN => clk1,
RESET => reset,
RESET_COUNT => trigger_count,
D0 => d2_out,
CLK_OUT => clk2
);

counter3_inst : counter PORT MAP (CLK_IN => clk2,
DEL => delta_in,
SIGN => sign_in
);

counter_top_inst : counter_top PORT MAP (CLK_IN => int clk,
RESET => reset,
TRIG_IN => trigger_count,
DIN => dout_count,
CLK_OUT => clk0
);

counter1_inst : counter PORT MAP (CLK_IN => clk0,
RESET => reset,
RESET_COUNT => trigger_count,
D0 => d0_out,
CLK_OUT => clk1
);

counter2_inst : counter PORT MAP (CLK_IN => clk1,
RESET => reset,
RESET_COUNT => trigger_count,
D0 => d2_out,
CLK_OUT => clk2
);

counter3_inst : counter PORT MAP (CLK_IN => clk2,
RESET => reset,
RESET_COUNT => trig_count,
D0 => d3_out,
CLK_OUT => clk3
);

display_drive3_inst : display_drive3 PORT MAP ( D0 => d0_out,
D1 => d1_out,
D2 => d2_out,
D3 => d3_out,
CLK_IN => int_clk,
LED => led_out,
LED_dot=> led_dot_out,
AN1 => an1_out,
AN2 => an2_out,
AN3 => an3_out,
AN4 => an4_out);
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity newbrain is
    Port ( CLK : in std_logic; -- 5MHz clock
            RESET: in std_logic ; -- reset any counters
            DONE_REG : IN STD_LOGIC;
            TRIG_REG : out std_logic;
            RUN : OUT STD_LOGIC -- run pin to boost
        );
end newbrain;

architecture Behavioral of newbrain is

type brain_machine is (powerup,
    resetsys,
    powerboost,
    regulate,
    busy
);

signal state: brain_machine:= power_up;
attribute ferm_encoding: string;
attribute fai_encoding of state: signal is "compact"; -- This tells XST to use binary encoding

signal delaycount: integer range 0 to 5000000:=0;

signal trigger : std_logic :='0';

signal boost_on : std_logic :='0'; -- Boost - no power
signal run_boost : std_logic :='0'; -- run pin to Boost - no power

begin
    process (CLK, RESET, DONE_REG)
    begin
        if (RESET = '0') then
            delay_count <=0;
            state<= power_up;
            boost_on <='0';
            run_boost <='0';
        elsif (CLK = '1' and CLK'event) then
            case state is
                when power_up =>
                    if (delay_count = 5000000) then --everyone powers up except buck-boost 1
                        state <= reset_sys;
                        delay_count <=0;
                        boost_on <='0';
                    else
                        state <= power_up;
                        run_boost <='0';
                        delay_count <= delay_count +1;
                    end if;
                when reset_sys =>
                    if (boost_on = '0') then

    end process;
end Behavioral;
state <= power_boost;
else
state <= regulate;
end if;

when power_boost =>
if (delay_count = 5000000)
then
state <= regulate;
delay_count <=0;
boost_on <='1';
else
state <= power_boost;
run_boost <='1';
delay_count <= delay_count +1;
end if;

when regulate =>
if (delay_count = 0)
then
trigger <='1';
delay_count <= delay_count +1;
elif (delay_count = 3)
then
if (DONE_REG = '1')
then
state <= busy;
delay_count <=0;
else
trigger <='0';
state <=regulate;
end if;
else
delay_count <= delay_count +1;
end if;
when busy =>
if (delay_count =50000000)
then
state <= regulate;
delay_count <=0;
else
state <= busy;
delay_count <= delay_count+1;
end if;
end case;
end if;
end process;
TRIG_REG <= trigger;
RUN <=run_boost;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity sweep_dither2 is
    Port (CLK : in std_logic; -- 5MHz clock
            RESET: in std_logic;
            TRIG_DITHER : in std_logic;
            SUPPLY_MAX : IN STD_LOGIC;
            SUPPLY_MIN : IN STD_LOGIC;
            ADC_DIN : IN STD_LOGIC_VECTOR (11 downto 0); -- output from a/d after changing the supply voltage
            TRIG_DAC : out std_logic; -- trigger dac to sweep supply voltage
            TRIG_ADC: OUT STD_LOGIC;
            DONE : OUT STD_LOGIC; -- done signal to brain
            DEL : OUT STD_LOGIC_VECTOR (7 downto 0); -- input to dac
            SIGN : OUT STD_LOGIC_VECTOR (11 downto 0); -- increment/decrement dac counter '0' = + '1' =
            DEL: OUT STD_LOGICVECTOR (7 downto 0);
            SIGN: OUT STD_LOGIC;
            DEL: OUT STD_LOGIC_VECTOR (7 downto 0);
            SIGN: OUT STD_LOGIC;
    end sweep_dither2;

