UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Development of a basic but full gate library, using AQFP cells, for the development of complex multi-gates.

J.S. Ketteringham
24009385

Report submitted in partial fulfilment of the requirements of the module Project (E) 448 for the degree Baccalaureus in Engineering in the Department of Electrical and Electronic Engineering at Stellenbosch University.

Supervisor: Prof C. J. Fourie

July 2023

Acknowledgements

I would like to thank my dog, Muffin. I also would like to thank the inventor of the incubator; without him/her, I would not be here. Finally, I would like to thank Dr Herman Kamper for this amazing report template.

UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Plagiaatverklaring / *Plagiarism Declaration*

1. Plagiaat is die oorneem en gebruik van die idees, materiaal en ander intellektuele eiendom van ander persone asof dit jou eie werk is.
   Plagiarism is the use of ideas, material and other intellectual property of another's work and to present is as my own.

2. Ek erken dat die pleeg van plagiaat 'n strafbare oortreding is aangesien dit 'n vorm van diefstal is.
   I agree that plagiarism is a punishable offence because it constitutes theft.

3. Ek verstaan ook dat direkte vertalings plagiaat is.
   I also understand that direct translations are plagiarism.

4. Dienooreenkomstig is alle aanhalings en bydraes vanuit enige bron (ingesluit die internet) volledig verwys (erken). Ek erken dat die woordelikse aanhaal van teks sonder aanhalingstekens (selfs al word die bron volledig erken) plagiaat is.
   Accordingly all quotations and contributions from any source whatsoever (including the internet) have been cited fully. I understand that the reproduction of text without quotation marks (even when the source is cited) is plagiarism

5. Ek verklaar dat die werk in hierdie skryfstuk vervat, behalwe waar anders aangedui, my eie oorspronklike werk is en dat ek dit nie vantevore in die geheel of gedeeltelik ingehandig het vir bepunting in hierdie module/werkstuk of 'n ander module/werkstuk nie.
   I declare that the work contained in this assignment, except where otherwise stated, is my original work and that I have not previously (in its entirety or in part) submitted it for grading in this module/assignment or another module/assignment.

Studentenommer / *Student number*: 24009385	Handtekening / *Signature*
Voorletters en van / *Initials and surname* J.S. Ketteringham	Datum / *Date* 24/07/2023

Abstract

English

The Adiabatic Quantum Flux Parametron (AQFP) is a new and exciting technology in superconducting digital circuits with a strong promise towards advancing the field of quantum computing and low-power, high-performance electronics. These AQFP circuits exhibit unique superconducting properties, such as zero electrical resistance and make use of Josephson junctions to create a robust platform for qubit-based computation. The absence of dissipation losses in superconducting circuits leads to reduced power consumption, making AQFP circuits highly energy-efficient. Although potential applications for AQFP technology extend beyond quantum computing, the implementation in this paper is restricted to developing basic logic gates from superconducting AQFP circuits to create quantum-computing circuits.

Afrikaans

Die Adiabatic Quantum Flux Parametron (AQFP) is ’n nuwe en opwindende tegnologie in supergeleidende digitale stroombane met ’n sterk belofte om die veld van kwantumrekenaars en laekrag, hoëprestasie-elektronika te bevorder. Hierdie AQFP-stroombane vertoon unieke supergeleidende eienskappe, soos geen elektriese weerstand, en maak gebruik van Josephson-aansluitings om ’n robuuste platform vir qubit-gebaseerde berekening te skep. Die afwesigheid van dissipasieverliese in supergeleidende stroombane lei tot verminderde kragverbruik, wat AQFP-stroombane hoogs energiedoeltreffend maak. Alhoewel potensiële toepassings vir AQFP-tegnologie verder strek as kwantumrekenaars, is die implementering in hierdie vraestel beperk tot die ontwikkeling van basiese logiese hekke van supergeleidende AQFP-stroombane om kwantumrekenaarstroombane te skep.

Contents

List of Figures

List of Tables

Nomenclature

Variables and functions

$p(x)$	Probability density function with respect to variable x.
$P(A)$	Probability of event A occurring.
ε	The Bayes error.

Acronyms and abbreviations

AQFP	Adiabatic Quantum-Flux-Parametron
CMOS	Complementary metal-oxide-semiconductor
EDP	Energy delay product
JJ	Josephson Junction
PDP	Power delay product
Qubit	Quantum Bit
RSFQ	Rapid single flux quantum
SQuID	Superconducting quantum interference device

Chapter 1

Introduction

1.1. Problem Statement

The increasing amount of new technologies such as Information of Things(IoT), Information and Communication Technologies(ICT), and Artificial Intelligence(AI) are leading to drastic increases in global power consumption. Thus the future depends heavily on finding new ways to produce large amounts of renewable energy or adapting current electronics to become more energy efficient. In this report the focus is on a specific type of superconducting logic element known as the AQFP(Adiabatic Quantum-flux-parametron).

The AQFP is an energy-efficient superconductor logic element that makes use of adiabatic switching to maximize energy efficiency to the point where AQFP circuits can operate with energy dissipation near the thermodynamic and quantum limits [1]. No element can work at 100% efficiency so the term thermodynamic limit describes the best possible efficiency a device can have according to the laws of thermodynamics and the quantum limit refers to the smallest possible amount of energy that can be used or lost. When comparing the AQFP to its current CMOS counterpart, it becomes evident that the AQFP can function using between 100 to 10,000 times less power than CMOS components. [2]. This includes the power used to manufacture, run and cool both components. This project aims to develop a basic but full gate library using AQFP circuits and use these basic logic gates to develop more complex multi-gates, namely the Full Adder and the Half Adder. The use of these AQFP circuits will allow computing devices to run at high clock speeds up to $5GHz$ but at a fraction of the power.

1.2. Objective

By achieving the following objectives I will have answered the problem statement.

- An understanding in the Josephson Junction and how it forms part of the AQFP circuit.
- How AQFP's can be used to create basic logic gates.
- Using JoSIM to create and simulate working logic gates.
- Compiling these logic gates to make complex multi-gates.
- Using Klayout to design the physical AQFP cells.
- Using InductEx to extract the inductance's from the KLayout model.
- Rerunning JoSim to ensure the values obtained from the physical model are correct.

1.3. Summary of Work

Initially, a netlist of fundamental AQFP logic components is constructed. The inductance values for these components are adapted from a previous research project conducted by MIT [3]. Subsequently, JoSim is used to simulate and verify the functionality of these components and to ensure they can be effectively combined to construct more complex multi-gates, such as the Half Adder and Full Adder. Once the correctness of individual gates and more complex multi-gates is established, the logic gates are designed in a physical design software called KLayout. Following the design phase, InductEx, a specialized software tool, is utilized to extract real inductance values from the physical models created in KLayout. These newly extracted inductance values are then compared with the initial values specified in the netlist.

JoSim is rerun to confirm that the components continue to operate correctly with these newly extracted inductance values. The physical model is iteratively refined until the extracted values closely align with the initially designed values. Once the extracted values reach an acceptable level of accuracy and the components function as expected, further adjustments are made to optimize the circuit's performance. These optimizations are directed at achieving higher output currents and enhancing overall circuit performance.

1.4. Scope

The work covered in the following project lies within the field of superconducting electronics specifically AQFP circuits. The following report will discuss how AQFP logic can be used to create basic logic gates and how they can be compiled to create complex multi-gates. This project will explain the different types of software that will be used and what each software is used for. Finally the result of the work is shown in the results section that demonstrates the working gates, confirming the project has been a success.

Sections that relate to AQFP circuits but have not been covered include flux trapping moats. AQFP circuits are extremely sensitive to external magnetic flux. Thus to prevent undesired magnetic flux interfering, holes are created in the ground plane of the AQFP circuit to attract and trap this flux. By trapping this magnetic flux in moats that can be distant from important parts of the superconducting circuit, the negative effects of the magnetic flux interference can be avoided [4]. However, it was found that fluxions trapped in moats can induce currents in superconductor circuit loops, which can reduce the output of AQFP circuits. Research has been done to identify the optimal position and size of these moats to ensure the optimal performance of the AQFP circuit is maintained while still providing the superconducting circuit with resistance to external magnetic flux. However, for the following project the AQFP buffer is made with oat positions obtained from the following report [5]. The other circuits are made without moats to simplify the design and make it easier to see

Boundaries are this and that within AQFP e.g the switching frequency is 5GHz as this is what is defined as adiabatic. Why does it differ from RSFQ circuits. What is the outcome that is expected. Constraints including Splitter circuit maybe and why everything needs a buffer.

Include that the constant cell could have been optimized to produce an output that is closer to that of the buffer. Due to this flaw the 3input AND and OR gates are not exactly good. Maybe look at flux trapping and why it was or wasnt covered.

1.5. Roadmap

1.5.1. Chapter 2

Chapter 2 includes the Literature review which covers related work.

1.5.2. Chapter 3

Chapter 3 is the system design, which includes the Josephson junction in the AQFP circuit and the basic design of the AQFP logic gates and multi-gates.

1.5.3. Chapter 4

Chapter 4 is the physical design section, which covers the design of each of the basic logic gates in KLayout software. InductEx is used to extract the inductance's of components based on the size of materials in KLayout.

1.5.4. Chapter 5

Chapter 5 is the result section, which uses JoSim to simulate the AQFP circuits using the inductance values extracted from the models in chapter 4. This section shows the output of all the logic gates and multi-gates and includes why they are successful.

1.5.5. Chapter 6

Chapter 6 is the summary and conclusion.

Chapter 2

Literature

2.1. Terminology

- **Lithography**: Lithography is the process of transferring a mask pattern onto a substrate
- **Critical Current**: The maximum electric current density that a given superconducting material can carry, before switching into the normal state.
- **Quantization**: This term means that a physical quantity can only take on specific discrete values rather than any value within a continuous range.
- **Magnetic Flux**: Magnetic flux is a measure of the quantity of magnetic field passing through a given area or circuit. It is symbolized by ϕ and is measured in units of weber (Wb).
- **Josephson Junctions**: Josephson junctions are devices consisting of two superconducting electrodes separated by a very thin insulating barrier. In these junctions, electrons can tunnel through the insulating barrier without resistance when the junction is cooled to low temperatures temperatures. They are a key component in superconducting electronics.
- **Hamiltonian**: The Hamiltonian of a system specifies the total energy of the system. The sum of its kinetic energy and its potential energy.

2.2. Software

2.2.1. JoSim

The following text is copied directly off the JoeyDelp GitHub page on the overview of JoSim. Can I do this and cite the GitHub page some how or do I need to phrase the following text in my own words

JoSim was the software used to simulate and test the netlists of the various AQFP logic circuits. JoSIM was developed under IARPA contract SuperTools(via the U.S. Army

Research Office grant W911NF-17-1-0120). JoSIM is a SPICE syntax circuit simulator specifically created to handle superconducting elements such as the Josephson junction. It reads in a standard SPICE deck, creates an A matrix and solves the linear algebra problem Ax=b. The linear algebra package KLU by Tim Davis is used to solve the system of equations. A two stage distribution platform is implemented in JoSIM with the first being a command line interface (CLI) binary that is always built as standard during compilation. This is accompanied by a C++ library (libjosim). JoSIM allows output of results in various formats such as comma separated value (CSV) or raw SPICE output. JoSIM has inherent support for .PARAM commands that allow components to have variable values as well as expression parsing. JoSIM implements the RCSJ model of the Josephson junction and only supports transient analysis at present. JoSIM takes a .cir file as input and produces a .dat/.csv file as output.

2.2.2. KLayout

Following segments of this text have also been compied from KLayout documentation, is this allowed? KLayout was the software used to design the layout for the AQFP logic circuits. KLayout is a computer aided design (CAD) software and is used to layout patterns for lithography. The CAD patterns define areas that will be exposed by either a photomask maker or an electron beam writer. Create CAD patterns using software designed specifically for chip design to ensure compatibility with lithography equipment. KLayout is a free and powerful CAD software developed for chip design engineers. It is intuitive, fast, accurate and extensible via an integrated development environment (Ruby and Python).

2.2.3. InductEx

Following segments of this text have also been compied from InductEx documentation, is this allowed? InductEx was the software used to extract the inductance values from the physical design in KLayout allowing one to test and simulate the circuit in JoSim with real values to gain insight on whether the circuit will work or not. InductEx is a three-dimensional integrated circuit parameter extraction and layout verification software with capabilities that range from multi-terminal inductance extraction to capacitance, impedance and S-parameter calculations, magnetic and gradient field analysis, flux trapping analysis, packaging evaluation and more.

