Perfect Timing II

Design Guide for Clock Generation and Distribution

Corporate Headquarters
Cypress Semiconductor
198 Champion Ct.
San Jose, CA 95134 USA
Tel: *800) 858-1810 (toll free in U.S.)
Direct: (408) 943 2600
Fax: (408) 943 4730
www.cypress.com

Europe Headquarters
Cypress Semiconductor
Muenchner Str. 15A
Zorneding bei Muenchen
D-85604 Germany
Tel: (49)8106-24480
Fax: (49)8106-20087

Asia Headquarters
Cypress Semiconductor
541 Orchard Road
#07-02 Liat Towers
Singapore 338881
Tel: (65) 735-0338
Fax: (65) 735-0228

Japan Headquarters
Cypress Semiconductor
Harmony Tower 17F
1-32-2 Honcho, Nakano-ku
Nakano-ku, Tokyo 164-0012 Japan
phone: (81)3-5371-1921
Fax: (81)3-5371-1955

Practical implementation of clock generation and distribution circuits for use in high-speed digital designs.
Perfect Timing
A Design Guide for Clock Generation and Distribution

Ernie Buterbaugh
Cypress Semiconductor

CONTRIBUTORS
Steve Gerber
Gary Giust
Kelly Maas
Craig McKelvey
John Wisniewski
Stuart Woodrow
Kris Young
Jianmei Zhu
Cypress Semiconductor
# Table of Contents

**Contributors**  ................................................................. i

| Chapter 1 | Introduction | 1-1 |
| Chapter 2 | Clock Buffer Basics | 2-1 |
|           | Early Buffers | 2-2 |
|           | Skew | 2-3 |
|           | Output Skew | 2-3 |
|           | Part-to-Part Skew | 2-4 |
|           | Propagation Delay | 2-5 |
|           | Uneven Loading | 2-5 |
|           | Input Threshold Variation | 2-5 |
|           | Non-PLL Based Clock Drivers | 2-6 |
|           | PLL-Based Clock Drivers | 2-7 |
|           | What is a PLL? | 2-8 |
|           | Zero Delay Buffer | 2-9 |
|           | Lead or Lag Adjustments | 2-9 |
|           | Using External Feedback | 2-11 |
|           | Frequency Multiplier | 2-12 |
|           | Specialty Application Clock Buffers | 2-12 |
|           | Differential Clocks | 2-12 |
|           | PECL Buffers | 2-13 |
|           | LVDS Buffers | 2-14 |
|           | Spread Aware Buffers | 2-14 |
|           | Which One to Choose? | 2-15 |
|           | Conclusion | 2-15 |

| Chapter 3 | Timing Budget | 3-1 |
|           | A Simple Timing Budget | 3-2 |
|           | Distributed Clocks | 3-2 |
Chapter 4  Clock Jitter ............................................. 4-1
  Deterministic Jitter ............................................. 4-1
  Random Jitter .................................................. 4-3
  Gaussian Statistics ............................................. 4-3
  Jitter Mathematics ............................................. 4-4
  Measuring the Jitter ............................................. 4-5
  Application Jitter ............................................. 4-6
  Cycle-to-Cycle Jitter ............................................. 4-6
  Period Jitter .................................................. 4-7
  Long-Term Jitter ............................................... 4-8
  Diagnostic Techniques ......................................... 4-9
  Conclusion .................................................... 4-9

Chapter 5  Power Supply Filtering ...................... 5-1
  Test Fixture .................................................... 5-1
  PLL Test ....................................................... 5-2
  PLL Noise Rejection ............................................. 5-3
  Non-PLL Buffer ................................................. 5-4
  Filtering the Noise ............................................. 5-5
  Measured Impact on PLL ....................................... 5-7
  Non-Sinusoidal Noise ............................................. 5-8
  Ferrite Bead Recommendations ................................ 5-9
  Resistive Filters ............................................... 5-9
  Supply Noise In Differential Signaling ........................ 5-10
  Bypass Capacitors ............................................. 5-11
  Voltage Droop .................................................. 5-11
  Capacitor Filtering ............................................. 5-12
  What Frequencies to Bypass ................................... 5-14
  Capacitor Types ............................................... 5-15
  The Bypass Rules ............................................... 5-17
  Conclusion .................................................... 5-17

Chapter 6  PCB Layout Considerations ....................... 6-1
  Power and Ground Planes .................................... 6-1
  Ground Island ................................................. 6-3
# Chapter 9: Electro-Magnetic Interference

- Causes of EMI: 9-2
- Measuring EMI: 9-4
- Reducing EMI: 9-5
- Clock Modulation: 9-6
- Harmonic Frequencies: 9-10
- Bandwidth Requirements: 9-12
- Down Spread and Center Spread: 9-14
- Modulation Rate: 9-15
- Tracking Spread Spectrum: 9-15
- Conclusion: 9-16

# Chapter 10: IBIS and SPICE Models

- SPICE Models: 10-1
- IBIS Models: 10-3
- IBIS Version 2.1 Update: 10-6
- IBIS Simulation: 10-8
- IBIS Tips and Tricks: 10-9
- IBIS vs. SPICE Simulation: 10-11
- Conclusion: 10-12

# Chapter 11: Probing High-Speed Clocks

- Oscilloscope Bandwidth: 11-1
- Sample Rate: 11-2
- Probe and Connection: 11-3
- Ground Leads: 11-4
- Conclusion: 11-6

# Chapter 12: Clock Generators

- Resonators and Crystals: 12-1
- Oscillators: 12-3
- Programmable Oscillators: 12-3
- Clock Generators: 12-3
- Calculating P, Q and Post Dividers: 12-5
- Parts Per Million (PPM): 12-6
- Fractional N: 12-7
- Part Configuration: 12-7
- Multiple PLLs: 12-8
- Conclusion: 12-8
Chapter 13  Cascading PLLs ................................. 13-1
  A Cascaded PLL ........................................ 13-1
  Acquisition and Tracking ......................... 13-2
  Jitter Accumulation .................................. 13-2
  Phase Noise ......................................... 13-3
  Tracking Skew ....................................... 13-9
  Selecting PLLs ...................................... 13-10
  Cascading PLLs ...................................... 13-10
  Conclusion ........................................... 13-11

Chapter 14  Skew Control ................................. 14-1
  Layout ................................................. 14-3
  Component Selection and Data Sheet Parameters 14-4
  Extrinsic Skew ...................................... 14-6
  Trace Matching ..................................... 14-7
  Other Skew Adjustment Methods ................. 14-8
  Induced Skew ....................................... 14-9
  Conclusion .......................................... 14-10

Chapter 15  Clock Component Packaging .......... 15-1
  Introduction ....................................... 15-1
  IC Packaging Functions ......................... 15-1
  Package Physical Comparisons .................. 15-2
  Package Electrical Characteristics .......... 15-3
  Package Thermal Characteristics ............. 15-3
  Solder Reflow Profile ......................... 15-6
  QFN Package Overview ......................... 15-8
  Printed Circuit Board Design Considerations for QFN Package ... 15-9
  QFN Thermal Pad .................................. 15-10
  Conclusion .......................................... 15-10

Chapter 16  Motherboard Clock Validation .......... 16-1
  Introduction ....................................... 16-1
  PC Clock Overview ................................ 16-1
  The Clock Architecture ........................... 16-2
  Measurement Procedures and Equipment .... 16-4
  Cycle-to-Cycle Jitter Measurements .......... 16-4
  Rise/Fall Time and Duty Cycle Measurement Procedures ... 16-7
  Serial Signaling Overview ...................... 16-8
  SRC CLOCKS ........................................ 16-12
  System Measurements ......................... 16-14
  Measuring Long-Term Jitter on the USB and Dot Clocks .... 16-16
  Dot Clock Long-Term Jitter Measurement with Histogram .... 16-19
  Conclusion .......................................... 16-22
Have you ever designed a clock circuit where you did something just because it had been done that way in the past? Or have you ever used a particular part without knowing why, purely because it was on another design? This happens quite often with today’s designs, particularly with clock generation and distribution circuits.

This book has been written by engineers for engineers who want to design clock circuitry using the best methods. Its focus is on the practical implementation of clock generation and distribution circuits for use in high-speed digital designs. This material was gathered from many time-tested designs, as well as from new techniques learned to address the needs for faster clock frequencies.

The ultimate goal is to have clean, solid clocks. Many companies now employ full departments dedicated to signal integrity. They perform simulations, design reviews, and a variety of analyses to ensure the best operation of the clock. There are several factors designers need to consider that affect the clock waveform. This book will address many of these key design issues for clocks.

Clock generators play a key role in designs today. In the pursuit of high-speed, many systems have adopted synchronous design styles. With this methodology comes the need for a variety of frequencies and many copies of the same clock. In most systems, these clocks need to be in phase with one another. If they are not, precious cycle time is lost. Skew between clocks becomes very important in keeping all of the devices operating at their peak rates. Specialized clock buffers have led the way in providing clean, accurate clock signals.

Delay between clocks has also been minimized with the use of phase-locked loops (PLLs). These devices give designers the flexibility to align clock edges or allow them to be moved either ahead or back in time to increase their data valid windows. They can also compensate for trace length delays and unique chip timings. Clock buffers aid an engineer in creating the best possible designs.
Chapter 2 discusses clock drivers with low-skew outputs and zero-delay buffers. This chapter discusses the clock drivers' principles of operation and highlights some typical applications. These two types of devices are the cornerstones of clock distribution and are designed to specifically generate clock signals with clean, symmetrical edges.

With synchronous systems, timing budgets are key to successfully meeting setup and hold times. Chapter 3 discusses the many factors that need to be addressed when calculating timing margins. Jitter, skew, phase error and a host of other variations can steal valuable time in a given cycle. But how do these contributors interact? This chapter addresses the necessary methods for analyzing the different factors. It also introduces a new method called Total Timing Budget™. This budget aids the designer in determining the contributors to the cycle time without over margining the numbers.

Jitter is an often-used term when designing with clocks, however, it has many misunderstood definitions. Chapter 4 discusses the definitions and the origin of jitter, and unravels the mysteries sometimes associated with data sheet labels. Each type of jitter is discussed as is how its measurements are taken. This allows the designer to assess the types of jitter that are really of a concern to the design. It also highlights the characteristics of jitter to aid in the isolation of the source that may cause excessive jitter problems.

There are many factors that can affect a clock's performance. Among them is the penetration of noise into the power system. Noise has the effect of adding jitter and delay to clock buffers and PLLs. Chapter 5, discusses the effects of noise and how they can be minimized. By using bypass capacitors and ferrite beads, clean, solid power can be achieved. This chapter shows a variety of results through extensive testing of clock systems.

Also vital in providing solid, clean power to the clocking components is the layout of the power planes. Chapter 6 discusses the techniques for printed circuit board design that provide solid power performance. From planes to cutouts, from vias to bypass capacitor wiring, this chapter gives a detailed explanation of the rules to follow. Also addressed are the layout rules for the individual clock signals. To provide error-free waveforms, a variety of signal layout considerations are given. Crosstalk, impedance imbalances, vias and line widths all play key roles in signal integrity and need to be considered when designing a clock circuit.

As digital clocks increase in speed, they become transmission lines. A variety of reflections can occur that will cause false triggers if they are not properly terminated. Chapter 7 discusses the many aspects of line termination for clock signals. There are several techniques available for terminating signals and, therefore, a variety of methods are discussed. With the popularity of differential signals in today's high-speed world, both LVPECL and low-voltage differential signaling (LVDS) termination is addressed.

When working with clock buffers and PLLs, engineers often ask, “How many PLLs can I put in a series?” Chapter 8 addresses this very question by detailing the nature of PLLs and their loop bandwidths.
Electro-magnetic interference, (EMI), is a very important factor in devices used by the general public. Many systems must pass rigorous testing standards before the product is released. The designer is often faced with fixing an EMI problem just prior to the final phase of the design. But on many occasions, it is not well understood what caused the EMI and, worse yet, what could be done about it. Chapter 9 discusses where EMI originates and how it is measured. There are some unique properties of clocking, such as harmonics, that are important to understand when determining the effects of EMI. This chapter explains these relationships and discusses ways to suppress these unwanted signals.

Before releasing a design for manufacture, simulating the clock signals is recommended to ensure adequate operation. The popular tools used today for signal integrity are the Input/Output Buffer Information Specification (IBIS) and Simulation Program with Integrated Circuit Emphasis (SPICE) model simulators. Chapter 10 shows these models and what information about the clock component is contained inside. This gives the engineers insight into what the simulation really represents.

Once the design is complete and boards are in test, how does an engineer validate the clock signal for proper operation? Can any probe or oscilloscope be used? Chapter 11 discusses the common pitfalls associated with probing high-speed digital clocks with inadequate equipment. It also outlines the methods that can be used to measure the actual clock signal.

And finally, Chapter 12 discusses the clock generator. The clock generator is an extension of the zero delay buffer, but with many added features and benefits. It allows for the generation of seemingly unrelated frequencies as needed for the many clocks in today’s synchronous systems. This chapter addresses how these devices work, as well as some of the key attributes that make clock generators indispensable.

Hopefully, you will find the information in this book clear and concise, and above all, beneficial. It is meant for you, the designer, as a reference handbook. If you have any comments or suggestions on new information that should be incorporated, we would like to hear from you. Please send an email to: perfect_timing@cypress.com
Clocks are the basic building blocks for all electronics today. For every data transition in a synchronous digital system, there is a clock that controls a register. Most systems use Crystals, Frequency Timing Generators (FTGs), or inexpensive ceramic resonators to generate precision clocks for their synchronous systems. Additionally, clock buffers are used to create multiple copies, multiply and divide clock frequencies, and even move clock edges forwards or backward in time. Many clock-buffering solutions have been created over the past few years to address the many challenges required by today’s high-speed logic systems. Some of these challenges include: High operating and output frequencies, propagation delays from input to output, output to output skew between pins, cycle-to-cycle and long-term jitter, spread spectrum, output drive strength, I/O voltage standards, and redundancy. Because clocks are the fastest signals in a system and are usually under the heaviest loads, special consideration must be given when creating clocking trees. In this chapter, we outline the basic functions of non-PLL and PLL-based buffers and show how these devices can be used to address the high-speed logic design challenges.

In today’s typical synchronous designs, multiple clock signals are often needed to drive a variety of components. To create the required number of copies, a clock tree is constructed. The tree begins with a clock source such as an oscillator or an external signal and drives one or more buffers. The number of buffers is typically dependent on the number and placement of the target devices. Figure 2.1 illustrates the concept of the clock tree.

In years past, generic logic components were used as clock buffers. These were adequate at the time, but they did little to maintain the signal integrity of the clock. In fact, they actually were a detriment to the circuit. As clock trees increased in speed and timing margins reduced, propagation delay and output skew became increasingly important. In the next several sections, we discuss the older devices and why they are inadequate to meet the needs of today’s designs. The definitions of the common terms associated with modern buffers follow. Finally, we address the attributes of the modern clock buffer with and without a PLL. The FTG that is often used as a clock source is a special type of PLL clock buffer. A discussion of these devices can be found in Chapter 12.
Early Buffers

A clock buffer is a device in which the output waveform follows the input waveform. The input signal propagates through the device and is re-driven by the output buffers. Hence, such devices have a propagation delay associated with them. In addition, due to differences between the propagation delay through the device on each input-output path, skew will exist between the outputs. An example of a non-PLL based clock buffer is the 74F244 that is available from several manufacturers. These devices have been available for many years and were suitable for designs where frequencies were below 20 MHz. Designers would bring in a clock and fan it out to multiple synchronous devices on a circuit card. With these slow frequencies and associated rise times, designers had suitable margins with which to meet setup and hold times for their synchronous interfaces. However, these buffers are not optimal for today’s high-speed clocking requirements. The 74F244 suffers from a long propagation delay (3 to 5 ns) and long output-to-output skew delays. Non-PLL based clock buffers have improved in recent years and use more advanced I/O design techniques to improve the output-to-output skew. As the clock period gets shorter, the uncertainty or skew in the clock distribution system becomes more of a factor. Since clocks are used to drive the processors and to synchronize the transfer of data between system components, the clock distribution system is an essential part of the system design. A clock distribution system design that does not take skew into consideration may result in a system with degraded performance and reliability.
**Clock Skew**

Skew is the variation in the arrival time of two signals specified to occur at the same time. Skew is composed of the output skew of the driving device and variation in the board delays caused by the layout variation of the board traces. Since the clock signal drives many components of the system, and since all of these components should receive their clock signal at precisely the same time in order to be synchronized, any variation in the arrival of the clock signal at its destination will directly impact system performance. Skew directly affects system margins by altering the arrival of a clock edge. Because elements in a synchronized system require clock signals to arrive at the same time, clock skew reduces the cycle time within which information can be passed from one device to the next.

As system speeds increase, clock skew becomes an increasingly large portion of the total cycle time. When cycle times were 50 ns, clock skew was rarely a design priority. Even if skew was 20% of the cycle time, it presented no problem. As cycle times dropped to 15 ns and less, clock skew requires an ever-increasing amount of design resource. Now typically, these high-speed systems can have only 10% of their timing budget dedicated to clock skew, so obviously, it must be reduced.

There are two types of clock skew that affect system performance. The clock driver causes intrinsic skew and the printed circuit board (PCB) layout and design is referred to as extrinsic skew. Extrinsic skew and layout procedures for clock trees will be discussed later in this book. The variation of time due to skew is defined by the following equation:

\[ t_{\text{SKEW}} = t_{\text{SKEW_INTRINSIC}} + t_{\text{SKEW_EXTRINSIC}} \]

Intrinsic clock skew is the amount of skew caused by the clock driver or buffer by itself. Board layout or any other design issues, except for the specification stated on the clock driver data sheet do not cause intrinsic skew.

**Output Skew**

Output skew \( t_{\text{SK}} \) is also referred to as pin-to-pin skew. Output skew is the difference between delays of any two outputs on the same device at identical transitions. Joint Electronic Device Engineering Council (JEDEC) defines output skew as the skew between specified outputs of a single device with all driving inputs connected together and the outputs switching in the same direction while driving identical specified loads. Figures 2.2 and 2.3 show a clock buffer with common input Cin driving outputs Co1_1 through Co1_n. The absolute maximum difference between the rising edges of the outputs will be specified as output skew. Typical output skew in today's high performance clock buffers is around 200 picoseconds (ps).
Part-to-Part Skew

Part-to-part skew (t_{DSK}) is also known as package skew and device-to-device skew. Part-to-part skew is similar to output skew except that it applies to two or more identical devices. Part-to-part skew is defined as the magnitude of the difference in propagation delays between any specified outputs of two separate devices operating at identical conditions. The devices must have the same input signal, supply voltage, ambient temperature, package, load, environment, etc. Figure 2.4 illustrates t_{DSK} from the preceding example. Typical part-to-part skew for today’s high performance buffers is around 500 ps.
Propagation Delay

Propagation delay (t\textsubscript{PD}) is the time between specified reference points on the input and output voltage waveforms with the output changing from one defined level (low) to the other (low). Propagation delay is illustrated in Figure 2.3. Non-PLL based devices in today's high performance devices range from 3 to 7 ns. PLL-based buffers are able to zero out this propagation delay with the aid of Phase Detectors, Loop Filters and Voltage Controlled Oscillators (VCOs).

Uneven Loading

When using a high-speed clock buffer or PLL, care must be taken to equally load the outputs of the device to ensure that tight skew tolerances are maintained. Inherent in each output of the clock driver is an output impedance that is mostly resistive in nature (along with some inductance and capacitance). When each of these resistive outputs is equally loaded, the tight skew specification of the clock driver is preserved. If the loads become unbalanced, the (RC) time constants of the various outputs would be different, and the skew would be directly proportional to the variation in the loading.

Input Threshold Variation

After the low skew clock signals have been distributed, the clock receivers must accept the clock input with minimal variations. If the input threshold levels of the receivers are not uniform, the clock receivers will respond to the clock signals at different times creating clock skew. If one load device has a threshold of 1.2 volts and another load device has a threshold of 1.7 volts and the rising edge rate is 1V/ns, there will be 500 ps of skew caused by the point at which the load device switches based on the input signal. Most manufacturers center the input threshold level of their devices near 1.5 volts nominal for (TTL) input devices. This input threshold will vary slightly from manufacturer to manufacturer especially as conditions (such as voltage and temperature) change. The TTL specification for the input threshold level is guaranteed to be a logic high when the input voltage is above 2.0 volts and a logic low when the input voltage level is below 0.8 volts. This leaves a 1.2-volt window over voltage and temperature. Components with Complementary Metal Oxide Semiconductor (CMOS) rail swing inputs have a typical input threshold of V\textsubscript{CC}/2 or about 2.5 volts, which is much higher than the TTL level. If the threshold levels are not uniform, clock skew will develop between components because of these variations. There are many I/O standards which have emerged and all must be taken into consideration when providing clocks to different subsystems. Table 2.1 listed below which lists the more prevalent standards along with the input threshold voltages.
There are two main types of modern clock driver architectures: a buffer-type device (non-PLL) and a feedback-type device (PLL).

In a buffer-style (non-PLL) clock driver, the input wave propagates through the device and is “re-driven” by the output buffers. This output signal directly follows the input signal and has a propagation delay (tPD) that ranges from 5 ns to over 15 ns. These devices differ from the buffers in the past such as the 74F244 in that they are designed specifically for clock signals. On a 74F244, there are eight inputs and eight outputs. To create a one to eight buffer, all eight inputs are tied together. This causes excess loading at the inputs on the driving signal. A one to eight clock buffer has only one input and hence only one load. The output rise and fall times are also equally matched and therefore do not contribute to duty cycle error. With their improved I/O structure, the pin-to-pin skew is kept to a minimum. The output skew of this device, if it is not listed on the data sheet, can be calculated by subtracting the minimum propagation delay from the maximum propagation delay.

**Table 2.1 Input Threshold Voltage**

<table>
<thead>
<tr>
<th>I/O Standard</th>
<th>VCC/IO</th>
<th>VREF</th>
<th>VIH(min)</th>
<th>VIL(max)</th>
<th>VIL(min)</th>
<th>VIL(max)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.8V Interface</td>
<td>1.8V</td>
<td>—</td>
<td>1.08</td>
<td>2.25</td>
<td>-0.3</td>
<td>0.682</td>
</tr>
<tr>
<td>LVCMOS2</td>
<td>2.5V</td>
<td>—</td>
<td>1.7</td>
<td>3.0</td>
<td>-0.3</td>
<td>0.7</td>
</tr>
<tr>
<td>LVCMOS3</td>
<td>3.3V</td>
<td>—</td>
<td>2.0</td>
<td>3.9</td>
<td>-0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>LVTTL</td>
<td>3/3.3V</td>
<td>—</td>
<td>2.0</td>
<td>3.9</td>
<td>-0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>GTL+</td>
<td>—</td>
<td>1.0V</td>
<td>1.1</td>
<td>1.3</td>
<td>-0.3</td>
<td>0.9</td>
</tr>
<tr>
<td>HSTL Class I</td>
<td>1.5V</td>
<td>0.75V</td>
<td>0.8</td>
<td>1.9</td>
<td>-0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>HSTL Class II</td>
<td>1.5V</td>
<td>0.75V</td>
<td>0.8</td>
<td>1.9</td>
<td>-0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>HSTL Class III</td>
<td>1.5V</td>
<td>0.9V</td>
<td>0.78</td>
<td>1.9</td>
<td>-0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>HSTL Class IV</td>
<td>1.5V</td>
<td>0.9V</td>
<td>0.78</td>
<td>1.9</td>
<td>-0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>SSTL2 Class I</td>
<td>2.5V</td>
<td>1.25V</td>
<td>1.33</td>
<td>3.0</td>
<td>-0.3</td>
<td>1.17</td>
</tr>
<tr>
<td>SSTL2 Class II</td>
<td>2.5V</td>
<td>1.25V</td>
<td>1.33</td>
<td>3.0</td>
<td>-0.3</td>
<td>1.17</td>
</tr>
<tr>
<td>SSTL3 Class I</td>
<td>3.3V</td>
<td>1.5V</td>
<td>1.5</td>
<td>3.9</td>
<td>-0.3</td>
<td>1.5</td>
</tr>
<tr>
<td>SSTL3 Class II</td>
<td>3.3V</td>
<td>1.5V</td>
<td>1.5</td>
<td>3.9</td>
<td>-0.3</td>
<td>1.5</td>
</tr>
</tbody>
</table>

**Non-PLL Based Clock Drivers**

There are two main types of modern clock driver architectures: a buffer-type device (non-PLL) and a feedback-type device (PLL).

In a buffer-style (non-PLL) clock driver, the input wave propagates through the device and is “re-driven” by the output buffers. This output signal directly follows the input signal and has a propagation delay (tPD) that ranges from 5 ns to over 15 ns. These devices differ from the buffers in the past such as the 74F244 in that they are designed specifically for clock signals. On a 74F244, there are eight inputs and eight outputs. To create a one to eight buffer, all eight inputs are tied together. This causes excess loading at the inputs on the driving signal. A one to eight clock buffer has only one input and hence only one load. The output rise and fall times are also equally matched and therefore do not contribute to duty cycle error. With their improved I/O structure, the pin-to-pin skew is kept to a minimum. The output skew of this device, if it is not listed on the data sheet, can be calculated by subtracting the minimum propagation delay from the maximum propagation delay.
The 10 ns \( t_{PD} \) clock driver delay shown in Figure 2.5 does not take into account the affects of the board layout and design. These types of devices are excellent for buffering source signals such as oscillators where the output phase does not need to match the input. A variety of the non-PLL based buffers are available on the market today and typically range from as few as 4 outputs to as many as 30. Some devices also include configurable I/O and internal registers to divide the output frequencies.

Among the highest performance non-PLL based Low Voltage CMOS (LVCMOS) clock buffers available today is the B9940L. The B9940L is a low-voltage clock distribution buffer with the capability to select either a differential LVPECL or a LVCMOS/LVTTL compatible input clock. The two clock sources can be used to provide for a test clock as well as the primary system clock. All other control inputs are LVCMOS/LVTTL-compatible. The eighteen outputs are 2.5V- or 3.3V-compatible and can drive two series terminated 50-Ohm transmission lines. With this capability, the B9940L has an effective fanout of 1:36. Low output-to-output skews of 150 ps, a device to device skew of 750 ps, and a high-end operating frequency of 200 MHz, makes the B9940L an ideal clock distribution buffer for nested clock trees in synchronous systems.

These devices still face the problems of device propagation delay. The propagation delay through these devices is about 5 ns. This delay will cause skew in systems where both the reference clock to the buffer and the outputs of the buffer need to be aligned. These devices also have the drawback that the output waveform is directly based on the input waveform. If the input waveform is a non-50% duty-cycle clock, the output waveform will also have a less-than-ideal duty cycle. Expensive crystal oscillators with tight tolerances are needed when using this type of buffer in systems requiring near 50/50 outputs.

These devices also lack the ability to phase adjust or frequency multiply their outputs. Phase adjustment allows the clock driver to compensate for trace propagation delay mismatches and setup and hold time differences, and frequency multiplication allows the distribution of high and low frequency clocks from the same common reference. Expensive components and time-consuming board routing techniques must be used to compensate for the functional shortcomings of these buffer-style clock driver devices. PLL-based devices have been incorporated to address all of these shortcomings.

**PLL-Based Clock Drivers**

The second type of clock distribution device uses a feedback input that is a function of one of its outputs. The feedback input can be connected internally or externally to the part. If it’s an external feedback, a trace is used to connect an output pin to the feedback pin. This type of device is usually based upon one or more PLLs that are used to align the phase and frequency of the feedback input and the reference input. Since the feedback input is a reflection of an output pin, the propagation delay is effectively eliminated. In addition to very low device propagation delay, this type of architecture enables output signals to be phase shifted to compensate for board-level trace-length mismatches. Outputs can be selectively divided, multiplied, or inverted while still maintaining very low output skew.
PLLs have a number of desirable properties that include the ability to multiply clock frequencies, correct clock duty cycles and cancel out clock distribution delays. Many PLL-based clock buffers have been brought to market in recent years to aid clock tree designs that require zero propagation delay from the input signal to the output. A completely integrated PLL allows alignment in both the phase and the frequency of the reference with an output. We will look at some of the more prevalent PLL-based clock buffers and their features in the following sections.

What is a PLL?

The basic PLL is a feedback system that receives an incoming oscillating signal and generates an output waveform that oscillates at the same frequency as the input signal. It is comprised of a phase/frequency detector (PD), a low-pass filter, and a voltage-controlled oscillator as shown in Figure 2.6. In order for the PLL to align the reference (REF) input with an output, the output must be fed back to the input of the PLL. This feedback (FB) input is used as the alignment signal on which all other outputs are based.

![Figure 2.6 PLL Block Diagram](image)

The Phase Frequency Detector (PD) evaluates the rising edge of the REF input with respect to the FB input. If the REF input occurs before the FB input indicating that the VCO is running too slowly, the PD produces a Pump Up signal that lasts until the rising edge of the FB input. If the FB input occurs before the REF input, the PD produces a Pump Down signal that is triggered on the rising edge of the FB input and lasts until the rising edge of REF. This Pump Down pulse forces the VCO to run slower. In this way, the PD forces the VCO to run faster or slower based on the relationship of the REF and FB inputs. The output of the VCO is the internally generated oscillator waveform. The input voltage that controls the frequency of the VCO is a measure of the input frequency — as the input frequency changes so does this voltage. The PLL is designed to operate within a limited band of frequencies. If the input frequency is outside this band, the circuit will not lock-on to the input signal and FREF and FOUT will be different. As long as FREF remains within the tracking range of the circuit, FOUT = FREF. However, if FREF moves out of range, the circuit goes out of lock, and once again the input and internal frequencies will be different. In the absence of a REF input, the condition of the output is device-specific. For instance, with the loss of a reference input, the Cypress CY2308 ZDB will tri-state all outputs. However, the outputs of the CY7B991V operate at the device’s slowest speed while the outputs of the CY7B994V...
will run at their highest frequency. Therefore, the specifics of the device need to be known if the design will be placed in this condition. (There are now buffers that support dual clock inputs if the loss of an input clock is expected.) The filter converts these Pump Up and Pump Down signals into a single control voltage ($F_{\text{CONT}}$) and its magnitude is dependent on the number of previous Pump Up and Pump Down pulses that have occurred. The range of the voltage produced by the filter is guaranteed to force the VCO into any frequency within the selected frequency range.

**Zero Delay Buffer**

A zero delay buffer (ZDB) is a device that can fanout one clock signal into multiple clock signals with zero delay and very low skew between the outputs. This device is well suited for a variety of clock distribution applications requiring tight input-output and output-output skews. A simplified diagram of a ZDB is shown in Figure 2.7. A ZDB is built with a PLL that uses a reference input and a feedback input. The feedback input is driven by one of the outputs. The phase detector adjusts the output frequency of the VCO so that its two inputs have no phase or frequency difference. Since the PLL control loop includes one of the outputs and its load, it will dynamically compensate for the load placed on that output. This means that it will have zero delay from the input to the output that drives feedback independent of the loading on that output. Note that this is only the case for the output being monitored by the Feedback input and all other outputs have an input to output delay that is affected by the differences in the output loads. Please see the section “Lead or Lag Adjustment” for a discussion of this topic.

The Cypress Semiconductor CY2308 is a dual bank, general purpose ZDB providing eight copies of a single input clock with zero delay from input to output and low skew between outputs. This popular buffer is designed for use in a variety of clock distribution applications and will be used throughout this book as the typical Zero Delay, PLL-based buffer. The capability to externally connect the feedback path on the device provides skew-control and opens up opportunities for some interesting applications.

**Lead or Lag Adjustments**

Lead can be defined as the output of the buffer transitioning earlier in time than the input reference signal. It can also be viewed as negative delay. Lag, on the other hand, is the output clock transitioning later in time than the input and is a positive delay. To adjust the lead or lag of the outputs on the CY2308, we must understand the relationships between REF and FBK, and the relationship between the output driving FBK and the other outputs. First, we need to understand a few properties of PLLs. The PLL senses the phase of the FB pin at a threshold of $V_{DD/2}$ and compares it to the REF pin at the same $V_{DD/2}$ threshold. All the outputs start their transition at the same time (including the output driving FBK).
Changing the load on an output changes its rise time and therefore how long it takes the output to get to the $V_{DD}/2$ threshold. Using these properties to our advantage, we can then adjust the time when the outputs reach the $V_{DD}/2$ threshold relative to when the REF input reaches the $V_{DD}/2$ threshold. The output driving FB however cannot be adjusted; it will always have zero delay from the REF input at $V_{DD}/2$. Loading the output used for the feedback more heavily can advance in time the other outputs. The other outputs can also be delayed in time by loading the output for the feedback more lightly than the other outputs. Figure 2.8 shows how many picoseconds the outputs are moved versus the difference in the loading between the feedback output and the other outputs. As a rough guideline, the adjustment is 50 ps/pF of loading difference. Note that the ZDB will always adjust itself to keep the $V_{DD}/2$ point of the output at zero delay from the $V_{DD}/2$ point of the reference. If the application requires the outputs of the zero delay buffer to have zero delay from another output of the reference clock chip, the output of the clock chip that is driving the ZDB must be loaded the same way as the other outputs of the clock chip or the outputs of the ZDB will be advanced/delayed with reference to those other outputs.

Adding additional capacitance beyond 30 pF is not suggested due to the possibility of degrading the clock edges and adding more jitter to the outputs.

Adjusting the lead or lag of the output skew with a capacitor has its benefits. However, it does have imperfections because of variations in the capacitor itself. For small delay adjustments, it is more precise to use trace delay that matches the needed lead or lag times. For larger amounts of delay, a programmable skew device such as the CY7B994V should be considered.
Using External Feedback

Many ZDBs have an open external feedback path that is simply closed by driving any output into the FB pin for ZDB operation. However, the feedback path can be used for other interesting applications. Using a discrete delay element in the feedback path will generate outputs that lead the input signal. Sometimes designs require some copies of a clock that are early compared to the remaining copies of the input clock. Figure 2.9 shows a circuit implementation to generate such early clocks using a ZDB.