architecture Behavioral of sweep_dither2 is

        type dither_machine is (ready, read_initial_i0, setup, trigger_dac, calc_slope, select_comp_slope, comp_slope_sweep, comp_slope_dither, read_iout, done_brain);

        signal state: dither_machine:= ready;
        attribute fsm_machine:= ready;
        attribute fsm_encoding: string;
        attribute fsm_encoding of state: signal is "compact"; -- This tells XST to use binary encoding

        --delay variables
        signal delay_count : integer range 0 to 450005:=0;

        --slope detection etc.
        signal 10_msb : std_logic_vector (11 downto 0) :="000000000000";
        signal 11_msb : std_logic_vector (11 downto 0) :="000000000000"; -- latest current value read
        signal slope_current : std_logic_vector (11 downto 0) := "000000000000";
        signal prev_slope : std_logic_vector (1 downto 0) :="00"; -- '00' = first slope measurement, '01' = current_slope > ref, '10' = current_slope < ref
        constant iref : std_logic_vector (11 downto 0) := "000000111100"; -- ***** come back here

        --inputs to dac
        signal delta : std_logic_vector (7 downto 0) := "00000101"; -- delta change in dac output current (for voltage step sizes)
        signal dac_sign : std_logic :='0'; -- '0' = increment, '1' = decrement