2.3. Related work

Superconducting logic is presented by two main branches: digital single-flux-quantum (SFQ) and adiabatic superconducting logic (ASL). These two branches are represented by

the two following sections: RSFQ and AQFP superconducting logic technologies.

2.3.1. Adiabatic quantum flux parametron (AQFP) logic

AQFP logic is a type of superconducting logic technology that operates on the principles of quantum adiabatic computing. AQFP technology operates based on the quantization of magnetic flux in superconducting Josephson junctions. This means that the magnetic flux passing through the AQFP circuit is restricted to discrete, quantized values due to the behavior of the Josephson junctions. The AQFP buffer (the most basic circuit at the center of all AQFP logic) is made from two Josephson junctions often known as jj1 and jj2. A single flux quanta is induced in either of the Josephson junctions and causes a circulating current to flow within the junction. The circulating current in these loops represents the quantum state. If the current circles the left loop the state is a logical ”1” and if the current circles the right loop, the state is a logical ”0”. AQFP's distinguishing feature is its adiabatic nature. AQFP logic relies adiabatic switching to minimize power consumption, where quantum information is processed by carefully controlling the switching of the system's Hamiltonian (the sum of its kinetic and potential energy). This adiabatic nature allows it to function efficiently at very low power consumption near the quantum limit.

2.3.2. Rapid Single-Flux-Quantum (RSFQ) logic

RSFQ logic is another type of superconducting logic similar to that of AQFP. Similarly to AQFP, RSFQ technology also operates based on the quantization of magnetic flux in superconducting Josephson junctions. RSFQ circuits, similar to AQFP circuits, induce a single flux quanta in the Josephson junction. However, in RSFQ technology only one Josephson junction is required and the presence or absence of flux within the junction represents a logical ”1” or ”0”. Whereas AQFP technology makes use of two Josephson junctions and whether the single flux quanta is present in the left or right junction determines whether the output is a logical ”1” or ”0”.

2.3.3. Differences between AQFP and RSFQ technology

Some of the key differences between AQFP and RSFQ technology stem from the different layout, uses and characteristics as follows.

- AQFP superconducting logic is based on adiabatic quantum computing principles. AQFP circuits rely on the controlled flow of magnetic flux quanta through loops to perform logic operations. Whereas, RSFQ is a classical superconducting digital logic technology that operates by using the discrete quantization of magnetic flux in superconducting Josephson junctions. RSFQ technology is not inherently quantum.

- AQFP technology operates at around 5GHz and switches adiabatically, due to this adiabatic characteristic, AQFP logic operates at a power consumption closer the the thermodynamic limit. RSFQ can operate at extremely high clock speeds at around 100+ GHz making it faster than AQFP circuits. However, due to the fact that RSFQ circuits are not adiabatic at theses speeds they use significantly more power than AQFP circuits. In classical computing systems it is still possible for RSFQ circuits to use less power than AQFP circuits.

- AQFP is primarily used for quantum computing and quantum information processing such as qubit control and quantum gate operations. Whereas, RSFQ is primarily used for classical digital computing tasks, such as high speed digital signal processing or analog-to-digital converters.

2.3.4. Similar Projects

MIT Lincoln Laboratory has completed and developed a full list of basic AQFP logic cells [3]. These cells are driven by a 4 phase clock generated by two AC sources and a DC source. Their aim was to design working logic gates using AQFP circuits. To ensure the basic gates worked, the input vs the output of the gates were recorded and if the desired output was received at a sufficient amplitude, the gate was considered a success. Their results proved their gates to be working and reliable. Although this project is similar, their results provide only the values of the inductance's and simple schematics of their AQFP components. These results achieved by MIT are used as a benchmark in the project to follow and are used to develop an entirely unique AQFP layout resulting in unique inductance values.
Two reports, titled "Adiabatic quantum flux parametron cell library adopting minimalist design" and "Adiabatic quantum-flux parametron cell library designed using a 10kA cm2 niobium fabrication process," have been created with the intention of achieving the same set of results. However, these documents are restricted from access and were not used or referenced in this report.

2.4. I needed to know this before i could do this

I'm not sure if there is anything I should put in here?

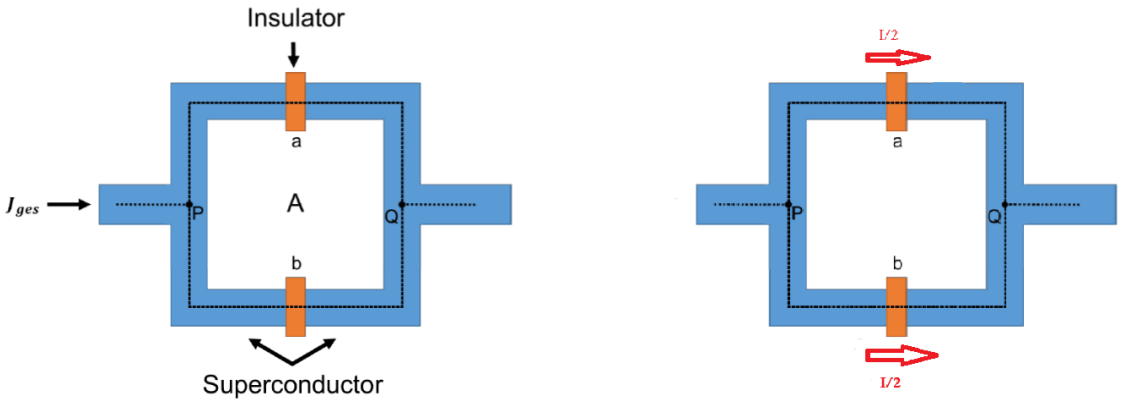
Chapter 3

System design

3.1. The Josephson Junction in AQFP's.

A Josephson junction is the key component in superconducting electronics with its main ability to carry a dissipation-less phase-driven current. Before understanding the working of the Josephson Junction, it is important to understand basic superconductivity. If you cool particular metals to very low temperatures, close to 0 Kelvin, a phase transition occurs. At this critical temperature, the metal goes from its normal state, where it has electrical resistance, to its superconducting state, where the electrical resistance of the metal is zero. This in turn means that there is no energy loss in superconducting metals. However, there is a maximum superconducting current known as the critical current.

A Josephson Junction is made by sandwiching a thin layer of insulating material between two layers of superconducting material. While the Josephson Junction is in its superconducting state, current can tunnel through the small insulator between the two metals with no resistance [6]. This superconductive loop is able to temporarily store a qubit with no loss in electrical current. However, if the current surpasses its critical limit a voltage will form across the junction and the metal will revert to its normal properties with electrical resistance. The Josephson Junction is built into a super conducting loop as shown in 3.1a. The two insulators highlighted in orange form two Josephson junctions at position a and b. These Junctions are joined with superconducting material to form a loop. When no external magnetic field is applied 3.1b the current flows through each side of the SQUID in equal parts.



(a) The basic SQUID layout. **(b)** Current through the SQUID.

The Josephson Junctions are set up as shown by the X's labeled JJ1 and JJ2 in 3.2 to create the basic superconducting AQFP buffer. First an input current is passed through inductor Lin. Then an AC excitation current is applied to Lx which creates a magnetic field. This magnetic field induces a small screening current that flows in either the first or second loop based on the direction of the input current. If the input current is positive and flows into the cell, the left Josephson junction is switched introducing a Single Flux Quanta into the left loop. The result is a large downward output current which represents a logical "1". When the input current is switched, the right Josephson Junction is switched, which reverses the output current representing a logical 0. The output of the AQFP travels along the inductor labeled Lq. This inductor is coupled to another inductor, Lout, which connects the output of the buffer to other components.

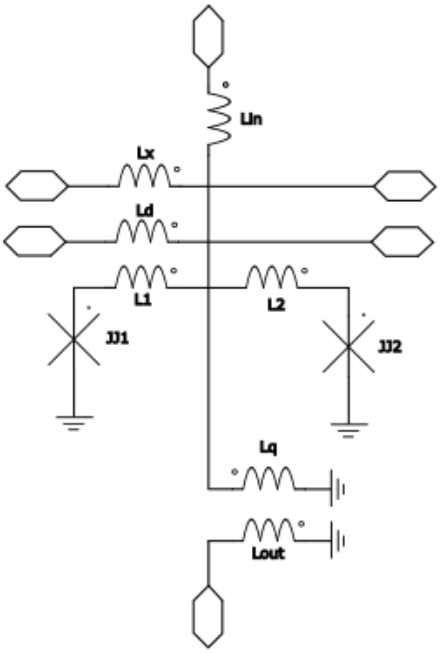


Figure 3.2: How Josephson Junctions are used to create the buffer

3.2. How AQFP's can be used to create basic logic gates.

With the ability to create a logical "1" or "0" AQFP's possess the capacity to construct logic gates employing the principle of majority logic. Majority logic involves taking multiple inputs (either a logical "0" or logical "1") and generating an output that corresponds to the most prevalent input. The truth table for a majority gate with three inputs is as follows.

Table 3.1: Truth Table for a 3 input Majority Gate

Input 1	Input 2	Input 3	Output
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

This concept is realized by combining the outputs of three components into a unified node. In AQFP logic, a logical 1 is denoted by an 80uA output, while a logical 0 is represented by a -80uA output. The value is not important, however, the fact that a logical 0 and a logical 1 are the same value with different signs enables majority logic to function effectively. The utilization of negative values to signify logical zeros allows for the addition or subtraction of signals, allowing the dominant signal to persist, the result is majority logic.
However, there are some major constraints regarding AQFP technology. First, individual components in AQFP circuits need to be clocked in order. This means the signals need to have the same path length to ensure that the two signals are passed through the logic gates in order.
The second problem stems from the division of current when a branch in the signal occurs. If the same signal is required in more than one operation the signal is split, this division leads to a division of the current amplitude in each branch, which can affect the outputs of logic gates such as AND and OR gates. To address this issue the two segments of the original signal are each passed through buffers. This ensures that the output signal maintains the correct amplitude. Therefore, splitters and buffers are essentially the same component. As a result of these problems AQFP circuits are made from approximately 50% buffer circuits apart from the components required to achieve the function of the circuit. This is shown by an example in 3.3.

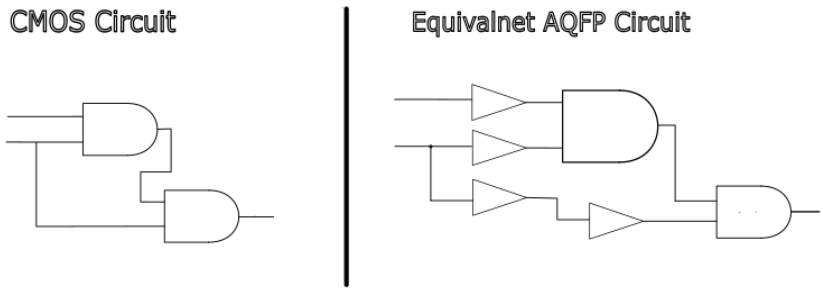


Figure 3.3: CMOS vs AQFP circuit Layout

3.2.1. The Excitation Current and DC offset

Two AC excitation currents are used to power and clock the following circuits. The two AC currents (Ix1 and Ix2) are 90° apart in phase. These excitation currents apply an AC magnetic flux with an amplitude of 0.5Φ. Logic operations are performed from phase $\theta 1$ to $\theta 4$ with a phase separation of 90°. Logic gates at $\theta 1$ and $\theta 3$ are clocked at the rising and falling edge of Ix1, respectively. Logic gates at $\theta 2$ and $\theta 4$ are clocked at the falling and rising edge of Ix2, respectively. Both sinusoidal excitation signals have a frequency of 5GHz which allows 5 sequential logic operations to happen every $200ps$. There is an additional DC current (Id) that applies a DC magnetic flux of $+-0.5\Phi$.

