CLOCK BUFFER IC WITH DYNAMIC IMPEDANCE MATCHING AND SKEW COMPENSATION

by

Aris Balatsos

A thesis submitted in conformity with the requirements for the degree of Masters of Applied Science
Graduate Department of Electrical and Computer Engineering
University of Toronto

© Copyright by Aris Balatsos (1998)
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author’s permission.

L’auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L’auteur conserve la propriété du droit d’auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-34127-5
CLOCK BUFFER IC WITH DYNAMIC IMPEDANCE MATCHING AND SKEW COMPENSATION

Masters of Applied Science 1998
Aris Balatsos
Department of Electrical and Computer Engineering
University of Toronto

Abstract

This thesis describes a novel approach for distributing low skew clock signals across large digital systems independent of environmental and process variations. The technique is integrated into a multi-output clock buffer circuit that can handle a scalable number of clock loads in a point-to-point configuration. The circuit contains an impedance-locked loop that continuously monitors the impedance of output clock traces and adjusts driver impedance for optimal matching. Reflections from clock destinations contain propagation delay information that is used by delay-locked loops to continuously adjust the departure time of outgoing clocks to ensure low relative skew at clock destinations. A theoretical evaluation of the deskewing technique is presented along with results from a prototype clock buffer device. Testing indicates operational frequencies as high as 50MHz with 40cm clock lines, and skew on the order of 600-800ps as measured at the clock destinations.
Acknowledgments

There are many people and organizations who deserve to be acknowledged for their generous support and encouragement throughout the course of this thesis.

I am indebted to my thesis supervisor, David Lewis, for his expert guidance and assistance that helped make this thesis a reality. His knowledge and experience contributed immensely to the positive outcome of this thesis.

A sincere thank you is directed to the National Science and Engineering Research Council (NSERC) who provided financial assistance for this project. My appreciation also goes out the Canadian Microelectronics Corporation (CMC) for implementing prototype circuits, and for supporting Canadian university research in VLSI design.

I would also like to acknowledge Marcus van Ierssel for the excellent job he did producing the circuit board used to test prototype chips. In addition, ATI Technologies Inc. kindly provided access to a high performance oscilloscope and logic analyzer that greatly facilitated testing. Jaro Pristupa provided invaluable help when computer network and CAD tool problems surfaced.

A very big thank you goes out to my parents and family for their ceaseless encouragement and support throughout my life. I am also grateful for the friendship and assistance of my associates: Jamil Ahmed, Silas Li, Vineet Joshi, Derek Tam, Luigi Dipepe, Gary Liu, and the many other helpful members of the EECG/VRG groups at the University of Toronto.
Table of Contents

ABSTRACT ........................................................................................................I
ACKNOWLEDGMENTS ..................................................................................II
TABLE OF CONTENTS ..................................................................................III
LIST OF TABLES ..............................................................................................VII
LIST OF FIGURES ............................................................................................VIII

CHAPTER 1
INTRODUCTION..................................................................................................1
  1.1 Current Focus ............................................................................................2
  1.2 Thesis Overview .......................................................................................4

CHAPTER 2
BACKGROUND..................................................................................................5
  2.1 Problem Description ....................................................................................5
  2.2 Previous Work ............................................................................................8
    2.2.1 Static Matching ...................................................................................8
    2.2.2 Dynamic Matching ..............................................................................9
      2.2.2.1 On-chip PLL’s ..............................................................................9
      2.2.2.2 On-chip DLL’s ............................................................................10
    2.2.3 System Wide Approaches ..................................................................10
    2.2.4 Dedicated Wire Feedback Approaches ..............................................11
    2.2.5 Avoiding Dedicated Feedback Paths ................................................12
  2.2.3 Impedance Matching ............................................................................13
  2.3 Thesis Goals .............................................................................................14
  2.4 Transmission Line Fundamentals .............................................................15
    2.4.1 Modeling Transmission Lines ............................................................16
4.4.2 Single-to-Differential Converter ................................................................. 85
4.4.3 Delay Lines .................................................................................................. 85
  4.4.3.1 Fixed Delay Lines .................................................................................. 86
  4.4.3.2 Tapped Delay Lines .............................................................................. 86
  4.4.3.3 Tap Control Word Update Timing .......................................................... 88
  4.4.3.4 Sampled Delay Lines ............................................................................ 90
4.5 Phase Detector and DLL Tap Controller ...................................................... 91
  4.5.1 Tap Controller .......................................................................................... 94
4.6 Propagation Delay Measurement .................................................................. 97
  4.6.1 Leading One Detector ............................................................................... 99
4.7 Automatic Impedance Matching .................................................................. 100
  4.7.1 Automatic Impedance Matching Overview .............................................. 101
  4.7.2 ICW Bus Updates .................................................................................... 103
  4.7.3 Architectural Details ................................................................................ 104
  4.7.4 Fixed Delay Sizing .................................................................................. 107
    4.7.4.1 Future Considerations Regarding Impedance Sampling ................. 109
  4.7.5 Reset State Determination ...................................................................... 110
  4.7.6 Output Driver .......................................................................................... 112
4.8 Input/Output Circuitry .................................................................................... 115
  4.8.1 Input Demultiplexor ................................................................................ 115
  4.8.2 Output Multiplexor .................................................................................. 115

CHAPTER 5
VLSI IMPLEMENTATION .................................................................................... 118
5.1 Basic Circuit Elements .................................................................................. 118
  5.1.1 Delay Elements ....................................................................................... 118
  5.1.2 Sampling and Storage Elements .............................................................. 121
  5.1.3 Comparators ............................................................................................ 123
    5.1.3.1 Voltage Threshold Generation .......................................................... 126
  5.1.4 Output Driver ........................................................................................... 127
    5.1.4.1 Output Driver Latchup Considerations .............................................. 128
  5.1.5 Tapped-Delay Line Selection Circuits ...................................................... 129
5.2 Layout and Floorplanning ............................................................................. 130
  5.2.1 Layout Strategy ....................................................................................... 130
  5.2.2 Clock Bay Floorplan ............................................................................... 131
List of Tables

TABLE 1. Reflection Coefficient Terms ..........................................................................................38
TABLE 2. Convolutions Required For The Computation of Equations (3.26) and (3.27) ..................................................44
TABLE 3. Output Multiplexor’s Input Signal Descriptions ..........................................................116
TABLE 4. Differential Delay Element Performance (45°C) ......................................................119
TABLE 5. CBC Pin Assignments ..................................................................................................137
TABLE 6. Current Estimates For The CBC ................................................................................142
TABLE 7. Test Board Clock Trace Lengths ..................................................................................148
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Effect of clock skew in a typical digital circuit</td>
<td>6</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Typical PLL architecture</td>
<td>9</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Typical DLL architecture</td>
<td>10</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Dedicated One-Wire Feedback System</td>
<td>11</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Dedicated Two-Wire Feedback System</td>
<td>12</td>
</tr>
<tr>
<td>Figure 6</td>
<td>General transmission line model</td>
<td>17</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Lossless transmission line model</td>
<td>17</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Example lossless transmission line of length L terminated by a resistive load</td>
<td>19</td>
</tr>
<tr>
<td>Figure 9</td>
<td>Voltage versus time plots for two points on the transmission line of Figure 8</td>
<td>22</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Load voltage waveform for the same transmission line configuration as Figure 8, but with the load resistance changed to 3Z_0</td>
<td>24</td>
</tr>
<tr>
<td>Figure 11</td>
<td>(a) Series terminated transmission line. (b) Parallel terminated transmission line</td>
<td>25</td>
</tr>
<tr>
<td>Figure 12</td>
<td>(a) Single source driving multiple parallel terminated lines. (b) Single source driving multiple series terminated lines</td>
<td>29</td>
</tr>
<tr>
<td>Figure 13</td>
<td>(a) Voltage transient response for the load end of the transmission line of Figure 13. (b) Transient response for the source end of the transmission line</td>
<td>32</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Series terminated transmission line model for the CBC outputs</td>
<td>37</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Thévenin equivalent circuit model</td>
<td>37</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Plot of the source and load waveforms for the transmission line model using ramp inputs</td>
<td>46</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Transmission line model including up to 5th order reflections</td>
<td>47</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Comparison of the transmission line models based on step and ramp inputs</td>
<td>49</td>
</tr>
<tr>
<td>Figure 19</td>
<td>Linear approximations used for the source (a), and load waveform (b)</td>
<td>51</td>
</tr>
<tr>
<td>Figure 20</td>
<td>Load waveform and its constituent terms for the step-input transmission line model</td>
<td>53</td>
</tr>
<tr>
<td>Figure 21</td>
<td>Transmission line model used for the HSPICE simulations</td>
<td>60</td>
</tr>
<tr>
<td>Figure 22</td>
<td>Propagation delay estimation error as predicted by the ramp, step, and piecewise linear models for the case where C_S is fixed at 10pF</td>
<td>63</td>
</tr>
</tbody>
</table>
Figure 24. Propagation delay estimation error as predicted by the ramp, step, and piecewise linear models for the case where $C_S$ equals $C_L$..............64
Figure 25. Propagation delay estimation error as predicted by the HSPICE model for the case where $C_S$ is fixed at 10pF, and where $R_S$ is matched to characteristic impedance.............................................65
Figure 26. Propagation delay estimation error as predicted by the HSPICE model for the case where $C_S$ equals $C_L$, and where $R_S$ is matched to the characteristic impedance.............................................66
Figure 27. Propagation delay estimation error as predicted by the HSPICE model for the case where $C_S$ is fixed at 10pF and where $R_S$ equals 90% and 110% of $Z_0$.................................................................67
Figure 28. Propagation delay estimation error as predicted by the HSPICE model for the case where $C_S$ equals $C_L$, and $R_S$ equals 90% and 110% of $Z_0$.................................................................68
Figure 29. Propagation delay versus transmission line length for typical "slow", "medium", and "fast" transmission line configurations (analytic model).................................................................70
Figure 30. Propagation delay versus transmission line length for typical "slow", "medium", and "fast" transmission line configurations (HSPICE model).................................................................71
Figure 31. High-level structure of the CBC.................................................................73
Figure 32. DLL architecture of the CBC.................................................................75
Figure 33. Example lead-lag phase detector waveforms........................................75
Figure 34. Clock bay block diagram.................................................................77
Figure 35. Idealized waveforms for the output of TDL1, the clock bay, and the low threshold comparator.................................................................79
Figure 36. Clock bay DLL and its associated delays........................................81
Figure 37. Delay element cell and its relationship to neighbouring delay elements. 85
Figure 38. Single-to-Differential converter.................................................................85
Figure 39. TDL2 delay line arrangement.................................................................87
Figure 40. Tap selection circuit used in the TDL1 delay line........................................88
Figure 41. Example circuit consisting of a 32 element delay line and its tap selection structure.................................................................89
Figure 42. Phase detector schematic.................................................................92
Figure 43. Tap controller update FSM.................................................................93
Figure 45. Tap controller schematic.................................................................96
Figure 46. High and low threshold comparator outputs ($V_{low}$ and $V_{high}$) for a rising transition on the clock bay output ($V_{src}$).................................................................97
Figure 47. Simplified block diagram for the propagation delay measuring circuit...98
Figure 48. Leading one detector and its surroundings........................................100
Figure 49. Transient voltage step voltage for different output configurations.......101
Figure 50. Simplified automatic impedance matching circuit block diagram........102
Figure 51. ICW bus update example.........................................................104
Figure 52. Impedance detector and impedance controller block diagram........105
Figure 53. Simulations of various “fast” and “slow” transmission line
configurations.................................................................108
Figure 54. ILL voltage step level after reset..............................................111
Figure 55. Clock Bay Output Driver.........................................................113
Figure 56. Output driver resistance versus impedance control word............114
Figure 57. One of six 4:1 multiplexors used in the output multiplexor...........116
Figure 58. Differential delay element......................................................119
Figure 59. Delay element layout.............................................................120
Figure 60. Static latch schematic..............................................................122
Figure 61. (a) N-type and (b) P-type TSPC latches.................................122
Figure 62. CMOS differential amplifier.....................................................124
Figure 63. Layout of CMOS differential amplifier......................................125
Figure 64. Complementary self-biased differential CMOS amplifier (CDSA)
schematic.................................................................126
Figure 65. Output driver layout.................................................................128
Figure 66. Clock bay floorplan.................................................................131
Figure 67. Floorplan for the propagation delay estimator and the TDL2 delay line. 133
Figure 68. Floorplan of the phase detector and TDL1 tap controller.............134
Figure 69. Floorplan of the impedance control and comparator circuits.........135
Figure 70. CBC floorplan showing all three clock bays..............................137
Figure 71. CBC layout...........................................................................138
Figure 72. CBC test board layout..............................................................148
Figure 73. Source and load waveforms for clock bay outputs during ILL and DLL
reset.................................................................153
Figure 74. Source and load waveforms during ILL lock and DLL reset.........153
Figure 75. Source and load waveforms after ILL and DLL lock...............154
Figure 76. Averaged (16X) source and load waveforms after ILL and DLL lock...154
Figure 77. Load waveforms before and after DLL lock
(in 100X persistence mode)..........................................................156
Figure 78. ILL impedance control words before and after ILL lock..............157
Figure 79. TDL1 tap control word for clock bay 0 (in test circuit 2) before and after
DLL lock...........................................................................158
Chapter 1
Introduction

Most digital circuits today are designed as synchronous systems, which means they require a precisely controlled clock signal to regulate their internal operation. However, as clock frequencies increase, it becomes more and more difficult to distribute a suitably synchronized clock to all parts of a digital system. This stems from the fact that at high frequencies the wires through which clock signals travel can no longer be thought of as perfect conductors.

At high frequencies clock traces must be thought of as transmission lines. Signals in transmission lines have a finite propagation speed which leads to non-zero delay. In addition, transmitted signals experience reflections that can lead to oscillation and distortion unless a transmission line is terminated properly.

The end result is that clock signals reaching one section of a digital circuit can arrive seriously skewed relative to clocks in another part of a circuit. This clock desynchronization is commonly known as clock skew, and, if it is large enough, can cause a digital circuit to fail. As explained in section 2.1, this failure occurs because of hold or
setup time violations at the flip-flops of a digital system. Synchronous circuits can be designed to tolerate some degree of clock skew, but, in general, clock skew will cause a performance hit, or at worst failure.

Circuit designers have come up with many ways to reduce clock skew. Perhaps the simplest approach is to route clock signals along equal length clock traces with similar clock loads, and hope that wire delays and distortions affect all paths equally so that the overall clock skew is kept low.

This approach works well for low clock frequencies, but, at higher speeds, process, voltage, temperature and loading (PVTL) differences between clock paths become a factor, and affect clock traces in ways that cannot be predicted or controlled easily.

To reduce clock skew even further, designers have had to become more creative. They have introduced circuits that use feedback to maintain the synchronization of clock signals even when the delay of clock paths is gradually changing. Initial approaches developed on-chip phase-locked loops, and delay-locked loops that reduce the clock skew associated with bringing clock signals into an IC. Recently designers have developed custom clock chips containing feedback circuits that attempt to synchronize clocks signals system wide not just within one IC. In all cases, feedback permits tighter control of clock skew.

1.1 Current Focus
This thesis focuses on the design of a clock buffer chip that attempts to deskew clock signals at the system level up to the input of any target IC requiring a clock input. Like earlier clock buffer circuits, the intent is to use feedback to control clock skew; however, the design described here is unique in the way it obtains its feedback information. Reflections in clock traces are used to estimate the propagation delay along a clock trace. This information is then used to deliberately pre-skew outgoing clock signals so they
arrive synchronized at their destinations. The key idea with this approach is that no dedicated return path is required since reflections themselves provide the necessary feedback.

Furthermore, before such a reflection based approach can be used to deskew clock signals, proper termination of each clock trace is needed. Proper termination (also known as impedance matching) leads to controlled reflections that can be measured easily to obtain the desired propagation delay information. Thus an important component of this thesis is an automatic means of performing impedance matching.

The basis of a proposed impedance matching technique once again relies on reflections to determine the proper impedance level. An impedance-locked loop is developed that is designed to continuously update the termination impedance to track impedance changes due to temperature and voltage shifts.

Although the idea of using reflections for impedance matching and for deskewing is not new ([9], [10], [13]), the circuits presented in this thesis are novel in that they attempt to continuously monitor and dynamically adjust to changes in impedance and signal skew. Previous approaches perform these operations only once during initialization, and require the assistance of an external control computer. They have no capacity to track changes in impedance or skew during normal circuit operation. In addition, this reflection based technique has not been implemented before in any clock buffer circuits.

The design presented in this thesis describes a scalable clock buffer chip (CBC) implemented with three clock outputs, and capable of deskewing signal traces of half a meter or more in length. Initial testing of several prototype chips received from the Canadian Microelectronic Corporation (CMC) indicates that the circuit operates at up to 50MHz with clock trace lengths of 40cm. Clock skew appears generally limited to 700ps or less at clock destinations.
1.2 Thesis Overview

Chapter 2 begins by providing a more detailed look at the problem of clock skew, and the range of clock frequencies where clock skew becomes significant. It also delves more deeply into the previous work undertaken regarding automatic impedance matching and clock synchronization. Next, the specific goals of this thesis are presented, followed by a brief tutorial on transmission line fundamentals (which knowledgeable readers can skip).

Chapter 3 focuses on a theoretical analysis of the technique of obtaining propagation delay information from signal trace reflections. Analytical and piece-wise linear models are developed for transmission line configurations with both ramp and step inputs. Detailed error predictions are presented for these models, and they are compared with results from HSPICE [33] circuit simulations.

The design of the clock buffer chip (CBC) built for this thesis is presented in chapters 4 and 5. Chapter 4 describes the block and gate-level design of the CBC, while chapter 5 presents the transistor and layout level issues. Chapter 5 includes a description of the simulation and testing performed on the design, and also contains estimates of key performance parameters.

Finally, chapter 6 describes test results obtained from a series of prototype circuits obtained from CMC. Measured maximum operating speeds, deskewing performance, and other operating parameters are presented in this final section.
Chapter 2
Background

This chapter contains background information regarding the motivation and goals of this thesis. Section 2.1 describes the problem of clock skew and the limitations it imposes on system performance. Next, section 2.2 describes the previous work done in addressing clock skew issues. The goals of this thesis are presented in section 2.3, followed by a tutorial on transmission line fundamentals in sections 2.4 and 2.5.

2.1 Problem Description

The clock buffer chip (CBC) built for this thesis addresses the problem of clock synchronization in digital systems. Greater attention has been paid to this problem in recent years as clock skew has become an important factor that limits the performance of high speed digital systems.

Clocks derived from the same reference source experience some degree of skew relative to each other when distributed across a digital system. Skew is generated by unequal
delays along the signal paths between where a clock is generated and the multiple places where it is used.

![Diagram](image)

**Figure 1.** Effect of clock skew in a typical digital circuit. (Figure adapted from [1], p.324).

Clock skew can lead to both setup and hold time violations. Consider the circuit of Figure 1(a), where the clock is shown flowing in the direction of data flow. Delays in the clock path lead to skewed versions of the system clock arriving at the two flip-flops. If delay₂ is greater than the sum of the clock-to-Q delay of FF₁, the logic delay, and the setup time of FF₂, then a hold time violation will occur. As shown in Figure 1(b), FF₂ will sample data from the current clock cycle and miss data for the previous cycle! This situation can only be corrected by adding delay to the data path from FF₁ to FF₂ (which may require an increase in the cycle time), or by reducing clock skew.

If the clock signal is routed in a direction opposite to data flow, then clock skew will probably not lead to hold time violations; however, setup time violations at FF₂ are a strong possibility since clk₂ will arrive early relative to clk₁. The clock cycle time may
have to be extended to prevent this problem from occurring. This will reduce system performance.

Clock delay is generated from three sources:

- Delay due to buffering at the clock source.
- Propagation delay along the signal traces between the clock source and destination integrated circuits (ICs).
- Delay caused by sensing, buffering, and distributing the clock signal to the flip-flops inside the destination circuit.

In each case, delay is sensitive to variations caused by process, voltage, temperature, and loading (PVTL). In addition, signal propagation delays depend on the distance separating the clock source from the target IC, and this distance can vary considerably between different ICs on a circuit board. Variations caused by PVTL and distance, can lead to clock skew so large that it severely affects today’s high speed systems, and cannot be ignored.

According to [2], integrated circuit (e.g. I/O buffer, clock-tree, and internal logic) delays can vary by a factor of 2:1 or more across typical PVTL variations. This rather large change is to be expected since dozens of variables influence the ultimate performance of a given IC. Small changes (10% or less) in each of the variables have a large combined effect ([1], pp. 243-248).

As described, in section 2.4, transmission line effects, which include propagation delays, reflections, and distortion, have significant impact on the performance of high clock rate digital systems. Signal propagation speeds are close to the speed of light, and are typically 20cm/ns to 15cm/ns. This means that a system containing logic distributed across a typical 50cm circuit board experiences approximately 3ns of clock skew. Even if all logic is located at identical distances from the clock source, skew problems remain; the expected variation in the propagation velocity from trace to trace leads to skew that grows by
8ps/cm [12]. So, even with equal length traces, skew can reach a few hundred picoseconds over small distances.

In addition, reflections on poorly terminated signal traces, can lead to delays that are significantly longer than those expected from the propagation delay along a signal trace (see section 2.4.4). Proper termination is required on all clock lines to ensure that these reflections are controlled. PVTL variations in the transmission line parameters make proper termination more difficult.

With all the above sources of clock skew present in digital systems, skew can easily reach 3-7ns [2],[12] and this forms a significant fraction of the clock periods of today’s high speed systems (running at operating frequencies from 30-300 MHz). For practical reasons, the fraction of a clock period that is devoted to providing a safety margin against clock skew should not be larger than 10% to 20%. This motivates work into circuits that can synchronize clocks and reduce clock skew to low levels.

### 2.2 Previous Work

Two major approaches are used to deal with clock skew. The first is referred to as static matching, and the second as dynamic matching.

#### 2.2.1 Static Matching

Static matching was the first technique developed to control clock skew. It attempts to route clock signals across equal delay paths from the clock driver to the clock’s point of use (POU). This requires matching pad and clock buffer delays across all ICs by using identical driver circuits. It also requires equal length signal traces (i.e. radial clock routing) and tightly controlled impedances.
The success of such a method is limited by PVTL variations as discussed in the previous section. In addition, routing equal length clock traces is difficult and can waste board area. Static matching still serves a purpose, but it is not adequate for very high speed digital systems.

2.2.2 Dynamic Matching

Dynamic matching is a more recent approach in dealing with clock skew. It uses feedback to align clocks at their point of use to some reference clock. The major differences between each form of dynamic matching are the types of synchronization circuits used, and the scale over which they are applied (i.e. locally for one IC, or globally across a digital system).

2.2.2.1 On-chip PLLs

One of the first dynamic matching approaches for reducing clock skew targeted the third type of clock delay mentioned in section 2.1. That is, the delay associated with buffering and distributing the clock inside an IC to its POU. This approach uses an on-chip phase-locked loop (PLL) for clock synchronization as shown in Figure 2 [3], [4], [5].

![Figure 2. Typical PLL architecture](image)

The PLL synthesizes a new clock from an oscillator that is frequency and phase aligned with the external clock signal. It compares the synthesized clock at the output of the IC's
clock tree with the external clock, and adjusts the synthesized clock until the two align. In this way, the delay associated with the IC's input buffer and clock tree is avoided.

2.2.2.2 On-chip DLLs

In recent years, a new type of phase aligning circuit has been used on-chip to minimize the clock skew introduced while delivering an external clock signal to its POU inside an IC. This approach uses a delay-locked loop (DLL) instead of a PLL to perform synchronization, as shown in Figure 3 [6], [7], [8]. The advantage of the DLL over the PLL stems from reduced sensitivity to noise, and faster lock acquisition time [6], [8].

![Figure 3. Typical DLL architecture](image)

A fixed frequency reference clock is needed by a DLL since DLLs perform only phase locking. The DLL works by introducing delay into the input clock until the clock at the POU aligns with an edge of the external clock. It adjusts a variable delay until alignment is achieved.

2.2.2.3 System Wide Approaches

On-chip skew reduction techniques work very well at reducing skew from an IC's clock-input to the clock's POU inside the IC. In fact, nearly all modern ICs that perform high speed I/O employ some sort of on-chip DLL or PLL for clock synchronization. However, this approach does not address the skew problems associated with board level clock propagation delays, and clock driver delay variations.
To tackle this problem, dedicated multiple-output clock buffer chips were developed (see below) that obtain feedback from different POUss in a digital system. This feedback information is used to synchronize the clocks at multiple distributed POUss to a single master clock.

