Spring 2018

# IDPAL - Input Decoupled Partially Adiabatic Logic: Implementation and Examination

Kevin A. Johnson
*Old Dominion University*

# IDPAL - INPUT DECOUPLED PARTIALLY ADIABATIC LOGIC: IMPLEMENTATION AND EXAMINATION

by

Kevin A. Johnson

B.S. May 2016, Old Dominion University

A Thesis Submitted to the Faculty of Old Dominion University in Partial Fulfillment of the Requirement for the Degree of

MASTER OF SCIENCE

ELECTRICAL AND COMPUTER ENGINEERING

OLD DOMINION UNIVERSITY

May 2018

Approved by:

_____

Lee A. Belfore, II (Director)

_____

Chunsheng Xin (Member)

_____

Chung Hao Chen (Member)

# ABSTRACT

## IDPAL - INPUT DECOUPLED PARTIALLY ADIABATIC LOGIC: IMPLEMENTATION AND EXAMINATION

Kevin A. Johnson

Old Dominion University, 2018

Director: Dr. Lee A. Belfore, II

This thesis presents the experimental results of a four-phase IDPAL eight-input exclusive-OR gate. The following problems with IDPAL are addressed: multistage circuits malfunctioning, simulation convergence anomalies, and inferring input information through the power clock current. EPAD MOSFETs, which provide a low threshold voltage, are shown to be unsuccessful in correcting the malfunctioning behavior of multilayer circuits. A solution to multilayer IDPAL circuits malfunctioning, called IDPAL with discharge, is shown. The differences between simulation waveforms produced by LTspice and the experimental circuits recorded by a Tektronix's Oscilloscope are investigated. IDPAL is implemented and analyzed using ALD MOSFETs for the following adiabatic families: 2N-2P, IDPAL, and IDPAL with discharge.

# DEDICATION

*This thesis is dedicated to Nancy and Kirk Johnson, my parents.*

# ACKNOWLEDGEMENTS

This research preformed in the thesis was made possible by Dr. Lee Belfore and Old Dominion University. I thank Dr. Lee Belfore for his guidance and assistance in the implementation of the work described in this thesis. I want to thank Dr. Chunsheng Xin and Dr. Chung Hao Chen for their comments and suggestions. I am grateful for the equipment and opportunities that Old Dominion University has provided for me. The Electrical and Computer Engineering Department has helped me develop the skills required to perform this research. My family has been very supportive and helpful throughout my entire life. If it wasn't for the help and encouragement of everyone, this research would not have been possible. Thank you.

# Table of Contents

# LIST OF FIGURES

# 1. INTRODUCTION

Low energy computing is a critical concern for extending the life of battery powered applications. For example, a cardiac pacemaker is a battery powered device inserted into a patient's body which lasts roughly 8 years [1]. Decreasing the energy consumption of the microprocessor would extend the life of the device and reduce the number of replacements throughout the patient's life. Traditional energy reduction techniques are slowing due to accelerating manufacturing complexities with the traditional digital circuit architectures [2]. An attractive method to decrease the amount of energy lost per computational instruction is known as adiabatic computing. Theoretically, adiabatic circuits consume zero energy per computation [3]. The MOS (Metal-Oxide-Silicon) transistors used to create adiabatic gates have a nonzero resistance unlike an ideal switch [4]. Non-ideal adiabatic circuits are commonly referred as pseudo-adiabatic circuits [5]. Efficient charge recovery logic (ECRL) is an adiabatic logic family that is based on CMOS (Complementary MOS) and has been shown to require less energy per cycle than traditional CMOS [6]. Further energy recovery has been accomplished by decreasing the threshold voltage on the MOS transistors [6]. The energy recovery capability of low threshold voltage MOS demonstrates the ability to further remove charge.

Initial studies of Input-Decoupled Partially Adiabatic Logic (IDPAL) introduced by Cutitaru show lower power consumption and suggest better resistance to power analysis attacks compared to ECRL [7]. Previous to the work discussed in this thesis, IDPAL has only been analyzed through simulation. The simulation results suggest that IDPAL is at least as efficient as other adiabatic logic families. Applying low threshold voltage MOSFETs to IDPAL may further reduce energy consumption as seen with ECRL.

Subsequent simulation studies of more complex multilayer IDPAL circuits in more realistic scenarios by Belfore, showed that Cutitaru's formulation begins to malfunction at low frequencies (~10kHz). The malfunction is a result of trapped charge on the gates of the MOS transistors. This trapped charge is not fully recovered, resulting in one or more transistors not switching off and circuit outputs reporting incorrect logic values. A subsequent formulation by Belfore includes strategically placed transistors to nonadiabatically discharge nodes holding trapped charge.

This thesis will take the design of IDPAL gates from simulation and produce the first circuit implementation of IDPAL to verify the simulation results match laboratory measurements. The test circuit used to evaluate IDPAL circuits is an eight-input exclusive-OR logic function organized into three layers. A four-phase trapezoidal power clock, required to operate adiabatic circuits, was generated by the combination of a field programmable gate array (FPGA) and a digital to analog converter (DAC). The FPGA computed the digital samples of the power clock while the DAC converted the digital value into the analog domain. Linear Technologies SPICE (LTspice) and transistor models provided by Advanced Linear Devices (ALD) were used to simulate the adiabatic circuits.

## 1.1 TRADITIONAL CMOS CIRCUITS

The metal-oxide-semiconductor field-effect transistor (MOSFET) is used in almost every integrated circuit because of its simplicity and efficiency. The complementary metal-oxide-semiconductor (CMOS) logic circuit technology uses both n-channel and p-channel MOSFETs

to provide a low impedance path to ground or $V_{DD}$, respectively. The energy losses are divided into two categories, static and dynamic [8]. In a static environment, there are subthreshold currents, reversed-biased diode currents, and tunneling currents that transfer energy passively [9], which are effectively leakage currents. Subthreshold currents occur when the gate to source voltage is less than the threshold voltage and increases with decreasing feature size. These currents can only be removed by eliminating the voltage across the source and the drain. The reversed-biased diode current is found in any diffusion region where the potential difference causes the depletion region to expand. The dynamic properties of switching CMOS logic is determined by the manner in which MOSFETs are connected. Traditional CMOS logic suffers from short-circuits, charging, and discharging capacitive loads. During output transition of a logic gate, both the p-channel and n-channel MOSFETs form a conductive path from $V_{DD}$ to GND. This short interval allows for current to flow in a similar way to that of a short-circuit. This is not an inherent problem to CMOS, but rather, the physical connections of any traditional logic gate.

## 1.2 ADIABATIC CIRCUITS

Over 50 years ago, Gordon Moore predicted that the integrated circuit industry would continue doubling the number of transistors on a single integrated circuit every 2 years [10]. This predication still holds true and has led to very dense integrated circuits. Although digital circuits have increased in complexity, the functional operation of a logic gate has remained unchanged. Examination into the loss of power reveals that the architecture of traditional CMOS gates can be modified to improve efficiency. Specifically, the energy stored on a capacitor is not recovered by

traditional gates and is simply dissipated to ground. The energy stored in the form of capacitance is proportional to the voltage squared. This power is dissipated in the form of heat is permanently lost. A CMOS architecture, termed adiabatic logic, was created to address wasted energy of traditional CMOS architectures.

Adiabatic logic is a form of digital CMOS logic that focuses on dissipating less energy when outputs change. A set of rules were created to constrain the intervals when transistors are either open or closed [3]. The first rule is to never turn on a transistor when there is a voltage across the source and the drain. The second rule is to never turn off a transistor when current is flowing through it. Although this set of rules is straight forward, it often requires additional transistors and clocks to achieve [11]. The power supplies of adiabatic circuits must be able to dynamically source and sink currents in a periodic fashion, typically called a power clock. When current is returned to the power supply it must be stored and not dissipated as heat. Implementing adiabatic logic increases complexity and resource requirements with the advantage of being energy efficient. There are other potential benefits to adiabatic logic, such as increasing the difficulty to preform power analysis attacks [12].

## 1.3 PROBLEM STATEMENT

Efficient processors are essential for battery powered digital devices. An Old Dominion University doctoral student, Mihail T. Chutuaru, has proposed a new type of adiabatic logic called Input-Decoupled Partially-Adiabatic Logic (IDPAL). This new architecture has been shown through simulation to be more power efficient than any other adiabatic logic [7]. The

viability of IDPAL has only been shown through simulations, and shortcomings were identified in simulation related to trapped charge in the circuit. Up to this point, there have been no physically functioning implementations of IDPAL. The next logical step to designing fully integrated adiabatic circuits based on the IDPAL family is to evaluate a simplified digital circuit and analyze the strengths and weaknesses of the design. The simulation models should preform comparable to the physical models. The use of IDPAL in multiphase power clock systems is questionable based on the simulation method used in [7]. Inferring information about the input through the power clock current has, again, only been analyzed through simulation [13].

The plan is to create three variations of adiabatic circuits with the ability to vary the threshold voltage of the MOSFETs. Once the individual adiabatic gates are functioning properly, they can be cascaded to form a multilayer system capable of performing more complex functions. The output of this multilayer circuit will be recorded with an oscilloscope and compared against the simulated outputs.

## 1.4 CONTRIBUTIONS

The first actual circuit prototype of IDPAL. A comparison between simulated and experimental results for IDPAL, ECRL, IDPAL with discharge, and their low threshold voltage variations. Depending on the architecture, the experiment and simulation are shown to be nearly identical or inconsistent during certain stages of the power clock. The simulations involving low threshold voltage transistors show increased charge removal, but experimental results show the

opposite to be true in most cases. Current measurements for IDPAL and a first order analysis of results.

## 1.5 OVERVIEW

The work described in this thesis attempts to build a physical design of IDPAL circuits and compare the experimental results to simulation models. This thesis will appear in the following arrangement. Chapter 2 will discuss IDPAL circuits and adiabatic families. Chapter 3 discusses physical models for designing and testing this thesis. Simulation models and results can be found in chapter 4. Differences and similarities between the physical models and simulations will be in chapter 5. Conclusive results of this thesis are found in chapter 7.

# 2. BACKGROUND

The different adiabatic logic families and their functionality are described and compared in this chapter. The energy efficiency and physical implementation are discussed near the end of the chapter.

## 2.1 ADIABATIC FAMILES

There are several adiabatic families, but only a subset of them will be reviewed. The adiabatic families include Efficient Charge Recovery Logic (ECRL), Input-Decoupled Partially-Adiabatic Logic (IDPAL), and a modification to IDPAL, referred to as IDPAL with discharge. The most prominent design of adiabatic circuits relies on an alternating power supply. This power supply is an alternating voltage source between zero and a defined maximum. This source is named a power clock (PC) because it provides both the energy to the circuit and clock. The alternating PC enables a computation to be performed, and later, energy to be recovered in a micro-pipelined fashion. ECRL, also referred to as 2N-2P, is the simplest adiabatic model because it requires the least number of transistors to implement the desired transfer function. One of its key benefits of this family is the lower power consumption compared to traditional CMOS logic. The relatively new adiabatic family known as IDPAL requires additional transistors to, again, preform the same function with a greater efficiency. Although IDPAL has been shown to correctly perform computations, simulations of cascaded IDPAL gates subjected to a sequence of inputs has shown degradation in the signal's propagation. To remedy this failure, Dr. Lee Belfore

has proposed a non-adiabatic solution to remove the charge collected throughout the operation of the circuit. This modification called IDPAL is considered a different family of adiabatic logic with the potential of increasing the operational frequency of IDPAL. This modification will be referred to as IDPAL with discharge.

## 2.2 ADIABATIC VERSUS TRADITIONAL CMOS

Traditional digital circuits are well known for their simplicity, scalability, and fast transition time. Adiabatic logic has shown preform more efficiently and has securities to differential power analysis attacks [12]. Adiabatic logic remains an attractive alternative to CMOS in mobile applications where battery life takes priority over performance.

Adiabatic circuits typically require additional transistors, clocks, and pipelining single logic gates when compared to traditional digital circuits. Adiabatic gates require true and complementary inputs for a signal. They also produce true and complementary outputs in a single logic gate. The main advantage to adiabatic logic is the increased energy efficiency in highly active digital circuits. Adiabatic logic is more energy efficient than traditional CMOS logic at lower frequencies [14]. In this sense, adiabatic circuits minimize energy loss during switching while maximizing the amount of energy recovered. Adiabatic circuits utilize a gradually changing power clock to provide a constant current for charging the capacitances on transistors.

Traditional CMOS circuits use a square-wave clock. Adiabatic circuits use trapezoidal or sinusoidal oscillators for both clock and power. The trapezoidal power clocks have 4 stages, in

which they periodically cycle through. These stages are idle, charge, hold, and recover. In each logic gate, the circuit gradually changes its output during the charge stage. The hold stage allows for the next logic gate time to sample the output during its, out of phase, charge stage. During the recover stage, the energy stored in the form of capacitance, is returned to the source. Figure 1 displays the trapezoidal clock with the four stages marked. This modification to the traditional square-wave clock is due to the charge and recovery stages of adiabatic logic. The ideal operation of adiabatic logic is to recover energy in the circuit by extracting the remaining charge back to the supply after preforming the desired function.

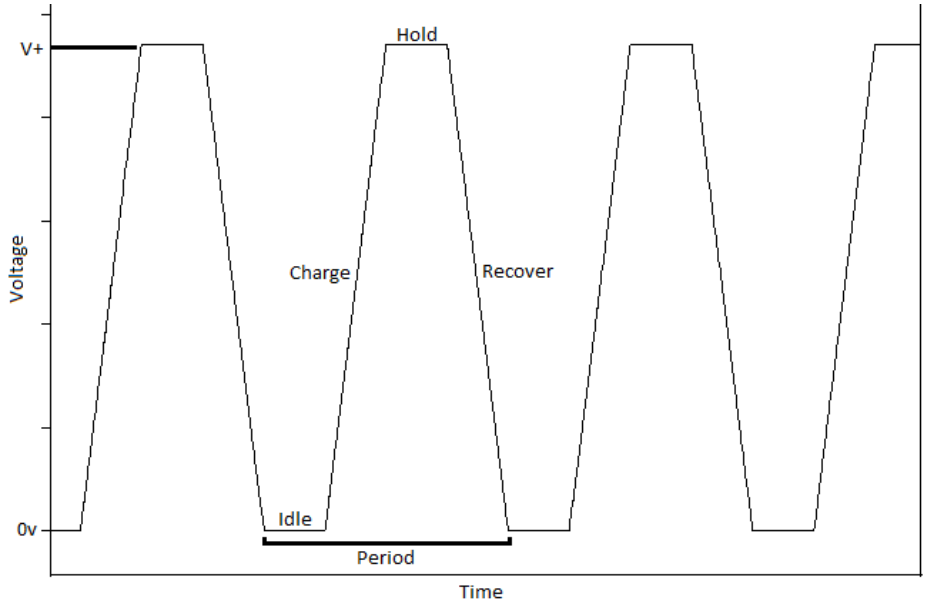


*Figure 1: Trapezoidal Power Clock with marked stages*

## 2.3 ADIABATIC FAMILY 2N-2P

The adiabatic family 2N-2P is known for its inverter/buffer gate comprised of 2 NMOS and 2 PMOS transistors as depicted in Figure 2. The inputs are labeled A and An, while the

outputs are labeled F and Fn. The “n” following the signal names A and F imply an inverted value. In other words, when A is a logical 0, An is assumed to be a logical 1. The 2N-2P gate functions as a differential inverter or buffer depending on which output is non-inverted. As the power clock (PC) rises from zero to V+, the output, F, will either rise because input A is low, or fall because input A is high. On the next stage, called the hold stage, the outputs are held constant. This is when the next layer of logic evaluates the output of gate. The recovery stage follows the hold stage. During the recovery stage, F or Fn that was a logical 1 will fall with the PC. At the hold stage of the PC, both outputs are ideally 0v and no energy is remaining in the circuit.

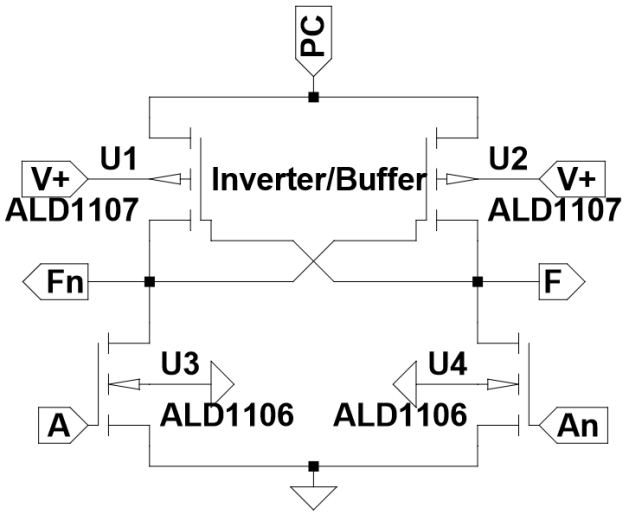


*Figure 2: 2N-2P Inverter/Buffer Gate*

The other adiabatic gates, such as the AND, NAND, OR, and NOR for 2N-2P are derived by adding NMOS transistors in complementary pull-down networks. The 2N-2P NAND and

AND gate can be converted into the NOR and OR gate by flipping the inputs and outputs. This is demonstrated in Figure 3. When the inputs of the NAND gate are both a logical 1, the output Fn and the U2 PMOS is connected to ground by the stacked NMOS transistors on the left. Since the PMOS gate is a logical 0, the MOSFET connects the PC to the output F. During the hold stage of the power clock, the output will reach the close to the maximum amplitude of the power clock.

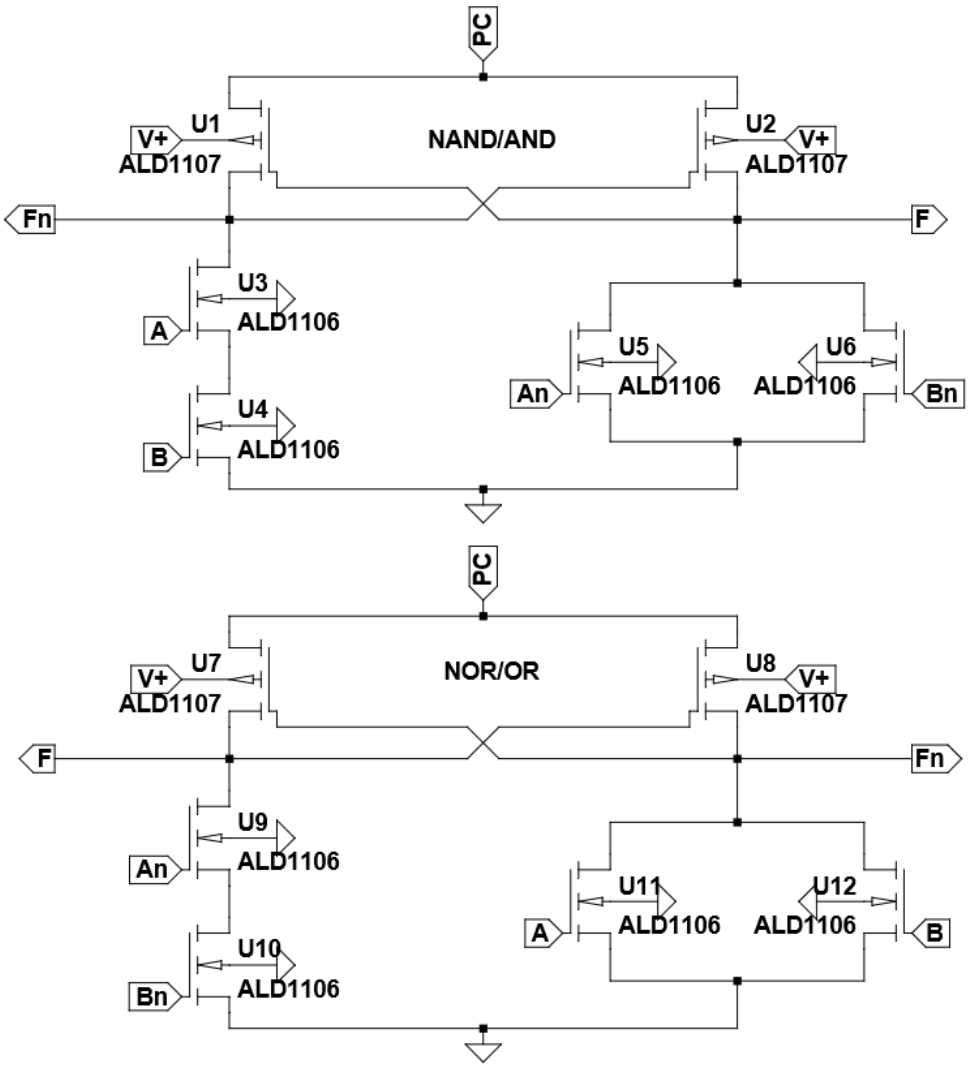


*Figure 3: 2N-2P NAND/AND and NOR/OR Gates*

## 2.4 INPUT-DECOUPLED PARTIALLY-ADIABATIC LOGIC

The new adiabatic family, known as IDPAL, decouples the input to the 2N-2P inputs via additional NMOS transistors. This is shown in Figure 4. The voltage supplied to the NMOS gates responsible for creating a low impedance path to ground originates from the PC instead of the inputs. If the inputs are transitioning states, it has little to no effect on the circuit while the PC is in the hold stage. In other words, input decoupling isolates changes to output while the PC is low.

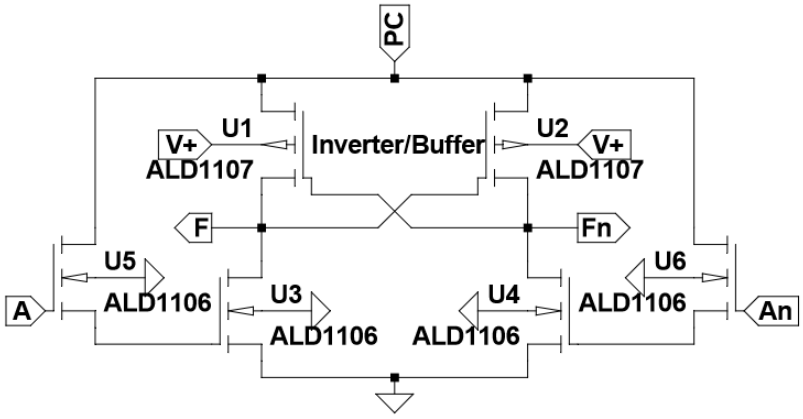


*Figure 4: IDPAL Inverter/Buffer Gate*

All IDPAL logic gates utilize the same decoupling NMOS transistor connections to provide input to a 2N-2P inverter/buffer gate. Other IDPAL gates are formed through a fashion like that of the 2N-2P gates. Stacked transistors are added to implement additional logic functions, these can be seen on either side of Figure 5.

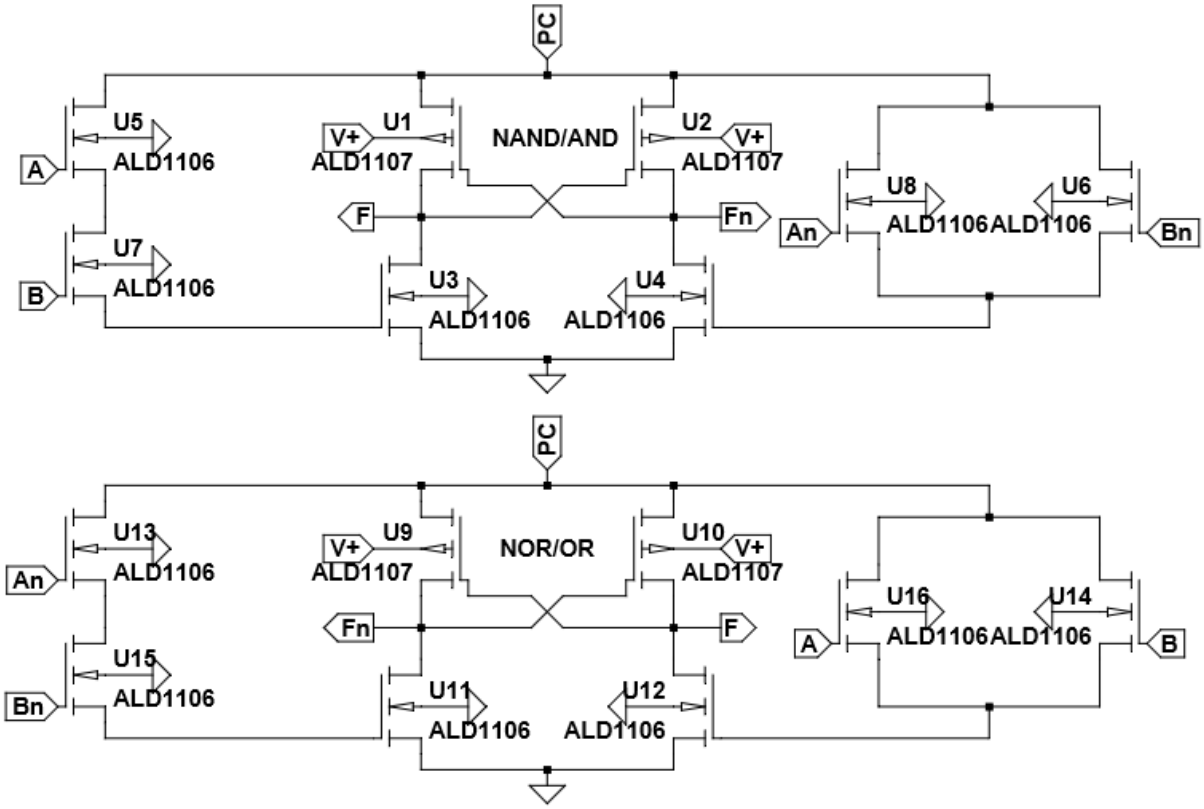


*Figure 5: IDPAL NAND/AND and NOR/OR gates*

## 2.5 IDPAL WITH DISCHARGE MODIFICATION

Through simulation, it was determined that accumulated charges on the NMOS gates would cause cascaded gates to malfunction over time. A modification to IDPAL was made to correct for charge build up, but at the cost of 2 more NMOS transistors to every adiabatic gate. This is called IDPAL with discharge because the additional NMOS transistors remove this charge. The charge was seen through simulation to accumulate on the NMOS transistor gates which drive the output to ground. A solution to remedy this problem in a non-adiabatic manner is

to remove any excess charge while the PC is in the hold stage. During this period of inactivity, the complementary power clock (PCn) actively discharges the gates. Although this solution works, it requires extra constraints to be efficiently implemented. A problem with temporary short circuits exists when the PC and PCn are above the threshold voltage of the NMOS transistors simultaneously. Two potential corrections to this problem are later discussed but are not implemented in this design.