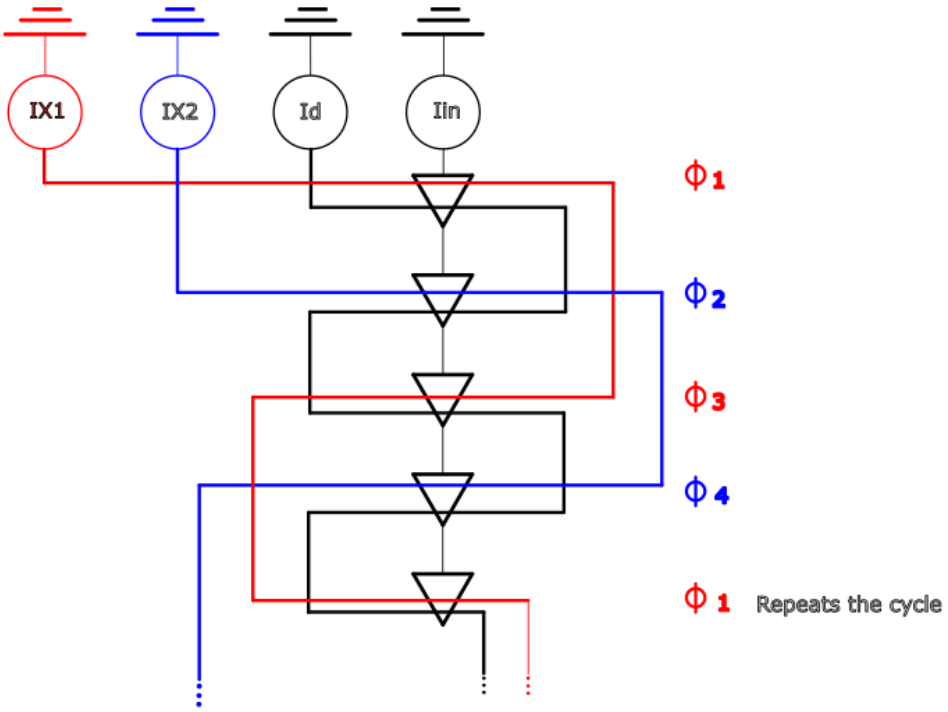
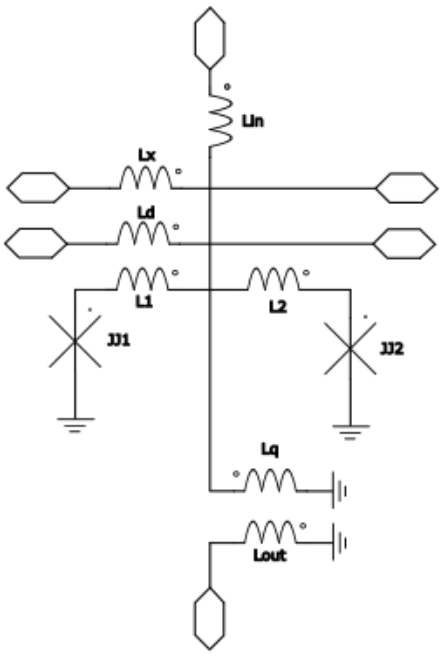


Figure 3.4: Excitation currents.

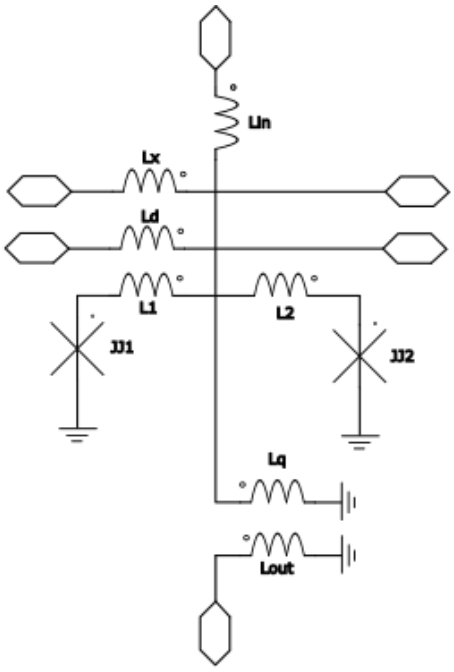
3.2.2. The Buffer and NOT gate

Creating the buffer using the AQFP. The buffer is the most basic logic component within AQFP and forms the base for all other AQFP based logic components. The buffer is a basic element that can temporarily hold a qubit value or propagate it through a circuit. The buffer is set up as seen in 3.5a. The inductors Lx and Ld as shown in the figure are not connected to the buffer but are coupled to inductor L1 and L2. Lx and Ld carry the AC excitation current and the DC offset current respectively. This allows the AC and DC current as mentioned earlier to clock and power the entire AQFP buffer. The coupling factor between the inductors L1 ,L2 and inductors Lx and Ld need to be equal in value as well as the inductance value of L1 and L2 as the symmetry of the AQFP cell is important to ensure that there is no bias in the cell as a slight bias could favour either a negative or positive output.

The output current of the AQFP passes through inductor Lq which is coupled to an output inductor Lout. This is done to ensure that the output current carried in inductor Lq is almost independent of what is connected to the input and output ports. Note the polarity of the inductors as shown by the dots on Lq and Lout in 3.5a. When creating gates such as the Not, NAND and NOR gates the polarity of inductor Lout is simply changed to induce a negative output current rather than a positive one, this is done by simply changing the coupling factor to be negative or reversing the polarity of Lout as shown by the dots on Lout in 3.5b.



(a) The basic Buffer Layout.



(b) The basic Not gate Layout.

3.2.3. The Constant Cell

The constant cell is designed output a constant logical 0 or 1 on every clock cycle. This means there is no input current and inductor Lin can be removed from the buffer design. However, the output still needs to be predictable even without an input current. The constant cell is designed by changing the architecture of the symmetrical buffer and making it asymmetrical to favour either a positive or negative current. This is achieved by simply allowing the value of inductor L1 or L2 to be larger subsequently the inductor with the larger value shall also have a greater coupling factor with respect to Lx and Ld. This allows the constant cell to output either a zero or a 1 without an input. If inductor L2 is larger the coupling between Lx, Ld and L2 will be larger and a single flux quanta will be induced in the second Josephson junction favouring the logical 0 output. Constant cells are useful and necessary when designing the AND and OR gates using majority logic as seen in the next section.

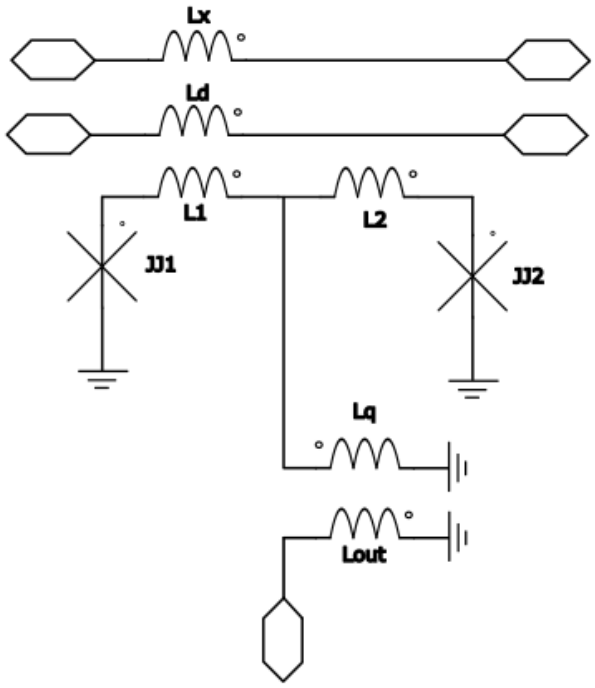


Figure 3.6: Layout of the constant cell

3.2.4. The AND Gate

The AND gate is constructed by arranging two buffers and one constant cell in parallel as seen in 5.5. These two buffers serve as the inputs to the AND gate, generating a logical 0 or 1 based on their respective inputs. The outputs of the three gates are coupled to their own respective Lout inductors to ensure that the other output signals do not interfere with each other. These inductors are then connected by a majority circuit, which produces the output corresponding to the most common value. This means the final output will only be 1 when both of the inputs to the AND gate are 1 and the output will be 0 for any other combination of inputs, recreating the behaviour of an AND gate.

3.2.5. The OR Gate

The OR gate is made similarly to the AND gate but replaces the constant 0 cell with a constant 1 cell. The coupling of the inductors at the output is kept the same to ensure there is no interference with the individual output signals. Once again all the individual outputs are combined using a majority circuit. Replacing the constant 0 cell with a constant 1 means the final output will be 1 when any or both of the inputs are 1, recreating the behaviour of an OR gate.

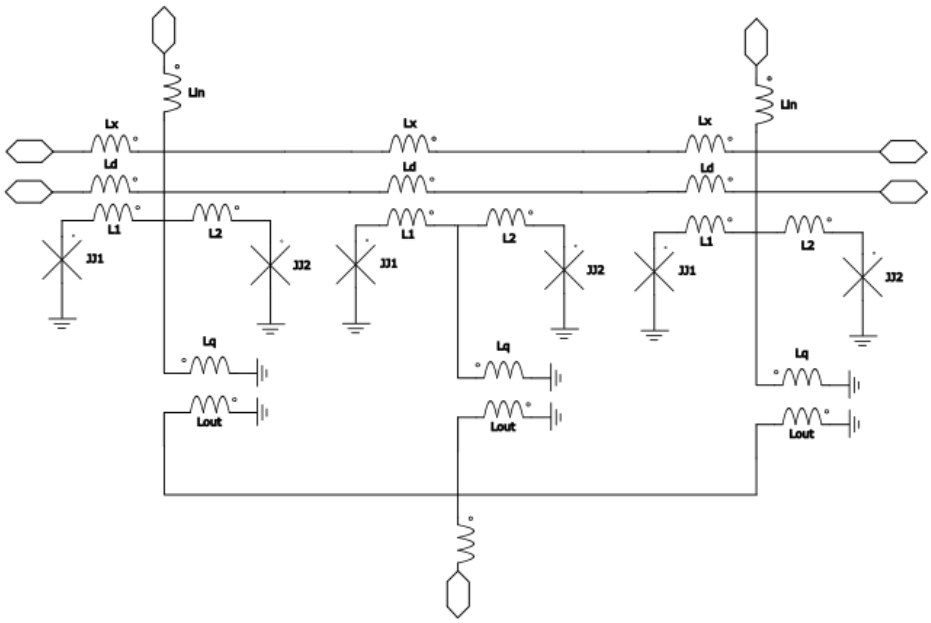


Figure 3.7: Circuit Layout of the 2 Input AND/OR Gate

3.2.6. The 3 Input AND/OR Gate

The three input AND/OR gate is made similarly to its two input counterpart but instead of two buffers and one constant cell it now contains three buffers and two constant cells. The outputs are once again all coupled to their respective Lout inductors separately to ensure their individual output signals are not disturbed and the Lout inductors are joined using majority logic. The 3 input AND gate is made by using two constant 0 cells while the OR gate is made from two constant 1 cells. The circuit layout of the three input AND/OR Gate can be seen in 3.8. The output of the 3 input AND gate is only 1 when all three of the inputs is equal to 1, and the output of the 3 input OR gate is equal to 1 as long as 1 or more of the inputs is equal to 1.

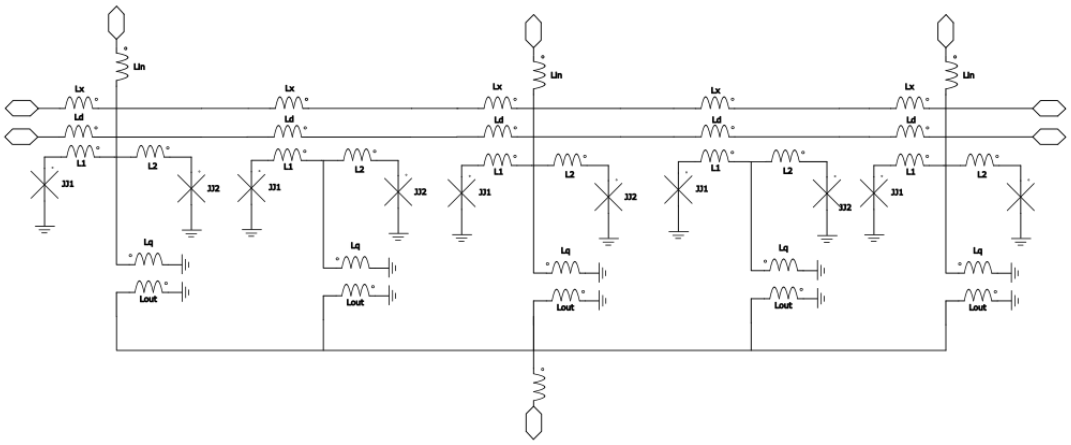


Figure 3.8: Layout of the 3 input AND Gate.