2.2.2.4 Dedicated Wire Feedback Approaches

A typical one-wire feedback system is shown in Figure 4 [9]. The clock buffer measures the total round trip delay via the difference between the reference clock and the return clock. This delay is equal to $2(T_{\text{delay}} + T_{\text{prop}} + 2T_{\text{buffer}})$. By dividing this delay in two, the clock buffer can determine the one-way delay to the input of the target IC's clock tree. $T_{\text{delay}}$ can then be increased or decreased to adjust the arrival time of the clock at the clock tree, until the clock is phase-aligned to the reference clock.

Such a system can deskew clock signals up to the input of the clock tree inside the target IC [9], [11]. However, it cannot deskew signals to the point of use inside the IC. For this, an on-chip DLL or PLL is still needed.

A two-wire feedback system such as that presented in Figure 5 [12] can deskew signals up to the point of use inside the target ASIC (without using an on-chip DLL or PLL). The
time difference between \textit{feedback1} and \textit{feedback2} returns information on the clock distribution delay \textit{inside} the ASIC, while \textit{feedback1} provides information on the delay up until the clock-tree of the target IC. In [12], independent DLLs use this dual feedback to reduce the skew between the reference clock and the clock at each POU to 1ns or less.

Note that using only the \textit{feedback2} signal is not enough to properly deskew signals at the POU. The reason is that, by halving the round-trip delay time of \textit{feedback2}, one obtains \((T_{\text{prop}}+2T_{\text{buffer}}+T_{\text{clocktree}}/2)\) which is not the delay to the POU. Instead, the delay regulator must compute and compensate for a delay of \((T_{\text{prop}}+2T_{\text{buffer}}+T_{\text{clocktree}})\). This can only be accomplished by using a two wire feedback system.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure5}
\caption{Dedicated Two-Wire Feedback System. The delay between the two feedback signals contains information on the clock tree delay \textit{inside} the target IC. The Delay Regulator, which is essentially a DLL, delays the reference clock so the clock arriving at the POU is phase aligned to the reference clock.}
\end{figure}

\subsection{2.2.2.5 Avoiding Dedicated Feedback Paths}

A technique presented in [9] and [10] describes a novel approach that allows feedback information to be obtained without using dedicated feedback paths. (Incidentally, this technique was developed independently for this thesis, but not before these earlier papers were published).
This approach uses transmission line reflections to estimate the delay between the clock buffer and the target IC. The interval between a transition at the output of the clock buffer, and the return reflection from the target IC is measured to obtain the round trip delay. In effect, the outgoing clock is used as a test signal to obtain propagation delay information. This type of approach is comparable to the one-wire feedback system discussed earlier. (More will be said about this technique in section 2.5 and Chapter 3).

The work presented in [9] and [10] does not discuss a clock buffer circuit, but instead describes special I/O pads that uses this delay estimation technique to statically deskew outgoing signals. Each pad is configured via a scan-based Test-Access Port. An external controller initially runs each pad through its paces to adjust the output impedance settings, and to measure and compensate for propagation delays.

This I/O pad approach is area efficient, high-speed, and flexible, but it has several drawbacks. It requires a sophisticated off-chip controller to configure the pads, and a lengthy configuration phase (needing tens to hundreds of microseconds per pad). There is little opportunity for adjusting the pad configuration on-the-fly to compensate dynamically for variations in delay and transmission line impedance. This approach falls somewhere in between static and dynamic matching as the effects of changes in voltage and temperature cannot be tracked easily.

2.2.3 Impedance Matching

Matching of driver impedance to transmission line impedance is necessary to control reflections. On-chip output impedance matching circuits are presented in several papers [13], [14], [15]. They all use an output driver with different parallel pull-up and pull-down paths that can be enabled independently to change the output impedance. The method in [15] matches the output impedance to that of external resistors connected to several
unused output pads. It assumes that all outputs have the same impedance and that this impedance does not change with time.

Reference [13] describes an effective impedance matching technique that is fully automatic. It was implemented in the I/O pad deskewing circuit discussed in the previous section. Once again, this circuit uses transmission line reflections to determine the degree of impedance matching. It also requires a sophisticated off-chip controller to determine the proper impedance settings which is a slow, but nevertheless flexible way of performing adjustments. The circuit does not track impedance settings on-the-fly, but must be periodically stopped and reinitialized.

2.3 Thesis Goals

There are two primary goals for this thesis. The first is to characterize the effectiveness of using measurements of transmission line reflections for determining an estimate of signal propagation delay. There is little information in the literature regarding the expected performance of this promising technique. A performance analysis is presented in Chapter 3.

The second goal is to construct a device that addresses clock skew issues at the system level using a dynamic matching approach. The following is a summary of the goals of the device:

- Dynamic measurement of the clock buffer-to-target propagation delay using transmission line reflections.
- Independent, dynamic matching of output driver impedance to clock trace impedance in order to control reflections.
- Integration of the above techniques into a multi-output, point-to-point clock buffer that uses a DLL architecture to synchronize destination clocks to a reference clock.
The first two techniques have not been applied in any clock buffers to date. The development of suitable circuits and architectures for realizing them is desirable due to their novel nature. Since the propagation delay estimation technique does not require a return path or special circuitry at the destination IC, such a clock buffer will be an attractive circuit for controlling skew in many applications. Also, the dynamic nature of the clock buffer ensures that PVTL variations are properly tracked.

2.4 Transmission Line Fundamentals

Before discussing the design of the clock buffer chip, a review of transmission lines is in order. This discussion builds upon traditional (lumped-parameter) circuit analysis techniques, and no specialized knowledge is necessary other than basic circuit theory and the Laplace transform. Only the details essential to understanding this thesis will be covered.

As mentioned in section 2.1, all signal paths are transmission lines. Signals travel down signal traces at close to half the speed of light, and experience phase delay, reflections, and possibly distortion. Propagation delays across 50cm traces can easily reach 3ns.

In the past, transmission line effects could be ignored by logic designers since the operating speeds were so low that the wiring delays were negligible compared to the oscillation period of the circuits, and reflections tended to be buried in slow edge transitions. For example, previous designs using Transistor-Transistor Logic (TTL) operate with gate delays of 15-25ns [16], and clock periods on the order of 100ns, so the assumption that wires are perfect zero-delay conductors was a relatively safe one.

The size of typical digital logic designs has stayed nearly constant over the past several decades with maximum signal path lengths ranging up to 1m; however, the switching
speeds and edge transition speeds have climbed steadily. Today, switching speeds in the 30-300 MHz range are common place, as are sub-nanosecond edge transitions times.

Transmission line effects become important when the wavelength of the input signal begins to approach the length associated with a circuit’s wiring. Wavelength, \( \lambda \), is related to the input frequency, \( f \), by the speed of light, \( c \), using equation (2.1).

\[
\lambda = \frac{c}{f} \tag{2.1}
\]

For example, at frequencies of 500MHz, signal wavelengths are on the order of 60cm, and transmission line effects cannot be ignored. A signal sent from a source to a destination located a mere 30cm away will arrive phase shifted by 180 degrees compared to a signal arriving at a destination adjacent to the source.

At lower operating speeds the effects are not quite as dramatic, yet they take a toll on performance. For instance, a 40MHz signal has a period of 25ns, so wire delays of 3ns are significant. In addition, if signal paths are not terminated properly, reflections can lead to oscillations that do not settle until much longer than the 3ns wire delay, as explained in section 2.4.4.

2.4.1 Modeling Transmission Lines

Transmission lines cannot be treated as lumped parameter circuits, so a distributed approach must be used. Distributed parameters are expressed as a function of length. A frequently used model for transmission lines is shown in Figure 6 ([17], ch.1). It consists of many small segments of length \( \Delta l \) containing lumped parameter elements. For this model to be valid, \( \Delta l \) must be much smaller than the wavelength of the applied signal.
$L_o$ is a distributed inductance in henries per meter, while $C_o$ is a distributed capacitance in farads per meter. $R_o$ is a distributed series resistance in ohms per meter, and $G_o$ is a distributed shunt conductance in mhos per meter.

\[
\begin{array}{c}
\text{R}_o\Delta l & \text{L}_o\Delta l & \text{R}_o\Delta l & \text{L}_o\Delta l & \text{R}_o\Delta l & \text{L}_o\Delta l \\
\text{C}_o\Delta l & & \text{G}_o\Delta l & \text{C}_o\Delta l & & \text{G}_o\Delta l \\
\end{array}
\]

\[\Delta l\]

*Figure 6. General transmission line model.*

If $R_o$ and $G_o$ are assumed to be zero, then the model reduces to that of an *ideal lossless* transmission line, as shown in Figure 7. This approximation is valid for short lines since they have small losses.

\[
\begin{array}{c}
\text{L}_o\Delta l & \text{L}_o\Delta l & \text{L}_o\Delta l \\
\text{C}_o\Delta l & \text{C}_o\Delta l & \text{C}_o\Delta l \\
\end{array}
\]

\[\Delta l\]

*Figure 7. Lossless transmission line model.*

### 2.4.2 Impedance of a Transmission Line

A common approach to understanding transmission lines is to derive and solve the differential equations governing the above models, and subsequently extract fundamental properties from these solutions. However, much can be learned, without involving differential equations, by simply applying basic AC analysis techniques.
One of the most important points that such an analysis can highlight is the fact that the impedance looking into a ladder network of inductive and capacitive elements, such as that shown in Figure 7, looks like a pure resistance ([17], p.7). This resistance is known as the characteristic impedance, $Z_0$, of the transmission line, and it plays an important role in analyzing transients on transmission lines.

The characteristic impedance is given by equation (2.2). $Z_0$, equals a purely resistive impedance formed by the distributed inductance and capacitance of the transmission line.

$$Z_0 = \sqrt{\frac{L_o}{C_o}} \quad (2.2)$$

It can also be shown that a similar result holds for transmission lines having a small distributed series resistance ([17], p.10). Typical values for the characteristic impedance vary from 50Ω to 125Ω ([18], p. 10).

### 2.4.3 Propagation Velocity of Ideal Transmission Lines

The propagation velocity along an ideal lossless transmission line is given by equation (2.3) ([17], p. 13).

$$v = \frac{1}{\sqrt{L_o C_o}} \quad (2.3)$$

This velocity is independent of the frequency applied to the transmission line, and is generally close to the speed of light in a vacuum, $c$. The exact propagation velocity depends on the physical structure of the transmission line, and the dielectric permittivity and magnetic permeability of the dielectric separating the signal wire from the reference wire of the transmission line. These factors determine the precise values for the distributed inductance and capacitance used in equation (2.3).
Typical values of the propagation velocity range from \(2/3\) to \(1/2\) the speed of light, which equates to propagation delays of 5 ns/m to 6.5 ns/m ([19], Appendix A).

### 2.4.4 Reflections on Transmission Lines

In reality, infinitely long transmission lines, such as those discussed in the previous sections, are not possible. Signals travelling down finite length transmission lines undergo reflections at each end, and these reflections must be dealt with to avoid excessive oscillations, delays, and overshoots/undershoots.

*Figure 8. Example lossless transmission line of length \(L\) terminated by a resistive load. The total propagation delay across the transmission line, \(T_{pd}\), is obtained by dividing the distance, \(d\), by the propagation velocity given by equation (2.3).*

For example, consider the uniform ideal transmission line shown in Figure 8. Initially, it is in steady state with zero volts across it, and no current flowing. At time zero, a voltage step of magnitude \(V_S\) begins travelling from the source to the load together with a current step of magnitude \(I_S = V_S/Z_0\) (determined by the characteristic impedance of the line).

After a time, \(T_{pd}\), the step arrives at the load. Since the load is not matched to the characteristic impedance of the transmission line, Ohm's Law cannot be satisfied at the load end without a reflection of voltage and current. An expression for the magnitude of the reflected signal can be derived as follows.
The incident current, $I_S$, the load current, $I_L$, and the reflected current, $I_R$, obey the law of conservation of current at a node.

\[ I_S = I_L + I_R \]  \hspace{1cm} (2.4)

Applying Ohm’s Law to each current term in equation (2.4) leads to equation (2.5) where $V_S$ is the incident voltage, $V_R$ is the voltage of the reflected wave, and $V_L$ is the voltage across the load after the reflection.

\[ \frac{V_S}{Z_o} = \frac{V_L}{R_L} + \frac{V_R}{R_L} \]  \hspace{1cm} (2.5)

Noting that $V_L$ is simply the sum of $V_S$ and $V_R$, and solving for the reflected voltage leads to equation (2.6).

\[ V_R = \left( \frac{R_L - Z_o}{R_L + Z_o} \right) V_S = \rho_{Lv} V_S \]  \hspace{1cm} (2.6)

The term $\rho_{Lv}$ is called the load's voltage reflection coefficient. The derivation of the current reflection coefficient, $\rho_{Li}$, is similar, and equals $-\rho_{Lv}$. Usually one coefficient, $\rho_L$, is used for both, as shown in equation (2.7) where the resistance, $R_L$, has been replaced with the general impedance $Z_L$.

\[ \rho_L = \rho_{Lv} = -\rho_{Li} = \left( \frac{Z_L - Z_o}{Z_L + Z_o} \right) \]  \hspace{1cm} (2.7)

Note that the reflection coefficient's absolute value is never greater than one. For the circuit in Figure 8, $\rho_L$ is equal to $-1/2$ so the voltage reflected from the load is equal to $-1/2V_S$.

After another delay of $T_{pd}$, the reflected wave leaving the load returns to the source, where it is reflected by the source resistance of the driver. In this case the source reflection coefficient has the form given by equation (2.8).
\[ \rho_S = \rho_{sv} = -\rho_{si} = \frac{Z_S - Z_o}{Z_S + Z_o} \]  

(2.8)

For the example circuit, the source resistance is zero, so the reflection coefficient, \( \rho_S \), is equal to -1. Total reflection occurs, and a negative version of the incident wave begins its journey back towards the load. Reflections from the load and source continue, gradually dying down with each successive reflection from the load (as a result of the load’s attenuating reflection coefficient of -1/2).

The voltage for a certain time and location on the transmission line is determined by the sum of the initial voltage reaching that point, and all the subsequent reflections that follow it up to the time of interest. Figure 9 graphically demonstrates the voltage on the transmission line at two points: one located halfway down the line (Figure 9(a)), and the other located at the load (Figure 9(b)).

Consider the plot shown in Figure 9(a). At time \( T_{pd}/2 \), the initial voltage step reaches the mid-point of the transmission line, raising the voltage immediately to \( V_S \) volts. The initial step continues on towards the load, where it generates a reflected step of \(-1/2V_S\) (since \( \rho_L = -1/2 \)). This reflection crosses the mid-point of the line at time \( 3/2T_{pd} \) lowering the voltage from \( V_S \) to \( 1/2V_S \). A little later, a negative (total) reflection at the source sends a positive step of magnitude \( 1/2V_S \) back towards the midpoint returning the line voltage back to its initial value of \( V_S \) (at time \( 5/2T_{pd} \)). The oscillations continue indefinitely, decaying by a factor of 50% with every interval of \( 2T_{pd} \).
Figure 9. Voltage versus time plots for two points on the transmission line of Figure 8. Plot (a) is for a position located halfway between the source and load, while plot (b) represents the voltage at the load. A similar plot for the source would show a constant voltage of $V_s$ volts, because the incident and reflected signals cancel immediately, thus preserving the initial voltage.

In the end, a steady state voltage is reached, shown by equation (2.9). It is determined from the sum of the initial voltage, and the infinite series of reflected waves that follow.

$$V_{\text{steady state}} = (1 + \rho_L + \rho_S \rho_L + \rho_S^2 \rho_L^2 + \ldots) V_s$$  \hspace{0.5cm} (2.9)

By grouping the odd and even terms inside the parentheses of equation (2.9), a factor of $(1+\rho_L)$ can be taken out, producing equation (2.10).

$$V_{\text{steady state}} = (1 + \rho_L)(1 + \rho_S \rho_L + \rho_S^2 \rho_L^2 + \ldots) V_s$$  \hspace{0.5cm} (2.10)

The middle term represents an infinite geometric series for the common ratio $\rho_S \rho_L$. This well known infinite series converges to $1/(1-\rho_S \rho_L)$ as long as $|\rho_S \rho_L|<1$. Convergence is always guaranteed for transmission lines if either the source or load has a finite non-zero resistance, as is the case for Figure 8.

Assuming equation (2.9) converges, the steady state voltage can be rewritten as follows.
\[ V_{\text{steady state}} = \left( \frac{1 + \rho_L}{1 - \rho_S \rho_L} \right) V_S = \left( \frac{Z_L}{Z_S + Z_L} \right) V_S \]  

(2.11)

This result is the same as what one obtains when transmission line effects are ignored, which makes intuitive sense since transmission line effects are transient, and should converge with the ideal-wire case over the long term.

The steady state current is obtained in a similar manner, and is given by equation (2.12).

\[ I_{\text{steady state}} = \left( \frac{1}{Z_S + Z_L} \right) V_S \]  

(2.12)

Reflections lead to undesirable waveforms on transmission lines. As Figure 9(b) demonstrates, they can increase the switching delay of a signal significantly beyond what one would expect from the pure propagation delay of \( T_{pd} \). In this case, the load waveform takes five times longer to cross the 3/4 supply voltage threshold than if there were no reflections.

Even more serious problems stem from the oscillations that develop under certain situations which can lead to false switching. Figure 9(a) shows an example of such an oscillation at the mid-point of a transmission line where the voltage reaches the 50% level twice on its way to \( V_S \) volts. At the least, this type of oscillation reduces the noise margin of a circuit.

The clock buffer chip built for this thesis uses point-to-point signalling, so receivers are never connected to the middle of a signal trace; however, similar oscillations can also occur at the ends of a transmission line. Figure 10 gives an example of an oscillation, which develops when the load is equal to \( 3Z_0 \).
Other types of reflection problems are overshoots (on rising edges), and undershoots (on falling edges). Figure 10 demonstrates an overshoot condition. If these excursions above and below normal voltage levels are large enough, they can damage circuits.

![Figure 10. Load voltage waveform for the same transmission line configuration as Figure 8, but with the load resistance changed to 3Zo. Both oscillations and overshoots are present in the waveform.](image)

It should be apparent that reflections need to be controlled. Fortunately, there are ways of accomplishing this by "matching" the load or source impedance to the characteristic impedance of the transmission line, thereby ensuring that the magnitudes of the reflected signals are small. This topic will be addressed in the next section.

### 2.4.5 Series and Parallel Matching of Transmission Line Impedance

There are several ways of limiting reflections on transmission lines. Two approaches are shown in Figure 11, and they involve the use of a resistor matched to the characteristic impedance of the transmission line to suppress reflections. The resistor can be placed in series with the transmission line at the source, or in parallel with the load. Each approach has its advantages and disadvantages.
Figure 11. (a) Series terminated transmission line. (b) Parallel terminated transmission line. Both circuits are connected to ideal sources and loads.

Series termination uses a resistor, $R_S$, set equal to $Z_0$. To see how this sort of termination works, assume that a step of magnitude $V_S$ is applied to the transmission line at time zero. The initial voltage applied to the transmission line is determined by the voltage divider formed by $R_S$ and $Z_0$ (located at the input to the line), and is given by equation (2.13).

$$V_I = \left( \frac{Z_o}{Z_o + R_S} \right) V_S$$

For a matched line, $V_I$ equals $1/2V_S$, and this voltage step is transmitted to the load where the load reflection coefficient of $+1$ results in total reflection. The reflected signal arrives back at the source after a time of $2T_{pd}$, raising the line's voltage to its final value of $V_S$. No further reflections occur because the source reflection coefficient is zero (i.e. $R_S$ is matched to $Z_0$). Thus after a delay of $2T_{pd}$, the entire line settles to its final value with no further reflections. Of course, parts of the transmission line closer to the load settle in less time; the load reaches its final value at a time of $T_{pd}$. 
Parallel termination also requires a resistor, $R_L$, set equal to $Z_0$. Here, an initial voltage pulse of magnitude $V_S$ travels unhindered until it reaches the load. Since the load's reflection coefficient is zero, no further reflections occur, and the transmission line is stable after a time of $T_{pd}$. Thus parallel-terminated lines settle faster than series terminated lines; however, the parallel terminated line has the disadvantage that there is DC power dissipation when a non-zero voltage is being driven.

Although a series terminated line does not settle completely until after $2T_{pd}$, the $1/2V_S$ step that persists at the source during this time can be used to good advantage. In fact, this step is used by the clock buffer chip's propagation delay estimator to obtain an estimate for the source-to-load delay. More will be said about this in section 2.5, and Chapter 3.

2.4.6 Non-Uniform Transmission Lines

Throughout the preceding sections, it was assumed that transmission lines had a uniform characteristic impedance. However, it is worth studying what happens when a transmission line's characteristic impedance is not constant.

When a line has a discontinuity where the characteristic impedance changes from one value to another, Ohm's Law cannot be satisfied without a reflection occurring at the discontinuity. A fraction of the incident wave is reflected back towards the source, as determined by the reflection coefficient for the junction. (The reflection coefficient is identical to equation (2.6) but with the load impedance replaced with the characteristic impedance of the discontinuous section of the line.) The fraction of the incident wave that is not reflected, is transmitted past the discontinuity, where it continues on towards the load.

When such a discontinuity is present, reflections are generated at three points: at the source, the load, and also at the discontinuity. If series or parallel matching is used then
reflections will not be present from one of these three points and the situation is improved somewhat. Analysis of such multiple reflections is straight-forward, but tedious, and will not be covered here.

Fortunately, major discontinuities are not very common, since circuit board fabrication processes tend to generate reasonably uniform signal paths if the number of vias are kept to a minimum. However, what can cause problems are acute and 90° bends in signal traces, and they should be avoided in high speed circuit designs. Circuit board manufacturers recommend using either smooth curves or 135° bends to round corners to diminish the effects of these discontinuities. Using 135° bends is probably the best compromise between good uniformity and ease of manufacturability ([19], p. 38).

Connector discontinuities are harder to deal with, since it is difficult to match connector impedance to transmission line impedance. Coaxial cable connections are useful here, but generally too bulky for high density circuits.

2.4.7 Driving Multiple Transmission Lines

Another situation that arises, particularly with regards to a clock buffer chip, is having one source drive multiple transmission lines. Such lines can be terminated in either a series or a parallel manner as shown in Figure 12.

In the parallel terminated case (Figure 12(a)), the presence of the matched load resistors ensures that no reflections are generated at the load, so full termination is possible even with lines of varying length. The drawback however, is that DC power dissipation becomes very severe now that several parallel matching resistors must be driven at once. A source driving 5V on three 50Ω lines must be able to sustain a large DC current of 300mA.
Figure 12. (a) Single source driving multiple parallel terminated lines. (b) Single source driving multiple series terminated lines. The source resistance is set to $Z_0/n$, where $n$ represents the number of lines being driven.

Series termination is less effective when it comes to driving multiple lines. Figure 12(b) presents one series terminated configuration that can be used to drive multiple lines. Since the series resistance must be matched to the parallel combination of transmission lines, it is given a value of $Z_0/n$, where $n$ is the number of lines being driven. However, reflections are not fully absorbed at the source since each incident reflection sees a reflection coefficient of $(1-n)/n$ which is non-zero for $n>1$. However if multiple lines are used, and if all these lines are of the same length, then the reflected wave on an individual line is exactly cancelled out by the transmitted waves from the other lines ([18], pp. 42-43).

Although the clock buffer device designed for this thesis (which uses source termination) was not designed to drive multiple lines, it can operate in this configuration if the lines being driven are of equal length, and if the source impedance of $Z_0/n$ can be reached by the output driver of the chip. In fact, equal length lines are required for another reason, and that is to guarantee the proper functioning of the propagation delay estimator used in each clock bay of the clock buffer chip.
Better configurations for series termination exist ([18], p. 44), but they will not be covered here since they are beyond the scope of this discussion.

2.5 Transients on Transmission Lines

The sources and loads used in typical circuits are often far from ideal. In addition to having non-zero output resistance, and finite input resistance, they also have a capacitive component. With digital circuits, capacitive loads stem from the parasitic capacitance present on the inputs and outputs of logic gates, and there is very little one can do to avoid them. It is important to understand the impact that such capacitances can have on the waveforms transmitted through transmission lines.

In general, such capacitances delay and distort signals. The delay is similar to that observed with 1st order resistor-capacitor (RC) circuits, while the distortions are complex and difficult to describe.

A tool that is of great importance in the analysis of such signals is the Laplace transform. This section will use the Laplace transform to derive equations for the transients present in source terminated transmission line configurations like the one shown in Figure 13. The configuration is similar to that used at the outputs of the clock buffer chip built for this thesis, and will be analyzed in greater detail in Chapter 3.