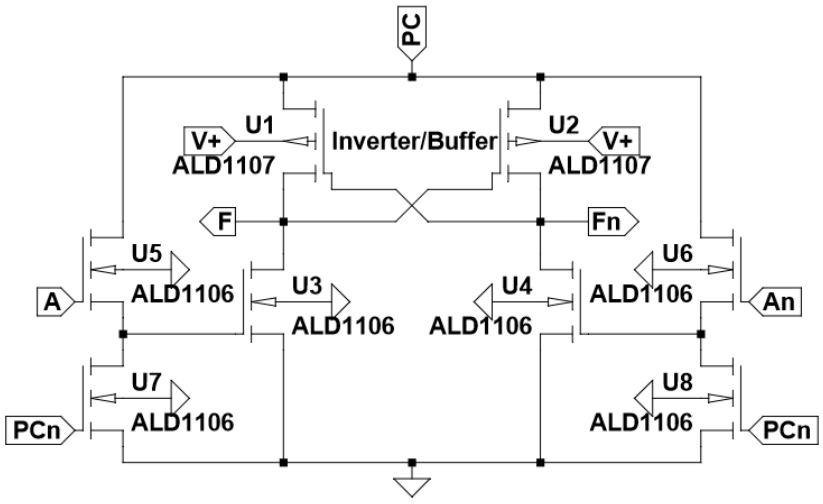


*Figure 6: IDPAL with Discharge Inverter/Buffer Gate*

When IDPAL with discharge is implemented as more complex gates like NAND, AND, OR and NOR, only 2 NMOS transistors are added just like the Inverter gate.

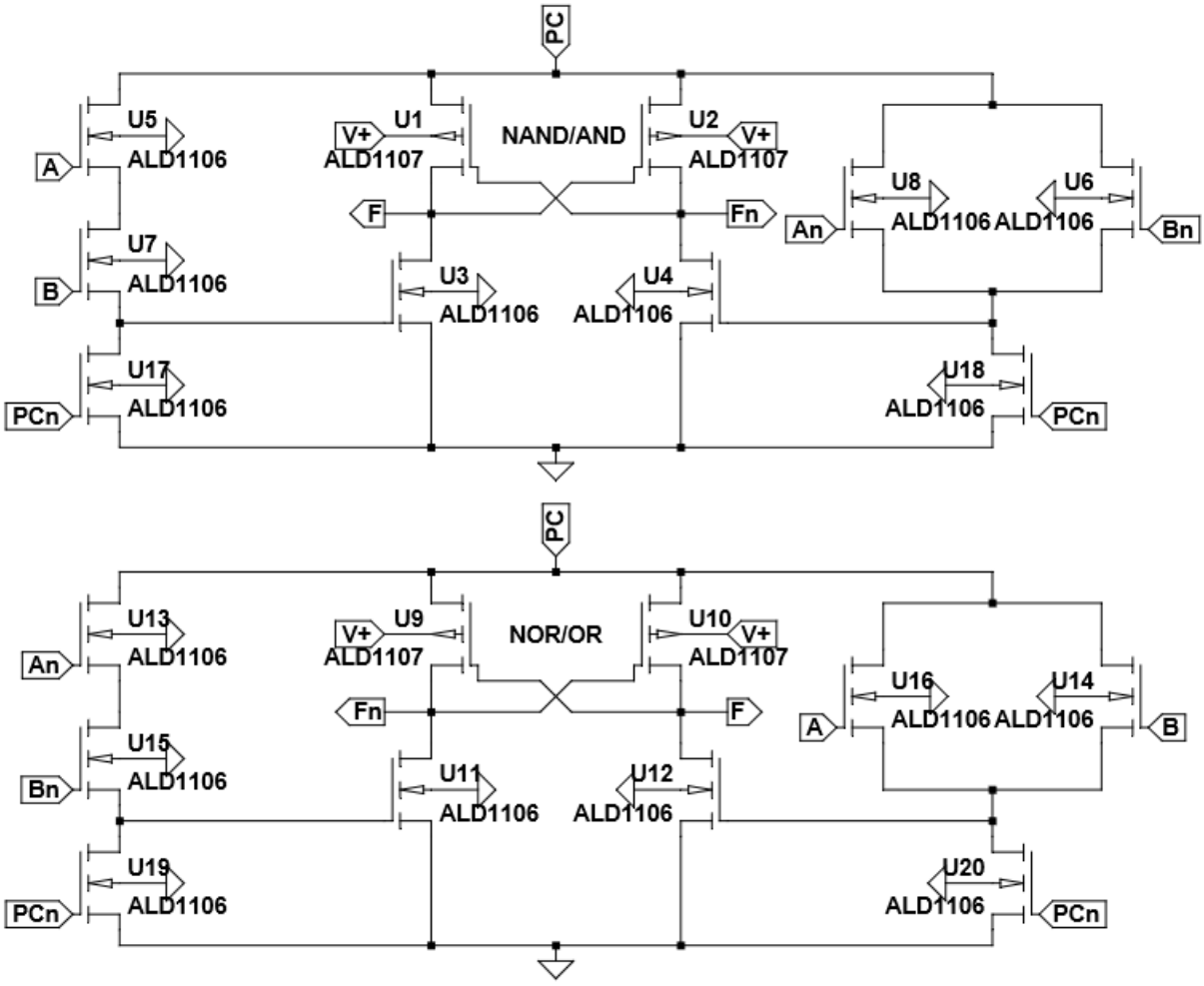


*Figure 7: IDPAL with discharge NAND/AND and NOR/OR gates*

# 3. SIMULATION

This chapter outlines the steps taken to create simulation models of 2N-2P, IDPAL, and IDPAL with discharge. The exclusive OR gate was chosen to be the model because the output has an equal probability of being a 0 or 1 for a uniformly random input. This equal probability is preserved when cascading XOR gates to test several of layers of logic. Advanced Linear Devices (ALD) MOSFET models used in simulation were level 2 transistor models. These models may not provide accurate simulation results in all cases. Occasionally, simulations would not converge on a solution and LTspice parameters would have to be modified. All simulations discussed here use identical parameters.

## 3.1 MOSFETs

Advanced Linear Devices (ALD) MOSFETs had been used by [13] in simulation to analyze power analysis attacks on IDPAL. The ALD 1106 NMOS and 1107 PMOS transistors were used to simulate the three different adiabatic models. The ALD 1108xx family of transistors are monolithic N-Channel MOSFETs that use a technology called Electrically Programmable Analog Device (EPAD) to adjust the threshold voltage of the MOSFETs. The last two digits in the 1108xx family represent the threshold voltage in tenths of a volt. The advantage of these EPAD MOSFETs is that their threshold voltage is significantly less than that of a typical N-Channel MOSFET. The 110804, 110802, and 110800 were selected because of their linearly

decreasing threshold voltage. The EPAD MOSFETs have been show through simulation to extract more of the trapped charge contained in IDPAL. It was shown in [6] that low threshold voltage MOSFETs reduce the energy consumption per clock. These EPAD MOSFETs are implemented in the experimental design as an alternative solution to IDPAL with discharge.

### 3.2 EXCLUSIVE OR DESIGN

The exclusive OR circuit was designed to work with the existing configuration internal to the IC's containing EPAD MOSFETs. This allows for different transistor families to be arranged in a similar configuration. The ALD1106 model in Figure 8 shows the 2N-2P MOSFET connections required to perform a single exclusive OR function. This 2N-2P circuit utilizes stacked NMOS transistors on either side of the figure to form the logical XOR function. These NMOS stacked transistors will either form a conductive channel or a high impedance path to ground. To understand the operation of the circuit through example, suppose two digital inputs A and B were to be XORed. When input A and B are 0, there complimentary inputs An and Bn are both 1. In this specific example where the inputs are 0 volts, the stacked NMOS transistors on the right create a low impedance path from F1 to ground. The voltage corresponding to F1 would ideally be 0 volts in this case. On the opposite side of the figure, the transistors create a high impedance from F1 to ground. The PMOS transistor U2 located at the top right has its gate connected to Fn1 and therefore has a low impedance path from PC1 to F1. The operation of the 2N-2P XOR gate relies on all inputs having the condition where one cascade of NMOS transistors have a low impedance while the other cascade has a high impedance. This cascade impedance condition is true for 2N-2P, IDPAL, and IDPAL with discharge.

### 3.2 LTSPICE ADIABATIC MODELs

Linear Technologies provides a free Simulation Program with Integrated Circuit Emphasis (SPICE) software called LTspice. This software was used to simulate different adiabatic families at different frequencies with both regular and EPAD MOSFETs. Figure 8 through Figure 19 are single adiabatic XOR gates running with digital inputs and a power clock frequency of 100kHz. The simulation outputs are plotted with the inputs to the adiabatic circuits located at the top and the outputs on bottom. The PC and digital inputs transition from 0 to 3.3 volts.

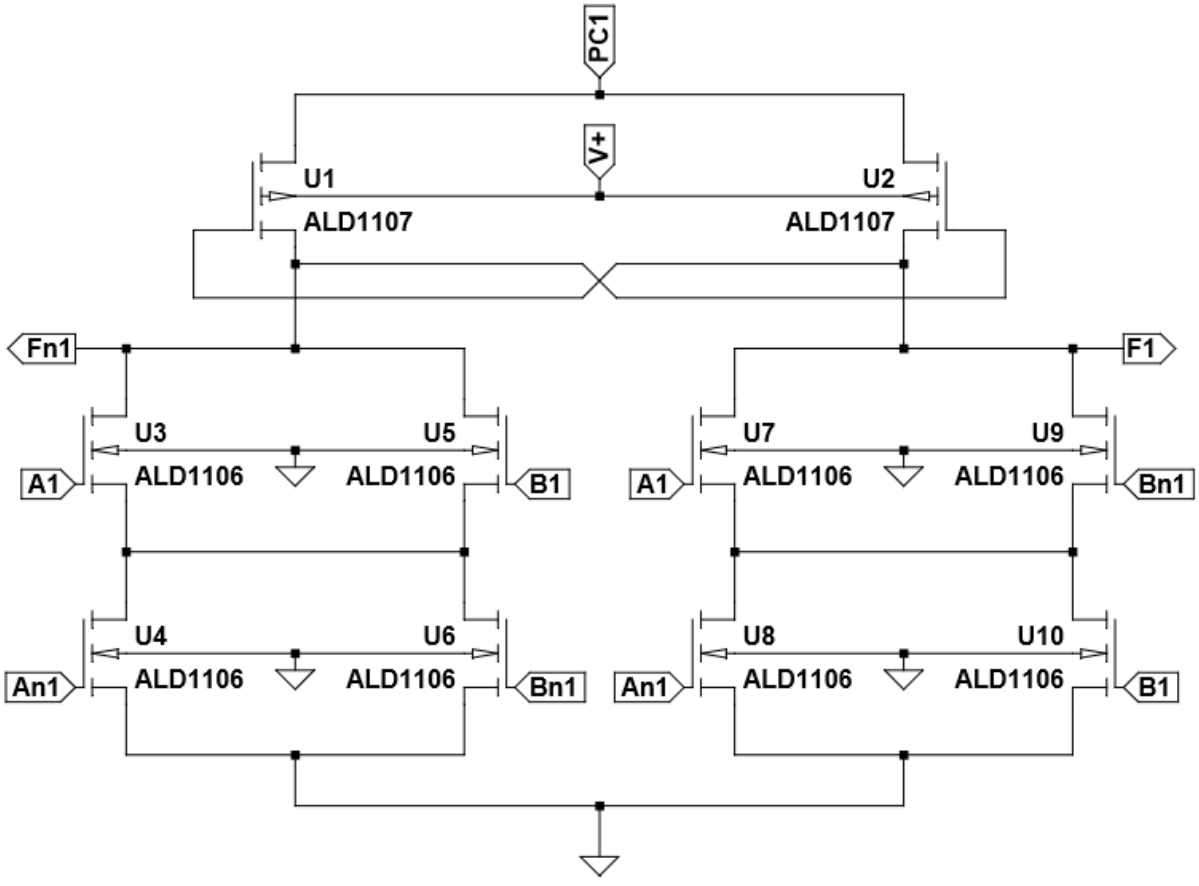


*Figure 8: ALD1106 2N-2P XOR/XNOR gate*

Figure 9 is a plot of the 2N-2P gate which is constructed from the ALD1106 MOSFETs. This simulation shows that the circuit is producing the correct output waveform for an XOR gate. When the power clock is decreasing from 3.3 to 0 volts the output follows it until roughly 1.0v. This is due to the PMOS transistors becoming high impedance because the voltage applied from gate to source is less than the threshold voltage. When the inputs transition the remaining trapped charge is non-adiabatically removed.

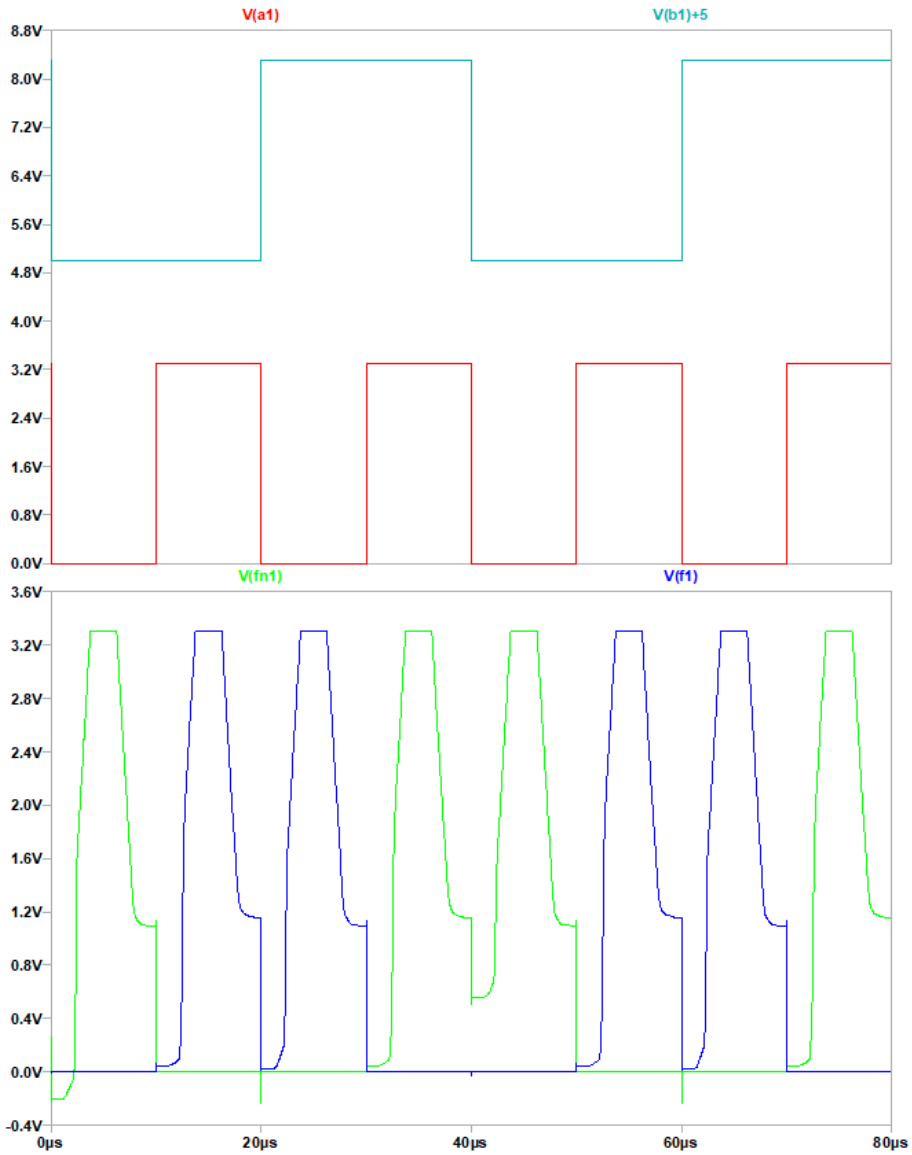


*Figure 9: ALD1106 2N-2P 100kHz Waveform (simulated)*

The IDPAL XOR gate has two additional NMOS transistors connected to either NMOS cascade. In the previous 2N-2P example, the output was directly connected to the cascade and would create a low impedance path from one of the outputs to ground. In the IDPAL XOR gate, the stacked NMOS transistors will form a path, complementary to each other, from PC1 to the gate of either NMOS U3 or U4. Charging the NMOS gate of either U3 or U4 will connect Fn1 or

F1 respectively to ground. Therefore, Fn1 will be 0 volts when both A1 and B1 are 0 volts. IDPAL has a complementary output when compared to 2N-2P.

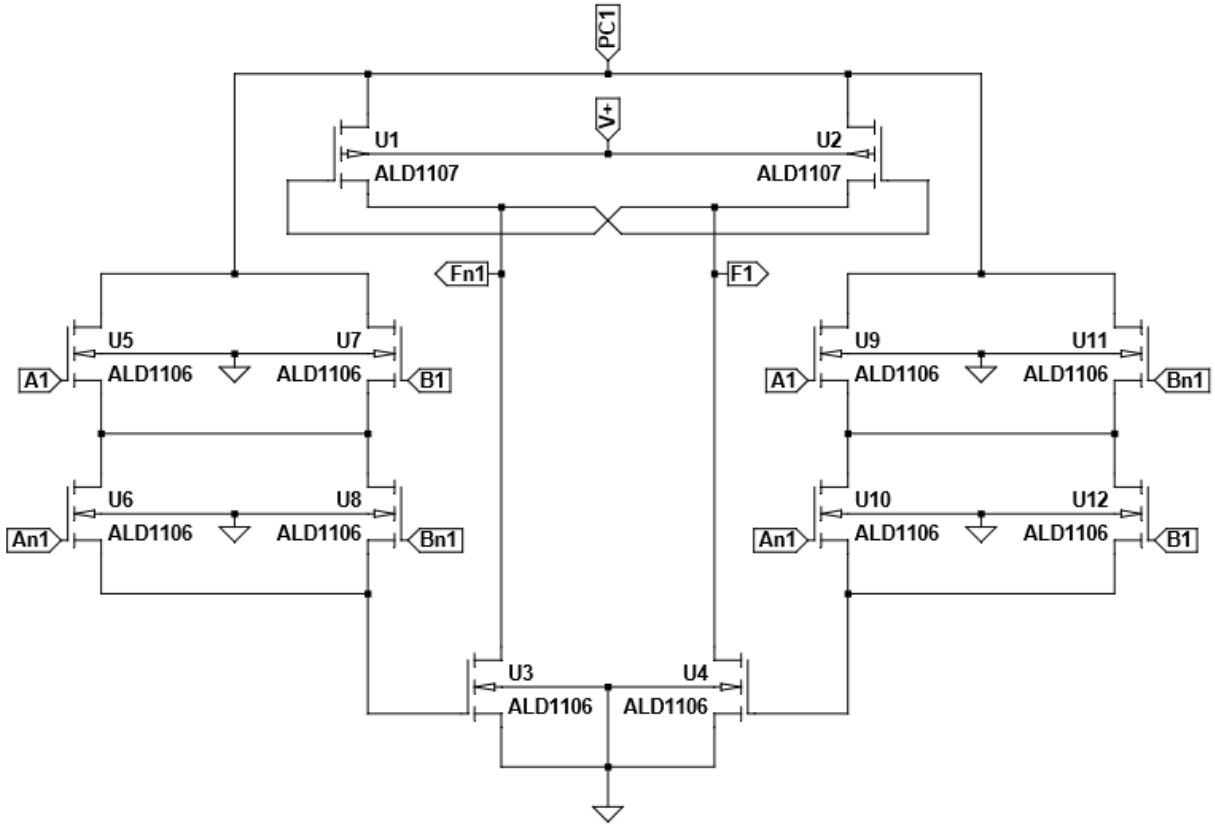


*Figure 10:ALD1106 IDPAL XOR/XNOR gate*

Figure 11 is a plot of the IDPAL gate which is constructed from the ALD1106 MOSFETs. This simulation shows that the circuit is producing the correct output waveform for an XOR gate. Again, when the power clock is decreasing from 3.3 to 0 volts the output follows it until roughly 1.0v because of the PMOS transistor threshold voltage. The output waveform transitions to ground when the power clock is greater than the threshold voltage of the ALD1106 NMOS transistors. This is caused by the power clock sourcing the current to charge the NMOS gate which grounds the output.

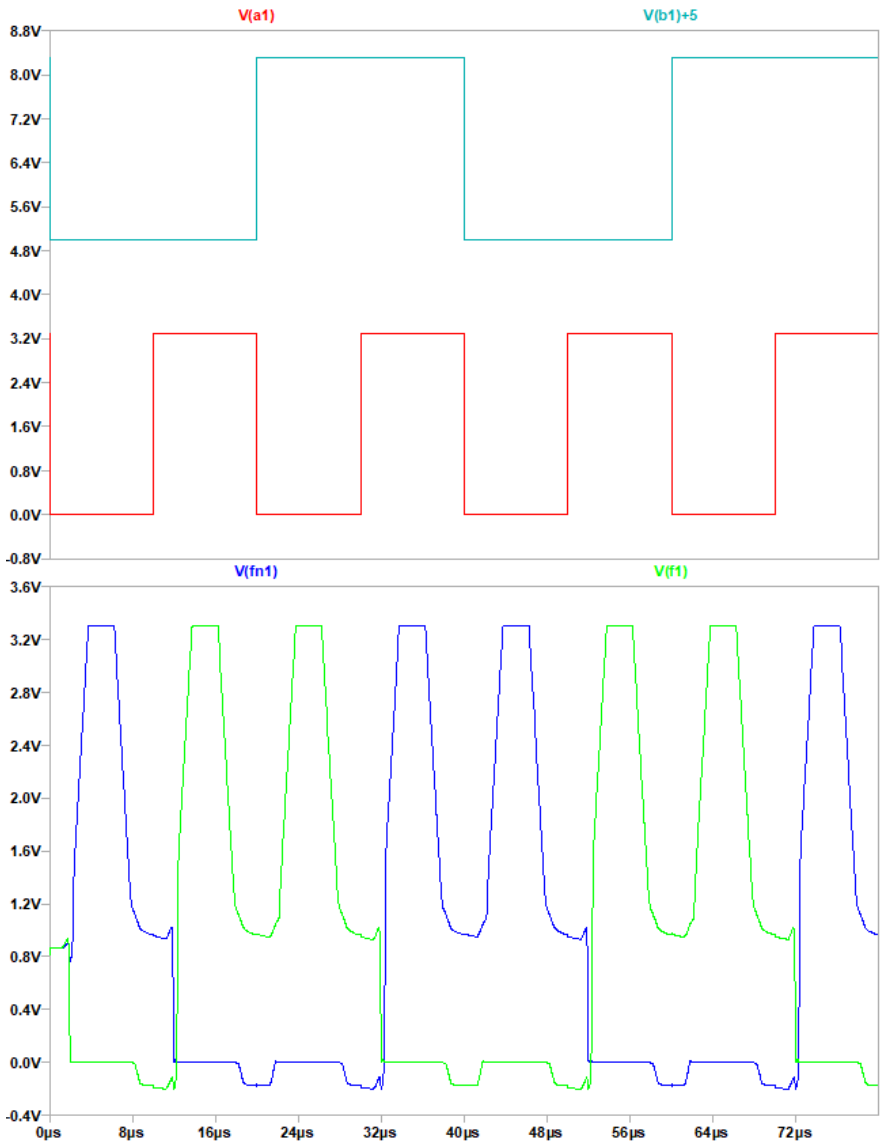


*Figure 11: ALD1106 IDPAL 100kHz Waveform (simulated)*

IDPAL with discharge shown in Figure 12 has two additional NMOS transistors when compared with the original IDPAL formulation. The discharge transistors are controlled by PCn1 to draw off the remaining charge while the circuit is inactive. In a four-phase power clock system, the discharge transistors may conduct current though the stack of NMOS transistors due to the power clocks being above the threshold voltage of the transistors at the same time. One

solution is to design a six-phase power clock system to remove the problem, but this would require additional hardware. Another solution is to lower the power clock voltage so that both power clocks are never above the threshold voltage of the transistors simultaneously. This maximum power clock voltage would be twice the threshold voltage of the MOSFETs.