Another simple approach to lead or lag output clocks is to insert trace delay into the feedback path. The outputs of the buffer will lead the input by the amount of trace delay added in the feedback path. This approach provides a precise method for delay adjustment. Some designers will embed a very long trace into the board from an output pin to the feedback pin. At the ends of each trace segment, the designer places pads for zero ohm resistors. This allows for incremental additional delay into the feedback path to align the outputs to the precise phase. Figure 2.10 shows an example where the feedback path is 5 inches shorter than the other output traces. This initially allows the remaining outputs to be later in time than the input. By placing a capacitor on the feedback line, the outputs can be moved forward in time.
Frequency Multiplier

By using an external divider in the feedback path we can also create a frequency multiplier out of a simple ZDB. As shown in Figure 2.11, an /N divider in the feedback path will cause all outputs to run at a frequency which is N times the input frequency. Whatever the multiplication factor, input and output frequencies must be in the range of the PLL.

![Figure 2.11 Frequency Multiplier](image)

Specialty Application Clock Buffers

This chapter has focused primarily on general purpose clock drivers and zero delay buffers. However, there are a number of clock buffers available that have been designed to fit the requirements of particular subsystems that require precision clocking. For example, there are devices with frequencies that comply with specific standards such as MPEG video and T1 communication, and a variety of other specifications. There are also clock components designed to address a particular application. An example of this is the SDRAM clock buffer.

DDR SDRAMs uses double data rate architecture to achieve high-speed operation. The double data rate architecture is essentially a 2n-prefetch architecture with an interface designed to transfer two data words per clock cycle at the I/O pins. A single read or write access for the DDR SDRAM consists of a single 2n-bit wide, one-clock-cycle data transfer at the internal DRAM core and two corresponding n-bit wide, one-half-clock-cycle data transfers at the I/O pins. Very specific clocking requirements have been specified by JEDEC to ensure proper operation of these DDR SDRAMs. DDR SDRAMs require high-speed SSTL-2 clocking solutions and the v857 has emerged to support these requirements. The v857 is offered by many vendors and is a high-performance, low-skew, low-jitter PLL clock driver. It takes one pair of differential input signals and fans out to 10 pairs of differential output with low skew and low jitter at SSTL-2 voltage levels. All data inputs and outputs are SSTL_2 level compatible with JEDEC standard for SSTL-2 and can drive up to 14 DDR SDRAM loads.

Differential Clocks

There are two common electrical methods to transmit data from a source to a destination. One method uses a “single-ended” signaling concept that makes use of two conductors between the transmitter and receiver. It uses a dedicated signal-line to send the signal from transmitter to receiver and a common ground return shared by all signals. The other
method is a differential signaling where true and complement forms of the signal are sent from the transmitter to the receiver. While this also uses two conductors between the transmitter and receiver, they now both carry active signals and neither is shared with other signals. Differential signaling uses twice as many signal lines as a single-ended signaling.

Differential signaling has a number of important advantages over the single-ended signaling because of the ability of the differential receiver to reject any signal that is common to both lines. This ability is commonly referred to as common-mode noise rejection. The common mode rejection occurs because the receiver is only sensitive to a difference between the two inputs. When the two differential paths are closely linked, the noise will be apparent on both signals and rejected at the destination. With a single ended approach, the external noise will be apparent on the signal line itself. Because the differential signaling method rejects the common-mode noise signals, lower voltage levels can be used for reliable transmission of serial data. An additional benefit from the lower voltages used for differential signaling is reduced comparable power levels.

Differential PECL, LVPECL and LVDS clocks have become quite popular means of clocking high-speed logic in recent years.

**PECL Buffers**

As clock speeds increase above 100 MHz, noise immunity is of particular concern. PECL and LVPECL clocks are particularly good for clocking high-speed devices. More and more devices are starting to require PECL clock inputs to drive their logic. Framers, serializer/deserializer (SERDES), Switch Fabrics and FPGAs are among the latest devices supporting PECL and LVPECL inputs.

LVPECL stems from emitter coupled logic (ECL) but uses a positive rather than a negative supply voltage. It also uses 3.3 volt power supply rather than 5V. The ECL VDD and VEE pins have traditionally been powered from a -5.2V supply, VDD being grounded and VEE set at -5.2V where the intent is to achieve the lowest VDD noise by grounding the VDD pins. In more recent designs, however, ECL is often used with +5.0V instead of -5.2V and is commonly referred to as PECL (VDD set to +5.0V and VEE tied to ground). Since VDD noise is not a major concern, this permits the use of a standard logic supply.

As clock speeds rise beyond 100 MHz, the advantages of using ECL and PECL become more obvious. Most of these advantages involve the use of differential signal transmission. Differential signals are less susceptible to ground noise problems as all noise becomes common-mode. Single-ended CMOS is much more susceptible, since ground bounce and other noise affect logic thresholds, degrading noise immunity. Logic levels are less critical in differential signaling as the threshold can tolerate significant signal attenuation. Differential circuits also tend to generate less noise in the power supply. ECL is designed with termination resistors that allow high-frequency signals to propagate with minimal overshoot and reflection.
A new series of ZDB are becoming available that support LVPECL. The LVPECL differential driver is designed for low-voltage, high frequency operation to over 400 MHz. It significantly reduces the transient switching noise and power dissipation when compared to conventional single-ended drivers.

**LVDS Buffers**

LVDS (Low Voltage Differential Signaling) has become a popular means of transporting binary data across boards and backplanes in recent years as numerous low cost data buffers and FPGAs have begun to support this transmission scheme. To achieve high data rates and keep power requirements low, LVDS uses a differential voltage swing of only 350 mV (typical, in point-to-point applications). Furthermore, the LVDS CMOS current-mode driver design greatly reduces quiescent power supply requirements. LVDS data and clock buffers have rapidly emerged to support this standard as defined in TIA/EIA-644.

**Spread Aware™ Buffers**

The use of Spread Spectrum Timing (SST) technology has been popular in the motherboard and printer markets for some time. It is being used in virtually all motherboard designs using chipsets that support greater than 100-MHz busses. Spread spectrum timing signals are used in a variety of applications including PCI, CPU, and memory buses. Nearly all motherboard chipset vendors are designing their parts to work with spread spectrum timing signals. While the fast-paced motherboard market has quickly adopted the technology, it has also been embraced by other markets. The technology was developed solely for the purpose of reducing peak EMI.

Spread spectrum timing is a very effective tool for reducing EMI and may be easily integrated into many different systems without affecting other circuit elements. The one type of circuit element that may cause a timing problem when driven by a Spread Spectrum timing signal is a downstream PLL. A downstream PLL is a device that receives a reference timing signal from another PLL-based device, including those that use SST technology. “Downstream” may also apply to PLLs within Clock and Data Recovery (CDR) circuits. In a CDR application, the ability to properly track a modulated serial datastream is critical for clock extraction. For this reason, tracking skew is very important in downstream PLL applications. A ZDB used on a memory module to buffer the clock signal and provide the correct timing to latch the data could be considered as a likely downstream PLL. Although these devices may work properly with very stable reference inputs, if they cannot track a dynamically changing input signal, the output timing signals will not be synchronized to the system timing. Clock buffers that can track the variations of a spread spectrum input are said to be “spread aware.”

Spread Aware ZDBs are specifically designed to receive a spread spectrum modulated input signal. The PLL characteristics of Spread Aware devices will track the frequency modulation on the input signal with minimal accumulated tracking skew. Therefore, spread spectrum modulation present on the ZDB input signal would also be present on the output signals.
This will reduce the EMI emissions of the system. In addition, since the PLL tracking skew has been minimized, the system designer will have the benefit of the greatest possible timing margins.

**Which One to Choose?**

The clock circuit is usually critical to the operation of the system. If the clock circuit fails, the system fails. Because of this, the proper selection of the clock driver/buffer is usually critical to the success or failure of the design. When selecting the clock driver/buffer there are several parameters and characteristics for which the designer can watch to ensure the correct and reliable operation of the system. Since clock drivers tend to operate at high frequencies, it is important to ensure that the clock driver has low power dissipation. Unlike a buffer or latch that changes state only when one of the inputs changes, every output on the clock driver changes state every clock cycle at the fastest rate available in the system. This means that the clock driver is most likely switching more power in a smaller package than any other component in the system. While heat sinks and other cooling methods will help, it is best to start with a clock buffer/driver that inherently dissipates low power.

Choosing between a clock buffer or PLL is usually dependent upon the need for zero delay in the driver or the need to increase the clock frequency. If zero delay or increased frequency is needed, a PLL is the obvious choice. An advantage of a PLL over a buffer is the ability to correct the duty cycle in the clock line if there is distortion.

**Conclusion**

As system clocking speeds increase, the issues of skew and noise begin to receive primary consideration. Increasing the clock frequencies and requiring tighter tolerances in the clocking circuits will achieve future gains in system performance. Low skew clock buffers and PLL clock drivers will assist the designer in meeting the system requirements for speed, skew and noise. However, the clock circuit must be designed as a clocking system with consideration given to all aspects of the clock distribution network including the driver, receivers, transmission lines and signal routing. If the designer is aware of the problems that can develop, the difficulties can be avoided. The best method to identify the clock tolerances is to create a timing budget. The timing budget identifies the effects that each element of the tree has on the clock signals. The next chapter discusses the timing budget in detail.
Timing Budget

To ensure proper operation of a synchronous system, all of the timing elements must fit within the given cycle. When the sum of the timing elements exceeds the available time, the system will malfunction. It is therefore necessary to create a timing budget that identifies all of the factors that require a portion of the clock period and evaluate their influence. This allows the designer to understand how fast the design can operate and how much margin there is in the system. Chapter 3 focuses on the elements that contribute to the timing budget and highlights a newer, more precise way to accumulate the timing.

There are a variety of factors that contribute to the timing budget. Many of these are obvious, but some are a bit more subtle. To illustrate, we will build a timing model and add the associated timing elements piece by piece. Figure 3.1 shows a simplified design with the basic components of a synchronous system. Register A drives an output QA on every rising edge of CLOCK. QA attaches to a trace with a delay of \( t_{FLIGHT1} \) and is the input to some amount of combinatorial logic. The delay associated with this logic is represented as \( t_{LOGIC} \) and can be thought of as the “work” that takes place in the cycle. The result of this logic again travels through a trace with a delay of \( t_{FLIGHT2} \) and drives the resulting register. Register B is also clocked with the same CLOCK as Register A. The data from the combinatorial logic must arrive prior to the clock (\( t_{SETUP} \)) and must be held past the clock (\( t_{HOLD} \)) for the register to operate properly.

Figure 3.1 Simple Synchronous System
A Simple Timing Budget

Using this simplified model, we can begin to create a simple timing budget. The timing budget uses the following components:

- $t_{CLK}$ to $Q$: Delay from CLOCK rising to $Q_A$ valid
- $t_{FLIGHT1}$: Time taken to send $Q_A$ output to Logic input on a printed circuit board
- $t_{LOGIC}$: Time taken to perform logic operations
- $t_{FLIGHT2}$: Time taken to send logic output to Register B input on a printed circuit board
- $t_{SETUP}$: Time when Data at Register B must be valid before CLOCK rising
- $t_{HOLD}$: Time when Data at Register B must be valid after CLOCK rising

This breakdown of timing elements above assumes the clock signal has no jitter and that it reliably makes a constant period. However, most clock sources contain some amount of jitter. This not only includes clock generators, but applies to oscillators as well. Therefore, in addition to the above timing elements, the timing budget analysis must be concerned with the variance of the clock signal from one cycle to the next.

For example, if the first clock edge arrives later than nominal, but the second clock edge arrives earlier than nominal, the result shortens the effective time by which all activities must complete. To allow for the worst case in such a scenario, designers must allow for the maximum cycle-to-cycle jitter of the clock within the timing budget. Therefore, the following parameter is added to the simplified model above:

- $t_{REFjitter}$: Maximum Cycle-to-Cycle Jitter on Reference Clock

Distributed Clocks

It is often necessary to operate a large number of logic blocks, or subsystems, in the same clock domain. In order not to overload the clock signal, designers must create replicas of the reference clock using a buffer.

As discussed in the previous chapter, there are two main categories of clock buffers:

1. Clock Buffers
2. Zero Delay Buffers (ZDBs)

Figure 3.2 shows the simplified system that uses a fanout clock buffer to distribute its reference clock to four elements in the same clock domain. Although the buffer creates four copies of its input signal, the four outputs do not appear at the signal pins at precisely the same time. Because of different lengths of bond wires in a package and variances in silicon processes, some outputs in a buffer will arrive at their output pins sooner than others. The difference in the output timing is pin-to-pin skew. By inserting the fanout buffer to the design, additional timing elements must be added to complete the timing budget, as follows.
t\textsubscript{PD}: Maximum propagation delay for the input clock signal to make it through the buffer

t\textsubscript{SKEW}: Maximum pin-to-pin skew of the buffer

Figure 3.2 Clock Buffer Distribution

**Zero Delay Buffers**

The consequence of using a fanout buffer is that the propagation delay (t\textsubscript{PD}) can be very long and may leave insufficient time to complete all of the other tasks within a time period. Thus, designers can use ZDBs when they need to duplicate and send the reference clock source to more than one destination.

A ZDB uses PLL technology to replicate an input signal. It can also generate an output signal that is an integer multiple or dividend of the input while maintaining phase alignment. For example, the input reference could be 33.3 MHz and the resulting output could be 100 MHz or conversely, the input could be 100 MHz and the output 33.3 MHz.

By maintaining phase alignment with the input, ZDBs alleviate the large propagation delay of a fanout buffer. However, they bring a new set of considerations in meeting the timing budget.

First, the design must consider the cycle-to-cycle jitter of the ZDB instead of the reference input. Cycle-to-cycle jitter has very high frequency components and most ZDBs will filter out input jitter that is higher than a few megahertz. Consequently, t\textsubscript{REF jitter} no longer affects the timing budget and is replaced by the cycle-to-cycle jitter of the ZDB (t\textsubscript{jitter c-c}).

Second, ZDBs do not offer exactly zero offset between the input and the output. That difference is called either phase error or input-to-output skew. For ZDBs that offer an external feedback, this parameter is partially under the designer’s control. By adjusting the trace length of the feedback signal, designers can affect the timing relations between input
and output. The delay will still be represented as $t_{PD}$ in the timing budget, but it will have a smaller variation than a non-PLL based buffer. The delay can also be positive or negative and is discussed in more detail in the next sections.

Clock Timing Budget

When constructing the timing budget, the arrival of the clock edges is a key component. It is often the most complex portion as well. To properly understand the margins of the clock circuit, a window needs to be created that shows the earliest arriving clock and the latest arriving clock. These values are the most negative and most positive phase differences with respect to the reference clock. Figure 3.3 illustrates the timing window. The window is marked by the edge of the earliest clock ($T_{FIRST}$) and the edge of the latest clock ($T_{LAST}$).

![Timing Window Diagram](image)

Figure 3.3 Timing Window

With this window, we can determine the margins at the device that is to be clocked. To illustrate this in a system environment, suppose we have a circuit that is shown in Figure 3.4. A 25-MHz oscillator sources a CY26114 clock generator. The generator creates two copies of a 100-MHz clock and two copies of a 50-MHz clock and each clock drives a ZDB. A CY2509 ZDB is used for the 100-MHz signals while a CY2309 is used for the 50-MHz clocks. This is a typical system where different types of ZDBs may be used to match speed and function.

Although half of the outputs are 50 MHz and the other half are 100 MHz, this particular system requires that all outputs be phase aligned to each other. A reference clock is used to evaluate how far in time each clock phase occurs. In this case, the reference is the output of the clock generator since any delays and skew prior to this point have no effect on the outputs of the ZDBs. To create the timing window, the parameters for skew, delay, and jitter that were identified earlier need to be added, as shown in Figure 3.5.
For the components in this design, there are several parameters that are required from their respective data sheets.

<table>
<thead>
<tr>
<th>Component</th>
<th>Jitter</th>
<th>Skew</th>
<th>Delay</th>
</tr>
</thead>
<tbody>
<tr>
<td>CY26114</td>
<td>200 ps</td>
<td>250 ps</td>
<td>N/A</td>
</tr>
<tr>
<td>CY2509</td>
<td>100 ps</td>
<td>250 ps</td>
<td>±350 ps</td>
</tr>
<tr>
<td>CY2309</td>
<td>200 ps</td>
<td>250 ps</td>
<td>±350 ps</td>
</tr>
</tbody>
</table>

The timing window measures the earliest clock to the latest clock at the destination. For each of the outputs, the window is generated as in Figure 3.6. The window represents the earliest clock to the latest clock at all of the destination devices. To create the window, all of the timing elements from the reference clock to the destination clocks need to be added. A window should be created for each part and the widest opening should be chosen. In this case, the CY2309 has the largest jitter, delay and skew components so we will show the window for those outputs. Begin by adding all of the fixed skews in the clock paths such as pin-to-pin skew. Some data sheets will specify a bank-to-bank skew and a part-to-
part skew. Use whichever applies to the clock tree configuration. Notice that the window will widen if there is trace skew in the path between the buffers and the loads or between the clock generator and the buffers. This model assumes they are equally matched in length and there is no variation. The value of 0 ps is used in the window. Then add any other forms of delay associated with the components. This includes any input to output variations. Although PLLs “zero” the part delay, there is a delay tolerance and it needs to be included. Finally, the jitter for all components is included at both ends of the window.

Figure 3.6 Data sheet Timing Window

From this example, the window is shown to be 2.0 ns. With the system operating at 100 MHz, the window represents a large variation. However, these are all worst-case numbers and there are hefty margins associated with each value.

**Measured Data**

To evaluate how data sheet values match against actual components, the design in Figure 3.4 was fabricated and tested. Each clock output was measured for skew, delay, and jitter. The maximum positive phase delay, or in other terms the latest clock with respect to the reference was measured to be +571 ps. The maximum negative phase delay, that is the earliest arriving clock is measured as –346 ps. These numbers include the maximum jitter measured of 120 ps. Figure 3.7 shows the variations at each clock buffer and the total timing window. Each box in this figure represents the ZDBs used in the clock tree of Figure 3.4. Each output of the ZDB is measured with respect to the reference clock (from the CY26114) and the earliest and latest delay is recorded for each buffer. Notice that some of the clocks precede the reference while others are later in time. The earliest and latest clock on any of the outputs bound the timing window.

The measured timing window yields a result of 917 ps while the calculated timing window using data sheet parameters was 2 ns! This is a significant difference. Because the simple summation of data sheet values produces a very conservative result, some clock buffer manufacturers are including a new parameter that combines the jitter, delay, and skew. One of the terms that is used for this value is Total Timing Budget (TTB).
Figure 3.7 Measured Timing Window

**Total Timing Budget (TTB) Window**

Since PLL devices became commercially available over ten years ago, the method by which data sheets report jitter, skew, and other performance parameters have been the same. In the meantime, operating frequencies have increased considerably. As seen in the previous example, the estimated budget using these numbers can become very large.

**Minimizing the Window**

The TTB parameter includes the delay, skew and jitter components specific to a clock buffer and provides one number with which to work. TTB can be defined as the maximum window in which all transitions of any output clock of a device will occur with respect to its input reference clock. This substantially reduces the task of calculating the timing budget numbers. It also avoids the compounding of guard bands associated with the individual numbers. It should be noted that TTB doesn’t replace the traditional numbers as they may define other key aspects needed during the circuit design.

The TTB parameter is guaranteed over the device’s data sheet temperature, voltage, frequency, and input clock’s rise time range. It also includes the manufacturer’s process variables and hence is an absolute maximum guaranteed window for all conditions that are stated in the data sheet.

Some data sheets provide TTB values for worst-case conditions and frequencies. However, this too may add more guardband than necessary for a design operating at a specific frequency, whereas the specification will include the complete operating range. Thus, TTB values may be published not only for the maximum frequency but also for specific and common operating rates. For instance, a ZDB that operates at up to 200 MHz may have a published TTB number for the 200 MHz, and for 100 MHz, 66.6 MHz and 33.3 MHz, as these are common frequencies used in designs. This provides more accurate information to the designer for the timing budget.

To illustrate the benefit of TTB, let’s analyze the parameters associated with a ZDB that is
used in a timing budget. The three factors are pin-to-pin skew, input-to-output-delay (phase error), and cycle-to-cycle jitter. Figure 3.8 shows these three parameters for a typical ZDB.

![Figure 3.8 ZDB Traditional Timing Budget](image)

In the figure, T represents the contribution due to temperature, R is the input rise time variation, F is the frequency of operation, V is the voltage fluctuation and P is the process variation. The amount of time associated with these parameters in this figure is 1.5 ns, which includes a guard band for each individual parameter.

For the same component, the TTB window is much smaller, as shown in Figure 3.9.

![Figure 3.9 ZDB TTB](image)

The TTB number, which includes all of the variations as before, specifies 1.3 ns instead of the 1.5 ns. This gives 200 ps back to the designer for timing margin. This TTB number is for the full operating frequency range of the buffer. However, if the system is operating at a fixed rate of 133 MHz, a different, more precise TTB number may be provided. This allows the designer to use a specific TTB number for 133 MHz to further enhance the budget. Figure 3.10 shows this time line. Notice that the value F does not appear as it represents the frequency variation that is now fixed.

![Figure 3.10 ZDB TTB at Frequency](image)

This has dramatically improved the variables used in the timing budget numbers. By using the TTB at 133 MHz, the timing budget parameters for the clock buffer enhanced the budget from 1.5 ns to 696 ps, which is a savings of over 800 ps. This is a significant savings for a system with a cycle time of 7.5 ns.
**TTB Characterization Data**

Creating a TTB number for a component follows the normal path of part characterization. The difference between a TTB characterization and a standard (non-TTB) characterization is that TTB requires additional measurements under a variety of conditions. Components from each of the process corners are tested in a variety of conditions that include: temperature, voltage, input clock condition, feedback clock condition and frequency. The numbers are tabulated and the TTB parameter is generated.

Figures 3.11 to 3.14 show the effects that temperature, voltage, frequency, and rise time of the feedback signal have on the budget windows. Notice in particular the delay difference in the frequency range. Using the delays associated with the exact system frequency can yield much better timing margins. This is true using the TTB method or using the traditional timing budget method. Note that these graphs illustrate the effect these parameters have on delay; however, the amount of variation is device-specific.

![Figure 3.11 Temperature Variation](image1)

![Figure 3.12 Voltage Variation](image2)

![Figure 3.13 Frequency Variation](image3)

![Figure 3.14 Feedback Rise-Time Variation](image4)
**Conclusion**

Creating a timing topology and allocating the timing budget is one of the fundamental efforts in system architecture for complex synchronous systems. This chapter illustrated some of the basic considerations on how selection of clock distribution components can impact timing budget considerations. Delay, skew, and jitter are all key elements that must be addressed.

Data sheets that list the timing parameters individually are not sufficient to meet a very tight timing budget. Instead, a parameter that represents the part’s total timing budget should be used. This number better reflects the operation of the device without excessive guard banding. In the absence of this number, designers should work closely with their timing solutions supplier to understand the projected performance of each clock device.
Jitter plays an important role when designing clock generation and distribution circuits. This is especially true when using PLL-based buffers. Excessive jitter can rob a design of its precious cycle time or cause data to be latched incorrectly. Understanding the concepts of jitter and knowing its definition can help you avoid these problems. There are also different ways to measure jitter and often they are confused with one another. This can lead to erroneous calculations and cause a system to malfunction. To further confuse the matter, jitter can be expressed in many different ways. Component data sheets often specify jitter in terms of absolute, peak-to-peak, root-mean square (RMS), one Sigma, and unit interval (UI). This chapter focuses on the definitions of the different types of jitter and how they are measured.

Fundamentally, clock jitter can be defined as the deviation in a clock’s output transition from its ideal position. Before discussing the specifics of jitter in clocks for high-speed digital designs, its sources in general should be identified. Jitter can be divided into two essential types: Random jitter (RJ) and non-random or deterministic jitter (DJ). Total jitter is equal to the sum of deterministic jitter and random jitter. But these two types of jitter have very different behaviors. It is helpful to understand the characteristic of each when evaluating jitter within a system.

**Deterministic Jitter**

Deterministic jitter is non-random and bounded. This type of jitter can be traced to a specific source. DJ is due to signal noise, crosstalk, power supplies, and other similar sources. High values of this type of jitter are usually an indication of a problem within the design. DJ is typically non-Gaussian/see the gaussian statistics section and tends to grow by fixed amounts. There are four categories that contribute to DJ: duty cycle distortion, data dependent, sinusoidal, and uncorrelated bounded. Some systems, when describing jitter, will specify the sinusoidal jitter (SJ) component separately.
Duty Cycle Distortion and Pulse Width Distortion are different names for the same thing. This type of jitter is also referred as half-period jitter. It is the difference in the width of the pulse with its output low versus the width of the pulse when the output is high. Ideally, these two widths are equal. When they are not, the crossing points for signal transitions (low-to-high and high-to-low) can be different. This is most evident when viewing a signal with an eye diagram.

Data Dependent Jitter and Inter-Symbol Interference describe the same condition but viewed from different perspectives. Data dependent jitter is measured in the time domain while inter-symbol interference is analyzed in the frequency domain. Data dependant jitter appears in a system when a pattern changes from a fixed stream of pulses to a unique bit position. The rising and falling edges are altered from one position to another based on a data stream. An example of this condition is a regular clocking waveform changing into double width pulses and back again. From the frequency domain perspective, it can be viewed as pulse spreading due to bandwidth limitations. This category of jitter is normally found in datastreams and is not typical in digital discrete clocks.

Sinusoidal Jitter alters the position of the rising and falling edges of the waveform by varying amounts. The amount of the timing displacement follows a regular sinusoidal pattern.
Uncorrelated bounded jitter comprises the remaining deterministic jitter not categorized by any of the three previous types. It is uncorrelated in that there is no bit pattern dependency but bounded as it has a fixed limit. Power supply noise and crosstalk are contributors of this type of jitter.

**Random Jitter**

Random jitter is often due to a large number of forces in nature; each of which follows uniform statistical probability. Random jitter follows Gaussian probability densities and its limits are unbounded. Because it is statistical, it forms a bell shaped curve with the tails extending towards infinity. In theory, its maximum values can be limitless and therefore Random Jitter is often bounded using standard deviations. In practice, arbitrary limits are imposed and peak-to-peak values are available. Random jitter normally comes from elements within the environment. It can be caused by thermal noise, radiation, semiconductor crystalline structures, and a variety of other indistinguishable components.

**Gaussian Statistics**

When many independent random factors interact and accumulate, data will follow a distribution called the Gaussian probability distribution function. This distribution is also called a Normal distribution. The Gaussian distribution has some special mathematical properties that form the basis of many statistical measurements.

A Gaussian curve follows a bell shaped distribution. The mean of the samples are aligned in the middle, hence are the highest point in the curve. This is also the point at which the horizontal axis marks its zero origin. As mentioned earlier, the tails of the curve extend towards infinity. Since the data is unbounded in theory, a set of windows is defined to represent various amounts of data.

The standard deviation, or one sigma (σ), is defined as containing 68.26% of all measurements on one side of the mean. Two sigma (2σ) contains 95.4% and four sigma (4σ) contains 99.994% of all measurements on a side of the mean. For pure Gaussian mathematics, all measurements are possible. Although, theoretically Gaussian jitter has no bounds, the probability of an event occurring very far from the median becomes extremely remote. In practice, there are bounds that do not exceed fourteen sigma in Gaussian systems. In general, plus or minus six sigma (contains 99.9999999% of all sampled data) is the maximum deviation from the mean to use for calculating peak-to-peak jitter for a Gaussian distribution of jitter. But there are some specific applications that use a higher sigma value.
Jitter Mathematics

The Gaussian distribution shows the samples graphically but there are several mathematical equations that need to be highlighted. Although test measurement equipment will provide these values, it is helpful to understand their meaning and how they are computed. This can be crucial when extracting jitter parameters from a device data sheet and relating it to the design requirements.

Since jitter in a clock is measured on an edge, or event in general terms, it can be represented as \( E \). The time at which the signal rises is therefore \( t_E \). The clock signal will have many rising edges, hence any occurrence of \( n \) events can be represented as \( E[n] \), which will occur at times \( t_E[n] \). Using these representations, instantaneous jitter between an ideal edge and the actual measured edge can be expressed as:

\[
J[n] = t_E[n]_\text{Ideal} - t_E[n]_\text{Actual}
\]

The IDEAL edge of a signal can easily be calculated from the period of the clock. This can be an adjacent edge or \( n \) edges away in time. Using \( n=0 \) for the reference edge, the IDEAL edge at time \( n \) is equal to the following:

\[
t_E[n]_\text{Ideal} = t_E[0] + n/f_{\text{CLOCK}}
\]

With the Gaussian distribution, sigma (or root-mean-square average) is often used to represent a portion of the data. This parameter is often specified in device data sheets as \( \text{Jitter}_{\text{RMS}} \). To calculate the RMS value, the mean must first be derived. Given \( N \) as the total number of samples in the measurement, the mean jitter amount can be expressed as:

\[
J_{\text{Mean}} = \frac{1}{N} \sum_{N=1}^{N} j[n]
\]

Using the mean, the one sigma, or RMS value from the mean is computed.

\[
\sigma_{\text{one sigma}} = \sqrt{\frac{1}{N-1} \sum_{N=1}^{N} (j[n] - J_{\text{Mean}})^2}
\]
Another common parameter specified in device data sheets is the peak-to-peak jitter. Graphically, it is the width of the base of the distribution. Mathematically, peak-to-peak jitter is derived from subtracting the minimum jitter in the sample set from the maximum jitter.

\[ J_{\text{PEAK-TO-PEAK}} = \max\{J[n]\} - \min\{J[n]\} \]

**Measuring the Jitter**

Due to the speeds of the high-speed digital signals, it is important to measure jitter with the correct equipment. Often, engineers will use oscilloscopes with persistence to measure jitter. Although this provides a rough measurement it is very inaccurate and suffers from trigger instability.

It is, therefore, required that specialized equipment be used to measure jitter. These are called Timing Interval Analyzers (TIAs) and specialized Digital Storage Oscilloscopes. A variety of jitter test equipment is available from companies such as LeCroy Corporation, Tektronix, and Agilent Technologies. These instruments not only allow the precise capture of data, they incorporate the mathematical analysis and graphical displays that easily extract jitter values.

Jitter is often displayed with the use of histograms. The vertical axis is the number of samples and the horizontal axis is time. The display shown in Figure 4.4 follows a Gaussian distribution and therefore indicates Random Jitter is the primary source. Notice that the tails do not extend to infinity and have fixed values because the sample size is finite. The mean jitter value is in the middle of the sample set and standard deviation is used as a measure of the dispersion about the mean. RMS, one sigma, and standard deviation all have equivalent meanings. The peak-to-peak jitter is the difference between the highest and lowest value captured and is represented as the width of the base.
When Deterministic jitter is coupled with the random jitter component, a non-Gaussian distribution with multiple nodes is created. Depending on the components, it may be bimodal, meaning two peaks, or it may be multi-modal, containing several peaks. Each peak will form a Gaussian distribution in itself.

Application Jitter

These basic principles can be applied further to specify and quantify jitter in high-speed digital clocks. Jitter, in its application sense, can be defined in three ways: Cycle-to-cycle jitter, Period jitter, and Long-Term jitter.

Cycle-to-Cycle Jitter

Cycle-to-cycle jitter is the change in a clock's output transition from its corresponding position in the previous cycle. This is also sometimes referred to as short-term jitter. This type of jitter is the most difficult to measure and usually requires a Timing Interval Analyzer (TIA) or some other specialized jitter test equipment. Figure 4.6 shows a graphical representation of cycle-to-cycle jitter. $J_1$ is the jitter value measured by subtracting the cycle time $t_2$ from cycle time $t_1$. The next jitter value, $J_2$, is the value measured by subtracting the cycle time $t_3$ from cycle time $t_2$. The maximum of the $J$ values measured over time, typically ten thousand cycles, is the maximum cycle-to-cycle jitter.

When designing clock and distribution circuits for synchronous systems or processor-based systems where a PLL may be downstream, cycle-to-cycle jitter is important. PLLs may not be able to track this high frequency jitter so timing skew may result.

There are several different definitions used to describe cycle-to-cycle jitter that may be seen in different data sheets. Cycle-to-cycle jitter is also known as adjacent cycle jitter. Often,
cycle-to-cycle jitter will be provided as a single number that correlates to the largest $J$ value recorded from all samples. This is the maximum variation from one cycle to the next and can lead or lag in time from the previous cycle. Therefore, this number is sometimes expressed in ± units.

Sometimes a device’s data sheet will specify a peak-to-peak value for the cycle-to-cycle jitter. This represents the difference of the minimum cycle and the maximum cycle throughout the sample set. Therefore, this number is greater than the normal cycle-to-cycle jitter and is larger than what is expected between adjacent cycles.

![Figure 4.6 Cycle-to-Cycle Jitter]

**Period Jitter**

Period jitter measures the maximum change in a clock’s output transition from its ideal position. Figure 4.7 shows a clock waveform where the first rising edge is fixed and the next rising edge is measured. This second edge may lead or lag the ideal cycle width. The maximum of this value measured over time, typically ten thousand cycles, is the maximum period jitter.

Period jitter can impact the performance of a synchronous system by cutting into the timing margin budget. It can affect cycle time as well as data setup and hold times. There are several ways to describe period jitter.

Many data sheets will also label period jitter as absolute jitter. These values and peak-to-peak jitter all describe the same measurement. The number is the difference between the earliest and the latest signal. Referring to a Gaussian distribution, this is measuring the width of the base of the peak. It can also be thought of, referring to Figure 4.7, as the latest $t_1$ minus the earliest $t_1$ recorded in the sample set.

Another method for indicating period jitter is using a form of an average. The methods of RMS, standard deviation, and one sigma all describe a subset of the peak-to-peak period jitter and mean the same thing. Again, from the Gaussian distribution, one sigma, or one standard deviation, is $68.26\%$ of the sample size. Jitter expressed in any of these terms uses the jitter measured from this subset of data collected. By definition, this value will be less than the peak-to-peak amount.
Long-Term Jitter

Long-term jitter measures the maximum change in a clock’s output transition from its ideal position over many cycles. This type of jitter measurement generally applies to a few specific applications. The term “many” depends on the application and the frequency of the clock. For PC motherboards and graphics applications, “many” usually refers to a period of 10 to 20 microseconds. For other applications it will be different. Figure 4.8 shows a graphical description of long-term jitter. Cycle 0 begins with the rising edge of the clock. Long-term jitter is the amount of deviation in cycle N of the ideal clock edge. Cycle N’s actual clock edge may lead or lag the ideal clock edge.