![Figure 13. Source terminated transmission line with a capacitive load.](image-url)
A typical input used for transient analysis is a step waveform, since it is a reasonable approximation to a digital input, and is straightforward to work with; more realistic ramp inputs will be studied in Chapter 3. The input to the system of Figure 13 is a step function of height $V$.

$$v_S(t) = Vu(t)$$  \hspace{1cm} (2.14)

Applying the Laplace transform to equation (2.14) gives

$$V_S(s) = \frac{1}{s}$$  \hspace{1cm} (2.15)

The initial voltage applied to the transmission line, $V_I$, is determined from equation (2.13) and equation (2.15).

$$V_I(s) = \frac{V}{2s}$$  \hspace{1cm} (2.16)

Next, the load reflection coefficient, $P_L(s)$, is obtained by applying equation (2.7). It is shown below with the variable $\tau_L$ representing the load's time constant, $Z_0C_L$.

$$P_L(s) = \frac{1}{\frac{1}{\tau_L} + s}$$  \hspace{1cm} (2.17)

The equation for the load after the first reflection is given by the sum of the incident voltage at the load, and the reflection from the load. It is shown in equation (2.18). (In order to account for the source-to-load propagation delay, multiplication by exponentials of the form $e^{-st}$ is required.)

$$V_L(s) = V_I(s)e^{-T_{ps}s} + P_L(s)V_I(s)e^{-T_{ps}s}$$  \hspace{1cm} (2.18)

The time domain version of the load voltage equation, valid for all time, is given by equation (2.19). Note that what was multiplication in the Laplace domain must now
become convolution in the time domain, where the convolution operator is denoted by the \( \otimes \) symbol.

\[
v_L(t) = v_l(t - T_{pd}) + p_L(t) \otimes v_l(t - T_{pd})
\]  
(2.19)

The equivalence between convolution in the time domain and multiplication in the Laplace domain is very useful. Convolutions that are very difficult to evaluate in the time domain reduce to simple multiplications in the Laplace domain.

Simplifying equation (2.18) leads to equation (2.20).

\[
V_L(s) = \frac{V}{\tau_L s} e^{-T_{pd}s}
\]  
(2.20)

Applying partial fraction expansion (PFE) to break up the result into terms with simpler denominators produces equation (2.21).

\[
V_L(s) = V \left( \frac{1}{s} + \frac{1}{s + \frac{1}{\tau_L}} \right) e^{-T_{pd}s}
\]  
(2.21)

Finally, converting the resulting terms to their time domain equivalents leads to the complete load voltage equation.

\[
v_L(t) = V \left( 1 - e^{-\frac{t - T_{pd}}{\tau_L}} \right) u(t - T_{pd})
\]  
(2.22)

The equation for the source end of the transmission line is similar in that it contains two voltage terms: one for the initial voltage step, and another for the reflection from the load. The difference, however, is that the initial voltage at the source is present at time zero, and the reflection from the load arrives at time \( 2T_{pd} \). Recall that, with the load, both terms appear at time \( T_{pd} \).
The Laplace domain equation for the source end of the transmission line is given by equation (2.23). It is applicable to all times $\geq 2T_{pd}$.

$$V_{src}(s) = V_I(s) + p_L(s) V_I(s) e^{-2T_{pd}s}$$  \hfill (2.23)

The time domain source voltage equation, valid for all time, is given by equation (2.24).

$$v_{src}(t) = \frac{V}{2} u(t) + V \left( \frac{1}{s} - e^{\frac{-t-2T_{pd}}{\tau_L}} \right) u(t - 2T_{pd})$$  \hfill (2.24)

Plots of the source and load transients given by equation (2.22) and equation (2.24) are shown in Figure 14(a) and Figure 14(b) respectively. It is evident from these plots that the load experiences additional delay compared to when there is no capacitor present. In addition, the source experiences a sharp downward spike due to the load reflection.

![Figure 14](image-url)

*Figure 14. (a) Voltage transient response for the load end of the transmission line of Figure 13. (b) Transient response for the source end of the transmission line. In both Figures (a) and (b), the dashed line indicates the expected waveform in the absence of the load capacitance.*

The sharp downward spike is exaggerated compared to what is observed in reality, because of the infinitely fast rise time of the input supplied to this circuit. In practice, the spike is significantly smaller since realistic input waveforms behave more like a ramp than...
a step signal. Also, there is a relatively large source capacitance that further diminishes the signal rise time, and the magnitude of the downward spike (see Chapter 3).

The load reflection is complicated, because the load end of the transmission line does not behave as if it has a constant reflection coefficient. When the incident wave first strikes the load, the load readily accepts current due to the rapid change in voltage. The load acts like a short to ground which leads to a negative reflection coefficient. Afterwards, when the incident voltage begins leveling off, the capacitor no longer accepts current as easily as it did before, and this leads to a positive reflection coefficient. Of course, this description is somewhat of an oversimplification, but it is useful in understanding the observed behaviour.

As mentioned in section 2.4.5, the $2T_{pd}$ wide step that is generated in a source terminated transmission line configuration (see Figure 14(b)) provides an indirect estimate of the source-to-load propagation delay. The clock buffer chip measures the width of this step to determine the round-trip delay from the source to the destination. This information is used to de-skew outgoing signals, and remove the effects of the source-to-load propagation delay.

Measuring the width of the step is performed by logic that is triggered at thresholds set to just above and below the step’s voltage, and this will be discussed in greater detail in Chapters 3 and 4. However, at this point, it is important to note that the thresholding technique, by its very nature, is sensitive to the transients generated by parasitic capacitance on the transmission line. For example, in Figure 14(b), the large 1/2V downward spike appearing at time $2T_{pd}$ would be a serious impediment to the success of such a thresholding technique, since the spike would most definitely affect the lower level threshold circuit.
For this reason, Chapter 3, is devoted to a more detailed understanding of the transients expected in realistic source terminated transmission line configurations. Another effect that Chapter 3 attempts to quantify is the error inherent in the source-to-load delay estimate.
Removing skew from clocks in digital systems improves performance, and is the primary goal of the clock buffer chip designed for this thesis. The clock buffer chip (CBC) is intended to de-skew outgoing signals in order to effectively remove variations in the clock buffer-to-load propagation delay.

As mentioned in sections 2.4.5 and 2.5, the outputs of the CBC use a series terminated transmission line configuration (Figure 11(a)), which allows it to obtain an estimate of the propagation delay from observations of the waveforms at its outputs. Each output transition has an associated voltage step whose width provides an estimate of the round trip delay from the source to the load.

It is of vital importance to analyze how well measurements of this voltage step translate to an estimate of the source-to-load propagation delay. Chapter 3 attempts to address this question from a theoretical standpoint. It presents detailed models for the source terminated transmission line configuration being used, and attempts to provide solutions for the parameters of interest. It presents a comparison of the true source-to-load
propagation delay, with the delay estimate obtained from measurements taken of the source voltage step. From these comparisons, the validity of this propagation delay estimation technique can be demonstrated.

Since the models being studied are relatively complex, a simplified technique is also developed for estimating the expected performance. This technique relies on piecewise-linear approximations of the analytic solutions of the transmission line models, and can be applied very quickly. Some degree of accuracy is lost in the simplification, as described in section 3.4.4, but the results are still valid over a large range of configurations.

Solutions for the most detailed model - one using ramp inputs - are presented first, followed by those for a simplified model using step inputs. It is from the later model that the high-speed piecewise-linear approximation technique is developed. For each model, the expected delay estimation performance of the CBC is determined and analyzed.

Finally, the ramp-input, step-input, and piecewise linear models are compared against an HSPICE [33] model for several different transmission line configurations. This serves as a check upon the correctness and validity of the analytic solutions.

3.1 Series Terminated Transmission Line Model

A general model for the series terminated transmission line configuration used in the CBC is shown in Figure 15. It is based on the typical input and output structures associated with CMOS digital circuitry. The source resistance, $R_S$, is (ideally) matched to the characteristic impedance of the transmission line, $Z_0$, to minimize reflections, and is on the order of $40\,\Omega$ to $130\,\Omega$. Both the source capacitance, $C_S$, and the load capacitance, $C_L$, have values in the $5\,\text{pF}$ to $25\,\text{pF}$ range, while the load resistance, $R_L$, is very high (essentially infinite).
The first step in deriving expressions for this model consists of finding the Thévenin equivalent circuit for the source (i.e., output) circuitry. Figure 16 graphically describes the necessary transformation.

![Figure 15. Series terminated transmission line model for the CBC outputs.](image)

![Figure 16. Thévenin equivalent circuit model.](image)

Computing the Thévenin model leads to equations (3.1) and (3.2).

\[
V_T(s) = \frac{V_P(s)}{1 + sR_S C_S} \tag{3.1}
\]

\[
Z_S = R_S \| \frac{1}{sC_S} = \frac{R_S}{1 + sR_S C_S} \tag{3.2}
\]

Next, the load impedance, \(Z_L\), is obtained from the parallel combination of \(C_L\) and \(R_L\), resulting in equation (3.3).

\[
Z_L = R_L \| \frac{1}{sC_L} = \frac{R_L}{1 + sR_L C_L} \tag{3.3}
\]
The initial voltage applied to the transmission line is given by equation (2.13) (with the source resistance \( R_S \) replaced by \( Z_S \)); however, it is worth repeating here for convenience.

\[
V_I(s) = \left( \frac{Z_o}{Z_S + Z_o} \right)V_T(s)
\]

(3.4)

Substituting equations (3.1) and (3.2) into (3.4) leads to a revised equation for the input voltage.

\[
V_I(s) = \left( \frac{1}{R_S C_S} \right) \left( \frac{R_S + Z_o}{s + \left( \frac{R_S + Z_o}{R_S C_S Z_o} \right)} \right)V_P(s)
\]

(3.5)

Reflection coefficients for the source and load are given by equations (2.7) and (2.8), and, after substituting values for \( Z_S \) and \( Z_L \) into them, one obtains equations (3.6) and (3.7).

\[
P_S(s) = \frac{\left( \frac{R_S - Z_o}{R_S C_S Z_o} \right)}{\left( \frac{R_S + Z_o}{R_S C_S Z_o} \right)} s
\]

(3.6)

\[
P_L(s) = \frac{\left( \frac{R_L - Z_o}{R_L C_L Z_o} \right)}{\left( \frac{R_L + Z_o}{R_L C_L Z_o} \right)} s
\]

(3.7)

To simplify working with the reflection coefficients, some of the terms in equations (3.6) and (3.7) will be replaced by the variables \( a \), and \( d \), as indicated in Table 1.

<table>
<thead>
<tr>
<th>Reflection Coefficient Terms</th>
</tr>
</thead>
<tbody>
<tr>
<td>( a = \frac{R_S + Z_o}{R_S C_S Z_o} )</td>
</tr>
</tbody>
</table>

At this point it is possible to begin computing voltages along the transmission line of Figure 15. The equation in question is a function of the position along the line, \( X \), and the
time of interest, \( t \), as indicated by equation (3.8). Line position \( X \) takes on values from 0 to \( L \), where \( L \) is the length of the transmission line. Variable \( t_{pd} \) represents the propagation delay measured in units of ns/m, and variable \( V_{dc} \) is the voltage of the transmission line prior to stimulation (usually 0V).

\[
v(X,t) = [v_i(t-t_{pd}X) + \rho_L \otimes v_i(t-t_{pd}(2L-X))] \\
+ (\rho_L \otimes \rho_S) \otimes v_i(t-t_{pd}(2L+X)) \\
+ (\rho_L \otimes \rho_L \otimes \rho_L) \otimes v_i(t-t_{pd}(4L-X)) \\
+ (\rho_L \otimes \rho_S \otimes \rho_L \otimes \rho_S) \otimes v_i(t-t_{pd}(4L+X)) \\
+ \ldots] + V_{dc}
\]

Each term in equation (3.8) that contains one or more \( \rho_S \) or \( \rho_L \) variables represents a reflection term. There are potentially an infinite number of terms; however, for matched transmission lines, it is usually the first few terms that play the biggest role in shaping the waveform. Analytic solutions that describe more than two reflections are extremely lengthy, so it is fortunate that only the first two reflections (i.e. one at the load and one at the source) are required to describe the voltage step at the source end.

Computer software was used to develop analytic models for greater numbers of reflections to ensure that tertiary and higher order reflections are typically small enough not to interfere with measurements of the source voltage step. This also permitted a more detailed comparison of the analytical model with simulations based on the analog circuit simulator HSPICE [33].

The approach used to solve for the source and load waveforms of Figure 15 consists of two stages. Stage one involves using the Laplace transform to compute the time domain version of the input to the transmission line, \( v_i(t) \), and of the two reflection coefficients for the first two reflections (i.e. \( \rho_L(t) \), and \( \rho_S(t) \otimes \rho_L(t) \)). Step two consists of performing time domain convolutions between (delayed versions of) the transmission line input, and these reflection coefficients.
The reason for choosing not to solve the entire solution using the Laplace transforms is the difficulty of performing the inverse Laplace transform for the whole solution. Evaluating the later parts of the computation in the time domain is a better approach, since it involves relatively simple convolutions between exponential and step function terms.

### 3.2 Series Terminated Transmission Line With A Ramp Input

A ramp input, \( r(t) \), is a more realistic model of a digital input than a step function. It is described by equation (3.9), where \( V_M \) is the maximum value of the ramp, and \( t_R \) is the transition time of the ramp. Both \( V_M \) and \( t_R \) can be considered to be constants.

\[
r(t) = \begin{cases} 
0 & \text{for } t < 0 \\
\left(\frac{V_M}{t_R}\right) t & \text{for } 0 \leq t < t_R \\
V_M & \text{for } t_R \leq t 
\end{cases}
\]  
(3.9)

The term \( V_M/t_R \) is simply the slope of the ramp, and from this point forward it will be represented by the variable \( m_R \). An equation representing a ramp input can be formed by the sum of two linearly changing inputs as indicated by equation (3.10). Function \( u(t) \) is used to represent a step function in this equation.

\[
r(t) = m_R [tu(t) - (t - t_R)u(t - t_R)]
\]  
(3.10)

Applying the Laplace transform to the above leads to equation (3.11), which will be used as variable \( V_p(s) \) in equation (3.5).

\[
R(s) = V_p(s) = \frac{m_R}{s^2} \left(1 - e^{-st_R}\right)
\]  
(3.11)
3.2.1 Input Waveform

Equations (3.5) and (3.11) can now be combined to obtain the input waveform to the transmission line. The result is shown in equation (3.12), where the variable \( a \) from Table 1 has been used to improve clarity.

\[
V_I(s) = \left( \frac{m_R}{RSC_S} \right) \left( \frac{1}{s^2} \right) \left( \frac{1}{s + a} \right) (1 - e^{-st})
\]  

(3.12)

Applying the inverse Laplace transform to equation (3.12) leads to the time domain version of the input waveform.

\[
v_I(t) = \left( \frac{m_R}{RSC_S} \right) \left[ \left( \frac{at - 1 + e^{-at}}{a^2} \right) u(t) - \left( \frac{a(t - t_R) - 1 + e^{-a(t - t)}{a^2} \right) u(t - t_R) \right]
\]  

(3.13)

3.2.2 Reflection Coefficients

Equation (3.7) is converted to the time domain by performing partial fraction expansion and then applying the inverse Laplace transform. Equation (3.14) summarizes the result.

\[
\rho_L(t) = \left( \frac{2}{C_LZ_o} \right) e^{-dt} u(t) - \delta(t)
\]  

(3.14)

To obtain the compound reflection coefficient representing a 2nd order reflection (i.e. \( \rho_S(t) \otimes \rho_L(t) = \rho_2(t) \)), the product of equations (3.6) and (3.7) is taken, followed by application of the inverse Laplace transform. The resulting equation is naturally more complex than equation (3.14) and is shown as equation (3.15), where \( \gamma \) and \( \mu \) represent constants defined by equations (3.16) and (3.17).

\[
\rho_2(t) = [\gamma e^{-dt} - \mu e^{-at}] u(t) + \delta(t)
\]  

(3.15)

\[
\gamma = \left( \frac{2}{C_LZ_o} \right) \left[ \frac{(R_L + Z_o)R_SC_S + (R_S - Z_o)R LC_L}{(R_S + Z_o)R_LC_L - (R_L + Z_o)R_SC_S} \right]
\]  

(3.16)
\[
\mu = \left( \frac{2}{C_S Z_o} \right) \frac{(R_S + Z_o) R_L C_L + (R_L - Z_o) R_S C_S}{(R_S + Z_o) R_L C_L - (R_L + Z_o) R_S C_S}
\]  
(3.17)

For upcoming calculations, it is helpful to define versions of \( v_L(t) \), \( \gamma \), and \( \mu \) without the leading constant terms since these terms can be combined and simplified. The new variables are denoted with a prime symbol and are defined according to equations (3.18) to (3.20).

\[
v(t) = \left( \frac{m_R}{C_S R_S} \right) v_L(t) \quad (3.18)
\]

\[
\gamma = \left( \frac{2}{C_L Z_o} \right) \gamma' \quad (3.19)
\]

\[
\mu = \left( \frac{2}{C_S Z_o} \right) \mu' \quad (3.20)
\]

### 3.2.3 Load And Source Voltage Waveforms

Applying equation (3.8) to the load and source leads to equations (3.21) and (3.22) respectively.

\[
v_L(t) = v(L, t) = v_L(t - t_{pd} L) + p_L(t) \otimes v_i(t - t_{pd} L) \quad (3.21)
\]

\[
v_S(t) = v(0, t) = v_L(t) + p_L(t) \otimes v_i(t - t_{pd} 2L) + p_2(t) \otimes v_i(t - t_{pd} 2L) \quad (3.22)
\]

An equivalent form of these equations is shown in equations (3.23) and (3.24). The convolution theorem of equation (3.25) has been used to perform the conversion, where variable \( t_d \) represents a constant time delay. This conversion simplifies the \( v_i(t) \) term.

\[
v_L(t) = v_i(t - t_{pd} L) + p_L(t - t_{pd} L) \otimes v_i(t) \quad (3.23)
\]

\[
v_S(t) = v_i(t) + p_L(t-t_{pd} 2L) \otimes v_i(t) + p_2(t-t_{pd} 2L) \otimes v_i(t) \quad (3.24)
\]

\[
A(t - t_d) \otimes nB(t) = A(t) \otimes B(t - t_d) \quad (3.25)
\]
Since equations (3.23) and (3.24) consider only up to 2nd order reflections, they are only valid until a time of $t_{pd}3L$ for the load and $t_{pd}4L$ for the source (i.e. 3 or 4 one-way trips across the transmission line); however this time period includes the important initial rise of the load, and also the voltage step at the source. Both these items need to be characterized to determine the performance of the CBC's delay estimation technique.

Evaluating equations (3.23) and (3.24) is done by performing time domain convolutions of equation (3.13) with equations (3.14) and (3.15) and then summing the results. The initial steps of the evaluation are shown in equations (3.26) and (3.27).

$$v_L(t) = \left( \frac{m_R}{C_S R_S} \right) \left( \frac{2}{C_L Z_o} \right) \left[ v_I'(t) \otimes e^{-d(t-t_P)} u(t - t_P) \right]$$

(3.26)

$$= \left( \frac{m_R}{C_S R_S} \right) \left( \frac{2}{C_L Z_o} \right) L(t)$$

$$v_S(t) = \left( \frac{m_R}{C_S R_S} \right) v_I'(t)$$

(3.27)

$$+ \left( \frac{m_R}{C_S R_S} \right) \left( \frac{2}{C_L Z_o} \right) \left[ (1 + \gamma') [v_I'(t) \otimes e^{-d(t-t_P)} u(t - t_P)] \right]$$

$$- \left( \frac{m_R}{C_S R_S} \right) \left( \frac{2}{C_L Z_o} \right) \mu' [v_I'(t) \otimes e^{-\alpha(t-t_P)} u(t - t_P)]$$

$$= \left( \frac{m_R}{C_S R_S} \right) \left[ v_I'(t) + \left( \frac{2}{C_L Z_o} \right) (1 + \gamma') L(t) - \left( \frac{2}{C_S Z_o} \right) \mu' S(t) \right]$$

Variable $t_P$ in the load voltage equation is equal to $t_{pd}L$, but for the source equation it is equal to $t_{pd}2L$.

Both the $\rho_L(t-t_P)$, and $\rho_2(t-t_P)$ reflection coefficient terms (i.e. equations (3.14) and (3.15)) contain one exponential function and one delta function, which must be convolved with the input to determine the load and source waveforms. Convolutions of the input with the delta functions have already been performed to obtain equations (3.26) and (3.27); however, convolutions of the input with the exponential functions have not been done yet. The remaining convolutions are denoted $L(t)$, and $S(t)$ in equations (3.26) and (3.27).
The $L(t)$ and $S(t)$ convolutions consist of a sum of smaller convolutions between the individual terms of $v'_r(t)$ and the exponentials in the reflection coefficient terms. Performing these smaller convolutions requires that only three general types of convolutions be evaluated. These convolutions are summarized in Table 2 along with their results. In actuality, there are six types of convolutions, but the additional three are simply repetitions of those shown in Table 2 with the ramp transition time, $t_R$, set to zero.

Variables $a$ and $d$ shown in the Table 2 were defined earlier in Table 1, while variable $t_p$ represents the propagation delay, which for the source and load takes on values that are multiples of $t_{pd}$.

**TABLE 2. Convolutions Required For The Computation of Equations (3.26) and (3.27)**

<table>
<thead>
<tr>
<th>Type</th>
<th>Convolution = Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>$(t - t_R)u(t - t_R) \otimes e^{-d(t-t_p)} u(t - t_p) =$</td>
</tr>
<tr>
<td></td>
<td>$\left( -\frac{1}{d^2} + \frac{t - (t_p + t_R)}{d} + \frac{e^{-d(t-(t_p+t_R))}}{d^2} \right) u(t - (t_p + t_R))$</td>
</tr>
<tr>
<td>2.</td>
<td>$u(t - t_R) \otimes e^{-d(t-t_p)} u(t - t_p) =$</td>
</tr>
<tr>
<td></td>
<td>$\left( \frac{1}{d} - \frac{e^{-d(t-(t_p+t_R))}}{d} \right) u(t - (t_p + t_R))$</td>
</tr>
<tr>
<td>3(a)</td>
<td>$e^{-a(t-t_R)} u(t - t_R) \otimes e^{-d(t-t_p)} u(t - t_p) =$</td>
</tr>
<tr>
<td></td>
<td>$\left( \frac{e^{-a(t-(t_p+t_R))}}{d-a} - \frac{e^{-d(t-(t_p+t_R))}}{d-a} \right) u(t - (t_p + t_R))$</td>
</tr>
<tr>
<td>(For $a \neq d$ )</td>
<td></td>
</tr>
<tr>
<td>3(b)</td>
<td>$e^{-a(t-t_R)} u(t - t_R) \otimes e^{-d(t-t_p)} u(t - t_p) =$</td>
</tr>
<tr>
<td>(For $a = d$ )</td>
<td>$((t - (t_p + t_R)) e^{-a(t-(t_p+t_R))}) u(t - (t_p + t_R))$</td>
</tr>
</tbody>
</table>

A complication arises with convolution type 3 in Table 2. There are two solutions, depending on whether variables $a$ and $d$ (defined in Table 1) are equal or not. When evaluating convolution $L(t)$ in both equations (3.26) and (3.27), usually variable $a$ does not
equal $d$, so type 3(a) convolutions are needed. If $a=d$ then convolution $L(t)$ reduces to convolution $S(t)$ which uses type 3(b) convolutions.