        --output signals
        signal regulation : std_logic :='0';
        signal setup_done : std_logic :='0';
        signal brain_done : std_logic :='0';
signal trigger : std_logic := '0';
signal trigger_adc : std_logic := '0';
signal dither_on : std_logic := '0';
begin
process (CLK, RESET, TRIG_DITHER, ADC_DIN, SUPPLY_MAX, SUPPLY_MIN)
begin
if (RESET = '0') then
  delay_count <= 0;
  trigger <= '0';
  brain_done <= '0';
  state <= ready;
  setup_done <= '0';
  regulation <= '0';
  dither_on <= '0';
  trigger_adc <= '0';
elsif (CLK = '1' and CLK'event) then
  case state is
  when ready =>
    if (TRIG_DITHER = '1') then
      state <= read_initial_i0;
      delay_count <= 0;
      setup_done <= '0';
      prev_slope <= "00";
      i0_msb <= "0000000000000000";
      i1_msb <= "0000000000000000";
    else
      state <= ready;
    end if;
  when read_initial_i0 =>
    if (delay_count = 0) then
      trigger_adc <= '1';
      state <= read_initial_i0;
      delay_count <= delay_count + 1;
    elsif (delay_count = 2) then
      trigger_adc <= '0';
      delay_count <= delay_count + 1;
    elsif (delay_count = 5000000) then
      delay_count <= 0;
      state <= setup;
      i0_msb <= ADC_DIN;
    else
      delay_count <= delay_count + 1;
    end if;
  when setup =>
    if (delay_count = 4500000) then
      -- after --s read iout
      delay_count <= 0;
      if (setup_done = '0') then
        state <= trigger_adc;
        if (dither_on = '0') then
          dac_sign <= '0';
        else
          if (SUPPLY_MIN = '1') then
            dac_sign <= '0';
          else
            dac_sign <= '1';
          end if;
        end if;
      else
        dac_sign <= '1';
      end if;
    end if;
end case;
end if;
end process;
end begin;
end if;
end if;
delta <= "00000101";
else
il_msb <= ADC_DIN;
state <=calc_slope;
end if;
else
delay_count <= delay_count +1;
end if;
when calc_slope =>
state <= select_comp_slope;
if (dac_sign = '0') then
if (il_msb>i0_msb) then
slope_current <= il_msb-i0_msb;
else
slope_current <= "000000000000";
end if;
else
if (il_msb>i0_msb) then
slope_current <= "000000000000";
else
slope_current <= i0_msb-il_msb;
end if;
end if;
when select_comp_slope =>
if (dither_on = '0') then
state <= comp_slope_sweep;
else
state <= comp_slope_dither;
end if;
when comp_slope_sweep =>
if (SUPPLY_MAX = '1') then
regulation <= '0';
state <= done_brain;
delay_count <=0;
else
state<=trigger_dac;
delay_count <=0;
dac_sign <= '0';
delta <= "00000101";
if (slope_current <iref) then
regulation <= '1';
end if;
end if;
when comp_slope_dither =>
if (prev_slope ="00") then
slopes to determine current position
if (slope_current > iref) then
if (SUPPLY_MAX = '1') then
regulation <= '0';
state <= done_brain;
delay_count <=0;
else
state <= trigger_dac;
prev_slope <="01";
dac_sign <= '0';
delta <= "00000101";
end if;
delay_count <= delay_count +1;
end if;
enend if;
enend if;
enend if;
enend if;
enend if;
end if;
else
    state <= trigger_dac;
    prev_slope <= "10";
    dac_sign <= '1';
    delta <= "00000101";  -- by -0.5V **** come back here
end if;
else
    if (dac_sign = '0') then
        if (slope_current > iref) then
            if (SUPPLY_MAX = '1') then
                regulation <= '0';
                state <= done_brain;
                delay_count <=0;
            else
                state <= trigger_dac;
                prev_slope <= "01";
                dac_sign <= '0';
                delta <= "00000101";  -- sweep/dither up
            end if;
        end if;
    else
        if (slopecurrent <iref) then
            if (SUPPLY_MIN = '1') then
                state <= done_brain;
                regulation <= '0';
                delay_count <=0;
            else
                state <= trigger_dac;
                dac_sign <= '1';
                delay_count <=0;
                delta <= "00000101";
            end if;
        end if;
    end if;
    else
        if (slope_current < iref) then
            if (SUPPLY_MIN = '1') then
                state <= done_brain;
                regulation <= '0';
                delay_count <=0;
            else
                state <= trigger_dac;
                dac_sign <= '1';
                delay_count <=0;
                delta <= "00000101";
            end if;
        end if;
    end if;
end if;
end if;
else
    if (slope_current > iref) then
        if (prev_slope ="01") then
            if (prevyslope ="10") then
                regulation <= 'l';
                dac_sign <= '0';
                delta <= "00001111";
            else
                dac_sign <= '0';
                delta <= "00000101";
            end if;
        end if;
    end if;
end if;
end if;
end if;
end if;
end if;
end if;
end if;
end if;
end if;
end if;
end if;
when trigger_dac =>
  if (delay_count = 0) then
    trigger <= '1';
    delay_count <= delay_count + 1;  
  elsif (delay_count = 2) then  -- trigger_dac signal high only for 3 clk cycles
    trigger <= '0';
    delay_count <= 0;
    if (setup_done = '0') then  -- still in setup process. i0 initialized.
      next if
      state <= setup;
      setup_done <= '1';
    elsif (regulation = '1') then  -- load in regulation. supply voltage is increased by pre-determined delV to ensure regulation
      state <= done_brain;
      regulation <= '0';
      dither_on <= '1';
    else
      state <= read_iout;
    end if;
  else
    delay_count <= delay_count + 1;
  end if;
when read_iout =>
  if (delay_count = 4500000) then
    delay_count <= 0;
    i0_msb <= i1_msb;
    i1_msb <= ADC_DIN;
    state <= calc_slope;
  else
    delay_count <= delay_count + 1;
  end if;
when done_brain =>  -- tells the brain the voltage sweep is done and at optimum point (that is at pre-programed iref)
  if (delay_count = 2) then
    delay_count <= 0;  -- brain_done signal high only for 3 clk cycles
    state <= ready;
    brain_done <= '0';
  else
    state <= done_brain;
    brain_done <= '1';
    dither_on <= '1';
    delay_count <= delay_count + 1;
  end if;
end case;
end if;
end process;
TRIG_DAC <= trigger;
TRIG_ADC <= trigger_adc;
DONE <= brain_done;
SIGN <= dac_sign;
DEL <= delta;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity daccounter is
  Port ( CLKIN : in STD_LOGIC;
         RESET : in STD_LOGIC;
         PBl : IN STD_LOGIC;
         UPDOWN : in STD_LOGIC;
         DELTA: IN STD_LOGIC_VECTOR (7 DOWNTO 0);
         SUPPLY_MIN : OUT STD_LOGIC;
         SUPPLY_MAX : OUT STD_LOGIC;
         CS : OUT STD_LOGIC;
         CLK_DAC : OUT STD_LOGIC;
         DIN : out STD_LOGIC;
         SHDN : OUT STD_LOGIC
      );
end daccounter;