A classic example of a system affected by long-term jitter is a graphics card driving a display. Assume that a pixel of data is meant for a pixel at coordinates (10,24) on the screen. Because of a clock with excessive long-term jitter, this data may be driven at a pixel with coordinates (11,28) on the screen. Over an extended period of time, the data meant for pixel (10,24) may be driving a pixel far away from its ideal position. Since this shift is normally consistent over all pixels, the overall effect of the jittery clock is to cause an image shift from its ideal display position. This effect is sometimes called “running” of the screen.

In most general applications using clock buffers and generators, jitter is expressed in term of time such as picoseconds (ps). In some applications, often in the data communications sector, jitter is expressed in terms of UI (unit interval). UI normalizes the amount of jitter to the period of the clock. One UI is equal to the period, therefore jitter UI can be expressed as:

\[ \text{Jitter UI} = \frac{\text{Jitter (sec)}}{1 \text{ UI (sec)}} \]
Diagnostic Techniques

As seen from this chapter, there are a variety of sources that can contribute to excessive jitter. Power supply noise is a large contributor as outlined in Chapter 5 but other sources may be possible. The following paragraph describe the leading factors that cause excessive jitter in clock buffer and PLLs.

The power supply pin decoupling capacitance or layout may be ineffective. Check the board layout against the PLL recommended layout guideline for capacitor value and connections for minimum inductance.

The power supply noise is reaching the PLL supply pin. It is often easiest and fastest to verify this fact by supplying the PLL with an external power supply. Then check for any improvement. Although it can be cumbersome to lift the VDD supply pins, it provides positive feedback that the power supply caused the jitter. Care must be taken to maintain VDD pin decoupling capacitance. This technique avoids trying to characterize the supply noise, then trying to obtain power supply rejection curves that correspond to the frequencies of interest.

The input reference clock may have excessive noise or modulation within the PLL loop bandwidth. Connect the input reference clock to a spectrum analyzer and check the level of sidebands below the loop bandwidth.

The output trace has crosstalk to other switching signals on the board layout. The length and spacing of signal traces next to the clock trace should be checked. It is preferable to have a ground trace next to the clock lines or extra spacing between the other signals.

The measurement equipment or method is not appropriate and is providing false readings. Measuring jitter from an oscilloscope screen is not precise because of scope trigger jitter and retrace jitter. It is preferable to use a high sampling rate scope with dedicated jitter analysis firmware or software. This uses the scope-sampling clock that is more precise.

Verify that the component isn’t defective and that the chosen PLL meets the jitter design requirements.

Conclusion

Jitter is a common parameter associated with clock generation and distribution. Controlling excessive jitter is necessary for achieving the most efficient and error-free design. By understanding the terms and knowing the measurement techniques, the engineer can design and verify stable clock waveforms that offer solid performance.
Power Supply Filtering

The power supply is the foundation for providing optimal clock signal performance. As the quality of the voltage supply degrades, so does signal integrity. Noise in the supply will affect jitter and skew in clock buffers. The designer can choose whether the amount of timing degradation is acceptable to the total budget or incorporate filtering and layout techniques to minimize it. This chapter will address the application of ferrite bead filtering.

Ferrite beads offer a level of filtering to produce the absolute best quality in signals. But choosing the correct ferrite bead can be quite difficult. The specifications for beads often do not provide the information needed to select the right component. To evaluate the performance of power plane filters, a test set-up is needed that allows a controlled injection of “noise” into the supply. The effect on a variety of clock buffers and PLLs in terms of jitter and skew can then be analyzed.

Test Fixture

Figure 5.1 shows the test fixture used for the jitter and skew evaluations. The component in the middle of the diagram is the Device Under Test (DUT), which will be populated with various clock PLLs and buffers. A pulse generator supplies a controlled input waveform into the device. To view and capture the output signal, a Time Interval Analyzer (TIA) is used. A waveform generator is also attached to the input voltage of the DUT that will be used to create the noise component.
For the jitter contribution versus frequency plots, the entire supply plane is modulated by a sine wave source. A frequency is swept across the region of test (10 kHz to 5 MHz) to simulate the effects of noise propagating through the power plane. Every test sweeps through this range, as noise isn’t limited to a single frequency. The 0.1 µF VCC decoupling capacitors are left connected to each VCC pin that allows the DUT to operate in a similar manner to an actual operating environment where decoupling capacitors are populated. The noise signal passes through a 22 µF bulk capacitor and overdrives the 0.1 µF decoupling capacitors. The main supply will drive a slightly higher DC voltage than normal to compensate for the voltage drop across R1.

**PLL Test**

In the first evaluation, a PLL-based ZDB is the device under test. For a PLL, phase offset jitter is the critical parameter that needs to be measured. Figure 5.2 shows a simplified drawing where the reference input (REFIN) is connected to the function generator and is also used as the trigger for the test equipment. One of the outputs is connected to the feedback path in the PLL and is the point where the jitter is measured.

In Figure 5.3, the scope trace from the TIA shows the input signal and the resulting signal through the device. By injecting a 300 mV noise signal into the power supply, the PLL outputs are affected in both amplitude and phase offset. The PLL input is a well-defined signal while the output varies considerably. The additional jitter measured on the output is a direct function of the noise and can be expressed with the following equation:

\[
\text{Additional Jitter (ns)} = \frac{\text{Noise (V)}}{\text{Slew Rate (V/ns)}}
\]
The Noise value in the equation is simply the amplitude of the noise present on the power supply pin of the part. The Slew Rate is the transition rate of the output driver independent of any noise. Hence, additional jitter is reduced as the rise time of the signal increases, given the same noise level.

The phase offset jitter added by the supply noise has a bimodal distribution as shown in the histogram. It is deterministic and bounded jitter because it is injected by a finite noise source with a known amplitude. The source can be traced to the filter of the VCO and the trip point variation of the input reference signal. The VCO is directly affected by the varying voltage inputs because of noise. This translates into jitter on the outputs. However, some PLL designs are more tolerant than others due to the use of better noise filtering circuits.

**Figure 5.3 Power Noise in PLL**

**PLL Noise Rejection**

Because some PLLs reject noise better than others, they react differently to the same level of noise on their power supplies. In Figures 5.4 and 5.5, two different PLLs are tested for their jitter component with the injection of noise. Each received 100 mV, 50 mV, and 25 mV noise signals sweeping through the frequency range. Both exhibited higher amounts of jitter as the noise amplitude increased. However, PLL2 in Figure 5.5 shows jitter of 100 ps at 200 kHz whereas PLL1 in Figure 5.4 exceeds 350 ps jitter at this same frequency. At the 25 mV noise level, PLL2 peaked at 50 ps, whereas PLL1 exhibited nearly 100 ps of jitter. Additionally, the frequency at which PLL1 had its peak value was at 200 kHz while PLL2 was at 5 MHz. Clearly, there is a difference in the performance of these two parts. Unfortunately, data to determine noise rejection is not typically listed on a standard datasheet and isn’t obvious to the designer.
Non-PLL Buffer

The propagation delay ($t_{pd}$) and output skew can be critical elements in a non-PLL based clock buffer. Using the same test methodology, the effect that power supply noise has on these parameters can be determined. Similarly, a waveform function generator drives the input to the buffer and is also the signal used to trigger the TIA. The output of the buffer is analyzed for increased delay.

Figure 5.6 Buffer Delay Test

The buffer driver is also affected by the supply voltage noise. The scope trace in Figure 5.7 shows a variation in the delay through the device as the noise sweeps through the frequency range. In typical CMOS buffers, the supply voltage affects the triggering of the part by changing the gate switch point on the input. More significantly, it changes the internal propagation time of the buffer. The noise causes variations in the voltage supply and hence varies the $t_{pd}$. The net effect is that the output signal will exhibit jitter because of the delay variation caused by the noise.
The increase in jitter, which is really input-to-output skew, affects all output pins of the buffer equally. Therefore, for simple buffers, the output-to-output skew remains constant. However, if the buffer contains complex features and the delay paths internally vary, the output-to-output skew could change.

**Filtering the Noise**

Noisy supply planes are a result of a variety of sources. Power supply switching circuits, digital components with busses, and even clocks can contribute to the noise. Many digital synchronous designs have buffers switching many outputs simultaneously causing current spikes large enough to generate power supply noise. To control noise in the power plane from affecting the clock circuitry, filters such as ferrite beads can be used.

The ferrite bead provides AC isolation between the power plane and the PLL (or buffer). By using beads, noise from the power plane is attenuated before reaching the PLL. Likewise, digital noise from a PLL or buffer with many simultaneous switching outputs is also attenuated before reaching the power plane.

The ferrite bead is chosen over a plain inductor because it provides better AC isolation. It does this by storing part of its created magnetic field in the ferrite core material. The ferrite bead also has a lower DC resistance than an inductor for an equivalent impedance value. This is important because the voltage drop is usually undesirable when attempting to meet the margins required for component power. The ferrite bead equivalent circuit is shown in Figure 5.8.
The ferrite bead is represented as an inductor and resistor in series. The resistance however, is dependent on the frequency of the signal and is therefore labeled as R(f). The ferrite bead appears resistive at high frequencies but is best analyzed with a plot showing impedance versus frequency. The characteristic behavior of a typical ferrite bead is shown in Figure 5.9.

When designing filters, it is preferred to minimize the DC resistance (R) so that the voltage drop is kept small. At the same time, the impedance (Z) needs to be as large as possible at the frequency that is to be filtered. In actuality, it was found through testing that choosing the lowest DC resistance is not always the best choice. And finally, the inductive (X) portion of the bead should also be small. A large X will distort the filter’s effectiveness, and there is a likelihood of a resonance condition occurring.

To evaluate how well noise is suppressed using ferrite beads with varying characteristics, a different test jig is used. Figure 5.10 shows a noise generator that will sweep a frequency range of 10 kHz to 1 MHz through a test ferrite bead. The bead is used with a bulk capacitor to complete the filter design. A 10 µF ceramic surface mount capacitor with a low effective series makes the filter more efficient. This becomes especially important at the lower frequencies when the bead impedance drops off.

Channel 1 (CH1) of the oscilloscope measures the amplitude of the voltage (V_OUT) through the filter under test. Channel 2 (CH2) measures the noise voltage (V_IN) injected into the filter. As the input signal sweeps through the frequency range, a comparison of the voltage ratio can be plotted. This test was performed on five different ferrite beads and the results are shown in Figure 5.11. The plot shows the voltage attenuation in dB that can be expressed as:

\[
\text{Attenuation (dB)} = 20 \times \log \frac{V_{\text{OUT}}}{V_{\text{IN}}}
\]
Figure 5.11 Ferrite Bead Results

One particular bead stands out among the others. Due to resonance near the 100 kHz frequency, the bead actually increased the noise ratio rather than attenuating it. Therefore, this one is not recommended for the clock filter design.

Knowing how the beads react in the noise test system, the impedance-frequency plots can be analyzed for correlation. Figures 5.12 and 5.13 show the plots for two different ferrite beads, one recommended and the other, not recommended.

Figure 5.12 Recommended Bead

Figure 5.13 Not Recommended Bead

It is not obvious from these manufacturer datasheet curves which bead is suitable for a filter as the variations are subtle. The bead that is “not recommended” for clock noise filtering has a characteristic of a long inductive tail extending to lower frequencies. It’s not that this is a bad bead design, it merely means that this particular one isn’t suited for this application.

Measured Impact on PLL

The bead test fixture provides the ability to verify that noise in the power supply is attenuated. The beads that pass this test can be placed into the jitter test fixture and validated with clock PLLs and buffers. Using the same ZDBs as before (Figures 5.4 and 5.5 show the results without the ferrite bead), a new set of tests are executed with the ferrite beads in place.
Four of the ferrite beads used in the bead test are subjected to 50 mV of noise and the PLL output is measured for jitter. The first observation is that the overall jitter has dropped considerably as seen in Figures 5.14 and 5.15. Note that these curves are the same scale as Figures 5.4 and 5.5. However, the attenuation of jitter varied slightly based on the specific bead in the circuit. PLL1 shows peak jitter ranging from 125 ps to less than 25 ps, depending on the bead. With no bead in place, the jitter peaked at 200 ps at the 50 mV level.

PLL2 also shows a marked improvement. With no bead filtering the noise, the jitter peaked at 150 ps. With any of the four ferrite beads in place, the jitter is extremely small and inconsequential because this particular PLL contains additional internal noise rejection capabilities.

**Figure 5.14 PLL1 Jitter with Filter**

**Figure 5.15 PLL2 Jitter with Filter**

**Non-Sinusoidal Noise**

The analysis so far has used a sine wave noise source. To properly validate the filtering capability of the beads, non-sinusoidal noise needs to be injected. By using noise generated by a switching power supply, noise with dominant peaks and many harmonics can be tested. The operating frequency of the switching power supply is 250 kHz and exhibits lower frequency modulation when under load. Within each switcher period, there is an initial step where the voltage changes abruptly and contains harmonics into the MHz region. This represents a real-world circuit board with power plane noise.

The spectrum analyzer plots in Figure 5.16 display the spectral purity of the PLL output at a fixed frequency of 66 MHz. The system board is under heavy load to maximize the switcher noise. The goal is to have the peak as sharply defined as possible, whereas a wider curve indicates jitter. The plots show the result of two different ferrite beads populated in the system where the one on the left has a sharp, well-defined peak and the plot on the right shows a broader, less-defined peak. Therefore, the left plot shows a recommended ferrite bead.
Ferrite Bead Recommendations

It is very difficult to determine which ferrite bead will perform the best based on the manufacturers specifications. By using data from a test environment, a bead that will filter the noise can be selected. Based on these tests, the following beads seem to perform well in filtering power supply noise for clock buffers. (However, they are by no means the only beads that will work):

1. Murata BLM21BD601SN1, BLM18BD421SN1, BLM18BD601SN1
2. Vishay-Dale ILB1206-300
3. TDK ACB2012M-300, ACB2012L-120
4. Steward HZ0805C202R
5. Fair-Rite 2512063017Y0

The beads that were tested and found not to work well had the following common characteristics and should be avoided for filter design:

1. Not characterized at lower frequencies such as 10 kHz
2. Very low DC resistance (< 0.1 ohm).
3. Long inductive “tail” extending to low frequencies.

Resistive Filters

Some PLL components have a separate V_DD pin (V_{DDA}) that powers the PLL core while the other V_DD pins power the output drivers, as shown in Figure 5.17. This usually gives more flexibility in the design of an external supply filter. For example, the PLL manufacturer datasheet may specify I_{DDA} (for V_{DDA} pin) drawing 15 mA max. With the current this low, it becomes possible to use a resistor instead of a ferrite bead. Choosing a value such as 8 ohms results in only a 0.12V drop and may be within operating specifications. For example, if using a 3.3V supply, the voltage drops to 3.18V at the V_{DDA} pin. PLL manufacturer datasheets may contain specific filter recommendations or layout guidelines that include a suggested filter. The advantage of using a resistor instead of a ferrite bead is that the lower frequencies are better filtered. This can be important for applications where
long-term jitter is important. Long-term jitter is the edge stability measured many clock cycles away, often 20 µs or more. Applications that may be affected by this include graphics Dot clock generation and serial link communication clocks.

Figure 5.17 Resistive Filter

Figure 5.18 Poor Method to View LVPECL Driver

Supply Noise In Differential Signaling

The previous analysis had focused on single ended interface signals. As operating frequencies have climbed, designers are using differential signaling. In this next example, a clock driver with an LVPECL interface is analyzed. The component is a non-PLL fanout buffer operating at 1 GHz. For these tests, the same modulation is applied to the VDD pins as was previously described. Since these are differential drivers, it is generally expected that it is more immune to supply noise.

Figure 5.18 shows an oscilloscope view of the output driver’s differential pair. The measurement shows the amplitude and offset change with supply noise levels. LVPECL is referenced to the voltage supply, so it is expected that the common mode voltage moves with the modulation on VDD. However, the largest concern is with the signal crossing point perturbation and not the common mode voltage. It becomes difficult to check with this view how much the crossing points have actually been affected in time; a high performance differential probe or a differential-capable time interval analyzer should be used.

Figure 5.19 shows the jitter contribution of the LVPECL clock driver operating at 1 GHz captured by a timing interval analyzer. In this series of tests, noise with amplitudes ranging from 25 mV to 300 mV was injected into the system. The graph shows that noise does affect a differential driver by moving its crossing point. At lower amplitude levels, the jitter was measured to be less than 5 ps. At the highest level of noise, 300 mV, the jitter reached nearly 6 ps. Although the amount of jitter is small, it can represent a significant portion of a period when clock speeds reach high frequencies.
Conclusion

Noise does affect the performance of clock buffers and PLLs. Having a clean, solid power source is fundamental in providing good clock waveforms. For the jitter critical designs, ferrite beads should be considered. Although it is difficult to select which bead will perform best, the analysis and recommendations presented in this chapter provides the necessary guideline information.
The Printed Circuit Board (PCB) provides two significant functions in an electrical design. It provides the mechanical locations for the components that reside on the board and it provides the connectivity between the components. These connections play an important role in providing a solid pathway to the power system and to ensure the highest degree in signal integrity. This chapter addresses the many aspects of PCB planes and traces for clocks.

The planes and traces can be separated into two categories. There are the pathways for the power source and there are traces required to carry the actual signals. In the first part of this chapter, PCB layout as it applies to the power planes is discussed. Later in the chapter, layout considerations for signal traces are addressed.

**Power and Ground Planes**

Power planes are large sheets of a conductive material that typically reside on entire PCB layers. They provide four primary functions to the circuit, as follows:

1. A low impedance path for power from its source to the components on the PCB.
2. A physical channel to vent and move heat from the components.
3. Electrostatic shielding between the electromagnetic fields of signal traces that run on both sides of the planes.
4. A sheet capacitance for the ground plane that exist on other layers of the PCB. This in turn provides additional AC bypassing within the power circuitry of the PCB.

The first and foremost functionality of a power plane is to reduce the resistance that causes a voltage drop between the component and the power source. The thickest power plane available will provide the best results. For example, using a two-ounce copper power plane instead of a one-ounce will cut in half any point-to-point power path resistance. It can be thought of as having two resistors (and inductors) in parallel. The increased plane thickness reduces both the DC resistance and the AC inductance drops. The drop in the DC resistance allows the power supply to reach the component cleanly, while the reduction in AC inductance provides a low impedance path for the signal return currents.
As a secondary benefit, the thicker plane also increases the ability to sink heat out of the component. The bond wires and lead frames are a major thermal path in non-heatsinked components.

The planes also aid in the reduction of Electro-magnetic Interference (EMI). They provide a lower impedance path across which the EMI develops and a larger faraday shield to short out these radiated fields.

The sheet capacitance that the power plane provides is proportional to its size, its distance from the ground plane, and the dielectric constant of the material between them. It has the benefit of providing bypass capacitance particularly at the high frequencies. While it is far from being sufficient to provide all of the bypassing needs of a high-speed logic design it should be utilized to its maximum. The capacitance of the planes can be calculated by the following equations:

\[
C = 0.0885 \frac{E \varepsilon_R [(N-1)A]}{t}
\]

Where:
- \( E \varepsilon_R \) = Relative dielectric constant
- \( N \) = Number of plates
- \( A \) = Area of one side of one plate in square centimeters
- \( t \) = Thickness (separation of plates) in centimeters

\[
C = 0.225 \frac{E \varepsilon_R [(N-1)A']}{t'}
\]

Where:
- \( E \varepsilon_R \) = Relative dielectric constant
- \( N \) = Number of plates
- \( A' \) = Area of one side of one plate in square inches
- \( t' \) = Thickness (separation of plates) in inches

For example, using a 10-inch by 10-inch FR-4 board with an \( E \varepsilon_R \) of 4.1 and a 0.005-inch separation between the power and ground plane, the capacitance is calculated as:

\[
C = 0.225(4.1)[(2-1)*100]/0.005 = 18,450 \text{ pF}
\]

This is equivalent to 184 pF per square inch. Table 6-1 shows the dielectric constants for several common materials used in PCB design today. It is always advisable to consult your fabricator for the precise \( E \varepsilon_R \) value as different epoxies are used when constructing a PCB. Dielectric constants of PCB also change with frequency as shown in the table.
Table 6-1 Dielectric Constants

<table>
<thead>
<tr>
<th>Material</th>
<th>$\varepsilon_r @ 1$ MHz</th>
<th>$\varepsilon_r @ 300$ MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>FR-4, Tetra Functional</td>
<td>4.2 - 4.6</td>
<td>4.0 - 4.3</td>
</tr>
<tr>
<td>FR-4, Hi-Grade Multifunctional</td>
<td>4.2 - 4.6</td>
<td>4.1 - 4.4</td>
</tr>
<tr>
<td>Polyimide</td>
<td>4.2 - 4.6</td>
<td>4.1 - 4.3</td>
</tr>
<tr>
<td>GETEK</td>
<td>3.9 - 4.1</td>
<td>3.9 - 4.0</td>
</tr>
<tr>
<td>BT</td>
<td>3.6 - 4.1</td>
<td>3.55 - 4.0</td>
</tr>
<tr>
<td>CE</td>
<td>3.6 - 4.0</td>
<td>3.6 - 4.0</td>
</tr>
</tbody>
</table>

**Ground Island**

With clock generators and buffers, it is recommended to create a ground island directly beneath and on the same layer as the device. Connections are then made from that island using short traces to the ground pins, as shown in Figure 6.1. These traces should be as wide as possible.

The ground island that is then present should be stapled with vias to the inner ground plane or planes. The general rule is one via per ground pin as close to where the components ground trace meets the island. Adding more vias in parallel reduces the effective impedance. Beyond two vias per ground pin becomes increasingly less effective. Also, the more vias that are added, the ground plane impedance where they attach can be impacted. The PCB fabrication process usually has a guideline on the via requirements.

![Figure 6.1 Ground Island](image-url)
On smaller buffers, a large ground island may not be practical or necessary. In this case, a short, wide trace may be used to supply the power to the $V_{DD}$ pin and the $V_{SS}$ pin connected to the ground plane with a via. Figure 6.2 shows a small clock buffer with these types of connections. The $V_{DD}$ trace is filtered with a ferrite bead prior to the via connection to the power plane. The ferrite bead and its associated capacitors should be as close as possible to the clock buffer.

![Figure 6.2 Small Buffer Layout](image)

**Vias**

Vias are commonly used to connect the power plane to the power traces that ultimately attach to the power pins of the components. They can also have an impact on the power signal quality.

First, vias produce a higher resistance than a copper trace. As noted earlier, the higher the resistance, the greater the voltage drop. This is due to the material typically used in the fabrication of the via, which is plated granular copper. The resistance of a via changes based on the thickness of the copper that is plated in the via hole. Therefore, larger vias have a lower DC resistance and hence develop less of a DC voltage drop for any given current.

Vias also add inductance to the power trace. This inductance causes high frequency noise that is present on the power plane to stay on the plane (which is good) but it also isolates the capacitance effect of the power plane from the components on the other ends of the vias. Filling vias with solder, using heavy plating, enlarging their size and using multiple vias per power connection are the preferred methods of lowering both the resistive and inductive parasitic effects they have on power connections.

**Power Traces**

Similar to vias, a trace also has some amount of resistance, capacitance and inductance. The resistance must be kept to a minimum to avoid voltage drop on the trace. For the power plane to reach the power trace and ultimately the component, a via is also required.

If the connections to the device are made in the correct sequence, the resistance and
Inductance of the trace and the via will isolate the component’s noise from passing into the power plane. This effect increases as the frequency of the noise rises. To achieve this isolation, the power trace must pass from the via, to the bypass capacitors pad and then to the component, as shown in Figure 6.3. This order is important to create an island of protected trace between the bypass capacitor and the component.

It is important that the trace between the component and bypass capacitor be as short as possible. The goal is to keep the inductance between the devices to a minimum. A short wide trace will produce better results than a long narrow trace.

![Figure 6.3 Power Trace to Component](image)

**Signal Return Paths**

The power planes play a key role in the return currents of high-speed digital clocks. At very slow speeds, the return current follows the path of least resistance. But at higher speeds, the return currents follow the path of least inductance. This can be either on the power plane or the ground plane directly below the signal trace. Normally, the return path is on the ground plane in standard PCB designs. When the return current reaches the driving component, the bypass capacitors provide the bridge to the proper voltage plane, as shown in Figure 6.4.

To maintain good signal integrity and to minimize crosstalk, a clean, unobstructed return path needs to be provided. For example, if a return signal encounters a 5-ohm trace within the ground plane, the integrity of the normal signal on the 50 ohm trace can be compromised.

There are a few simple rules to follow for planes. First, it should be as continuous as possible. Placing excessive clearances around holes that pass through it may make fabrication easier, but it can cause the return current to travel through a non-optimal path. The effect of the clearance holes should be analyzed. Second, cutouts in the ground plane must be avoided. Similar to the clearance holes for the vias, a ground plane cutout will force the return path current around the cutout if the clock signal trace crosses above the cutout. This will increase the inductance of the path and will decrease the rise time of the clock on the signal trace. It will also increase the potential for crosstalk.
Signals
The second area with which we need to be concerned is the routing of clock signals. The goal, obviously, is to provide a pathway for the clock that has no noise, has uniform impedance, and does not produce any EMI. The next section discusses the effects that board layout has on these attributes.

Crosstalk
Crosstalk can exist between traces on a PCB. If this occurs on a clock signal with sufficiently strong amplitude, a false trigger can occur. Therefore, crosstalk needs to be minimized in the design.

As a signal travels through a trace, it creates a magnetic field. It also reacts to other magnetic fields within its path. Therefore, the trace acts as both a field generator and an antenna. The voltages that external fields cause are proportional to the strength of the external fields and the length of trace that is exposed to the field. Often, the trace that causes the crosstalk is termed the Source or Aggressor and the trace that is affected by the first is called the Victim.

Crosstalk between traces is a function of both mutual inductance and mutual capacitance. A model of the inductance and capacitance between traces is shown in Figure 6.5. Its magnitude is proportional to the distance from the source trace, the speed of the signal edge rate, and the impedance of the victim trace. In digital systems, crosstalk caused by mutual inductance is typically equal to or larger than the crosstalk associated with mutual capacitance.
To illustrate the effects of spacing, the mutual inductance $L_M$ can be calculated with the following equation:

$$L_M = \frac{L}{1 + (s/h)^2}$$

Where:  
$\hspace{0.5cm}$
$L =$ Inductance of the wire
$s =$ Separation between the wires
$h =$ Height above the plane

This shows that moving the traces away from each other (value $S$) or by moving the traces closer (value $H$) to the plane, the mutual inductance is reduced by the square of the change. And since crosstalk is proportional to the mutual inductance, the magnitude of the crosstalk is also reduced. Figure 6.6 illustrates this model.

The mutual capacitance $C_M$ injects current $I_M$ into the victim trace and can be calculated as:

$$I_M = C_M \frac{dV_s}{dt}$$

Where:  
$\hspace{0.5cm}$
$V_s =$ is the source voltage.

Figure 6.7 illustrates the effects of trace impedance on crosstalk coupling. The higher the impedance of the victim trace, the more susceptible it is to noise from crosstalk. Figure 6.8 shows a similar effect but with various widths of the trace line. Wider traces produce less crosstalk coupling. Therefore, minimum coupling is created with maximum spacing, maximum trace widths, and minimum impedance.
To minimize the effects of magnetic field coupling, three basic rules should be followed. First, separate the traces with more distance. The effect that is seen is directly proportional to the square of the distance of the elements and therefore doubling the distance will reduce the coupling by a factor of four.

The second method of decreasing the coupling is to shield the target trace. This can be done by either routing it on another layer or placing protective or guard traces between the two. In this way, the magnetic lines will have a portion of their energy developed in the protective traces, usually at ground potential, and the field that cuts across the trace being shielded is less.

A third way to reduce the impact of crosstalk is to use differential signaling. This type of signal rejects noise from crosstalk if both the true and complement signal are equally effected. This is commonly called common mode noise rejection. However, if the crosstalk affects one trace of the differential pair and not the other, the noise coupling can be a factor.

**Guard Traces**

In some instances, victim traces need to be shielded or guarded from source traces to prevent crosstalk, as shown in Figure 6.9. This is not often the case as increasing the signal separation will solve the crosstalk issues. But in stubborn cases or when the absolutely lowest noise environment must be obtained, guard traces should be considered. In their simplest form, a guard trace consists of a single grounded trace between a source signal trace and the victim clock trace. The guard trace should be grounded on both ends.

While not prevalent in digital logic designs, these structures have found much use in the area if guarding sensitive analog signals. However, with reducing signaling levels due to higher frequency clocks, the effective crosstalk noise becomes an increasing concern. As the signaling level decreases, the percentage that this noise represents (signal to noise ratio) becomes worse. Noise whose magnitude is 100 mV riding in a 3.3-volt clock is only...
3%, or a 33:1 S/N ratio. If this 100 mV signal exists in a differential clock that only swings 1.0 volt, then it becomes 10% noise or a 10:1 S/N ratio problem. This can easily cause an unwanted clock. Therefore, care must be taken when designing with smaller signal swings to avoid crosstalk.

![Figure 6.9 Guard Traces](image)

## Layers

To keep noise at a minimum and to achieve the best in signal integrity, control of the PCB layers is required.

As was mentioned before, spacing the power and ground planes close together gains additional bypassing capacitance. Of more importance is to create a reference plane that provides constant trace impedance. The reference plane is usually the ground plane. The thickness of the substrate between controlled impedance traces and the ground plane must be selected to be both manufacturable and able to keep the desired characteristic trace impedance within acceptable limits. As the spacing of layers becomes less, the ability to hold tight, repeatable spacing values decreases. A 1-mil variation on a 10-mil spacing is 10% and on a 20-mil spacing is only 5%. You should always consult with your fabricator on the tolerances.

When power planes are correctly bypassed, they are at the same AC potential as the ground planes. Therefore, they can be used as a reference for controlled impedance traces. Using both power and ground planes, however, is unwise as the transition between the two has a high potential of causing noticeable impedance discontinuities in the traces impedance and this is a source of unwanted reflections.

## Vias in Traces

When routing a clock trace, the signal should remain on the same signal plane. Using a via to route to other planes diminishes the signal quality of the clock. Vias should therefore be avoided.
Vias exhibit resistance, capacitive reactance and inductive reactance. Because they are composed of granular copper they exhibit more resistance than an equal length of PCB trace. The largest signal degradation that vias cause to the clocks that pass through them is due to their inductance and it comes from the granular copper that is plated into the via holes. Inductance is a function of frequency and is greater with faster speed clocks. Adding a via has a similar effect as adding a resistor in the path whose value increases with a higher rate clock.

If vias cannot be avoided and have to be used in a clock trace, the following guidelines should be considered:

1. Make vias as large as possible (more area equals less inductance and resistance).
2. Plate external layers with the maximum thickness of copper during fabrication.
3. Plating or filling the via holes solid.

**EMI Generation**

PCB traces act as small antennas to the clock signals on the traces. In practical terms, the goal is to provide a “poor” antenna to prevent unwanted EMI.

There are a few trace configurations that should be avoided. If a trace, otherwise an antenna, is created whose length is 1/4, 1/2, 1 or any other integer multiple of one times the electrical length of the fundamental frequency of the clock, then a close to ideal radiating source will be created. This trace will have along its length a standing voltage wave of maximum (or near maximum) voltage magnitude. This produces the optimal configuration to cause the maximum electromagnetic coupling to the surrounding area. If the trace is on the surface of a PCB, the area on three sides is air. It can also couple with other radiating traces and hence raise the level of EMI. (See Chapter 9 for more details on EMI.)

**Differential Clock Traces**

Differential clocks require some special attention to achieve the most benefit from its inherent properties. This type of clock signal is often used to eliminate the effects of noise in the system, but this effect can be voided with poor signal routing. When using differential clocks, the following rules should be followed:

1. The traces (true and complement) must be physically equal in length. To construct them otherwise will cause a phase lead or lag condition. This will degrade the duty cycle or clock-to-clock period and cause a bimodal clocking condition. In this situation there are two distinct half cycle periods created in the resulting clock.
2. Treat the signals alike. If you must place a via to transition between layers, place the vias electrically at the same point (within a minimum of 0.005 inch occurrence in each trace).
3. Keep the traces constantly spaced. It is important that they maintain this spacing as they round corners.

4. Keep the two traces (true and complement) close to each other. Because they are differential does not mean that they are immune to external magnetic noise fields (crosstalk). If one trace is closer to the source than the other, it will have a highest voltage induced in it. Keeping the trace pairs as close as possible will minimize this effect.

**Conclusion**

PCB layout is an important aspect of design especially when implementing high-speed digital clocks. The layout can affect the signal response, termination, EMI and a host of other factors. Care needs to be taken in both the layout of the power planes as well as the signal layout. It is important to remember that traces, vias, and planes have different characteristics when high-speed signals are present. They no longer act the same when a DC signal is applied. With the higher frequencies, these materials act as inductors and can attenuate the signal. This can severely impact the operation of the design. Layout can also affect other factors such as crosstalk and EMI. Excess noise can be coupled into signals on the board or into the external environment causing unwanted inference. By understanding how these signals operate and influence each other, and by following a few ground rules, a reliable, quiet design can be achieved.
A primary goal of any timing generation and distribution circuit is to have clean, reliable clock edges at the destination of the trace. Without proper termination, impedance mismatches will occur and reflections will be evident. The resulting reflected pulses may be large enough to falsely trigger a device input. This can lead to disastrous consequences especially when the signal is a clock.

There are several methods that can be used to terminate clock traces, each has a variety of tradeoffs. There are choices that impact signal integrity, power, and board area. We will discuss each alternative and address its tradeoffs. Before analyzing the specifics of terminations, let’s first look at what happens when an electrical wave propagates down a transmission line with no termination. Figure 7.1 shows an ideal line with a clock source (driver) and some device (load) at the end. This is an ideal model because no parasitic elements are assumed.

\[
\begin{array}{c}
S \\
R_D \\
Z_0 \\
R \\
A
\end{array}
\]

**Figure 7.1 Ideal Transmission Line**

Where:  
- \(S\) = Signal into the transmission line  
- \(R_D\) = Impedance exhibited by the driver source  
- \(Z_0\) = Trace impedance  
- \(A\) = Signal accepted into the load  
- \(R\) = Signal reflected from the load
When the clock driver output is a signal whose voltage is $V$, then signal strength $S$ is equal to: $V \times Z_O/(Z_O + R_D)$. This wave will continue to the load end of the transmission line where part of it will be accepted and the remaining part will be reflected. The accepted portion into the Load can be calculated as $A_L = 2 \times Z_L/(Z_L + Z_O)$ where $Z_L$ is the load impedance and the portion reflected back toward the beginning of the line is determined by $R_L = (Z_L - Z_O)/(Z_L + Z_O)$. The wave of magnitude $R_L$ will continue propagating until it encounters the next impedance change, which is at the source. Again, there will be an accepted and a reflected function. The reflected function from the source is $R_S = (R_D - Z_O)/(R_D + Z_O)$. And this wave will propagate towards the load end of the trace.