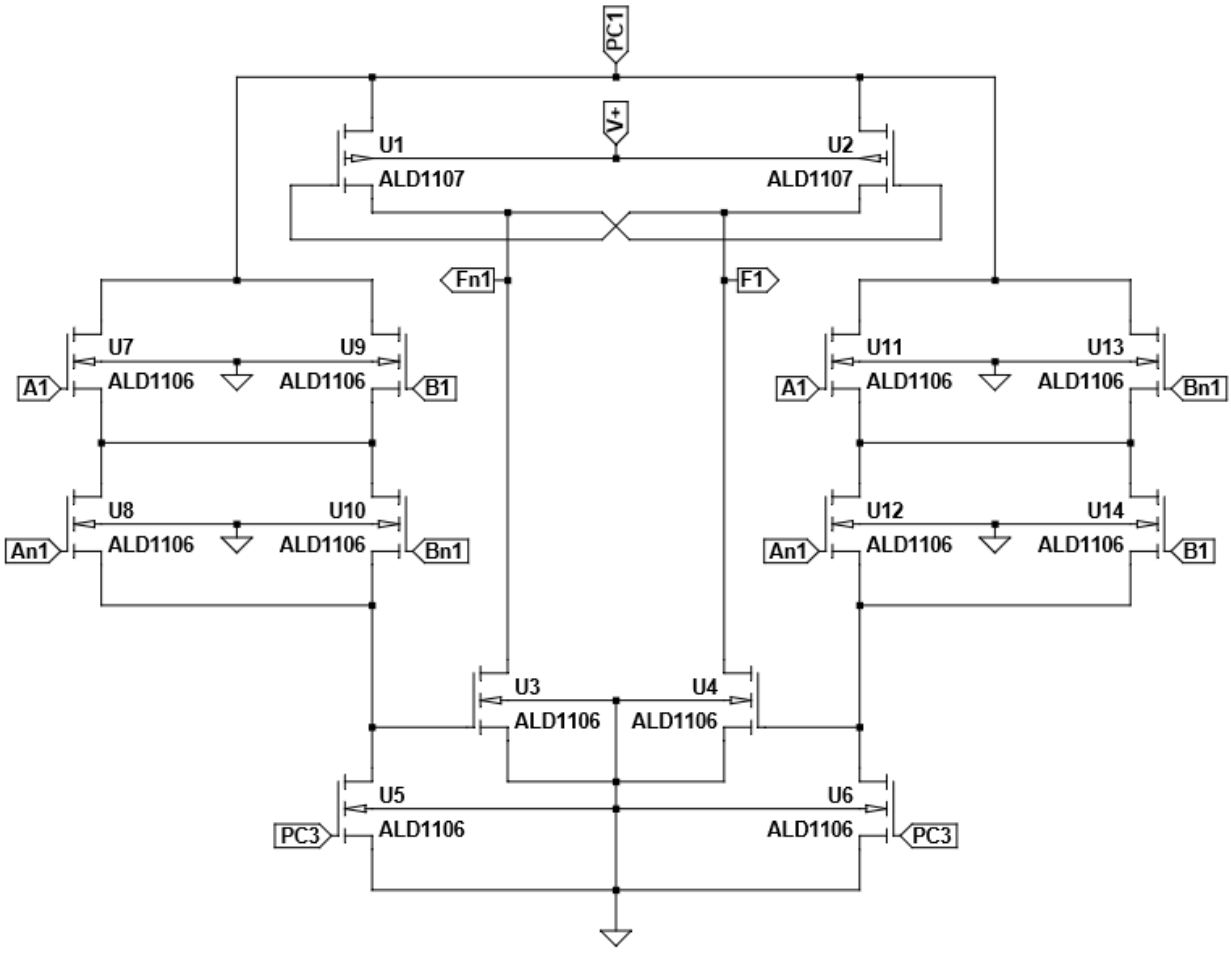


*Figure 12: ALD1106 IDPAL with discharge XOR/XNOR gate*

Figure 13 is a plot of the IDPAL with discharge gate which is constructed from the ALD1106 MOSFETs. This simulation shows that the circuit is producing the correct output

waveform for an XOR gate. Again, when the power clock is decreasing from 3.3 to 0 volts the output follows it until roughly 1.0v because of the PMOS transistors. The output waveform is like the IDPAL waveform expect when the complementary power clock is greater than the threshold voltage of the ALD1106 MOSFETs. At this point, both gates on either NMOS transistors are connected to ground via the additional discharge transistors. The output rises according to the previous output until the complementary power clock is below the threshold voltage of the MOSFET.

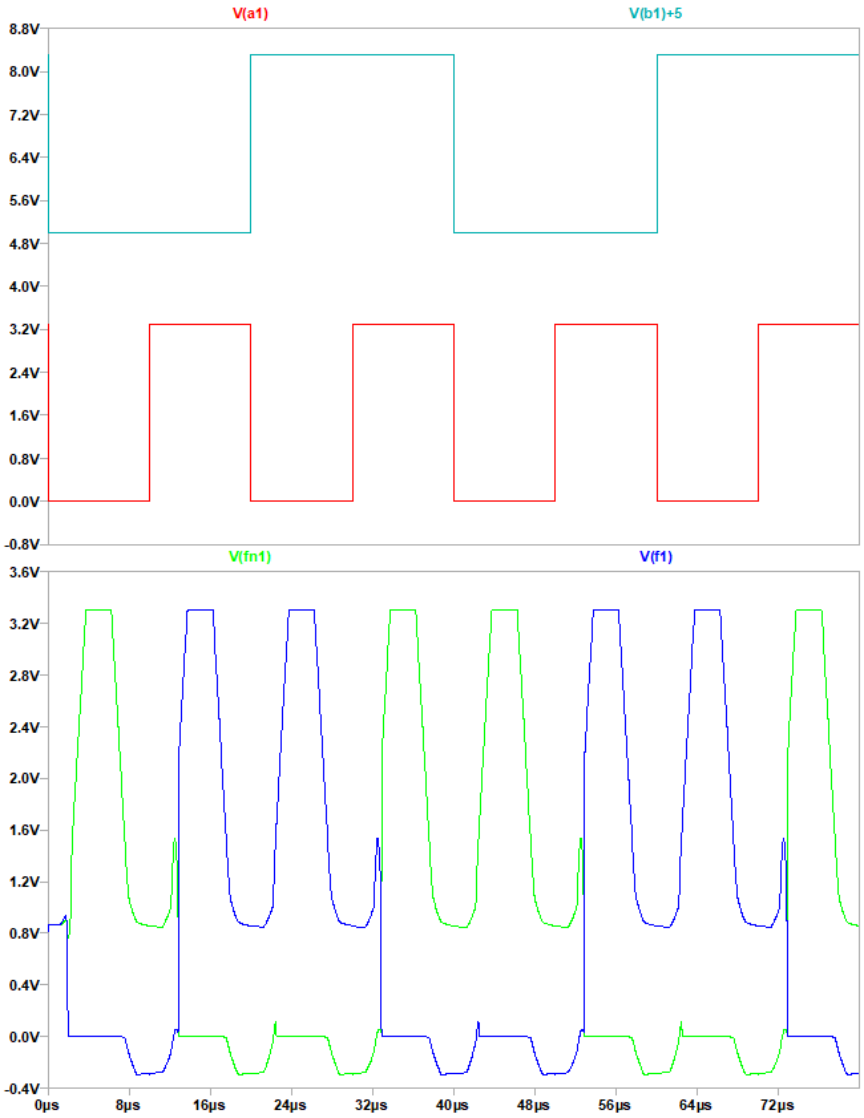


*Figure 13: ALD1106 IDPAL with discharge 100kHz Waveform (simulated)*

The ALD1108xx 2N-2P circuit in Figure 14 has bulk of the EPAD MOSFETs connected to the source. The EPAD MOSFETs used here have a lower threshold voltage than traditional MOSFETS and will likely aid in removing trapped charge from the gate of the ALD1106 MOSFET. The IC containing 4 EPAD MOSFETs had the bulk connections made internally. Also, the sources of a pair of transistors were internally connected. Connecting the bulk to the

source instead of the bulk to ground requires roughly twice the silicon area to create a single transistor. Moving to adiabatic integrated circuits with this design is not desirable. However, there was no alternative EPAD MOSFETs without these internal connections at the present time.

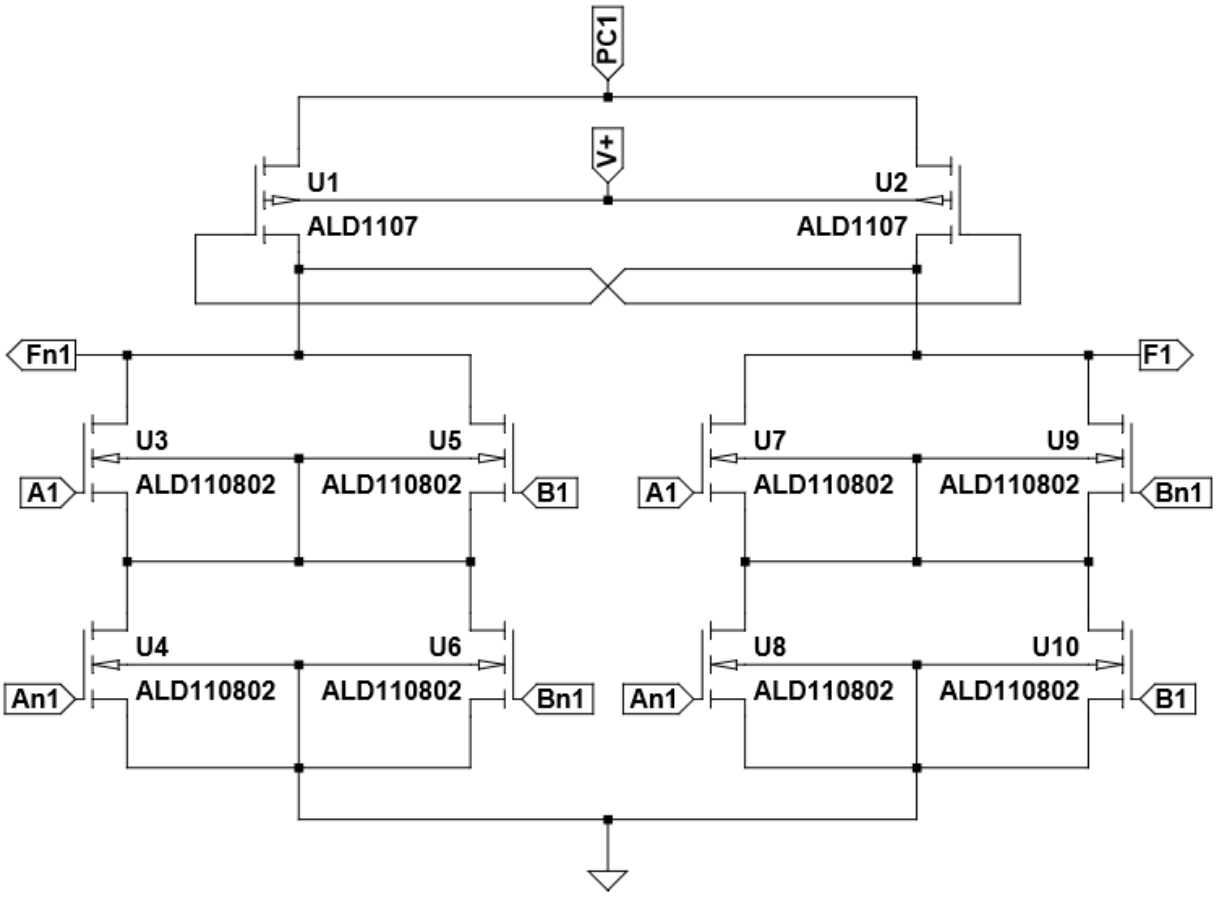


*Figure 14: ALD1108XX 2N-2P XOR/XNOR gate*

The plot in Figure 15 shows the increased fall rate of the output when the PC is in the recover stage. This suggests that the EPAD MOSFETs are able to remove charge faster than the ALD1106 MOSFETs.

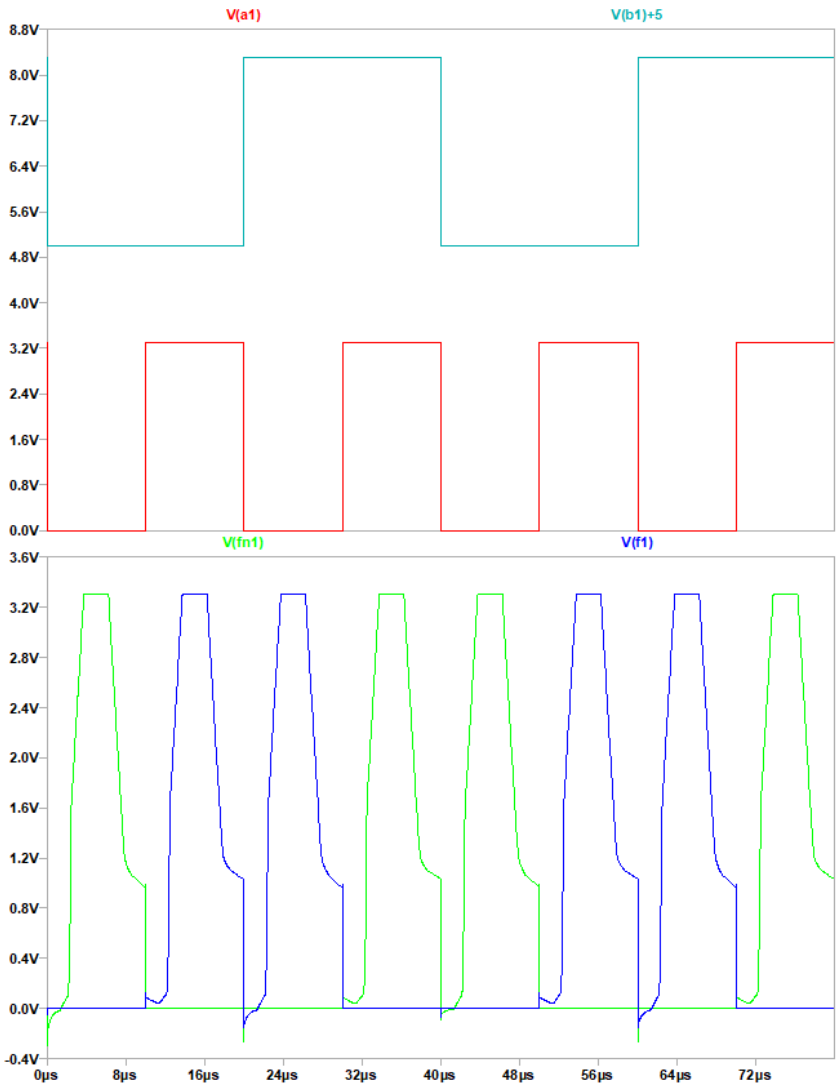


*Figure 15: ALD110802 2N-2P 100kHz Waveform (simulated)*

Figure 16 is the IDPAL XOR gate implemented using ALDs EPAD MOSFETs. This design is the same as the ALD1106 IDPAL model except for the transistor bulk connection.

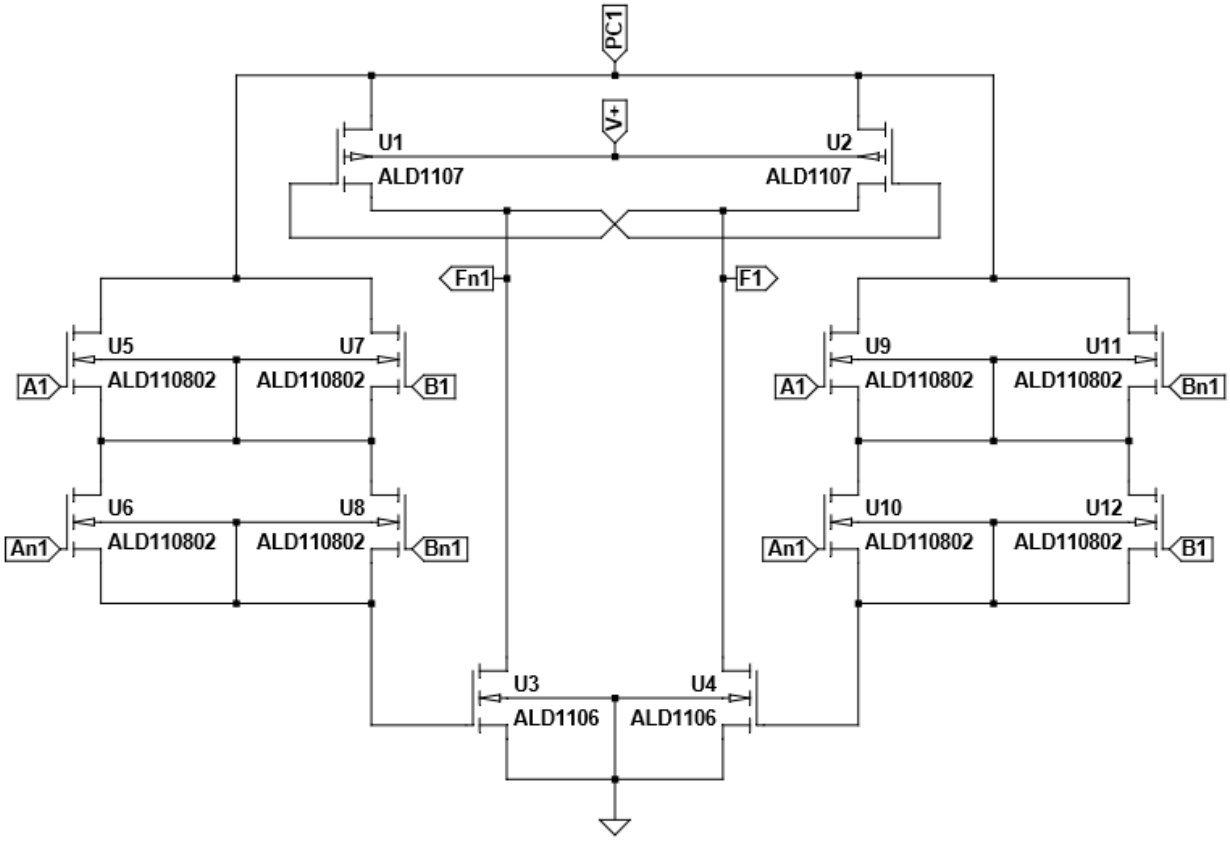


*Figure 16: ALD1108XX IDPAL XOR/XNOR gate*

Figure 17 shows an unexpected output from the EPAD IDPAL implementation. The outputs are less ideal than the ALD1106 outputs. When input A is high the output of the XOR gate reaches 2.6 volts. This indicated that one of the 2 NMOS transistors responsible for driving the output to ground is not functioning properly.

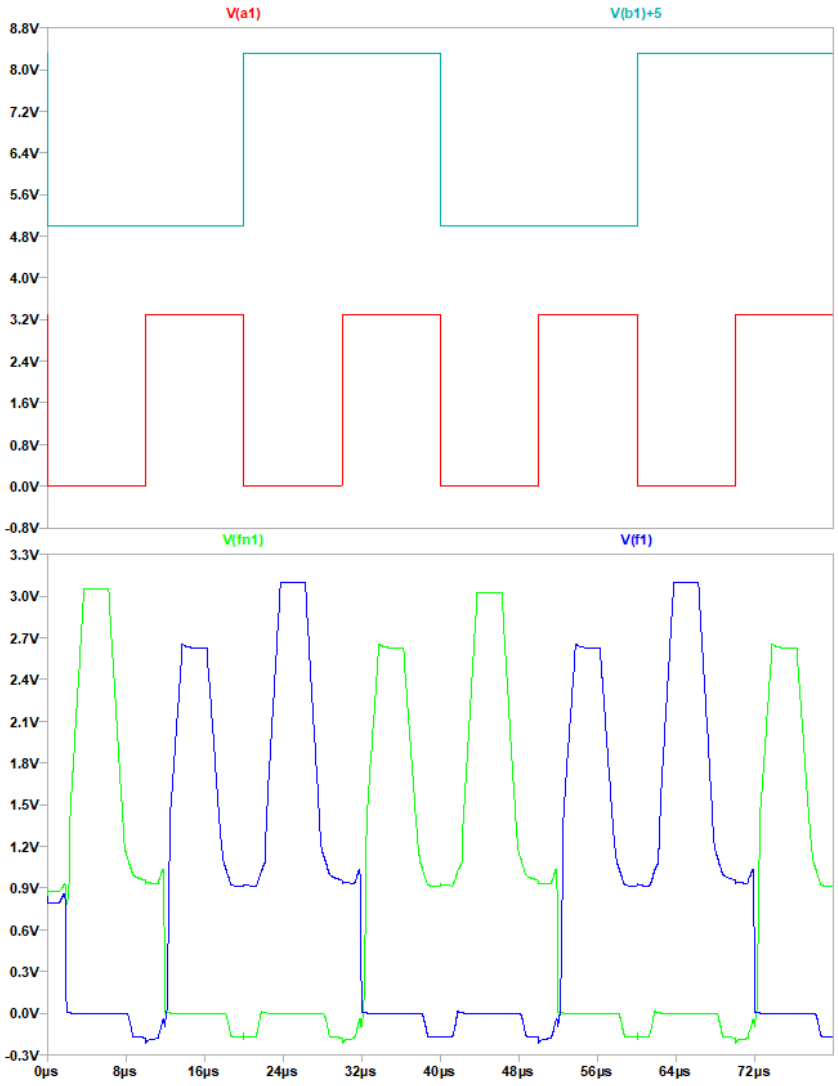


*Figure 17: ALD110802 IDPAL 100kHz Waveform (simulated)*

The ALD1108xx IDAPL with discharge XOR gate utilizes the ALD1106 MOSFETs for discharging because of their smaller silicon footprint and higher threshold voltage.

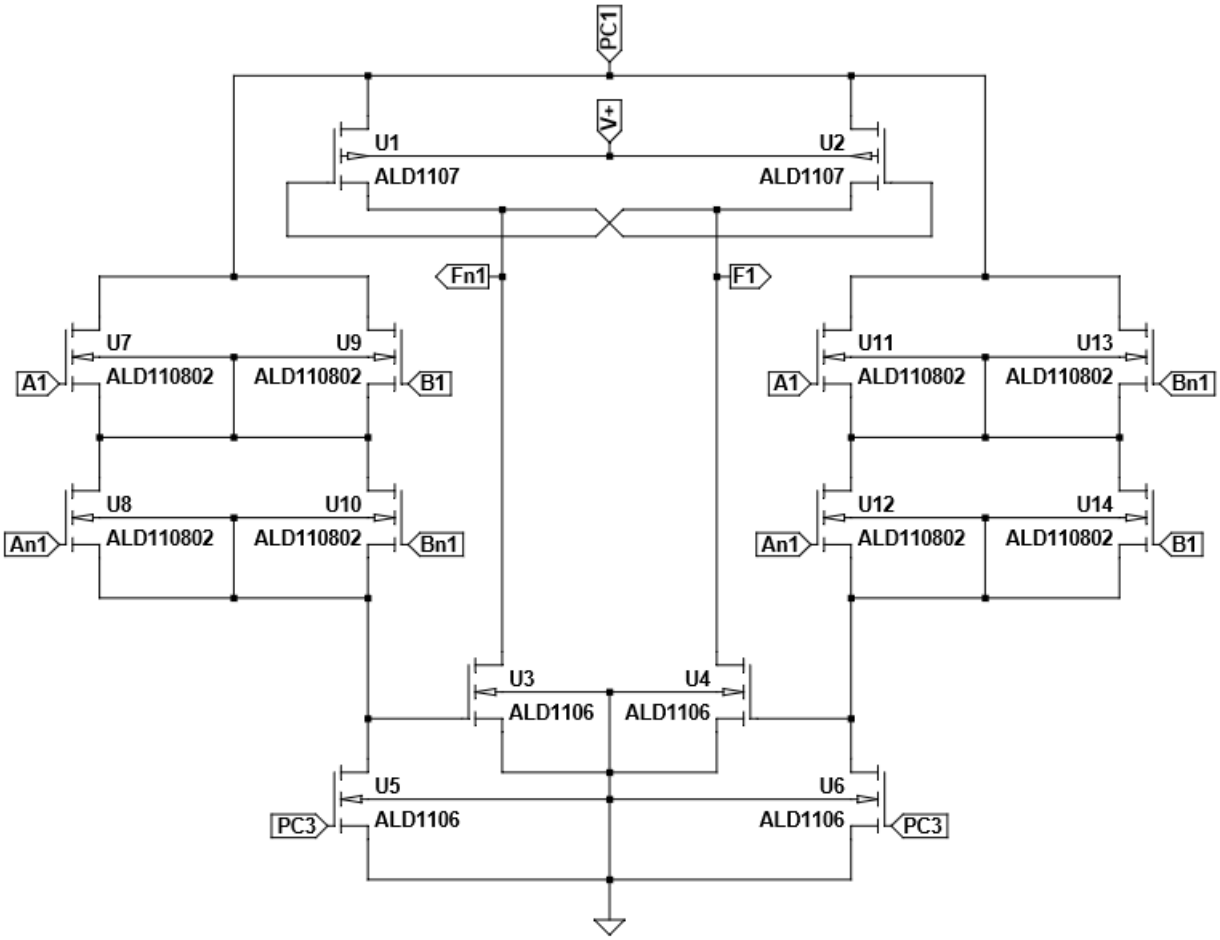


*Figure 18: ALD1108XX IDPAL with discharge XOR/XNOR gate*

Figure 19 is a plot of the IDPAL with discharge implemented with EPAD MOSFETs. The plot shows better functionality than IDPAL with EPAD. The transistor simulation models provided by ALD were extremely difficult to simulate at times which could indicate low fidelity models or an incompatibility with LTspice. Further investigation is required.

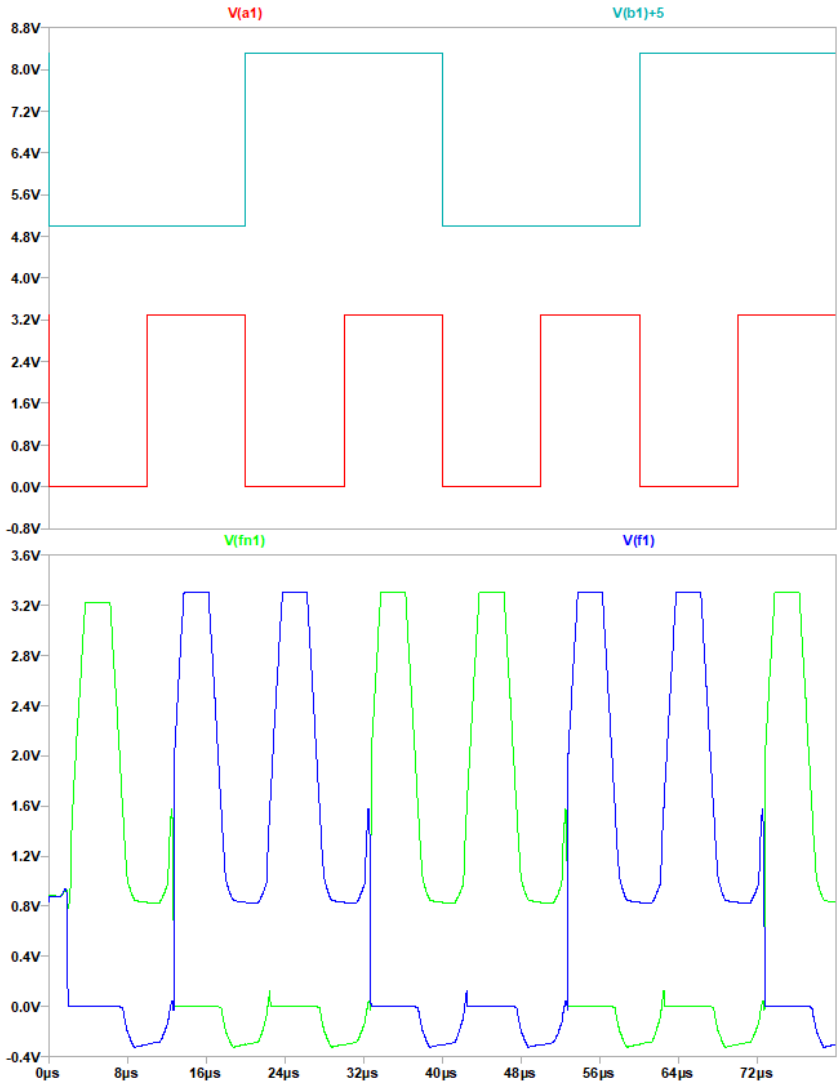


*Figure 19: ALD110802 IDPAL with discharge 100kHz Waveform (simulated)*

# 5. EXPERIMENTAL RESULTS

## 5.1 Printed Circuit Boards

Two Printed Circuit Boards (PCB) were created to match simulation models. One version of the PCB is designed to support the ALD1106 ICs while the other supports the ALD1108xx ICs. Each PCB contains two XOR gates. Each gate can be interfaced through its set of two male headers located at the top of the board. This header is used to provide power, inputs, outputs, and the power clock. Underneath the header, a test point exists for every signal that is entering or leaving the PCB. These test points are used to record the output waveforms in the results section. The MOSFETs used to implement the XOR function are located on either side of the board. The ALD1106 boards in Figure 20 support the 14-pin dual in-line package (DIP) while the ALD1108xx boards in Figure 21 support the 16-pin DIP. The DIP switches allow the user to switch between the different adiabatic models. Standoffs were installed on the PCBs to prevent unintentional connections to be made when the PCB is placed on various surfaces.