architecture Behavioral of daccounter is

  type dac_machine is (reset_dac,
                       ready,
                       setup,
                       read_din,
                       busy,
                       cs_high,
                       wait_dac
                     );

  signal state: dac_machine:= reset_dac;
  attribute fsm_encoding: string;
  attribute fsm_encoding of state: signal is "compact"; -- This tells XST to use binary encoding

  signal count: std_logic_vector (7 downto 0):= "10100001"; --161
  constant count_ref: std_logic_vector (7 downto 0):= "10100001"; --161
  signal min : std_logic :='0';
  signal max : std_logic :='0';

  signal clk_out: std_logic :='0';
  signal cs_out : std_logic :='1';
  signal din_out : std_logic :='0';

  signal delay_count : integer range 0 to 6 :=0;
  signal clk_cyc : integer range 0 to 8 :=0;
  signal busy_count : integer range 0 to 50 :=0;
  signal Tcks : integer range 0 to 2:=0;

begin

  process (CLK_IN, RESET, PBl, DELTA, UP_DOWN)
  begin
    if (RESET='0') then
      state <= reset_dac;
      cs_out <= '1';
      busy_count <=0;
      clk_out <= '0';
      Tcks <=0;
  end if;
end process;

begin
  process (clk_cyc)
  begin
    if (clk_cyc='1') then
      busy_count <= busy_count + delay_count;
      Tcks <= Tcks + delay_count;
      count <= count + DELTA;
    else
      if (busy_count='0') then
        state <= setup;
      end if;
    end if;
  end if;
end process;

begin
  process (clk_out)
  begin
    if (clk_out='1') then
      cs_out <= '0';
      din_out <= '0';
      clk_out <= '0';
      Tcks <= Tcks + delay_count;
    else
      if (t <= delay_count) then
        state <= busy;
      end if;
    end if;
  end if;
end process;

begin
  process (reset_dac)
  begin
    if (reset_dac='1') then
      state <= reset_dac;
      cs_out <= '1';
      busy_count <=0;
      clk_out <= '0';
      Tcks <=0;
    end if;
end process;

begin
  process (ready)
  begin
    if (ready='1') then
      state <= ready;
    end if;
end process;

begin
  process (setup)
  begin
    if (setup='1') then
      state <= setup;
    end if;
end process;

begin
  process (read_din)
  begin
    if (read_din='1') then
      state <= read_din;
    end if;
end process;

begin
  process (busy)
  begin
    if (busy='1') then
      state <= busy;
    end if;
end process;

begin
  process (cs_high)
  begin
    if (cs_high='1') then
      state <= cs_high;
    end if;
end process;

begin
  process (wait_dac)
  begin
    if (wait_dac='1') then
      state <= wait_dac;
    end if;
end process;

begin
  process (clk_DAC)
  begin
    if (clk_DAC='1') then
      state <= clk_DAC;
    end if;
end process;