Notice that a wave will continually bounce back and forth on the trace until it eventually dampens out. If the reflected wave is substantial and is near the threshold value of the load component, a false clock can be realized. Also note that the equations allow the signal strength to be either positive or negative and therefore satisfy the situations where a signal can be over-damped as well as under-damped.

**Low Impedance Driver**

Let’s look at a situation where the source impedance is less than the trace impedance. This is typical of a standard or high drive clock buffer into a typical trace. We will assume that the output impedance of our clock driver is $17\Omega$ and the trace impedance is $50\Omega$. We will also assume that the input impedance of our load $Z_L$ is very high and representative of typical loads. If the output of the clock drives a voltage of $V$, the signal into the transmission line is:

$$S = 50\Omega/(50\Omega + 17\Omega) \times V = \frac{3}{4} V$$

This equation tells us that we have $\frac{3}{4}$ $V$ propagating down our trace. When this signal reaches the end of the trace, it will have two components: the accepted signal ($A_L$) and the reflected signal ($R_L$). The accepted signal is the voltage seen by the load device and the reflected signal is sent back to the source.

$$A_L = 2 \times 1M\Omega/(1M\Omega + 50) \times \frac{3}{4} V = 1\frac{1}{2} V$$

$$R_L = (1M\Omega - 50\Omega)/(1M\Omega + 50\Omega) \times \frac{3}{4} V = \frac{3}{4} V$$

In this situation, the device at the end of the trace actually sees $1\frac{1}{2}$ $V$ at its input and $\frac{3}{4}$ $V$ is transmitted back to the source. This is the case of a clock signal that exhibits overshoot. Figure 7.2 illustrates the effect of this impedance mismatch. The Round Trip Time is the propagation delay of the trace times two, thus equally the time it takes for the signal to travel down the trace and back.
The reflected wave ($\frac{3}{4} V$) will propagate back to the source until it encounters the 17 ohm mismatch. It then will have the accepted and reflection components, as we described for the end of the trace. The reflected wave can be calculated as:

$$R_S = \frac{(17 \Omega - 50 \Omega)}{(17 \Omega + 50 \Omega)} \times \frac{3}{4} V$$

$$= -\frac{1}{2} \times \frac{3}{4} V$$

$$= -0.375 V$$

Notice two key factors in the reflected wave: its magnitude is still of considerable value and the signal is negative, as shown in Figure 7.3. This negative signal will propagate back towards the load end of the trace and again exhibit accepted and reflected properties. The load will see a signal alternating above and below the desired voltage $V$ while approaching $V$. The width of the pulses is equal to two times the trace delay.

**High Impedance Driver**

Now let’s look at the opposite case where the driver has a higher value than the trace impedance. This depicts a situation where the clock driver may be very weak and may have a slow rise time.

We will assume that the output impedance of our clock driver is 150 ohms and the trace impedance is 50 ohms. And again, we will also assume that the input impedance of our
load is sufficiently high. If the output of the clock drives a voltage of \( V \), then the signal into the transmission line is:

\[
S = \frac{50\Omega}{50\Omega + 150\Omega} \times V = \frac{1}{4} V
\]

This indicates that \( \frac{1}{4} V \) propagates down the transmission line. When this signal reaches the end of the trace, it will again have two components: the accepted signal and the reflected signal. As before, the accepted signal is the voltage seen by the load device and the reflected wave is sent back to the source.

\[
A_L = 2 \times \frac{1 \text{ M} \Omega}{1 \text{ M} \Omega + 150} \times \frac{1}{4} V = \frac{1}{2} V
\]

\[
R_L = \frac{(1 \text{ M} \Omega - 150\Omega)}{(1 \text{ M} \Omega + 150\Omega)} \times \frac{1}{4} V = \frac{1}{4} V
\]

In this situation, the device at the end of the trace only sees \( \frac{1}{2} V \) at its input and \( \frac{1}{4} V \) is transmitted back to the source. This is a case of undershoot on the clock signal. Figure 7.3 illustrates this condition.

The reflected wave (\( \frac{1}{4} V \)) will propagate back to the source until it encounters the 150 ohm mismatch. It then will have the accepted and reflected components as we described for the load end of the trace. The reflected wave can be calculated as:

\[
R_S = \frac{(150 \text{ ohms} - 50 \text{ ohms})}{(150 \text{ ohms} + 50 \text{ ohms})} \times \frac{1}{4} V = \frac{1}{2} \times \frac{1}{4} V = 0.125 V
\]

Figure 7.4 Overdamped Clock Trace

This positive signal will propagate back toward the end of the trace and again exhibits accepted and reflected properties. The effect it ultimately has is a stair-stepping signal approaching the value \( V \), as shown in Figure 7.4. The width of each stair is equal to twice the signal trace delay.

Notice that in both the low impedance driver and the high impedance driver cases, the signal at the load end is quite different from the signal at the source end. Since the load is the device that will accept and react to the clock signal, we want to optimize the signal at this point. Ideally, we want to have a single point-to-point connection for each clock on our board to achieve the best signal quality. This can generally be accomplished using clock fanout buffers. However, there are times when we may want to drive multiple loads from a single driver.
When to Terminate

Before we address how to terminate, let’s first answer the question of when to terminate. For clock signals, it is best practice to always terminate the trace. This will prevent any unwanted pulses from occurring and prevent unwanted false triggers. If circumstances prevent proper termination then the trace delay must be short as compared to the edge rate of the clock. A general guideline is that the trace delay must be at least six times faster than the rise time of the clock. Any trace longer than that ratio must be terminated or reflections will become a factor. All clock traces should be simulated either with IBIS or Spice model simulation to ensure proper termination.

Source Termination

By far, Source Termination is the most popular form of termination for the typical clock circuit. Source termination, or more commonly known as Series Termination, uses a resistor placed in series with the trace as close to the source as possible. The intent of the resistor is to match the output impedance of the clock driver to the impedance of the trace. This allows the reflected wave to be absorbed when returning.

There are several key advantages to Series Termination. First, it is simple to implement and requires little board area. Second, it does not create a constant DC current flow and therefore doesn’t consume excessive power. And third, it is relatively easy to calculate the necessary resistor values.

However, there are a few drawbacks for Series Termination. Series Termination can only be used if the load devices are at the end of the trace. We will see why when we discuss the wave propagation. The other drawback is that it may be difficult to achieve a perfect termination since we are trying to match the output impedance of a clock driver, and the output driver impedance changes throughout its I-V curve, therefore the impedance value is variable.

![Figure 7.5 Series Termination](image)

Since we want to match the impedance of the trace to the impedance of the driver, we add the resistor $R_S$ to the trace, as shown in Figure 7.5. We want to set the value such that the output impedance of the driver ($R_D$) plus the Series resistor ($R_S$) is equal to the trace impedance ($Z_0$). This can be expressed in the following equation:

$$R_D + R_S = Z_0$$
Clock drivers have some amount of impedance, and in general, the lower the output impedance, the faster the signal will transition. To determine $R_S$ we need to know $R_D$. Some component datasheets will specify the value to be used for $R_D$ for a particular driver. It can also be derived from the IBIS model of the device. Many IBIS model simulators extract the value for easy access or you can browse the model itself to get the value.

When the output buffer drives the clock signal, the wave will propagate along the trace. Since there is an impedance mismatch at the point of $R_S$ and $Z_O$, we have both reflected and accepted properties. Since $R_S$ was chosen such that $R_S$ plus $R_D$ is equal to $Z_O$, the transmitted voltage will be half that of the output driver—a voltage divider. The half wave will propagate down the trace to the end where it will experience its reflected and accepted functions. The amount that will be accepted and reflected is:

$$A_L = 2 \times \frac{1}{1 \, \Omega + Z_O} \times \frac{1}{2} \, V = V$$

$$R_L = \frac{1}{1 \, \Omega - Z_O} \times \frac{1}{2} \, V = \frac{1}{2} \, V$$

The input of the load device will have the full voltage $V$ and the $\frac{1}{2} \, V$ is propagated back to $R_S$.

Notice we have several different impedance changes in our model. There is the output driver impedance, the series resistor, and the trace impedance. Reflections and acceptances will occur at every impedance boundary. It is therefore imperative that $R_S$ is sufficiently close to the output driver or a complex set of reflecting waves will occur. The distance that is tolerable is dependent on the rise time of the signal. The faster the rise time, the closer the resistor needs to be to the source. In order to have the reflections become a non-factor, the trace propagation time between the source and $R_S$ should be one-tenth of the signal rise time. For example, if we have a clock signal that has a 2 ns rise time, the flight time between the driver and the resistor should be no greater than 200 ps. If the trace characteristics are such that the propagation delay is 175 ps/inch, the maximum distance $R_S$ should be from the driver is 1.14 inches.

---

**Figure 7.6 Series Terminated Trace**

Single Clock Driver (19.8 Output Impedance), 31.9-ohm Series Resistor, 6-inch Long Trace, 51.7 ohm Stripline (Effective), Single Load

Vertical: 1V/div
Offset: -2.0V
Horizontal: 1 ns/div
Delay: 0.00 ns
In a Series Terminated trace, the receiver “sees” driver impedance equal to the trace impedance $Z_0$. Knowing this relationship, we can determine the rise time of the signal at the receiver based on the RC time constant.

$$t_R = 2.2 \frac{Z_0}{C_L}$$

where:

- $t_R$ = Rise time of the signal from 10% to 90%
- $Z_0$ = Trace impedance
- $C_L$ = Load capacitance

This equation gives us the fastest rise time possible at the receiver with a series terminated trace. Further, we can add the rise time of the driver ($t_D$) to provide us with the actual rise time at the receiver ($t$):

$$t = (t_D^2 + t_R^2)^{1/2}$$

The power dissipated by series termination is small compared to other types of termination. However, it isn’t easily derived. To calculate the power dissipated by $R_T$, our series termination resistor, we need to determine its voltage drop. The difficulties are due to the fact that the voltage across the resistor is changing based on several factors. The first reason is obvious: $V_{OH}$ and $V_{OL}$ are changing. However, since the series termination resistor acts as a voltage divider when the signal first propagates down the line, and remains in that condition until the wave returns, we have a dependency on the length of the trace. Once the reflected pulse returns, we no longer have the same voltage drop, hence a changed power dissipation. In conjunction with the length of the trace, clock frequency is also of importance. For every clock cycle, a voltage drop is presented across the resistor. We can then express the power dissipated as:

$$P_T = f * 2^T * \frac{(V_{OH} - V_{OL})^2}{2R_T}$$

Where:

- $P_T$ = Power dissipation in the termination resistor
- $f$ = Clock frequency
- $T$ = Trace delay time
- $R_T$ = Value of the termination resistor

**End Termination**

There are several types of end termination: the Split Termination (also known as Thevenin equivalent), Parallel Termination, AC Termination, and several others. Most of these methods use resistors to match the trace impedance at the end of the trace instead of the source. We will focus on these three and how they apply to clock signals.

End termination differs from Series termination in that the trace is terminated at the end of the trace. This results in little or no reflections returning back to the source. This allows for populating additional devices along the trace (which may cause signal reflections and introduce other termination issues).
Thevenin Equivalent

Thevenin equivalent termination uses a pull-up/pull-down pair of resistors at the end of the trace (see Figure 7.7). The parallel combination of the two resistors is set equal to the impedance of the trace. However, care must be taken not to exceed the high and low drive currents ($I_{OH}$, $I_{OL}$) of the driver. In Figure 7.8, a simulation of a thevenin terminated trace is shown. Notice that the signal does not reach the voltage rails due to the termination.

![Figure 7.7 Thevenin Equivalent Termination](image1)

![Figure 7.8 Thevenin Terminated Trace](image2)

To prevent any reflections, the junction of $R_1$ and $R_2$ must be equal to the value of the trace impedance. The values of $R_1$ and $R_2$ can be expressed with the following formula:

$$\frac{1}{R_1} + \frac{1}{R_2} = \frac{1}{Z_o}$$

If the output driver is symmetrical and provides equal drive for high as well as low, then $R_1 = R_2$. If they are not equal, then the values need to be adjusted to prevent overloading the driver. We can express the $I_{OH}$ and $I_{OL}$ with the following two equations:

$$I_{OH} < (V_{CC} - V_{OH}) \frac{1}{R_1} - (V_{OH}) \frac{1}{R_2}$$

$$I_{OL} > (V_{CC} - V_{OL}) \frac{1}{R_1} - (V_{OL}) \frac{1}{R_2}$$

Notice that the expression for $I_{OH}$ typically results in negative numbers on either side of the inequality since the driver is sourcing the current.

To determine the rise time at the receiver in a thevenin terminated trace, we use an $R_C$ time constant as we did with series termination. However, there are a few differences. Instead of $Z_o$, the receiver “sees” the trace impedance $Z_o$ in parallel with the thevenin equivalent of $Z_o$, hence the result is $Z_o/2$ ohms. Applying this to our $R_C$ equation we notice that the rise time of the signal can be twice as fast as the series terminated trace:

$$t_R = 2.2 \frac{Z_o/2}{C} = 1.1Z_o C_R$$

Further, we can add the rise time of the driver ($t_D$) to provide us with the actual rise time at the receiver ($t$):

$$t = (t_D^2 + t_R^2)^{1/2}$$
The thevenin equivalent termination resistors are always consuming power. For clock signals, we can safely assume a fifty percent duty cycle, hence, half the time the output is \( V_{OH} \) and the other half it's \( V_{OL} \). We can then determine the power dissipated by each resistor.

\[
P(R_1) = \frac{(V_{CC} - V_{OH})^2 + (V_{CC} - V_{OL})^2}{2*R_1}
\]

\[
P(R_2) = \frac{(V_{OH})^2 + (V_{OL})^2}{2*R_2}
\]

**Parallel End Termination**

Parallel end termination uses only a pull-down resistor at the end of the trace. The value of this resistor is equal to the trace impedance \( Z_O \) to terminate the signal. The advantage this method has over the Thevenin equivalent circuit is that the constant consumption of power is eliminated. However, care again must be taken to ensure the output driver is not overloaded.

![Parallel End Terminator](image)

**Figure 7.9 Parallel End Terminator**

In this case, \( I_{OH} \) (which is negative) must be able to supply the necessary current through the terminating resistor based on the following equation:

\[
I_{OH} < -\frac{V_{OH}}{R_1}
\]

The rise time at the receiver for the parallel termination is the same as the Thevenin equivalent termination method:

\[
tr = 2.2 \left( \frac{Z_O}{2} \right) Cr = 1.1Z_O Cr
\]

Further, we can add the rise time of the driver \( (t_D) \) to provide us with the actual rise time at the receiver \( (t) \):

\[
t = (t_D^2 + tr^2)^{1/2}
\]

The power dissipated for the pulldown circuit is only one component of the Thevenin Equivalent power calculation. Since we only have one resistor to ground, the power is equal to the following equation:

\[
P(R_1) = \frac{(V_{OH})^2 + (V_{OL})^2}{2*R_1}
\]
AC Termination

AC termination is a variation of parallel termination and addresses the constant dissipation of power. AC termination adds a capacitor between ground and the termination resistor to prevent the constant flow of DC current, as shown in Figure 7.10. This is accomplished by charging the capacitor to a voltage halfway between the driver’s high and low voltage. The voltage across the termination resistor is cut in half and thus saves power. For the capacitor to charge correctly, the signal must be DC balanced meaning the signal must be transitioning equal periods of high states as well as low states. This is easily satisfied for our typical clock signal.

The value of R is the same as in the parallel termination method and is set to equal Z₀. In order to select C, we first must know the period of the clock present on the trace. We need to ensure that the capacitor doesn’t significantly discharge during the high or low portions of our clock cycle. We therefore set C so that the RC time constant is much greater than that of the period of the clock. Although this method sounds simple enough, it does not work well with long traces. It is also difficult to simulate with many IBIS model simulators since several clock cycles need to occur to charge the capacitor. For this reason, series termination or one of the other end termination methods may be a better choice.

AC Termination Part II

Some designers will use a variation of the AC termination described above. Instead of using the capacitor to provide a biasing voltage for the period, the capacitor is used to block the flow of DC current. During the transition of a clock signal, the capacitor provides a path to ground that allows the termination resistor to operate properly. However, when the signal reaches the high level, there will be no current flow through the termination resistor to ground because of the capacitor. This prevents the dissipation of power during the high (and low) states of the clock. Note that the driver must be able to supply the needed output drive current during the signal transition through the resistor to ground. The value of the capacitor is small and typically 10 pF. However, the signal waveform should be measured to ensure proper operation.

Figure 7.10 AC Termination

Figure 7.11 Multiple Loaded Trace
**Multiple Loads**

We’ve covered the point-to-point single ended clock cases with the most common termination techniques, but what about multiple destinations? Ideally, clock signals should have a single source for every load device, but there are times when two or more loads may be required. We will look at several of the common scenarios and address the areas of concern.

Suppose we have two devices that we want to drive and both are very close physically to each other. There is only one clock driver output available and no option to add a larger buffer or second component. We also assume the trace is sufficiently long and therefore acts as a transmission line. We then have a single driver and a single trace with multiple loads as show in Figure 7.11.

If we use the Series termination method, \( R_S \) is calculated as the difference between the output impedance of the driver and the effective trace impedance. To prevent unwanted reflections from occurring due to the multiple loads, we must first analyze the stub length (which is the short trace segment between two separate destination loads) and signal rise time.

If the distance between the two loads is short, the two loads will act as one capacitive lumped load. However, if they are not, we will have an un-terminated stub and our signal will reflect at each point and result in multiple accepted and transmitted waves — basically, a poorly terminated trace. So the key question is how close do the loads need to be? That answer depends solely on the rise time of the driving signal. A good rule of thumb is: the length delay of the trace stub must be six times faster than the rise time of the signal. This guideline allows the transitioning signal to propagate up and down the stub rapidly enough to appear as a single pulse.

The rise time at this load will be different than a single load. If you recall from the earlier section with the single load, the rise time can be expressed as:

\[
\tau_R = 2.2 \cdot Z \cdot C
\]

Our capacitance has increased because there are two capacitive loads in parallel thus doubling the value of \( C \). There is also additional trace capacitance because of the small stub connecting the two loads. Multiple loads at the end of the trace will work provided the stub is short and the rise time is still suitable for your application, as shown in Figure 7.12. However, it is always recommended to simulate your design.
Another common way to implement series termination driving two loads is to use two termination resistors and two separate traces, as shown in Figure 7.13.

This allows the loads to be physically far apart from each other with each segment terminated. The calculation for determining $R_s$ is a bit different from the single series termination trace. In this case, we have the source driving a series resistor and trace impedance in parallel with another series resistor and trace. We can express this as:

$$R_D + R_{S1} || R_{S2} = Z_0 || Z_0$$

Notice when we add the second load, the trace impedance is halved since it is viewed in parallel. This results in the values of $R_s$ being less than the single trace termination. Note that if the parallel trace impedance is less than the output driver impedance, the trace cannot be properly terminated. Let’s take a moment to review the wave propagation in the dual trace scenario. The driver will launch a signal with amplitude $V$ into the trace. At the load side of the series termination resistors, $\frac{1}{2}V$ will begin to propagate down the trace.

When the signal encounters the end, it will reflect back (the load will see $V$) propagating back towards the series termination resistor. Once encountered, some signal portion will be reflected and some will propagate beyond the resistor. To minimize continuing reflections, the overall trace lengths need to be equal in length. Also, the termination
resistors must be close to the source as given previously by the guideline in series terminated traces. If the trace lengths to the resistors have a flight time slower than one-tenth the rise time of the signal, this segment will act as a transmission line by itself. This will cause a reflection when the propagating signal encounters the resistors. This scenario is known as a bifurcated trace and should be avoided.

Care also needs to be exercised when selecting the clock driver. As each trace is added in parallel, the effective impedance is lowered. If this value is smaller than the output impedance of the clock driver, no value of $R_S$ will be able to properly terminate the trace. Therefore, select a driver with output impedance that will allow for termination. As in the case before, it is advisable to simulate the design with an IBIS or Spice simulator to verify the trace will perform as expected.

**Differential Signals**

Differential signaling is gaining popularity in timing circuits as a result of ever increasing clock rates. Differential signaling becomes a real benefit when distributing a clock along a backplane from one card to another due to its ability to tolerate ground voltage shifts. However, there are several different types of differential drivers and to complicate the situation, many published methods for terminating.

Differential signaling propagates two signals along the board traces as shown in Figure 7.14. The second signal is equal in amplitude and opposite to the polarity of the first and the receiving input switches based on their difference. Two major benefits as a result. First, if noise occurs within the system and it affects both signals, it is canceled out as a result of the difference of the signals at the receiver input. This noise is known as common mode noise in that it is common to both signals. The second major benefit is that the ground reference between the two devices can vary without causing a false trigger. This again is due to the fact that the receiver is comparing the two input signals with each other independent of a voltage reference. The degree of noise rejection by the receiver is dependant of the type of signaling and the termination method used.

![Figure 7.14 Differential Signals](image)

For distributing clocks within a system, LVDS, LVPECL, and Differential LVCMOS are the most popular drivers used today. We will focus on these styles and address the proper termination.
**LVDS**

Low Voltage Differential Signaling (LVDS) is defined by two industry standards: the IEEE-Scalable Coherent Interface (SCI) and the ANSI/TIA/EIA-644 specification. The EIA-644 specification defines the electrical characteristics of the signaling and is generally followed for most LVDS applications.

The output stage of an LVDS driver is defined as a current source. Therefore, termination is not only required to eliminate unwanted reflections, it is also necessary to provide a voltage differential for the receiver inputs.

The voltage swing range of LVDS is from 250 mV to 450 mV with 350 mV as the typical value for the driver. Due to its small voltage swing, LVDS is able to reach faster transition speeds than that of LVCMOS or LVTTL. The specification allows for 622 Mbps but a variety of devices have exceeded these speeds. Another characteristic of LVDS is that it centers its voltage swing at 1.2V as shown in Figure 7.15, and is known as its common-mode voltage $V_{CM}$. This provides an additional margin for noise originating on the power and ground planes.

![Figure 7.15 LVDS Signal Levels](image)

There are many published methods for terminating LVDS circuits that include no components to as many as half a dozen. We will examine several of them and highlight the benefits of each.

The first type of termination is no external termination as shown in Figure 7.16. This assumes that the chosen LVDS receiver has an internal resistor to provide the necessary differential voltage for the comparator. The range of value for the internal resistor is typically from $100\,\Omega$ to $120\,\Omega$. The benefit of this configuration is obvious: no components are necessary. With a fixed termination value inside the receiver, the proper value cannot be chosen to match that of the impedance of the trace. Thus, reflections may occur on each of the signals.

![Figure 7.16 LVDS Internal Termination](image)
The most common termination found on LVDS traces is the single resistor terminator as shown in Figure 7.17. This resistor provides the necessary differential voltage for the input of the receiver and also terminates the two signals to each other. The value of the resistor is chosen based on the impedance of the trace:

\[ R_T = 2 \times Z_O \]

It can be thought of as having two resistors equal to \( \frac{1}{2} R_T \) in series and the center tap as having a floating ground. For LVDS, this is actually 1.2 volts. The signals will propagate down the trace having equal but opposite voltages centered on 1.2V. When the signals encounter the termination network, they appear as equal to the trace impedance, and hence no reflection occurs. As the two signals collide, they cancel each other so do not continue to propagate down the opposite trace.

![Figure 7.17 LVDS Single Termination](image)

But let's examine the case where the two signals are not exactly aligned. This can be a result of mismatched trace lengths, variations in board material, output driver skew, or a combination of these fluctuations as well as other sources. In this case, the two signals propagate down their respective traces but they are separated in time. As they encounter the termination resistor no reflections occur, however they do not cancel each other out. The signal that arrives early will begin its journey back to the source on the other trace until it encounters the other signal. It will partially cancel. The signal arriving later will still reach the termination resistor slightly attenuated because of the collision of the other signal, but it will not completely dissipate in the resistor network. It will also begin to propagate back to the source but on the other trace. The magnitude of the two signals is a function of the time difference \( T \) as it relates to the rise/fall time of the pulse. As each signal arrives at the source, it will encounter a low impedance driver and thus reflect back down the line. Depending on the magnitude, this could be substantial. If the trace length delay aligns with the clock period, these pulses could become quite large. Therefore, special attention should be afforded to signal skew especially for LVDS clock signals. Alternative solutions may also be used to dampen the unwanted pulses.

One method used to address the skew between the differential traces is to add a capacitor as shown in Figure 7.18. The resistor \( R_T \) can be split in two, each being half its original size and equal to \( Z_O \). A capacitor is then added at the mid point. If the differential signals are truly equal and opposite, only a DC value of \( V_{CM} \) will be realized at this junction. However, if the signals are skewed, an AC signal will appear. The capacitor is used to terminate these extraneous signals. The size of \( C \) is chosen such that the RC time constant is slightly larger than skew between the signals.
To provide superior LVDS receiver operation, the inputs are ideally referenced to 1.2V. If we use two resistors instead of one, we are also able to improve the characteristics of the termination. The connection at the two resistors is connected to V_{TT}, the termination voltage that is set equal to the V_{CM}. Note that V_{TT} isn’t connected to ground, doing so would cause the receiver to malfunction as shown in Figure 7.19.

LVPECL

LVPECL (Low Voltage Positive Emitter-Coupled Logic) is a variation of ECL but uses positive supply voltage instead of negative voltages. LVPECL outputs have specific loading requirements to ensure proper operation. Because of its open-emitter output design, it can source current but it cannot sink current. To allow the output to switch, some form of pull-down is required as part of the termination.

The voltage swings for LVPECL are larger than LVDS with typical V_{OL} at 1.6V and V_{OH} at 2.35 as shown in Figure 7.120. Similarly due to its small voltage swing, LVPECL is able to reach faster transition speeds than that of LVCMOS or LVTTL.
As with other types of drivers, LVPECL also needs to be terminated into a matching impedance transmission line to avoid unwanted reflections. It is differential so we must ensure all modes are terminated as with LVDS. There are several common ways to terminate LVPECL traces.

One method is Thevenin equivalent termination as shown in Figure 7.21. This is the same as the termination we discussed for single ended clock signals and is used for both signals. We also need to consider biasing and therefore must select the proper resistor values for impedance matching as well as the voltage levels.

![Figure 7.21 LVPECL Thevenin Termination](image)

Because LVPECL does not swing rail to rail and we must provide a pulldown path, a termination voltage $V_{TT}$ is required. This value is usually specified as $V_{CC} - 2V$, which for a 3.3V system yields $V_{TT}$ to be 1.3V. This value is slightly less than $V_{OL}$ thus provides the required pulldown path. Knowing $V_{TT}$, we can now calculate the values of $R1$ and $R2$ for the termination.

$$R1 = \frac{V_{CC} \times ZO}{V_{TT}}$$

$$R2 = \frac{V_{CC} \times ZO}{(V_{CC} - V_{TT})}$$

For example, if we have a 3.3V system with a 50-ohm trace, the values of $R1$ and $R2$ are computed to be:

$$R1 = \frac{3.3 \times 50}{1.3} = 127\Omega$$

$$R2 = \frac{3.3 \times 50}{(3.3 - 1.3)} = 82.5\Omega$$
The Thevenin equivalent termination design will yield excellent results for clock signals since all modes are terminated. No reflections will occur between each signal or back through any of the resistors to the other signal path as shown in Figure 7.22. As in the case with the single ended signals, the termination resistors need to be near the receiver. The ideal position is just past the receiver along the trace.

The disadvantage to the Thevenin equivalent termination is the amount of power dissipated. Since it provides a resistive path from V\text{CC} to ground for each signal, current is constantly flowing. The power dissipated for a 3.3V system in this configuration is nearly 52 mW for each signal or 104 mW for the differential pair.

\[
\text{Power} = \frac{V^2}{R} = \frac{(3.3)^2}{(127 + 82.5)} = 52 \text{ mW}
\]

An alternative to achieve lower power is the parallel termination method, also known as shunt bias termination. This method uses a pulldown resistor equal to Z\text{O} on each signal to the termination voltage V\text{TT}, as shown in Figure 7.23. This type of biasing requires an additional voltage supply of 1.3V to provide the V\text{TT} level. The least amount of power is dissipated using this method. If we use V\text{OH} and V\text{OL} as 2.35V and 1.6V, respectively, for a 50% duty cycle clock, the power is about 11 mW for a 50\Omega trace.

\[
\text{Power} = \frac{1}{2} * (V\text{OH}-V\text{TT})^2/R + \frac{1}{2} * (V\text{OH}-V\text{TT})^2/R
\]
In some systems, $V_{TT}$ of 1.3V may be difficult to obtain and a termination method that doesn’t require a unique voltage can be beneficial. The Y-Bias configuration connects both of the termination resistors for each differential signal to a single load resistor. Since the net sum of the current in the two signals remains constant, the voltage drop across this load also remains constant. $R_L$, the load resistor, is selected to provide adequate biasing.

![Figure 7.24 LVPECL Y-Bias Termination](image)

$R_L$ is determined by summing the currents of a logic 1 and a logic 0, then calculating the resistance to provide an equivalent $V_{TT}$:

$$I_{OH} = \frac{(V_{OH} - V_{TT})}{R_T}$$

$$I_{OL} = \frac{(V_{OL} - V_{TT})}{R_T}$$

$$R_L = \frac{V_{TT}}{I_{OH} + I_{OL}}$$

For a 3.3V system, we can simplify and combine these equations:

$$R_L = R_T$$

Therefore, the value of $R_L$ should be set to the termination value. At DC and low frequencies, the power supply is assumed to be a short circuit. However, at very high frequencies, the power supply appears as near infinite impedance. A 0.1 µF to 0.01 µF capacitor is connected to $V_{CC}$ to create an AC short at high frequencies.

**Shunt Bias with Pulldown**

Probably the most versatile and easy to implement termination network for LVPECL buffers is the Shunt Bias with Pulldown, as shown in Figure 7.25. This method starts with the standard Shunt bias resistors whose $R_T$ value is equal to the trace impedance. Instead of a separate power supply to source $V_{TT}$, a voltage divider resistor network is used. Capacitor $C1$ is used to provide an AC path to ground in the event the differential signals have skew. This value typically ranges from 0.01 µF to 0.1 µF. The final piece of this circuit is a pair of 150-ohm pulldown resistors at the source. Because LVPECL drivers are open emitter and do not sink current, some ringing may occur in the high-to-low transition. These resistors will
help prevent this condition. Also, for designs where the signals are driven through a backplane to another card, they will provide the proper pulldown necessary for the driver if the receiving card is unpopulated.

![Diagram of Shunt Bias with Pulldown](image)

**Figure 7.25 Shunt Bias with Pulldown**

In Figure 7.26, the simulation of this design is shown. The DC signal on the scope trace just above 2 volts is the voltage divider of R1 and R2. By adjusting this bias voltage, the signal levels can be optimized for the specific LVPECL receiver in the design. The \( V_{\text{TT}} \) bias voltage applied is slightly higher than the Shunt Bias method because of the 150-ohm pulldown resistors. The other two signals are the inputs to the differential receiver.

![Graph of LVPECL Shunt with Pulldown](image)

**Figure 7.26 LVPECL Shunt with Pulldown**

Differential Clock Driver (12.3 Output Impedance), 50-ohm Resistors to bias of 2v Vtt, 4-inch Long Trace, 49.1 ohm Stripline (Effective), two 50-ohm pulldown resistors.

**Terminating Unused Outputs**

Clock buffer devices may have more outputs than needed for a design. We have addressed the various termination techniques for loaded outputs so now let’s focus on what should be done with the unused pins. For devices whose output structure requires termination in order to transition, such as LVPECL, unused outputs can be left unattached. For all other types of output drivers we may need additional termination.
The two areas of concern with unused outputs are EMI and clock skew. Additional EMI results from the unused pin acting as a small antenna and radiating the transitioning signal. This can be of consequence if the pin size is large or if there are many pins together that create a small grid. However, most modern package styles have geometrically small pins and a few unconnected outputs have no appreciable impact on EMI.

For applications where pin-to-pin skew is critical, un-terminated unused outputs may be of consequence. There are two different factors that can affect the output skew of the device. For very fast signals, those that are operating at less than 1 ns rise times, frequency harmonics can feed back into the device. This causes unwanted noise within the component similar to power supply noise. The result is excessive skew on the outputs. A second factor that affects the output skew is a delta in the output driver current. However, this is very dependent on the internal design of the component. If the clock generator or buffer has a single bank output, meaning all outputs have the same internal voltage feed, there is no difference in the current due to unloaded outputs. But with devices with multiple banks and a difference in current between the two banks, skew between the banks (and their respective outputs) can increase. This is a result of one bank with more loads than its neighboring bank due to unterminated outputs. This will cause the output driver banks to draw different amounts of current and thus increase the skew. The magnitude of the skew is small and typically is less than 30 ps.

When small amounts of EMI or slight variations in skew needs to be avoided, unused outputs should be addressed. Many clock buffers have output driver enables either controlled by dedicated inputs or through a serial control bus. These conditions can be managed by disabling unused outputs. Without driver enables, the other option is to terminate. Termination can be accomplished with either a capacitor or a resistor. By using a small capacitor to ground, generally 5–10 pF, the output is terminated into a lumped load. The capacitor also slows the edge rate to keep EMI to a minimum. Instead of the capacitor, a resistor to ground can also be used to terminate the output. The value of the resistor is chosen large enough as to not exceed the \( I_{OL} \) of the driver. The disadvantage of the resistor design is that it continuously consumes power whereas the capacitor design does not.