Equality is a relatively rare occurrence, so when would $a=d$? Equating $a$ and $d$, and making the reasonable assumption that $R_L$ is very large, leads to equation (3.28), that describes the values of $R_S$ that ensure equality given a fixed $Z_o$, $C_S$, and $C_L$. The equation indicates that valid (i.e. non negative) values of $R_S$ that lead to the $a=d$ situation are only obtained if the capacitance ratio $C_S/C_L$ is greater than one. Furthermore, realistic values of $R_S$ (i.e. $3Z_o$ to $1/3Z_o$) are obtained for capacitance ratios in the range of 4/3 to 4. Such capacitance ratios and source resistances are to be expected, so, although equality is rare, it still may occur in realistic transmission line configurations. Therefore, a general solution to the source and load voltages must include two solutions for the $L(t)$ convolution.

$$R_S = \frac{Z_o}{\left(\frac{C_S}{C_L} - 1\right)} \quad (3.28)$$

The result of convolutions $L(t)$ and $S(t)$ are shown in equations (3.29) and (3.30). Together with equations (3.26) and (3.27), they form the complete solution for the source and load voltage waveforms for a ramp input (up to a time of $t_{pd}3L$ for the load and $t_{pd}4L$ for the source).

$$L(t)_{a\neq d} = \left[\frac{t-t_p}{da} - \frac{a+d}{(ad)^2} - \frac{1}{d-a}\left(\frac{e^{-d(t-t_p)}}{d^2} - \frac{e^{-a(t-t_p)}}{a^2}\right)\right]u(t-t_p)$$

$$+ \left[\frac{t-t_p-t_R}{da} + \frac{a+d}{(ad)^2} + \frac{1}{d-a}\left(\frac{e^{-d(t-t_p-t_R)}}{d^2} - \frac{e^{-a(t-t_p-t_R)}}{a^2}\right)\right]u(t-t_p-t_R) \quad (3.29)$$

Performance Analysis
\[ S(t) = L(t) \bigg|_{a = d} = \left[ -\frac{2}{a^3}(1 - e^{-a(t-t_p)}) + \frac{t - t_p}{a^2} \right] \left[ 1 + e^{-a(t-t_p)} \right] u(t-t_p) \]

\[ + \left[ \frac{2}{a^3}(1 - e^{-a(t-t_p-t_R)}) - \frac{t - t_p - t_R}{a^2} \right] \left[ 1 + e^{-a(t-t_p-t_R)} \right] u(t-t_p-t_R) \]

A typical plot of the source and load waveforms is shown in Figure 17. For comparison, a more detailed model including fifth order reflections is shown in Figure 18 for the same configuration. This later model was evaluated by the symbolic math package called MAPLE [20].

\[ \text{Voltage (V)} \]

\[ \text{Source Waveform} \]

\[ \text{Load Waveform} \]

\[ \text{Time (s)} \]

Figure 17. Plot of the source and load waveforms for the transmission line model using ramp inputs. This plot is for the following configuration: \( C_S = C_L = 10pF \), \( R_S = Z_o = 65\Omega \), \( R_L = 100k\Omega \), \( L = 25cm \), \( V_M = 5V \), \( t_R = 1ns \), \( t_{pd} = 6ns/m \).
3.3 Series Terminated Transmission Line With A Step Input

The source and load waveform equations for a ramp input are lengthy, and awkward to work with. In an effort to simplify matters, voltage waveforms will be derived for a step input. Although the results are not as accurate as for a ramp input, they are significantly simpler to compute, and have adequate accuracy for use in performance evaluation of the CBC.

The step input, $v_p(t)$ and its Laplace transform, $V_p(s)$ are shown in equations (3.31) and (3.32). Variable $V_M$ represents the height of the step.

$$v_p(t) = V_M u(t) \quad (3.31)$$
$$V_p(s) = \frac{V_M}{s} \quad (3.32)$$

Equation (3.32) is substituted into equation (3.5) to obtain the input to the transmission line, $V_I(s)$. Applying the inverse Laplace transform to the result generates the time domain version of the input, $v_I(t)$. 
\[ V_l(s) = \frac{V_M}{R_s C_s} \left( \frac{1}{s + a} \right) \frac{1}{s} . \] (3.33)

\[ v_l(t) = \frac{V_M Z_o}{R_s + Z_o} (1 - e^{-at}) u(t) = \left( \frac{V_M Z_o}{R_s + Z_o} \right) v'_l(t) \] (3.34)

The reflection coefficients presented in equations (3.14) and (3.15) are still in effect, as are the source and load voltage equations presented in equations (3.23) and (3.24). However, the initial stages of the convolution are slightly different from equations (3.26) and (3.27), because of the different constants in the input waveform. The modified equations are presented below.

\[ v_L(t) = \left( \frac{V_M}{R_s + Z_o} \right) \left( \frac{2}{C_L} \right) L(t) \] (3.35)

\[ v_S(t) = v_f(t) + \left( \frac{V_M}{R_s + Z_o} \right) \left[ 2\left( 1 + \gamma \right) L(t) - \frac{2u'}{C_s} S(t) \right] \] (3.36)

Convolutions \( L(t) \) and \( S(t) \) above are still defined as convolutions between \( v'_l(t) \) and an exponential term (see equations (3.26) and (3.27)), but they need to be recomputed for the step input. Fortunately Table 2 can still be used to facilitate the computations. The results are indicated in equations (3.37) and (3.38), where variables \( d \) and \( a \) are constants defined earlier in Table 1, and variable \( t_p \) is equal to \( t_{pdL} \) for the load waveform, and \( t_{pd2L} \) for the source waveform.

\[ L(t)|_{a = d} = \frac{1}{d(d - a)} \left[ d - a + ae^{-d(t - t_p)} - de^{-a(t - t_p)} \right] u(t - t_p) \] (3.37)

\[ S(t) = L(t)|_{a = d} = \left[ \frac{1 - e^{-a(t - t_p)}}{a} - (t - t_p)e^{-a(t - t_p)} \right] u(t - t_p) \] (3.38)

The last four equations provide the complete solution for the source and load waveforms for a step input. They are valid until time \( t_{pd3L} \) for the load and time \( t_{pd4L} \) for the source.
A plot comparing the ramp and step models is shown in Figure 19. It is for a similar transmission line configuration as for Figures 17. The waveforms for the step input have nearly the same shape as those for the ramp input; however they precede the ramp waveforms by approximately 70% of the ramp transition time, $t_R$. Since this difference is nearly constant, it has little effect on the usefulness of the step model in analyzing the CBC's delay estimation performance. A more detailed comparison of the models is presented in section 3.5.

![Graph showing comparison of ramp and step source and load waveforms](image)

*Figure 19. Comparison of the transmission line models based on step and ramp inputs. The transmission line configuration is identical to that of Figure 17.*

### 3.4 Piecewise Linear Approximation Model

Both the ramp and step models suffer from a disadvantage that prevents them from being used to easily analyze the delay estimation performance of the CBC. The limitation stems from the fact that it is not possible to analytically solve for the time when the source and load waveforms cross a given voltage threshold. Solutions of this type are required to
compare the width of the source voltage step to the true source-to-load delay. For the ramp and step models, one must resort to numerical methods to obtain these solutions.

Techniques are available that can be used to approximate the ramp and step models in a manner that can be solved. One approach is to mathematically simplify and approximate the waveform using a technique like Taylor series expansion. This approach was not used since the region of convergence of the Taylor series approximation turned out to be too small to be practical (<1ns).

Another approach to function approximation involves curve fitting. For example, polynomial approximations (e.g. spline curves) of up to degree 4 can be used to approximate key parts of a waveform, and expressions for the times of interest can be obtained by finding the roots of these polynomials for a given threshold voltage.

An even simpler approach - one that is applied below to the step input model - is to use linear segments to approximate portions of the voltage waveforms that are of interest. It is a trivial matter to solve for the time when these linear segments cross various thresholds, and this permits an expression for the expected performance of the CBC to be developed. The challenge then is to develop an accurate method of selecting these linear segments, and to obtain a measure of how well they approximate the step-input transmission line model.

### 3.4.1 Linear Segment Selection

For the source waveform, two voltage thresholds are needed to determine the step width. One threshold is set to approximately 1/4 the supply voltage, and the other to 3/4 the supply voltage. This isolates the source voltage step, since, for matched transmission lines, the step appears near half the supply voltage. The difference in the times for the two threshold crossings gives the step width and an approximate time for the round-trip delay along the transmission line.
Analytic solutions for the lower threshold crossing at the source can be obtained since the initial portion of the waveform (before reflections reach the source) is quite simple as indicated by equation (3.34). However, the upper threshold requires a linear approximation to obtain the crossing time.

For the load waveform there is only one threshold to deal with. The source-to-load delay can be obtained from the time when the load crosses a threshold of half the supply voltage. This threshold is used because it is the typical switching threshold of CMOS digital logic. No analytic solution is available for the time when the load waveform reaches this threshold, so a linear approximation is applied here. Figure 20 summarizes the locations where linear approximations are needed for both the source and load waveforms.

![Figure 20. Linear approximations used for the source (a), and load waveform (b).](image)

A linear approximation (equation (3.39)) is straightforward to determine given two unique points on the original curve described by the ordered pairs \((t_1, V_1)\), and \((t_2, V_2)\).

\[
V(t) = \left( \frac{V_2 - V_1}{t_2 - t_1} \right) t + \frac{V_1 t_2 - V_2 t_1}{t_2 - t_1} \tag{3.39}
\]
An expression for time as a function voltage is given by equation (3.40).

\[ t(V) = \left( \frac{t_2 - t_1}{V_2 - V_1} \right)V + \frac{V_2 t_1 - V_1 t_2}{V_2 - V_1} \]  

(3.40)

3.4.2 Load Waveform Approximation

Equations (3.35) and (3.37) describe the load waveform for the case where reflection coefficient terms \(a\) and \(d\) (see Table 1) are *not* equal. Only this case will be approximated since it is the most common situation. The case where \(a\) and \(d\) are equal can always be handled (with little loss of accuracy given the overall limitations of the approximation) by modifying the transmission line configuration slightly until they are different.

An examination of the load equations reveals that the load waveform is the sum of a constant and two decaying exponential terms. One exponential term has a time constant of \(1/a\), and the other a time constant of \(1/d\). Together, these time constants determine the overall shape of the load waveform, as shown in Figure 21.
Figure 21. Load waveform and its constituent terms for the step-input transmission line model. The line configuration is identical to that of Figure 17.

A reasonable method for selecting the two points for the linear approximation consists of selecting points at time instants related to these two time constants. Visual inspection of graphs generated from MAPLE simulations indicates that selecting points where the two exponential functions decay from 60% to 75% of their initial value leads to good approximations around the half supply voltage threshold. The two approximation points derived from these time constants never coincide (since this approximation applies only if $a \neq d$), so a linear approximation can always be obtained.

Assuming the transmission line is matched, and $R_L >> Z_o$, the time constants $1/a (= \tau_a)$ and $1/d (= \tau_d)$ reduce to equations (3.41) and (3.42) respectively.

$$\tau_a = \frac{1}{a} = \frac{1}{2} Z_o C_S$$  \hspace{1cm} (3.41)

$$\tau_d = \frac{1}{d} = Z_o C_L$$  \hspace{1cm} (3.42)
From these equations, it is easy to see that the difference in the time constants depends on the source and load capacitances. For similar sized capacitances, the exponential using $\tau_a$ decays faster, so it is (arbitrarily) selected as the first point of the approximation, and the second point is obtained from $\tau_d$.

The equation for determining the time it takes for an exponential to decay to fraction $X$ of its initial value is given by equation (3.43). If $X$ is equal to $(1-e^{-1})$, or approximately 0.632, then the decay time equals one time constant. This convenient value of $X$ is used to obtain the times for the two points used in the approximation. The times are given by equations (3.44) and (3.45), where $t_p$ is the propagation delay to the load ($= t_{pdL}$).

\[
\begin{align*}
    t_{\text{decay}} &= -\ln(1-X)\tau \\
    t_1 &= \tau_a + t_p \\
    t_2 &= \tau_d + t_p
\end{align*}
\]

Substituting the times shown in equations (3.44) and (3.45) into the load voltage equation (i.e. equations (3.35) and (3.37)) results in equations (3.46) and (3.47) which determine the two approximation points. These points can then be used with the linear approximation equation - equation (3.40) - to obtain the threshold crossing time for the load waveform.

\[
\begin{align*}
    V_1 &= \frac{V_M}{(R_s + Z_o)C_L d(d-a)} \left[ d - a + ae^{-\frac{d}{a}} - de^{-1} \right] \\
    V_2 &= \frac{V_M}{(R_s + Z_o)C_L d(d-a)} \left[ d - a + ae^{-1} - de^{-\frac{a}{d}} \right]
\end{align*}
\]

### 3.4.3 Source Waveform Approximation

Equations (3.36) to (3.38) comprise the source waveform for a step input. It is clear that the source waveform has a lot in common with the load waveform, which hints that an approximation similar to that used for the load can be applied to the source.
For the lower threshold crossing time, no approximation is needed since the initial rise of the source is described by an equation that can be solved easily for the crossing time. This time is given by equation (3.48) and is derived from equation (3.34).

\[ t = -\frac{1}{a} \ln \left( 1 - \left( \frac{R_S + Z_o}{V M Z_o} \right) V \right) \]

(3.48)

Obtaining an expression for the upper threshold crossing time requires a linear approximation. The rise of the source voltage at the end of the source voltage "step", (i.e the rise due to reflections from the load) is caused by terms that are similar to those constituting the load waveform discussed previously. Like the load, this portion of the waveform consists of the sum of a constant, and two exponential terms, but there is also an additional term of the form \( te^{-at} \). It is this later term that prevents implementing an identical approach as with the load when selecting anchor points for the linear approximation.

The reason for not being able to use the 63% decay time (i.e. single time constant interval) to obtain the anchor points is that the \( te^{-at} \) term has a slower decay that reinforces the \( e^{-at} \) terms. In fact, the \( te^{-at} \) term starts from zero, reaches a maximum after one time constant and only then begins to decay.

MAPLE simulations indicate that reasonable approximations are obtained for a variety of transmission line configurations when the anchor points are selected using longer decay times that are on the order of two time constants (i.e. 86% decay). Using 86% decay times tends to place the anchor points of the linear approximation at, or somewhat above, the 3/4 supply voltage threshold, and leads to better results since the initial rise of the \( te^{-at} \) term (to its maximum value) does not influence the approximation.
Equations (3.49) and (3.50) give the times for the points used in the linear approximation. Variable $t_p$ represents the round trip propagation delay from the source to the load and equals $t_{pd}2L$.

\[
t_1 = 2\tau_a + t_p \quad (3.49)
\]

\[
t_2 = 2\tau_d + t_p \quad (3.50)
\]

Substituting these times into the source voltage equation (i.e. equations (3.36) to (3.38)), produces voltages for the anchor points of the approximation as illustrated by equations (3.51) and (3.52). As with the load, these equations are only for configurations where variables $a$ and $d$ are not equal.

\[
V_1 = \frac{V_M}{(R_S + Z_o)} \left[ Z_o \left( 1 - e^{-2a_{t_p}} \right) + \frac{2(1 + \gamma')}{C_L} \left( \frac{e^{-2a_d}}{d(d-a)} - \frac{e^{-2}}{d-a} + \frac{1}{d} \right) \right]
\]

\[
- \frac{2\mu'}{C_S} \left( \frac{1 - 3e^{-2}}{a} \right) \right]
\]

\[
V_2 = \frac{V_M}{(R_S + Z_o)} \left[ Z_o \left( 1 - e^{-2a_d} \right) + \frac{2(1 + \gamma')}{C_L} \left( \frac{e^{-2a_d}}{d(d-a)} - \frac{e^{-2}}{d-a} + \frac{1}{d} \right) \right]
\]

\[
- \frac{2\mu'}{C_S} \left( \frac{1 - e^{-2a_d}}{a} - \frac{2e^{-2a_d}}{d} \right) \right]
\]

The anchor points can then be applied to the linear approximation equation (i.e. equation (3.40)) to obtain an estimate for the crossing time of the upper threshold of the source waveform.
3.4.4 Error Estimate for the Piecewise Linear Approximations

One advantage of an approximation technique like Taylor series expansion is that it is straightforward to generate an error bound for the approximation ([21], pp. 509-519). This is not the case for the linear approximation technique used here; once the approximation is available it is not always possible to determine a worst case error bound. Nevertheless, an attempt will be made to develop an error bound for the source and load equations of the step input model.

The bound is developed by first determining the worst case time error of a linear approximation applied to the exponential function shown in equation (3.53). Variable $\tau$ is the exponential's time constant, and $k$ is an arbitrary scaling factor. The equation for time as a function of voltage is given by equation (3.54)

\[ V(\tau) = ke^{-\frac{\tau}{\tau}} \quad (3.53) \]
\[ t(V) = -\tau \ln\left(\frac{V}{k}\right) \quad (3.54) \]

The approximation uses points on the exponential at times $p\tau$ and $q\tau$ where $p$ and $q$ are constants. The resulting points are $(p\tau, ke^{-p})$, and $(q\tau, ke^{-q})$. Substituting these points into the linear approximation equation shown in equation (3.40) leads to equation (3.55).

\[ t_{approx}(V) = \frac{(q-p)\tau}{k(e^{-q} - e^{-p})} V + \frac{(pe^{-q} - qe^{-p})\tau}{e^{-q} - e^{-p}} \quad (3.55) \]

The error in time is given by the difference between equations (3.54) and (3.55).

\[ t_{error}(V) = t(V) - t_{approx}(V) \quad (3.56) \]

Taking the derivative of equation (3.56), setting it equal to zero, and solving the resulting equation gives the voltage corresponding to when the maximum time error occurs. This voltage is given by equation (3.57).
Substituting the voltage of equation (3.57) into equation (3.56) leads to an expression for the worst case time error for the approximation of the exponential. It is interesting to note that the error in time is independent of the scaling factor of the exponential, \( k \).

\[
V_{\text{error}} = \frac{k(e^{-q} - e^{-p})}{p - q}
\]  
\[ (3.57) \]

This error only applies in the region between the two points used to generate the linear approximation. If the approximation is used to compute the time for a voltage beyond this interval, the error may be greater.

Ideally, one would want to repeat the above procedure on the source and load equations, not just a simple exponential. Unfortunately, it becomes very difficult to compute an expression for the maximum error in time. In fact, it is very difficult to compute the error even for a term of the form \( te^{-at} \) which happens to be present in the source voltage equation, as one is forced to solve equations containing Lambert functions.

For now, only an approximate bound is available. It is generated by applying equation (3.58) to each exponential, and \( te^{-at} \) term in the load or source equations, and summing the individual errors to get a rough total error. The validity of such a bound is debatable, especially when applied to the source waveform. Validity depends on how accurately an exponential waveform represents the load voltage, and the first reflection at the source.

Calculating the error bound for the load voltage over a wide variety of transmission line configurations leads to a worst case error of 600ps, with typical error being 2 to 4 times smaller. Greater error is apparent with larger capacitances, and also when the time constants differ greatly (i.e. the points making up the approximation do not bracket the voltage threshold tightly). The error bound is not very useful when the time constants are
nearly identical since the interval between the points of the approximation is small and unlikely to contain the voltage threshold. In this case the error is underestimated.

It may be that the best approach for obtaining an error bound, amounts to comparing the step and linear models over many simulations. Results for 224 impedance matched transmission line configurations indicate a worse case error of 240ps. If configurations with lengths less than 5cm are ignored, then the error drops to a respectable 50ps.

(The simulations varied the load capacitance from 5pF to 20pF, the source resistance from 50ohms, to 95ohms, and the length from 2cm to 30cm. The source capacitance was set either to 10pF or equal to the load capacitance.)

3.5 Performance Evaluation of the Delay Estimation Technique

Computations were carried out to determine the delay estimation performance of the CBC as predicted by the ramp-input, step-input, and piecewise linear transmission line models. In addition, HSPICE [33] simulations were carried out for comparison against the analytic models.

The HSPICE model is almost identical to the analytic models, except it also includes comparators at the source for determining the voltage threshold crossing times. Figure 22 graphically depicts the HSPICE model, including the comparator schematic.
Figures 23 to 28 at the end of this section graph the predicted delay-estimation performance of the clock buffer chip (CBC) as determined from these models. They illustrate the estimation error across many different transmission line configurations. Estimation error is determined by measuring the true source-to-load delay via the load waveform, and subtracting from it one half the duration of the source voltage step.

To obtain the true source-to-load delay, one determines the time the load waveform crosses the 1/2 supply voltage threshold, and then subtracts the load rise time (since this rise time is not part of the propagation delay).

At the source end, the propagation delay estimate is obtained from the times the source waveform crosses the 3/4 and 1/4 supply voltage thresholds. These thresholds isolate the voltage step at the source (which appears near 1/2 the supply voltage).

Figures 23 to 28 plot the estimation error as a function of source resistance for four different transmission line lengths (2cm, 5cm, 15cm, 25/30cm), and for four different types of load capacitance (5pF, 10pF, 15pF, 20pF). Results are displayed first for the three
analytic models (ramp, step, and piecewise linear models), and then for the HSPICE model.

Figures 23 and 24 indicate the estimation error predicted by the ramp, step, and piecewise linear transmission line models. Figure 23 contains the graph set for the case where the source capacitance is fixed at 10pF, and Figure 24 the set for the case where the source capacitance equals the load capacitance. In both situations, the source resistance is matched to the characteristic impedance of the transmission lines.

Some general trends are obvious after examining these graphs. First of all, the shortest length lines (2cm to 5cm) experience greater estimation error. Multiple reflections get buried in the signal edges when lines are very short, masking the source voltage step, and making measurements less accurate. Another trend is a general increase in the error as the source resistance, and load capacitance increase. In fact, the worst case error tends to occur with the highest resistances and capacitances. This error is compensated somewhat by the fact that such configurations tend to be inherently slow so that the increased error has less of an impact on performance than it would on a faster circuit. A final but important trend is the weak dependence of estimation error on transmission line length for lengths beyond 5 to 10cm.

From the graphs, it is clear that the ramp, step and piecewise linear models agree on the estimated error to within 100ps or so (as they should). A more demanding comparison, however, is with the HSPICE results presented in Figures 25 and 26. These figures are for nearly identical circumstances as those for the analytic models. They contain the estimation error predicted directly from measurements of the source and load waveforms, as well as measurements taken via the HSPICE model's comparators. The direct measurements agree quite well with the analytic models, while the comparator measurements have a similar spread, but appear shifted closer to the origin, resulting in smaller worse case error.
Finally, two HSPICE graph sets are presented for *unmatched* transmission lines (Figures 27 and 28). Here the source resistance, $R_S$, equals 90% and 110% of the characteristic impedance, $Z_0$. An important observation is that the worst case error is still approximately the same as for the matched case; however, there is a lot more spread between the predicted errors for different transmission line configurations.

What conclusions can be drawn from these graphs regarding the validity of the CBC's delay estimation technique? When dealing with medium to long transmission lines, the analytic and HSPICE results indicate that the delay-estimation error is generally less than 250ps to 400ps. For short lines, however, the error increases to 500ps or 600ps, (except in a few cases where high capacitance and resistance push the error to 1ns).

Note that once the source and load configuration is fixed, the predicted estimation error stays nearly constant across all but the shortest lines as the transmission line length is varied. This bodes well for the CBC's delay estimator, since each output of the CBC is likely to have a configuration roughly similar to the other outputs, except for transmission line length. The CBC's performance depends not on obtaining a precise match to the absolute delay, but on ensuring the *relative* delay between all output transitions is small upon their arrival at their loads. HSPICE results indicate that the relative delay is less than 200-240ps for nearly all trace lengths greater than 5-10cm.
Figure 23. Propagation delay estimation error as predicted by the ramp, step, and piecewise linear models for the case where $C_S$ is fixed at 10pF. The error for different load capacitances and line lengths is plotted against the resistance of the matched driver/signal trace.
Figure 24. Propagation delay estimation error as predicted by the ramp, step, and piecewise linear models for the case where $C_S$ equals $C_L$. The error for different source/load capacitances and line lengths is plotted against the resistance of the matched driver/signal trace.
Figure 25. Propagation delay estimation error as predicted by the HSPICE model (directly from the source/load waveforms and using the comparator outputs) for the case where $C_S$ is fixed at 10pF, and where $R_S$ is matched to characteristic impedance.
Figure 26. Propagation delay estimation error as predicted by the HSPICE model (directly from the source/load waveforms and using the comparator outputs) for the case where $C_S$ equals $C_L$, and where $R_S$ is matched to the characteristic impedance.
Figure 27. Propagation delay estimation error as predicted by the HSPICE model (directly from the source/load waveforms and using the comparator outputs) for the case where $C_S$ is fixed at 10pF and where $R_S$ equals 90% and 110% of $Z_\sigma$. 
Figure 28. Propagation delay estimation error as predicted by the HSPICE model (directly from the source/load waveforms and using the comparator outputs) for the case where $C_S$ equals $C_L$ and $R_S$ equals 90% and 110% of $Z_0$. 
3.6 Expected Propagation Delay

So what are the true propagation delays that can be expected for a given transmission line? This information is necessary for designing the CBC since the CBC will have to make measurements of the source waveform over the entire source-to-load round-trip delay. The graphs on the following pages attempt to address this question. They are based on a propagation delay of 5.8ns/m which is typical for modern circuit boards ([19], Appendix A).

There are two sets of graphs: the set shown in Figure 29 is for the analytic models, and the set of Figure 30 is for the HSPICE model. The transmission line configurations vary from "slow" to "fast" as one moves from top to bottom across each graph set. The graphs on the right are for a source capacitance fixed to 10pF, while the graphs on the left are for a source capacitance equal to the load capacitance. It is easy to see that both the analytic and HSPICE models give nearly identical results.

An early design decision for the CBC was to attempt to de-skew lines with lengths up to 40-45cm for the "medium" speed transmission line case (middle row of each graph set). Although the graphs do not extend to 45cm, there is an obvious linear trend which can be used to extrapolate the results to this zone with little error. This extrapolation leads to the requirement that the CBC must make measurements for propagation delays of up to 4ns - equivalent to round-trip delays of 8ns.