begin
  process (DIN)
  begin
    if (DIN='1') then
      state <= DIN;
    end if;
end process;

begin
  process (SUPPLY_MIN)
  begin
    if (SUPPLY_MIN='1') then
      state <= SUPPLY_MIN;
    end if;
end process;

begin
  process (SUPPLY_MAX)
  begin
    if (SUPPLY_MAX='1') then
      state <= SUPPLY_MAX;
    end if;
end process;

begin
  process (CS)
  begin
    if (CS='1') then
      state <= CS;
    end if;
end process;

begin
  process (DELTA)
  begin
    if (DELTA='1') then
      state <= DELTA;
    end if;
end process;

begin
  process (CLKIN)
  begin
    if (CLKIN='1') then
      state <= CLKIN;
    end if;
end process;

begin
  process (RESET)
  begin
    if (RESET='1') then
      state <= RESET;
    end if;
end process;

begin
  process (PBl)
  begin
    if (PBl='1') then
      state <= PBl;
    end if;
end process;

begin
  process (UPDOWN)
  begin
    if (UPDOWN='1') then
      state <= UPDOWN;
    end if;
end process;

begin
  process (SHDN)
  begin
    if (SHDN='1') then
      state <= SHDN;
    end if;
end process;
end process;

end architecture;
elsif (CLK_IN='1' and CLK_IN'event) then -- optimized for 5MHz clock
  case state is
    when reset_dac =>
      if (Tcks = 2) then
        Tcks <=0;
        state <= setup;
        count <= count_ref;
        busy_count <= busy_count +1; -- (start counting Tcslo)
        cs_out <= '0';
        delay_count <= 0;
      else
        state <= reset_dac;
        Tcks <= Tcks +1;
      end if;

    when ready =>
      if (PB1 = '1') then
        if (UP_DOWN = '0') then -- increase supply voltage => decrease counter
          if (count > DELTA) then
            count <= count-DELTA;
            max<='0';
          else
            count <= "00000000";
            max <= 'l';
          end if;
        else -- decrease supply voltage => increase counter
          if ((count_ref-count) > DELTA) then
            count <= count+DELTA;
            min<='0';
          else
            count <= count_ref;
            min <= 'l';
          end if;
        end if;
        busy_count <= busy_count +1; -- (start counting Tcslo)
        cs_out <= '0';
        state <= setup;
        delay_count <= 0;
      else
        cs_out <= '1';
        state <= ready;
      end if;

    when setup =>
      if (delay_count = 3) then
        state <= read_din;
        delay_count <= 0;
        clk_cyc <= 0;
      else
        delay_count <= delay_count +1;
      end if;
      busy_count <= busy_count +1;

    when read_din =>
      if (clk_cyc = 8) then -- done reading din
        state <= busy;
        clk_cyc <= 0;
        clk_out <= '0';
      else
        state <= read_din;

end case;
if (delay_count = 0) then
  din_out <= count(7-clk_cyc); -- sending D7 to D0
  delay_count <= delay_count + 1;
elsif (delay_count = 2) then
  clk_out <= '1';
  delay_count <= delay_count + 1;
elsif (delay_count = 4) then
  clk_out <= '0';
  delay_count <= delay_count + 1;
elsif (delay_count = 6) then
  delay_count <= 0;
  clk_cyc <= clk_cyc + 1;
else
  delay_count <= delay_count+1;
end if;
end if;

when busy =>
  if (busy_count =50) then -- end Tcslo and din shifted to dac register
    state <= cs_high;
    busy_count <=0;
    cs_out <= '1';
  else
    busy_count <= busy_count +1;
    state <= busy;
  end if;

when cs_high => -- Tcshi time
  if (delay_count = 3) then
    delay_count <=0;
    state <= wait_dac;
  else
    delay_count <= delay_count +1;
  end if;

when wait_dac =>
  if (PB1='0') then -- wait till the previous trigger is dead
    state <=ready;
  else
    state <= wait_dac;
  end if;
end case;
end if;
end process;