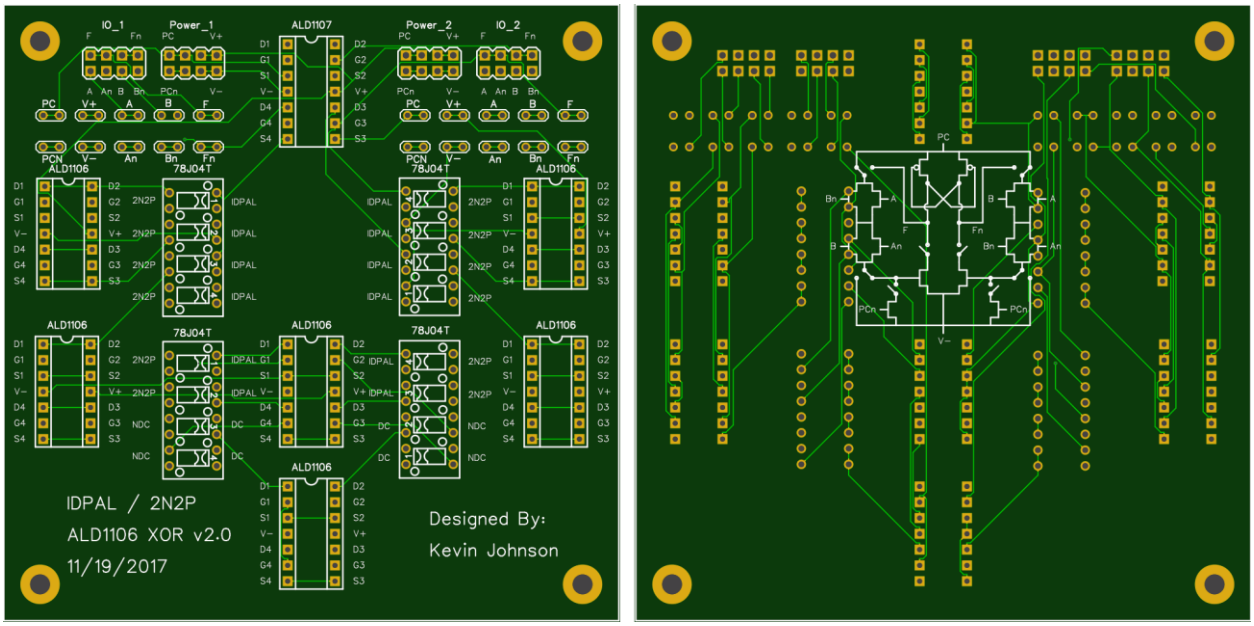


*Figure 20: ALD1106 PCB*

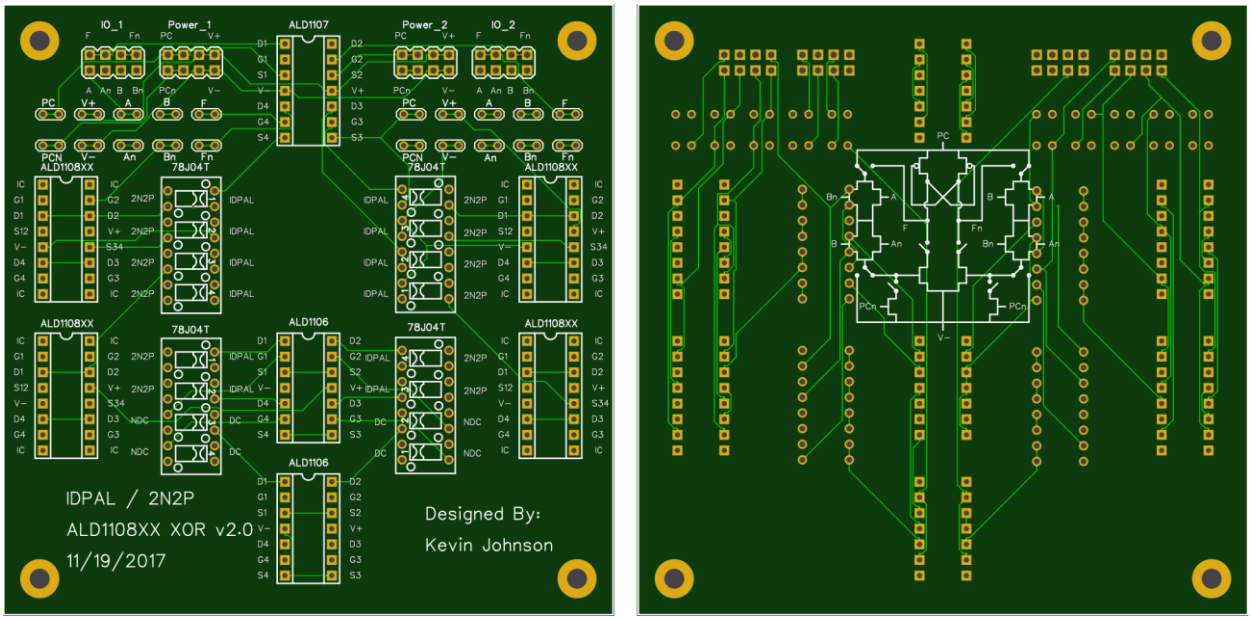


*Figure 21: ALD1108XX PCB*

The boards in previous figures were populated with the 16-pin IC sockets, 2x4 Male Headers, Test Point Loops, and Switches. The IC sockets allow for the ALD1108xx ICs to be replaced by the ALD110800, ALD110802, and the ALD110804. In Figure 22 the sockets are populated with the ALD110804 EPAD MOSFETs.

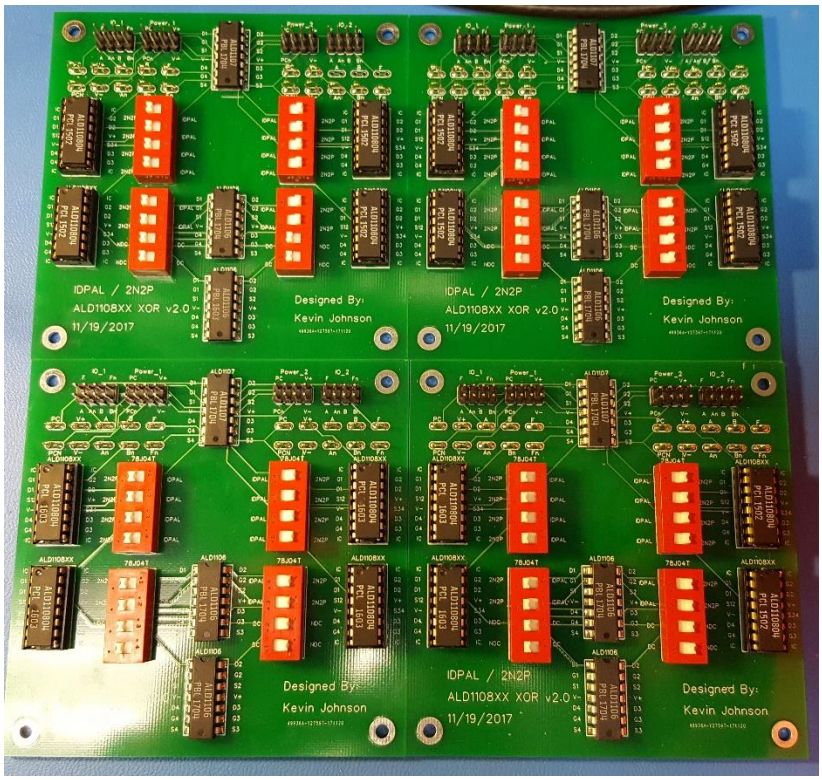


*Figure 22: PCBs populated*

## 5.2 DAC Power Clock

A four-phase trapezoidal power clock generating a voltage between 0 and 3.3 volts at a maximum rate of 100kHz was required to test the adiabatic circuits. Two high-speed digital to analog converters (DAC) were used to generate the four trapezoidal waveforms. The AD9740 DAC was selected because its slew rate is over the required 100 millivolts per 1 micro-second to generate the 100kHz power clock. The AD9740 DAC outputs complementary currents which

reduced the number of DACs in half. A Field-Programmable Gate-Array (FPGA) computed the trapezoidal waveform, supplied the digital value, and the clock for the DAC. The DE0-nano was selected as the FPGA because the speed exceeded the requirements and it was readily available. An operational amplifier was used to scale the voltage from 960mV output of the DAC to the 3.3 volts used for testing. The TLE2084A J-FET Operational amplifier was selected to meet these voltage and frequency requirements. Figure 23 is a simplified schematic that outlines the connections between the different hardware.

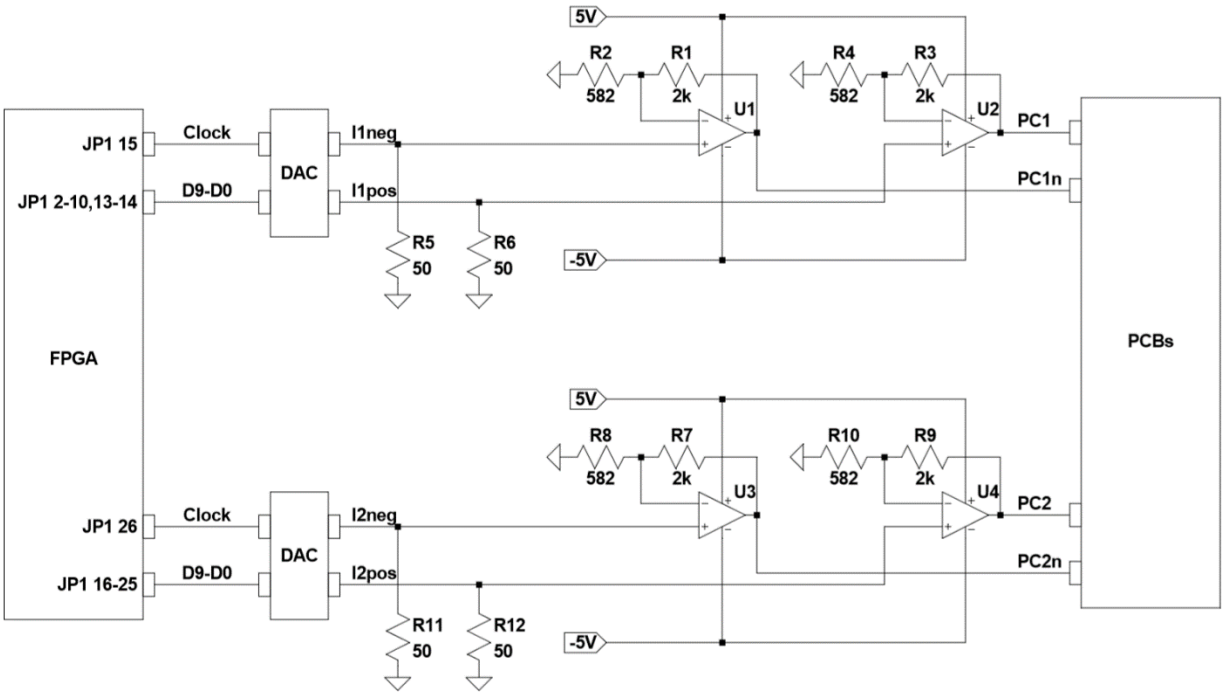


*Figure 23: Power Clock Configuration*

Figure 24 shows the individual nets, passive components, and IC pinouts for generating the power clock. The FPGA 40-pin header is shown on the left of the image. Its purpose is to

periodically supply digital values that form a trapezoidal waveform on the output of the DAC. The AD970 updates the IOUTA and IOUTB outputs on the rising edge of the input clock. The current is passed through an equivalent of 50-ohms to produce a voltage. This output voltage has a resolution of 937.5uV and ranges from 0 to 960mV. The voltage is amplified from 960mV to 3.3V by an operational amplifier. A trim-potentiometer adjusts the gain of the operational amplifier. The final output resolution is approximately 3.223mV per step. Each power clock output is connected to a 100-ohm resistor because driving a capacitive load, such as the adiabatic circuits, would cause overcorrecting oscillations to develop.

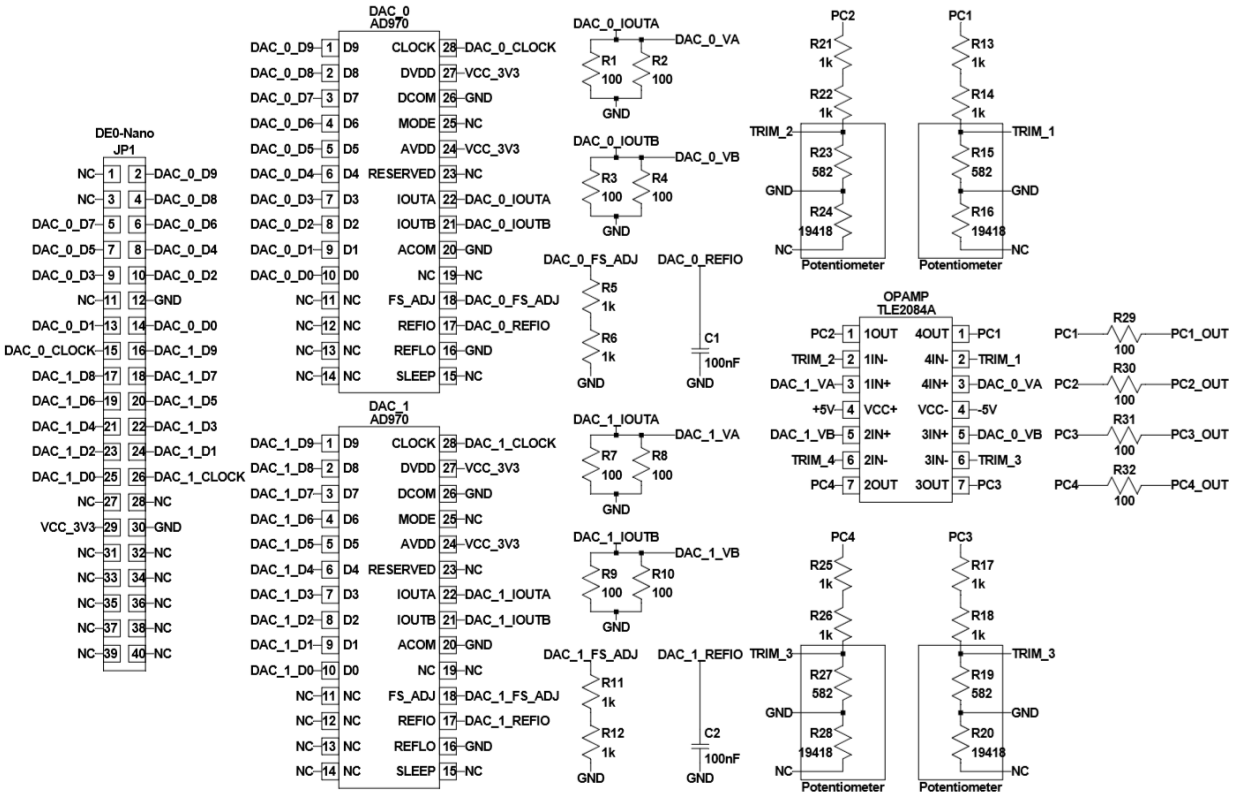


*Figure 24: Power Clock Connections*

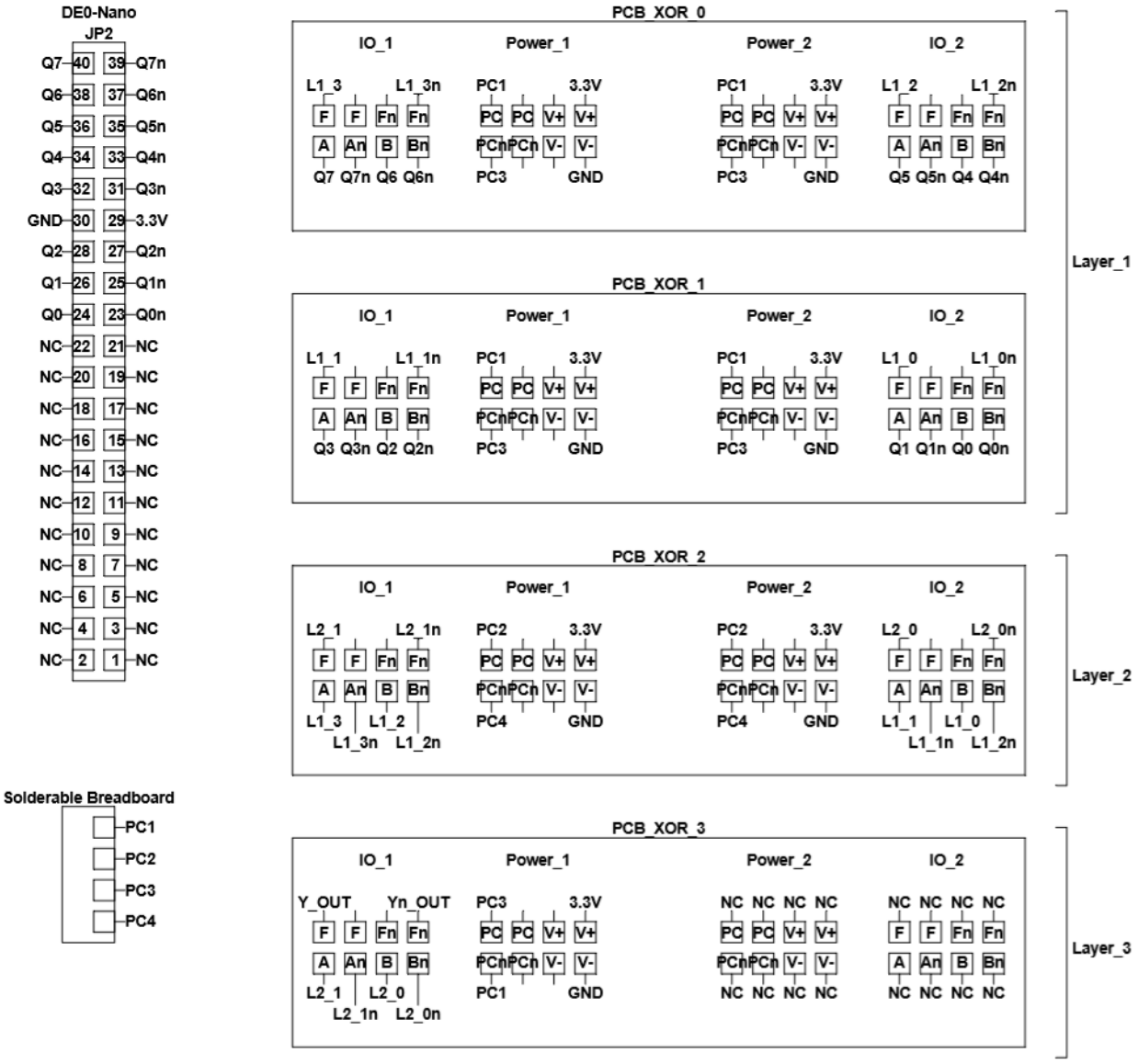


*Figure 25: PCB Connections*

The AD9740 DACs were soldered to SMT Breakout PCBs which were then soldered to a solderable breadboard. Figure 26 shows the final board used in the experiments.

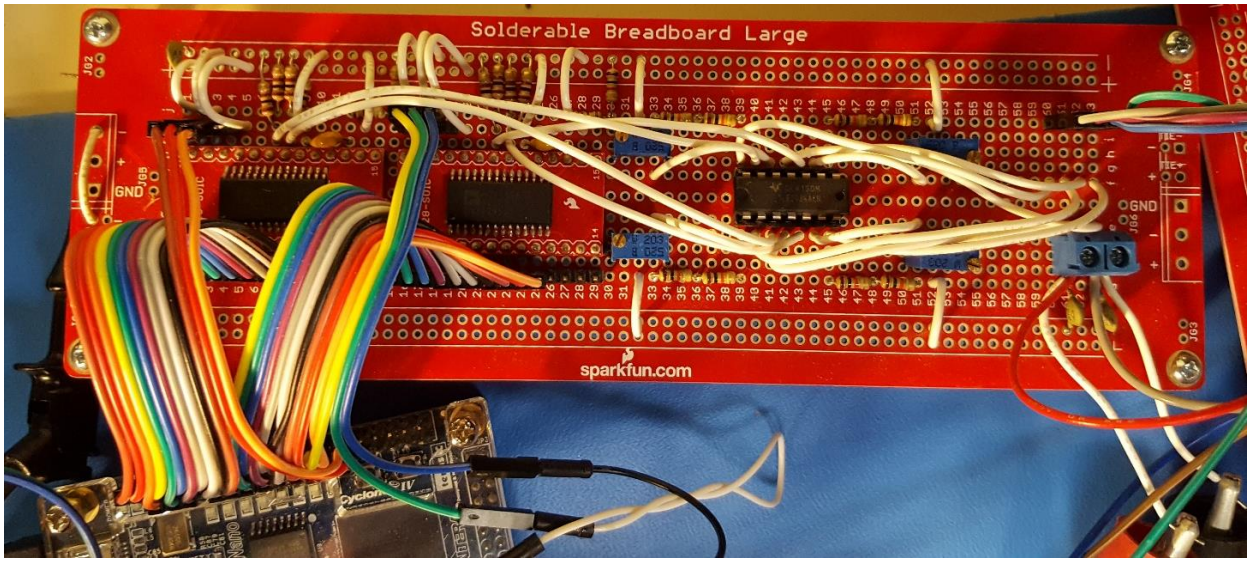


*Figure 26: Solderable Breadboard Power Clock Generation*

**5.3 2N-2P Results**

The 2N-2P results are a baseline for correct operation because they were thoroughly tested in [6]. These results of this chapter show how the ALD MOSFETs behave in the 8-input XOR configuration. Figure 27 shows the LTspice simulated waveform and the oscilloscope recorded waveform overlapping. The Key in the figures below lists four signals F(exp), Fn(exp), F(sim), Fn(sim). The signals with (exp) are the experimental voltages recorded with an oscilloscope while the signals with (sim) are the simulated voltages from LTspice. The signal F(exp) is the output of the XOR function while Fn(exp) is the complement F(exp).

Figure 27 validates that the simulation and experimental results are similar in 2N-2P. This indicates that the ALD1106 MOSFET models used in simulation should match that of most

other adiabatic families. The experimental waveform matches the simulation waveform when the power clock is both rising and falling. The output did not reach the peak of the power clock during the hold period. The circuit appears to be exhibiting a secondary charging behavior because the max voltage appears to be converging on 3.3v. This behavior cannot be attributed to the resistor placed in series with the power clock because the resistor was also accounted for in simulation. The difference in peak voltage is an additional voltage dropped across the PMOS during this hold period. It is unclear as to why the simulation models behave in a more ideal manner during the hold and idle stage of the power clock.

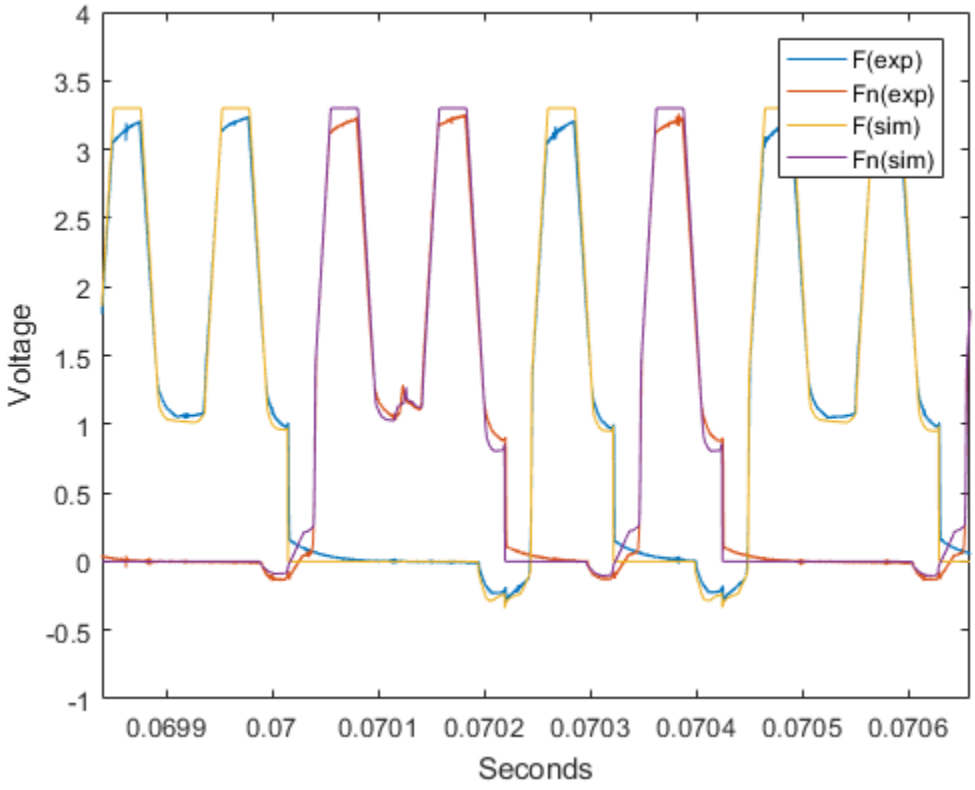


*Figure 27: ALD1106 2N-2P 10kHz Results*

Figure 28 shows the experimental results with the simulation results laid on top for the ALD110804 EPAD MOSFETS. The figure shows that the maximum voltage of 3.3 volts was not reached in the experimental results. It was, however, reached in the simulated models. In the 1 to 2 volt range, the experimental and simulated waveforms continue to diverge from each other. This is in comparison to the ALD1106 results. The valleys of the experimental waveform are closely resembled by the ones observed in the ALD1106 models.