With this propagation delay information in hand, and with confirmation of the validity of the delay estimation technique used in the CBC, it is now time to begin discussing the design of the CBC's architecture and circuits. This is the focus of the upcoming chapters.
Figure 29. Propagation delay versus transmission line length for typical "slow", "medium", and "fast" transmission line configurations. These results were obtained from the analytic models.
Figure 30. Propagation delay versus transmission line length for typical “slow”, “medium”, and “fast” transmission line configurations. These results were obtained from the HSPICE model.
Chapter 4 presents the system and gate-level design of the clock buffer chip (CBC) built for this thesis. In general, low-level circuit design details, including VLSI layout issues, are deferred until the following chapter.

An overview of the high-level structure of the CBC is presented in section 4.1, followed by a description of the DLL and clock bay architectures in sections 4.2 and 4.3. Section 4.4 discusses delay generation, while section 4.5 describes the phase detector and tap controller of the DLL. Sections 4.6 and 4.7, detail the propagation delay estimator, and the automatic impedance matching circuit. Finally, section 4.8 describes the input and output structures used in the CBC to control and observe the internal logic.

4.1 Clock Buffer Chip (CBC) Overview

The CBC consists of a scalable number of clock outputs that are synchronized to a reference clock. Each output block is called a clock bay. As shown in Figure 31, several clock bays, and one fixed delay are arranged together to form the clock buffer chip. The
prototype chip built for this thesis was implemented using three clock bays, but there is no fixed limit on the number of clock bays.

Figure 31. High-level structure of the CBC.

Both the input clock and the reference clock are distributed to all clock bays. Furthermore, an impedance control word (ICW) flows from one of the clock bays, called the "impedance master", to the other bays.

The fixed delay provides a reference clock. Each clock bay synchronizes its output to the reference clock such that all the output signals arrive at their intended destinations with low skew relative to one another. Within each clock bay, a delay-locked loop (DLL), and a propagation delay estimator circuit work together to achieve this synchronization. To control reflections, an automatic impedance matching circuit is also used, and it ensures that the impedances of the output drivers are well matched to the impedances of the output signal traces.

Automatic impedance matching requires that impedance measurements be made at each output; however, as explained in section 4.7, impedance measurements can only be made reliably on long signal traces. Thus, the impedance master is programmed externally to
correspond to a clock bay driving a long signal trace. It generates impedance control settings for itself and for all clock bays driving short signal traces.

The remaining clock bays all drive long lines, so, ideally, they should have the option of making their own impedance measurements or, alternatively, just using the ICW provided by the impedance master. This choice should either be decided externally by the user through dedicated inputs or, automatically by the clock bay itself by using its built-in propagation delay estimator (see section 4.6) to check if the line is long enough for automatic impedance measurements. (This automatic approach still needs to use the impedance master's ICW initially to allow the delay estimator in each bay to start up properly. Then, if the delay estimator reports that the signal trace is long enough, local impedance measurements can begin and the impedance master's ICW can be abandoned.)

In an effort to reduce pin count and area, the prototype CBC takes a simplified approach to the above. It allows the user to select which clock bay is the impedance master, and then forces all bays to use the impedance master's ICW (regardless of the length of their signal traces). Since the impedance of any signal trace is relatively well matched to that of the other traces on a circuit board (as a result of being manufactured under the same conditions), this approach is a reasonable way to reduce the cost of the prototype clock buffer chip.

4.2 DLL Architecture of the CBC

Figure 32 presents the DLL architecture used in the clock bays of the CBC. Feedback is used to phase-lock two signals via an adjustable delay. The phase detector generates an error signal, $E$, that is integrated (i.e. low-pass filtered) and used to adjust an adjustable delay until the target signal, $X$, is aligned with the reference signal, $Y$. The unknown variable delay corresponds to the output driver's delay together with the propagation delay from the CBC to the target IC.
Using the large fixed delay in Figure 32 is optional in a DLL, but it offers the advantage of always having a clock edge available at $X$ to compare against $Y$ - even at start-up. When the fixed delay is not present, the DLL must absorb one clock cycle until a signal is available at $X$. The disadvantage of using a fixed delay is that the reference signal is only available on-chip, but such a limitation is often acceptable for a clock buffer device.

**Figure 32. DLL architecture of the CBC.**

DLLs can be implemented in a digital or analog manner. All-digital DLLs are used in the CBC. Variable delays are implemented using tapped-delay lines controlled by digital tap-control words. Phase detection uses a lead-lag type phase detector that drives an integrator formed by an up/down counter. Figure 33 shows the phase detector waveforms for different $X$ and $Y$ inputs. More will be said about the circuits making up the DLL in upcoming sections.

**Figure 33. Example lead-lag phase detector waveforms.** Here $X$ is sampled at every rising edge of $Y$. The result is an error signal $E$ that directs the integrator to increase or decrease the variable delay to bring $X$ and $Y$ into alignment.
A digital approach was selected for several reasons. As will be explained in section 4.6, propagation delay estimation is simplified since it is easier to perform the division-by-two required to obtain the one-way propagation delay from the round-trip delay estimate. Also, digital hardware description languages can be used to model and simulate different circuit configurations rapidly. In addition, when noise is present, digital impedance-control and tap-control signals are easier to distribute on-chip to where they are used, and off-chip to where observers can monitor the circuit’s behaviour.

Of course, digital approaches have some disadvantages. They often require more area than analog techniques due to the binary nature of their internal signals. They also consume more power, because of their rail-to-rail signal swings.

4.3 Clock Bay Overview

Figure 34 depicts the high-level block diagram for a clock bay. Logic for the DLL is present, as well as circuitry devoted to automatic impedance control and propagation delay measurement. Understanding the operation of the clock bay at this level primarily involves an understanding of the DLL, so the automatic impedance control and delay measurement circuits will not be discussed in detail until later.
Figure 34. Clock bay block diagram. The clock bay circuitry is contained within the larger dashed outline. The delay of TDL1 is controlled by a 6-bit binary encoded control signal, while the delay of TDL2 is controlled by a 26-bit one-hot encoded signal.

The left half of the DLL generates the reference clock by delaying the input clock through a fixed delay of 78 delay elements. This reference clock is then applied to one input of the phase detector. The second input to the phase detector is also derived from the input clock, but it must first traverse a 64 element tapped delay line (TDL1), an output buffer, a comparator, and finally a 26 element tapped delay line (TDL2).

TDL1 is the primary variable delay of the DLL. It is controlled by the phase detector and tap controller. TDL1's delay is adjusted to synchronize the signals reaching the input ports of the phase detector. Adjusting TDL1 adjusts the phase of the signal leaving the clock bay at the output pad, as well as the signal reaching the phase detector's rightmost input.

If TDL2 was not present, the clock bay would synchronize its output signal (at the PAD) to the reference clock. However, TDL2 is used to introduce a delay that approximates the delay from the clock buffer to the target IC. In this way, the phase detector receives a
signal that approximates the signal reaching the target IC. The phase detector can then adjust \textit{TDL1} to align this signal to the reference clock. Ideally, the actual signal arriving at the target IC should then also be aligned with the reference clock.

Each clock bay resets to a state where both \textit{TDL1} and \textit{TDL2} are set to half their maximum delay. For short signal traces, the propagation measurement circuit will reduce the delay of \textit{TDL2}, and as a result the phase detector will have to increase the delay of \textit{TDL1} to maintain phase alignment. Conversely, for long signal traces, \textit{TDL2}'s delay will be larger than its reset level, and as a result \textit{TDL1}'s delay will become smaller. The net effect is that the clock signals from each clock bay arriving at their destinations - both near and far - will be synchronized to the reference clock.

It is clear from Figure 34 that the input to \textit{TDL2} is not taken directly from the output pad of the clock bay. Instead it is taken from a comparator that compares the output pad’s voltage against a threshold set to 1/4 the supply voltage. As shown in Figure 35, this low-threshold comparator removes the wide voltage step that appears on signal transitions due to the source-terminated outputs of the CBC. This is the same voltage step that is used to derive an estimate of the propagation delay from the CBC to the target IC (see Chapter 3).
Figure 35. Idealized waveforms for the output of TDL1, the clock bay, and the low threshold comparator (assuming the output driver and comparator have zero delay). The phase detector only attempts to lock onto the rising edge leaving TDL2 so the shifted falling edge of the comparator output has no effect on performance.

An alternative input to TDL2 is the output of TDL1 since this signal has well defined edges. Unfortunately, using such an input removes the output buffer from the loop of the DLL, and any potentially large variations in the delay of the output buffer cannot be tracked. It is for this reason that the previous approach (based on using a low-threshold comparator to provide the TDL2 input) is preferable.

4.3.1 Resetting the CBC

Resetting the clock bay is an important and somewhat involved process. Two reset signals, reset1, and reset2, are required for the proper initialization of the clock bay.

The clock signals flowing around the DLL are what drive the various blocks of the clock bay including the phase detector, propagation delay estimator, and impedance matching circuitry. Thus, care must be taken to ensure that the DLL does not start-up in a state that prevents clock signals from flowing around the loop of the DLL, as this would lock-up the entire circuit. To avoid this problem, the tap-control words for TDL1 and TDL2, as well as the impedance control word sent to the output buffer, are reset to sensible values.

In addition, accurate propagation delay measurements depend on properly terminated signal traces. So, before propagation delay measurements can begin and be assigned to
TDL2, the automatic impedance control circuit must be given enough time to lock the impedance of the output driver to that of the output trace.

Initially, both reset signals are activated. This asynchronously sets the tap controllers of TDL1 and TDL2 to half their maximum values. This also asynchronously sets the impedance of the output driver to some intermediate value (48Ω). Clock signals are now guaranteed to flow through the DLL, and they are used to load the reset settings synchronously into registers in the tap and impedance control units.

Afterwards, reset1 is disabled, and the automatic impedance matching circuit begins operating normally. However, the rest of the clock bay is maintained in its reset state by reset2. This is necessary to give the automatic impedance matching circuit time to perform the impedance lock.

When the impedance lock is complete (typically after 30-60 clock cycles), reset2 is finally disabled, and the DLL and propagation delay measurement circuits begin operating normally.

4.3.2 Loop Equations

Various relationships are of interest with regards to the clock bay DLLs. In particular, an expression relating the loop delays in the phase-lock state is valuable for sizing the delay lines in the DLL (see section 4.3.3). In addition, an equation representing the maximum clock period of the CBC is necessary to understand its performance limitations. Figure 36 presents the DLL delays that will be incorporated into the following analysis.
Figure 36. Clock bay DLL and its associated delays (shown in bold). Both $T_{\text{var}}$ and $T_{\text{prop}}$ are variable delays that have maximum values of $T_{\text{var MAX}}$ and $T_{\text{prop MAX}}$. $T_{\text{update}}$ represents the time required to update the tap setting of TDL1 after a rising edge of the reference clock.

A relationship between $T_{\text{fixed}}$, $T_{\text{var}}$ and $T_{\text{prop}}$ can be obtained from Figure 36. This relationship will be used later when discussing delay line sizing. $T_{\text{fixed}}$ represents the delay in the left portion of the DLL. It consists of many delay elements followed by a buffer that allows the reference clock to drive the phase detector and the TDL1 tap controller. In order to obtain phase lock, $T_{\text{fixed}}$ must equal the delay through the right portion of the DLL, which is given by the sum of $T_{\text{buffer}}$, $T_{\text{comp}}$, and the delay through TDL1 and TDL2. In particular, just after reset, equation (4.1) summarizes the approximate delay relationship between the two halves of the DLL.

$$T_{\text{fixed}} = \frac{T_{\text{var MAX}}}{2} + T_{\text{buffer}} + T_{\text{comp}} + \frac{T_{\text{prop MAX}}}{2}$$  (4.1)

The minimum operating period of the DLL is also an important consideration. It is determined by the minimum time required to update TDL1 after a clock edge exits it. TDL1 must be completely updated before the next clock edge - either falling or rising - reaches its active tap. Updates are governed by the reference clock since it clocks the phase detector and tap controller. Thus, when phase locked, the minimum clock period is
given by equation (4.2). A factor of two is present since the DLL only has half a clock cycle to update the tap before the next clock edge reaches the active tap of $TDL1$.

$$T_{min} = 2(T_{buffer} + T_{comp} + T_{prop} + T_{update})$$  (4.2)

Equation 4.2 is also valid when several clock bays are examined as a group; however, the propagation delay, $T_{prop}$, must now be set to that of the longest clock trace across all bays, otherwise known as $T_{propGROUPMAX}$. This is summarized by equation (4.3).

$$T_{min} = 2(T_{buffer} + T_{comp} + T_{propGROUPMAX} + T_{update})$$  (4.3)

### 4.3.3 Delay Line Sizing

Delay line sizing requires that one determine the size of the three delays - $T_{varMAX}$, $T_{propMAX}$, and $T_{fixed}$ - that are at the discretion of the designer. All other delays are inherent in the actual circuit implementation and the only requirement is that they be minimized.

As discussed, in section 3.6, the maximum one-way propagation delay accommodated by the CBC is approximately 4ns. $T_{propMAX}$ is set equal to this delay. In terms of delay elements, which have a delay of roughly 150ps, the total delay translates to approximately 26 delay elements.

The primary delay line, $TDL1$ must accommodate the entire range of variation in propagation delay so it must be at least as large as $T_{propMAX}$. Furthermore, it must also accommodate the worst case variation in delay between the two paths of the DLL that lead to the phase detector. Process, voltage, temperature, and loading (PVTL) differences between the two paths are responsible for these variations.

PVTL variations in delay are limited by the fact that the delay elements used in each delay line are identical, and so variations will track well between the two paths of the DLL.
Nevertheless a substantial amount of delay on the right side of the loop (containing $TDL1$, the output buffer, etc.,) is not associated with delay elements and consists of buffers and comparators that have a different structure from delay elements. The delay of these structures may not track so well between the two halves of the DLL.

A conservative value of +/-3ns was selected for the possible range of variation in delay (i.e. 6ns total variation) between the two halves of the DLL. Thus the length of the main delay line, $TDL1$, was set to 10ns (i.e. $T_{propMAX}$ plus 6ns). This corresponds to approximately 64 delay elements.

Finally, the fixed delay was determined from equation (4.1) after estimating suitable delays for $T_{buffer}$ and $T_{comp}$. Circuit simulations (using HSPICE [33]) were used to fine tune the fixed delay so that the DLL was nearly phase locked in its reset state. Fine adjustment was necessary since there are many hard-to-estimate delays present in the loop, including ones not indicated in equation (4.1) (due to the output delay of $TDL1$ and $TDL2$). The final size of $T_{fixed}$ was set to approximately 11.5ns (i.e. 76 delay elements) based on these simulations.

4.4 Delay Line Design

One of the most prevalent structures in the CBC are delay lines. They are constructed from cascades of delay elements. This section will discuss the design of the delay elements and the delay line structures of the CBC.

4.4.1 Delay Elements

There are many ways to implement a delay element. The primary requirement is that the delay element provide as small a delay as possible, and that it be very compact. Cascades of these delay elements can then be used to implement delays that are precise multiples of the individual element's delay. A good candidate for such a delay element is a CMOS
inverter. In the 0.8µm process used to implement the CBC, inverter delays can be as small as 220-150ps.

The variable delay lines used for TDL1 and TDL2 are tapped after every delay element to provide a variable exit point from the delay line, thereby implementing a variable delay. However, tapping the delay line after every delay element requires a non-inverting delay element, which prevents inverters from being used.

One solution is to transmit both the signal and its complement through two parallel inverter based delay lines. Taps can then be made from alternating delay lines to obtain an output with the same polarity at every tap. Unfortunately, such a structure is very sensitive to variations in delay between the two delay lines. If the delay elements in one delay line are slightly faster than in the other, this difference will be magnified when the delay elements are cascaded, and will lead to skew.

A differential delay element, however, can eliminate this skew since only one delay line is required to transmit the differential signals. Such a delay element is shown in Figure 37. To prevent loading on the tap ports from influencing the delay of the differential inverter, two standard inverters are cascaded and connected to each differential output. These inverters buffer the internal signals and isolate the delay line from external circuitry.
Figure 37. Delay element cell and its relationship to neighbouring delay elements. To match delay variations across the CBC, the same kind of delay element is used in all delay lines. This means that all delay lines contain tap ports even though they are not always used (see section 4.4.3).

4.4.2 Single-to-Differential Converter

A single-to-differential converter circuit generates the complementary inputs required for each differential delay line. As shown in Figure 38, the circuit consists of an inverter driving two paths. One path performs an even number of inversions, and the other path an odd number of inversions. The transmission gate and inverters are sized to ensure equal delay through both paths.

Figure 38. Single-to-Differential converter.

4.4.3 Delay Lines

Fixed, tapped, and sampled delay lines are used in the CBC. These three kinds of delay lines will be discussed in the following sections.
4.4.3.1 Fixed Delay Lines

Fixed delay lines are straightforward to implement. They consists of a single-to-differential converter followed by a cascade of delay elements. Outputs of the final delay element are buffered using standard inverters, and then applied to the destination circuit (e.g. phase detector and TDL1 tap controller). Only the tap ports of the final delay element are used in a fixed delay line. The other tap ports are left unconnected.

4.4.3.2 Tapped Delay Lines

Tapped delay lines are used to implement programmable delays in the CBC. They are identical to fixed delay lines, except that each tap port is connected to an output selection circuit. Selection circuits consists of large multiplexors that select a single tap for output. The output selection is under the control of a digital tap-control word (TCW). NMOS pass transistors are used to implement the multiplexors in the CBC due to their small area, and the relatively small load they present to the TCW.

TDL2 (i.e. the delay line representing the propagation delay estimate) uses a 26-bit one-hot encoded TCW to select one of 26 output taps as shown in Figure 39. Use of a one-hot encoded TCW is a natural choice due to the manner in which the propagation delay estimator circuit computes the propagation delay (see section 4.6).

The tap selection circuit consists of an array of NMOS pass transistors. Each pass transistor has one node connected to a buffered tap output from a delay element, and the other node is shared with all pass transistors. The extra buffering prior to each pass transistor is needed to allow each tap to drive the large capacitance associated with the shared output node.

The TDL2 tapped-delay line uses both tap ports on each delay element to generate a differential output from the delay line. In effect, there are two output selection circuits that connect to a comparator to produce the final output. A major benefit of using a comparator...
is that it corrects for the reduced signal swing due to the NMOS pass transistors, and it also improves the edge transition times of the output.

Figure 39. TDL2 delay line arrangement.

TDL1, the primary delay line in the clock bay DLL, has a longer, but otherwise identical cascade of a single-to-differential converter and delay elements. However, the tap selection circuitry is different.

TDL1 uses a 6-bit binary-encoded tap control word to select one of 64 delay elements, since, with so many delay elements, using a one-hot encoded TCW is not practical. A binary-encoded tap control word permits hierarchical tap selection. This offers the benefit of less output delay, because the tree-like structure of the tap selection circuit reduces the capacitance that must be driven by individual tap outputs. Taps are selected using binary selection across groups of 2, 4, 8, 16, 32, and then 64 delay elements as shown in Figure 40. Inverters are used inside the tap selection "tree" to reduce the RC delay through the NMOS pass transistor network.

As with TDL2, two tap selection circuits are used to generate a differential output from the delay line. This differential output is sent to a comparator that restores full signal swing, and improves the output rise/fall times.
Figure 40. Tap selection circuit used in the TDL1 delay line. The circuit accepts a 6-bit differential tap control word, and 64 signals from the delay elements. Two such tap selection "trees" are used in the TDL1 delay line to generate the differential output that is fed to the final output comparator.

Future implementations should investigate an alternative tap selection circuit that is similar to that used in addressing digital memories. In this case, the output taps are arranged in rows and columns and are selected by word and bit-line addresses. Such a decoding is used in folded resistor string digital-to-analog converters ([28], section 12.1), and may offer reduced output delay.

4.4.3.3 Tap Control Word Update Timing

It is imperative that the tap control word settings of a tapped-delay line never change while there are signal transitions propagating through the output structures of the delay line. If the TCW settings happen to change at this critical moment, glitches may be introduced into the output.
For non-hierarchical tap selection, glitch free outputs can be guaranteed if the TCW settings switch anytime before a signal edge travelling down a delay line gets close to the active output tap. The signal edge should be at least one TCW update interval away from the active tap. (The TCW update interval is simply the RC delay experienced as the tap controller drives a new TCW setting.)

For hierarchical output structures, the situation is more complicated. While it is true that only one delay element drives the output node, many different delay elements drive various internal nodes within the output structure. It is this situation that complicates the timing.

A worst case scenario is presented in Figure 41, but for a smaller version of the hierarchical tap selection circuit. For clarity, internal inverters are not used, and pass transistors are represented as dots. A negative signal transition is propagating through the delay line in the figure, as the tap control word switches from 10000₂ to 01111₂. What restrictions are needed on the tap update timing to avoid glitches?

Figure 41. Example circuit consisting of a 32 element delay line and its tap selection structure. Node values for the 2nd to 5th tier in the output hierarchy are marked for the instant shown.
When the TCW switches, the selected path switches from path PA to PB. However, as node NC demonstrates, parts of path PB close to the output initially may be charged to a different polarity than the old output. This can happen even if the signal edge in the delay line is far away from the active tap! In this example, path PC is the cause of this problem.

Charge sharing between node NC and the output may now occur when the TCW switches, which may lead to a glitch on the output. This is aggravated by the fact that the lower bits of the TCW experience more load and tend to switch slower than the upper bits. Thus, a strong signal may not be present to charge up the output node quickly to the right polarity.

The simplest solution is to avoid changing the TCW while there is any signal transition propagating down the delay line. However, this approach is overly conservative. A better alternative is to make sure that a strong signal is available close to the output to charge it to the right polarity as quickly as possible.

This can be accomplished by making sure the transition in the delay line proceeds the end of the update of the TCW by enough time to ensure that the lower three tiers of hierarchy along path PB have the correct old value. If this is done for the circuit of Figure 40 (i.e. not the example circuit), then the selected inverter in the middle of the figure will have the correct old value at its input so it can quickly charge up the output node to prevent any glitch. Using an interval of 8 to 10 delay elements (1.5 to 2ns) is adequate. Note that the T_update variable in equation (4.1) must now include this safety margin as well as the TCW update time.

4.4.3.4 Sampled Delay Lines

Sampled delay lines are delay lines where some or all delay elements are connected to flip-flops (via their tap ports). All flip-flops are triggered at the same time, and they sample the signal throughout the delay line. They are used in the propagation delay
estimator to time the duration of various events in terms of delay element delays. At the start of an event, the input to the delay line is toggled, and, at the end, the flip-flops sample the delay line to see how far the input transition has propagated down the delay line. More will be said about this in section 4.6.

4.5 Phase Detector and DLL Tap Controller

As mentioned in section 4.2, the phase detector in the clock bay DLL is a "lead-lag" type phase detector which, in its most basic form, is implemented by a single D-flip flop. The output of the D-flip flop indicates to the tap controller whether it should increase or decrease the variable delay of the DLL. The DLL continuously jitters between two taps when phase locked.

To work well in the DLL, the phase detector must be able to detect changes in phase due to delays as small as one delay element. HSPICE simulations indicate that the phase detector can respond to phase differences due to delays as small as 100-150ps. This resolution is a little smaller than the delay of one delay element.

The phase detector used in the DLLs of the CBC is somewhat more complicated than the lead-lag type mentioned above. It not only outputs a lead-lag signal, but also a hold signal that tells the tap controller when the DLL is locked. When the hold signal is active, the tap controller does not change its tap settings, and this reduces short-term jitter.

Identifying the phase locked condition requires two samples of the target clock for every rising edge of the reference clock. One sample is taken immediately, and the other taken one inverter delay later. When the first sample is '0' and the second '1', the hold signal becomes active. Figure 42 indicates the circuit that implements the phase detector.
Figure 42. Phase detector schematic. Master-slave flip-flops are used for sampling, and are formed by “P” latches which latch when the clock input is high, and “N” latches which latch when the clock is low. The phase detector has 1-1/2 clock cycles of latency.

In actuality, the hold signal is not fed directly to the tap controller, but is sent to a finite state machine (FSM) that governs how often the tap controller updates the primary variable delay line (TDLI). Tap adjustments cannot be made every clock cycle, because the phase detector and tap controller have 4 clock cycles of latency. So, every five clock cycles the FSM signals the tap controller to update the TDLI tap control word. If the hold signal is active, then the FSM skips the update since the tap settings do not need to be modified.