A good practical approach to unused outputs is to place pads for either the capacitor or resistor termination in the design. Keep the trace lengths very short on the buffer output pins to prevent unwanted EMI. During the initial testing of the design, analyze the output skew of the clock buffer to ensure it is within tolerance. If excess skew is observed, populate the termination components.
Conclusion

Clock signals are the foundation of any high-speed digital design. For the circuit to operate without fail, signal integrity is required. With either single ended or differential signals, the goals are the same: eliminate unwanted reflections. Reflections caused by changes in impedance can be managed by selecting a suitable termination technique. After the termination technique has been chosen and the circuit sketched, it is always recommended to simulate the design.
Bypass capacitors, also commonly referred to as decoupling caps, are often the most confusing component in a high-speed design. Engineers know that bypass capacitors need to be in the design, but the purpose and operation of the part is often unclear. This section attempts to demystify the use of bypass capacitors for all types of design.

**Capacitor Basics**

The bypass capacitor has three major purposes. It is used to prevent $V_{CC}$ and ground bounce on the power supply pins, it provides a low impedance path from the power plane to the ground plane, and it provides a signal return path between power and ground planes. Supply-line ripple and impedance control are addressed in this section, and the signal return paths are discussed in the chapter titled PCB Layout Considerations.

Decoupling capacitors shunt unwanted AC variations (noise) in two directions by introducing a low-impedance path to ground. They prevent noise from entering the device from the power plane and they also suppress noise from the device being coupled into the power plane.

Decoupling caps are more than a capacitor. They are in effect a capacitor in series with an inductor and a resistor, as shown in Figure 8.1.

![Figure 8.1 Capacitor Model](image-url)
The series L and R are called equivalent series inductance (ESL) and equivalent series resistance (ESR), respectively. The effective impedance of the equivalent circuit is written as the following equation, where f is the frequency. Figure 8.2 graphs this equation for typical parasitic ESR and ESL values found in a 1.0 μF ceramic capacitor.

$$Z = \sqrt{ESR^2 + \left(2\pi f \cdot ESL - \frac{1}{2\pi f \cdot C}\right)^2}$$

**Figure 8.2 Impedance of Capacitor with ESR=0.015 Ohms, ESL=2nH, C=1µF**

To understand the operation over frequency, manufacturers provide such curves for each capacitance value, package type, and composition. Figure 8.2 shows that the minimum capacitor impedance is limited by its ESR value. This impedance occurs at the series resonant frequency (fo), which can be calculated from the equation below. On either side of this frequency the impedance increases. The ideal capacitor has zero impedance at those frequencies that need bypassing.

$$fo = \frac{1}{2\pi \sqrt{ESL \times C}}$$

In reality, the capacitance curve supplied by the manufacturer will change once it is mounted on the PCB board due to the mounting inductance. This inductance shifts the resonant frequency, fo, (i.e., minimum impedance point in Figure 8.2) down to roughly 70% of its original (isolated capacitor) value. The mounting inductance is dominated by the pad and via structures used to connect the capacitor to the power plane. Other chapters in this book discuss proper layout techniques to minimize this inductance.

To show the influence of ESL on the capacitor impedance, Figure 8.3 compares the original capacitor from Figure 8.2 with another capacitor having an ESL (arbitrarily set) four times less. The lower ESL capacitor not only pushes the resonant frequency higher, but also reduces the impedance over a wider frequency range, making this capacitor much more...
effective for decoupling. Minimizing a decoupling capacitor's mounting inductance is an often overlooked but important aspect of good PCB design.

![Figure 8.3 Reducing ESL Increases the Effectiveness of High-Frequency Capacitors](image)

**Classes of Ceramic Capacitors**

There are many different types of capacitors available for decoupling. The ceramic type capacitors are the most effective for bypassing higher speeds than other types of discrete capacitors. Capacitors can be affected by temperature, voltage, and time. Classes are used to rate the properties associated with ceramic capacitors.

Class 1 capacitors, or temperature compensating capacitors, have predictable temperature coefficients and in general do not exhibit any aging characteristics. Thus, they are the most stable capacitor available. The most popular Class 1 multilayer ceramic capacitors are C0G (NP0) temperature compensating capacitors. EIA Class 2 and 3 capacitors provide a wider range of capacitance values and temperature stability. The most commonly used Class 2 dielectric is X7R. X7R provides intermediate capacitance values that vary only ±15% over the temperature range of −55°C to 125°C.

Class 3 dielectrics are the Y5V and Z5U series. The Y5V provides the highest capacitance values per physical volume and is used where limited temperature changes are expected. The capacitance value for the Y5V can vary from 22% to −82% over the −30°C to 85°C temperature range. The Z5U dielectric falls between X7R and Y5V in both stability and capacitance range.

Variations in voltage have little effect on Class 1 dielectrics, but they do affect the capacitance Class 2 dielectrics. In class 2 capacitors, the application of DC voltage reduces the capacitance while the application of an AC voltage within a reasonable range tends to increase capacitance. Figure 8.4 shows the effects of AC voltage on an X7R capacitor.
The capacitance of bypass capacitors also changes as a function of the actual applied DC voltage. The effect of the DC voltage is shown in Figure 8.5 at 20°C.

Another factor that affects the capacitance of a specific device is time. Class 2 ceramic capacitors change capacitance with time as well as temperature, voltage and frequency. This change with time is known as aging. Aging is seen as an exponential loss in capacitance over time. A typical curve of an aging rate for several different ceramic capacitors is shown in Figure 8.6.
If a Class 2 ceramic capacitor has been sitting idle for a period of time and begins to age, it can be “reset” by heating the device. By raising the temperature above its curie point (125°C for 4 hours or 150°C for ½ hour) the part will reverse its aging factor and return to its initial capacitance value. Because the capacitance changes rapidly after de-aging, the capacitance measurements include a time period for the process. Various manufacturers use different time bases but the most popular one is twenty-four hours after the “last heat.” The aging curve can be altered depending on voltage and other stresses.

**Power Distribution System Impedance**

The goal of any power distribution system is to provide sufficient voltage to the load over all operating conditions the system is using. Figure 8.7 illustrates a simplified model of a power distribution system. Ideally, the power supply is a constant voltage source with no output impedance (i.e. $Z = 0\Omega$) across all frequencies. In reality, the power supply has a low impedance only at low frequencies. Higher frequencies require decoupling capacitors to lower the overall impedance.

Figure 8.7 Simplified Model of a Power Distribution System

Figure 8.8 shows how decoupling capacitors lower the total impedance for higher frequencies. In this example, the power supply (i.e. DC/DC converter) is modeled with an ESR of 1 mΩ, and an ESL of 60 nH. The supply impedance increases with frequency as expected because of its inductive nature. A separate curve is shown for four parallel tantalum capacitors, each having an ESR, ESL, and C of 30 mΩ, 4 nH, and 300 µF, respectively.
Although the supply impedance is large for high frequencies, and the tantalum capacitor impedance is large for low frequencies, the parallel combination of both achieves the lowest impedance across all frequencies. In this way capacitors can be added to the system to lower the overall impedance. Capacitors with lower values of capacitance reach further into the higher frequencies. A typical design will use several different values of capacitors to provide a low impedance across the desired frequency range.

How low an impedance is needed is determined by how much power supply ripple can be tolerated for a given change in load current. The \( Z_{target} \) equation below calculates this target impedance, where \( \Delta V \) and \( \Delta I \) are the maximum change in load voltage and current.

\[
\Delta V = V_{DD} \times \text{Ripple} = 3.3V \times 2\% = 17\,\text{m\Omega}
\]

\[
\Delta I = 4A
\]

For example, if the maximum ripple that can be tolerated when drawing 4A from a 3.3V supply is 2%, then the target impedance is 17 m\( \Omega \). Decoupling capacitors are then selected to assure that this impedance is not exceeded over the entire frequency range in which all possible current transients are expected. In practice, current transients are usually about half of the maximum current, and so this calculation may overestimate the required impedance by a factor of two. The PCB designer needs to understand the nature of the board’s current consumption to determine whether \( Z_{target} \) or twice this value is acceptable.

**VCC and Ground Bounce**

The bypass capacitance is sized to satisfy a maximum load current requirement. When a clock buffer’s outputs switch, the power terminals will sag due to a voltage drop caused by unwanted package parasitics between the chip and PCB.
This drop, or glitch as it refers to a temporary condition, occurs because of the inductance and resistance of the device lead frame and bonding wire. This condition is also commonly referred to as VCC (i.e., power) and ground bounce. Figure 8.9 shows this voltage drop across the package inductance at the power node. Although not shown, a similar situation exists where the chip connects through the package to ground.

VCC and ground bounce modulate the supply voltages seen internally by the chip, which reduces circuit headroom, increases EMI, increases all types of jitter within the chip, and erodes noise margin. If large enough, it eventually leads to unstable systems (i.e., intermittent logic failures). The amount of voltage drop, V, seen by the chip can be estimated using L as the package inductance, and di as the change in load current per unit time, dt.

\[ V = L \frac{di}{dt} \]

The bypass capacitor’s function here is to supply this momentary need in current with its stored charge to minimize this drop. As charge moves from the bypass capacitor on the board to the bypass capacitor on the chip, the voltage drop, or “ripple,” goes away on the chip and appears on the PCB supply rail. This charge transfer between capacitors causes the terminal voltage on one capacitor to go down as the terminal voltage on the other capacitor goes up. With time, the drop on the PCB rail also goes away as charge is supplied by the power supply. Effective decoupling systems minimize the amplitude and duration of this ripple. Figure 8.10 shows the benefit of minimizing VCC and ground bounce on output clock jitter (measurements taken on CY28410).
Unfortunately, IC industry trends toward lower supply voltages, faster switching speeds, and increased logic density on chips have increased the switching noise appreciably. Additionally, customers’ drive to reduce costs have resulted in the IC industry continuing to use inexpensive packages (with their inherent poor package parasitics) for higher and higher frequencies. Such forces have made $V_{CC}$ and ground bounce a major source of failure today. Aside from using differential signaling or changing to a better package, other methods to minimize these failures include keeping power pin traces as short as possible, using multiple power pins per device, double-bonding wires, using connectors with low pin resistance, and avoiding making cuts in the ground plane. Proper decoupling capacitor design also helps prevent such failures, as discussed in the following sections.

**How High is High Frequency?**

It used to be that “high frequency” referred to designs using clock or signal speeds more than a couple hundred MHz. However, since 0.25 µm process technology (and below), transistors switch so fast that the frequencies present in the switching edge exceed the classical frequency related to the periodicity of same waveform. The classification of high speed design has therefore shifted its focus from clock frequencies to edge rates. Using this classification, all designs using advanced process technology must be designed according to high-speed practices. This includes control signals operating at lower frequencies (i.e., 1 MHz). Figure 8.11 illustrates why.
Figure 8.11 shows a spectrum analyzer plot of the frequencies present in a 50 MHz clock waveform coming from an output buffer fabricated in today’s process technology. Given the time varying waveform, this spectrum could be calculated by taking its Fourier series. The sequence of hills in the spectrum’s envelope is characteristic of a sinc function (i.e., the Fourier transform of a square pulse). This figure shows that the frequencies making up this 50 MHz clock extend past 4 GHz! Although most of these higher frequency components can be safely ignored due to their relatively small amplitudes, there is still appreciable content up to about 500 MHz. If the power distribution system servicing this clock is designed to only 50 MHz then the resulting waveform would appear rounded, and timing margins would shrink. Even if the historical practice of designing up to the third harmonic (i.e., 150 MHz in this case) is followed, there may not be enough high frequency content to preserve the edge rates assumed in the timing analysis. One way to estimate the required frequency range during the design phase is to do a Fourier transform of the time domain signals. If this isn’t possible, a rough generalization suggests that frequencies up to one-half the fastest signal transition (rise or fall) frequency need to be bypassed.

\[ f_{\text{BYPASS}} = 0.5 \div \text{Transition Rate} \]

It should be noted that decoupling is really a four-part system, as shown by Figure 8.12. The four parts are comprised of: the power supply, the bulk (low frequency) capacitors, the ceramic (high frequency) capacitors, and the intrinsic capacitance in the board. Each provides decoupling in its respective frequency bands with the power supply addressing the lowest frequencies and the intrinsic board capacitance addressing the highest frequencies. The difference between the capacitance between these parts gives rise to peaks and valleys between effective frequency ranges of each part. These peaks and valleys will always exist, but they can be moved slightly.

Figure 8.12 Decoupling is a Four-Part System

Generally, engineers have limited ability to alter the characteristics of the power supply and board plane capacitance after they are chosen for the system, but it is possible to vary the
bulk and ceramic capacitor values. The goal is to have the four-part system provide a sufficiently low impedance path throughout the required frequency range. By adjusting the values of the bypass capacitors, the small impedance peaks can be shifted to obtain the lowest impedance for the frequencies of interest. The following sections discuss how this is done.

**Designing with Discrete Capacitors (<200 MHz)**

Discrete capacitors can be broadly categorized as either electrolytic bulk or ceramic high-frequency capacitors. Bulk capacitors are used to maintain the target impedance above frequencies where the switching power supply impedance is no longer useful. Continuing with our earlier example, the switching supply reaches the target 17 mΩ around 40 KHz in Figure 8.8. The switching power supply impedance refers to the impedance from the DC/DC converter, or voltage regulator module, that converts one DC voltage to another. It usually provides low impedances up to the KHz or tens of KHz range. Bulk capacitors are used to extend the target impedance to the MHz range. For example, the tantalum capacitors used in Figure 8.8 extend the 17 mΩ impedance up to 2.5 MHz. Tantalum capacitors are available between 1 and 1000 µF, and are operated at half of their rated voltage to lower their failure rate.

Ceramic capacitors are small, inexpensive and very reliable. They extend the target impedance to roughly 200 MHz. Commonly used ceramic capacitors values range from 0.001 to 1 µF, and may be bought either individually or as packaged arrays. Arrays have the benefit of lower ESR, but cost up to ten times more. The different types of dielectrics used to make ceramic capacitors have different characteristics and are classified as discussed previously. Above 200 MHz, plane capacitance is required to extend the target power distribution impedance, as discussed in the next section.

A common way to lower the overall impedance is to combine identical capacitors in parallel, as shown in Figure 8.13. Like resistors, the impedance of capacitors halves as the number of devices in parallel doubles. If the number of capacitors needed to produce the target impedance becomes unreasonable, a lower ESR capacitor may be used to reduce the total number of components.

![Figure 8.13 Doubling the Number of Capacitors in Parallel Cuts Their Impedance in Half](image)
Capacitors of different values used together can create an undesired anti-resonance peak. Also referred to as parallel resonance, or impedance-hole, this peaking occurs between the series resonant frequencies of the individual capacitors, as shown in Figure 8.14. Notice that the resulting impedance spike is higher than the individual impedance of either capacitor. Such peaking results from a parallel LC tank circuit that forms when one capacitor behaves as a capacitor and the other behaves like an inductor. When this occurs, the goal of meeting a target impedance is not met. But instead of avoiding using capacitors of different values, one or more of several solutions may be used.

Figure 8.14 Capacitors of Different Values Combine to Produce Unwanted High Impedance Spikes

One condition leading to large parallel resonance is if the capacitor values are widely different. Choosing capacitor values closer together reduces impedance peaking. A third (or however many more are needed) capacitor value can then be added to recover the original frequency range. Typically decade selection of bypass capacitor value is sufficient (ie. 0.1, 0.01, 0.001 µF, etc.)

Another solution is to reduce the parasitic inductance associated with the capacitors. This can be achieved by mounting the capacitors on low inductance pads, paying careful attention to minimize the distance to the power plane below them. The parallel resonant peaking is proportional to the quality, or Q, of the circuit, which is proportional to inductance divided by resistance. This is the reason very low ESR capacitors need to have very low mounting inductance; otherwise the Q, or peaking, is very large. Recall also that minimizing mounting inductance has the added benefit of lowering the impedance over a broad frequency range.

Another solution to reduce parallel resonance is to choose a capacitor with a larger ESR and place several in parallel. A larger ESR has the same effect as reducing the mounting inductance (since Q, or peaking, is proportional to inductance, and inversely proportional to ESR). This is a particularly effective option because additional capacitors can be combined in parallel to reduce the overall impedance, without affecting the parallel resonance. The reason Q doesn't change is because the parallel combination of capacitors reduces the effective ESR and ESL by the same amount (see Figure 8.13). This change in ESL and ESR is normalized out when they are divided to calculate Q since they scale with each other, so Q remains constant while the overall impedance decreases.
When designing a bypass capacitor system with discrete devices, it is common practice to use more capacitors in parallel as the frequency increases. This is because ESR values for large capacitors are naturally low, but increase as the capacitor value decreases. Thus for higher frequencies the number of capacitors placed in parallel must increase to meet the target impedance. As frequencies approach 100–200 MHz, ceramic capacitors have sufficiently large ESR such that the required number devices in parallel may no longer be practical to use. In this case, minimizing the mounting capacitance is critical to increase the series resonant frequency (i.e., minimum impedance frequency) as high as possible. At this point the PCB power plane capacitance must take over to extend the target impedance to higher frequencies.

Systems without very low target impedances, may not require using combinations of differently sized ceramic capacitors. In this case, fewer required components may be achieved by choosing a single capacitor value, and using it in parallel as many times as needed to achieve the target impedance over the desired frequency range. This scheme is popular when switching current transients are sufficiently low.

**Designing with Plane Capacitance (>200 MHz)**

For frequencies above roughly 200 MHz, the power plane capacitance in the PCB is used to meet the power distribution system’s target impedance for minimizing power supply ripple. A typical PCB stack-up is shown in Figure 8.15. Although neighboring signal lines are both drawn running perpendicular to the page, in reality they are laid out running perpendicular to each other to minimize crosstalk (they're just drawn this way in the diagram to avoid confusing them with the power or ground planes).

The parallel plate’s capacitance (C) can be calculated using the equation below where \( E \) (permittivity of vacuum) is \( 0.2249 \times 10^{-12} \text{ F/in} \), \( Er \) (dielectric permittivity) is about 3.8 to 4.2, \( A \) is the area, and \( T \) is the dielectric thickness.

\[
C = \frac{E \times Er \times A}{T}
\]

Note that this equation shows the smaller the distance between planes (i.e., \( T \)), the larger the capacitance that results. Thus, the power plane should be placed as close as possible to the ground plane.

A power plane pair (i.e., \( V_{DD} \) and Ground) forms an unterminated transmission line, and therefore has all of its associated characteristics. Forward and reverse traveling waves combine to create standing waves, in which the plane’s transmission line looks like an open circuit for multiples of a half wavelength, and a short circuit for a quarter and three-quarters of a wavelength. Assuming the plane pair extends over the entire PCB board, we can use the length of the board to calculate where the resonant frequencies are. The propagation
velocity of a signal in a transmission line, where \( c \) is the speed of light in air (11.8 in/ns), can be expressed as:

\[
V_p = \frac{c}{\sqrt{Er}}
\]

Figure 8.15 Cross-Section of a Typical PCB Stack-Up

The propagation velocity can be used to calculate where the impedance spikes occur. If \( Er \) is 4 for a particular PCB, the propagation velocity is 6 in/ns. One wavelength of the board would be the distance the signal travels in one cycle. If the PCB is 10 inches long, the frequency corresponding to one wavelength is 6 in/ns divided by 10 inches, or 600 MHz. Thus, resonance occurs at \( \frac{1}{4} \), \( \frac{1}{2} \), \( \frac{3}{4} \), and 1 wavelengths; parallel resonant frequencies are 150, 300, 450, and 600 MHz. Figure 8.16 shows the PCB impedance for this example. Although the resonant frequencies can be estimated, field solving software is required to accurately graph such curves. When discrete capacitors are added to the PCB, they will shift the resonant peaks slightly. However, their influence extends only a few hundred MHz, so the PCB impedance will resonate thereafter at the calculated wavelengths.

Figure 8.16 Example PCB Impedance
The amount of capacitance required in the power plane can be estimated by:

\[
\frac{dV}{V_{DD}} = \frac{Clines}{C_{plane}}
\]

Where \(dV\) is the ripple that can be tolerated, \(V_{DD}\) is the supply voltage, \(Clines\) is the total capacitance of all lines that can be simultaneously switched (including parasitics), and \(C_{plane}\) is the power plane capacitance.

If the PCB is physically too small to place the required amount of plane capacitance, any available space in neighboring signal planes may be filled to make up the difference. Connect all filled signal planes to power and ground planes by vias.

The key point to remember is that a digital system is composed of many frequencies that need to be bypassed simultaneously. Since currents travel in the path of least impedance, the smallest loop area is the objective. A PCB plane power/ground pair can be viewed as an effective localized capacitor capable of minimizing loop area for higher frequency bypassing.

Bypass Guidelines

As seen by the series of plots, the package size and value of decoupling capacitors makes a difference. How the capacitors are connected plays a key role in how well the buffer or PLL performs. The inductance of the leads must also be minimized. Vias placed below the pads as shown in Figure 8.17 provide the lowest inductance and should be used whenever possible. Long, narrow traces should be avoided as they increase inductance. Passing through a via is acceptable provided that the path is lower inductance than an alternative longer trace. The following is a list of guidelines for designing with bypass capacitors.

1. Select the largest value of capacitor in the smallest available package
2. Design the bypass system to satisfy power-line ripple and target impedance requirements. Balance the number of capacitors per value across the network to shunt a broad range of frequencies with minimum impedance.
3. Keep bypass capacitors close to components
4. Minimize mounting inductance
5. Use capacitors in parallel to lower the overall impedance
6. Consider multi-cap packages to assist layout of multiple and/or different-valued capacitors.
7. Use power plane capacitance for frequencies above 200 MHz
8. Use good PCB layout techniques
**Example Design**

An example bypass capacitor system is designed to illustrate the above concepts. The process begins by calculating the following parameters.

1. Maximum change in load current: 2.2 Amps
2. Maximum ripple that can be tolerated: \(3.6 \text{ V} \times 1.5\% = 54 \text{ mV}\)
3. Target power distribution system impedance: \(Z_{\text{target}} = \frac{54 \text{ mV}}{2.2 \text{ Amps}} = 25 \text{ m}\Omega\)
4. Frequency range for target impedance: \(f_{\text{BYPASS}} = \frac{0.5}{0.5 \text{ ns}} = 1 \text{ GHz}\)
5. Total capacitance from lines drawing maximum simultaneous switching current: \((8\text{-bit bus}) \times (3 \text{ pF load capacitance per bus line}) \times (3 \text{ pF/in board capacitance}) \times (10 \text{ inch bus line length}) = 720 \text{ pF}\)
6. Required plane capacitance: \(C_{\text{plane}} = \frac{(720 \text{ pF}) \times (3.6 \text{ V})}{(54 \text{ mV})} = 48 \text{ nF}\)
7. Permittivity of PCB dielectric: \(\varepsilon_r = 4.0\)
8. Propagation velocity of signal in PCB: \(\frac{11.8 \text{ in/ns}}{\sqrt{4.0}} = 6 \text{ in/ns}\)
9. Distance into board where resonant frequencies wish to be measured: 10 in
10. Estimate of frequencies where PCB impedance looks like an open circuit: \((0.5, 1, 1.5, \text{ etc.}) \times \left(\frac{6 \text{ in/ns}}{10 \text{ in}}\right) = 300, 600, 900, \text{ etc. MHz}\)
11. Estimate of frequencies where PCB impedance looks like a short circuit: \((0.25, 0.75, 1.25, \text{ etc.}) \times \left(\frac{6 \text{ in/ns}}{10 \text{ in}}\right) = 150, 450, 750, \text{ etc. MHz}\)

The low frequency (<200 MHz) part of the design consists of picking the number and size of discrete capacitors that need to be combined with the DC/DC converter to provide the target 25 mW power distribution impedance on the PC board. The high frequency (>200 MHz) part of the design assures that at least 48 nF of plane capacitance exists to keep the ripple within limits. Although a field solver is needed to accurately model resonant interactions at high frequencies, a spreadsheet can be used for quick analysis of the decoupling impedance.

The completed design is shown in Figure 8.18. The decoupling impedance is derived from four different types of systems. From low to high frequency, these are the switching power supply impedance, bulk (i.e., tantalum) capacitors, high-frequency (ceramic) capacitors, and plane capacitance. Although single capacitor impedances are plotted in the figure, parallel capacitors were used to reduce the total impedance. Table 8.1 lists the components used to satisfy the 25 mW target impedance across all impedances.
Table 8.1 Decoupling Capacitor System

<table>
<thead>
<tr>
<th>Component</th>
<th>Quantity</th>
<th>C (µF)</th>
<th>ESR (Ω)</th>
<th>ESL (nF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Switching supply</td>
<td>1</td>
<td>—</td>
<td>0.001</td>
<td>60</td>
</tr>
<tr>
<td>Tantalum</td>
<td>2</td>
<td>100</td>
<td>0.05</td>
<td>4</td>
</tr>
<tr>
<td>Ceramic 1.0 µF</td>
<td>2</td>
<td>1</td>
<td>0.02</td>
<td>2</td>
</tr>
<tr>
<td>Ceramic 0.1 µF</td>
<td>3</td>
<td>0.1</td>
<td>0.03</td>
<td>1.2</td>
</tr>
<tr>
<td>Ceramic 0.01 µF</td>
<td>6</td>
<td>0.01</td>
<td>0.05</td>
<td>1</td>
</tr>
<tr>
<td>Ceramic 0.001 µF</td>
<td>10</td>
<td>0.001</td>
<td>0.06</td>
<td>0.9</td>
</tr>
</tbody>
</table>

This decoupling example addresses the entire system. When it comes to decoupling a specific chip, the placement of these capacitors becomes very important. Of course, it's not possible to place all of them directly next to any particular chip. What is important is to minimize the inductance between the capacitor on the board and the chip supply pins to make the capacitor more effective at higher frequencies. As the capacitor value drops, its associated ESL also drops, so it becomes more important to place smaller value capacitors closer to the chip to help prevent the inductance from being dominated by the board. A 0.1 µF or 0.01 µF capacitor is commonly placed on each of the chip's power and ground pairs. However, the system designer must ultimately balance this relationship based on the chip's switching current requirements, the noise in the system, and the tolerable voltage ripple on the chip's supply pins.
Conclusion
Bypass capacitors reduce power supply ripple and minimize $V_{CC}$ and ground bounce by providing the necessary charge to satisfy changes in load current. The industry’s trend toward shrinking devices has shifted the classification of high-speed design from clock frequencies to edge rates. Designs today must therefore include sufficient plane capacitance to satisfy frequencies above the limit where discrete capacitors are effective, which is around 200 MHz. An example bypass system is shown highlighting many of the techniques presented above.
Any device capable of generating signals with frequencies in the RF range is a potential source of Electro-Magnetic Interference (EMI). These signals can cause interference in the normal operation of electronic devices such as radios, televisions, cell phones and other types of equipment. The primary sources of EMI in most systems are the clock generation and distribution circuits. In this chapter, we will discuss how EMI is generated, how to measure it, and how to minimize its impact.

Interference is caused by electro-magnetic waves that are produced by charged particles moving in an electric field. This condition occurs wherever electric signals exist. There are regulatory agencies that require devices that produce EMI to adhere to a certain set of rules and regulations. Among these rules and regulations is a requirement that the source of radiation not be greater than a pre-determined level at a certain distance from the source within a fixed frequency range. In the United States, the regulatory agency that governs the control of EMI is the Federal Communications Commission (FCC).

But what happens when the radiation sources are too great? How does the clock driving a processor affect a radio or telephone? FM radio stations operate in a frequency range of 88 MHz to 108 MHz. A 200 kHz guard band separates the stations from each other. In order to have good clarity and to receive stations from far away, radio receivers use high gain amplifiers to pick up weak signals.

A typical frequency in synchronous devices is 33.3 MHz and is often used as the clock source for PCI busses, ASICs, FPGAs, and processors. Associated with the 33 MHz is a series of harmonic frequencies. (Harmonics are integer multiples of the fundamental frequency and are discussed later in this chapter.) The 3rd harmonic of 33.3 MHz is 99.9 MHz and thus a board containing 33 MHz can cause distortion on a radio that is tuned to 99.90 MHz.
Causes of EMI

Clock sources can contribute to EMI in two ways. EMI can be produced through the repetitive nature of a synchronous clock and from an improperly terminated trace. The energy from the clocks radiates into a field through an antenna. An antenna might be in the form of PCB traces, PCB rework wires, components with insufficient shielding, connectors, cables (shielded or unshielded), or improperly grounded equipment.

In high-speed digital devices, fixed frequency clocks are the primary source of EMI because they are always operating at a constant frequency that allows energy to increase to higher levels. Signals that are non-repetitive or asynchronous will not generate as much EMI.

As the need for higher throughput has driven faster clock frequencies, signal transition rates have also increased. But with the faster rise and fall times comes an even larger increase in the energy level of the radiated signal. Figure 9.1 shows two signals that have the same frequency, amplitude, duty cycle, and phase. However, they differ in the signal transition rate. The clock with the faster rise time will have a measurably higher amount of radiated energy than the slower transitioning signal. The second factor that contributes to EMI is an improperly terminated trace. As discussed in Chapter 7 on termination, a trace can exhibit overshoot and undershoot when there is an impedance mismatch. When this condition occurs, radiated energy will increase. Depending on the severity of the overshoot and undershoot levels, this could represent as much as 3 to 4 dB of EMI at a particular signal, or node in EMI terms. If there are ten to twenty nodes with severe overshoot then passing the FCC compliance test is in jeopardy. Figure 9.2 shows an improperly terminated trace where overshoots are evident. The same circuit with series termination contains no overshoots, as shown in Figure 9.3. This clock will also exhibit less EMI.

![Figure 9.1 Different Signal Transition Rates](image)
Figure 9.2 Overshoot in Signal

Figure 9.3 Properly Terminated Trace
Measuring EMI
Measuring EMI is accomplished by placing a radiating device in a known environment such as an Anechoic chamber or Open Area Test Site (OATS). In this environment, the level of energy coming from the radiating source is measured and compared to the limits specified by the FCC. In 1975, Part 15 was established that relates to equipment that does not intentionally generate RF energy. The FCC specifies a maximum energy level that can be radiated at a particular frequency at a specified distance.

There are other groups responsible for regulating radio interference besides the FCC. Two noteworthy organizations are The International Special Committee On Radio Interference (CISPR) and The Voluntary Control Council for Interference by Information Technology Equipment VCCI. CISPR was established in 1934 by a group of international organizations to address radio interference. It is a non-governmental group composed of National Committees of the International Electrotechnical Commission (IEC) as well as other international organizations. CISPR is the standard typically followed in Europe.

The VCCI was formed in December 1985 by four Japanese industry associations in response to a government request that electronics manufacturers participate in the control of EMI. VCCI is considered the standard for Japan.

The FCC has two main classes of radiation levels: Class A and Class B. A Class A digital device is: “A digital device marketed for use in a commercial, industrial, or business environment and not intended for use by the general public or in the home.” A Class B device is defined as: “A digital device marketed for use in the home, although it could be used elsewhere.” Class B levels are more difficult to meet than class A level 5.

The following chart shows the voltage and dB levels allowed under the FCC Rules and Regulations, Part 15, for Class A and B Digital Devices.

<table>
<thead>
<tr>
<th>Table 9.1 FCC Class Limits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency (MHz)</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>30 – 88</td>
</tr>
<tr>
<td>88 – 216</td>
</tr>
<tr>
<td>216 – 960</td>
</tr>
<tr>
<td>&gt;960</td>
</tr>
</tbody>
</table>

When a testing lab measures a particular device, a report is generated containing a detailed chart and graph of the levels of energy measured at all frequencies from 30 to 1000 MHz. If the device being measured is compliant, the graph will show that the peak energy levels of any frequency are below the values in Table 9.1. However, if peak values exceed these limits, a change in the design is necessary to reduce the EMI.
Many IBIS simulation tools also provide the capability to analyze potential EMI hazards, see Figure 9.4. While these models are not exact, they can pinpoint areas of concern during the design phase of the project. This provides more time to make the necessary corrections.

Reducing EMI

There are several methods with which to solve an EMI problem in a digital system. The designer could choose to shield the design, filter a signal, or remove the energy from the offending source. These methods could be used individually or in conjunction with others.

The first method, shielding, is not an electrical solution but a mechanical implementation. Shielding uses metallic packaging to keep the EMI from escaping the unit. This method has been used often in the past but it can sometimes be a costly solution. It also doesn’t lend itself to an easy fix when an EMI problem is found shortly before a product release.

The remaining methods, filtering and energy removal, isolates the trace that is radiating the EMI. To identify which trace (or traces) is causing the problem, a test in the anechoic chamber or an EMI simulation should be performed. From this testing, an emission report will identify which frequencies exceed the specified limits. These particular frequencies are typically called hot spots. By knowing the frequencies (including the harmonics) the clock trace can be identified. The sections later in this chapter discuss frequency harmonics and how it relates to the fundamental frequency.

Since poorly terminated signals can cause hot spots, the first solution is to ensure all signals are properly terminated. The signals that are causing the EMI should be simulated and the traces should be analyzed for overshoot and undershoot. If there are exceptional amounts, then adjust the termination values to create a better waveform.
If all the signals are properly terminated and little to no overshoot is present, then the transition rate of the clock needs to be addressed. A substitution of a slower speed buffer may supply the answer. Many clock buffers have an option for high-speed or low-speed outputs. Often, these parts are either pin-for-pin replacements or the device has a programmable slew rate. If the lower drive is acceptable for the system, this may be the best solution. This method directly addresses the clock trace that is causing the problem and typically there is no additional cost to implement.

If a slower device is not available, filtering is a common way to slow the edge rate of a signal. This usually involves adding a capacitor to the signal that will soften the edge rate based on an RC time constant. The values of the capacitors generally range from 5 to 15 pF. Often designers will include these capacitors, which need to be placed near the source, in their schematics but not populate them unless an EMI problem is exposed. If the clock trace uses series termination, the capacitor can be placed on either side of the resistor to reduce EMI. However, for optimal termination and signal integrity, the capacitor needs to be placed between the driver and the resistor, as shown in Figure 9.5.

![Figure 9.5 EMI Reducing Caps](image)

Although this method reduces EMI, it does however, degrade the signal integrity of the clock. Instead of sharp, clean edges suitable for high-speed clocks, the edges become rounded. Also, capacitors may need to be added for every clock copy in the design.

**Clock Modulation**

Another method to address signals that emit excessive radiation is clock modulation. Clock modulation, also known as Spread Spectrum, began to appear in computer system designs as early as 1995. Most personal computers designed today use spread spectrum technology to reduce EMI. Spread spectrum provides a cost effective method to keep EMI low.