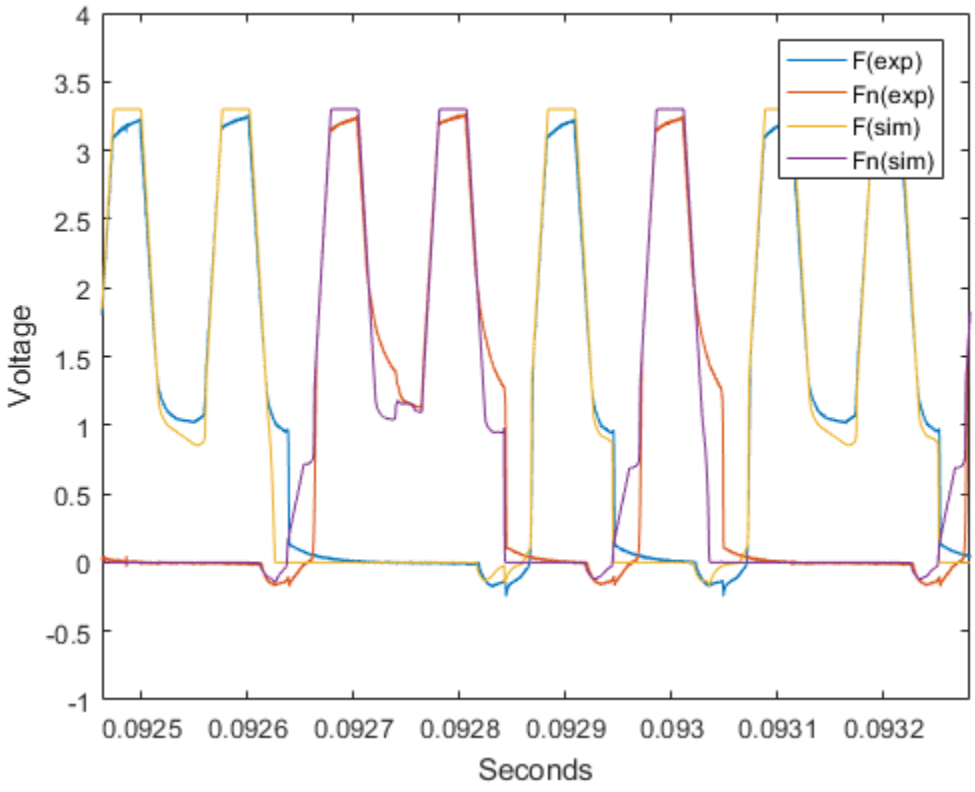


*Figure 28: ALD110804 2N-2P 10kHz Results*

Figure 29 shows the output waveform for the ALD110802 EPAD MOSFETs with a power clock frequency of 10khz. The experimental waveform is significantly different compared to the input when the power clock is in the idle state. The simulation shows the voltage decrease

much further, indicating that the charge is being removed. The experimental waveform appears to reach the same voltage as the ALD1106 MOSFETs and ALD110804 MOSFETs. This voltage is below the desired 3.3 volts referred to in the figure.

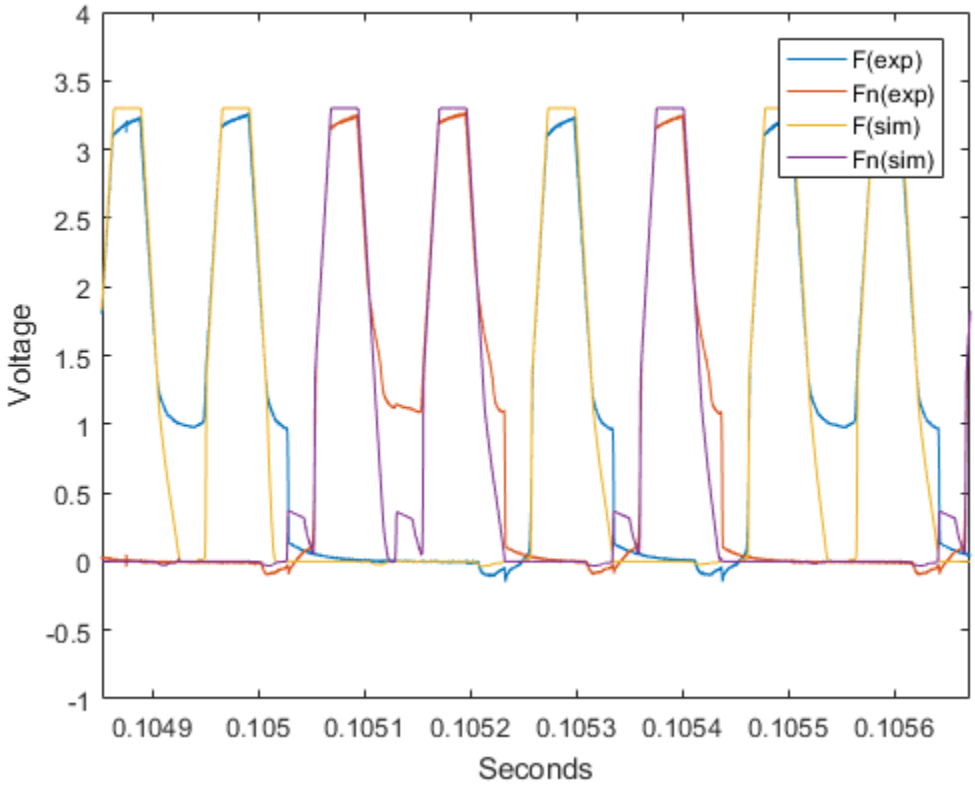


*Figure 29: ALD110802 2N-2P 10kHz Results*

In Figure 30, the experimental waveform closely matches the simulated waveform during the charging stage of the power clock for the ALD110800. The trend of discharging faster than the experimental model that has been seen previously, continues to be seen in this model, as well. The EPAD MOSFETs are shown to have a higher voltage during the recovery stage than the traditional MOSFETs.

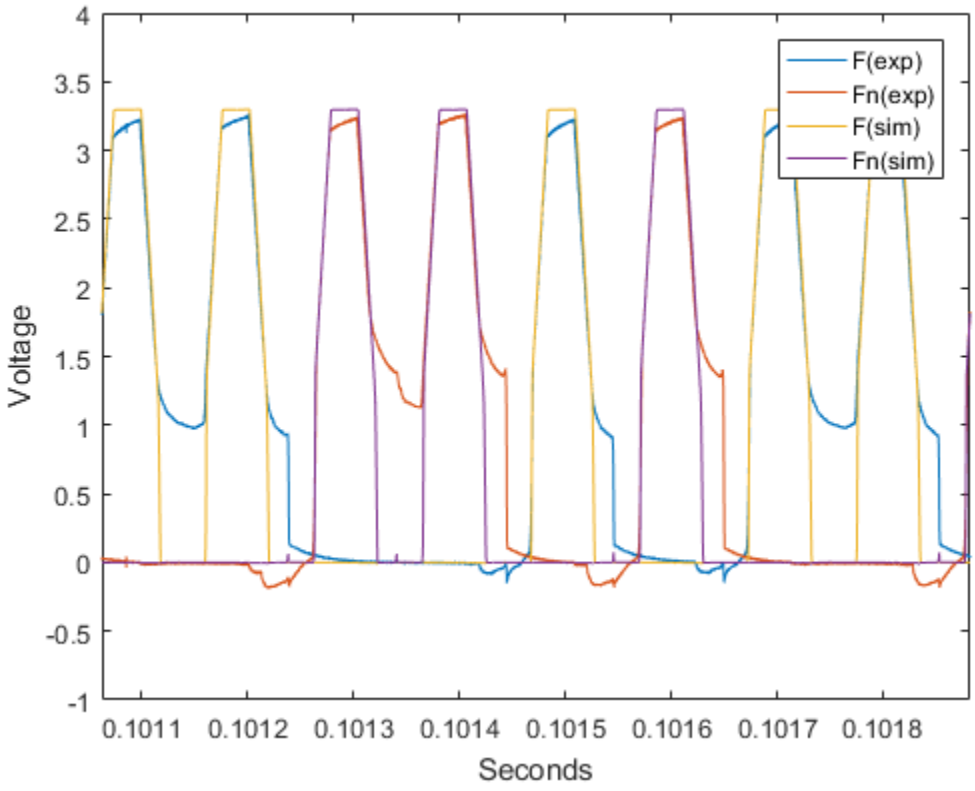


*Figure 30: ALD100800 2N-2P 10kHz Results*

Figure 31 shows that the peak voltage of the experimental models does not match the peak voltage of the simulation models. The rise and fall of the voltage tends to be relatively close. The simulated dip below 0 volts does not actually happen in the experimental results. The experimental results and the simulated results both tend to have the same general shape to them, indicating that the simulation and experiment were not too far off.

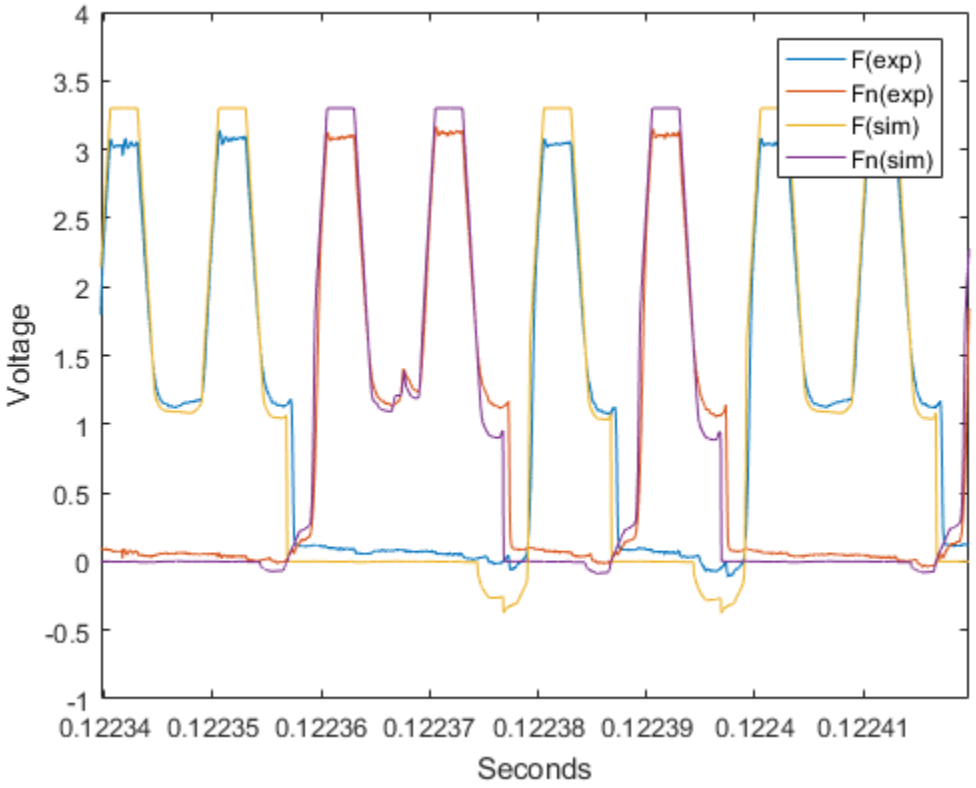


*Figure 31: ALD1106 2N-2P 100kHz Results*

Figure 32 shows the EPAD MOSFETS. The simulated and experimental voltage did not match up. The experimental voltage only reached around 3.0 volts, whereas the simulated voltage was 3.3 volts. There were a few dips below 0 volts for the simulated data, and though they did not perfectly match up, there were also dips below 0 volts for the experimental data. The

experimental and simulated data had some similarities, but they show a significant divergence. There were a few times when decreasing voltage made the experimental data very inconsistent with the simulated data.

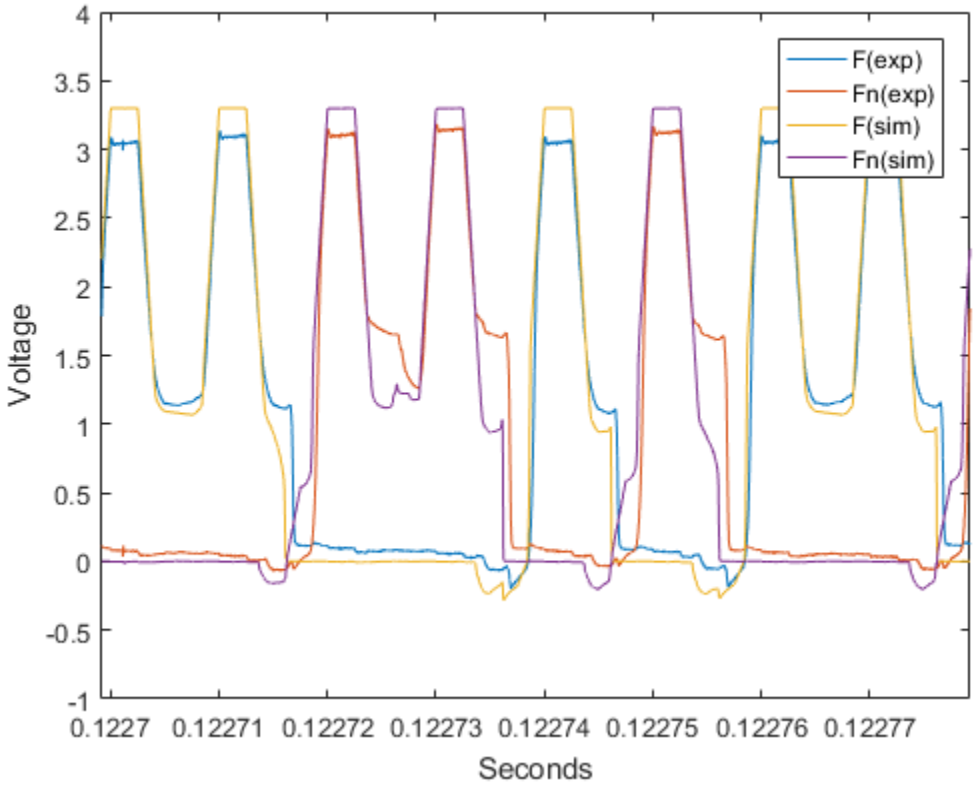


*Figure 32: ALD110804 2N-2P 100kHz Results*

Again, the peak between the experimental and simulated data were inconsistent in Figure 33. The simulated peaks were around 3.3 volts, whereas the experimental peaks only reached about 3.0 volts. An increase in voltage for simulated and experimental results tend to be relatively close. There were only a few major discrepancies between the simulated and experimental results. Decreasing voltages was much more inconsistent. It usually took longer for

the experimental voltages to decrease than it did for the simulated volts to decrease. Resting at 0 volts was close, but not perfect between the four models.

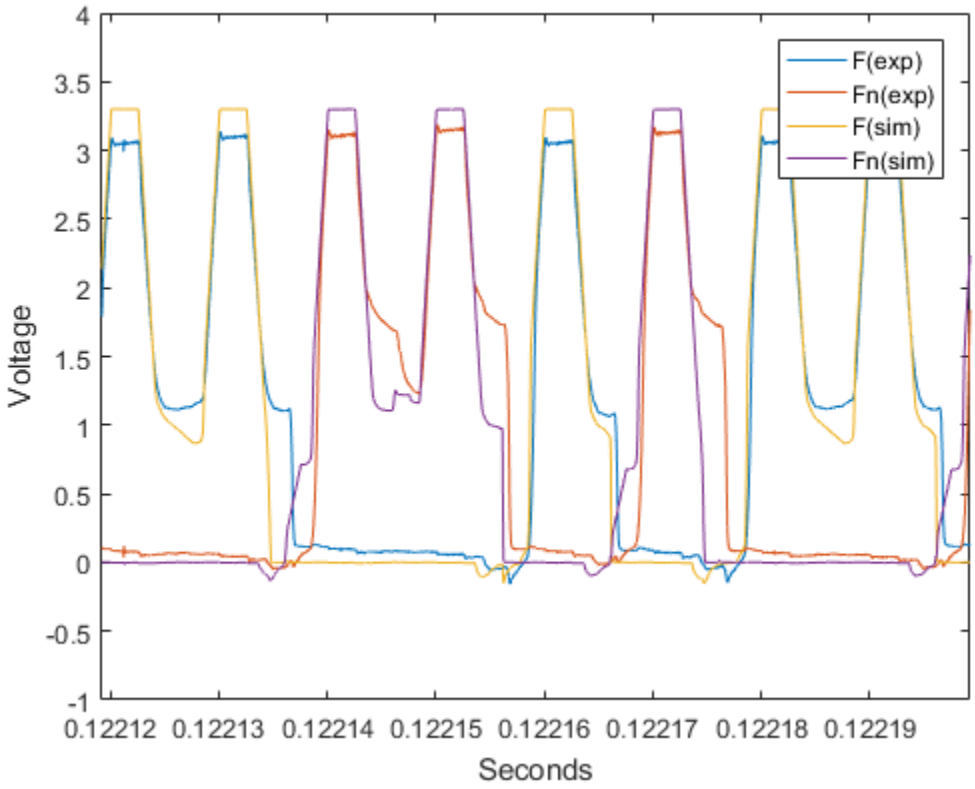


*Figure 33: ALD110802 2N-2P 100kHz Results*

The 2N-2P results at 100kHz were very inconsistent, as shown by Figure 34. For the most part, the experimental results and the simulated results diverged as they did with the 10kHz case. The experimental results never reached a peak of 3.3 volts. The rate in which the experimental output voltage fell was inconsistent with the simulation. Experimental results appeared to dip below 0 volts more frequently than simulated results. This is likely caused by the transistor being off and secondary effects of a voltage applied to one terminal.

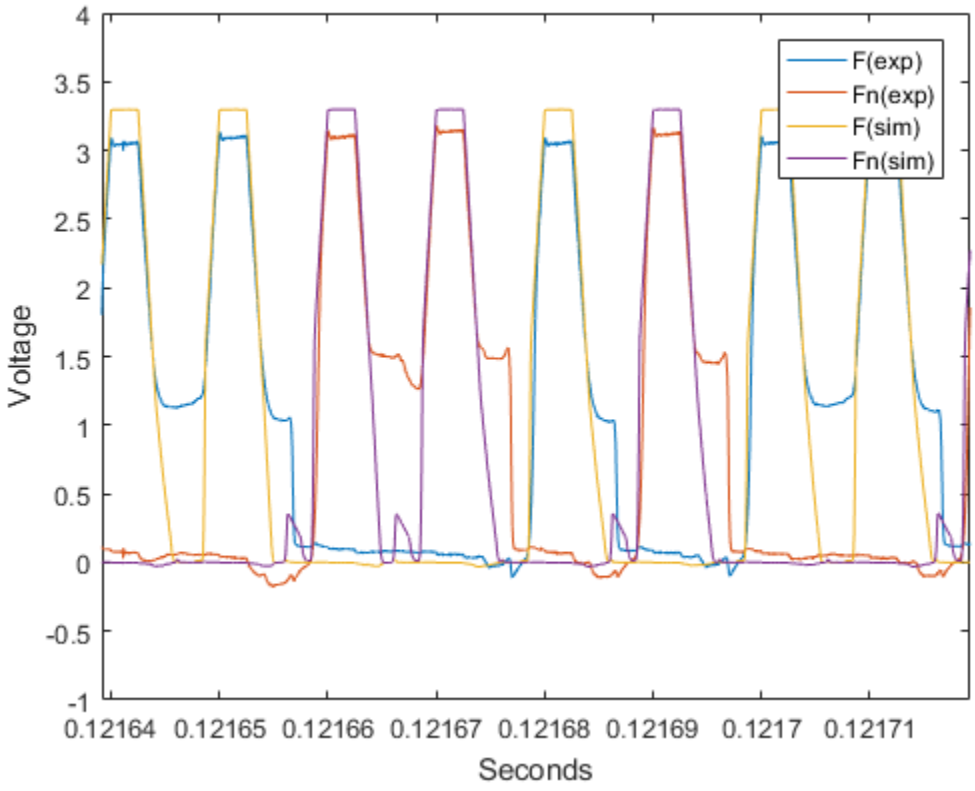


*Figure 34: ALD110800 2N-2P 100kHz Results*

### 5.4 IDPAL Results

The IDPAL 10kHz result in Figure 35 shows that the circuit is preforming correctly. Simulation shows incorrect behavior with the output F rising with each power clock. Another observation is the output waveform in the simulation does not idle at 1 volt. Again, the experimental voltages never peak at the simulated 3.3 volts. Experimental and simulated results appear to be close when the experimental results reached 0 volts, but the simulation output does not drop below 0 volts. The results for this model were surprising. The experimental results show

that the circuit works correctly, despite simulation results showing the circuit fails to operate correctly. It is surmised that this is a result of the level 2 spice models provided by ALD.

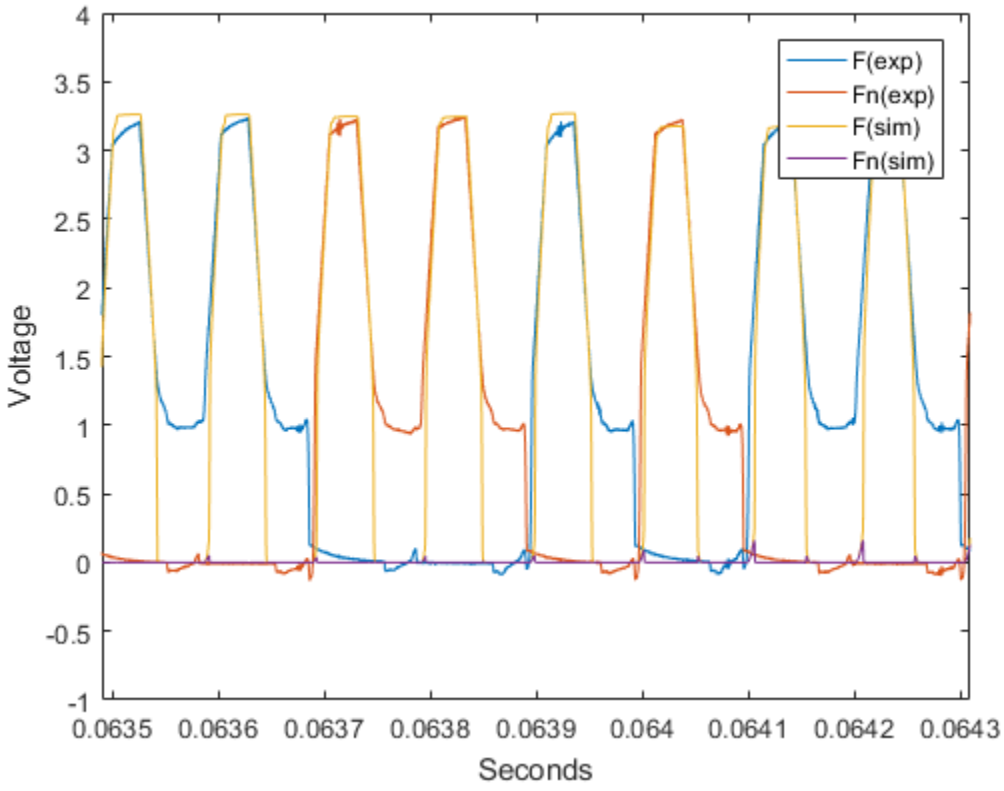


*Figure 35: ALD1106 IDPAL 10kHz Results*

Figure 36 displays the results for IDPAL implemented with ALD110804 EPAD MOSFETs. Although the waveform appears worse that the ALD1106 results, the simulation did come close to matching the experiment. For the most part, the experimental and simulated data tend to rise and fall together. When the simulation model has the voltage dipping below 0, the

experimental model is usually dipping below 0 as well. The experimental peak and simulated peak appear to be close to each other in maximum voltages.

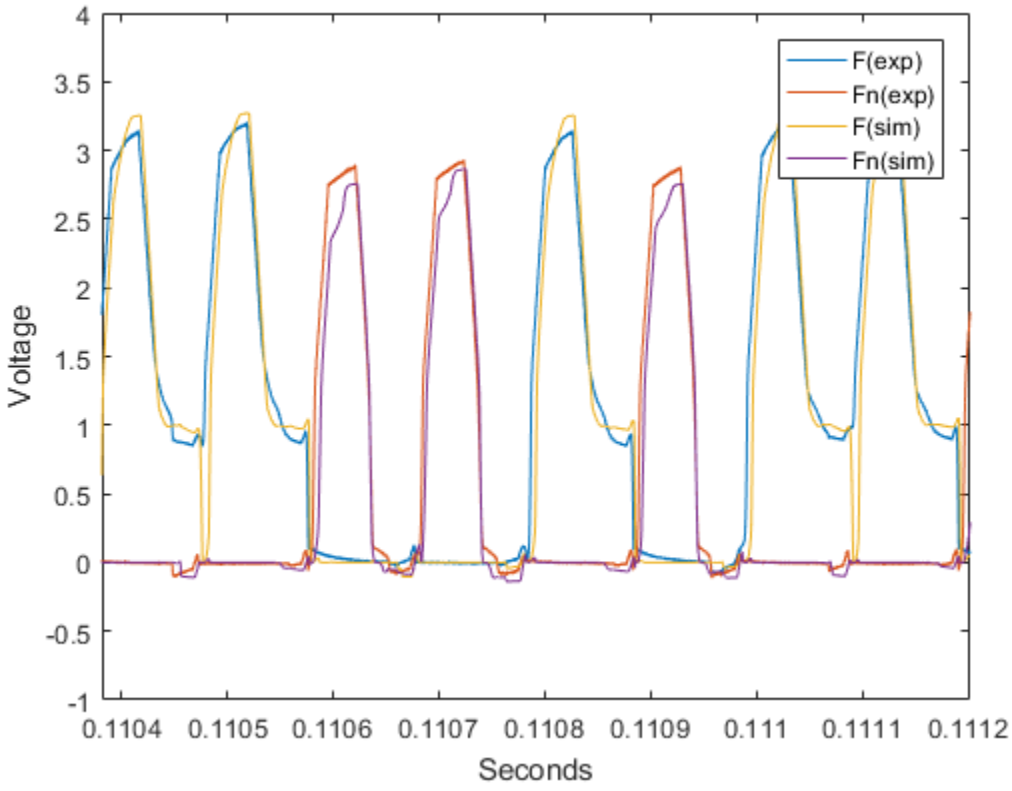


*Figure 36: ALD110804 IDPAL 10kHz Results*

Figure 37 shows some discrepancies again, but both the experimental and simulated waveforms reach higher voltages. Although rising at a consistent rate, the voltage seems to drop inconsistently between simulated and experimental data. The experimental output rested at a lower voltage when the power clock was in the idle state. However, the 2n2p models with EPAD MOSFETs showed experimental outputs with a higher voltage during the idle time.