The FSM’s state diagram is shown Figure 43. The state encodings are one-hot encoded so that the FSM structure becomes very regular, much like a shift register. This simplifies the physical layout at the expense of a few additional flip-flops. These flip-flops are negative edge triggered master slave flip-flops - formed using an “N” latch followed by a “P” latch - in order to function correctly with the “P” latch output of the phase detector.
Figure 43. Tap controller update FSM. The states are one-hot encoded so that state $A=100000_2$, state $B=010000_2$, etc. The update signal (state bit "$b_C$") goes high every 5 clock cycles (in state "C") if hold is not active.

Figure 44 shows the block diagram of the phase detector, tap controller, and update FSM block. In addition, there is a flip-flop for synchronizing the off-chip reset signal ($\text{reset2}$ from section 4.3.1) with the internal clock so that the reset signal does not violate any flip-flop setup and hold times when applied to the FSM and tap controller.

The tap controller also accepts an asynchronous reset signal ($\text{resetl}$ from section 4.3.1) that asynchronously maintains the outputs of the tap controller in their reset state to ensure that TDL1 receives a valid TCW at start-up (i.e. before any clock edges reach the phase detector).

State Encoding:

\[ b_A b_B b_C b_D b_E b_F \]

- $b_A = b_F$
- $b_B = b_A$
- $b_C = \overline{\text{hold}} \text{ AND } b_B$
- $b_D = \overline{\text{hold}} \text{ AND } b_B$
- $b_E = b_C \text{ OR } b_D$
- $b_F = b_E$
- $\text{update} = b_C$
4.5.1 Tap Controller

A 6-bit up-down counter is the basis for the tap controller. The counter is formed from a 6-bit ripple carry adder that adds or subtracts 1 from the existing count if update is active. Subtraction is performed by adding the 2's complement of 1 (i.e. $111111_2$) to the existing count. Addition adds 1 to the count by setting the carry-in bit to 1. Addition or subtraction is selected using the $\text{down} \rightarrow \text{up}$ signal.

An optimized ripple carry adder based on [1], pp. 515-520, makes use of two types of alternating 1-bit full adder circuits to remove extra inversions in the ripple carry chain. Type 1 adder circuits take uncomplemented inputs ($A, B, \text{CIN(carry in)}$) and output $\overline{\text{COUT}}$ (carry out) and $\overline{\text{SUM}}$. Type 2 adder circuits work in a complementary fashion, and use complemented inputs ($\overline{A}, \overline{B}, \overline{\text{CIN}}$), and output uncomplemented CARRY and SUM signals. Thus an alternating cascade of these two circuits means that the carry signal generated in each type of adder is directly usable by the next stage up the carry chain without extra inversions.

Each full adder circuit first computes the carry output and then uses this result to compute the sum. Type 1 full adder equations are given in equation (4.4) and Type 2 equations in equation (4.5) (where "\(+\)" represents OR, and "\(\cdot\)" AND). Note that the carry generation circuits are quite fast because they can be implemented using static CMOS with no more than two series transistors in either the pull-up or pull-down paths.

\[
\begin{align*}
\overline{\text{COUT}} &= A \cdot B + \text{CIN} \cdot (A + B) \\
\overline{\text{SUM}} &= A \cdot B \cdot \text{CIN} + \overline{\text{COUT}} \cdot (A + B + \text{CIN}) \\
\text{COUT} &= \overline{A} \cdot \overline{B} + \overline{\text{CIN}} \cdot (\overline{A} + \overline{B}) \\
\text{SUM} &= \overline{A} \cdot \overline{B} \cdot \overline{\text{CIN}} + \text{COUT} \cdot (\overline{A} + \overline{B} + \overline{\text{CIN}})
\end{align*}
\] (4.4)

Following each 1-bit adder is a multiplexer that selects either the new count (as computed according to the $\text{down} \rightarrow \text{up}$ counter input), or the old count that is fed back from the counter flip-flops. The $\text{update}$ signal from the phase detector FSM performs this selection.
The inverter at the output of the static CMOS multiplexer has additional transistors (implementing a NAND or a NOR) that are controlled by the reset2 signal. They permit the output to be clamped either high or low so that the counter can be forced into its reset state of 32 (100000₂). When a falling clock edge arrives, the multiplexer output is registered into the counter's flip-flops.

The flip flops after the multiplexor are formed from latches. After the first latch, each output signal branches into two paths to generate the output and its complement. These signals are buffered in a final stage that also permits the count to be asynchronously forced to its reset state (as mentioned in section 4.3.1). Asynchronous reset is required since the latches do not have an asynchronous reset, and the tap control word must be properly established before clock edges can flow around the DLL to implement the synchronous reset of the counter. Figure 45 presents the block diagram for the tap controller.
Figure 45. Tap controller schematic. Note that the tcw₀ and tcw₀ signals each split into two paths to allow for extra buffering. Tap controller latency is 1-1/2 clock cycles.
4.6 Propagation Delay Measurement

Propagation delay measurement attempts to measure the duration of the voltage step that accompanies every edge transition on the output pad of the CBC. This step is due to the source terminated clock buffer driving the output signal trace. The width of the step indicates the approximate round-trip propagation delay from the clock buffer chip to its destination. Measurements are made every rising edge of the outgoing clock signal.

Two comparators are used to generate signals that indicate the start and end of the voltage step. A low threshold comparator is triggered at 1/4 the supply voltage, and it generates a signal ($V_{\text{low}}$) that indicates the start of the voltage step. A high threshold comparator, triggered at 3/4 the supply voltage, is used to determine the end of the voltage step ($V_{\text{high}}$). Figure 46 graphically depicts the relationship of the comparator signals to the voltage step.

![Diagram showing voltage levels and thresholds](image)

*Figure 46. High and low threshold comparator outputs ($V_{\text{low}}$ and $V_{\text{high}}$) for a rising transition on the clock bay output ($V_{\text{src}}$).*

As mentioned in section 4.4.3.4, a sampled delay line is used to time the duration of the voltage step. The timing circuit actually uses two (nearly) equal length delay lines - one fixed and one sampled - to obtain the estimate as shown in Figure 47.

$V_{\text{low}}$ is toggled to a '1' on a rising edge of the clock bay output ($V_{\text{src}}$). The $V_{\text{low}}$ transition travels down a fixed delay line, and eventually clocks the sample register that is part of the sampled delay line.
$V_{\text{high}}$ toggles to a '1' one voltage step interval after $V_{\text{low}}$ does, and this transition travels down the sampled delay line. When $V_{\text{low}}$ finally clocks the sample register, the register captures a picture of how many delay elements $V_{\text{high}}$ is delayed by relative to $V_{\text{low}}$. The number of low register bits at the end of the sample register indicates the round-trip propagation delay in terms of delay elements.

To obtain the one-way propagation delay (instead of the round-trip delay), division-by-two is performed by only sampling every second delay element of the 52 element sampled delay line. Half a delay element of resolution is lost performing this simplistic division, but since “half delay elements” are not available in the TDL2 delay line this cannot be avoided. Some potential techniques for producing fractional delay-element delays are described in [22] and [23], and may be useful for future work.

![Simplified block diagram for the propagation delay measuring circuit.](image)

The outputs of the sample register are then applied to a leading one detector that identifies the transition bit in the register and outputs a signal with that bit set high and the other bits
set low. This one-hot encoded signal is registered and used as the tap control word for the
\textit{TDL2} delay line. Reset logic is also included in the leading one detector to force the \textit{TDL2}
tap control word to reset with only the middle tap high. This fixes the \textit{TDL2} delay line to
half its maximum delay at start-up.

Measurements are made once every clock cycle and pipelined through to the tap control
word register. From Figure 47, it is clear that the update latency is one clock cycle from
the moment the delay line is sampled until the \textit{TDL2} tap control word is updated.

\textbf{4.6.1 Leading One Detector}

An architecture for the leading one detector is presented below. It consists of a series of
logic gates that search for a ‘110’ pattern in the output from the sample register as shown
in Figure 48.

The simplest leading one detector can be implemented by searching for a pattern of ‘10’.
This simplified detector is easy to construct, but may be more prone to error if the
transition region is corrupted by noise. It is very important that the detector not be fooled
and identify more than one leading one since this would activate more than one tap on the
\textit{TDL2} delay line.

Pattern matching across more bits improves the robustness of the leading one detector, but
it also leads to a slower circuit. A compromise was reached by looking across three bits
since a three-bit scan will identify a single leading one even if the transition region looks
like ‘111...11010...000’ instead of the clean transition pattern of ‘111...111000...000’.

Three bits provides an adequate measure of robustness because the transition region in the
delay line typically does not straddle more than one or two delay elements, so the
transition region is expected to be very well defined (especially since only every 2nd delay
element is sampled by the sampled delay line).
**Figure 48.** Leading one detector and its surroundings. The detector is contained within the large dashed box. The three input NAND gates in the first levels of combinational logic search for a '110' pattern in the sample register. NOR gates are used to reset all bits low except bit "s12". Bit "s12" is reset high using a NAND gate.

### 4.7 Automatic Impedance Matching

Automatic impedance matching is used to control reflections on the output pins of the CBC and to ensure the propagation delay estimator functions properly.

A well matched source-terminated output has a transient voltage step positioned exactly half-way between the power supply rails on every signal transition. As shown in Figure 49, if the output driver has a lower impedance than the signal trace, the magnitude of the rising edge voltage step (see equation (2.13)) will be above half the power supply voltage. If the driver impedance is higher than that of the signal trace, the voltage step’s voltage will fall below half the supply voltage.
Figure 49. Transient voltage step voltage for different output configurations. $R_S$ is the source resistance of the output driver, while $Z_o$ is the characteristic impedance of the signal trace. For falling edges, the situation is reversed with the $R_S < Z_o$ voltage step lying below the $1/2$ voltage threshold, and the $R_S > Z_o$ step appearing above the $1/2$ threshold.

It should now be apparent why the propagation delay estimator needs a well matched output driver. A well matched output permits the $V_{\text{low}}$ and $V_{\text{high}}$ comparators (comparing against $1/4$ and $3/4$ the supply voltage respectively) to identify accurately the start and end of the voltage step. This in turn leads to an accurate determination of the round trip propagation delay between the clock output and its remote destination.

If the output is not well matched, and the voltage step shifts close to either the low or high comparator threshold voltages, then this will either delay the activation of the low comparator, or prematurely activate the high comparator. Either outcome leads to a poor propagation delay estimate.

### 4.7.1 Automatic Impedance Matching Overview

Automatic impedance matching uses four major blocks to implement an impedance locked loop (ILL), as shown in Figure 50. This type of circuit has many analogies with the DLL discussed earlier.
The first ILL block is a comparator stage that contains the low and high threshold comparators (used earlier in the propagation delay estimator circuit), and a mid-level threshold comparator (used only for automatic impedance matching).

Following the comparator block is an impedance detector stage that samples the mid-level threshold comparator using clocks generated from delayed versions of the high and low threshold comparator outputs. These samples are made near the middle of each transient voltage step to determine if the step is located either above or below half the supply voltage. The measurements indicate the direction in which the impedance control word should be adjusted to improve impedance lock.

There is a lot of similarity between the impedance detector in an ILL and the phase detector in a DLL. However, one important difference is that measurements are made on both rising and falling edges of the clock bay output to permit independent impedance matching for both the pull-up (PMOS) and pull-down (NMOS) networks of the output driver. In essence, there are two ILLs in operation at the same time: one for PMOS transistors and one for NMOS transistors.

Figure 50. Simplified automatic impedance matching circuit block diagram.
The measurement stage provides inputs to the third block of the ILL which consists of PMOS and NMOS impedance control word (ICW) controllers. Like the tap controller in the DLL, the ILL controllers consist of 6-bit up-down counters. The outputs of the counters connect to tri-state buffers that drive the PMOS and NMOS ICW buses. As discussed in section 4.1, only one user-defined "impedance master" is enabled to drive these buses.

Output driver circuitry is connected to the ICW buses, and it forms the fourth and final block of the ILL. Each driver adjusts the impedance of its pull-up and pull-down networks according to the data present on the ICW buses.

### 4.7.2 ICW Bus Updates

The PMOS (i.e. pull-up) ICW cannot be updated while a rising edge is being driven by the output driver of the clock bay. It must be changed while the output is not changing, and preferably just after a falling edge of the output clock, because this is when the pull-up network is truly inactive. Similarly, the NMOS (i.e. pull-down) ICW bus is updated just after a rising edge of the output clock because this is when the pull-down path is inactive.

Figure 51 indicates graphically when the impedance master changes the data on the ICW buses. Both the impedance master’s TDLI and output waveforms are indicated as well as those for another clock bay using the master’s ICW to drive a signal trace half as long. The waveforms assume the clock bay DLLs are phase-locked.
4.7.3 Architectural Details

Figure 51. ICW bus update example. The round-trip propagation delay for the impedance master is 4ns, while the delay for the clock bay driving the shorter trace is 2ns. Since the DLLs are phase-locked, the impedance master's clock signal leaves TDL1 1ns earlier than the signal of the other clock bay. In both cases, PMOS ICW updates occur close the falling edges of the clock bays' outputs, while NMOS updates occur close to the rising edges. Specifically, the updates occur near the middle of the voltage steps on the impedance master's output.

Within the largest dashed box are master-slave flip-flops formed from "P" and "N" type latches (see section 4.5 for an explanation of this terminology). Two flip-flops are used for both the NMOS and PMOS impedance sampling. The first flip-flop is made of static latches that use feedback to quickly resolve metastability, while dynamic latches are used for the second flip-flop. More will be said about these latches in the next chapter.

PMOS sampling is performed on the rising edge of the (delayed) low-threshold comparator output, so it uses cascades of "P-N" flip-flops (or equivalently, "N-P" flops driven by the complement of the comparator output). NMOS sampling uses "N-P"
flip-flops for sampling, because it samples the impedance on the falling edge of the clock bay output.

**Figure 52. Impedance detector and impedance controller block diagram.**

The fixed delays shown in Figure 52 generate clock signals that drive the impedance detector and controller. The delays are sized to ensure that impedance sampling occurs near the middle of the transient voltage step. There are many issues associated with sizing the fixed delays, and they will be discussed in the next section. However, at this point it is important to note that accurate sampling of impedance levels requires a wide voltage step (i.e. a long signal trace) to give the mid-threshold comparator time to settle. It is for this
reason that automatic impedance sampling cannot be performed for clock bay outputs that drive short signal traces.

The $V_{\text{low}}$ and $V_{\text{high}}$ inputs from the comparators are differential, and interface directly to the delay elements making up the two fixed delays. The single-to-differential conversion is done locally close to the comparators, and then the resulting signals are buffered and distributed to the impedance matching circuit and the propagation delay estimator. The comparators do not directly drive these circuits without buffering as was indicated in the simplified architectures shown in Figures 47 and 50.

In total, the automatic impedance matching circuit has a latency of 2-1/2 clock cycles between impedance sampling and the subsequent modification of the ICWs by the impedance controllers. As a result, a finite state machine (FSM) is used to generate an update signal that forces the 6-bit counters to only update the ICWs once every three clock cycles. The FSM is a simple three state counter circuit based on two flip-flops. It resets to ‘00’.

Both the 6-bit counters and FSMs are reset by the (external) reset, signal from section 4.3.1. The reset signal is synchronized by master-slave flip-flops so that it does not violate setup and hold times when applied to the FSMs and counters.

The 6-bit counters are identical to that used in the DLL tap controller of Figure 45 except that they do not have differential outputs. As a result, only three latches per output bit are required instead of the five (or nine in the case of the LSB) used in the counter of Figure 45.

An enable signal is also present, and it represents the tri-state control for the tri-state buffers. It is only activated for the “impedance master” clock bay.
4.7.4 Fixed Delay Sizing

Choosing an appropriate size for the fixed delays of the impedance detector is an important consideration. The intent is to select a delay that is long enough to ensure that the mid-level comparator output has enough time to settle before being sampled by the clocks produced by the fixed delays. What limits making the delay arbitrarily long is the fact that a longer fixed delay requires a wider voltage step for sampling which in turn necessitates longer signal traces. As mentioned in section 3.6, the maximum signal trace is limited to approximately 45cm by the propagation delay estimator.

After many simulations, the fixed delay was adjusted to move the sampling point to approximately 3.5ns after the onset of a rising or falling edge. Reasons for this decision are explained below.

Figure 53 shows HSPICE [33] simulation results for a clock bay output for several matched transmission line configurations. According to the figure, the voltage step's plateau is reached within 1 to 2ns of the onset of a rising or falling edge on the output. The rise time to the voltage step is given by equation (3.34) and is determined primarily by the driver impedance ($R_d$), line impedance ($Z_0$), and the source capacitance ($C_s$). For source capacitances that are 10pF or less, the step has a very well defined shape when the transmission line length is 25cm or more. On the other hand, the 20pF voltage step is very distorted with only a 25cm long line.
Figure 53. Simulations of various “fast” and “slow” transmission line configurations:
(A) $C_S = C_L = 20\, \text{pF}; \quad R_S = Z_o = 95\, \Omega$; (B) $C_S = C_L = 10\, \text{pF}; \quad R_S = Z_o = 95\, \Omega$;
(C) $C_S = C_L = 10\, \text{pF}; \quad R_S = Z_o = 50\, \Omega$; (D) $C_S = C_L = 5\, \text{pF}; \quad R_S = Z_o = 50\, \Omega$.

In all cases, the transmission line length is 25cm, and the supply voltage is 5V. The transmission line model includes the effects of the bonding wire connecting the die to the chip package.

Post layout extraction of the output driver indicates that its capacitance (which makes up a significant fraction of $C_S$) is approximately 4pF. This places the clock buffer output close to curve “D” in Figure 53. Thus, sampling the stable central portion of the voltage plateau requires a sampling instant at least 1ns after a rising or falling edge of the clock bus output.

In addition, the sampling point must be delayed even further because of the delay of the comparator that identifies if the voltage step is above or below half the supply voltage. The delay varies depending upon how close the impedance is to being in a matched state. When the circuit is far from being matched, the voltage step is far from the mid-level threshold and the comparator responds quickly. However, when the circuit is close to being matched, it takes the comparator longer to settle to a definite high or low output.
value. The comparator reaches its stable point approximately 1.6ns prior to the above mentioned sampling instant, even when within 10% of impedance lock.

Note the sampling instant includes a 500ps latency (i.e setup time) in the sampling latches. Thus the sampling clocks themselves arrive 4ns after the onset of the output transition while the latches sample the data 3.5ns after the transition. The mid-threshold comparator is ready to be sampled 2ns (3.5ns-1.6ns) after the transition.

This 1.6ns buffer zone prior to the start of the latch setup interval represents a safety margin to allow for PVTL variation in the fixed delay generating the sampling clocks. In this nominal state, HSPICE simulation results indicate that impedance lock can be achieved with less than 10% error for transmission line impedances from 40Ω to 90Ω.

Automatic impedance matching requires signal traces of approximately 30cm to operate correctly in order to ensure a wide enough voltage step. A signal trace of 35cm is recommended in order to accommodate PVTL variations that may potentially increase the delay of the comparators and of the two fixed delays that generate the sampling clocks.

4.7.4.1 Future Considerations Regarding Impedance Sampling

Future work should consider making the sampling point of the voltage step controllable externally (using tapped delay lines) to allow for more flexibility in accommodating manufacturing variations.

In addition, the optimal sampling point can even be determined automatically by building a simplistic on-chip time domain reflectometer (TDR). During a start-up phase, the TDR samples the approximate voltage at many different instants - from close to the onset of the output transition until several nanoseconds after the transition - in order to build up a picture of how wide the voltage step is. The circuit can then analyze this data to determine
the best position for sampling the voltage step that takes into account manufacturing variations.

To reduce the hardware requirements for the TDR, sampling does not have to be performed in one clock cycle simultaneously across the entire voltage step. Instead, it can be performed at a different position each clock cycle for several clock cycles using a 2 to 4 bit analog-to-digital converter. Over several clock cycles, a picture of the entire voltage step can be constructed.

Alternatively, the voltage step waveform can be fed into a comparator driving a sampled delay line that samples the comparator output simultaneously at many time instants, but for only one comparator voltage level. Over several clock cycles, different comparator threshold voltages can be used develop a multi-voltage-level picture of the transient step.

4.7.5 Reset State Determination

The reset impedance level of the output driver is very critical to the successful initialization of the automatic impedance matching circuit. This is because the sampling clocks for the automatic impedance matching circuit are derived from comparators (see Figure 50) driven by the clock bay output. The clock bay output has transient voltage steps on the edge transitions that may affect the low and high comparators.

If the voltage step on the output is below either the low comparator's threshold (for rising edges) or above the high comparator's threshold (for falling edges) then the comparators may trigger too late and not sample the middle of the voltage step; impedance lock would then never be acquired!

In order to avoid this problem, the output driver is reset into a state of relatively low resistance (48Ω). This leads to voltage steps on the clock bay output that are as far above the 1/4 comparator threshold (for rising edges) and as far below the 3/4 comparator
threshold (for falling edges) as possible so that the low and high comparators are triggered cleanly at the start of each voltage step. Thus properly triggered sampling clocks are available upon initialization.

When the ILL is locked, then the voltage step is close to the 1/2 supply voltage threshold and the high and low comparators will continue to function properly. Generally, the high and low comparators trigger reliably if the output step is 300-500mV away from the comparator thresholds. With the above initialization scheme, the voltage step is always at least 1000mV to 1250mV away from the low and high comparator thresholds.

Figures 54(a) and 54(b) present results from HSPICE simulations of the extracted ILL layout (with parasitic capacitors included) that indicate the reset level of the voltage step for 40Ω and 90Ω transmission lines. Even in the 40Ω simulation (which represents the worst case), there is a reasonable noise margin left between the low and high comparator thresholds and the transient voltage step.

![Figure 54](image)

*Figure 54. ILL voltage step level after reset (to 48Ω) for signal traces with characteristic impedances of (a) 40Ω and (b) 90Ω. The load capacitance is 10pF and the signal trace length is 32cm. The arrows indicate the available noise margin just after reset. The HSPICE netlist includes a bonding wire model for the output connection to the chip package.*
(Note: The reset level can be set to an even lower resistance then 48Ω, but this was not done because it leads to a longer ILL locking phase which increases the already lengthy HSPICE simulation times needed to verify the correct operation of the CBC.)

One area of future study is to examine if it is better to derive the impedance sampling clocks from the input of the output driver instead of from the comparators connected to the driver’s output (in other words, avoid using the comparators for this purpose). This may be a better approach because the input to the driver does not contain any transient voltage steps. The problem, of course, is that the PVTL variations of the output driver may be large and shift the sampling clocks and the corresponding sampling points by large amounts. If this approach is undertaken, some external (or even automatic) means of adjusting the sampling points should be implemented as discussed in section 4.7.4.1.

Note that the low and high comparators are still needed at the clock bay output even with this approach, because their outputs are still used by the propagation delay estimator and the TDL2 delay line in the DLL.

4.7.6 Output Driver

The output driver for the clock bay is shown in Figure 55. It consists of a series of exponentially wider CMOS inverters arranged in parallel. Each inverter pull-up and pull-down path is enabled or disabled independently by the PMOS and NMOS impedance control words (ICWs) obtained from the ICW bus.
As shown in Figure 56, the output driver has an output resistance that varies from 30Ω to more than 140Ω depending on the value of the ICWs.

An important requirement for this output driver is that its output resistance must vary monotonically with the magnitude of the impedance control word despite PVTL variations. The circuit will be monotonic to 6-bits resolution as long as variation between fabricated transistors is smaller than a few percent (>1/32). If this is so, then the ILL will systematically approach and lock tightly near the desired output impedance level.
Figure 56. Output driver resistance versus impedance control word. The resistance represents the resistance of the output driver at its 2.5V switching threshold. Although the CBC is designed to accommodate trace impedances as low as 40Ω, the output driver can reach 30Ω so it can deal with PVTL variations in the driver that might require a lower ICW.

It should be noted that the ICW bus does not connect directly to the output driver transistors. Instead the ICW signals are first applied to a circuit that asynchronously clamps them either high or low to force the output driver to reset into its reset state of 48Ω (ICW = 4010 = 1010002). This is necessary at start-up because the impedance controller has no asynchronous reset, and clock signals must be allowed to flow through the output driver in order to perform a synchronous reset. The reset logic is constructed from NOR and NAND gates (as shown in Figure 48), and is controlled by the reset1 signal from section 4.3.1.

After the reset logic, the ICW signals are buffered using CMOS inverters so they can quickly drive the large output driver transistors. The PMOS ICW is also inverted at this point since the ICW assumes a high signal activates the various parallel pull-up paths, but, for PMOS transistors, a low signal is required.
4.8 Input/Output Circuitry

Two main CBC logic blocks remain to be discussed. They are a 2:4 input demultiplexor that is used to select the "impedance master" clock bay, and secondly, an output multiplexor that makes groups of internal signals visible to external circuitry for testing purposes (i.e. for improved observability).