Clock modulation is a technique whereby an input reference clock at some frequency is modulated so that the frequency of the output clock varies slightly. For example, a 40 MHz reference clock with spread spectrum applied can produce an output swing from 39.60 to 40.40 MHz. This would represent a spread spectrum clock that has a 2% bandwidth centered on the reference frequency of 40 MHz. The purpose of modulating the frequency of a clock is to distribute the energy of a single or narrow band over a much wider band of frequencies. This reduces the amount of peak energy at any one frequency in the spectrum. The amount of EMI reduction is affected by the modulation profile, the percentage of frequency variation (bandwidth), and the modulation rate.
When a modulated clock changes from its minimum frequency to its maximum frequency, the spread spectrum logic applies a specific profile to the envelope that will best reduce EMI. This profile can be seen using several different types of equipment. The most effective piece of equipment to view the profile of a modulated clock is the modulation domain analyzer, which displays frequency over time. As the clock generator is increasing in frequency, the spread spectrum logic will cause the frequency to change only at predetermined times in the profile. The waveform of this profile is important in producing the maximum amount of dB reduction. Figure 9.6 illustrates the profile of a spread spectrum clock generator using the Lexmark profile, which is commonly referred to as a Hershey Kiss as it resembles the shape of the chocolate candy. Other profiles, such as a linear ramp are sometimes used to reduce EMI. However, they are found not to be as effective across the frequency spectrum as the Lexmark profile.

Figure 9.6. Lexmark Modulation Profile

In this particular profile example, spread spectrum is applied to a 65 MHz clock. The output frequency is sweeping from a minimum of 63.797 MHz to a maximum of 66.311 MHz. This is known as center spread because the 65 MHz clock is spread equally above and below its frequency. Notice that the rate of change in frequency is faster at the minimum and maximum peak frequencies and changing more slowly at the center of the frequency spectrum due to the profile.

The total amount of spread is called the bandwidth of the clock and is calculated by subtracting the minimum frequency from the maximum frequency:

\[ BW = F_{\text{MAX}} - F_{\text{MIN}} \]
The amount of spread of a modulated clock is most commonly represented in terms of a percentage of the reference frequency. The bandwidth, in percent, can be calculated by dividing the total spread amount by the reference frequency times 100:

\[
\text{BW\%} = \left( \frac{\text{BW}}{F_{\text{REF}}} \right) \times 100
\]

For the example in Figure 9.6, the BW and BW\% are calculated as:

\[
\text{BW} = 66.311 \text{ MHz} - 63.797 \text{ MHz} = 2.514 \text{ MHz}
\]

\[
\text{BW\%} = \left( \frac{2.514 \text{ MHz}}{65 \text{ MHz}} \right) \times 100 = 3.87\%
\]

This same 65 MHz clock can be observed on a spectrum analyzer for the purposes of determining dB reduction in the EMI of this clock. Figure 9.7 is a spectrum analyzer display of the 65 MHz clock with and without spread spectrum applied. The clock with no spread spectrum has a very narrow frequency range centered on 65 MHz. The energy peak is also higher than the other waveform. The wider scan is the clock with Spread Spectrum active. The center frequency is still 65 MHz as it is using a center spread technique. From this display, the amount of EMI dB reduction can be determined by measuring the difference between the peak energy in each of these clocks. This view shows a dB reduction of 6.48 dB at the fundamental frequency of 65 MHz. The one parameter in a spread spectrum clock that has the largest impact on dB reduction is the bandwidth of the modulated clock. If the bandwidth of this clock were increased, the dB reduction would increase.

![Figure 9.7 Spectrum Analysis of 65 MHz](image-url)
Another key aspect with a modulated clock is the modulation rate. This is the rate at which the modulation profile repeats itself. The modulation rate of the clock in Figure 9.6 is 27.86 kHz. There are several reasons why this low frequency modulation rate is important. If the modulation rate is below 20 kHz, it is possible to generate audible noise in the system. If the modulation rate is too high, in excess of 200 kHz, the effect of modulation might be defeated by the loop bandwidth of the filters used in downstream PLL's.

The Spectrum analyzer scan in Figure 9.7 is of the fundamental frequency of this 65.00 MHz clock. Most high-speed digital designs have problems complying with EMI regulations at harmonic frequencies rather than at the fundamental frequency. Since this 65.00 MHz clock, like most of the high-speed digital clocks, is a 50/50 duty cycle clock, the odd harmonics will contain higher energy levels than the even harmonics. Performing a Fourier transform on the relative energy level of digital clocks with duty cycles from 0 to 50% can prove this fact. Figures 9.8 through 9.11 show this analysis in graph form on the first, second, third and fourth harmonics of a frequency. These graphs plot relative energy against the duty cycle of the clock.

As can be seen in Figures 9.8 and 9.10, the odd harmonics have maximum energy levels at the 50% duty cycle and diminish as the duty cycle changes. However, as seen in Figures 9.9 and 9.11, the even harmonics have minimum energy levels when the duty cycle is exactly 50%.
Spread spectrum provides a clear difference in the peak energy of a 65 MHz digital clock at the fundamental frequency. However, the problems faced by many digital systems often occur at the higher harmonic frequencies.

**Harmonic Frequencies**

Modulation of the 65.00 MHz clock at a total spread of 3.87 % or 2.514 MHz has yielded a dB reduction of 6.48 dB. The harmonic frequency of a clock is an integer multiple of the fundamental frequency. The frequency of the 3rd harmonic is 3 x 65 MHz, which is 195 MHz, the 5th harmonic is 325 MHz, and the 7th harmonic is 455 MHz. Every frequency generated by the spread spectrum clock will also be a multiple meaning, for example that the bandwidth of the modulated clock at the 5th harmonic will be 2.514 MHz x 5 equaling 12.57 MHz. The bandwidth percent will not change but the actual frequency domain that energy is spread over will increase by a factor of five. This will cause the dB reduction of peak energy at the 5th harmonic to be much larger than at the fundamental frequency. The dB reduction does not multiply by the same factor since it’s a logarithmic scale. Figures 9.12 through 9.14 show the dB reductions of the 65 MHz clock at the higher harmonics. The reduction in energy ranges from 9.57 dB for the 3rd harmonic to 14.15 dB for the 9th harmonic.

![Figure 9.12 65-MHz 3rd Harmonic](image)
Figure 9.13 65-MHz 5th Harmonic

Figure 9.14 65-MHz 9th Harmonic
Each of these scans are taken from a very small four layer board with no other signal or circuit nodes to influence the results of these measurements. Actual results will vary based on the specifics of the design. However, these are representative of the type of data one should expect.

**Bandwidth Requirements**

As explained earlier, bandwidth is the amount of frequency modulation or spread applied to a particular clock. In terms of bandwidth percent spread, a typical application will use anywhere from 1 to 4%. A higher reduction in dB will result with a wider spread. However, at lower reference frequencies, a greater spread percentage is required to attain the same amount of reduction of a higher frequency reference. For example, compare a 10-MHz clock with a 28-MHz clock each with a 2% spread. The 28-MHz clock will be modulated over a 560-kHz range while the 10 MHz clock covers only a 200 kHz range. To see how this affects the amount of reduction, Figures 9.15 to 9.18 show the energy levels for each.

![Figure 9.15 10-MHz SSCG Profile](image)

Figure 9.15 10-MHz SSCG Profile
Figure 9.16 10-MHz, Small dB Reduction

Figure 9.17 28-MHz SSCG Profile

Figure 9.18 28-MHz Larger dB Reduction.
Figure 9.15 shows the modulation profile for the 10 MHz clock. From this view, the bandwidth of the spread can be verified to be 242.8 kHz. By dividing by the reference frequency, the bandwidth percent is 2.42%. Figure 9.16 shows the energy from the 10 MHz clock without spread spectrum and with spread spectrum active. Notice that there is only a small reduction in the energy peak.

Figure 9.17 shows the modulation profile for the 28 MHz clock. Again, from this view, the bandwidth of the spread can be verified to be 690.6 kHz and produces a percentage of 2.46%. Figure 9.18 shows the energy from the 28 MHz clock without Spread Spectrum and with spread spectrum active. This shows a larger reduction in energy.

Both references had Spread Spectrum applied using identical profiles. The percent of spread was also the same: 2.42% versus 2.46%. However, there was only a 0.73 dB reduction EMI on the 10 MHz while the 28 MHz realized a 3.74 dB reduction. The slower frequency needs to have a larger spread percentage to yield the same results.

**Down Spread and Center Spread**

Certain applications are frequency sensitive and are already operating at the maximum rate they can tolerate. A spread spectrum clock with a center-spread modulation will exceed the limits of this type of design. For this reason, spread spectrum also performs down spread. Down spread limits the modulated frequency to the reference clock. For example, a 40 MHz reference clock with an applied 2% down-spread spread spectrum signal will range from 39.2 MHz to 40 MHz.

One disadvantage of down-spread clocks is the effective center frequency of the clock is reduced by one-half of the bandwidth. With a spread spectrum clock providing a clean and symmetrical down-spread profile, the effective center frequency is calculated as:

\[
F_{\text{EFFCENT}} = F_{\text{MAX}} - \frac{1}{2} (F_{\text{MAX}} - F_{\text{MIN}})
\]

Applying this to the 40 MHz example above, the effective center frequency yields:

\[
F_{\text{EFFCENT}} = 40 - \frac{1}{2} (40 - 39.2) = 39.6 \text{ MHz}
\]

\[
F_{\text{MIN}} = 39.20 \text{ MHz}
\]

A center-spread spread spectrum clock is one that is modulating the frequency of the output clock symmetrically about the reference frequency. This means that for a clean modulation profile, the output frequency will increase and decrease the same amount above and below the reference frequency. The effective center frequency \(F_{\text{EFFCENT}}\) of a center-spread spread spectrum clock is the reference frequency itself. In Figure 9.19, the scan shows a center spread spectrum clock with a bandwidth of 2.42%. From the data in this scan, we see that the maximum frequency is 10.1289 MHz and the minimum frequency is 9.8861 MHz. Calculating for the effective center frequency yields the expected result.

\[
F_{\text{EFFCENT}} = 10.1289 \text{ MHz} - \frac{1}{2} (10.1289 - 9.8861) = 10.0075 \text{ MHz}
\]
Figure 9.19 10-MHz SSCG modulation profile

Modulation Rate

The modulation rate to obtain a low EMI clock profile is very important to the design engineer. This is the rate at which the clock sweeps from the minimum to maximum and back to minimum frequency. One complete cycle of the profile is the modulation rate. Devices that generate Spread Spectrum have different modulation rates and are specified in the data sheet for a particular manufacturer's part. This frequency can be as low as 20 kHz and as high as 150 kHz, depending on the design of the particular component.

Spread spectrum clocks often drive other components that contain a PLL. It is important that the downstream PLL has a loop bandwidth that allows the modulation frequency to pass. If the bandwidth is too low, the effect of spread spectrum on EMI will be diminished. Downstream PLLs should be specifically rated to accept spread spectrum clock signals. More information on multiple PLLs and loop filters can be found in Chapter 13, Cascading PLLs.

Tracking Spread Spectrum

Spread spectrum can be an effective method for reducing peak EMI and may easily be integrated into a clock design. However, in those designs where a PLL device may be "downstream", (driven by the spread spectrum clock), extra care should be taken. Since spread spectrum purposely modulates the clock, the downstream PLL must be able to track the frequency change to pass the modulated clock. If the PLL's bandwidth is too low, tracking skew will occur, adding jitter into the system. The designer needs to ensure that any downstream PLLs will track the modulated signal. If the downstream PLLs are part of a clock buffer circuit, Spread Aware™ type devices should be used. These devices are specifically designed and tested to operate with spread spectrum clock signals.
Conclusion

The generation and distribution of high-speed digital clocks can produce unwanted EMI. When the energy levels exceed allowable thresholds, the offending signals must be addressed. Both mechanical and electrical solutions exist, however a few circuit design changes may be the most cost effective. Clocks that are improperly terminated or have fast transition rates can lead to excessive emissions. To control these emissions, consider these guidelines:

- Properly terminate all clock signals
- Use slow rise time buffers when possible
- Consider filter capacitors on selected signals
- Use spread spectrum clocks for high-speed

By applying a few simple techniques, the radiated waves can be controlled. Proper termination, capacitor filters, and spread spectrum technology can solve many of these problems.
As the frequency of digital clocks increases, so does the difficulty in creating robust clock trees having quality signals. In previous chapters, we discussed the effects of fast edge rates and how to properly terminate traces. However, simulation is necessary to ensure signal integrity. There are two types of models used to simulate the signal characteristics commonly used in today's design environments: SPICE and IBIS. Each has the ability to simulate and analyze a circuit and give feedback on the performance of the design. Many of the available simulation tools include a scope trace viewer that allows designers to assess the performance of their design. The critical need is to know how a signal looks as it propagates down a trace and encounters impedance changes, terminations, stubs, and other devices that work to change the signal characteristics. Although some companies have dedicated signal integrity departments that specialize in simulation, the design engineer can gain critical information by understanding the effects certain aspects of the circuit have on signal quality. This provides confidence that the circuit will work when the first prototypes arrive. This chapter provides an overview of SPICE and IBIS simulation models and the information contained within.

**SPICE Models**

SPICE (Simulation Program with Integrated Circuit Emphasis) was created through research at the University of California at Berkeley (UCB) in the early 1970's. It was originally written in FORTRAN and was run on mainframe computers. SPICE 2G6 was the last Fortran version and later replaced with SPICE3 that is a C-language version of 2G6. SPICE3 is still maintained by the UCB as public domain software. Other versions of SPICE have been developed and marketed as proprietary commercial software including PSPICE, HSPICE, and AllSPICE. Most variants of SPICE have maintained similar syntax and algorithms to SPICE3, although they often add some proprietary functionality such as enhanced transistor models. SPICE simulators have been written for most platforms including UNIX workstations and Intel-based personal computers.

SPICE is a structural modeling language. It simulates the behavior of devices by running a numerical analysis on a structural model of the device. This structural model is created using basic components such as diodes, transistors, capacitors, resistors, and dependent...
current and voltage sources. The components are combined into a circuit by a file called the netlist. Figure 10.2 shows a SPICE netlist for the circuit shown in Figure 10.1. The numbers after the component names in the netlist are arbitrary numbers that correspond to the nodes of the circuit. Many implementations of SPICE contain schematic capture programs and will create these netlist files directly. SPICE uses many physical parameters to accurately model semiconductor devices such as diodes and transistors. These parameters are primarily related to the manufacturing process used to create the devices. Any parameters not specified in a model will be assigned default values.

When a SPICE simulation is run, the circuit is “solved” using current and voltage equations. Many aspects of the circuit can be examined such as transient analysis (behavior over time) and frequency sweep (behavior with respect to a sweeping input frequency). More complicated analysis can also be run such as a Monte Carlo analysis in which all components are varied per their individual tolerances. A set of curves is then plotted to show the behavior of the circuit as a function of the statistics of tolerance combinations. From this, you can see how the waveform might vary from device to device over a large number of test samples.

Models for devices such as integrated circuits can be put into a construction known as a sub-circuit that can then be distributed as a library file and called from SPICE to be included in a simulation. These sub-circuits comprise a netlist of standard components connected
and specified in such a way as to describe the whole device. A single sub-circuit for a device such as an operational amplifier can contain thousands of components in its netlist. Fortunately, the sub-circuit can be included as one component in your netlist.

XSPICE is an extension to Berkeley SPICE that was developed by the Georgia Tech Research Institute in 1992. XSPICE allows code-level modeling using the C-language to add new models directly to the SPICE core. This modeling capability expands the capability of SPICE simulators to include 12-state digital modeling, system level modeling, and the ability to include truly arbitrary behavior within models.

Signal integrity simulations can be performed with SPICE by modeling the buffers and by using transient analysis. The transmission line model is used to model board traces, cables, and other interconnection parts. SPICE includes both a lossy and lossless transmission line model. SPICE also includes coupled transmission line models for crosstalk simulation.

**IBIS Models**

Although SPICE is a comprehensive modeling and simulation environment, there are a few problems from a designer’s perspective. What may be the biggest drawback is the reluctance of integrated circuit manufacturers to release SPICE models for their devices. Due to its structural nature, a SPICE model must reveal many details of a device that manufacturers consider proprietary. The lack of readily available models often makes it difficult to use SPICE to validate circuit designs.

IBIS, (the Input/Output Buffer Information Specification), was created in 1993 by an industry-wide group of simulation experts to provide a standard method to exchange electrical component modeling data between integrated circuit manufacturers, simulation software vendors, and design engineers. This standard was developed and is maintained by regular “IBIS Open Forum” meetings that are held by the group. The latest approved version of the standard as of this writing is version 4.1 and is an official ANSI/EIA-656 and IEC 62014-1 standard. The official web site of the group is located at [http://www.eigroup.org/IBIS/](http://www.eigroup.org/IBIS/).

In contrast to the structural models of SPICE, an IBIS model is a behavioral model. It is not defined by the structure of the component to be modeled but rather by the way it behaves. IBIS models contain data based on how the component acts or reacts within a circuit as it is tested or operated. This data takes the form of current-voltage (I-V) tables, edge waveforms, lumped capacitance, inductance, and resistances. This data can come from actual measurements or full circuit simulation. Since the model is based on how the device behaves and not on mathematical formulas, non-linear effects can also be included in the model. Although IBIS includes many analog characteristics, it is really a digital simulation tool. The stimulation performed by an IBIS driver is a logic transition with rising edges, falling edges, or a series involving both edges.

The IBIS specification actually takes the form of a documented model file for a sample component. A model file is an ASCII text file that contains the models for all the drivers and
receivers for a single device or family of devices. Figure 10.3 shows the general input structure for an IBIS model. This includes the lumped package parasitics (capacitance, resistance, and inductance) and the Electric Static discharge (ESD) clamping structures and the capacitance of the input stage. The parasitics are specified as values while the ESD clamps are specified as I-V tables. Figure 10.4 shows the basic IBIS output structure. It contains the same basic structure as the input model with the addition of the behavior of the driver itself. These are specified as I-V tables and as voltage-time tables for rise-time/fall-time information.

![Figure 10.3 IBIS Input Structure](image)

![Figure 10.4 IBIS Output Structure](image)

There are three sets of possible values within an IBIS model. The values have different meanings depending on the context. In general, the “min.” values describe slow, weak behavior while the “max.” values describe fast, strong behavior. The “typ.” values refer to typical operating conditions. This behavior is determined by temperature and voltage extremes as well as manufacturing process tolerances. The conditions that cause a behavior may vary for different device families. For instance, a CMOS device will exhibit fast, strong behavior at low temperatures, while a bipolar device will exhibit this behavior at high temperatures. These conditions are also given in the model file.
Version 1.0 of the IBIS specification defines a baseline model to which all IBIS-generated models must be backward compatible. Version 1.1 is a refined revision and is generally accepted as the first major release. Version 1.1 defines the input and output structures given in Figures 10.3 and 10.4. Parameters GND_Clamp, Power_Clamp and Pullup/Pulldown information are provided in tables. Table 10.1 shows the first few lines of the Pulldown table for an IBIS model.

Table 10.1 IBIS Pulldown example

<table>
<thead>
<tr>
<th>Voltage</th>
<th>I (typ.)</th>
<th>I (min.)</th>
<th>I (max.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3.30</td>
<td>-2.09A</td>
<td>-2.12A</td>
<td>-2.06</td>
</tr>
<tr>
<td>-3.10</td>
<td>-1.93A</td>
<td>-1.96A</td>
<td>-1.90A</td>
</tr>
<tr>
<td>-2.90</td>
<td>-1.76A</td>
<td>-1.79A</td>
<td>-1.73A</td>
</tr>
<tr>
<td>-2.70</td>
<td>-1.59A</td>
<td>-1.63A</td>
<td>-1.56A</td>
</tr>
<tr>
<td>-2.50</td>
<td>-1.43A</td>
<td>-1.46A</td>
<td>-1.40A</td>
</tr>
<tr>
<td>-2.30</td>
<td>-1.26A</td>
<td>-1.30A</td>
<td>-1.23A</td>
</tr>
<tr>
<td>-2.10</td>
<td>-1.10A</td>
<td>-1.13A</td>
<td>-1.07A</td>
</tr>
<tr>
<td>-1.90</td>
<td>-0.93A</td>
<td>-0.97A</td>
<td>-0.90A</td>
</tr>
</tbody>
</table>

This is an I-V table that shows the current characteristics of the output stage of the device when the Pulldown is active and a specific voltage is applied to the output. The current in an I-V table is positive when the direction of the current is into the component. For the Pulldown and GND_Clamp, the table voltage is the actual voltage. Therefore, the first entry in Table 10.1 signifies a 2.09A current flow out of the device when a voltage of –3.3V is externally applied to the output pin. For the Pullup and Power_Clamp values, the table voltage is VCC relative meaning that Voutput = VCC – Vtable. For instance, an entry of –3.3V in a Pulldown table on a device with a VCC of 3.3V indicates an applied voltage of 6.6V on the output pin. All devices must be characterized for an input voltage ranging from –VCC to twice VCC although some table types are not required to span that entire range. This range is ideal for determining the effects of transient overshoots and undershoots.

Another important feature included in the first IBIS release is the Ramp specification. Ramp values show the rise and fall times for an output buffer. An example is shown in Table 10.2.

Table 10.2 IBIS Ramp Values Example

<table>
<thead>
<tr>
<th>Ramp Variable</th>
<th>Typ.</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>dV/dt_r</td>
<td>1.75/1.45n</td>
<td>1.56/1.38n</td>
<td>1.95/1.49n</td>
</tr>
<tr>
<td>dV/dt_f</td>
<td>1.78/1.53n</td>
<td>1.59/1.44n</td>
<td>1.98/1.59n</td>
</tr>
</tbody>
</table>

In the Ramp table, the dV/dt_r is the rising edge and the dV/dt_f is the falling edge of the signal. The number above the fraction is the voltage delta from 20% to 80% of the total
voltage swing. The lower number is the time this transition takes. As mentioned earlier, the **min** is the slow, weak behavior while the **max** is the fast, strong behavior. The resistive load for which this behavior is observed is also given in this section as the $R_{\text{load}}$ parameter. The Ramp values include the intrinsic component capacitance but do not take into account the package parasitics.

**IBIS Version Update**

Version 2.1 was the second major release and expanded the CMOS/TTL centric v1.1 to include other types of I/O, such as ECL, PECL, and other differential signals. There were many other extensions introduced in version 2.1, but one of the most important was the Rising_Waveform and Falling_Waveform specification. The Ramp values specify the rise and fall times for a buffer and implies a linear transition. The Rising_Waveform and Falling_Waveform specification describes the signal transitions using a voltage-time (V-T) table allowing an accurate representation of outputs with non-linear transitions such as those that employ transition control and waveform shaping. Figure 10.5 shows the test fixture for measurement of the V-T tables for the rising and falling waveforms. This level of detail allows a much more accurate description of the transitions than the simple fixture in the Ramp specification. Components are allowed, if not encouraged, to have multiple waveform sections with different test fixtures to provide the simulator more information for an accurate simulation.

Version 3.2 contained many enhancements to effectively model advanced buffer behavior such as bus hold and multi-stage buffers. A transit time specification was also added to model stored charge in the clamp diodes. A very important addition to version 3.2 was electronic board descriptions and package models. Both of these added controlled impedance trace descriptors to allow the modeling of traces within a package or on a board.

Version 4.1 contains many enhancements and corrections which are described in the readme.txt file on the IBIS website.

Electronic board descriptions (EBDs) are most commonly used for single in-line memory modules (SIMMs) and dual in-line memory modules (DIMMs) and are a description of board level interconnect. These descriptions are intended to reside in a separate .EBD file. Figure 10.6 shows the EBD path description that describes a board that is pictured in Figure 10.7. Several keywords are used to describe the circuit including the Fork construct.
to define branching of the traces, L for inductance, C for capacitance, and R for the trace impedance. The EBD can also define discrete and external components such as the 2 nH inductor shown in the example. A discrete R, L, or C component is specified by simply setting the length property to zero.

Figure 10.6 EBD Path Description

Figure 10.7 EBD Path Description Layout

The length and the L, R, and C values of the path descriptions are given in terms of unit length. The mathematics work out so that any unit of length can be used because the units fall out of the equations for both characteristic impedance $Z_0 \ (Z_0 = \sqrt{LC})$ and propagation delay $T_D \ (T_D = \sqrt{LC} \cdot \text{Length})$. The only firm requirement for unit length is that it is consistent with all the parameters for a given trace.

Package models allow more elaborate descriptions of a component package. Like EBD files, they include path descriptions to model traces, bond wires, and other methods of interconnection. More complicated behavior is also modeled such as pin-to-pin coupling. A package model can exist separately in a .PKG file or it can reside within an IBIS file between the [Define Package Model] and [End Package Model] keywords.
IBIS Simulation

IBIS simulation can show much about how a signal will behave as it propagates through a circuit. IBIS simulation can be performed during both pre-layout and post-layout design depending on the phase of your design cycle. Pre-layout simulation tools involve simulating a single net with varying configurations and allows for immediate feedback on the effects of those configurations. This “what-if” analysis can be done with factors such as trace length, trace impedance, placement of buffers, strength of buffers, and different terminations. Figure 10.8 shows part of the screen for Innoveda’s Hyperlynx LineSim pre-layout simulation tool. With this tool, you can graphically build a net using building blocks of drivers, receivers, transmission lines and terminations. “Scope Probes” can be placed on various net positions (selection shown by the arrow next to the buffer) to analyze the voltage behavior exhibited by a pulsing driver.

Figure 10.8 Hyperlynx LineSim Entry

Post-layout simulations use a PCB layout as the information for line characteristics instead of an entry tool. While you can usually simulate individual nets as in the pre-layout simulation, the real power in the post-layout simulation comes from the ability to simulate the entire board. You can simulate a whole board and generate a list of problems with overshoot, settling time, logic threshold, or skew. Constraints can be defined, and any net violating the constraints can be flagged. Traces in proximity to one another can be analyzed for crosstalk coupling effects between each other. Some simulators can even estimate EMI from the geometry of a trace and the signals on that trace.
IBIS Tips and Tricks

You will most likely encounter errors with IBIS models at some point when using IBIS simulators. If this happens, and the error messages from the simulator seem cryptic and nondescript, you may want to run the model through the IBIS "golden parser." The IBIS Open Forum manages this program for the purpose of validating IBIS models. The program accepts a file, parses through the model, and checks for any errors, inconsistencies, or discrepancies with the IBIS specification. The IBIS Open Forum maintains the source code for the parser, and executables for various operating systems are available and can be found on the IBIS Open Forum web site.

There is another important resource when working with an IBIS model and its specification. The IBIS Open Forum maintains an email reflector among its members specifically to address questions and issues from end users. One or more of the open forum experts usually answers questions quickly.

As mentioned earlier, IBIS simulation tools often extract the output impedance of the clock driver from the IBIS model. This value is needed to calculate the resistor value for series termination (see Chapter 7, Clock Termination). However, if you don’t have a simulator and if the output impedance value isn’t listed on the component data sheet, it can be derived by hand from the IBIS model.

There are several methods for extracting the output impedance. Because the value isn’t the same throughout the output voltage swing, the impedance is an approximation. Here are two of the most common ways to obtain the value. For either method, open the IBIS model file and locate the I-V tables for the output driver. The top of the file will list which model name represents each pin of the component. In the Pullup IV table, the current (I) is listed for a voltage (V) level in the rising edge of the clock driver.

The first method is a rough extraction, but relatively simple. Locate the voltage value for \( V_{CC}/2 \), which is a typical threshold value for an input device, and its corresponding current.
The table lists typical, minimum and maximum values. Calculate the output impedance by dividing the voltage by the current. For this number to be valid the driver must still be in its active region and the current value must not have reached saturation. If the current "flattens out," then the impedance calculation will not show the correct value.

The second and better method is to use two I-V points from the table and use the delta V and delta I values. From the Pullup table, locate a voltage (V) near Vcc/2, but where the current (I) is still in its active region. Locate a second lower voltage and its corresponding current (again, still in the active region). The output impedance is equal to the difference in the two voltages divided by the difference in their corresponding currents.

For example, Table 10.3 shows a portion of a Pullup I-V table for a clock buffer. Since the driver saturates at 1.3 volts for the typical case, use the value of 1.2 volts and its corresponding 95.71 mA current. A second point is arbitrarily chosen at 0.1 volts (but before the point at which the current starts to drop rapidly) and its current of 10.00 mA. The impedance is calculated with the following equation:

\[
\text{Output Impedance} = \frac{(V_2 - V_1)}{(I_2 - I_1)}
\]

\[
\text{Output Impedance} = \frac{(1.2 - 0.1)}{(0.09571 - 0.01000)}
\]

Output Impedance = 12 ohms

For this device, the estimated output impedance is 12 ohms. Notice that the output impedance will vary depending on the points chosen. However, this will provide an initial value from which to calculate a series termination resistor.

**Table 10.3 Clock Driver Pull-up I-V Table**

<table>
<thead>
<tr>
<th>Voltage</th>
<th>I(typ.)</th>
<th>I(min.)</th>
<th>I(max.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.10</td>
<td>-10.00 mA</td>
<td>-6.93 mA</td>
<td>-13.38 mA</td>
</tr>
<tr>
<td>0.20</td>
<td>-19.66 mA</td>
<td>-13.59 mA</td>
<td>-26.37 mA</td>
</tr>
<tr>
<td>0.30</td>
<td>-28.97 mA</td>
<td>-19.97 mA</td>
<td>-38.96 mA</td>
</tr>
<tr>
<td>0.40</td>
<td>-37.93 mA</td>
<td>-26.06 mA</td>
<td>-51.14 mA</td>
</tr>
<tr>
<td>0.50</td>
<td>-46.53 mA</td>
<td>-31.85 mA</td>
<td>-62.90 mA</td>
</tr>
<tr>
<td>0.60</td>
<td>-54.75 mA</td>
<td>-37.34 mA</td>
<td>-74.24 mA</td>
</tr>
<tr>
<td>0.70</td>
<td>-62.59 mA</td>
<td>-42.52 mA</td>
<td>-85.15 mA</td>
</tr>
<tr>
<td>0.80</td>
<td>-70.04 mA</td>
<td>-47.37 mA</td>
<td>-95.61 mA</td>
</tr>
<tr>
<td>0.90</td>
<td>-77.09 mA</td>
<td>-51.89 mA</td>
<td>-0.11A</td>
</tr>
<tr>
<td>1.00</td>
<td>-83.73 mA</td>
<td>-56.06 mA</td>
<td>-0.12A</td>
</tr>
<tr>
<td>1.10</td>
<td>-89.94 mA</td>
<td>-59.88 mA</td>
<td>-0.12A</td>
</tr>
<tr>
<td>1.20</td>
<td>-95.71 mA</td>
<td>-63.33 mA</td>
<td>-0.13A</td>
</tr>
<tr>
<td>1.30</td>
<td>-0.10A</td>
<td>-66.41 mA</td>
<td>-0.14A</td>
</tr>
<tr>
<td>1.40</td>
<td>-0.11A</td>
<td>-69.10 mA</td>
<td>-0.15A</td>
</tr>
<tr>
<td>1.50</td>
<td>-0.11A</td>
<td>-71.39 mA</td>
<td>-0.16A</td>
</tr>
<tr>
<td>1.60</td>
<td>-0.11A</td>
<td>-73.28 mA</td>
<td>-0.16A</td>
</tr>
<tr>
<td>1.70</td>
<td>-0.12A</td>
<td>-74.74 mA</td>
<td>-0.17A</td>
</tr>
<tr>
<td>1.80</td>
<td>-0.12A</td>
<td>-75.81 mA</td>
<td>-0.17A</td>
</tr>
<tr>
<td>1.90</td>
<td>-0.12A</td>
<td>-76.74 mA</td>
<td>-0.18A</td>
</tr>
</tbody>
</table>
IBIS vs. SPICE Simulation

The debate on whether IBIS or SPICE simulation is better has been a fierce one and may even be a wasted argument. Most designers who simulate circuits will tell you that they use both methods. Each method has its own strengths and weaknesses.

Because SPICE is a numerical structural simulation, a SPICE model can be made to be more accurate than an IBIS model. Although extensions are being added to IBIS to allow a greater range of expression, it still does not have the flexibility that SPICE has to add components to model the simulation closer to reality. IBIS is limited to the constructions of the IBIS specification. Also, not all simulators implement all the extensions and therefore some simulators can produce different results.

A second advantage of SPICE is the greater number of simulation tools. This is because SPICE has been available longer than IBIS and it is more widely known by design engineers. Most engineering curricula have been using SPICE in circuits courses for many years.

Third, SPICE is the only choice when working with analog circuits involving functions such as amplification; IBIS was created to model digital buffers only. However, many companies require Non-Disclosure agreements before the SPICE models can be released as they detail the inner working of the device.

The main advantage of IBIS is its simplicity and convenience. In general, an IBIS model is ready to use in any simulator with no modifications. Because an IBIS simulator is often made for that purpose only, the learning curve typically associated with the process and settings of SPICE simulators is greatly diminished.

A second advantage of IBIS is in the simulation time of large designs. Running a worst-case overshoot analysis on an entire complex board can be performed relatively quickly with IBIS models. Some IBIS tools allow you to look at multiple boards that are connected together. Generally, a complete board simulation would take much longer with SPICE. SPICE has to convert every component in a model to a mathematical relationship and then run iterative calculations to determine a solution. In an IBIS behavioral model, no such iterations are necessary.

A third advantage of IBIS is that it does not have convergence problems with certain combinations of models. These complicated problems, found in SPICE, have to do with the underlying mathematics of the simulation and are often tricky to overcome.

There are also ways to get some of the advantages from each simulation method. There are converters available that convert IBIS models to SPICE models and vice versa. These converters aren’t always easy to use but they can be handy as a starting point when you must use SPICE to customize a simulation and only have IBIS models for a device. There are also some SPICE simulators that will read in IBIS models and convert them behind the scenes to integrate with the SPICE simulation. For example, the popular PSPICE program from Cadence has a built-in IBIS translator for V1.1 models.
Conclusion

Most companies that offer timing technology products also have IBIS models available for their devices. IBIS simulation is easy to perform and can provide much insight into the operation of your design. This simulation is very powerful and allows for the many different types of board stack-ups, components, connectors, and discretes. SPICE modeling also provides an excellent environment and includes functional as well as signal simulation. However, this simulation can be more complex and the models for each component may be difficult to obtain. Using either tool, you can also quickly gain insight into signal integrity problems that arise both in existing circuits and in current designs.
Up to this point, we have discussed proper clock design techniques and have shown some simulations of the signals. But when actual hardware is analyzed, we need to understand exactly what limitations exist with the chosen test equipment. In many situations, engineers will probe high-speed clocks and the waveform will appear very different than expected. It is often thought that the design is not working properly but in many cases, the engineer exceeded the bandwidth of the equipment.