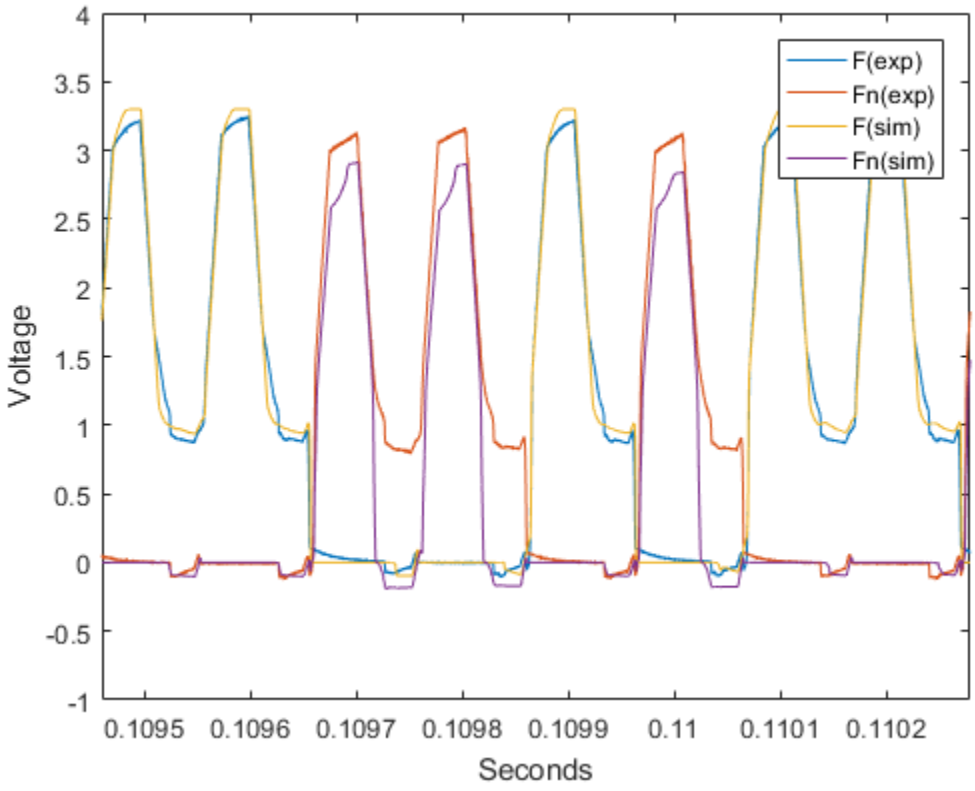


*Figure 37: ALD110802 IDPAL 10kHz Results*

Figure 38 shows simulated and experimental models very close to each other. This model utilizes ALD110800 with IDPAL at 10 kHz. The experimental and simulated results appear to rise and fall consistently, as well as reach a close, consistent peak. Simulated models are reaching a peak of 3.3 volts, while experimental models are reaching a peak just slightly below 3.3 volts. The experimental and simulated results also show similar dips below 0 volts.

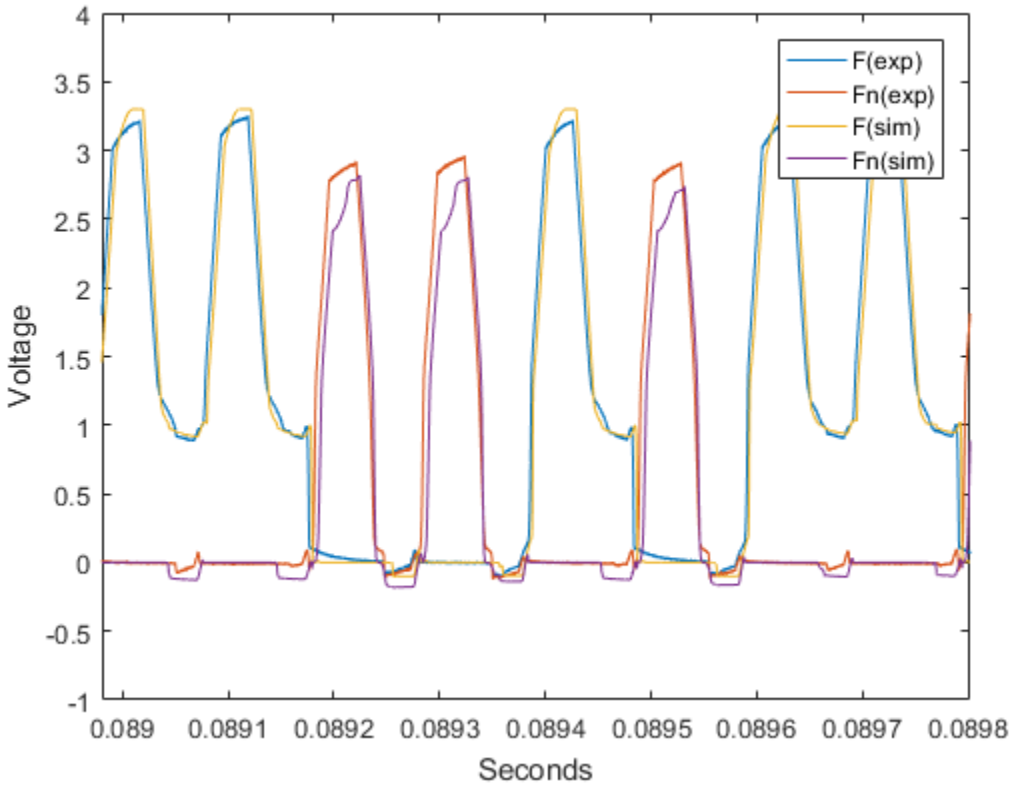


*Figure 38: ALD110800 IDPAL 10kHz Results*

IDPAL implemented with the ALD1106 MOSFETs running with a power clock frequency of 100kHz functions incorrectly. Figure 39 shows that both the experiment and simulation outputs are outputting a constant value. This result is expected given that the simulation failed to operate at a lower frequency. The experimental peak voltage appears to rise with each power clock.

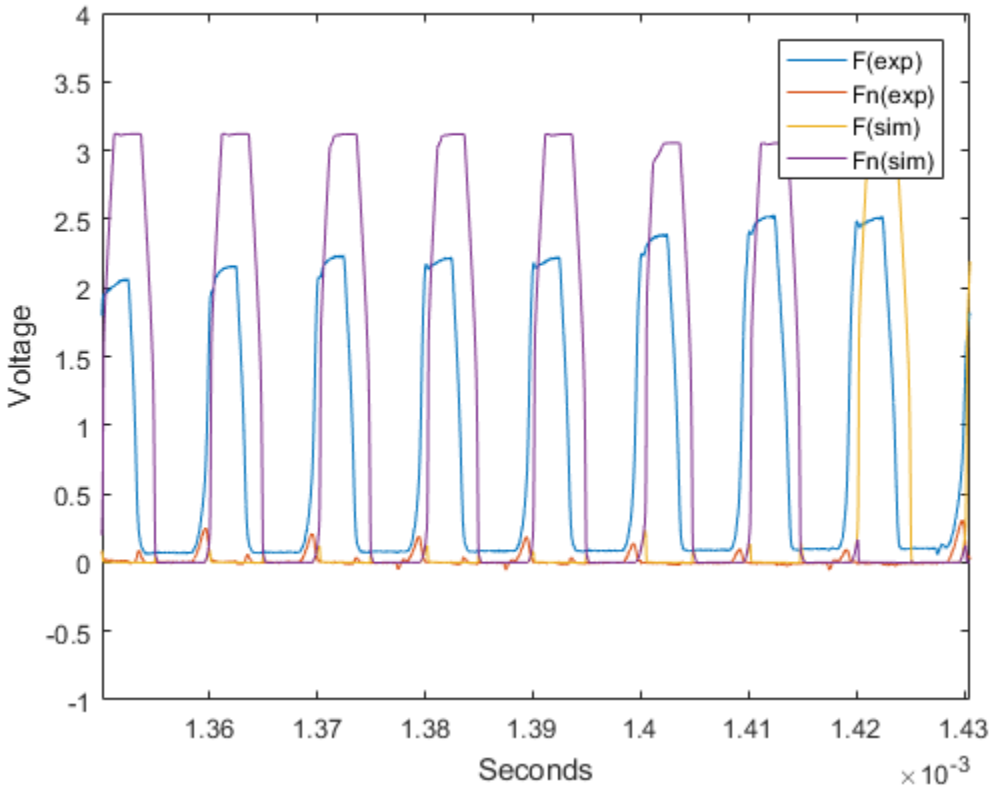


*Figure 39: ALD1106 IDPAL 100kHz Results*

IDPAL designed with ALD110804 EPAD MOSFETs did not function correctly at 100kHz. Simulation shows better performance in Figure 40 than the experiment. This is contradictory to the ALD1106 model running at 10khz. The general shape of the waves appears to match, but since the experimental output is already in a state of failure, it is hard to tell.

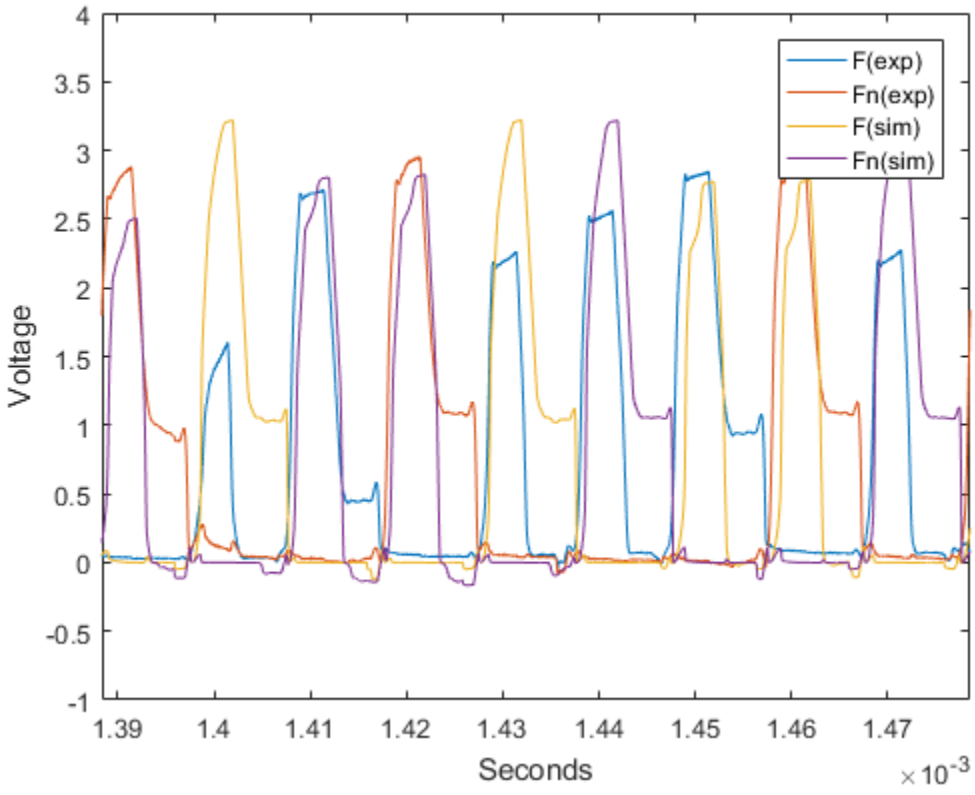


*Figure 40: ALD110804 IDPAL 100kHz Results*

Figure 41 shows the experimental IDPAL circuit with the 0.2 volt threshold MOSFETs has failed to function properly at 100kHz. This is more evidence for the inconsistency between the simulated and experimental results when the circuit begins to fail. The peaks are normally within 0.25 volts of each other, but there are a few peaks that are very off. Most of the time, voltage appears to rise and fall within a small range of each other, but again, there are a few variances with this. Experimental data does not appear to make it close to 0 volts before rising again, which differs from the simulation results. Simulation has more promising results. Neither of the waveforms look close to ideal and would not be suggested in the development of a more complex circuit.

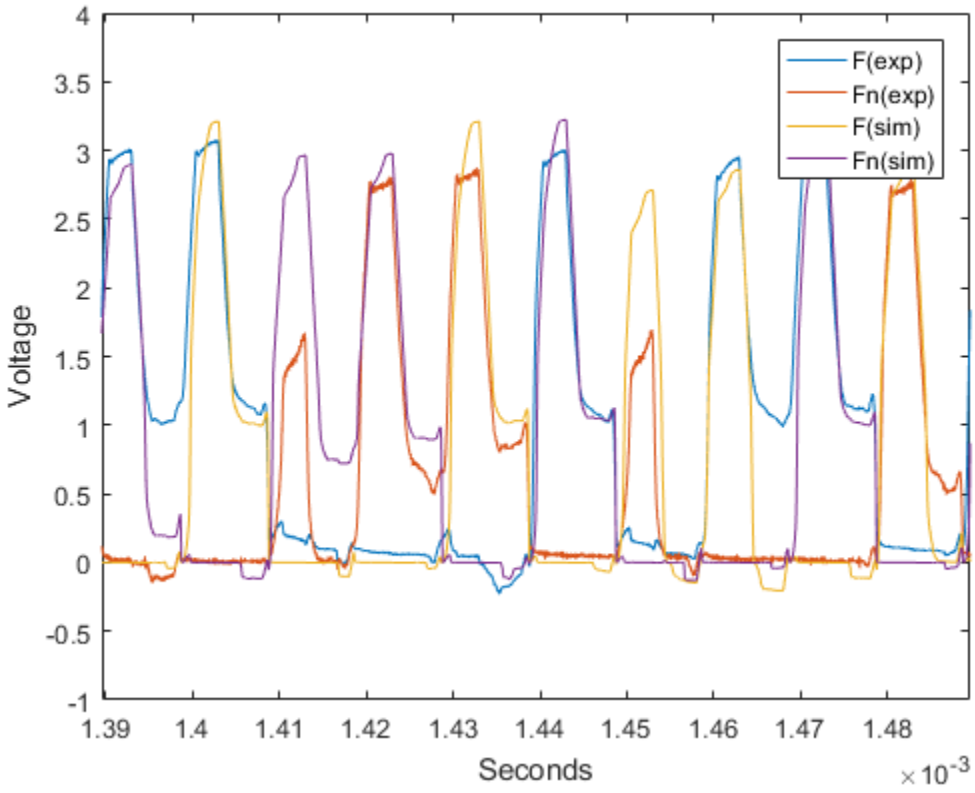


*Figure 41: ALD110802 IDPAL 100kHz Results*

The experimental results for IDAPL created with ALD110800 MOSFETs performed better than the other EPAD MOSFETs, but the output waveform is far from reaching the peak voltage. Experimental and simulation waves do not appear to follow much of the same shape in Figure 42. Simulation peak is sometimes higher than the experimental peak, while other times, the opposite is true. Experimental and simulation decreasing voltages typically do not match up. Increasing voltages are closer than decreasing voltages, but still are not near where they should be.

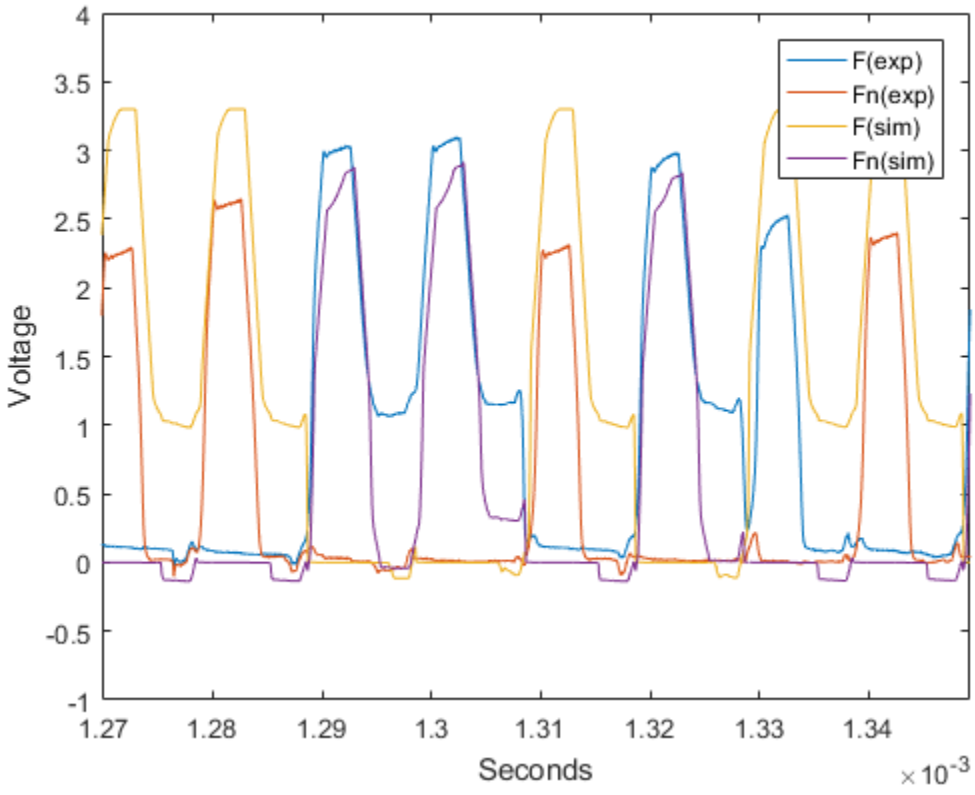


*Figure 42: ALD110800 IDPAL 100kHz Results*

### 5.5 IDPAL with Discharge Results

The results shown in Figure 43 for the IDPAL with discharge implemented with ALD1106 MOSFETs are promising. The experimental and simulation voltages peak very close to each other, though they are not perfect. Voltages rise and fall at seemingly the same rate. When the simulation voltage dips below 0, the experimental voltage follows. The circuit operates as expected.

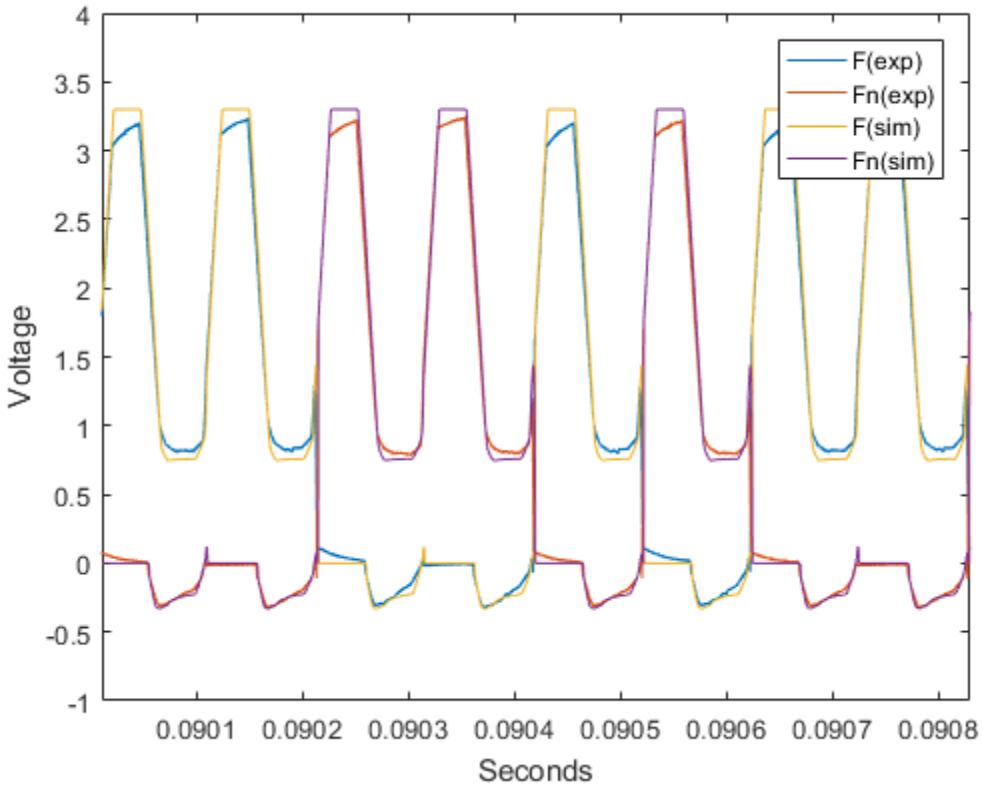


*Figure 43: ALD1106 IDPAL with Discharge 10kHz Results*

Figure 44 shows promising results for utilizing EPAD MOSFETs in IDPAL with discharge. Simulation and experimental models seem to rise and fall together. Peak voltages do not perfectly line up with each other, but they are within 0.25 volts of each other. Dips below 0 volts are consistent between simulation and experimental results. The circuit performs correctly.

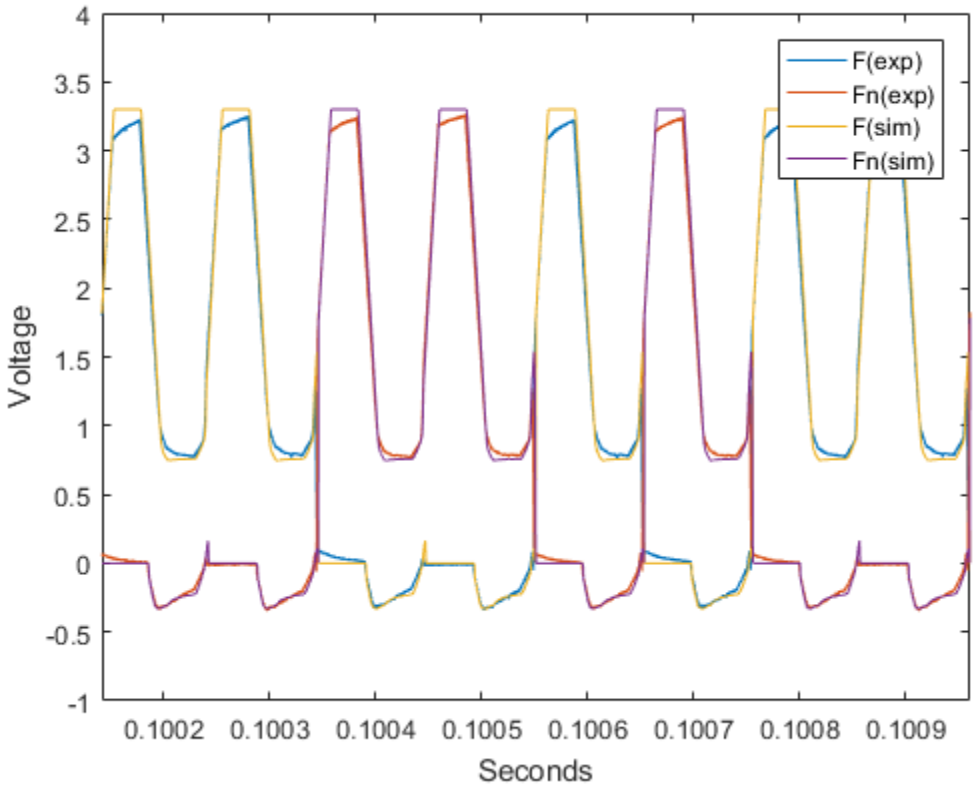


*Figure 44: ALD110804 IDPAL with Discharge 10kHz Results*

Voltage increase and decrease between experimental and simulation results remain consistent in Figure 45. For the most part, dips below 0 volts are consistent between simulation and experimental models. Experimental and simulation peaks are within a decent range of each other. Experimental and simulation models for IDPAL with discharge at 10 kHz show promising data.

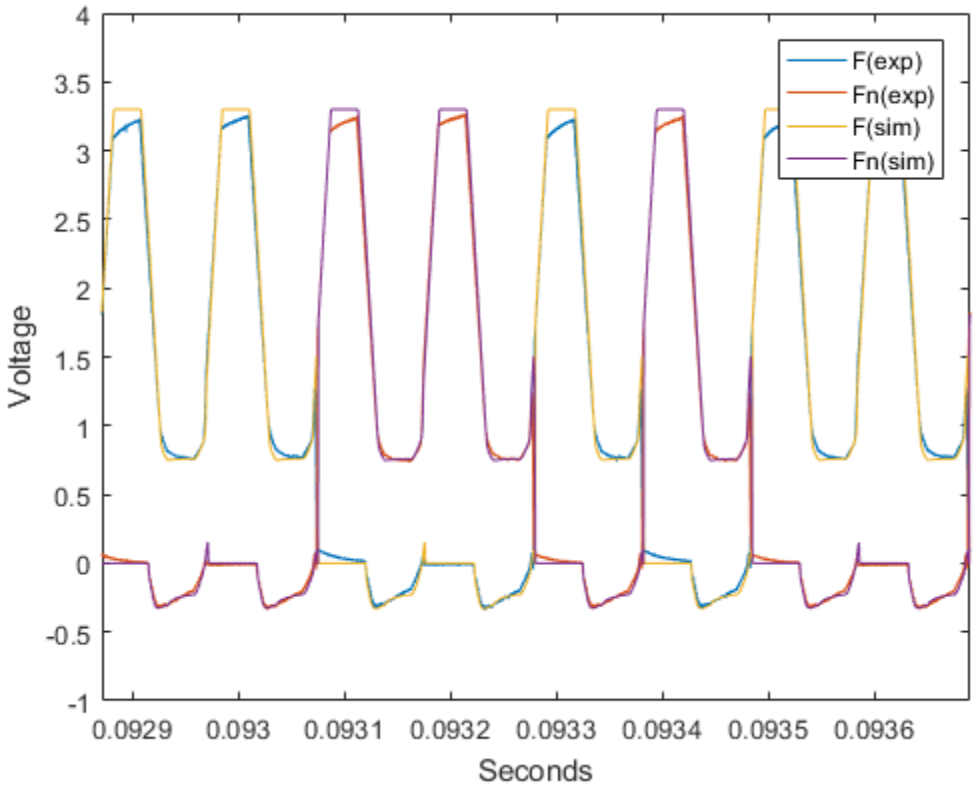


*Figure 45: ALD110802 IDPAL with Discharge 10kHz Results*

Figure 46 is beginning to show a little more variation between simulation and experimental data. The peaks between simulation and experimental data are no longer matching up as nicely. Increasing voltages are still looking very similar between simulation and experimental data. Decreasing voltages are looking just as good as increasing voltages. When the voltage starts to approach 0 volts is where there is slight variation between simulation and experimental results.