3.3. Compiling basic logic gates to make complex multi-gates.

3.3.1. The XOR gate

The XOR gate cannot be considered a basic logic gate as it is in CMOS. This is because XOR gates cannot be implemented using majority logic and instead makes use of the combination of AND and OR gates to achieve the same functionality. The XOR gate created and tested in JoSim was setup as follows.

3.3.2. The Half Adder

A half adder is used for adding together the two least significant digits in a binary sum and forms part of the Full Adder, a significant component used in logic Arithmetic operations. The Half adder design is shown in 3.10a.

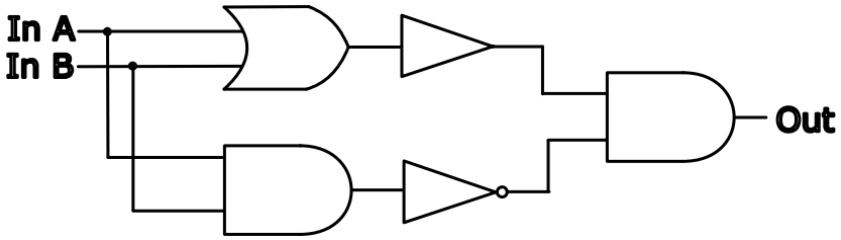
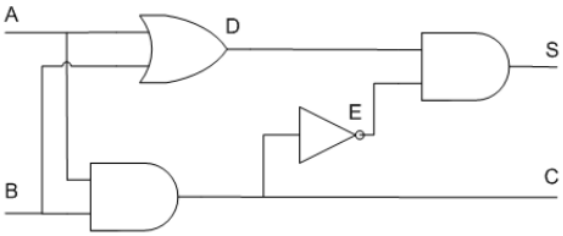


Figure 3.9: Layout of the XOR gate



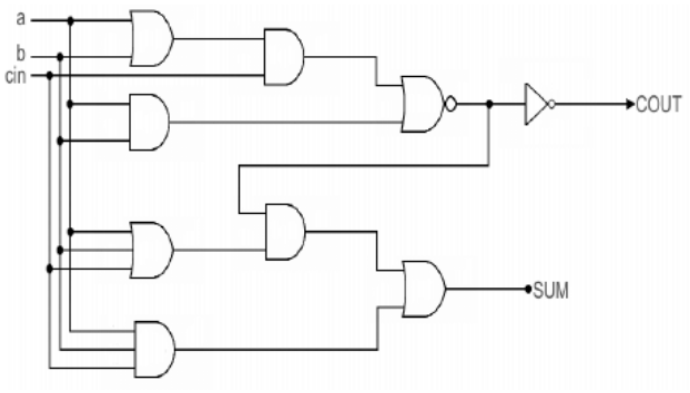
(a) The layout of the Half Adder.

Inputs		Outputs	
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

(b) Half Adder Truth Table.

3.3.3. The Full Adder

Full adders are useful complex logic gates used in ALUs (arithmetic logic units) which is a component in the CPU crucial to performing mathematical operations. Full adders are also used to create digital circuits like multiplexers. By combining AND, OR and NOT gates as shown in 3.11a the Full Adder operation was achieved. Figure 3.11b shows the truth table for the Full Adder.



(a) The layout of the Full Adder.

Inputs			Outputs	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

(b) Full Adder Truth Table.

Chapter 4

Physical Design

4.1. Using Klayout to design the AQFP cells

The following sections show the final design of the various AQFP logic gates in KLayout software. The final design is a result of multiple tests and comparisons to ensure that not only do the gates work, but that they have been optimized to ensure a reasonable output current and performance is achieved.

Note, inductors that are the same colour are on the same level. The ground plane (layer M4) is indicated by the large light blue area that surrounds all the other layers. In the following diagrams the ground layer is the bottom most layer. The black text in the following diagrams label the input pins, output pins and Josephson junctions as follows: "Pxin M5 M4", defines the pin that inputs current to inductor Lx which carries the AC excitation current. M5 represents the layer the inductor is on and M4 is the layer the pin is grounded by. "Pdcin M5 M4", has the same properties but carries the DC offset current. "J1 M6 M5", labels the first Josephson junction where connections are made to layer M6 and the junction is grounded through M5. "J2 M6 M5", is identical to the first Josephson junction but is labeled as the second Josephson junction. Finally "Pout M6 M4" and "Pin M6 M4", label the input and output pins to the AQFP buffer, they are both on layer M6 and are both grounded by layer M4. The first image 4.1 has additional labels L1, L2, Lq and Lout to make it easier to identify the different components.

4.1.1. The Buffer

The following image shows a broken-down version of the buffer in KLayout. The design is broken down to reveal hidden layers that lie under other layers.(Note that Lout has been split in half and the right half has been moved off-centre to reveal Lq below it.) The two red tracks running parallel to each other at the very top of 4.1 are the inductors Lx and Ld that clock the rest of the buffer circuit.

The darker of the two blue materials represents inductors Lin, L1, L2, and Lq as labeled in the image 4.1. The green and orange layers form part of inductor Lout, which lies partially above and partially below inductor Lq. This allows Lq and Lout to have a high

mutual inductance to ensure that the current on Lout does not decrease from that present in inductor Lq. The blue upside-down "T" that is formed from inductors L1, L2 and Lin is perfectly centred in comparison to the rest of the circuit to ensure all the mutual inductance's between various components are equal, symmetry is of key importance in the buffer circuit. All the lengths of the different components are chosen to create specific inductance values that allow the circuit to work optimally.

Include a new image that contains moats t show what the moats would look like explain what they are there. Then explain that these moats are excluded in other diagrams to allow them to be easily decipherable

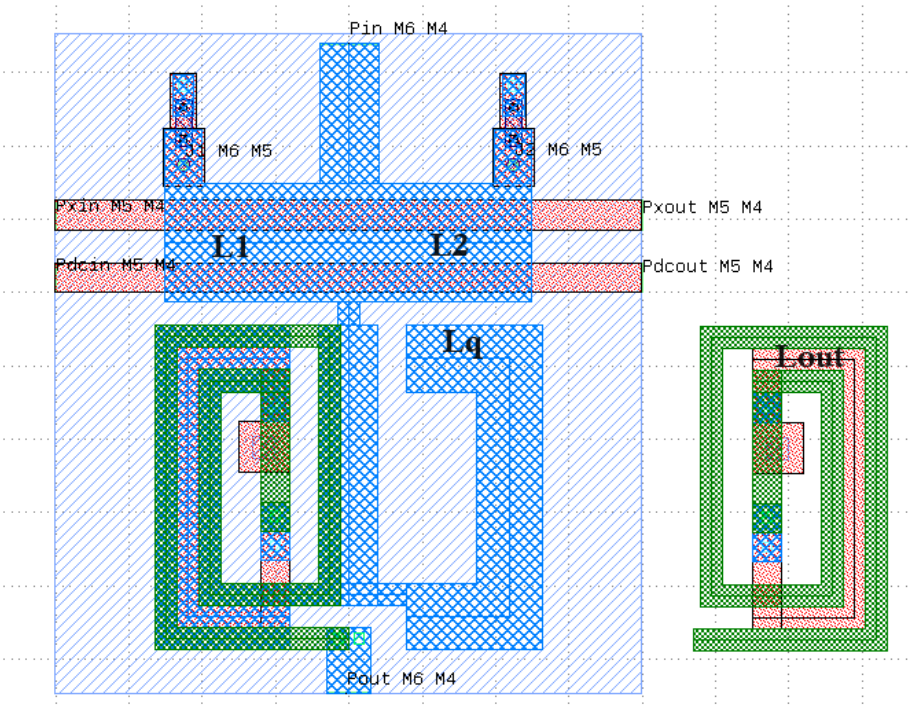


Figure 4.1: Deconstructed Buffer

4.1.2. The constant 0 cell

The following image shows a broken-down version of the constant 0 cell in KLayout. The design is once again broken down, removing part of Lout to reveal the hidden layers that lie under other layers.

The constant 0 cell has an almost identical layout to the buffer cell, however, the input inductor has been removed and the perfectly symmetrical circuit is now asymmetrical. The top of the circuit is moved off centre, to the right, to allow for inductor L2 to be greater than L1. This small change allows the mutual inductance between Lx, Ld and L2 to be greater than that of Lx, Ld and L1. Due to this difference the second Josephson junction is favoured and a single flux quanta is induced in the second JJ. This allows the circuit to have no input and to constantly output a logical 0. All the lengths of the

different components are chosen to create specific inductance values that allow the circuit to work optimally.

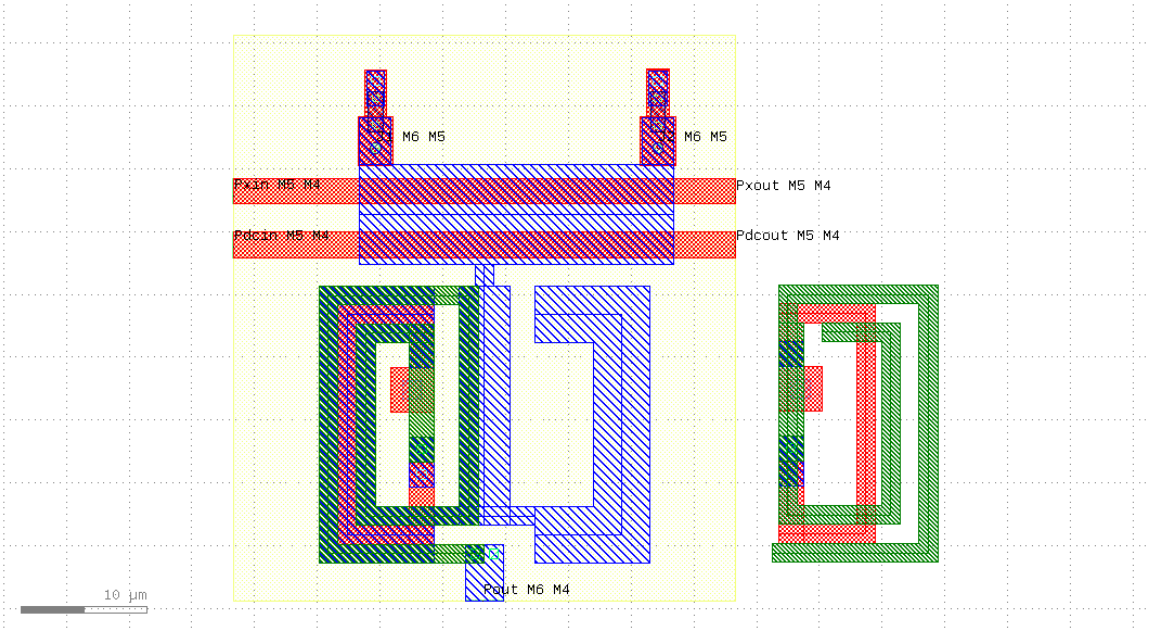


Figure 4.2: The diagram of the deconstructed Constant 0 cell.

4.1.3. The constant 1 cell

The following image shows a broken-down version of the constant 1 cell in KLayout. The design is once again broken down, removing part of Lout to reveal the hidden layers that lie under other layers.

The constant 1 cell has an almost identical layout to the buffer cell, however, the input inductor has been removed and the circuit is now asymmetrical. The top of the circuit is moved left off centre to allow for inductor L1 to be greater than L2. This small change allows the mutual inductance between Lx, Ld and L1 to be greater than that of Lx, Ld and L2. Due to this difference the first Josephson junction is favoured and a single flux quanta is induced in the first JJ. This allows the circuit to have no input and to constantly output a logical 1. All the lengths of the different components are chosen to create specific inductance values that allow the circuit to work optimally.