DIN <= din_out;
CLK_DAC <= clk_out;
CS <= cs_out;
SHDN <= '1';
SUPPLY_MIN <= min;
SUPPLY_MAX <= max;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity delay_adc2 is
  Port ( CLK : in std_logic; -- 5MHz clock
          RESET: in std_logic;
          TRIG_DELAY : in std_logic; -- only high for 3 clk cycles
          TRIG_I0: IN STD_LOGIC;
          TRIG_ADC : out std_logic
          );
end delay_adc2;

architecture Behavioral of delay_adc2 is

  type delay_machine is (ready,
                         trigger_high
                         );

  signal state: delay_machine:= ready;
  attribute fam_encoding: string;
  attribute fam_encoding of state: signal is "compact"; -- This tells XST to use binary encoding

  signal delay_count : integer range 0 to 3750005 :=0;
  signal trigger : std_logic :='0';

begin

  process (CLK, RESET, TRIG_DELAY, TRIG_I0)
  begin

    if (RESET = '0') then
      delay_count <=0;
      trigger <= '0';
      state<= ready;
    elsif (CLK = '1' and CLK'event) then
      case state is
        when ready =>
          if (TRIG_I0 = '1') then
            state <= trigger_high;
            delay_count <=3750000;
          elsif (TRIG_DELAY = '1') then
            state <= trigger_high;
            delay_count <=0;
          else
            state <= ready;
          end if;
        when trigger_high =>
          if (delay_count = 3750000) then
            trigger <= '1';
            delay_count <= delay_count +1;
          elsif (delay_count = 3750003) then
            state <= ready;
            trigger <= '0';
            delay_count <=0;
          else
            trigger <= '0';
            delay_count <= delay_count +1;
          end if;
      end case;

  end process;

end Behavioral;
end if;
end process;

TRIG_ADC <= trigger;

end Behavioral;
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_unsigned.all;
use IEEE.std_logic_arith.all;

entity adc_conv2 is
  Port (RESET : in STD_LOGIC;
        SAMPLE : IN STD_LOGIC;
        CLK_IN : IN STD_LOGIC;
        CLK_OUT : OUT STD_LOGIC;
        CONV_OUT : OUT STD_LOGIC;
        DIN : IN STD_LOGIC;
        DOUT : OUT STD_LOGIC_VECTOR (11 DOWNTO 0);
        TRIG_COUNTER : OUT STD_LOGIC
      );
end adc_conv2;

architecture Behavioral of adc_conv2 is

  type conv_machine is (ready,
    conv_high,
    conv_busy,
    read_data,
    busy,
    trigger_counter
  );

  signal state : conv_machine := ready;
  attribute fsm_encoding: string;
  attribute fsm_encoding of state: signal is "compact"; -- This tells XST to use binary encoding

  signal hold : std_logic := '0';
  signal count_cyc : integer range 0 to 21 :=0 ; -- finish the code here
  type Dref_Table is array (integer range 0 to 11) of std_logic;
  signal Dref : Dref_Table:= (
    0 => '0',
    1 => '0',
    2 => '0',
    3 => '0',
    4 => '0',
    5 => '0',
    6 => '0',
    7 => '0',
    8 => '0',
    9 => '0',
    10 => '0',
    11 => '0'
  );

  signal Dref_Ptr : integer range 0 to 12:= 0;
  signal dout_bin : std_logic_vector (11 downto 0):="000000000000";
  signal trigger : std_logic :='0';
  signal read_ready : std_logic :='0';
  signal delay_count : integer range 0 to 6:=0;

begin

  process(RESET, SAMPLE, CLK_IN, DIN)
  begin

    if (RESET='0') then

      hold <= '0';
      state <= ready;

  end process;