4.8.1 Input Demultiplexor

Two external select lines determine which clock bay will be the impedance master via an input demultiplexor. This circuit consists of four 2-input AND gates driven by the select lines and their complements. Since there are only three clock bays in the prototype chip the 2-bit select signal should never be set to '11' as this addresses a non-existent (fourth) clock bay.

4.8.2 Output Multiplexor

At any given time, one-of-four groups of six internal signals are routed to output pads via the output multiplexor. Six 4:1 multiplexors are used for this purpose, and they are driven by a user defined 2-bit "group select" signal. The design of the individual multiplexors is shown in Figure 57. It is constructed to limit the number of series PMOS transistors to two or less.
Figure 57. One of the six 4:1 multiplexors used in the output multiplexor.

Table 3 summarizes the internal signals accessible via the output multiplexor.

<table>
<thead>
<tr>
<th>Group Number</th>
<th>Signal Name</th>
<th>Clock Bay of Origin</th>
<th>Signal Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (select=00₂)</td>
<td>icwn₁</td>
<td>Active Impedance Master</td>
<td>6-bit NMOS Impedance Control Word</td>
</tr>
<tr>
<td>2 (select=01₂)</td>
<td>icwp₁</td>
<td>Active Impedance Master</td>
<td>6-bit PMOS Impedance Control Word</td>
</tr>
<tr>
<td>3 (select=10₂)</td>
<td>tcw₁</td>
<td>Bay 0</td>
<td>6-bit TDL₁ Delay line Tap Control Word</td>
</tr>
<tr>
<td>4 (select=11₂)</td>
<td>vhout₀ (bit 0)</td>
<td>Bay 0</td>
<td>Output of the propagation delay estimator’s sampled delay line.</td>
</tr>
<tr>
<td></td>
<td>vlout₀ (bit 1)</td>
<td>Bay 0</td>
<td>Clock signal driving the propagation delay estimator’s sampled delay line.</td>
</tr>
<tr>
<td></td>
<td>yout₀ (bit 2)</td>
<td>Bay 0</td>
<td>Phase detector’s target input.</td>
</tr>
<tr>
<td></td>
<td>zout₀ (bit 3)</td>
<td>Bay 0</td>
<td>Phase detector’s reference input.</td>
</tr>
<tr>
<td></td>
<td>yout₁ (bit 4)</td>
<td>Bay 1</td>
<td>Phase detector’s target input.</td>
</tr>
<tr>
<td></td>
<td>zout₁ (bit 5)</td>
<td>Bay 1</td>
<td>Phase detector’s reference input.</td>
</tr>
</tbody>
</table>

The first three groups of six outputs consist of the TDL₁ tap control word for clock bay 0, the NMOS ICW, and the PMOS ICW. The fourth group of six outputs contains a mixture of outputs from two clock bays. Signals from groups 3 and 4 are buffered at their source.
using two CMOS inverters before being sent to the output demultiplexor, while the impedance control words from the first two groups require no additional buffering.

Bits 0 and 1 of group four are from the propagation delay estimator of clock bay 0. Bit 0 is the output from the sampled delay line, while bit 1 is the clock signal (derived from the low threshold comparator) that triggers the sampling latches. The time difference between the corresponding edges of these signals represents the round-trip propagation delay that the delay estimator attempts to measure.

Bits 2 to 5 of group four provide access to the phase detector inputs of clock bay 0 and 1. This allows one to study the DLLs' locked states in more detail, and to make comparisons between clock bays. The one way propagation delay estimate made by each clock bay can also be inferred by comparing the output signals with the clock bay outputs themselves.

There is substantial output delay associated with the output demultiplexor and the output pads which can therefore affect the validity of timing measurements made using group four signals. In general, timing measurements made between signals will likely be affected by the different routing delays as signals are routed to the I/O pads; however, an effort was made to keep most output signals close together to minimize this problem. Furthermore, relative changes between signals can still be studied over time (and across different external circuit configurations) quite accurately.
Chapter 5
VLSI Implementation

Chapter 5 discusses the VLSI implementation of the clock buffer chip (CBC) built for this thesis. The detailed VLSI design of critical circuit elements is presented in section 5.1, followed by a discussion of layout and floorplanning in section 5.2. Buffering and interconnection issues are discussed in section 5.3, while power requirements are examined in section 5.4. The chapter concludes with a summary of the simulation and testing performed on the completed VLSI layout in section 5.5.

5.1 Basic Circuit Elements

5.1.1 Delay Elements

The delay element design selected for the CBC is presented in Figure 58. It is adapted from a circuit used in [24]. Adjusting bias voltage $V_{bn}$, and especially voltage $V_{bp}$ varies the current in the circuit, and therefore its switching speed. Both bias voltages are controlled off-chip to allow for external control over switching speed and power.
consumption. Although this differential inverter consumes more (DC) power than static CMOS inverters, switching noise due to transient current spikes is reduced [24].

![Figure 58. Differential delay element. Output voltage swing is between 5V and 1V. The widths of the PMOS/NMOS transistors forming the inverters are indicated under each inverter symbol. All transistor gate lengths are 0.8μm.](image)

The two inverters connected to the delay element's differential output have a PMOS-to-NMOS width ratio that is greater than 2 in order to shift their switching threshold up closer to the 3V mid-point of the differential signal swing. This ratio could have been increased even further by increasing the PMOS transistor width; however, the increased load on the output node did not make this practical.

Simulated delay element performance at various bias voltages is summarized in Table 4. These HSPICE [33] simulation results are based upon simulations performed on the extracted layout (with parasitic node capacitances included).

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Delay (ps)</th>
<th>Standby Current (mA)</th>
<th>Peak Current (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{bn}=5V, V_{bp}=0V$</td>
<td>145</td>
<td>1.6</td>
<td>2.8</td>
</tr>
<tr>
<td>$V_{bn}=4.5V, V_{bp}=0.5V$</td>
<td>160</td>
<td>1.3</td>
<td>2.7</td>
</tr>
<tr>
<td>$V_{bn}=4V, V_{bp}=1V$</td>
<td>180</td>
<td>1.0</td>
<td>2.7</td>
</tr>
<tr>
<td>$V_{bn}=3V, V_{bp}=2V$</td>
<td>250</td>
<td>0.6</td>
<td>2.5</td>
</tr>
</tbody>
</table>
Figure 59 indicates the layout of a single delay element cell. It measures 23μm by 43μm. Metal 2 is routed left-to-right across the cell, and carries power, bias voltages and cell-to-cell communication. Tap ports are located along the top of the layout.

![Figure 59. Delay element layout.](image)

Since the delay element is used hundreds of times in the CBC, care was taken to lay out the cell as small as possible, while still keeping the wire and gate lengths in each differential half of the element as equal as possible to achieve balanced delay.

In order to ensure that all delay elements are fabricated with similar characteristics across the entire chip, two approaches are used. The first is to maintain the same boundary conditions around each delay element. This is necessary because the local structures around a delay element influence the degree of etching that occurs, and the final dimensions of circuit components.

Cascading delay elements in long chains allows the left and right boundary conditions to be matched for nearly all delay elements. Also, mirroring delay lines about their bottom (vss) edge matches the boundary conditions along the lower edge, while the tap port inverters located at the top of the delay elements match conditions for the upper edge.
The second technique for ensuring similar delay characteristics is to orient all delay elements in the same general direction on the die (i.e. all delay lines may run vertically, but never horizontally as well). In this way, the effects of anisotropic (directionally varying) etching affect each delay element in a similar manner.

5.1.2 Sampling and Storage Elements
Two types of storage elements are used in the CBC. The first type is a static latch that is used for sampling signals that may be changing even during the sampling phase (as in the DLL phase detector, ILL impedance sampler, and propagation delay estimator). This latch uses the feedback provided by cross-coupled inverters to quickly resolve any metastability that may occur when sampling unpredictable inputs. The second type of storage element is a compact dynamic latch that is used with internal logic like counters and state machines for which latch setup and hold times can always be satisfied.

The static D-latch design is shown in Figure 60. Data is sampled and held when CLK=1, and CLKbar=0. When CLK=0 and CLKbar=1, the feedback path is disabled and the latch becomes transparent. Typical setup times are on the order of 600 to 500ps. Since metastability resolving time depends exponentially on MOSFET size [26], care was taken to ensure a good latch design. According to [25] the design of Figure 60 has better metastability performance than a D-latch constructed using pass transistor switches to enable or disable the feedback path (In Figure 60 tri-state inverters are used for this purpose). Also, to achieve the best metastability resolving times, the output node capacitance of the cross-connected inverters must be minimized, and the PMOS to NMOS width ratio must be typically set to less than or equal to 1 [25][26]. The latch of Figure 60 was used in the propagation delay estimator, while a scaled down version was used in the propagation delay estimator's sampled delay line in order to save area.
Dynamic latches are used primarily to save area. The design is taken from the true single phase clock (TSPC) logic technique discussed in [27]. Two types of latches, called N-latches, and P-latches, are used in an alternating fashion in order to avoid race conditions as shown in Figure 61. These latches have small setup times, and are compact.

When the clock is high, the N-latch reduces to two series inverters, and it becomes transparent, while at the same time, the P-latch is in its latched state. The P-latch holds its value in the parasitic capacitance of its output node since, with the clock in its high state,
the output is electrically disconnected from vdd and vss. When the clock is low the roles reverse; the P-latch becomes transparent, and the N-latch holds its state.

5.1.3 Comparators

Analog comparators are used within a clock bay for two purposes. The first and most important use is to provide signals that indicate when the clock bay output crosses the low, middle, and high voltage thresholds (set to 1/4, 1/2, and 3/4 the supply voltage). These signals are used by the propagation delay estimator and impedance matching circuits as discussed in sections section 4.6 and 4.7. The second use of comparators is to convert the differential outputs of the TDL1 and TDL2 delay lines to single-ended outputs.

Two types of comparator circuits are used in the clock bay. The first is a standard CMOS differential amplifier shown in Figure 62. This amplifier uses an active current-mirror load structure formed from PMOS transistors. At low frequencies, the small signal gain of the amplifier, $A_v$, is given by equation 5.1 ([28], section 3.8), where $g_{m1}$ is the transconductance of transistor $Q_1$ and $r_{ds1}$ is the small signal drain-source resistance of transistor $Q_1$. The bias current and the size of the drive transistors determines the value of $g_m$.

$$A_v = g_{m1}(r_{ds2} \parallel r_{ds4})$$

(5.1)
NMOS transistor Q₅ acts as a constant current source that provides a bias current for the amplifier. The bias voltage used is approximately 1.3V, but it is controlled off-chip for greater flexibility. Transistor sizing, is discussed in detail in [11], pp. 84-86, but in general, for common PMOS to NMOS mobility ratios, the sizes of the PMOS loads are set approximately equal to that of the NMOS current source so that the loads can deliver the necessary total current to the source.

The layout of the amplifier is shown in Figure 63. To ensure that the two halves of the differential circuit are well matched, a common-centroid layout technique is used ([28], section 2.4). This involves forming larger transistors from smaller, carefully interdigitated transistors in an effort to minimize gradient effects across a circuit.
Figure 63. Layout of CMOS differential amplifier.

The above amplifier works very well for determining when the clock bay output crosses the 1/4 and 3/4 voltage supply thresholds. Unfortunately, it is not responsive enough when applied to the more challenging task of determining when the voltage step on the clock bay output crosses the mid-supply threshold. The output takes a long time to settle when the input is close to the mid-supply threshold, so sampling the impedance level is more difficult. This limitation affects how well the impedance matching circuit can approach the ideal impedance lock point.

To overcome this difficulty, a larger but higher performance amplifier is used. This amplifier is shown in Figure 64, and is described in [29]. It is a complementary self-biased differential CMOS amplifier (CSDA). Its main benefits include a doubling of the differential gain compared the conventional CMOS amplifier discussed above, and a self-biased design that is tolerant of process, temperature and supply variations. The addition of capacitor $Q_7$ is reported to enhance the switching speed in some circumstances [29].
The circuit is formed starting from two standard CMOS amplifiers: one with an NMOS current source as in Figure 62, and a complementary one that uses a PMOS current source. The loads of these circuits are deleted and the two resulting circuits are connected together to form a complementary, but externally biased amplifier. By connecting the gates of Q6 and Q1 to the drains of Q4 and Q2, the circuit becomes self-biased and stable [29]. Since four amplifying devices (Q2 to Q5) are available in the resulting circuit, the differential gain is doubled compared to the amplifier of Figure 62.

Figure 64. Complementary self-biased differential CMOS amplifier (CDSA) schematic.

These comparator circuits are fairly simplistic in that they are formed using open-loop amplifiers. Higher-speed comparator designs are available, but they typically require an external clock signal for control purposes ([28], ch. 7). These high-speed circuits are definitely worth investigating in future work, especially as a replacement for the mid-level threshold comparator.

5.1.3.1 Voltage Threshold Generation

The 1/4, 1/2, and 3/4 voltage thresholds are generated on-chip in order to track variations in supply voltage. The threshold generation circuit is a voltage divider formed from a tapped cascade of four 1.5kΩ resistors. Capacitors (approximately 2.3pF each) on the
voltage divider's outputs filter noise injected by the comparators when they switch. This noise is injected through approximately 0.04pF of parasitic capacitance on the gates of the comparators' input transistors. The noise is less than +/-45mV, and causes negligible shift in the output signals from the comparators.

The voltage divider consumes approximately 0.81mA of DC current. The resistance, and therefore the current, was adjusted so that the threshold voltages could track changes in supply voltage that persist for longer than 10 to 100ns.

Resistive poly is used to implement the four resistors, while NMOS capacitors provide the necessary filter capacitance. The circuit occupies an area of 115μm x 75μm.

5.1.4 Output Driver

As discussed in 4.7.6, the output driver consists of an array of exponentially sized inverters that are connected in parallel and enabled independently by the individual bits of the impedance control word.

The widths of the driver transistors were determined by plotting $I_{DS}$ versus $V_{DS}$ curves for different sized transistors, and determining the resistance at the $V_{DS}=2.5V$ level (i.e the switching threshold of the inverter). This involves computing $V_{DS}/I_{DS}$ at the $V_{DS}=2.5V$ point.

HSPICE $I_{DS}$ versus $V_{DS}$ plots were generated for a large range of transistor widths. From these plots, equations 5.2 and 5.3 were computed that summarize the resistance of a transistor at the $V_{DS}=2.5V$ level (for a gate width of $W$ and a gate length of 0.8μm).

\[
R_{PMOS@V_{DS}=2.5V} = \frac{17600}{W} \tag{5.2}
\]

\[
R_{NMOS@V_{DS}=2.5V} = \frac{6800}{W} \tag{5.3}
\]
The largest output driver pull-up/pull-down path was sized to provide a resistance of 60Ω. The other five pull-up/pull-down paths were given widths that were progressively smaller by factors of 2 as indicated in the schematic of 4.7.6. When all pull-up/pull-down paths are active, the effective resistance is approximately 30Ω.

The layout for the output driver is presented in Figure 65.

![Output driver layout](image)

Figure 65. Output driver layout.

5.1.4.1 Output Driver Latchup Considerations

CMOS circuitry is susceptible to a phenomenon known as latchup. Latchup is caused by a parasitic npn-pnp circuit that is associated with all CMOS logic. Under the right conditions this parasitic circuit can be triggered to induce a damaging short circuit between the VSS and VDD supply lines ([1], pp. 156-163).

Latchup is triggered by transient voltage spikes that exceed normal operating voltages. It is especially likely to occur in I/O circuitry where there are large currents and large parasitics present. Special layout precautions are needed to prevent latchup.
The first precaution is to physically separate the NMOS and PMOS driver transistors as much as possible. Secondly, guard rings should be placed around both the PMOS and NMOS transistors. These rings reroute minority carriers that would otherwise help trigger the parasitic npn-pnp circuit.

PMOS transistors in the output driver are doubly guard-ringed with a continuous n+ ring connected to VDD (inside the N-well) and a p+ ring connected to VSS (just outside the N-well). NMOS transistors are surrounded by a single n+ ring connected to VSS.

5.1.5 Tapped-Delay Line Selection Circuits

The output structures of the tapped delay lines benefit from special attention during the layout phase. These output structures are pass transistor based, and the RC delay through them depends significantly on the layout arrangement of the transistors.

Recall that in the case of the TDL2 tapped-delay line, each differential output connects directly to a common output node. Thus, a relatively large capacitance must be charged and discharged through the active output pass transistor. To minimize this capacitance, each pass transistor is formed using two smaller interdigitated transistors that share their output diffusions. In addition, the gate is placed as close as possible to the contacts in this diffusion to further minimize the output capacitance. These efforts reduce the diffusion capacitance on the output node by 50% compared to using a non-interdigitated pass transistor. Propagation delay through the output structure is approximately 1.5 to 2ns.

TDL1 uses a hierarchical tapped-delay line output structure for its differential outputs. This is a faster architecture, although more difficult to lay out. To minimize diffusion capacitance, output pass transistors can be grouped to share diffusions (groups of 3 and 2 transistors work well). Capacitance reduction is lower at 20%, but it is still significant. The propagation delay through this (larger) structure is also 1.5 to 2ns.
5.2 Layout and Floorplanning

5.2.1 Layout Strategy

The clock buffer chip was laid out in a 3-level metal 0.8mm BiCMOS (N-well) process from Nortel [30]. The design was constructed in a hierarchical manner beginning with small basic cells (like delay elements and latches), and proceeding to larger, more complicated circuits (counters, delay lines, clock bays). Full-custom layout was used throughout the CBC except for standard I/O cells obtained from a commercial pads library.

A common wiring strategy was adopted in all circuit layouts to simplify the task of powering and interconnecting blocks at the intermediate and global layout levels. The key aspect of this strategy that allows everything to proceed efficiently is that the routing direction of each layer of metal is perpendicular to the layers of metal that it directly connects to.

Internal basic cell connections are preformed using polysilicon, Metal 1, and, where necessary, Metal 2. Metal 1 and poly are routed as needed within a cell; however Metal 2 is consistently routed in the direction of signal flow along a bit-slice. Cells are constructed with Metal 2 power supply rails at the edges of the cell and flowing in the direction of signal flow.

All bit-slices on the chip are arranged in the same direction so that Metal 2 power supply rails flow in a common direction. This permits easy connection to the global power supply nets formed from Metal 3. Metal 3 was used only for global power and bias voltage distribution and is routed perpendicular to Metal 2. Outside of basic cells, Metal 1 flows perpendicular to Metal 2 and is used for cell-to-cell communication across power supply rails.
Wherever possible, parallel flows or “bit-slices” of cells are mirrored along their common edges so that adjacent slices share power supply lines and N-wells. This leads to more compact layouts and better power distribution.

5.2.2 Clock Bay Floorplan

Figure 66 depicts the floorplan of an individual clock bay. The propagation delay estimation circuit and the TDL2 delay line are located on the left, while the DLL and ILL controllers occupy central locations. Communication between clock bays is carried in a vertical direction by the NMOS and PMOS impedance control word (ICW) bus. The TDL1 delay line and the output driver are positioned on the right. All vertical signals are in Metal 2, while horizontally routed signals use Metal 1.

**Figure 66.** Clock bay floorplan. The TCW and ICW signals represent the tap and impedance control words, while \( V_{\text{low}} \) and \( V_{\text{high}} \) are buffered (differential) outputs from the low and high threshold comparators.
By far, the largest clock bay block is the propagation delay estimator. Figure 67 indicates the floorplan of this unit. It is constructed from three delay lines oriented in a common direction. The lower two thirds of the layout are dedicated to propagation delay measurement, and, in addition to delay lines, contain sampling logic and a leading one detector. By making the leading one detector almost entirely from Metal 1 and poly, the TDL2 tap control word can be routed upwards over logic (in Metal 2) to the TDL2 delay line.
As shown in Figure 68, the phase detector and TDL1 tap controller consists of a relatively random arrangement of blocks dominated by a 6-bit up/down counter. A large fraction of the area is consumed by the latches and buffers associated with the differential outputs from the counter. The reference clock drives all logic in this section of the clock bay.
A mixed collection of circuits is found within impedance control unit. As shown in Figure 69, the low, mid and high threshold comparators are located at the bottom of this block together with the voltage reference circuitry. The low and high comparator outputs are converted to differential signals, buffered, delayed, and finally used to clock the mid-level comparator-sampling flip-flops (also at the bottom of the floorplan). Purely digital logic circuits are located at the top of the impedance control unit and consists of finite-state machines (FSMs), counters and reset logic.

The counter outputs drive tri-state buffers on the right side of Figure 69. These buffers are partially free of Metal 2 so that a portion of the ICW bus can be routed over them to save area. It was important to place the tri-state drivers parallel and close to the ICW bus so that the bus wire length is kept to a minimum. This leads to the smallest possible bus capacitance.
Figure 69. Floorplan of the impedance control and comparator circuits.

Due to the relatively homogenous nature of the TDLI delay line and the output driver, no additional floorplan figure is necessary beyond the general view of Figure 66.

Delay line TDLI uses a mirrored delay element orientation similar to that in the propagation delay measurement unit. TDLI’s hierarchical output structures are rather irregular and difficult to layout; they consists of two interleaved 64 input pass transistor trees. The lower two tiers of the trees are positioned directly against the delay element tap ports. Bits from the DLL tap control word (routed in Metal 2 on both sides of the long axis of the delay line) drive these pass transistors. The third level in the pass transistor hierarchy is located on the next layer out from the delay elements, followed by the internal inverters on the following layer, and so on. Upper levels of the pass transistor trees are also located progressively closer to the bottom of the delay line which is where the final
outputs must be sent. These differential outputs are merged by a comparator and then sent to the output driver.

5.2.3 Global Floorplan

The three clock bays of the prototype CBC are cascaded in the direction of Metal 2 to form a larger block with an aspect ratio of approximately 3:1 (see Figure 70). The output (debug) multiplexor, master select demultiplexor, and the fixed delay generating the DLLs' reference clock are all placed between two of the clock bays; they occupy approximately one third the area of a clock bay. Together, these structures form the core logic of the CBC.

I/O and power pads are arranged on two sides of core logic - parallel to the long dimension of the core. In addition, a power supply bus is located between each set of pads and the core logic to connect the Metal 1 wires coming from the power pads to the Metal 3 power net traversing the core logic.

The final chip size is 2605μm x 3180μm, and it contains 34,000 MOS transistors. Each individual clock bay measures 1mm x 1mm, and contains approximately 10,550 MOSFETs. These counts include the individual transistors that make up wider interdigitated transistors. For comparison the CBC schematic, which does not contain interdigitated transistors, has 21,150 MOSFETs, while the clock bay schematic contains 6,600 transistors.

Pin assignments for the clock bay are indicated in Table 5, while a view of the CBC layout is shown in Figure 71.
Figure 70. CBC floorplan showing all three clock bays. The clock input is located at the bottom while the primary outputs are along the top. (a) is the fixed delay line that generates the reference clock, (b) is the output multiplexer used for debugging, and (c) is the master-select decoder. Clock, reset, enable, and debug signals are routed along the signal bus at the lower edge of the clock bays. Positioned along this bus are distributed buffers (not shown) that help drive critical signals into the clock bays with minimum delay. Equal length wires are used to route the input and reference clocks to each clock bay.

<table>
<thead>
<tr>
<th>Number(s)</th>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2,6,10,14, 19,25,30, 32,37</td>
<td>vss</td>
<td>supply</td>
<td>Ground.</td>
</tr>
<tr>
<td>3,8,11,15 20,24,29, 31,36</td>
<td>vdd</td>
<td>supply</td>
<td>Power.</td>
</tr>
<tr>
<td>7,9</td>
<td>Vbp,Vbn</td>
<td>bias</td>
<td>PMOS, NMOS delay element bias voltages. (see section 5.1.1)</td>
</tr>
<tr>
<td>4</td>
<td>Vbias</td>
<td>bias</td>
<td>Bias voltage for the low and high voltage threshold comparators.</td>
</tr>
<tr>
<td>13</td>
<td>in</td>
<td>input</td>
<td>Clock input.</td>
</tr>
<tr>
<td>Number(s)</td>
<td>Name</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------</td>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>33, 26, 38</td>
<td>out0, out1, out2</td>
<td>output</td>
<td>Clock output from clock bay 0, 1 and 2.</td>
</tr>
<tr>
<td>18, 5</td>
<td>reset1, reset2</td>
<td>input</td>
<td>Reset signals for the CBC.</td>
</tr>
<tr>
<td>22, 23</td>
<td>ma0, ma1</td>
<td>input</td>
<td>Master select signal. (see section 4.8.1) (ma0=LSB, ma1=MSB)</td>
</tr>
<tr>
<td>1</td>
<td>oen</td>
<td>input</td>
<td>Output pad enable signal. Enables the debug output pads when high (signals o0 to o5). Disable to save power.</td>
</tr>
<tr>
<td>34, 35</td>
<td>os1, os2</td>
<td>input</td>
<td>Output-multiplexor group select signal. (see section 4.8.2) (os1=LSB, os2=MSB)</td>
</tr>
<tr>
<td>12, 16, 17, 21, 27, 28</td>
<td>o0, o1, o2, o3, o4, o5</td>
<td>output</td>
<td>Output-multiplexor (debug) outputs. For debugging and observing the internal state of the CBC. (see section 4.8.2) (o0=LSB, o5=MSB)</td>
</tr>
</tbody>
</table>

*Figure 71. CBC layout.*
5.3 Buffering and Interconnection Issues

5.3.1 Buffer Requirements

When a weak signal attempts to drive a large load, some form of buffering is necessary to ensure that the load is driven with minimum delay. Cascades of increasingly larger inverters are used to perform this buffering. The ratio by which each inverter stage is increased relative to the previous stage is known as the stage ratio, and the optimum value for this ratio is close to 3 ([1], pp. 229, 230). In many cases, larger ratios of up to 10 can be tolerated without too much loss of performance especially when area or power limitations preclude using a smaller ratio.