4.1.4. The branch

The branch is used to combine multiple buffers and constant cells together to create AND and OR gates. In 4.4 the three pin labels on the top of the diagram labeled Pa, Pb and Pc represent three different inputs to the inductors La, Lb and Lc. The label Po represent the output of the branch. Inductors La and Lc are identical in size and length meaning they have the same inductance. These inductors are responsible for carrying the outputs

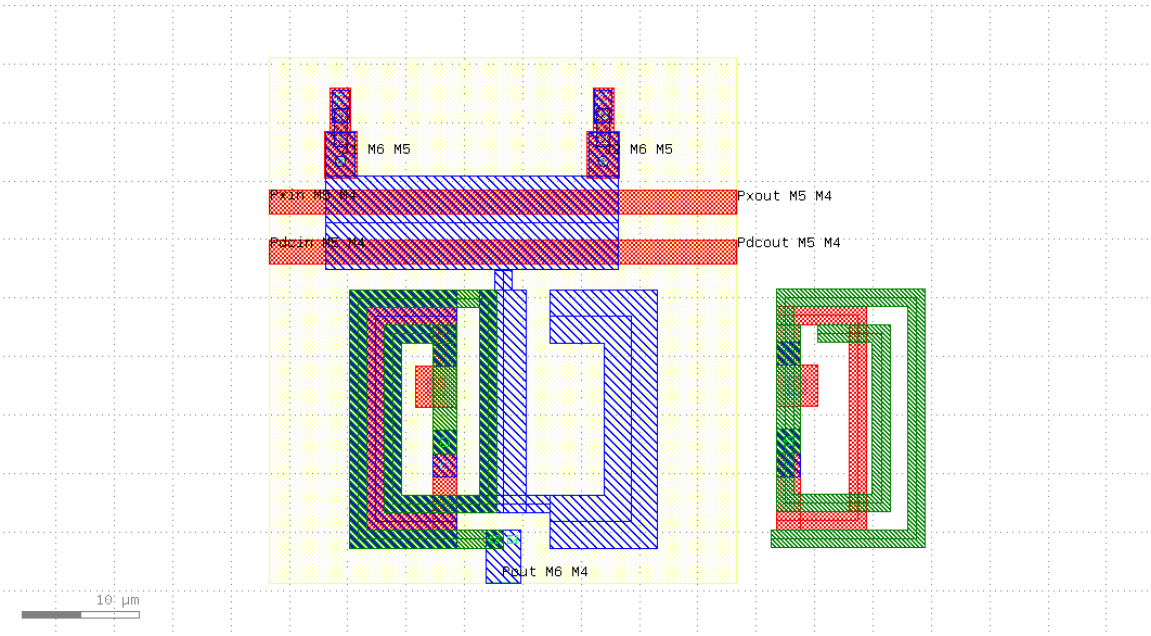


Figure 4.3: The deconstructed diagram of the Constant 1 cell.

of the two Buffer gates. Inductor Lb is much smaller in width and longer in length to ensure that its inductance is higher than that of La and Lc. Inductor Lb is responsible for carrying the output of the Constant 0 or Constant 1 circuit. The outputs of the two buffers and constant cells travel through La, Lb and Lc and are combined at Lo to create the final output for AND and OR gates. Similar branches can also be extended to combine more signals such as 3 input AND and OR gates.

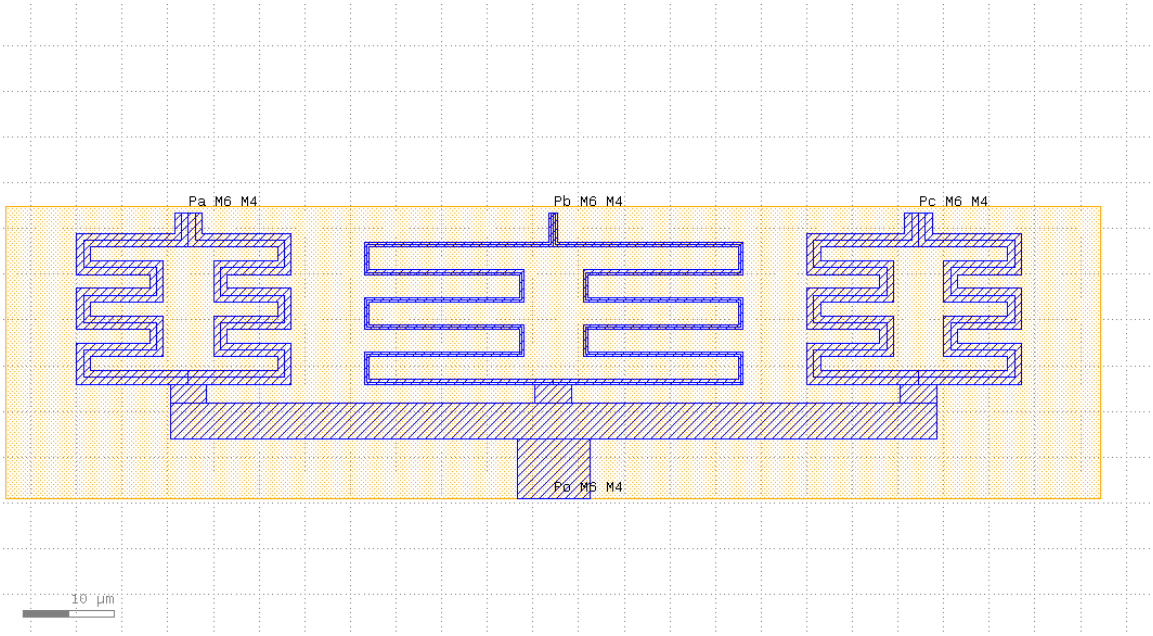


Figure 4.4: The physical Layout of the branch in KLayout

4.1.5. The And Gate

The following diagram shows the combination of the AQFP buffer, constant 0 cell and branch circuit to create the AND gate. Note the constant cell has the top inductor component shifted to the right, a characteristic of the constant 0 cell. The Lout inductor of each of the buffer circuits and the constant 0 circuit is connected to the inputs of the branch circuit. These output signals are combined and cancelled to allow the most dominant signal to remain. This final signal is the output seen at port Po.

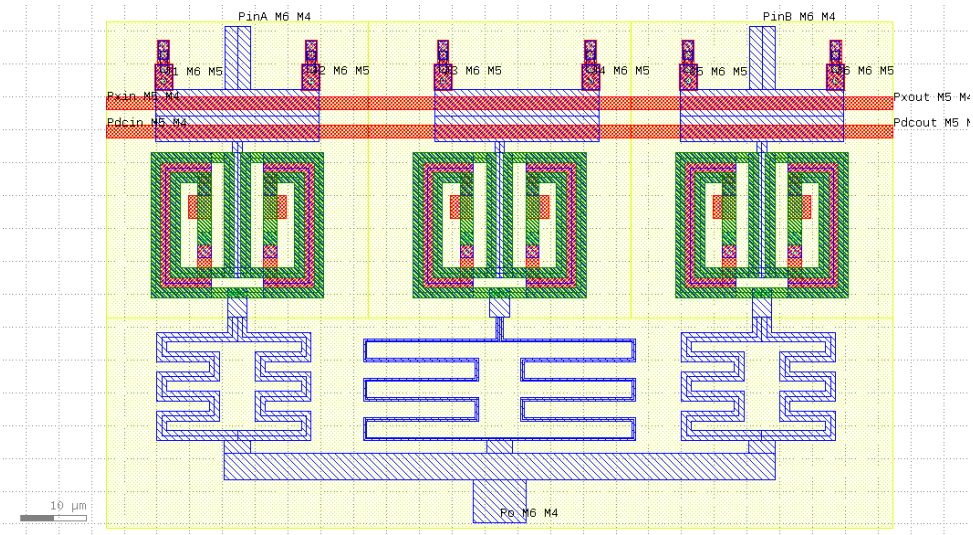


Figure 4.5: The physical layout of the AND gate in KLayout

4.1.6. The Or Gate

The following diagram shows the combination of the AQFP buffer, constant 0 cell and branch circuit to create the OR gate. Note the constant cell has the top inductor component shifted to the left, a characteristic of the constant 1 cell. The Lout inductor of each of the buffer circuits and the constant 1 circuit is connected to the inputs of the branch circuit. These output signals are combined and cancelled to allow the most dominant signal to remain. This final signal is the output seen at port Po.

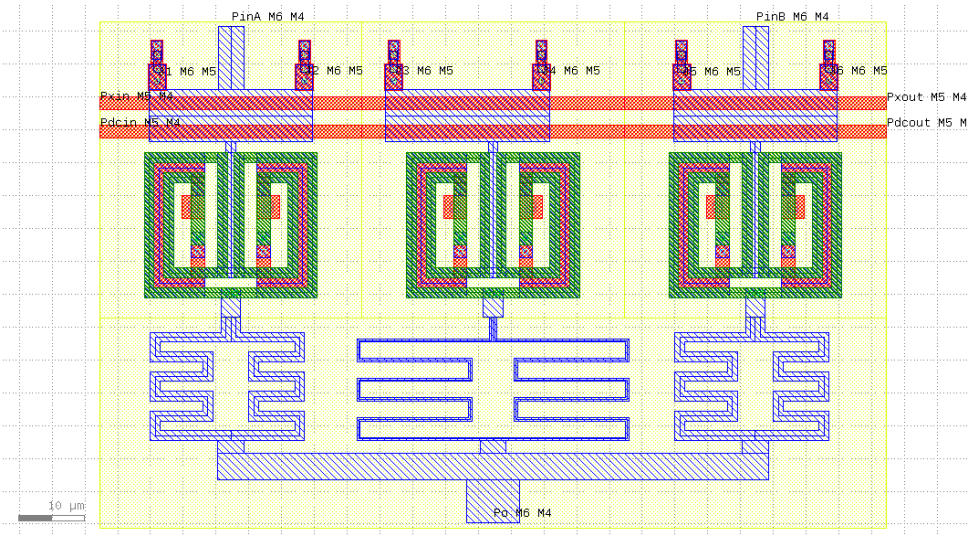


Figure 4.6: The physical layout of the OR gate in KLayout

Chapter 5

Results

5.1. Using JoSIM to simulate logic gates.

In the following sections the inductance's of the various KLayout models were extracted using InductEx, a software specifically made to perform this task. These extracted values replaced the values on the netlist that were previously obtained from MIT. This created a unique set of gates that could be tested using JoSim. To test the buffer an input signal will be passed through a series of 8 buffers to ensure the signal is maintained without major distortion. To test the logic gates the signal will initially be passed through a series of 4 buffers, then through the logic gate being tested and finally through another 4 buffers. This is to ensure that the slight distortions the signal experiences while being passed through a buffer has no affect on the logic gates and that the output doesn't change no matter the order of the gates.. When designing the following gates the output of the gates is aimed at being above 80uA. The output is also required to be clean without too much noise to ensure that a clear logical 1 or 0 can be read from the output and passed on to the next circuit.

Maybe explain what JoSim does and how it does it this is included in the JoSim package blah blah and jst type some more

5.1.1. Buffer Output

The square wave shown in the top left of 5.1 shows the input to the chain of buffers. The following buffers show that the signal is maintained. However, it must be noted that as the signal travels further down the chain of buffers more noise is introduced at the beginning of the signal. This is due to the delay of the signal through the buffers. The buffer is outputting a signal before it has received an input from the previous buffer, therefore, introducing noise onto the beginning of the transmitted signal. The output signal has a current with an amplitude above 80uA which is sufficient and the 1's and 0's in the output are clear and distinguishable, the buffer circuit works as expected.

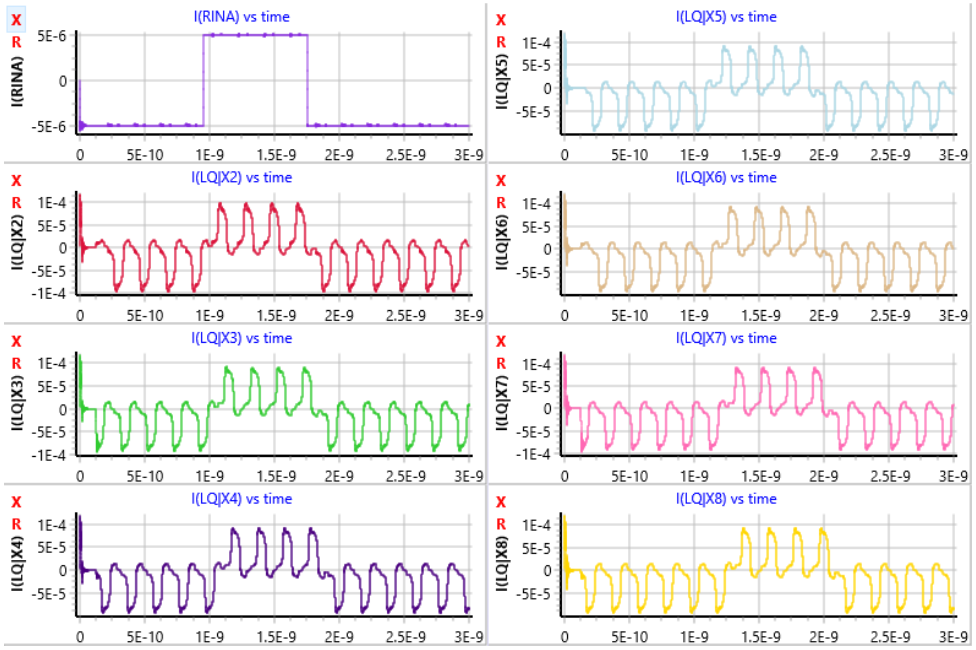


Figure 5.1: Output through a string of buffers.