end Behavioral;
countcyc <= 0;
delay_count <= 0;
Dref_Ptr <= 0;
dout_bin <= "000000000000";
trigger <= '0';
read_ready <= '0';
elif (CLK_IN = '1' and CLK_IN'event) then
case state is
when ready =>
  if (SAMPLE = '1') then
    state <= conv_high;
countcyc <= 0;
  else
    state <= ready;
  end if;
when conv_high =>
countcyc <= countcyc +1;  -- 1st CLK cycle
hold <= '1';
state <= conv_busy;
when conv_busy =>
  if (countcyc =3 ) then  -- 3rd CLK cycle
    state<=read_data;
    hold <='0';
countcyc <= countcyc +1;
    Dref_Ptr <= 0;
  else
    countcyc <= count_cyc+1;  -- 1st CLK cycle
  end if;
when read_data =>  -- 4th cycle start reading data
  if (count_cyc =15) then
    state<= read_data;
    count_cyc<= count_cyc+1;
    Dref_Ptr <=0;
    dout_bin <= Dref(0)&Dref(1)&Dref(2)&Dref(3)&Dref(4)&Dref(5)&Dref(6)&Dref(7)&Dref
    (8)&Dref(9)&Dref(10)&Dref(11);
  elsif countcyc = 20) then
    state <= busy;
    count_cyc<=0;
  else
    state <= read_data;
    Dref(Dref_Ptr) <= DIN;
    Dref_Ptr <= Dref_Ptr +1;
    count_cyc <= count_cyc +1;
  end if;
when busy =>
  if (SAMPLE = '1') then
    state <= busy;
  else
    if (read_ready = '0') then
      read_ready <='1';
      state <= conv_high;
    else
      delay_count <=0;
    end if;
  end if;
  else
    state<= trigger_counter;
    read_ready <= '0';
    --waiting for conversion to finish up
  end if;
when trigger_counter =>
  if (delay_count = 5) then
    state <= ready;
trigger <= '0';
delay_count <= 0;
else
   delay_count <= delay_count +1;
   trigger <= '1';
   state<= trigger_counter;
end if;
end case;
end if;
end process;

CLK_OUT <= CLK_IN;
CONV_OUT <= hold;
DOUT <= dout_bin;
TRIG_COUNTER <= trigger;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity counter is
    Port (CLK_IN : in STD_LOGIC;
          RESET : in STD_LOGIC;
          reset_count : in STD_LOGIC;
          do : out STD_LOGIC_VECTOR (3 downto 0);
          clk_out : out STD_LOGIC);
end counter;

architecture Behavioral of counter is
    signal count : std_logic_vector (3 downto 0);
    signal clk_next : std_logic := '0';
begin
    process (CLK_IN, RESET, reset_count) begin
        if (RESET = '0') then
            count <= "0000";
            clk_next <= '0';
        elsif (reset_count = '1') then
            count <= "0000";
        elsif (CLK_IN = '1' and CLK_IN'event) then
            if (count = "1001") then
                count <= "0000";
                clk_next <= '1';
            else
                count <= count +1;
                clk_next <= '0';
            end if;
        end if;
    end process;
    do <= count;
    clk_out <= clk_next;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity counter_top is
  Port (CLKIN : in STD_LOGIC;
        RESET : in STD_LOGIC;
        TRIGIN : in STD_LOGIC;
        DIN : in STD_LOGIC_VECTOR (11 downto 0);
        DO : out STD_LOGIC_VECTOR (3 downto 0);
        CLK_OUT : out STD_LOGIC)
); end counter_top;

architecture Behavioral of counter_top is

  type countermachine is (counting,
                          busy,
                          ready);

  signal state: countermachine := ready;
  attribute fem_encoding: string;
  attribute fem_encoding of state: signal is "compact";

  signal count: std_logic-vector (3 downto 0):="0000";
  signal clk_next: std_logic := '0';
  signal clk_count: stdlogicvector (13 downto 0):="00000000000000";
  signal delay_count: integer range 0 to 6:=0;