In the CBC, there are two general classes of signals requiring buffering. The most demanding are the clocking signals that include the input clock and reference clock. These signals require sharp edge transition times, and for this a stage ratio of 3-4X gives good results (i.e. 250-350ps rise/fall times).

Driving requirements for the second class of signals is more flexible since they are not used as clocking signals. These signals include, for example, the impedance control words and the tap control words. In this case, a driving ratio of up to 10X is adequate, and leads to edge transition times on the order of 1.5 - 2ns. The reduced buffering requirements for this class of signals is helpful since the area and power consumed by using a smaller stage ratio (of 3-4X) would be prohibitive considering the large loads these signals drive.

When buffering clock signals that are transmitted over long distances, distributed buffers are necessary. Instead of grouping buffers at the source, buffers are distributed along the path from the source to the destination. As a result, the largest buffers get placed very close to the logic being driven, which means the largest currents travel across low resistance paths from the driver to the target logic. The return currents to the driver also have a low resistance path, so resistive (IR) drop in the supply lines is lower. Using
distributed buffers also reduces wire delay because this delay grows exponentially with length and multiple buffers keep this wire length short.

Distributed buffers are used for the input clock, reference clock, and reset signals. The clock buffer chain uses a stage ratio of 3-5X, while the reset signal buffers use a ratio of 6-10X.

5.3.2 Wire Sizing Issues

For routing signals over short distances, wire load is not a major issue and can be ignored. In this case, HSPICE simulations of circuit schematics (which generally ignore wire load) are adequate for determining the peak currents expected in a wire. Peak current estimates can be used to size wires so that they meet the current density limits for the wire material. (If these limits are exceeded for long periods of time, wire failure will result due to electromigration effects).

For long wires, wire width does have a large impact on the load that a buffer must drive so wire loading cannot be ignored. Increasing a wire's width increases the load that must be driven by a buffer. This means that the current needed to maintain the same rise/fall time must also increase. As a result of the increased current, a new wire width must be calculated which leads to yet another revision of the load and current levels. This iterative procedure is repeated until a wire width is found that can handle the necessary current. A small 'C' program was written to automate these calculations and come up with a workable wire size. It tries all wire sizes from the minimum size upwards, looking for the first width that can handle the required current. Current, 'I', is estimated using the well know equation I=\(C\frac{dV}{dt}\) where 'C' represents load capacitance, and 'dV/dt' the signal rise/fall time. Overall, these calculations are quite conservative as wire widths in the CBC are based on peak rather than average currents. Once the wire load is determined, a
properly sized buffer can be constructed to drive the wire and load (with the desired rise/fall time).

Wire widths come into play when sizing power supply lines as well. Wider supply lines offer a lower resistance that experiences less resistive drop. There is no penalty for using wider power supply lines other than an increase in circuit area. Metal 3 offers the lowest resistance out of all three metal layers, so it is used for global power distribution.

5.3.2.1 Wire Propagation Delays

Signal propagation delay grows exponentially with wire length, because when length is doubled, both wire resistance and capacitance doubles. The 'C' program described above outputs an approximate (lumped) RC delay for a given wire length, width and load. For signals internal to a clock bay sent over metal wires, propagation delays are relatively insignificant; however, for global signals (lengths >1mm) wire delays can be significant (>100ps). It is for this reason that equal length wires are used to route the input and reference clocks to each clock bay.

On a smaller scale, wire propagation delays (and current density limits too) affect the length of polysilicon that can be used to form transistors. This impacts how wide the largest transistors can be. For the Nortel process, polysilicon gate widths are kept smaller than 25μm to avoid problems. Larger transistors are formed from interdigitated smaller transistors having their gates connected together with metal.

5.4 Power Requirements

HSPICE simulations of the extracted layout were performed to determine the power requirements for the CBC. This procedure is necessary so that reasonable data is available to estimate parameters such as the number of power pins required, the expected operating temperature, and the worst case resistive (IR) drop/bounce.
Due to the prohibitive amount of computer time needed by large HSPICE simulations, the entire clock buffer chip was not simulated as a unit. Instead, results from individual clock bay simulations were combined with simulations of global interconnect, distributed buffers, and miscellaneous logic located outside the clock bays. The current requirements are summarized in Table 6.

<table>
<thead>
<tr>
<th>Component</th>
<th>Average Current (mA)</th>
<th>Peak Current (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Individual Clock Bay and its Clock Bay Output Pad.</td>
<td>330</td>
<td>450</td>
</tr>
<tr>
<td>CBC Core Logic and Clock Bay Output Pad.</td>
<td>1080</td>
<td>1560</td>
</tr>
<tr>
<td>Clock Chip Input Pad and Debug Output Pads</td>
<td>80</td>
<td>240</td>
</tr>
<tr>
<td>CBC Core and Pads</td>
<td>1160</td>
<td>1800</td>
</tr>
</tbody>
</table>

According to Table 6, the CBC uses an average power of approximately 5.8W with a 5V supply. The reason for this relatively high power consumption is due to the DC power dissipation of the delay elements (see section 5.1.1). This power requirement can be reduced by up to 50% by adjusting the delay element bias voltages and also by disabling the debug output pads (via the output pad enable pin). At the low-power bias voltage setting, the delay elements run approximately 50ps slower than a typical CMOS inverter.

A total of nine pairs of \( vdd \) and \( vss \) pads are used to power the chip. These pads can provide a total of 1935mA of current into the chip (i.e. 210mA per pair). This estimate is based on the width of Metal 1 coming from each power pad, and the Metal 1 current density limit of 3.6mA/\( \mu \)m.
5.4.1 Operating Temperature

It is difficult to estimate the operating temperature of the CBC since this depends on the type of packaging and the thermal management techniques used. Transistor operating temperature (otherwise known as junction temperature, $T_J$) is computed by equation 5.4, where $P_{\text{average}}$ is the average power dissipation, $T_{\text{ambient}}$ is the external temperature, and $\theta_{JA}$ is the thermal resistance from the junction to the external world (measured in units of °C/W).

$$T_J = P_{\text{average}} \theta_{JA} + T_{\text{ambient}}$$ \hspace{1cm} (5.4)

Typical values for thermal resistance range from 40°C/W for plastic packages in still air, to as low as 3°C/W for ceramic packages using an air cooled heat sink (e.g. Intel Pentium processor) [31]. Assuming a value of 15°C/W for the thermal resistance, the CBC will operate at approximately 105°C in a room temperature environment.

5.4.2 IR Drop/Bounce

Since the chip is powered from two sides, the worst case IR drop/bounce is relatively limited. A worst case calculation assumes a peak supply current of 200mA forced across 1.2 squares of Metal 3 (at 0.0265Ω/square), 2.6 squares of Metal 2 (at 0.1Ω/square), and 8 squares of Metal 1 (at 0.1Ω/square). In this case, the total IR drop is $1.1 \Omega \times 0.2A$ or 0.21V.

The majority of the IR drop is due to the Metal 1 from the power pads to the power bus that runs parallel to the long dimension of the core logic.

The above calculation assumes no decoupling capacitance is present on the chip so that peak current requirements must be supplied entirely from the power pads. However, this is not entirely true since there is already significant capacitance on the supply lines. This capacitance is in the form of MOSFET transistors connected to power and ground. A capacitor is formed at the reverse biased junction between the source and the substrate.
(NMOS) and between the source and the N-well (PMOS) ([28], section 1.1). This capacitance will supply some of the peak current.

5.4.3 Power Considerations for Future Work

The design effort constructing the CBC focused on a type of delay element that, although capable of small delays, has significant DC power dissipation. However, power requirements can be drastically reduced by using a delay element with little or no DC power dissipation.

One approach to constructing such a delay element is to use a differential delay element that has two cross-coupled PMOS loads instead of PMOS loads that are connected to a bias voltage. In this case, the gate of one PMOS load is connected to the drain of its partner, and vice versa. These loads switch slower, but have very little DC power dissipation ([32], p. 98).

Considering the high speed CMOS technologies that are becoming available with feature sizes of 0.35μm and smaller, this approach might offer a reasonable power saving alternative to the delay element design of section 5.1.1.

5.5 Simulation and Testing

Several simulation tools were used to design and test the CBC. The digital hardware description language, Verilog [34], was used during the early design stages to model and test the DLL architecture. The HSPICE [33] circuit simulator was used extensively to perform detailed testing of both the CBC schematic and the extracted layout. In addition, various layout tools from the Cadence development environment were used to check the final layout for design rule violations, shorts, and for functional equivalence with the CBC schematic.
5.5.1 Verilog Modeling

Delay information from HSPICE simulations of basic cells (like delay elements and latches) was used to construct a detailed Verilog model of the DLL circuitry. The model permitted rapid examination of the timing in the DLL. Verilog simulations are orders of magnitude faster than HSPICE simulations, so many different delay line sizes and configurations can be studied in a short period of time. These simulations identified two reset problems that would have been time consuming to track down using HSPICE.

Only the DLL portion of the clock bays was modeled in Verilog, because the ILL portion is very dependent on analog inputs.

5.5.2 HSPICE Testing

All basic cells were tested using HSPICE to determine correctness, and operating speeds. Larger blocks, such as the propagation delay estimator, the phase detector, and impedance matching circuitry, were tested independently. The largest complete block that was tested was an entire clock bay.

Testing was performed on both the schematic views as well as extracted layouts. Tests on layouts of sub-blocks within a clock bay included parasitic capacitances, but simulations of the clock bay itself were performed without parasitics.

Both the DLL and ILL circuits were tested until the DLL and ILL loops were locked, with many different output configurations. This required simulations of up to 2.5μs duration for circuits containing up to 11,000 components. For some simulations, the supply voltage was bumped from 5V to 4V after the locked state was acquired. This tested the ability of the loops to reacquire lock after experiencing a disturbance that pushed them away from their locked state.
Additional testing was performed to check the operation of the whole clock chip. A special schematic and layout was constructed for the entire CBC that contained all the global routing, buffers, and top-level logic, but only a small subset of the clock bay circuitry (i.e. only the clock bay interface logic). This permitted the entire chip to be simulated (including parasitic capacitances) using HSPICE to check the correctness and timing of the top-level signals.

Layout versus schematic (LVS) checks were performed on all blocks in order to confirm that the layouts matched their respective schematics. In addition design rule checking (DRC) was performed on all cells.

LVS was not performed on the final layout which included pads from the commercial pads library, because no schematics were available for these pads. In this case, a very detailed visual examination of the layout was undertaken to check each pad connection. This included checking a large scale check plot of the chip.

Finally, the entire layout was extracted and critical signals and power nets were probed to confirm there were no shorts, and that all top-level signals were connected properly.

5.5.3 Performance

HSPICE simulations predict that the CBC will be able to operate effectively at frequencies of up to 40 or 50 MHz. Clock bay jitter is expected to be on the order of +/-2 delay element delays. More will be said about this in the next chapter which describes the experimental results obtained from CBC prototypes.
Chapter 6
Experimental Results

This chapter presents a summary of the performance results obtained from CBC prototypes fabricated through the Canadian Microelectronics Corporation (CMC). Sections 6.1, and 6.2 discuss the test circuit and procedures used to test the CBC, while section 6.3 details the performance results obtained. The testing procedure measured the following parameters:

- Maximum operating speed
- Clock deskewing performance
- Impedance matching/ILL jitter
- DLL jitter
- Power consumption

6.1 Test Circuit Design

A standard IC tester could not be used for testing the CBC since it provides no control over the length and impedance of the signal traces connected to the CBC clock outputs. Instead, a test board (shown in Figure 72) was manufactured containing different length
clock traces with impedances of approximately 70Ω. The board contains a nearly continuous ground plane to help ensure a uniform impedance for the clock traces.

Two independent test circuits were built on the same test board, each with different length clock traces. The longest traces were assigned to different outputs between the two test circuits so that different clock bay impedance masters could be tested. Table 7 lists the clock trace lengths connected to each test circuit.

**TABLE 7. Test Board Clock Trace Lengths**

<table>
<thead>
<tr>
<th></th>
<th>Test Circuit 1 Trace Length (cm)</th>
<th>Test Circuit 2 Trace Length (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clock Bay 0</td>
<td>35</td>
<td>25</td>
</tr>
<tr>
<td>Clock Bay 1</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>Clock Bay 2</td>
<td>22</td>
<td>10</td>
</tr>
</tbody>
</table>

*Figure 72. CBC test board layout. Clock trace widths appear as thin lines in the figure, but they are actually 0.06" wide. Compare this with the 0.1" spacing of the IC pins.*
Each clock trace is terminated in a 14-pin DIP socket that accommodates a CMOS 4069 inverter chip. The load capacitance of this device is approximately 6pF and is typical of many ICs.

As discussed in Chapter 4, two reset signals, reset1 and reset2, initiate the automatic impedance matching and the DLL lock. Having two independent resets is very valuable for testing purposes as the ILLs and DLLs can be examined separately. These reset signals are generated by debounced switches controlled by the user.

The clock input to the CBC is obtained from a fixed-frequency clock oscillator IC. These ICs come in a range of frequencies that cover the operating range of the CBC.

### 6.2 Test Procedure

Measuring the deskewing performance of the CBC requires a high bandwidth oscilloscope, and low capacitance probes. The high bandwidth permits precise timing measurements while the low capacitance probes ensure that the measurement process itself does not cause unacceptable distortion of the reflections on the clock traces. The scope used for testing was a four channel 1GHz (TDS784A) Tektronics scope with 1.5GHz active probes. These probes load the circuit being tested with less than 2pF of capacitance.

Measurements were made at the sources and destinations of each of the three clock bay outputs. Since the oscilloscope had four channels available simultaneously, a complete picture of all three CBC outputs (at both source and load ends) could be assembled by merging results from only two measurement captures. Observations were gathered (1) during ILL and DLL reset, (2) after ILL lock, and (3) after DLL lock.
In addition, the locking behaviour of the NMOS and PMOS ILLs and clock bay 0's DLL were monitored digitally via the debug port. Debug signals were observed using a logic analyzer that sampled the debug port every 4ns.

6.3 Test Results

6.3.1 Maximum Operating Frequency

The CBC was designed to operate at a maximum frequency of 50 MHz for clock traces of zero length, and progressively slower for longer trace lengths due to their longer propagation delays (see section 4.3.2).

To date, all prototype chips operated at 40MHz in test circuit 2 (containing a 40cm long trace). In addition, three chips were tested extensively at higher frequencies, and they worked well at 45MHz. One of these prototypes also ran successfully at 50MHz, but only while it remained cool to the touch.

Generally speaking, 45 MHz is the upper limit where the average prototype IC could be expected to operate reliably with test circuit 2. No chip worked above 50MHz with 40cm clock traces.

6.3.2 Deskewing Performance

6.3.2.1 Estimated Deskewing Performance

The absolute deskewing error for a single clock bay can be broken down into three components.

- one delay element of error due to the DLL (see section 4.5)
- one delay element of error due to the propagation delay estimator (see section 4.6)
- < 250-400ps error inherent in the delay estimation technique itself (see section 3.5).

Assuming delay element delays of 175ps, total worst case error will be about 600-750ps.
However, this is not the full story because the parameter of interest is not absolute error, but the relative error between clock bays. In this case, the error inherent in the propagation technique will be less than 200ps (see section 3.5) since the loads and trace impedances are similar. However, at the same time, there is a larger error associated with the clock bay circuitry itself.

The reason for this is that the 2 delay elements of clock bay error may occur in different directions between two clock bays, so worst case error may be as high as 4 delay elements plus 200ps. This translates to approximately 900ps of deskewing error. In many cases, the observed error will tend to be smaller since the worst case error is not always likely to occur.

It is conceivable that the error may be even higher in some cases if transitions on the inputs to a clock bay’s phase detector or propagation delay sampling flip-flops occur at just the right moment to force these circuits to become metastable. If this happens, then the overall clock bay error may increase (i.e. 2 delay elements of error for the DLL or 2 delay elements of error for the propagation delay estimator).

### 6.3.2.2 Measured Deskewing Error

Figures 73 to 76 show typical waveforms captured from one run of the CBC at 45 MHz. The figures show three source waveforms and three destination waveforms for the 40, 22 and 10cm clock traces. The source waveforms are annotated with the labels $S_i$, where $i$ equals ‘S’ for the short trace, ‘M’ for the medium length trace, and ‘L’ for the long trace. Load waveforms are shown directly below each of the source waveforms, and are annotated in a similar manner using the labels $L_i$.

Initially the CBC is fully reset, so the line reflections are quite large (Figure 73). After ILL lock (Figure 74), the peak magnitudes of the reflections drop by approximately 1V due to
the automatic impedance matching. Here the three destination waveforms show an uncompensated skew of 1.3ns due to the propagation delay differences between the 40, 22 and 10cm lines of test circuit 2. After DLL lock (Figures 75 and 76), the skew between the destination loads is reduced to 600ps which demonstrates that the automatic propagation delay estimator is working.

In general, the waveforms are much like those predicted by the transmission line models of Chapter 3; however, there are some differences. Distortions on the short line’s source-end are expected due to the multiple reflections that return to the source during an edge transition, but, there are also unexpected overshoots and oscillations at the start of the medium and long lines’ source-end voltage steps. These oscillations settle within a few nanoseconds so the impedance sampling circuit does not get a chance to match to them. (Recall that impedance sampling does not occur until 3.5ns after an edge transition). The end results is that an uncharacteristic overshoot/undershoot appears at the destinations since the circuit is not matched to the initial “surge” impedance associated with the start of the source-end voltage step oscillations.

A definitive cause for these oscillations is not known at this time, but a primary suspect is the extra self and mutual inductance related to the tightly wound clock traces of the medium and long lines. To study these effects properly, a 2-D circuit extractor is needed that can properly model the wound up transmission lines.

Five short coupled HSPICE transmission lines are a poor substitute to a 2-D circuit extractor’s circuit model, but they hint that the oscillations seen are related to the tightly wrapped medium and long clock lines. Further evidence of this comes from the fact that the impedance matching (based on the degree of overshoot at the destination) is better for the short clock trace which also happens to be nearly straight.
Figure 73. Source and load waveforms for clock bay outputs during ILL and DLL reset.

Figure 74. Source and load waveforms during ILL lock and DLL reset. Impedance sampling occurs at approximately 5.5ns and 14.0ns.
Figure 75. Source and load waveforms after ILL and DLL lock.

Figure 76. Averaged (16X) source and load waveforms after ILL and DLL lock.
Experimental results indicate that the destination clock skew is more or less independent of clock speed, however it is dependent on die temperature. With 4069 CMOS inverters acting as clock loads, skew is nearly always less than 600ps for the five prototype chips that have been tested extensively to date. Skew was measured between 4V and 1V on both rising and falling edges.

In order to observe worst case skew, observations were made with the oscilloscope set to high (100X) persistence mode. In this mode, each waveform widens into a band that indicates its envelop of variation over long periods of time. As shown in Figure 77, worst case error is 800ps for the rising edge and 1200ps for the falling edge. This is in agreement with the estimate given in section 6.3.2.1.

The rising edge shows lower skew than the falling edge. This may be due to the fact that 1) the propagation delay measurements are performed only for rising edges, and 2) the DLL phase detector phase locks only upon rising edges.
Figure 77. Load waveforms before and after DLL lock (in 100X persistence mode). The upper waveforms were acquired prior to DLL lock, and the lower curves after lock.

6.3.3 Impedance Matching/ILL Jitter

By monitoring debug groups 0 and 1 with a logic analyzer, the ILL jitter could be measured directly in digital form.

Figure 78 shows the locking behaviour of the ILL. The final lock point jitters about two and (not shown here) occasionally three impedance settings. The NMOS ILL locked at a different impedance setting than the PMOS ILL. Based on simulations of the output driver, these tap settings correspond to impedances close to 70Ω, which is the impedance that was targeted when sizing the clock traces on the test board. Tests on different prototype chips indicate that the PMOS ILL locks onto impedances in the 68 to 60Ω range, while the NMOS ILL locks onto impedances in the 71 to 80Ω range.
Since the ILL locking results above were obtained while the DLL was reset, this may not represent the worst case ILL jitter. An examination of the ILL jitter after DLL lock showed that the jitter increased occasionally to cover 4 tap settings when the DLL was operational.

6.3.4 DLL Jitter

Debug group 2 displays the tap control word (TCW) of clock bay 0's TDL1 tapped delay line. Jitter was observed to be limited to only 2 delay elements (resulting in 1 delay element of error) over time scales of 8 μs. DLL lock for clock bay zero is plotted in Figure 79.
Figure 79. TDLI tap control word for clock bay 0 (in test circuit 2) before and after DLL lock.

6.3.5 Power Consumption

The CBC draws 0.75A to 0.85A DC depending on the frequency of operation. (at vdd=5V, V_{bn}=4V, and V_{bp}=1V). This translates to a power consumption of 3.75W to 4.25W. Power is a concern because high temperatures affect delay element delay; however, as discussed in section 5.4.3, different delay element designs can greatly reduce the power consumption of the CBC.
Chapter 7
Conclusion

This thesis demonstrates a technique of using signal trace reflections to obtain propagation delay estimates. As demonstrated in section 3.5, propagation delay-estimation error for clock traces longer than 5-10cm is generally less than 250ps to 400ps for a wide variety of signal trace configurations. Moreover, between traces of similar impedance and loading, there is a weak dependence of estimation error on wire length. This makes reflection based deskewing circuits, like the one presented in this thesis, potentially accurate.

As demonstrated by a working prototype clock buffer, automatic impedance matching and propagation delay estimation can be integrated into a single circuit that can continuously adapt to process, voltage, temperature and circuit loading variations. The design implemented for this thesis is a scalable CMOS clock buffer circuit that can handle multiple destination loads in a point-to-point configuration.

For a 0.8μm BiCMOS process, operational frequencies as high as 45-50MHz have been observed for prototype chips driving clock traces as long as 40cm. Skew at the clock load inputs is typically less than 700ps, in agreement with the estimated +/-2 delay elements of
error typically expected from the CBC architecture. Worst case error can however be as large as +/-4 delay elements. Impedance matching to the 70\(\Omega\) clock traces of the prototype test board, was effective at reducing undesirable reflections, and permitted successful operation of the propagation delay estimation circuitry.

The three bay CBC prototype fabricated for this thesis occupies an area of 2.8x3.2mm, and contains 34,000 transistors. Power consumption is on the order of 3.75W to 5W (depending on bias voltage levels and clock frequency). Each clock bay occupies approximately 1\(\text{mm}^2\) and contains 10,550 transistors.

### 7.1 Future Work

Improved deskewing performance can be obtained by taking advantage of smaller geometry CMOS processes that can implement delay elements as small as several tens of picoseconds. In addition, plus-or-minus half a delay element of error can be removed from the propagation delay estimator circuitry by implementing fractional delay elements as suggested in section 4.6. In general, the entire CBC architecture can benefit from the use of fractional delay elements to improve timing resolution.

Significant power savings can be obtained through improvements in delay element design as well. Section 5.4.3 describes one design that dramatically decreases the amount of DC power dissipation.

The maximum clock frequency of the current DLL architecture is limited by the fact that tapped variable delay lines cannot be updated while clock edges are present near their active tap. The only way to guarantee this in the CBC is to allow only one clock edge in the main DLL loop at one time. This limits the DLL loop delay to be no more than half a clock period. However, other architectures - using perhaps multiplexed tapped delay lines,
or analog delay lines - may not be restricted as such, and may be able to achieve higher operational frequencies.

Future work should consider optimizing the timing of the impedance sampling. Techniques for implementing this are discussed briefly in section 4.7.4.1. Optimal sampling can extend the range of transmission line impedances that can be successfully handled by the CBC's automatic impedance matching circuitry, and can reduce matching error when impedance locked.
References