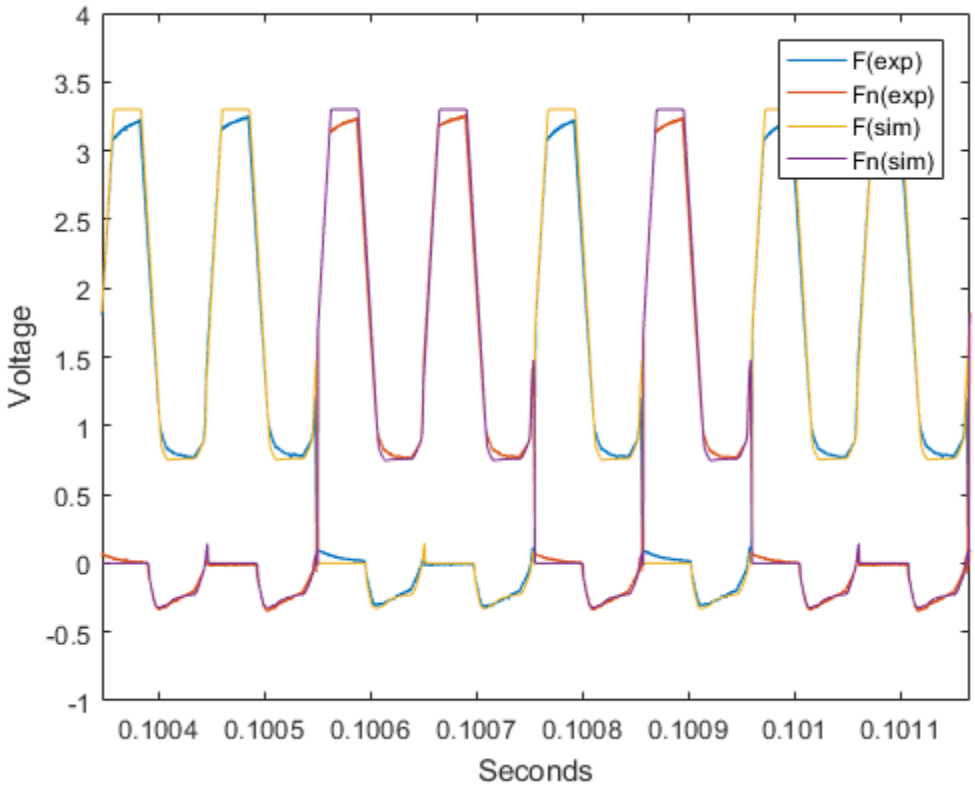


*Figure 46: ALD110800 IDPAL with Discharge 10kHz Results*

Figure 47 shows experimental peaks around 3.0 volts, and simulation peaks around 3.3 volts. There is some variation between simulation and experimental models as the voltage reaches 0. Increasing voltages seem to be near the same for both experimental and simulation. Decreasing voltages between the two models seem to be identical as well.

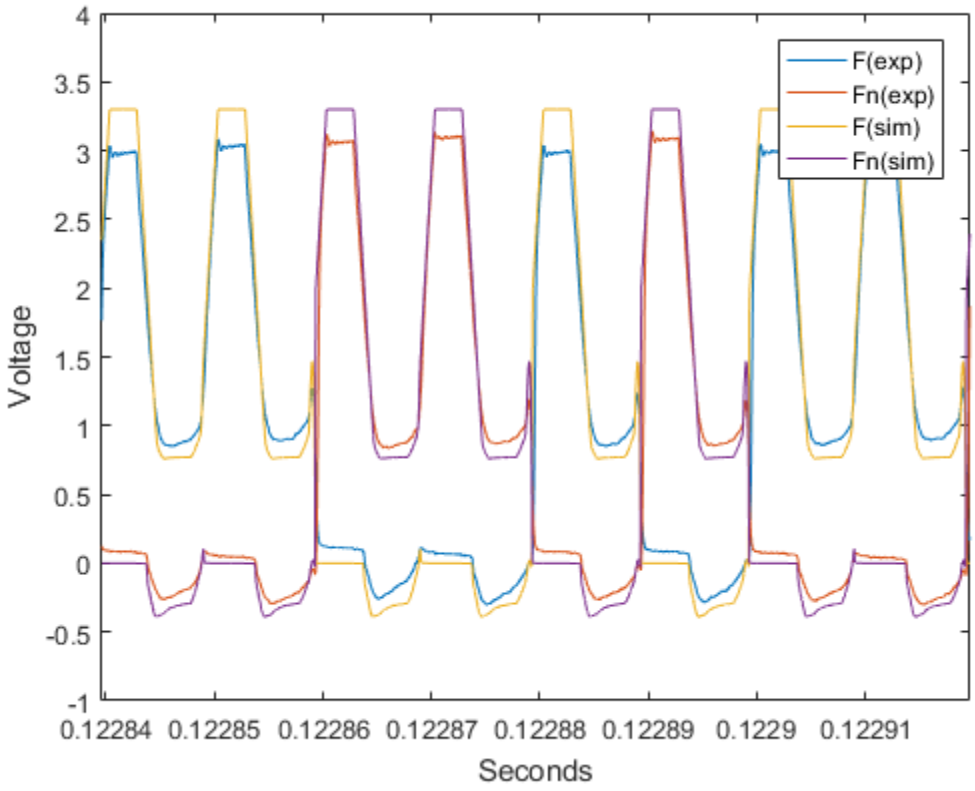


*Figure 47: ALD1106 IDPAL with Discharge 100kHz Results*

The peaks in Figure 48 are within 0.3 volts of each other. The troughs are also not exact. The increase and decrease of voltages seem to be consistent between simulation and experimental results. When the voltage approaches 0, there are inconsistencies between simulation and experimental models.

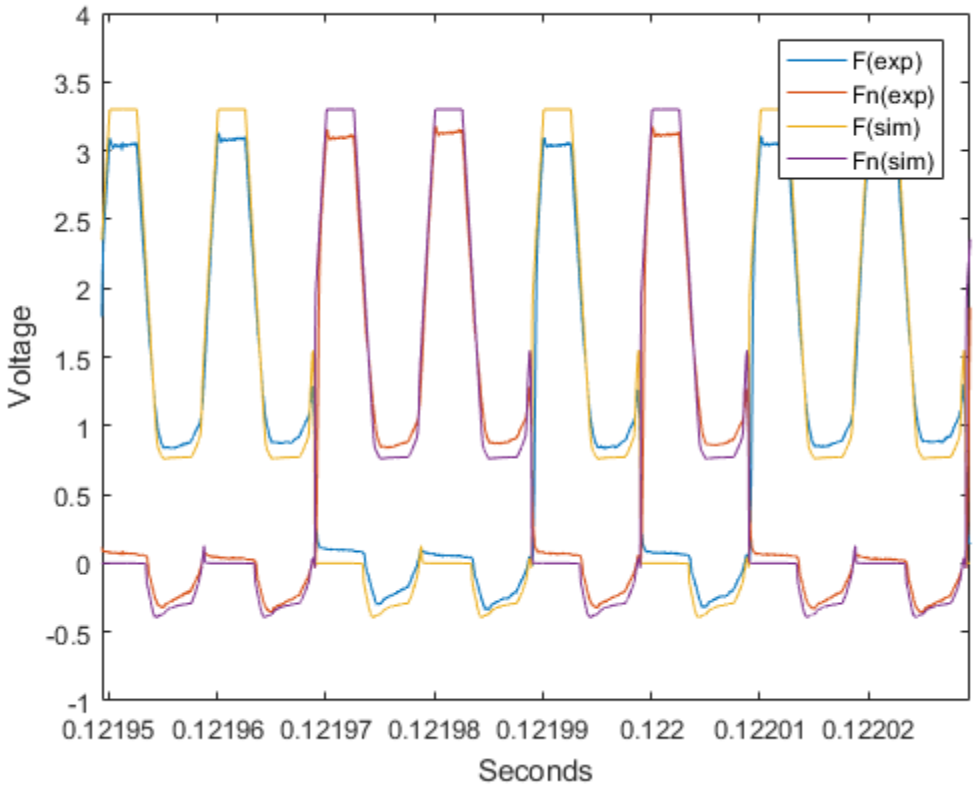


*Figure 48: ALD110804 IDPAL with Discharge 100kHz Results*

The peaks of the waves in Figure 49 are within 0.3 volts between the simulation and experimental models. These models also show variation at the trough of the waves. As the voltages approach 0, there is also some variation between simulation and experimental models. The models are, however, consistent with increasing and decreasing voltages.

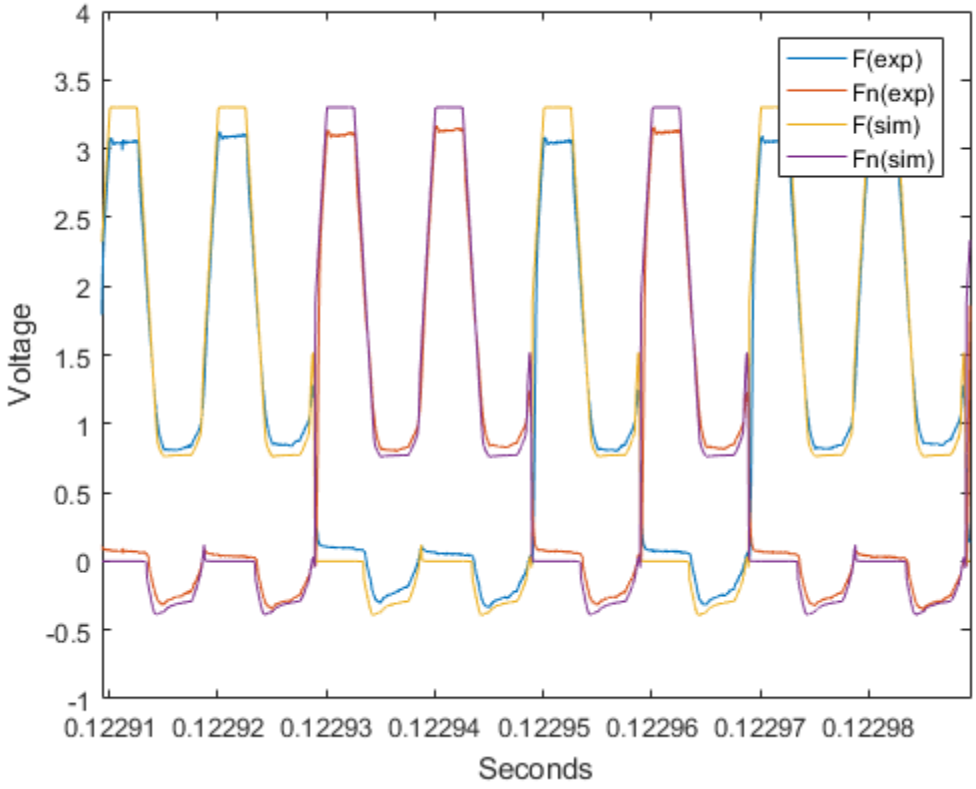


*Figure 49: ALD110802 IDPAL with Discharge 100kHz Results*

As voltages increase and decrease, the simulation model and experimental model seem to be similar. The peak and the trough of the two models are not perfect, but within a 0.3-volt range. There are inconsistencies as the voltages approach 0. Experimental models show less of a range than simulation models, but not by much. Overall, this model shows accurate results.

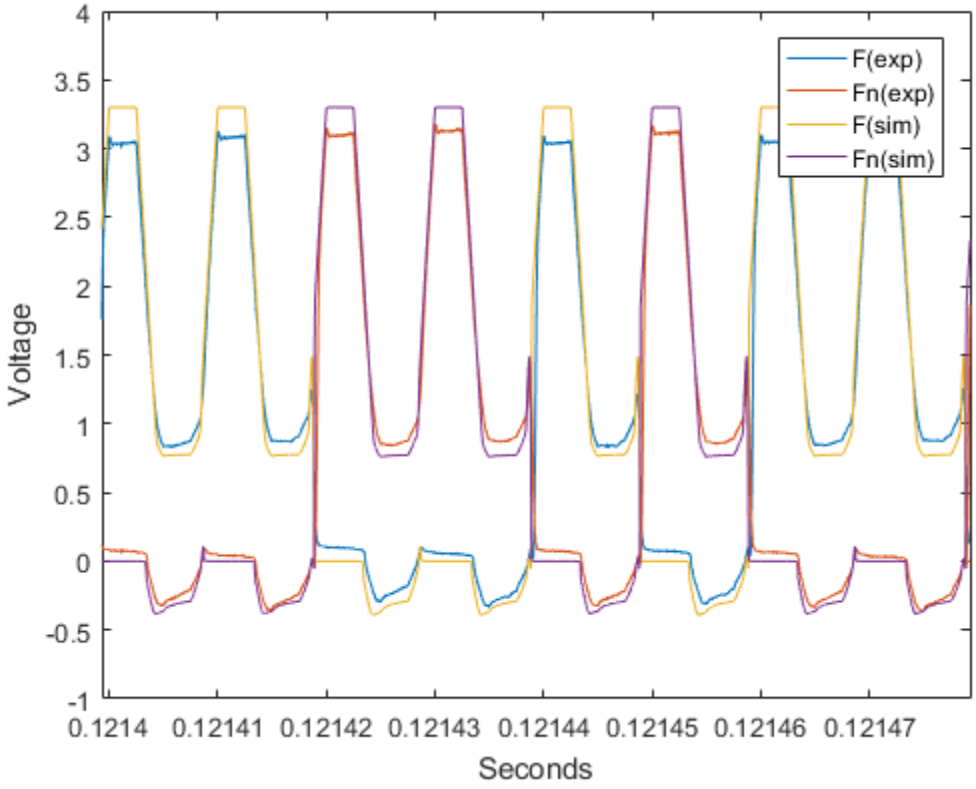


*Figure 50: ALD110800 IDPAL with Discharge 100kHz Results*

### 5.6 Current Measurements

The average current for each power clock phase is expected to increase with the number of gates connected to it. The number of logic gates for each power clock phase is as follows, PC1 has four, PC2 has two, PC3 has one, and PC4 has zero. The simulation for 2n2p implemented with ALD1106 running at a frequency of 10kHz is shown in Figure 51. This waveform shows peak currents of roughly 20uA with each successive phase transferring less current.

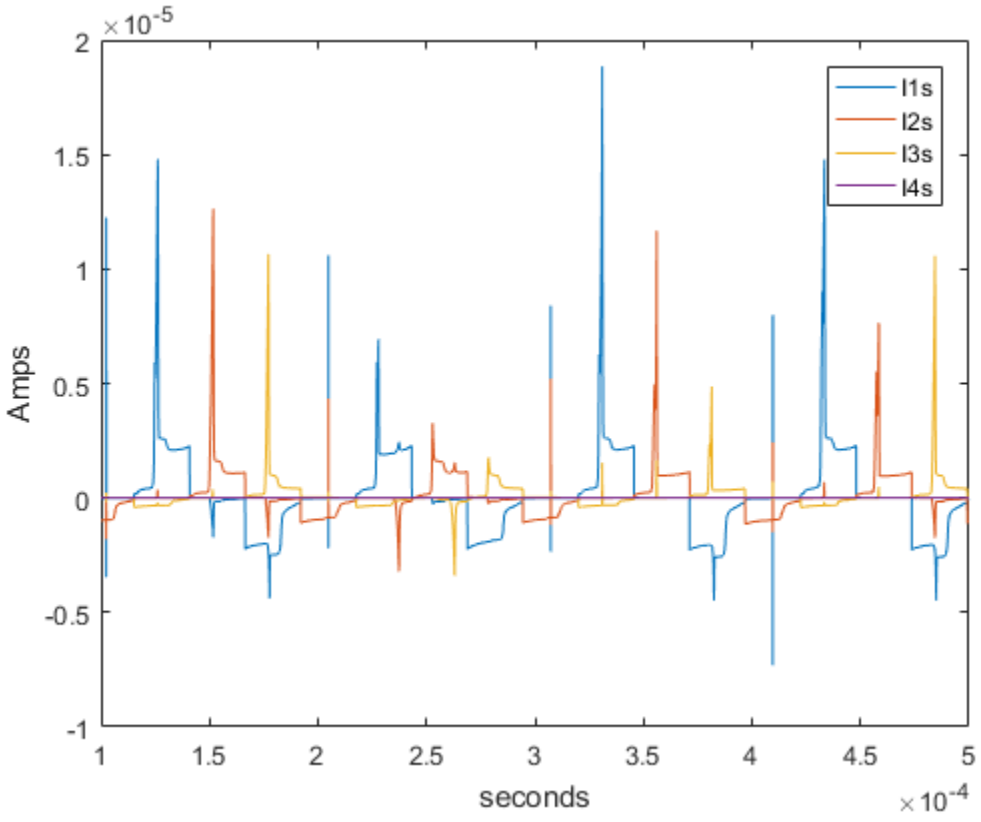


*Figure 51: ALD1106 2N2P 10kHz (simulated)*

The uCP120 analog current probe was used to make current measurements for each phase of the power clock. The uCP120 was configured to wide range mode and zoom out. This configuration provided a voltage to current ratio of 0.1 V/A. The vertical scale of the oscilloscope was manually set to fill the vertical dimension without clipping the input. Each power clock current was measured at different times using a rising edge trigger. Because the currents do not have a known relationship in time, the plots of the experimental current are overlapping to show the variation between them.

Unfortunately, the measurements taken with the uCP120 in this configuration, did not produce data with enough detail for analysis. Figure 52 is the current measurements from each of

the power clocks. It can be easily seen that there is no significant difference in the peak current of roughly 7mA. The results were two orders of magnitude off from the simulated current waveforms.

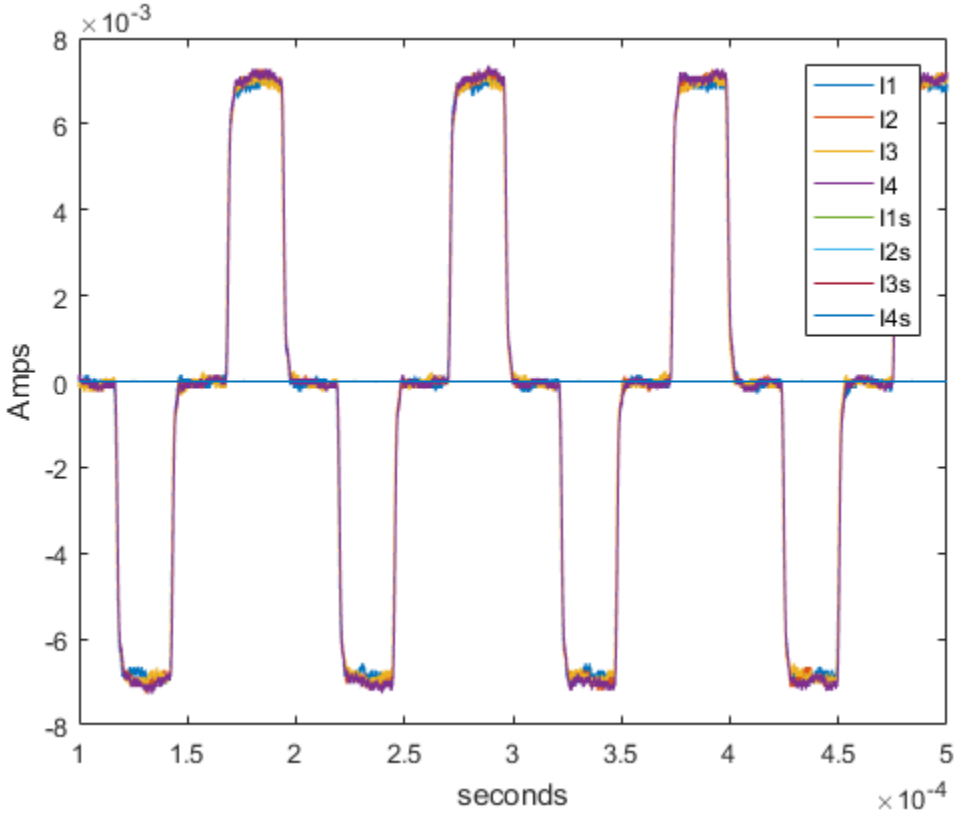


*Figure 52: ALD1106 2N2P 10kHz Results*

The simulated data shows an average current per power clock phase distribution shown in Figure 53. Phase four is unconnected and should be 0 amps. This shows that the decreasing current per phase is not directly related to the number of gates, but rather, is dependent on the previous inputs. The standard deviation for the simulated power clock currents is displayed in Figure 54.

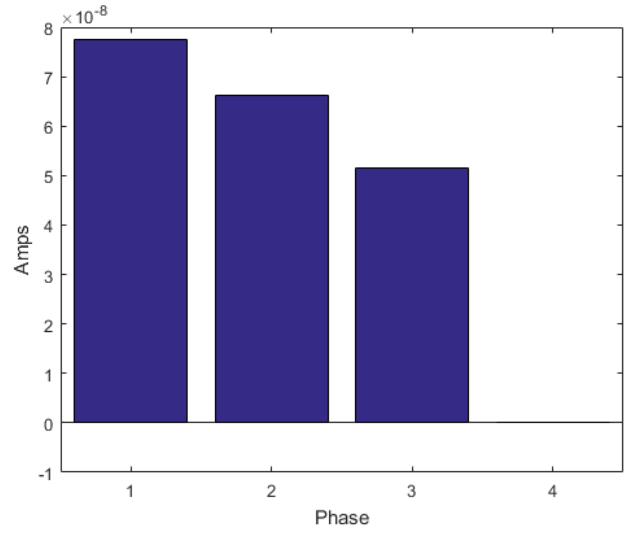


*Figure 53: ALD1106 2N2P 10kHz Simulation Average Current Per Phase*

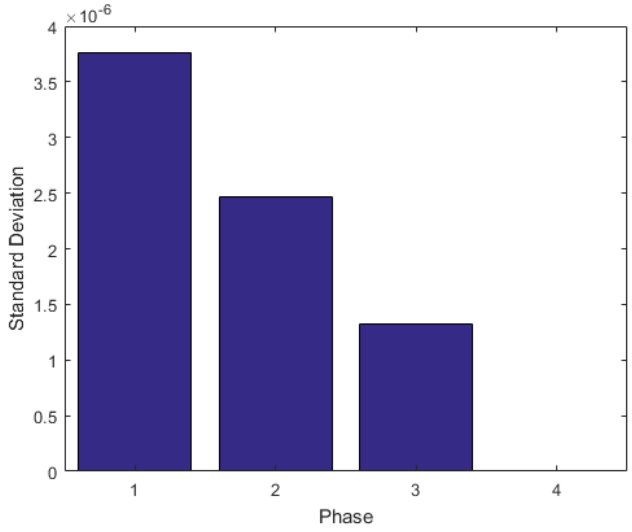


*Figure 54: ALD1106 2N2P 10kHz Simulation Standard Deviation Per Phase*

The average current per phase of the experiment data does not decrease with each phase. Although the data suggests that the circuit returns more current to the power source, PC4 is

unconnected and should be 0 Amps like Figure 53. The standard deviation for the simulated power clock currents is displayed in Figure 56. The consistency in the standard deviation per phase further indicates no significant difference between current phases.

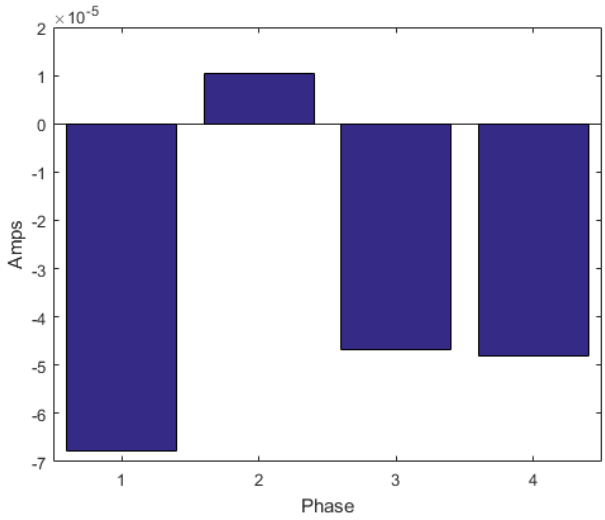


*Figure 55: ALD1106 2N2P 10kHz Experiment Average Current Per Phase*

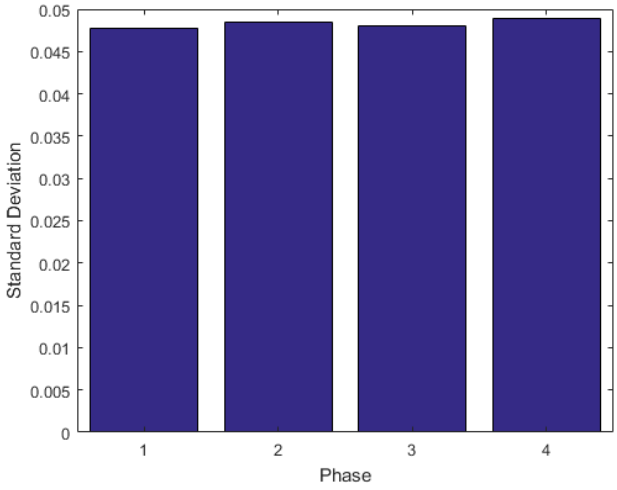


*Figure 56: ALD1106 2N2P 10kHz Experiment Standard Deviation Per Phase*

The 100kHz simulation currents were similar to the ones previously seen. The peak currents reached 80uA and each successive phase transferred less current.