In the past, oscilloscopes were primarily analog in design with a couple of input channels. They also used one inch round binding posts as the input connection, and twisted pair wires were needed to prevent outside interference. The bandwidths of these scopes were also quite limited. Oscilloscopes have progressed dramatically over the years with an abundance of signal capture features and storage capabilities added. They have also increased their bandwidths to meet the needs of probing high-speed digital clocks. In this chapter, we will examine the techniques needed to properly probe a high-speed clock. We will also answer the fundamental question: “Will my 1 GHz oscilloscope correctly show my 100 MHz clock signal?”

There are two areas of importance when probing high-speed clocks:

1. Oscilloscope Bandwidth
2. Probe and Connection

We will examine both of these and how each affects the resulting waveform display.

**Oscilloscope Bandwidth**

Of key importance, obviously, is the instrument being used. To properly measure a high-speed clock, the bandwidth of the oscilloscope must be wide enough to accept the signal accurately. With the use of digital storage scopes, we must also ensure that the sample rate and sample size is equally sufficient.

The scope bandwidth is defined as the frequency above which a sine wave’s amplitude is degraded by 3 dB or more. Typically a scope (the frame or mainframe) has a bandwidth,
and each plug-in or channel has its own bandwidth. This is most important when channels differ or when plug-in technology can alter the capability of the scope.

In a signal such as a clock, there are two events that need to be considered when measuring. There is the continuous, repetitive stream of waves and there is the single-shot pulse due to the rising and falling edges. There are times when each need to be measured, and their respective bandwidth requirements can differ. To view repetitive waves, the bandwidth of the scope needs to be at least three times the highest frequency measured. This ensures the complete waveform will be captured without distortion.

The single-shot pulse bandwidth is probably the most important when considering high-speed signals as this gives the limits for the clock edge rates. We need to ensure the scope will accurately capture the fast rise and fall times of our signal. Design engineers use time to measure rising and falling edges; however, oscilloscopes are specified in terms of bandwidth. Therefore, a conversion equation is required to compare the equipment to the signal. As mentioned earlier, the scope bandwidth is measure to the –3 dB point. To convert to an approximate rise time, we use the following equation:

\[ t = 0.338/F_{\text{3dB}} \]

This allows us to relate the scope's response to the signal that we need to measure. Some oscilloscopes use a slightly different method for measuring bandwidth and are listed as RMS bandwidth. To convert this measurement into time, use this equation as an approximation:

\[ t = 0.361/F_{\text{RMS}} \]

This suggests that a 1 GHz scope, at best, can accurately view a 340 ps rise time in a clock signal. But, before we can conclusively say this is our scope's limit, we need to analyze the sample rate and available memory.

**Sample Rate**

Since many of the modern oscilloscopes are digital and sample the incoming signal, we need to consider how fast they can capture the pulse. Sample rate is simply the number of measured points per second within a waveform.

When a digital storage scope samples the incoming waveform, a dot is used to represent its voltage and time. The faster the sample rate, the more dots are used to represent the signal. Interpolation is used to curve fit to the signal by connecting the dots and showing a contiguous waveform on the display. The sample rate, as compared to the bandwidth, should be at least four to one. An 8 or 16 to 1 ratio will provide excellent waveforms and, with values greater than 10, minimal interpolation is necessary.

After the signal is sampled, memory is required to store the dots. The amount should be large enough to allow capture of the complete signal at the given sample rate. When storage is limited, the amount of time viewed on the display can be affected.
Probe and Connection

An oscilloscope is only one element that affects the quality of the displayed signal. The probes and connecting cables are easy to forget. However, they are as significant as the scope. Further, they offer the most chance to provide error into the test system.

Similar to the scope, the probe also has a specified bandwidth. The signal that we wish to measure must first pass through the probe before it reaches the scope. This means the bandwidth of the probes must be considered in the same way as the scope. We can convert the edge rate response of the probe using the same formulas that we specified earlier for the scope.

To determine if the signal we are to measure can be properly displayed, a simple formula can be used as an approximation:

\[ t_{\text{DISPLAY}} = \sqrt{t_{\text{SIGNAL}}^2 + t_{\text{PROBE}}^2 + t_{\text{SCOPE}}^2} \]

**Figure 11.1 Scope Connections**

In this formula, \( t_{\text{SIGNAL}} \) is the fastest rise (or fall) time of the signal we need to measure. \( t_{\text{PROBE}} \) is the bandwidth of the probe converted into its rise time equivalent and \( t_{\text{SCOPE}} \) is the bandwidth of the scope converted into its rise time equivalent.

Knowing these relationships, let’s do an example. Suppose we have a 100-MHz signal with 1 ns edge rates driven by a clock buffer. We use an oscilloscope with a 400-MHz bandwidth and a probe with a 400-MHz bandwidth. To the casual observer, it appears that the test equipment will display the signal. Or will it?

First, we need to convert the bandwidths into equivalent edge rates:

\[ t_{\text{SCOPE}} = \frac{0.338}{400} = 840 \, \text{ps} \]

\[ t_{\text{PROBE}} = \frac{0.338}{400} = 840 \, \text{ps} \]
Now we can calculate what the display will show as the rise time of our signal:

\[ t_{\text{DISPLAY}} = (1^2 \text{ ns} + 0.84^2 \text{ ns} + 0.84^2 \text{ ns})^{\frac{1}{2}} \]

\[ t_{\text{DISPLAY}} = 1.5 \text{ ns} \]

The display will show the rise time of the 1 ns signal as 1.5 ns! Notice in these equations, the frequency of the signal does not matter. It is the transition time of the signal that is of importance. Furthermore, even though the scope and the probes individually could handle the signal edge rate, they did not when coupled together. Higher bandwidth scope and probes are needed to properly measure this signal.

**Ground Leads**

In addition to selecting the proper bandwidth of the equipment, we also need to be concerned with the probe attachment to the signal. Many probes have a plastic neck with a ground connector at the base. A common method for attaching a ground connector is to clip a 4-inch wire to the probe. An alligator clip at the other end usually connects to a ground connector somewhere on the board. Little consideration is given to the effects of this wire.

But, the ground wire plays a key role in accurately displaying the clock signal. When the signal is being measured, current flows into the probe, through the wire, and back into the board ground plane. This can be viewed as a circuit represented by an R, C and L, as shown in Figure 11.2.

![Figure 11.2 Scope Probe Circuit](image)

There are also several types of probes available, including passive and active (FET) voltage varieties. Probes specify their input impedance R and can range from less than 20K to over 10M ohms. Input capacitance C can be as high as 15 pF and down to less than 1 pF loading. Oscilloscope manufacturers typically list in their documentation the application for which a particular probe was designed. The inductance L is a function of the wire ground loop. By making a few assumptions, we can calculate an LC time constant that will show the limit of the signal response on the probe.

If we assume the ground wire is rectangular in nature, we can use the standard formula for calculating inductance:

\[ L (\text{nH}) = 10.16 \times [l^* \ln\left(\frac{2 \times w}{d}\right) + w \times \ln\left(\frac{2 \times l}{d}\right)] \]
Where:  
\[ L = \text{resulting inductance of the wire (nH)} \]
\[ l = \text{length of the wire (inches)} \]
\[ w = \text{the width of the wire (inches)} \]
\[ d = \text{diameter of the wire (inches)} \]
\[ \ln = \text{Natural Log} \]

We can calculate the inductance of the ground loop by applying the physical dimensions of the wire. For example, a wire with a diameter of 0.02 inches, a ground wire with the length of 3 inches, and a ground connection on the board about 0.5 inch (width) from the signal, yields approximately 150 nH. Figure 11.3 shows a wire rectangle and its dimensions.

**Figure 11.3 Wire Loop Inductance**

By assuming the ground on the probe to be similar to the rectangular wire, the same equation can be used to estimate the inductance of the ground strap, as shown in Figure 11.4.

**Figure 11.4 Probe Wire Loop Inductance**

By using this value of \( L \) and the \( C \) specified for the probe, we can estimate the rise time limit of the ground wire on our signal. If we choose a typical 5 pF probe, we can estimate the rise time with the following equation:

\[
 t_{\text{GROUND}} = 3.4 \times (L \times C)^{\frac{1}{2}}
\]

\[
 t_{\text{GROUND}} = 3.4 \times (150 \times 0.005)^{\frac{1}{2}} = 2.9 \text{ ns}
\]

This yields a value for \( t_{\text{GROUND}} \) to be 2.9 ns! This is an incredibly large impact on our ability to measure the signal. We now know that our original \( t_{\text{DISPLAY}} \) equation needs to include the ground loop parameter:

\[
 t_{\text{DISPLAY}} = \sqrt{t_{\text{SIGNAL}}^2 + t_{\text{PROBE}}^2 + t_{\text{SCOPE}}^2 + t_{\text{GROUND}}^2}
\]
If we apply the $t_{\text{GROUND}}$ parameter to the example we used earlier, the error on the display of the scope is even larger. Using the 100-MHz clock with the 1 ns rise time in the previous example and the scope and probe just described, we can estimate what the display will show:

$$t_{\text{DISPLAY}} = (12 \text{ ns} + 0.842 \text{ ns} + 0.842 \text{ ns} + 2.92 \text{ ns})^{\frac{1}{2}}$$

$$t_{\text{DISPLAY}} = 3.3 \text{ ns}$$

The 3.3 ns rise time on the scope is very different from the 1-ns clock on the board. When probing high-speed clocks, a key goal is to minimize the $t_{\text{GROUND}}$ component of this equation. By doing so, we maximize the use of the bandwidth of the scope and probe. As seen in the previous equations, the two attributes that cause the delay are the capacitance of the probe and the inductance of the ground wire. Since $C$ is a fixed value associated with the probe, select a probe with the smallest input capacitance as possible. Inductance is a function of the geometric size of the ground wire. As seen in the example, a long wire is insufficient to provide the necessary grounding for high-speed clocks. Changing the shape of the ground strap minimizes the inductance. By selecting the shortest and widest wire for the ground, inductance is minimized. Many probes today have a small spring wire that is short, but very thin. This method generally works, however it is not optimal. For very high-speed clocks, use a short, flat blade for the ground connection. The ground connection needs to be close to the signal being probed. There are test connectors available to which signals can be wired and have collars connected to ground. This is an ideal way to attach the probe to the signal as the probe tip slides into the collar. Care should be taken that the connector doesn’t add any stubs to the trace.

**Conclusion**

Having the proper test equipment is essential when probing high-speed clocks. Engineers often take for granted the waveforms on an oscilloscope as correct, however they can be easily misled. To view an accurate picture of the signal, the proper bandwidth of the scope and probe must be used as must be a low inductance ground connection, which is equally important. By following a few simple rules, the scope can display the real clock signal. Now you know the answer to the question, “Will my 1 GHz oscilloscope correctly show my 100-MHz clock signal?”
When people speak of clock generators, they often are referring to a piece of test equipment that will allow the generation of a particular frequency and waveform to stimulate a device under test. This general idea can be taken a step further and used as a clock source embedded in a design. Many designers are familiar with oscillators that provide a fixed clock, however, a variety of frequencies can be produced using a clock generator. This chapter will discuss this function and how it applies to an integrated circuit (IC) that can perform that same function on a PCB within your system.

Before discussing the aspects of clock generators, resonators and crystals should be examined, as they are the basic building blocks. Typically when someone talks about a resonator, they are referring to a ceramic resonator. When someone discusses a crystal, they are talking about a crystal resonator. Both of these devices provide similar functions but with different base substances and differing levels of accuracy.

**Resonators and Crystals**

Ceramic resonators (Resonators) are piezoelectric ceramic devices that are designed to oscillate at a particular frequency. The material from which they are made has the resonant property induced during the manufacturing process. Since this is under the control of manufacturing tolerances, and the quality factor of this material is far less than quartz, resonators cannot match the frequency stability of a crystal resonator (Crystal). This quality factor, also referred to as "Q," is the ratio of energy stored to energy dissipated, which is also defined as the ratio of reactance to series resistance at the resonant frequency. Resonators are typically used in cost sensitive, lower performance applications. They tend to cost half that of a crystal and are available in a smaller physical size. The drawbacks to resonators are that they lack the frequency and temperature stability of their crystal counterparts.

Crystals are similar to resonators but use a piece of cut quartz for their resonant element. Quartz has a naturally high “Q” value allowing it to maintain high accuracy and frequency stability over its entire operating range of temperature and voltage. The remainder of the discussion will use the crystal as the clock source for a clock generator.
When someone refers to a crystal, they are typically not referring to a piece of quartz alone, but rather one that has been packaged and tuned to resonate at a given frequency. A crystal, when properly stimulated, will act as a narrow band filter at its given frequency. Typically, a crystal is connected to a device that has an excitation and detection circuit. The gain amplifier in this circuit presents broadband noise to the crystal which filters out all but the energy present in the crystal bandwidth for the given crystal center frequency. This energy is allowed to grow through amplification and a portion thereof is fed back through the crystal. A square wave is created from the sinusoid input as it reaches its limits when forced through a high-gain system. There are many such devices that contain this type of excitation circuitry including Digital Signal Processors, microcontrollers, and microprocessors.

Crystals come in two basic types: series and parallel resonant. These two types are physically identical but vary in the manner in which they are tuned to their stated frequency. A parallel resonant crystal is adjusted to a given frequency with a specified value of load capacitance, whereas a series resonant crystal requires no load capacitance. The load capacitance is what the crystal “sees” at its pins to the excitation circuit. If a series resonant crystal is configured with a load capacitance, it will operate at a frequency higher than its stated value. Likewise, if a parallel resonant crystal is used without any load capacitance it will operate at a frequency lower than its stated value. Most excitation circuits will have a rated load input capacitance.

The parallel resonant configuration is the most common and is shown in Figure 12.1.

![Parallel Resonant Crystal Circuit](image)

**Figure 12.1 Parallel Resonant Crystal Circuit**

To calculate the load capacitors needed, the following formula can be used:

\[
C_L = \frac{(C1 \times C2)}{(C1 + C2)} + C_{STRAY}
\]

\(C_{STRAY}\) is the stray capacitance in the circuit and is typically between 2 and 5 pF. If using a parallel resonant crystal with a \(C_L\) of 20 pF, \(C1\) and \(C2\) will be approximately 27–33 pF, each depending on the level of stray capacitance. If after measuring the output of the circuit it is found that the frequency is too low, the values of \(C1\) and \(C2\) should be decreased so that the oscillation frequency increases. The opposite would be true if the frequency was too high. To make this whole process much easier, many clock generators...
are optimized for a particular crystal load. Additionally, some clock generators allow the user to change the load capacitance, which helps eliminate the variance of the stray capacitance found on the board from the above equation.

**Oscillators**

Taking the building block approach one step further, the oscillator is the next piece. An oscillator is a device that has the crystal, excitation, and detection circuitry in one single package. In this case, the output of the oscillator is a square wave that can be used by other devices that require a clock directly. This is typically the easiest clock source to use. The drawback is that a separate oscillator must be purchased for each required frequency because the frequencies are fixed.

**Programmable Oscillators**

The Programmable Oscillator is a hybrid device that consists of a Crystal and a clock generator in the same package. It performs a function similar to a dedicated oscillator. It offers the advantage of being "configurable," allowing the user to select a given frequency after, rather than before the purchase. The possible drawback is that the phase noise of this device may be higher than the dedicated oscillator, which may be of concern in the design in which it is used. The concept of programmability, which allows a variety of frequencies to be selected, will be discussed next.

**Clock Generators**

A clock generator, also known as a frequency timing generator (FTG) or frequency synthesizer, is a device that uses a crystal, an oscillator or a given clock as an input, and can create a wide range of frequencies as an output. As an example: a 16-MHz crystal can be used as an input into the FTG and in turn, generates an output of 14.31818 MHz. Even though the output frequency seems to have little in common with the input frequency, it can typically be created in an FTG with no deviation in frequency.

The benefit of an FTG, as discussed above, is allowing the user to configure the part to have numerous output configurations. This configuration can be done just prior to installing it on the board, or after, allowing the utmost in versatility. All of these benefits do however come with several drawbacks, the most significant being increased phase noise or jitter. This is mostly an issue in communications standards, such as SONET, FibreChannel and Gigabit Ethernet, whose extremely high speeds dictate an extremely pure clock source. An additional item to note is whether the output must be in phase with the input frequency. Some FTG's offer phase alignment for particular configurations only, while others do not offer this at all.

An FTG is made up of a divider (Q), phase detector (PD), charge pump (CP), voltage controlled oscillator (VCO), multiplier (P) and a post divider (N), as shown in Figure 12.2. This is essentially a PLL, as was discussed in Chapter 2, with the P, Q, and N dividers added.
The Q counter takes the reference clock input and divides it prior to sending it into one of the phase detector inputs. The P counter takes the output of the VCO and divides it down before it is given as the feedback (FB) into the second phase detector input (effectively causing multiplication). The Phase Detector determines which input signal, the reference/Q (REF) or the feedback/P (FB), occurs first. It sends an up or down signal to the charge pump to align the two signals. If the feedback is slow, it will send an “up” to the charge pump. The charge pump monitors the Phase Detector output and either increases or decreases the reference voltage that it supplies to the VCO. The VCO looks at the control voltage from the charge pump and creates the corresponding frequency as its output. Increasing or decreasing the control voltage will increase or decrease the VCO frequency. The N counter (post divider) divides the output of the VCO prior to its output of the device. By adjusting the P, Q and N values, the output is created from the input frequency. Both Q and N are clock dividers (pre and post-divide, respectively) while P is the multiplication factor.

In the simplest case, assume that the P, Q and N dividers are all set to one. This device then functions as a zero delay buffer (Figure 12.3). The reference clock drives the REF input of the phase detector. The phase detector follows the rising edge of the REF signal and causes the charge pump to speed up or slow down the VCO so that the FB signal at its other input matches every REF rising edge with a corresponding FB rising edge. Since this is actually the VCO output, the VCO runs at the same frequency as the reference input. By matching the FB and output buffer path lengths within the device, the output of the device has zero skew (propagation delay) relative to its input.

A more complex example is to generate the 14.31818-MHz output from a 16-MHz input. The values for P, Q and N are not one in this case and need to be calculated to get the desired result while not exceeding the limits of the VCO. The output frequency is a function...
of the VCO divided by N. The VCO frequency is a function of the input clock and the P and Q ratio. These two relationships can be expressed with the following equations:

Output frequency = VCO frequency/N

Input frequency/Q = VCO frequency/P

Combining them, the output frequency and ratios can be expressed with the following equations:

Output frequency = (Input Frequency/Q) * (P/N)

Output frequency/Input Frequency = P/(N*Q)

For the example of a 16-MHz input generating a 14.318182 output, there is the following substitution:

14.31818/16.000000 = P/(N * Q)

0.89488652 = P/(N * Q)

By using the values of Q set to 44, P equal to 315 and N equal to 8, the phase detector has an input frequency of 45.45454 kHz and the VCO has a frequency of 114.545455 MHz. Both of these frequencies are within their operating limits of the particular device chosen for this example. The methods of how to calculate these numbers by hand as well as through software algorithms are discussed next.

**Calculating P, Q and Post Dividers**

When calculating the values that are needed for the P and Q counters, there are several methods that can be used. The “by hand” approach is to factor the input and output frequencies into a multiple of primes:

16 x 10^6 = 2^{10} * 5^{6}

14.31818 x 10^6 = (2^{5} * 3^{2} * 5^{7} * 7)/11

Therefore, 16 MHz must be multiplied by (315/352) to get 14.31818 as an output.

To do this within a clock generator, the minimum and maximum requirements of the VCO as well as the phase detector must be maintained. To further restrict the possible values of P, Q and N, the clock generator must have enough bits of resolution in the P, Q and N counters to achieve the divide values needed.

By using a tool developed for this purpose, it is much easier to calculate what counter values are needed as well as what values are within the bounds of the phase detector and VCO. One such program is called CyClocksRT™, which can be found at http://www.cypress.com in the Timing Technology section. The user has a simple GUI
interface specifying the input frequency, the output frequencies, and some internal dividers. The prior example using this GUI is shown in Figure 12.4.

Figure 12.4 CyClocksRT GUI

**Parts Per Million (PPM)**

When the ratio of P, Q and N do not allow the output to operate at the desired frequency, the deviation is expressed in terms of PPM, or Parts per Million. In the case of a 16-MHz clock with an error of 5 PPM, the actual error would be 5 (parts) * 16 (millions of Hz) = 80 Hz. The 16-MHz frequency with a 5-PPM error rate is therefore 16.000080 MHz.

As seen, a single digit PPM is an extremely small amount of inaccuracy. Many clock inputs can tolerate a range of 50 to 200 PPM and still operate properly. Often, clock generators with high resolution P, Q and N counters, can provide single digit PPM levels with virtually any input/output combination.

The PPM that a clock generator specifies assumes the input has 0 PPM. If the input clock (oscillator, crystal, or clock source) varies in frequency, the output of the clock generator will also vary. For example, a 25 MHz 0 PPM clock driving a clock generator may produce a 48 MHz output with 0 PPM. If the 25 MHz clock has a 25 PPM error rate, the 48 MHz output will also exhibit a 25 PPM deviation.
Fractional N

Some generators incorporate a feature called Fractional N. Fractional N is a means of bouncing between multiple P/Q values to achieve a lower PPM or reduce the effective jitter when high P & Q values are used. As an example, choose an output frequency that cannot be exactly met and the closest result is 5 PPM. To get to the desired output frequency with 0 PPM, Fractional N can bounce between two different P/Q ratios, one at +5 PPM for 66.667% of the time and the other at -10 PPM for 33.333% of the time. The average, which is filtered by the FTG loop bandwidth to prevent jitter, is the desired frequency at 0 PPM. Using a Fractional N technique requires the FTG to have special circuitry to allow the “jumping” between various P/Q ratios. This is typically done by providing a table of multiple P/Q values and using a counter or state machine to cycle through several of them at a programmable rate. This same technique can be applied to a spread spectrum output.

Spread spectrum is the varying of an output clock within a given band. An example would be to have a 100-MHz clock that varies from 99.5 to 100.5 MHz at a 40 kHz rate. This can be accomplished using the same basic hardware as that used for Fractional N. The benefits and uses of this technique are covered in much greater detail in Chapter 9.

Part Configuration

Once the P, Q and N values are selected, a part needs to be programmed. Depending on the technology of the device, this can be accomplished in several different ways.

Mask programmed parts are hard coded at the die level. The appropriate connections are shorted or left open during the mask process of the wafer manufacturing process. This type of configuration may yield a very high number of die per wafer allowing the cost per chip to be very low. The downside of this technology is that once the “mask” is made, the device can only have the given configuration.

EPROM-based designs allow for more flexibility in the configuration process. They use an EPROM cell that is left “blank” at the wafer manufacturing process. This allows the device to be programmed with the desired configuration. Although this is a one time programmable (OTP) device, it still offers much greater flexibility than the mask-based part.

To get around the “program once” limitation, some FTG devices are SRAM-based. This allows the devices to be configured on board through a serial programming stream. The benefit is that the device can be configured an unlimited number of times, allowing for maximum flexibility of output frequencies. However, the power-up default is set to an initial configuration based on the original design criteria of the part and is fixed.

The latest technology is a hybrid approach. It uses a FLASH cell to store the configuration data thus allowing the user to select the power-up default. This same part also has SRAM-based registers that allow the user to re-configure the part an unlimited number of times while it is on the board. This hybrid technology offers the most flexibility.
Multiple PLLs
Multiple PLLs in a single package offer many benefits as well as a few detractors. The benefits of multiple PLLs within a part allow the user to generate multiple, seemingly unrelated, frequencies within a single device. The inherent problem with multiple PLLs is that each PLL may interact with the others, allowing additional phase noise (jitter) to be coupled into the VCO and out onto its respective output. This is typically not an issue for pure digital designs where phase noise is of less concern. The areas that are typically affected are items such as A/D devices and down-stream PLL-based devices.

Conclusion
With the multitude of frequencies required for many synchronous systems, timing generators can offer a flexible solution. They can generate a variety of clocks from a single input source. A large benefit is the ability to change the frequencies as the design progresses without changing the physical part.
Engineers are often faced with questions surrounding the use of cascaded phase-locked loops. Can I cascade two PLLs to generate a frequency and distribute it to my ASIC or microprocessor? How many PLLs can I cascade before I run into problems? What kind of problems will I run into when I put several PLLs in series? Unfortunately, the answers to these questions can be very complicated and they are highly dependent on the design of the PLLs chosen. To make matters even more difficult, digital engineers who typically are the ones implementing the clock circuits are rarely exposed to the types of analog interactions these components exhibit. This chapter will address these issues but will remain practical by avoiding complex equations and mathematical analysis. The focus will be on demonstrating the interaction of PLLs, understanding key terms, and assessing the risks of cascading PLLs.

**A Cascaded PLL**

A cascaded PLL is simply two or more phase-locked loops in series that produces an output clock. A simplified diagram of a cascaded PLL design is shown in Figure 13.1 with two PLLs where one PLL performs a frequency multiplication and the other performs zero-delay clock distribution of the multiplied clock.

PLLs of the past were normally found as standalone components. In the 1990’s, in a quest for faster signal processing, PLLs started to become embedded in a variety of devices. They are now used in clock generators, microprocessors, microcontrollers, ASICs, digital signal processors, oscillators (some programmable oscillators use embedded PLLs), and many other components. PLLs dramatically improve the data valid windows by adjusting clock skew and therefore accommodate the needs of short cycle time clocks.
They also allow slower frequencies to be multiplied to achieve much faster clock rates. This has led to the huge popularity of PLLs in high-speed digital designs. However, when adding a PLL into a design, care must be taken to ensure tolerances are not violated at a system level. When more than one PLL is used, the interaction of the PLLs can become complex and the analysis is at times difficult. PLLs have a closed-loop response that has an impact on the jitter transfer characteristics. They can actually amplify noise or jitter both can be source both internally and externally. The largest gain experienced in the frequency region is commonly called jitter peaking. When cascading like PLLs, jitter peaking grows exponentially from stage to stage. The effects of cascaded PLLs and jitter peaking on clock performance may range from not having any effect on system operation to rendering the system non-functional.

**Acquisition and Tracking**

PLLs are characterized as having two states: phase-locked or acquiring lock. Tracking is synonymous with the locked condition and simply describes the extent to which the loop can follow variations in the input clock frequency. PLLs operate on the phase of signals and therefore are susceptible to changes in the clock edges on the inputs. The transient response of a PLL is generally a very complex, non-linear process. In general terms, the PLL will follow the presence of a slowly occurring signal at the input and does not react to rapidly occurring transitions (frequencies outside of the PLL’s loop bandwidth). Understanding how the noise spectrum propagates through the PLL is addressed in our discussions of jitter accumulation, jitter peaking, tracking skew, and phase noise.

Acquisition is the process the PLL undergoes to align the output to the input phase. Although most designs have wide margins for acquisition time, there are situations where this period is important. When a PLL operates near the point of instability, the PLL lock time can become excessive. It should also be noted that noise associated with the voltage supply also has an impact on both the acquisition and tracking of the PLL.

**Jitter Accumulation**

The accumulation of jitter is typically measured as long-term jitter (refer to Chapter 4 for a complete discussion on jitter). A consequence of cascading PLLs is jitter accumulation. Every phase-locked loop has the inherent characteristic of transferring a spectrum of noise from the input to the output. Unfortunately, it also has the inherent characteristic of adding noise. These two characteristics are referred to as Jitter Transfer and Jitter Generation, respectively. Whether a PLL adds jitter or removes jitter from a clocking path is dependent upon whether the device eliminates more noise than it generates.

Devices that attenuate jitter have loop filtering configurations that eliminate unwanted clock jitter from the input without creating excess jitter in the process. These devices are typically marketed as low-noise or low-jitter PLLs. Their primary function is to eliminate high frequency noise. However, they tend to create unwanted noise at lower frequencies. The characteristic measurement that is the most meaningful is the plot of the closed loop gain for a PLL. This is referred to as the jitter transfer curve.
In Figure 13.2, a PLL has been plotted for its jitter transfer characteristic. The result is a measurement of the amount of jitter transferred from the input to the output along with the jitter generated in the process. PLLs eliminate high frequency jitter, as demonstrated by the roll-off beyond 2 MHz. For this particular PLL, the loop bandwidth is approximately 2.5 MHz where the output is -3 dB. At frequencies within the loop bandwidth the jitter is magnified from 0 dB to nearly 2 dB. The highest point of magnification is the jitter peak.

Figure 13.2 Jitter Transfer Curve

Figure 13.3 shows a set of histograms with jitter accumulation through a series of phase-locked loops. These are generated with a Time Interval Analyzer (TIA) that measures cycle-to-cycle jitter over an extended period of time. The first of the four plots shows the jitter performance of a 106.25 MHz clock source. The clock oscillator provides an output with a peak-to-peak jitter of 37 ps. As the signal progresses through a series of three Cypress Semiconductor CY2308 ZDBs, the cycle-to-cycle jitter expands from 37 ps to 55 ps through the first PLL, to 59 ps after the second PLL, and finally to 68 ps after the third PLL. This shows there is a moderate accumulation of short term, cycle-to-cycle jitter through each of the PLLs. By placing three ZDBs in series, the cycle-to-cycle jitter increased from 37 ps to 68 ps. For most systems today, this amount of increase is well within the margins of a system. However, we must also analyze the long-term jitter.

Phase Noise

While the TIA technique is typically used to measure clock jitter, a Phase Noise measurement can better demonstrate the impacts of jitter accumulation and jitter attenuation over varying portions of the frequency spectrum.

Phase noise is measured in the frequency domain and is expressed as a ratio of signal power to noise power measured in a 1 Hz bandwidth at a given offset from the desired signal. A series of four phase noise plots in Figure 13.4 shows the effect of cascaded PLLs in the frequency domain.
Plot 1: Clock Oscillator

Plot 2: PLL #1

Figure 13.3 Cascaded PLL Jitter
Figure 13.3 Cascaded PLL Jitter—cont’d.
Plot 1: Clock Oscillator

Plot 2: PLL #1

Figure 13.4 Cascaded PLL Phase Noise
Figure 13.4 Cascaded PLL Phase Noise—cont’d.
Plot 1 is the spectrally pure output of the 106.25 MHz clock oscillator. Note that the gain of the frequencies away from the center is flat. In plot 2, after the first PLL, the gain no longer is flat. The noise nearest the peak is the low frequency noise or long-term jitter. The noise at the tails is the high frequency component of the clock and is the cycle-to-cycle jitter.

The rise in the “shoulders” of the plot near the center frequency is a characteristic of the PLL. These shoulders appear at approximately 1 MHz from the 106.25 MHz carrier of the CY2308 zero delay buffer as highlighted in Figure 13.5, which is the loop bandwidth of the PLL in this device. As the signal progresses through the second PLL in plot 3, the noise component again increases throughout the frequency range. In plot 4, after cascading through five PLLs, the low frequency gain has increased to nearly 30 dB. Although the cycle-to-cycle jitter has a small amount of gain, the long-term jitter amplification is substantial.

One of the characteristics of PLLs is the ability to multiply. While it is not evident in the shoulders of these phase noise plots with non-multiplied outputs, there is a phase comparison that occurs during frequency multiplication that may cause a spur within the frequency domain. Multiplication also amplifies the input phase noise. The noise increases by a multiple approximately equivalent to the multiplication factor; however, it is only within the –3 dB loop bandwidth of the PLL.

Figure 13.5 1 MHz Phase Noise Shoulders
Tracking Skew

Tracking skew is the deviation of the output of the PLL from its input. Cascaded PLLs can have an adverse effect on the amount of skew exhibited. Modulations and excessive input noise that are sometimes created by jitter peaking, can lead a phase-locked loop into a condition of instability that results in less than optimal output conditions.

To demonstrate the impact of cascading PLLs and tracking skew, the cascade of PLLs in Figure 13.4 was extended to five ZDB PLLs. If we analyze the output of the fifth PLL with a TIA, we can observe a very interesting phenomenon. In Figure 13.6, the output of the fifth PLL shows a peak-to-peak jitter of 71 ps. However, the spectral distribution is no longer Gaussian. There is now a dual peak distribution that indicates that the PLL is unable to track the excess phase noise being presented at the input to this fifth PLL.

![Figure 13.6 Tracking Skew in Time Domain](image)

It should be noted that PLLs are normally capable of tracking long-term jitter. PLLs, by design, are incapable of tracking cycle-to-cycle jitter, because the PLL response time is typically slow. When the modulation occurs at a rate and level that is too difficult for a PLL to track, the PLL may give a ‘best-effort’ tracking which we refer to as tracking skew.
Selecting PLLs

There are a variety of options when designing a clock circuit. Some include the use of PLLs and some do not. If your design requires the use of multiple PLLs, then some analysis should be performed to ensure their proper operation.

The first step is to determine if a PLL really needs to be used. If the design requires low skew outputs but no phase relationship or no multiplication is required, then a non-PLL buffer may be the best option. If it is determined that a PLL device needs to be used, then determine how many PLLs are cascaded together. Keep in mind that some ASICs and microprocessors may have internal PLLs.

After determining that PLLs exist in your design, identify the intent of their use. They can be used to allow the propagation of jitter as in the case of a spread spectrum clock or they may be used to attenuate incoming period jitter. A designer should not incorporate a spread spectrum device in a design that requires low noise at the destination. Spread spectrum technology should be used for those designs where spreading the frequency distribution is useful in reducing the overall peak-power emissions (EMI) from the design.

Determining the loop bandwidth of a PLL device may be the most important consideration when cascading PLLs. The loop bandwidth of a commercially available device typically has been considered proprietary information to the device developer. However, as PLL circuit interaction has become more critical to understand, suppliers have been willing to provide loop bandwidth information. Although loop bandwidth is deterministic, it can be complicated to characterize with devices that have programmability. For instance, the Cypress Semiconductor CY22392 clock generator incorporates programmability that changes the loop bandwidth for a given output configuration.