begin
  process (CLKIN, RESET, DIN, TRIGIN)
  begin
    if (RESET = '0') then
      state <= ready;
      clk_count <="00000000000000";
      clk_next <='0';
      delay_count <=0;
      count <="0000";
    elsif (CLK_IN = '1' and CLK_IN'event) then
      case state is
      when ready =>
        if (TRIG_IN = '1') then
          state <= counting;
          count <="0000";
        else
          state <=ready;
        end if;
      when counting =>
        if (clk_count = ("00"&DIN)+("00"&DIN)) then
          state <= busy;
          delay_count <=0;
          clk_next <='0';
        else
          state <= counting;
          clk_count <= clk_count +1;
          if (count = "1001") then
            count <= "0000";
          end if;
        end if;
    end case;
  end if;
end process;
end Behavioral;
clk_next <= '1';
else
  count <= count +1;
  clk_next <= '0';
end if;
end if;
when busy =>
  if (delay_count = 6) then
    state<= ready;
    delay_count <= 0;
  else
    delay_count <= delay_count +1;
  end if;
end case;
end if;
end process;
D0 <= count;
CLK_OUT <= clk_next;
end Behavioral;
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity display_drive3 is
  Port ( DO : in STD_LOGIC_VECTOR (3 downto 0);
        D1 : in STD_LOGIC_VECTOR (3 downto 0);
        D2 : in STD_LOGIC_VECTOR (3 downto 0);
        D3 : in STD_LOGIC_VECTOR (3 downto 0);
        CLK_IN : in STD_LOGIC;
        LED: out STD_LOGIC_VECTOR (6 downto 0);
        LED_dot: out STD_LOGIC;
        AN1: out STD_LOGIC;
        AN2: out STD_LOGIC;
        AN3: out STD_LOGIC;
        AN4: out STD_LOGIC);
end display_drive3;

architecture Behavioral of display_drive3 is
  signal digit_count : std_logic_vector (1 downto 0):="00";
  signal count_out : std_logic_vector (3 downto 0):="0000";
  signal delay_count : integer range 0 to 51 :=0;
begin
  process (CLK_IN, D0, D1, D2, D3)
  begin
    if (CLK_IN = '1' and CLK_IN'event) then
      if (delay_count = 50) then
        delay_count := 0;
        if (digit_count = "00") then
          digit_count := digit_count +1;
          count_out <= D0;
          AN4 <= '0';
          AN3 <= '1';
          AN2 <= '1';
          AN1 <= '1';
          LED_dot <= '1';
        elsif (digit_count = "01") then
          digit_count := digit_count +1;
          count_out <= D1;
          AN4 <= '1';
          AN3 <= '0';
          AN2 <= '1';
          AN1 <= '1';
          LED_dot <= '0';
        elsif (digit_count = "10") then
          digit_count := digit_count +1;
          count_out <= D2;
          AN4 <= '1';
          AN3 <= '1';
          AN2 <= '0';
          AN1 <= '1';
          LED_dot <= '1';
        else
          digit_count := "00";
          count_out <= D3;
          AN4 <= '1';
          AN3 <= '1';
          AN2 <= '1';
          AN1 <= '0';
      end if;
    end if;
  end process;
end Behavioral;
LED_dot <= '1';
end if;
else
  delay_count <= delay_count +1;
end if;
end if;
end process;
with count_out Select
  LED<= "1001111" when "0001", --1
       "0010010" when "0010", --2
       "0000110" when "0011", --3
       "1001100" when "0100", --4
       "0100100" when "0101", --5
       "0100000" when "0110", --6
       "0001111" when "0111", --7
       "0000000" when "1000", --8
       "0001100" when "1001", --9
       "0000001" when "0000", --0
       "1111111" when others; -- A, b, etc.
end Behavioral;
Appendix C
Closed Loop System: Schematics, Board Layout and Test setup

The schematic for the Optimal Power Transfer Regulator (OPTR) including the DC/DC converter and the LED display is shown in the next page. Refer to the demo board manual of LTC 3789 (http://cds.linear.com/docs/Demo%20Board%20Schematic/1523asch.pdf) and Altera MAX V CPLD board (http://www.altera.com/literature/manual/rm_maxv_cpld_dev_board.pdf) for more detailed schematics on the two demo boards. The board schematic/floor plan is shown at the end of this Appendix.

The output of the OPTR board (V_ROYER) is connected to the supply node of the Royer circuit (Vdc) shown in Figure C-1. The receiver board (hysteretic tuning controller with PWM) is coupled as seen Figure C-1.

![Schematic Diagram](image)

Figure C-1: Royer circuit coupled to the Receiver with hysteretic tuning controller circuit