5.1.2. Not Gate Output

In figure 5.2 the output is a result of the input being fed into a single not gate and then passed through a series of buffers. The single not gate will reverse the input signal, which will be passed through 7 buffers to ensure the signals maintains its amplitude. The output signal has a current with an amplitude above 80uA which is sufficient and the 1's and 0's in the output are clear and distinguishable, the Not gate works as expected. Note, the output is the exact opposite to the previous output shown in 5.1.

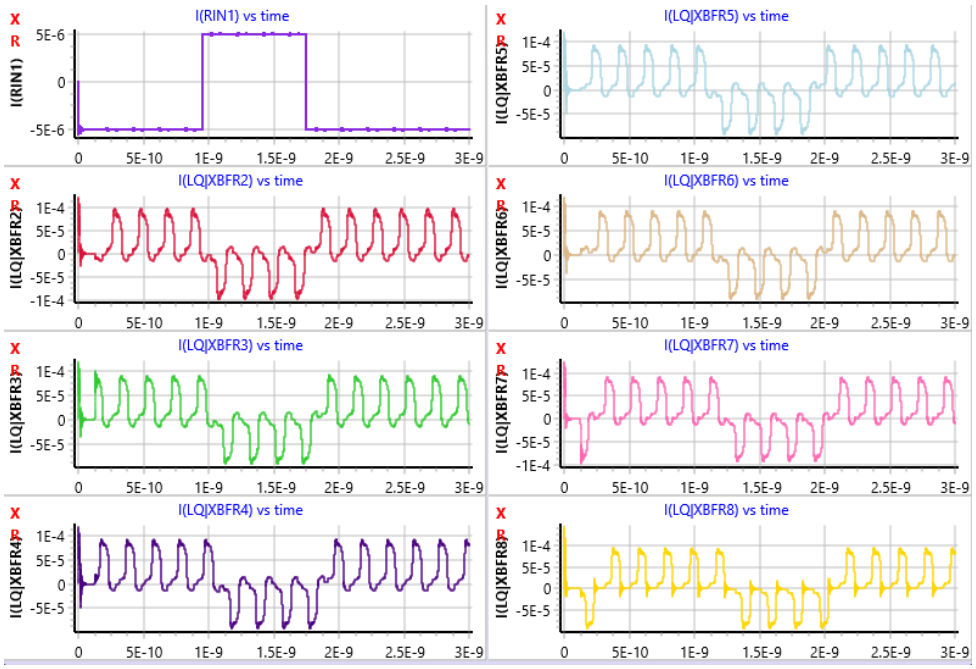


Figure 5.2: Output of the Not Gate through a series of buffers

5.1.3. Constant Cell "0" Output

The following image will show the output of the constant 0 cell fed only through one standard buffer. Due to the fact that the constant cell will only be used in majority logic and will almost always be added or subtracted from another signal the test is done differently. The output is only passed through one buffer to see how the output of the constant cell is read. This shows how the output signal appears when it is added or subtracted from the other signals in majority logic. The output is slightly more noisy than the buffer, however, the output clearly displays logical 0's and has an amplitude of roughly 80uA which is what is needed for the addition and subtraction of signals in majority logic, thus the constant cell is functioning correctly.
A slight inconsistency of the constant 0 cell are the small peaks that lie above 0uA. This is not a major problem as these amplitudes are not high enough for the buffer circuit to read, however, when the constant cell is used in majority logic these peaks can add up to be significant enough to cause an error. This is seen later in the 3 input AND and OR gates.

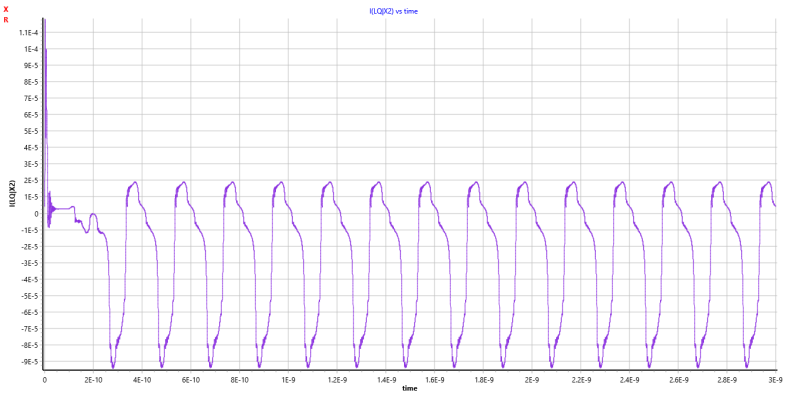


Figure 5.3: Constant 0 Cell output

5.1.4. Constant Cell "1" Output

The constant 1 cell is identical to the constant 0 cell but due to the opposite shift in inductors as mentioned in the physical design section the output is opposite as shown below in 5.4. Once again the output is roughly 80uA and is clear, thus the constant 1 cell is functioning correctly.
Once again the constant 1 cell displays small peaks that lie below 0uA, similar to the constant 0 cell but reversed. This has the same affect, there will be no major issue until the constant cell is used in majority logic. In logic such as the 3 input AND and OR gates multiple constant cells are used and the peaks can add and cause inconsistencies in the output.

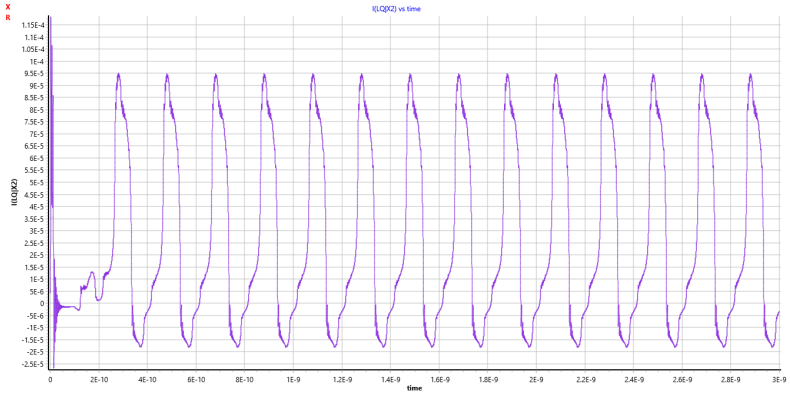


Figure 5.4: Output of the Constant 1 cell.

5.1.5. Output of the AND Gate

The output seen in 5.5 is a result of two inputs put through a series of buffers and then into an AND gate and the final result through another series of buffers. This is done to ensure the buffers are able to read the signal of the AND gate correctly and carry the signal over another series of buffers without distorting the signal. The result shows that the output is only one when both of the input signals is one. This is the correct functionality of the AND gate, however, the signal appears to be shifted with extra bits attached to the front of the expected signal. This can be attributed to the delay of the signal through the buffers and AND gates. The output is clear and has an amplitude greater than 80uA, thus the AND gate is functioning correctly.

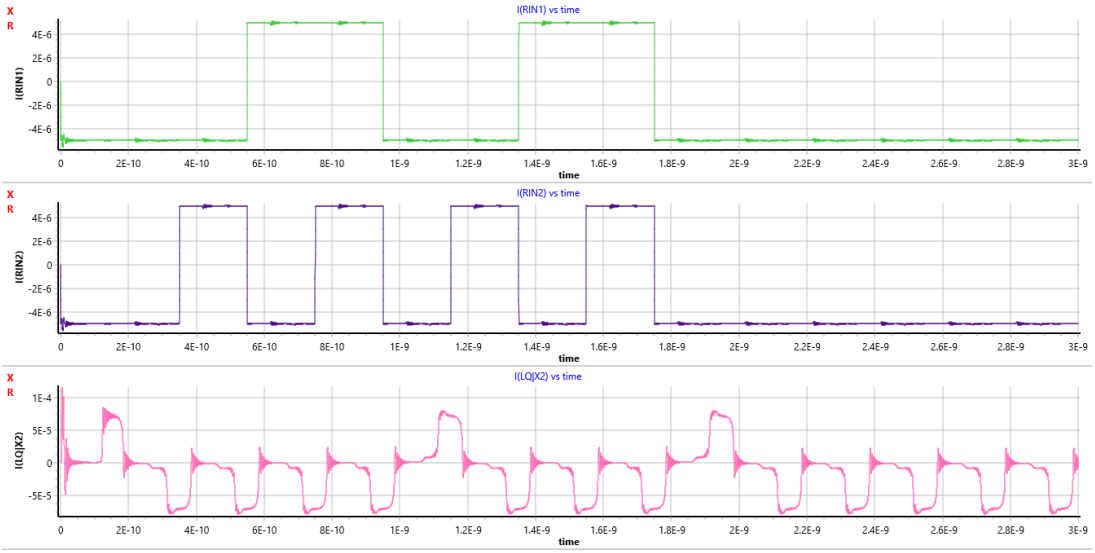


Figure 5.5: Input vs Output of the AND gate

5.1.6. Output of the OR Gate

The OR gate is tested exactly as the AND gate is, as described in section 5.1.5, of course replacing the AND gate with the OR gate. The result shows that the output is equal to 1 when either one or both of the inputs is equal to a logical 1. The final output is clear and is above 80uA, thus the OR gate functions as expected.

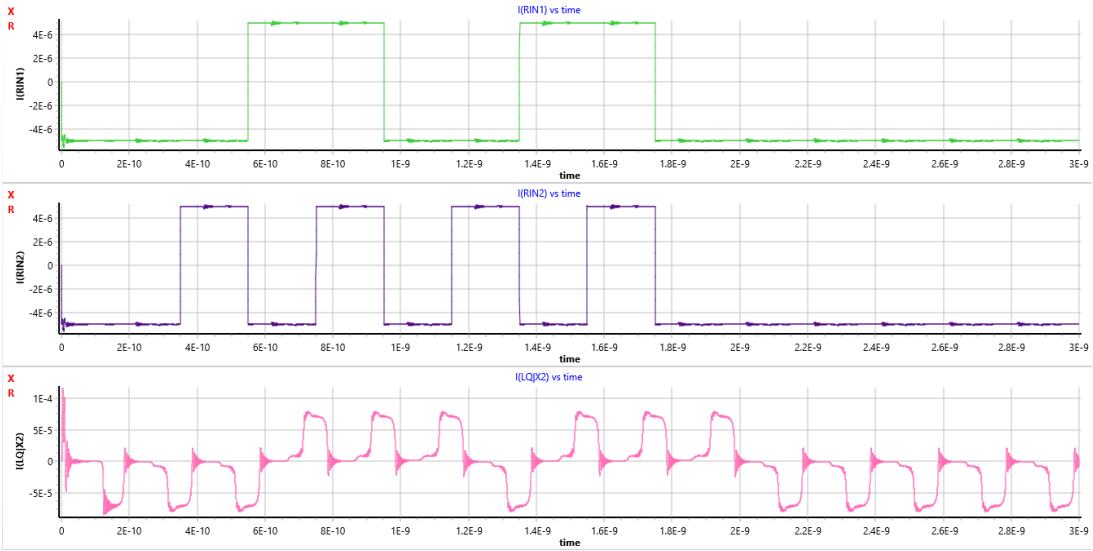


Figure 5.6: Input vs Output of the OR gate

5.1.7. The 3 input AND gate

Figure 5.7 shows the input to the 3 input AND gate. The output in figure 5.8 shows that the gate only outputs a 1 when all of the inputs to the gate are also 1. This behaves as an AND gate should, however, the output signal is recorded in the first and second buffer after the 3AND gate to show the difference between the 2 input AND gate and the 3 input AND gate. Due to the fact that the constant cells don't output a value exactly equal to that of the buffer the more buffers and constant cells you connect to a branch the more distorted and unreadable the output becomes. Although the output is correct as seen in the first and second buffer (top and bottom graph) it is easy to see the distortion in the first graph. This problem would be exaggerated as more constant cells are placed in parallel. The output in the second buffer is clear and has an amplitude above 80uA, although the immediate output is not perfectly clear the 3 input gate can be used if the constant cell is improved to output a signal that has a closer resemblance to that of the buffer or if two buffer cells are placed immediately after the 3 input AND gate.