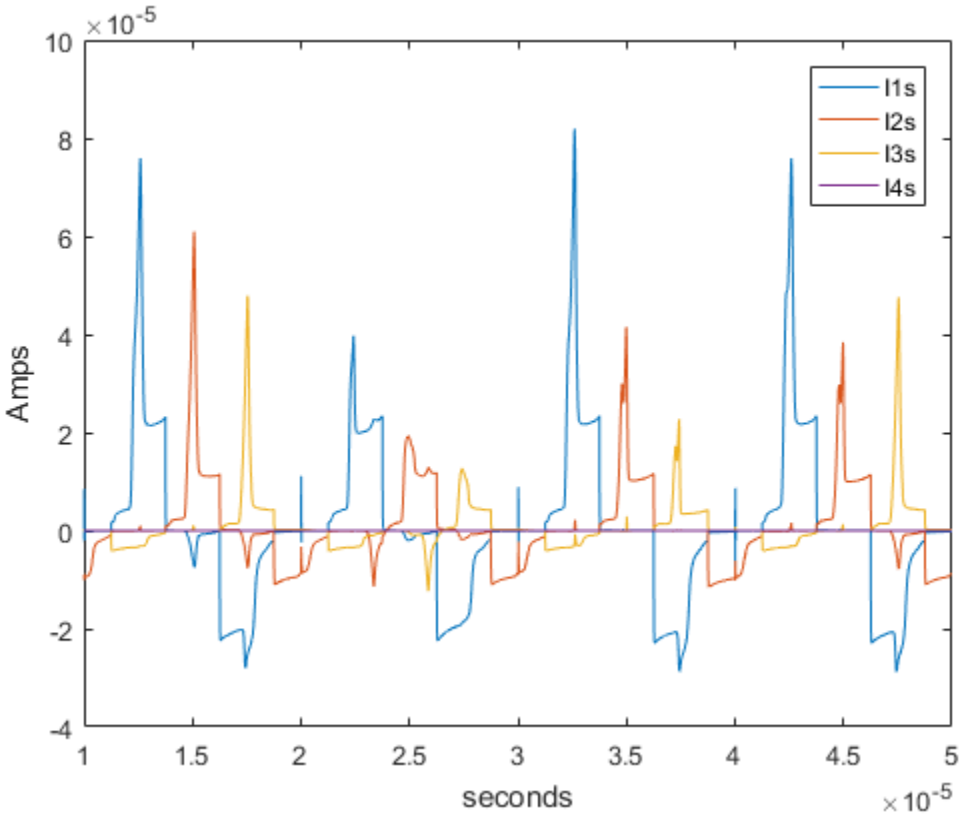


*Figure 57: ALD1106 2N2P 100kHz Current Per Phase (simulated)*

The simulated 100kHz averaged current per power clock phase has distribution shown in Figure 58. Again, phase four is unconnected and should be 0 amps. The standard deviation for the simulated power clock currents is displayed in Figure 59. The plots are roughly an order of magnitude greater than the 10kHz simulated data, which is to be expected.

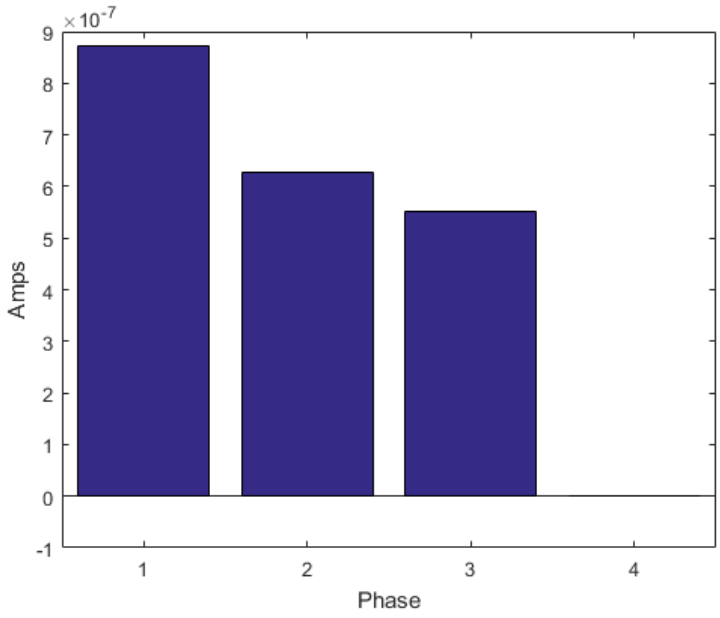


*Figure 58: ALD1106 2N2p 100kHz Simulation Average Current Per Phase*

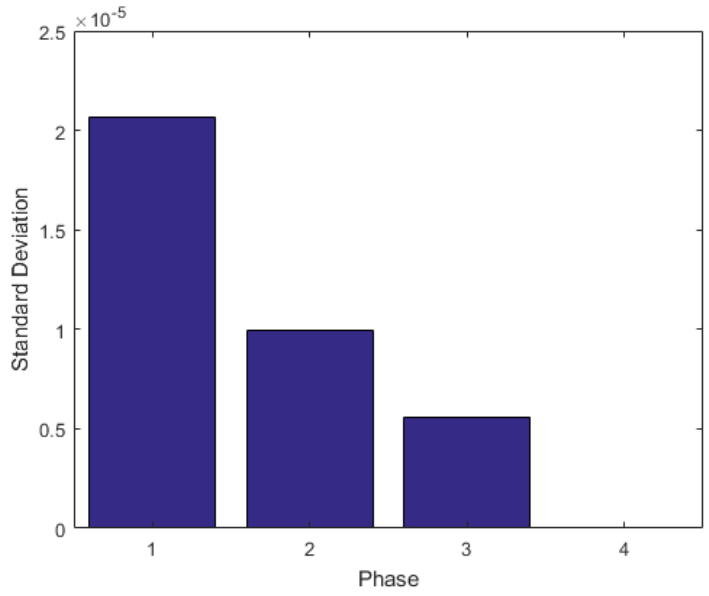


*Figure 59: ALD1106 2N2P 100kHz Simulation Standard Deviation Per Phase*

The uCP120 configuration remained the same while running the circuits with a power clock frequency of 100kHz. Again, the uCP120 did not produce accurate data. Figure 60 shows the peak current of the experiment to be roughly 40mA for each phase. The separate phases are aligned to show the slight variations in peak currents. The measured experiment currents were two orders of magnitude greater than the simulated ones.

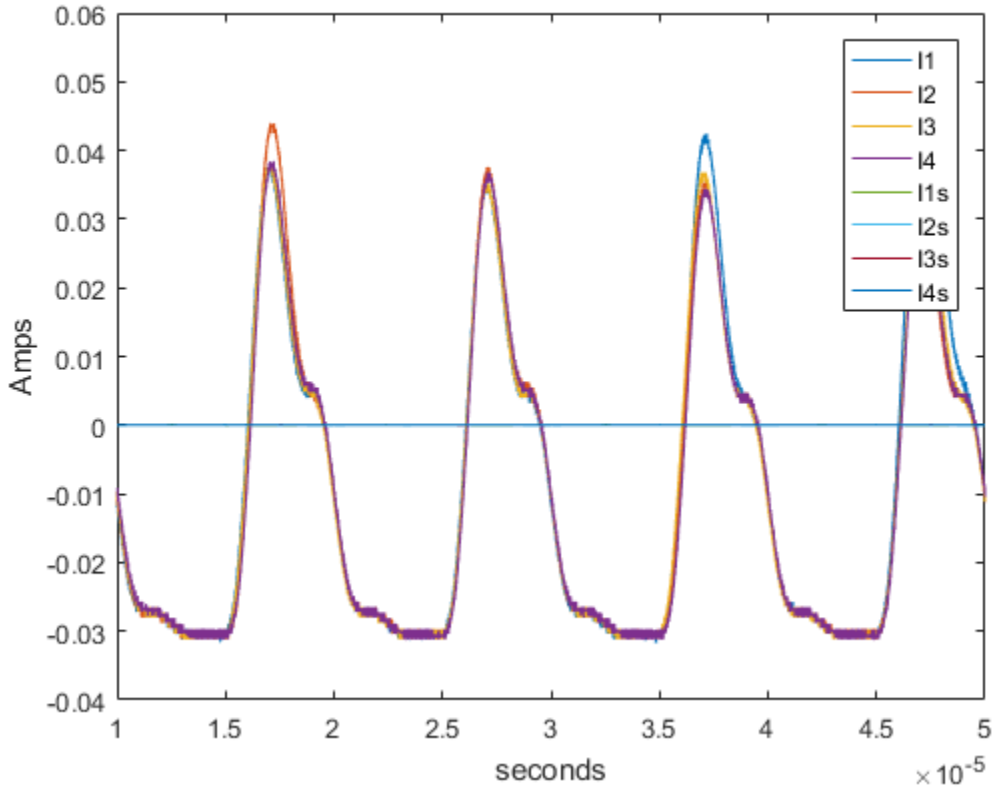


*Figure 60: ALD1106 2N2P 100kHz Experiment Current Per Phase*

The experiment average current for each phase of the power clock, shown in Figure 61, was negative value. Again, it appears the current measurements do not represent the actual

currents of the experiment. This belief is reinforced by the relatively constant standard deviation plot in Figure 62.

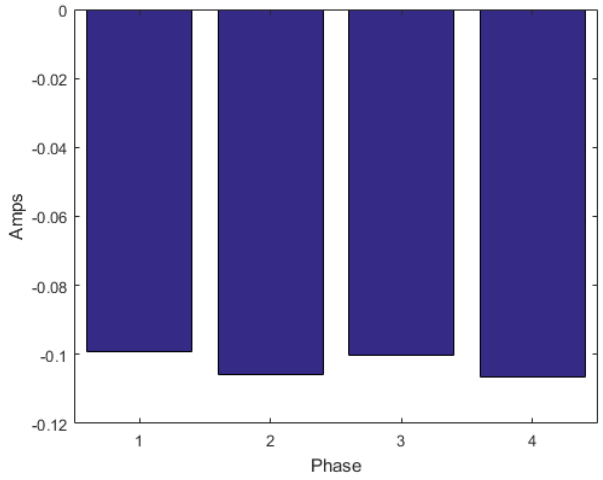


*Figure 61: ALD1106 2N2P 100kHz Experiment Average Current*

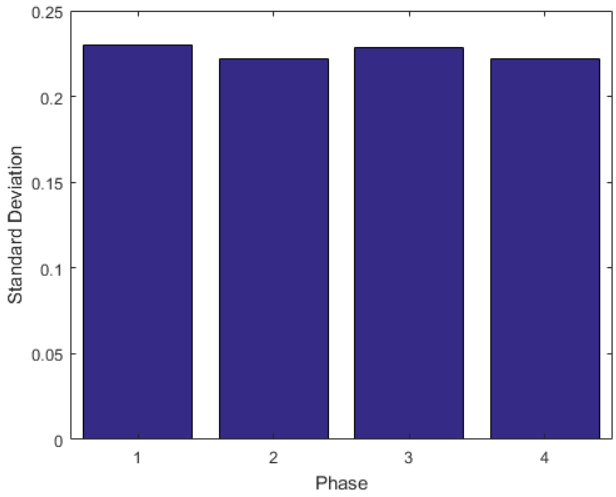


*Figure 62: ALD1106 2N2P 100kHz Experiment Standard Deviation Per Phase*

# 6. DISCUSSION

## 6.1 Results

The adiabatic gate simulation models preformed very similar to the physical models except for IDPAL with a power clock of 10kHz. The 10kHz IDPAL simulation showed that the circuit was malfunctioning while the results of the experiment show the circuit operating correctly. Since IDPAL with discharge worked in both simulation and in the experiment, the difference in behavior for IDPAL is most likely attributed to charge leaking off faster during the experiment than in the simulation models. This could be due to the parameters for the simulation models or insufficient isolation in the experiment. In either case, IDPAL fails to function as an eight-input exclusive-OR when operated with a 100kHz power clock. IDPAL with discharge and ECRL function correctly at both 10kHz and 100kHz. The EPAD MOSFETs appeared to improve upon IDPAL, but the circuit still malfunctioned. Although IDPAL with discharge corrects for this misbehavior, it does so in a non-adiabatic way which will require additional power clocks to efficiently operate. The EPAD MOSFETs showed overall improved performance when compared to the traditional ALD1106 N-Channel MOSFETs.

## 6.2 Future Work

The power clock current of the devices needs to be measured again and compared against the simulation models. The experiment current will provide insight into the actual efficiencies and resistances to power analysis attacks. If sufficient evidence shows a high correlation between simulation and experimentation, then IDPAL with discharge could be a viable option for low power and low frequency designs. A study has yet to be done for power analysis attacks on IDPAL with discharge, but the resistance is expected to be similar to IDPAL given the configuration. Future work should include designing complex circuits with IDPAL with discharge that require more than three layers.

# 7. CONCLUSIONS

The experimental adiabatic 8 input XOR function performed like the simulation version to a reasonable degree. Although the features appeared to be similar, there was a consistent difference between the simulated and experimental peak voltage. The accuracy of the simulation model is shown to decrease as the frequency increases. The 2N-2P models functioned correctly at both 10kHz and 100kHz. There became a noticeable difference in the experimental and simulation results with the EPAD MOSFETs. The 2N-2P model utilizing the ALD110800 MOSFETs displayed a better charge recovery in simulation than shown in the experimental results. The experimental IDPAL results with the ALD1106 MOSFETs operated correctly while the simulation clearly showed incorrect behavior. Although IDPAL with EPAD MOSFETs produced the correct waveform at 10kHz, the ALD1106 produced more favorable results. IDPAL running at a PC frequency of 100kHz failed in both experimental and simulated results across the different threshold voltages. The ALD110800 0-volt threshold MOSFET came closest to producing a correct waveform at 100kHz. IDPAL with discharge proved to be the best solution to increase the operating frequency of IDPAL. Both the experimental and simulated results show IDPAL with discharge working at 10kHz and 100Khz for any given MOSFET. Varying the threshold voltage of the EPAD MOSFETs produced no significant difference in the waveform outputs.

# REFERENCES

[1] Mallela, V. Sarma, V. Ilankumaran and N. S. Rao, *Trends in cardiac pacemaker batteries.*, Indian pacing and electrophysiology journal 4.4, 2004.

[2] Kaul, Himanshu and e. al, *Near-threshold voltage (NTV) design: opportunities and challenges*, Proceedings of the 49th Annual Design Automation Conference, 2012.

[3] M. P. Frank, *Common Mistakes in Adiabatic Logic Design and How to Avoid Them*, Embedded Systems and Applications 216, 2003.

[4] R. T. Hinman and M. F. Schlecht, *Power dissipation measurements on recovered energy logic*, IEEE Symp. on VLSI Circuits Dig. of Tech. Papers, 1994.

[5] M. M. S. Arsalan, *Comparative analysis of adiabatic logic styles,* Multitopic Conference, 2004. Proceedings of INMIC 2004. 8th International. IEEE, 2004.

[6] Moon, Young and J. Deog-Kyoon, *An efficient charge recovery logic circuit.*, IEEE jornal of solid-state circuits 31.4 (1996): 514-522, 1996.

[7] M. T. Cutitaru, *IDPAL - A Partailly-Adiabatic Energy-Efficient Logic Family,* Old Dominion University, 2014.

[8] Chandrakasan, A. P., S. Sheng and R. W. Brodersen, *Low-power CMOS digital design,* IEICE Transactions on Electronics 75.4, 1992.

[9] F. Semiconductor, *CMOS, the Ideal Logic Family,* Available: https://www.fairchildsemi.com/application-notes/AN/AN-77.pdf, 1983.

[10] G. E. Moore, *Cramming more components onto integrated circuits,* Proceedings of the IEEE 86.1, 1998.

[11] Mahmoodi-Meimand, Hamid and A. Afzali-Kusha, *Efficient power clock generation for adiabatic logic*, Circuits and Systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on. Vol. 4, 2001.

[12] Choi, Byong-Deok and e. al, *Symmetric adiabatic logic circuits against differential power analysis,* ETRI journal 32.1, 2010.

[13] M. McAllister, *IDPAL - Input Decoupled Partially Adiabatic Logic Family: Theory and Implementation of Side-Channel Attack Resistant Circuits,* Old Dominon University, 2014.

[14] V. I. Starosel'skii, *Adiabatic logic circuits. A review.,* Russian Microelectronics 31.1, 2001.

[15] U. o. California, *Adabatic Logic,* Available: http://nanocad.ee.ucla.edu/pub/Main/SnippetTutorial/AdiabaticLogic.pdf.

# APPENDICES

Below is the VHDL used to generate the trapezoidal power clock.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

entity dac_mod is
generic
(
    g_bit_depth     : integer := 10;
    g_cycle_bits    : integer := 10;
    g_init_state    : std_logic_vector := "00"
);
port
(
    clock           : in  std_logic;
    cycle_count     : in  std_logic_vector(g_cycle_bits-1 downto 0);
    step_size       : in  std_logic_vector(g_bit_depth-1 downto 0);
    update          : out std_logic;
    state           : out std_logic_vector(1 downto 0);
    dac_clock       : out std_logic;
    dac_value       : out std_logic_vector(g_bit_depth-1 downto 0)
);
end entity;

architecture arch of dac_mod is

    --Frequency Caclulation
    --T = 4 * ceil(1024 / step) * (2*(cycles+1) / sclk)
    --T = 8 * ceil(1024 / step) * ((cycles+1) / sclk)
    --f = 1 / T
    --f = sclk / (8 * ceil(1024 / step) * (cycles+1))
```

```
--100us / 10kHz : sclk = 50Mhz
--dclk=2000000.000000 : step=41 : cycles=24 : T=0.000100 : f=10000.000000 : err=0.000000

--10us / 100kHz : sclk = 50Mhz
--dclk=25000000.000000 : step=34 : cycles=1 : T=0.000010 : f=100806.451613 : err=806.451613
--dclk=16666666.666667 : step=49 : cycles=2 : T=0.000010 : f=99206.349206 : err=-793.650794

--100us / 10kHz : sclk = 100Mhz
--dclk=2000000.000000 : step=41 : cycles=49 : T=0.000100 : f=10000.000000 : err=0.000000

--10us / 100kHz : sclk = 100Mhz
--dclk=20000000.000000 : step=41 : cycles=4 : T=0.000010 : f=100000.000000 : err=-0.000000



-- DAC State Machine
constant c_rise_state   : std_logic_vector := "00";
constant c_max_state    : std_logic_vector := "01";
constant c_fall_state   : std_logic_vector := "10";
constant c_min_state    : std_logic_vector := "11";

--Cycle Counter - Increments between 0 and cycle_count at a rate of clock
signal a_cycle_counter  : std_logic_vector(g_cycle_bits-1 downto 0) := (others => '0');
signal q_cycle_counter  : std_logic_vector(g_cycle_bits-1 downto 0) := (others => '0');

--Cycle Update -
signal a_cycle_update   : std_logic                                 := '0';

--Next value - Increases or decreases by step_size at a rate of cycle_count
signal a_next_value     : std_logic_vector(g_bit_depth-1 downto 0)  := (others => '0');

--Value Limit -
signal a_value_limit    : std_logic_vector(g_bit_depth-1 downto 0)  := (others => '0');

--State Count -
signal a_state_count    : std_logic_vector(g_bit_depth-1 downto 0)  := (others => '0');
signal q_state_count    : std_logic_vector(g_bit_depth-1 downto 0)  := (others => '0');

--State Update -
```

```
    signal a_state_update   : std_logic                         := '0';

    --DAC value - Increases or decreases by step_size at a rate of cycle_count
    signal a_dac_value      : std_logic_vector(g_bit_depth-1 downto 0)  := (others => '0');
    signal q_dac_value      : std_logic_vector(g_bit_depth-1 downto 0)  := (others => '0');

    --DAC Clock -
    signal q_dac_clock      : std_logic                         := '0';


    --DAC State - Changes when dac_value is 0 or 2^g_bit_depth-1
    signal a_state          : std_logic_vector(1 downto 0)          := (others => '0');
    signal q_state          : std_logic_vector(1 downto 0)          := g_init_state;

begin

    --  ____________________________________________
    -- |                                    |
    -- |               Outputs              |
    -- |____________________________________________|
    update    <= a_cycle_update;
    dac_clock <= q_dac_clock;
    dac_value <= q_dac_value;
    state     <= q_state;


    --  ____________________________________________
    -- |                                    |
    -- |             Cycle Counter          |
    -- |____________________________________________|
    a_cycle_counter_ps : process(q_cycle_counter, a_cycle_update)
    begin
        if (a_cycle_update = '0') then
            a_cycle_counter <= std_logic_vector(unsigned(q_cycle_counter) + 1);
        else
            a_cycle_counter <= (others => '0');
        end if;
    end process;
```

```
q_cycle_counter_ps : process(clock)
begin
   if rising_edge(clock) then
      q_cycle_counter <= a_cycle_counter;
   end if;
end process;

--  ____________________________________________
-- |                                            |
-- |              Cycle Update                  |
-- |____________________________________________|
a_cycle_update_ps : process(q_cycle_counter, cycle_count)
begin
   if (q_cycle_counter = cycle_count) then
      a_cycle_update <= '1';
   else
      a_cycle_update <= '0';
   end if;
end process;

--  ____________________________________________
-- |                                            |
-- |               DAC State                    |
-- |____________________________________________|
a_state_ps : process(q_cycle_counter, q_state, a_state_update)
begin
   if (a_state_update = '1') then
      case q_state is
         when c_rise_state   => a_state <= c_max_state;
         when c_max_state    => a_state <= c_fall_state;
         when c_fall_state   => a_state <= c_min_state;
         when c_min_state    => a_state <= c_rise_state;
         when others         => a_state <= c_rise_state; --Never
      end case;
   else
      a_state <= q_state;
   end if;
end process;
```

```
q_state_ps : process(clock)
begin
    if rising_edge(clock) then
        q_state <= a_state;
    end if;
end process;


--  ____________________________________________
-- |                                            |
-- |               State Count                  |
-- |____________________________________________|
a_state_count_ps : process(a_state_update, q_state_count, step_size, a_cycle_update, q_dac_clock)
begin
    if (a_cycle_update = '1') and (q_dac_clock = '1') then
        a_state_count <= std_logic_vector(unsigned(q_state_count) + unsigned(step_size));
    else
        a_state_count <= q_state_count;
    end if;
end process;
q_state_count_ps : process(clock)
begin
    if rising_edge(clock) then
        if (a_state_update = '1') then
            q_state_count <= (others => '0');
        else
            q_state_count <= a_state_count;
        end if;
    end if;
end process;


--  ____________________________________________
-- |                                            |
-- |               Value Update                 |
-- |____________________________________________|
a_value_update_ps : process(q_state_count, a_state_count)
begin
    if (q_state_count > a_state_count) then
        a_state_update <= '1';
```

```
    else
        a_state_update <= '0';
    end if;
end process;

--  ____________________________________________
-- |                                            |
-- |                Next Value                  |
-- |____________________________________________|
a_next_value_ps : process(q_state, q_dac_value, step_size)
begin
    case q_state is
        when c_rise_state   => a_next_value <= std_logic_vector(unsigned(q_dac_value) + unsigned(step_size));
        when c_max_state    => a_next_value <= (others => '1');
        when c_fall_state   => a_next_value <= std_logic_vector(unsigned(q_dac_value) - unsigned(step_size));
        when c_min_state    => a_next_value <= (others => '0');
        when others         => a_next_value <= (others => '0'); --Never
    end case;
end process;

--  ____________________________________________
-- |                                            |
-- |                Value Limit                 |
-- |____________________________________________|
a_value_limit_ps : process(q_state, a_next_value, q_dac_value)
begin
    if    (q_state = c_rise_state) then
        if (a_next_value > q_dac_value)  then
            a_value_limit <= a_next_value;
        else
            a_value_limit <= (others => '1');
        end if;
    elsif (q_state = c_max_state)   then
        a_value_limit <= (others => '1');
    elsif (q_state = c_fall_state)  then
        if (a_next_value < q_dac_value)  then
            a_value_limit <= a_next_value;
        else
```

```
            a_value_limit <= (others => '0');
        end if;
    elsif (q_state = c_min_state)   then
        a_value_limit <= (others => '0');
                            else
                                a_value_limit <= (others => '0');
    end if;
end process;



--  ____________________________________________
-- |                                            |
-- |                DAC Clock                   |
-- |____________________________________________|
q_dac_clock_ps : process(clock)
begin
    if rising_edge(clock) then
        if (a_cycle_update = '1') then
            q_dac_clock <= not q_dac_clock;
        end if;
    end if;
end process;



--  ____________________________________________
-- |                                            |
-- |                DAC Value                   |
-- |____________________________________________|
a_dac_value_ps : process(a_cycle_update, a_value_limit, q_dac_value, q_dac_clock)
begin
    if (a_cycle_update = '1') and (q_dac_clock = '1') then
        a_dac_value <= a_value_limit;
    else
        a_dac_value <= q_dac_value;
    end if;
end process;
q_dac_value_ps : process(clock)
begin
```

```
        if rising_edge(clock) then
            q_dac_value <= a_dac_value;
        end if;
    end process;


end architecture;
```

# VITA

Kevin A. Johnson

Department of Electrical and Computer Engineering

Old Dominion University

Norfolk, VA 23529

## EDUCATION

- B.S. Electrical Engineering, Old Dominion University, Norfolk, VA, May 2016.

## AWARDS

- Dean's List, Frank Batten College of Engineering and Technology, Old Dominion University, Norfolk, VA, Fall 2012, Spring 2013, Summer 2013, Fall 2013, Spring 2014, Spring 2015, Fall 2015

## EXPERIENCE

- Digital circuit design for FPGAs
- Mixed signal simulations with LTspice

## COMPUTER PROFICIENCY

- C/C++, MATLAB, VHDL