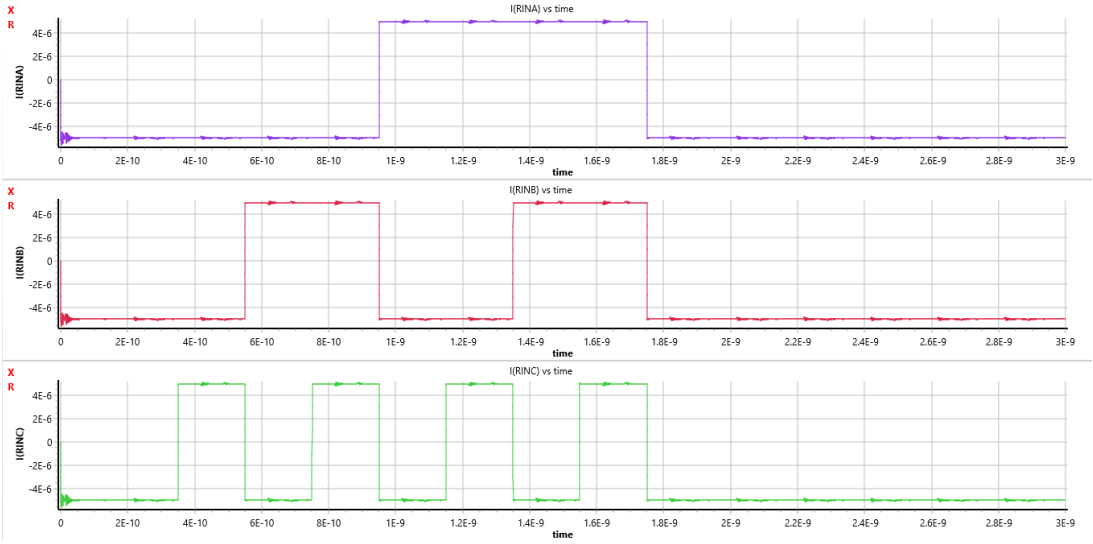


Figure 5.7: Input to the 3 input AND gate

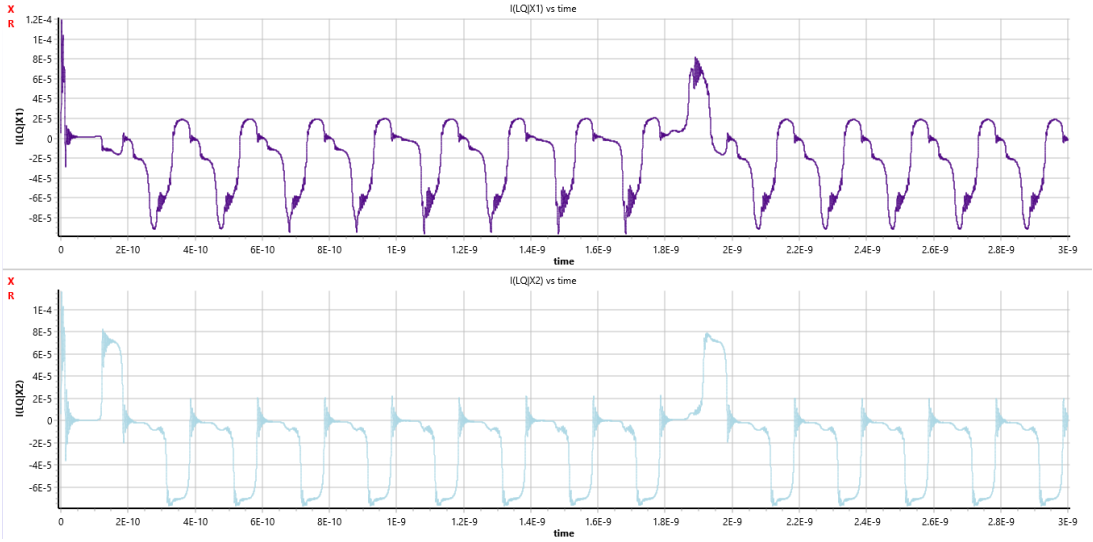


Figure 5.8: Output of the 3 input AND gate

5.1.8. Output of the three input OR gate

Figure 5.7 shows the input to the 3 input OR gate. The output in 5.10 shows that the gate outputs a 1 when either one or more of the inputs is equal to 1. This behaves as an OR gate should. Once again the output signal is recorded in the first and second buffer after the OR gate to show the distortion in the signal. The output in the second buffer is clear and has an amplitude above 80uA, although the immediate output is not perfectly clear the 3 input gate can be used if the constant cell is improved to output a signal that closely resembles that of the buffer or if two buffer cells are placed immediately after the 3 input gate.

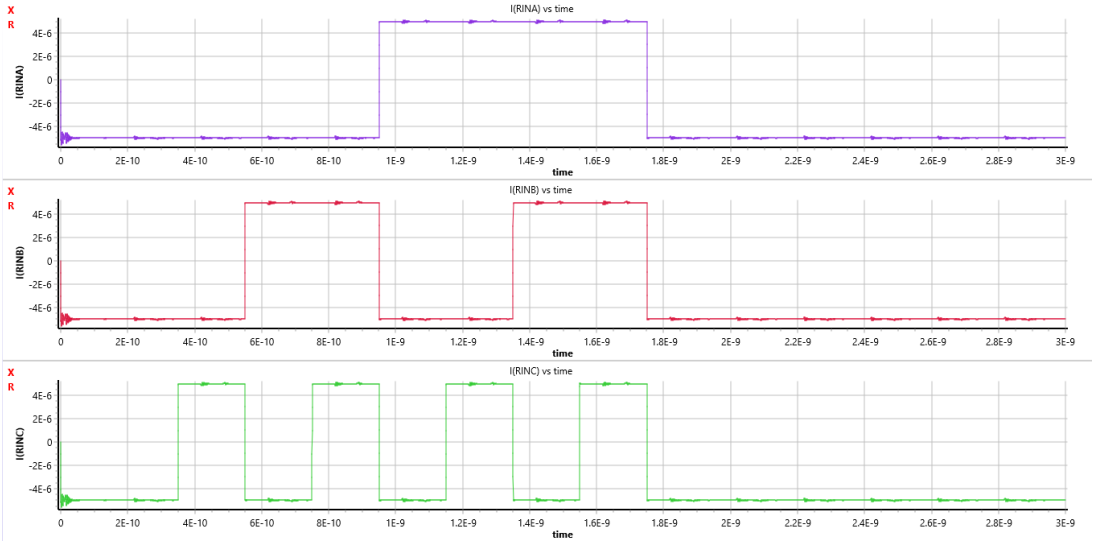


Figure 5.9: Input to the three input OR gate

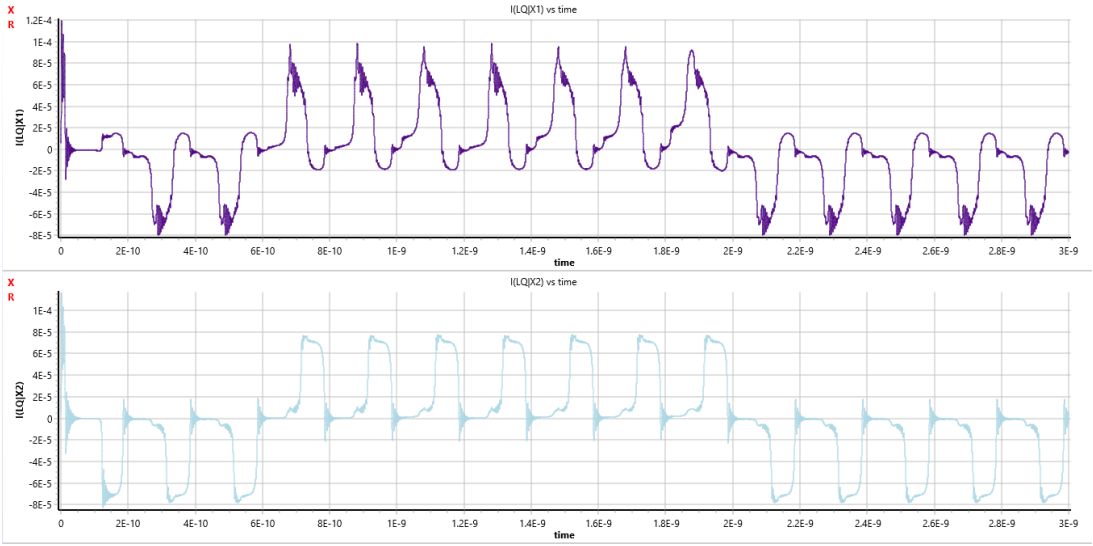


Figure 5.10: Output of the three input OR gate

5.1.9. Output of the XOR gate

The XOR gate cannot be implemented using majority logic as described in section 3.3.1 and must instead be created using a combination of AND, OR and NOT gates. The following output is a result of the two input signals passed through a series of buffers and then through a combination of AND and OR gates to create the XOR gate as described in a previous section. This result is then passed through another series of buffers to ensure the signal can be read and carried correctly without distortion. The output is clear and the current remains above 80uA, thus the XOR gate functions as expected.

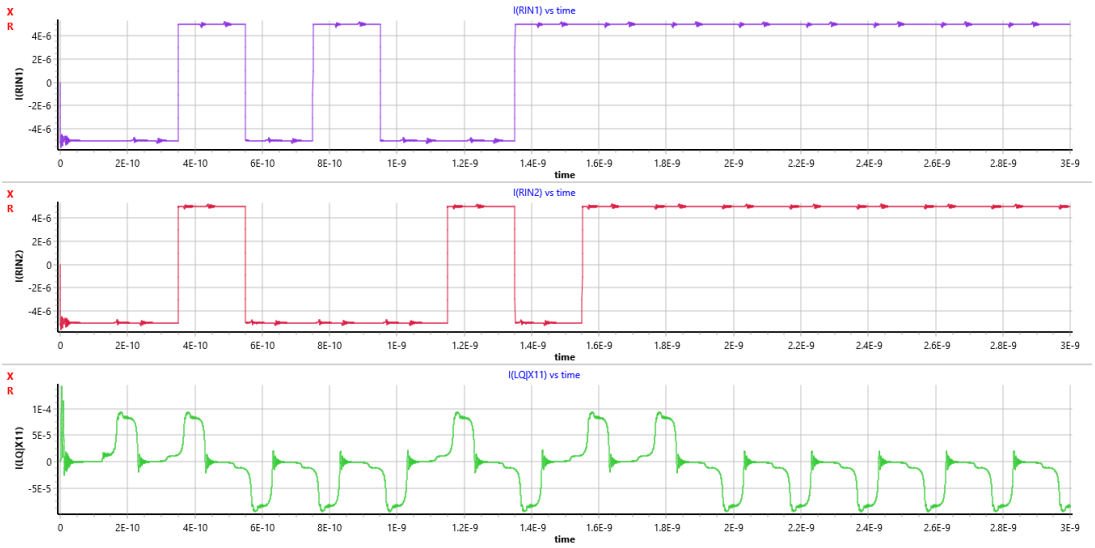


Figure 5.11: Output of the XOR gate

5.1.10. Output of the Half Adder

JoSIM is used to simulate the operation of the Half Adder and to record the output. The following 5.12 shows the input to the Half Adder while 5.13 shows the output of the Half Adder. The signal at the output for the Half Adder contains two initial bits that are unrelated to the truth table. This can be attributed to the signal delay through the Half Adder. Both the *Carry* and *Sum* output of the Half Adder contain these extra bits before the expected signal is received. After the input signal has finished following the truth table, the input is set to zero to force the output to zero. This allows the expected signal to be easily identified. As a result, four zero bits follow the expected signal at the output.

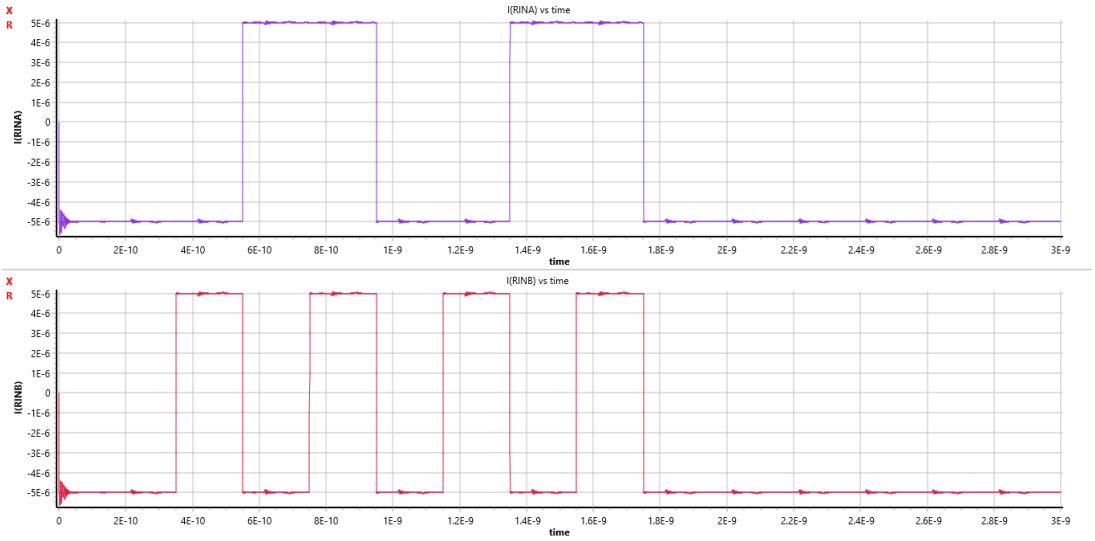


Figure 5.12: Input to the Half Adder

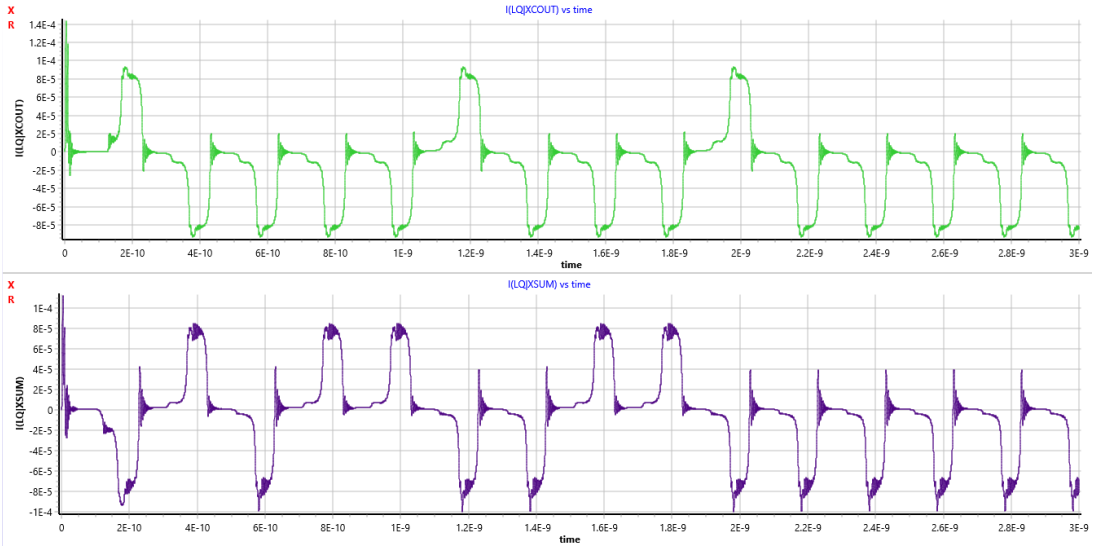


Figure 5.13: Output of the Half Adder

5.1.11. Output of the Full Adder

JoSIM software was used to simulate the working of the Full Adder design. The first figure 5.14 shows the input to the Full Adder while 5.15 shows the output of the Full Adder. The signal at the output for the Full Adder contains four initial bits that are unrelated to the truth table. This can once again be attributed to the signal delay through the Full Adder. Both the *Cout* and *Sum* output of the Full Adder have 4 noise bits before the expected signal is received. After the input signal has finished following the truth table, the input is set to zero to allow the output to be easily identified. As a result, two zero bits follow the expected signal at the output.

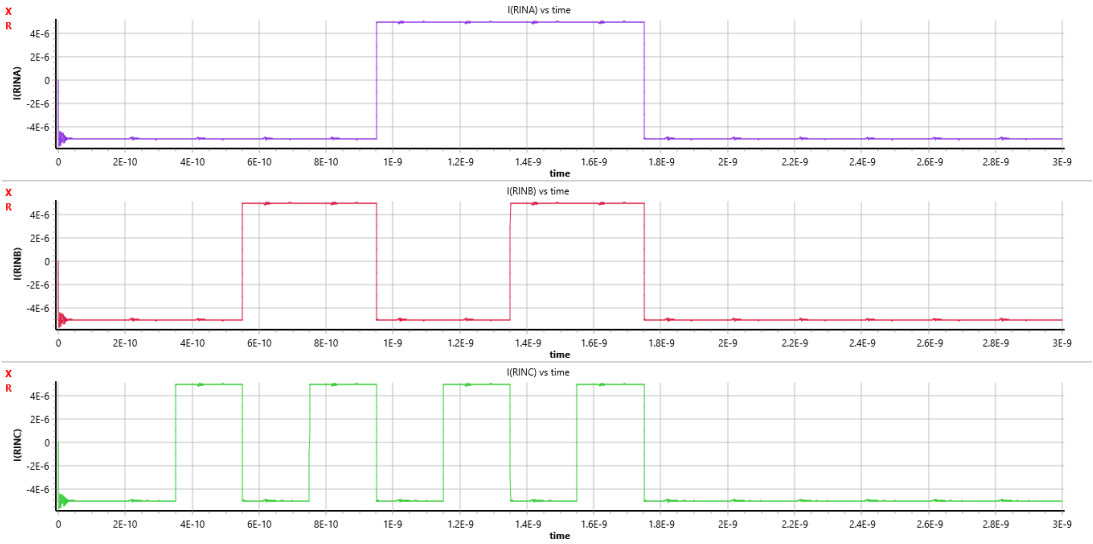


Figure 5.14: Input to the Full Adder.

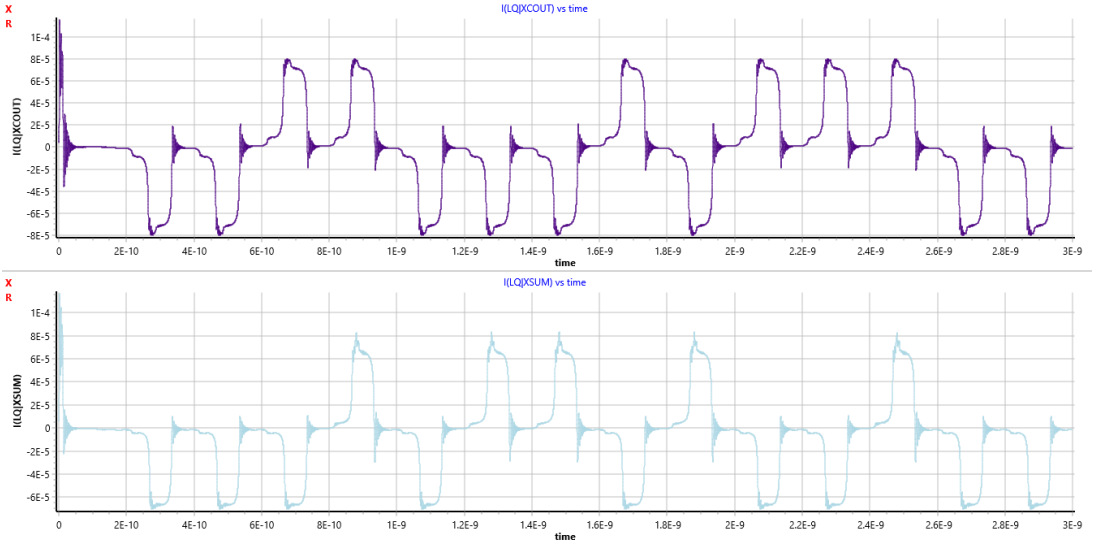


Figure 5.15: Output of the Full Adder

Chapter 6

Summary and Conclusion

6.1. Summary

In summary the development of the basic logic gates in AQFP technology was successful. InductEx extracted the real inductance's from the physical model made in KLayout. These inductance values were then used in the netlist of each of the the AQFP cells. JoSim simulated the results and the output of all the basic logic gates and complex multi-gates was correct and of a sufficient amplitude. The end result is a set of working AQFP logic gates that can be produced and have been tested to work with each other to produce complex computer systems that will function with low power consumption at low temperatures.

6.2. How I met the objectives

6.2.1. Understanding the Josephson junction.

What do I even say here. By creating a working buffer circuit that successfully holds and outputs a current according to the clocking signal an understanding of the Josephson junction is shown.

6.2.2. How AQFP can be used to perform basic logic operations.

By successfully implementing the AQFP buffer and constant gate, understanding the behavior behind majority logic and successfully creating AND and OR gates an understanding of how AQFP circuits can be used to perform logic operations is shown.

6.2.3. Understanding and using JoSim to simulate working logic gates

By successfully generating an output from JoSim that shows the function of an AQFP gate is correct an understanding of JoSim is shown.

6.2.4. Compiling these logic gates to make complex multi-gates

6.2.5. Using Klayout to design the physical AQFP cells

6.2.6. Using InductEx to extract the inductance's from the KLayout model

6.2.7. Rerunning JoSim to ensure the values obtained from the physical model are correct

6.3. Conclusion

In conclusion it is possible to recreate basic logic gates using AQFP logic and to use these gates to create more complex multi-gates such as the Half Adder and the Full Adder. There are a few constraints in AQFP logic that CMOS does not have, such as the need to clock every signal at the same time and path splitting.....

However, AQFP logic is still new compared to CMOS technology and with the successful implementation of AQFP buffers and constant gates, and a deep understanding of the underlying principles of majority logic, it becomes evident that these circuits have the potential to revolutionize logical operations in the realm of quantum computing. With the combination of low power consumption and high-speed operation in superconducting technology, AQFP logic opens up exciting possibilities.

Bibliography

[1] N. TAKEUCHI, T. YAMAE, C. Ayala, H. Suzuki, and N. Yoshikawa, "Adiabatic quantum-flux-parametron: A tutorial review," *IEICE Transactions on Electronics*, vol. E105.C, no. 1, pp. 1–30, 2022.

[2] M. Z. Y. Xie, Z. Li, P. Dong, G. Yuan, O. Chen, M. Pedram, and Y. Wang, "A life-cycle energy and inventory analysis of adiabatic quantum-flux-parametron circuits," https://arxiv.org/pdf/2307.12216.pdf#:~:text=Our%20analysis%20shows%20that%20while,comparison%20with%20the%20CMOS%20counterpart., JUL 2023, (Accessed on 08/08/2023).

[3] C. T. Y. N. University, "Coldflux logic cell library for mit-ll sfq process: Part 1 - aqfp," JUL 2022, (Accessed on 07/22/2023).

[4] Y. Yamanashi, H. Imai, and N. Yoshikawa, "Influence of magnetic flux trapped in moats on superconducting integrated circuit operation," *IEEE Transactions on Applied Superconductivity*, vol. 28, no. 7, pp. 1–5, Oct 2018.

[5] C. J. Fourie, N. Takeuchi, K. Jackman, and N. Yoshikawa, "Evaluation of flux trapping moat position on aqfp cell performance," *Journal of Physics: Conference Series*, vol. 1975, no. 1, p. 012027, jul 2021. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1975/1/012027

[6] L. KELLEY, "What are superconducting quantum interference devices or squids? - owlcation," https://owlcation.com/stem/What-Are-Superconducting-Quantum-Interference-Devices-or-SQUIDs, JUL 2022, (Accessed on 07/22/2023).

Appendix A

Project Planning Schedule

This is an appendix.

Appendix B

Outcomes Compliance

This is another appendix.