If a jitter transfer curve is available for the chosen PLL device. The jitter transfer curve will show the jitter peaking as well as the loop bandwidth of the PLL. Figure 13.2 earlier in the chapter shows a clean curve of a PLL clock buffer whereas Figure 13.7 shows a PLL with some deficiencies. This particular jitter transfer curve indicates excess amounts of jitter peaking at high frequencies.

Cascading PLLs

When cascading several PLLs it is important to either determine the loop bandwidth or to have the jitter transfer curve. Because PLLs exhibit jitter peaking, it is advisable to avoid cascading several devices that have similar jitter transfer curves. Unless there is a specific frequency spectrum that needs to be avoided, select devices that limit the overlap of jitter gain in their jitter transfer characteristic.
If the input reference is a spread spectrum signal then the intentional frequency modulation must not be rolled-off by the downstream PLLs. The loop bandwidth of the subsequent stages should be significantly greater than the spread spectrum modulation rate. If it is not, the spread spectrum modulation will be filtered out or the subsequent PLL outputs may have excess tracking skew. There are PLL clock buffers now available that are specifically designed to pass spread spectrum signals. These devices should be considered when buffering these types of signals.

If the input is a noisy reference and requires jitter attenuation, begin with a jitter attenuating PLL that has a close-in loop bandwidth and a sharp roll-off. This will avoid the unnecessary propagation and further peaking of the jitter through successive PLLs and the power supply system. All subsequent PLLs should once again avoid significant overlap in the gain section of the jitter transfer frequency curve.

Because PLLs are in so many devices, it is sometimes difficult to obtain either the loop bandwidth or the jitter transfer curves. In this circumstance, it is necessary to measure the short term and long-term jitter produced by the cascaded PLLs to ensure proper operation. In all cases, it is advisable to perform jitter measurements.

**Conclusion**

While there are many complex interactions that occur with cascaded PLLs, there are those who want a simple rule of thumb. Generally, more than three cascaded PLLs should be avoided. If the design requires more than three, obtain the jitter transfer curves and arrange the devices so that the jitter is attenuated. This will allow cascading six or more PLLs without creating excess jitter. For designs with critical timing margins and jitter sensitive components, measurements should be taken of the circuit to ensure proper operation.
Signal timing becomes increasingly important as system speeds increase, and clock skew often plays an important role in timing. “Skew” is the variation in the arrival time of two signals specified to occur at the same time. Any given design may use the same clock frequency at several places, and the skew between these multiple copies of the clock can make or break the system. The system designer should have a solid understanding of clock skew, know its sources and definitions, know how to apply data sheet specifications and, most importantly, know how to control skew.

While skew cannot be completely eliminated, good design practices can minimize it. Clock skew can be affected by several factors, including the buffer architecture, the types of components used, the layout of the traces, and the load on the traces.

The degree to which skew must be controlled will depend on the application. High-frequency synchronous logic typically has very tight budgets for setup, hold, and propagation times and can tolerate very little clock skew. Additionally, the timing specifications may vary from device to device such that exact skew requirements for a system can be determined only after a thorough timing analysis of every interface. Serial interfaces are very different, and while they might have a stringent frequency variation requirement, there is typically no clock skew requirement.

**Clock Tree Architecture**

Figure 14.1 shows a typical clocking network with multiple clock buffers. The waveforms show skew at various points in the clock tree. The skew comes from timing uncertainties in the buffers, as well as board-level effects. In some cases, the skew may not be important, but in other designs it must be carefully controlled. If the two devices on the left communicate with one another, it might be important to control the clock skew at points C’ and D’. Controlling skew traditionally means minimizing it, however, there are some situations where intentionally skewing the clocks is required.
To minimize skew, the number of cascaded devices should be limited. This means that one clock buffer used for fanout should not be followed by another fanout buffer if the goal is to minimize skew among all of the resulting clock signals. This is because each additional clock device in series adds more potential skew. While this is true in general, it is also important to note that skew specifications are not the same for every device, and there may be cases where two very-low-skew buffers in series can give less skew overall than a single high-skew buffer. If skew is important only across the outputs of a single device, this is not an issue. Note, however, that jitter can accumulate with cascaded PLL-based zero-delay clock buffers (see Chapter 8, Cascading PLLs).
Given the clock buffer configuration in Figure 14.2 where buffer B is a zero-delay device, we see that skew across clocks 1, 2, 4, 5, 6, and 7 will be greater than the skew across clocks 4, 5, 6, and 7.

Similarly, a single buffer has less skew than two buffers in parallel. This is primarily due to process variation from device to device. While multiple buffers should be avoided wherever possible, it is not always realistic and does not need to be the overriding consideration. The data sheet values to consider are output-to-output skew and device-to-device skew, which are discussed later in this chapter. Note that device-to-device skew is not affected by the number of devices, so three buffers in parallel will have the same overall skew as two. Once again, two (or more) low-skew buffers may have less overall skew than a single high-skew buffer, so clock distribution design can be aided by an early examination of the pertinent clock buffer specifications.

If several clocks of a given frequency are needed but only some of them have tight skew requirements, then those clock signals should be buffered together by the same low-skew device(s), where possible.

Driving multiple loads from a single output buffer might appear to be a good method for controlling skew by reducing buffer count, but it is generally discouraged for signal integrity reasons. (For a guideline on driving multiple loads, see the Multiple Loads section in Chapter 7, Clock Termination.) While it is sometimes convenient for a single output to drive two loads, the preferred practice is to drive all clocks as single-loaded, point-to-point trace.

Another issue arises when distributing clocks that contain spread spectrum for EMI reduction (see Chapter 9, Electromagnetic Interference for more information on spread spectrum). Because spread-spectrum devices modulate the output frequency, precautions must be taken to maintain minimal skew. First, there must be a single source for the spread-spectrum clock. Multiple spread-spectrum clocks generated from the same frequency will not be in phase due to the modulation. And second, downstream zero-delay clock buffers must be “spread aware”; otherwise excessive jitter will be generated.

**Layout**

Clock skew should also be considered during board floorplanning. There are many factors to consider when determining where to position components on a board, but skew should
be given appropriate attention. The designer should determine which devices belong to each clock domain and rank the relative skew importance of each appropriately. The components and clock source of skew-critical domains should be placed as close together as possible. If the relative clock trace length requirements are known, locate the clock source and buffer(s) so the traces are short and additional trace delay is not needed. Short traces minimize skew uncertainty and maximize signal integrity.

Of course, there are times when such goals are not achievable. In such cases, the goal is to maintain signal integrity and balance the distribution network with no more buffering than is necessary. For example, a clock source may be located on one side of the board while the destination devices must be placed on the opposite side. By locating the clock buffer near the destination devices, a single clock trace to the buffer need only traverse the length of the board instead of multiple copies of the clock. Another situation that often occurs is that the destination devices cannot be located near one another. In the case of non-adjacent destination devices, placing the fanout buffer midway between the destination devices usually provides the best results.

Component Selection and Data Sheet Parameters

Skew requirements can play a role in the component selection process. Zero-delay buffers address many issues, but one of their most obvious uses is to synchronize the output clocks to the input clock. For devices with external feedback, the designer has additional control of the overall input-to-output skew by controlling the delay of the feedback. Traditional non-PLL buffers can be considered when input-to-output skew—also known as propagation delay—and other zero-delay advantages are not needed. In either case, some devices have better skew specifications than others.

Output-to-output skew is the easiest to understand. It specifies the maximum timing difference between any two output clocks within a group of outputs. This group may be all of the outputs of a chip or it may be a subset, such as a bank or matched pair of outputs. Matched pairs and banks have the least amount of skew.

In Figure 14.3, output-to-output skew applies to clocks 3, 4, 5, and 6 and separately to clocks 7, 8, 9, and 10.
Device-to-device skew is also called part-to-part skew and is the worst-case skew across all output clocks on two or more devices. Note that the devices are assumed to be at the same temperature and voltage and there is no skew between their reference clocks. This parameter is used instead of output-to-output skew when two or more of the same devices are used in parallel. In Figure 14.3, device-to-device skew applies to the group of outputs 3, 4, 5, 6, 7, 8, 9, and 10, assuming there is no skew between input clocks 1 and 2.

Some zero-delay devices don’t specify device-to-device skew but instead specify a propagation delay difference. This is the maximum variation in propagation delay between different devices under the same voltage, temperature and frequency conditions. In other words, it is the process-dependent component of propagation delay skew. Add this parameter to the output-to-output skew to get the equivalent of device-to-device skew. Be careful to distinguish between propagation delay and propagation delay difference because their values are different. Also note that for device-to-device skew or its equivalent to be valid on zero-delay devices, the designer must use the same output clock for feedback on each component and ensure the trace delays are identical. Using different outputs for feedback on each device will add to the skew. In Figure 14.4, the top output is used for the feedback input on both devices.

Normally, there will be skew between the reference clocks of two buffers in parallel, as shown in Figure 14.4. Here, the total output skew will be the device-to-device skew (or equivalent) for buffers B and C, plus the output-to-output skew of buffer A.

Figure 14.2 presents one final case that is often seen where the skew across several outputs and devices 1, 2, 4, 5, 6, and 7 needs to be determined. The skew of outputs 1 and 2 is easily determined, however it has the added complexity of buffer B, which is a zero-delay buffer. The total skew is found by adding together the output-to-output skew of buffer A, the output-to-output skew of buffer B, and the maximum propagation delay of buffer B.
(For zero-delay devices, propagation delay is the difference between the reference input clock and the feedback input clock.) It is important to use the propagation delay value correctly. For example, if the propagation delay is ±350 ps, use 350 ps, not 700 ps, for the calculations.

**Extrinsic Skew**

We now shift the discussion to external (extrinsic) sources of skew. One example of an extrinsic source is unequal loading of clock outputs that can contribute to skew. A heavily loaded clock trace has a slower edge rate and therefore takes longer to transition from low-to-high versus a lightly loaded clock signal. Figure 14.5 shows the effect on a series-terminated clock with a difference of 15 pF in loading. It slows the edge rate, thereby adding delay, which might be very significant. This is effectively adding skew to the clocks.

![Figure 14.5 Loading Effects on Skew](image)

Some data sheets, such as those for the zero-delay buffers CY2305, CY2308 and CY2309, specify signal delay as a function of capacitive loading. This can be useful for skew calculations when two loads are known to be different. It is limited to about 30 pF however, which translates to approximately 1.3 ns of timing shift. Note that this applies to a lumped load on an unterminated signal and that the amount of shift will vary with process, temperature and voltage. Termination, especially series termination, can significantly magnify the effect of capacitive loading on timing. Because of these variables, this is not a recommended method for precise skew control.

Absolute loading can affect skew as much as differences in loading. In other words, the potential skew between two signals with equal 50 pF loads is greater than when the loads
are 5 pF. This occurs because of the process variation that can result in output drive strength variation from chip to chip. With large loads, the rise time increases and differences in output driver source impedance are magnified. Adding series termination to a clock output will also increase the time constant and magnify the effect. Component data sheets often specify skew for a specific maximum output load. The skew values are not guaranteed for larger loads, so the designer should take note of the specified test load for each device and attempt to stay within that value.

Skew can also be introduced by input threshold differences on the receiving chips. While it can be assumed that the high/low threshold of an input is static and is midway between the high and low input limits of the device, the actual thresholds, can in fact, vary from device to device and can be sensitive to supply noise. While this component of skew is difficult or impossible to calculate, limiting the amount of loading on the clock signals will keep it at a minimum.

**Trace Matching**

Electrical signals propagate at a finite velocity so the time it takes a signal to reach its destination will depend on the length of the trace. The board designer usually has some control over trace length and can readily use it to control timing skew. It is common practice to match the length of the clock traces to keep skew low. However, there are times when a designer may wish to use unequal line lengths to purposely cause skew to satisfy system timing requirements. In addition to line length, the trace configuration (i.e., microstrip, stripline, etc.) and the routing method can affect the delays. In all cases, trace widths and line impedances with proper termination should be maintained.

Propagation velocity is a function of the dielectric constant of the board insulator. FR-4 material, which is commonly used in PCBs, has a dielectric constant in the range of 4.3 to 4.7. Using the equation below, where \( \epsilon_\tau \) is the dielectric constant (or relative permittivity) and \( c \) is the speed of light, propagation delay for FR-4 material is about 180 ps per inch.

\[
\nu = \frac{c}{\sqrt{\epsilon_\tau}}
\]

This equation holds true for signals on the inner layers of a PCB (stripline) where there are ground layers above and below the signal. Traces on the outer layers of a PCB (microstrip) have a lower effective dielectric constant due to the low dielectric constant of air, which results in a greater propagation velocity. For the same FR-4 board, signals on the surface layers may propagate as quickly as 140 to 150 ps per inch. Therefore, when controlling the skew between two or more signals, all the traces should be routed either on the outside layer or all the traces should be routed on the inner layers.

Another variable that can affect skew control in long signal traces is unequal trace impedances. Signal impedance, which is a function of trace width, thickness and height above ground, can affect signal integrity and higher frequency losses. This means that to be equal, two signals must not only be the same length and on similar layers of the board, but they should also be the same width so that their characteristic impedances are the same.
When lengthening shorter traces to match the length of the longest clock trace, serpentine routing is often used, as shown in Figure 14.6. Using serpentine traces allows for precise control of a signal's propagation time and can be used to offset small skew differences. Care should be taken when routing serpentine traces to prevent closer-than-normal spacing. If the signals are tightly packed, the signal's propagation time may increase and thus not completely address the skew.

![Figure 14.6 Serpentine Layout for Length Matching](image)

Trace length adjustment using serpentine routing can be an excellent tool for controlling clock skew up to a couple of nanoseconds. It is important to remember that other variables, in addition to length, need to be matched for multiple traces to have truly equal delay.

**Other Skew Adjustment Methods**

Some clock buffers are designed with special features to assist the system designer in managing clock skew. Cypress has a family of skew-controlling buffers called RoboClock®. These types of components not only offer the same advantages as standard zero-delay buffers, but they allow the designer to control the frequency and skew of the outputs.

Figure 14.7 shows an example of RoboClock outputs in a variety of configurations. Some outputs may be configured to be the same frequency as the input clock while others may be configured as multiples or fractions of that frequency. There are also options to create an inverted version of the reference clock. There are two benefits of using this type of part in controlling skew. The first is the number of cascaded parts may be reduced because of the variety of output frequency configurations and, therefore, the overall skew is reduced. And second, the RoboClock device provides configurable output skew control that allows the outputs to be advanced or delayed in time. All of these adjustments are made using configuration pins.
Although Figure 14.7 shows an unmodified clock being used as the feedback input, it is possible to feed back any modified output. When an adjustment is made to the feedback clock, that adjustment is essentially applied in reverse to all of the other outputs. In other words, if a delayed output is connected to the feedback input, all other outputs are advanced. Likewise, when an advanced output is connected to the feedback input, all other outputs are delayed. This increases the range of adjustment and provides a considerable number of clocking and skew options to the imaginative system designer. (Always keep in mind that a PLL-based clock buffer will adjust the outputs of the buffer in order to phase- and frequency-align the feedback input to the reference input.)

**Induced Skew**

Up to this point, we have discussed ways to minimize skew between multiple clock signals at their destinations. However, there are some situations where the designer may wish to skew the clocks on the board in order to meet unique setup and hold time or to compensate for a very fast or very slow device. These clock-skewing methods work well in these situations, as well.

Both the serpentine trace configuration and the use of a RoboClock device allow the designer to purposely skew multiple clocks in incremental adjustments. For very fine control (tens to hundreds of picoseconds) of skew between multiple traces, the serpentine trace will produce excellent results. For larger skew (hundreds of picoseconds to several nanoseconds) and ease of skew control, a RoboClock-type of part is the best choice.

Another method for adding delay is to use discrete delay lines. These may be a reasonable choice when a large delay (tens of nanoseconds or more) and low accuracy are required. Delay lines often have taps for selecting between a variety of delays, which can be useful in some applications. Among their disadvantages, however, are that they are usually inappropriate for delays of less than a few nanoseconds or where high accuracy is required.
Passive delay lines are also prone to degrading the edge slew rate of the clock, and this degradation is proportional to the amount of delay. For all of these reasons, discrete delay lines are seldom used with high-speed digital logic.

**Conclusion**

Skew control is an important aspect of clock distribution circuits. Many factors can affect the arrival times of the clock signals including the number of buffers, the type of buffer, the trace layout and the loading of the traces. Knowing how each of these affects the clock design and by using techniques such as serpentine traces and skew-controlling RoboClock buffers, skew can be controlled to provide the very best system performance.

To summarize an approach to skew control:

- Know the skew requirements/tolerances for your system
- Know the chip specifications and how to apply them
- Think of skew when designing the clock distribution system
- Think of skew when doing board floorplanning and layout
- Consider extrinsic skew factors: loading, trace length, etc.
- Use appropriate means to achieve skew requirements
- Check clock timing after the board is built, ideally across temperature and voltage
Introduction
In the clock IC industry, the trend in component packaging is to get more signals and higher performance out of a package, while reducing cost, footprint size, and package profile. To support these desires, packaging technology has exploded in the last decade with a wide range of options. The new Quad Flatpack No-lead (QFN) package is the next step in the lower cost, higher performance progression from the popular SOIC, TSSOP, and TQFP package varieties. Exploring the new QFN package will provide an excellent review into package parameters critical to board designers, such as thermal characteristics and optimal layout for manufacturability.

IC Packaging Functions
The package of an integrated circuit provides three major functions, as described below:

Heat Dissipation:
One of the most important functions of an IC package is to dissipate the heat generated to the outside environment. The materials that are selected for the package and the construction of the package define the thermal efficiency of the package.

Signal Distribution:
For clock devices that operate at very high frequencies, the package design is very critical to signal integrity. Package design influences capacitance, inductance and resistance, which all can greatly influence the signal integrity.

Package Protection:
Another important function of the package is to protect the silicon chip inside and the fragile electrical connections to the chip from the outside environment. Mechanical damage, moisture, impurities, and UV light can cause damage to the sensitive IC inside.
Package Physical Comparisons

Tables 15.1 and 15.2 show physical characteristics for several 16-pin and 48-pin packages. These tables and drawings give a good idea of how physically different the packages can be for devices with the same number of pins. They also show the progression of technology towards reducing footprint size and pin pitch.

### Table 15.1 Physical Characteristics for 16-Pin Small Packages

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>SOIC (S)</th>
<th>SSOP (O)</th>
<th>TSSOP (ZS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (mm)</td>
<td>9.90 ±0.10</td>
<td>6.20 ±0.13</td>
<td>5.00 ±0.10</td>
</tr>
<tr>
<td>Width (mm)</td>
<td>6.00 ±0.20</td>
<td>7.80 ±0.20</td>
<td>6.375 ±0.125</td>
</tr>
<tr>
<td>Max. Height (mm)</td>
<td>1.75</td>
<td>2.00</td>
<td>0.95</td>
</tr>
<tr>
<td>Pitch (mm)</td>
<td>1.27</td>
<td>0.65</td>
<td>0.65</td>
</tr>
<tr>
<td>Footprint Area (mm²)</td>
<td>59.4</td>
<td>48.36</td>
<td>31.875</td>
</tr>
<tr>
<td>Weight (g)</td>
<td>0.15</td>
<td>0.13</td>
<td>0.05</td>
</tr>
</tbody>
</table>

### Table 15.2 Physical Characteristics for 48-Pin Packages

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>SSOP (O)</th>
<th>TSSOP II(ZS)</th>
<th>TQFP (A)</th>
<th>QFN (QFN)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length (mm)</td>
<td>15.875 ±0.125</td>
<td>12.50 ±0.10</td>
<td>9.00 ±0.25</td>
<td>7.00 ±0.10</td>
</tr>
<tr>
<td>Width (mm)</td>
<td>10.35 ±0.30</td>
<td>8.10 ±0.15</td>
<td>9.00 ±0.25</td>
<td>7.00 ±0.10</td>
</tr>
<tr>
<td>Max. Height (mm)</td>
<td>2.80</td>
<td>0.95</td>
<td>1.6</td>
<td>1.00</td>
</tr>
<tr>
<td>Pitch (mm)</td>
<td>0.65</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Footprint Area (mm²)</td>
<td>164.3</td>
<td>101.25</td>
<td>81</td>
<td>49</td>
</tr>
<tr>
<td>Weight (g)</td>
<td>0.65</td>
<td>0.33</td>
<td>0.18</td>
<td>0.13</td>
</tr>
</tbody>
</table>
Package Electrical Characteristics

Table 15.3 shows the electrical characteristics for a variety of packages, in order of increasing average inductance. From the data, we can see that the 48-pin QFN has the lowest inductance, resistance, and capacitance by a significant margin. The 32-pin PLCC has the highest inductance and resistance, while the 56-pin SSOP has the highest capacitance. Electrical data shown is an approximation based on modeling.

Table 15-3 Electrical Characteristics for Different Packages, In Order of Increasing Average Inductance

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>48-pin QFN</td>
<td>0.081</td>
<td>0.003</td>
<td>0.185</td>
</tr>
<tr>
<td>8-pin SOIC</td>
<td>0.987</td>
<td>0.007</td>
<td>0.24</td>
</tr>
<tr>
<td>8-pin TSSOP</td>
<td>1.1</td>
<td>0.011</td>
<td>0.2</td>
</tr>
<tr>
<td>48-pin TQFP</td>
<td>1.4</td>
<td>0.015</td>
<td>0.32</td>
</tr>
<tr>
<td>8-pin SSOP</td>
<td>1.48</td>
<td>0.012</td>
<td>0.28</td>
</tr>
<tr>
<td>48-pin TSSOP</td>
<td>2.345</td>
<td>0.049</td>
<td>0.45</td>
</tr>
<tr>
<td>56-pin SSOP</td>
<td>4.45</td>
<td>0.035</td>
<td>0.72</td>
</tr>
<tr>
<td>32-pin PLCC</td>
<td>5.5</td>
<td>0.065</td>
<td>0.565</td>
</tr>
</tbody>
</table>

Package Thermal Characteristics

All electronic circuits dissipate power, which is released into the ambient environment in the form of heat. Since integrated circuits can be damaged at high temperatures, care must be taken to ensure that the junction, or the die, of the IC stays under the maximum rated temperature under worst case conditions of ambient temperature, airflow, and power consumption.

Thermal resistance is the resistance of the package to heat dissipation and is inversely related to the thermal conductivity of the package. Heat from the IC is conducted through the package into the ambient environment, which increases the temperature of the die (junction temperature – TJ) above the ambient temperature (TA). The thermal conductivity of the silicon chip, die-attach epoxy, copper leadframe, and mold compound affect the rate at which the heat is dissipated. The geometry of the package and of the printed circuit board (PCB) greatly influences how quickly the heat is transferred to the PCB and away from the chip. The most commonly used thermal metrics for IC packages are thermal impedance, measured or modeled, from the chip junction to the ambient air surrounding the package (θJA, or θJA) and thermal impedance, measured or modeled, from the chip junction to the case (θJC, or θJC). Figure 15.1 is a thermal representation of a typical IC plastic package, with the thermal metrics identified. The resistor symbols are shown since heat flow is analogous to current flow in a simple electronic circuit. The temperatures are
like voltages, and the thermal impedances are like resistors. Power flowing through a thermal impedance is analogous to current flowing through a resistor.

![Figure 15.1 IC Package Thermal Metrics](image)

Mathematically, $\theta_{JA}$ is defined as:

$$\theta_{JA} = \frac{(T_J - T_A)}{P}$$

Where:

- $T_J =$ junction temperature of the chip
- $T_A =$ ambient temperature
- $P =$ power to the chip

Mathematically, $\theta_{JC}$ is defined as:

$$\theta_{JC} = \frac{(T_J - T_C)}{P}$$

Where:

- $T_J =$ junction temperature of the chip
- $T_C =$ package case temperature
- $P =$ power to the chip

When a device reaches a state of thermal equilibrium, the electrical power delivered is equal to the thermal heat dissipated, which is transferred to the surrounding environment. The maximum allowable power consumption ($P$) at a given ambient temperature ($T_A$) is computed using the maximum junction temperature for the chip ($T_J$) and the thermal resistance of the package ($\theta_{JA}$), as shown in the following equation:

$$P = \frac{(T_J - T_A)}{\theta_{JA}}$$

This maximum allowable power consumption is based on a maximum junction temperature ($T_{J \max}$). If the junction temperature rises above this value, irreversible damage to the silicon can occur. This equation can be used to determine the maximum power that can be dissipated by the particular package under specific ambient operating environment conditions.
As an example, the 16-pin SOIC package has a $\theta_{JA}$ of 136°C/W at zero airflow. The maximum power that the package can withstand at $T_A = 25°C$ and $T_J = 155°C$ is:

$$P = \frac{(155°C - 25°C)}{136°C/W} = 956 \text{ mW}$$

The effects of airflow on the thermal impedance of a package can also be considered. Airflow reduces the thermal impedance of a package. By using the thermal impedance value of the package at different airflows, the maximum power equation can be used to create a derating curve for each of the airflows. A typical set of derating curves for the 16-pin SOIC package is shown in Figure 15.2.

![Figure 15.2. Sample Derating Curves for 16-Pin SOIC Package](image)

The derating curves allow the circuit designer to see the effect of rising ambient temperature and changes in airflow on the maximum power dissipation allowed for a given package and device. The derating curve displays the maximum power dissipation ($y$-axis) of the package over a range of ambient temperatures ($x$-axis). The maximum power dissipation is directly derived from the maximum junction temperature for the particular device.

$\theta_{JA}$ is used if the package itself dissipates the heat energy into the ambient environment. An external heat sink, thermally attached to the case, can be used to dissipate more heat energy than the package by itself. If an external heat sink is attached to the case, the junction-to-case thermal impedance ($\theta_{JC}$) is used instead of $\theta_{JA}$, and an additional case-to-ambient theta ($\theta_{CA}$) is used in the calculation to model the heat dissipation characteristics of the specific heat sink used. This $\theta_{CA}$ is added in series with the package $\theta_{JC}$ to obtain an effective $\theta_{JA}$. This effective $\theta_{JA}$ is then used to calculate the junction temperature.

Table 15.4 shows the junction-to-ambient thermal impedance ($\theta_{JA}$) and the junction-to-case thermal impedance ($\theta_{JC}$) for a variety of packages, with no airflow. Figure 15.3 is a graph of these thermal impedances. The thermal data shown is an approximation based on modeling.
Table 15.4 Thermal Impedance Comparison for Packages with No Airflow. Die size for 16-pin packages is assumed 61.4 mils x 58.5 mils; die size for other packages assumed 80 mils x 93 mils.

<table>
<thead>
<tr>
<th>Package</th>
<th>Thermal Impedance (\theta_{JA}) (°C/W)</th>
<th>Thermal Impedance (\theta_{JC}) (°C/W)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-pin SOIC</td>
<td>136</td>
<td>55.3</td>
</tr>
<tr>
<td>16-pin SSOP</td>
<td>130.2</td>
<td>44.2</td>
</tr>
<tr>
<td>16-pin TSSOP</td>
<td>129.7</td>
<td>45.1</td>
</tr>
<tr>
<td>32-pin PLCC</td>
<td>65.0</td>
<td>36.7</td>
</tr>
<tr>
<td>48-pin SSOP</td>
<td>80.5</td>
<td>35.2</td>
</tr>
<tr>
<td>48-pin TSSOP</td>
<td>75.2</td>
<td>20.5</td>
</tr>
<tr>
<td>48-pin TQFP</td>
<td>68.4</td>
<td>32.7</td>
</tr>
<tr>
<td>48-pin QFN</td>
<td>56.0</td>
<td>—</td>
</tr>
</tbody>
</table>

Figure 15.3 Theta \(\theta_{JA}/\theta_{JC}\) Comparison Graph for Data in Table 15.4

**Solder Reflow Profile**

Packaging also determines the reflow heating conditions to be used when a device is being soldered to a printed circuit board. The traditional packaging solder finish is SnPb (Tin, Lead). Convection 220 is used on larger devices with SnPb solder finish, while the slightly hotter convection 235 is used on smaller SnPb solder finished devices. Pb-free packaging is also becoming increasingly popular, and is available in several varieties, including a tin-based Pb-free finish and a Ni-Pd-Au (Nickel, Lead, Gold) based finish. Pb-free packaging requires a considerably higher temperature during reflow than the traditional SnPb packaging. Tables 15.5 and 15.6 show the specifications for Convection 220, Convection 235, and Pb-free reflow profiles. Figures 15.4 and 15.5 show the graphs of these profiles.
Table 15.5 Convection Reflow Profile for SnPb, Convection 220 and Convection 235 (J-STD-020)

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average ramp-up rate (183°C to peak)</td>
<td>3°C/second max</td>
</tr>
<tr>
<td>Preheat temperature 125 (±25°C)</td>
<td>120 seconds max</td>
</tr>
<tr>
<td>Temperature maintained above 183°C</td>
<td>60 to 150 seconds</td>
</tr>
<tr>
<td>Time within 5°C of actual Peak Temperature</td>
<td>10 to 20 seconds</td>
</tr>
<tr>
<td>Peak Temperature, Convection 235</td>
<td>235+5/-0°C</td>
</tr>
<tr>
<td>Peak Temperature, Convection 220</td>
<td>220+5/-0°C</td>
</tr>
<tr>
<td>Ramp-down Rate</td>
<td>6°C/second max</td>
</tr>
<tr>
<td>Time 25°C to Peak Temperature</td>
<td>6 minutes max</td>
</tr>
</tbody>
</table>

Table 15.6 Convection or IR 260 -5/+0°C Reflow Profile for Pb-free (NEMI Recommended)

<table>
<thead>
<tr>
<th>Temperature</th>
<th>Profile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average ramp-up rate (217°C to peak)</td>
<td>3°C/second max</td>
</tr>
<tr>
<td>Preheat temperature 150 (±25°C)</td>
<td>60 to 120 seconds max</td>
</tr>
<tr>
<td>Time 50°C to Peak Temperature</td>
<td>3.5 minutes, 6 minutes max</td>
</tr>
<tr>
<td>Temperature maintained above 217°C</td>
<td>60 to 150 seconds</td>
</tr>
<tr>
<td>Time within 5°C of actual Peak Temperature</td>
<td>10 to 20 seconds</td>
</tr>
<tr>
<td>Peak Temperature</td>
<td>260 -5/+0°C</td>
</tr>
<tr>
<td>Ramp-down Rate</td>
<td>6°C/second max</td>
</tr>
</tbody>
</table>

Figure 15.4 Convection 235 Reflow Profile for SnPb
One new package format that is advancing performance while reducing profile and footprint is the Quad Flat-pack No-lead (QFN) package. This package is typically available in 32-, 40-, 48-, and 56-pin varieties, and has the same or similar pin out as the Thin Plastic Quad Flatpack package (TQFP). QFN packages are up to 40% smaller than their TQFP counterparts. Also, according to QFN modeling data, thermal performance is improved up to 20% and electrical performance is improved up to 80% for inductance and up to 40% for capacitance over the TQFP. The MicroLeadFrame (MLF) package is sometimes seen and is synonymous with the QFN package.

Figure 15.7 illustrates how the package height is reduced to a minimum by having both the die and wirebond pads on the same plane. When mounted, the leads are directly attached to the board without a space-consuming stand off which is inherent in a leaded package. Figure 15.7 also illustrates how the ends of the leads are flush with the edge of the package. This configuration allows for the maximum die size within a given footprint, while maximizing the board space efficiency. In addition to these features, the QFN package has excellent thermal dissipation and reduced electrical parasitics due to its efficient and compact design.
Printed Circuit Board Design Considerations for QFN Package

Figure 15.8 shows the bottom and side views of the full lead option, indicating the dimensions needed to design the PCB pad pattern for PCBs. Since most packages are square with dimension D equal to dimension E and the leads are along the E direction for dual packages, the side view dimensions (D, S, D2, & L) are used to determine the land length on the PCB.

The PCB pad pattern dimensions to be determined are shown in Figure 15.9. In the figure, the dimensions ZDmax and GDmin (and ZEmax and GEmin) are the outside-to-outside and inside-to-inside pad dimensions, respectively. The dimension X and Y indicate the width and the length of the pad, respectively. Two additional clearances C\text{LL} and C\text{PL} are also defined to avoid solder bridging. While C\text{LL} defines the minimum distance between land to land for the corner joints on adjacent sides, C\text{PL} defines the minimum distance between the inner tip of the peripheral lands and the outer edge of the thermal pad. To design a proper pad pattern, tolerance analysis is required on package and PCB board dimensions.

Figure 15.8 QFN Bottom & Side Views

Figure 15.9 QFN Bottom & Pad View
QFN Thermal Pad

The QFN package is designed to provide superior thermal transfer performance. Since the QFN has no leads in the traditional sense, many of the packages have a thermal paddle on the bottom of the package. This is shown in Figure 15.9 with the D2’ and E2’ dimensions. For components with this paddle, special attention is required to the heat transfer area below the package to ensure a good thermal bond to the circuit board. To take full advantage of this function, the PCB must have features to effectively conduct heat away from the package. This can be achieved by incorporating a thermal pad and thermal vias on the PCB. While the thermal pad provides a solderable surface on the top of the PCB to solder the package die paddle on the board, thermal vias are needed to provide a thermal path to inner and/or bottom layers of the PCB to remove the heat.

Normally, the size of the thermal pad should at least match the exposed die paddle size. However, depending upon the die paddle size, the thermal pad size needs to be modified in some cases to avoid solder bridging between thermal pad and the perimeter pads. This is done by defining a clearance between the outer edges of the thermal pad and the inner edges of perimeter pads. This clearance is defined as CPL in Figure 15.9 and is fixed at 0.15 mm here. With this constraint, the maximum size of the thermal pad is calculated by the following relationship:

\[ D2'_{TH \ max} = GD_{min} - 2C_{PL} \]

It should be noted that the D2’_{TH} dimension gives the theoretical maximum value. Since the size of the exposed die paddle on the component may actually be much smaller than this value, the actual D2’ dimension should be modified by:

\[ \text{Component } D2 \leq D2' \leq D2'_{TH \ max} \]

Before placing a pad and via array, the manufacturer of the device should be consulted for layout guidelines. For many of the higher power devices that use the thermal paddle, application notes are generally available that recommend proper PCB layout procedures. Also note that not all QFN packages use the thermal paddle. Many of the low power devices don’t require the use of the paddle to dissipate the heat.

Conclusion

Packaging is a very important characteristic of IC design and performance. Larger packages may be easier to probe during system test but smaller packages offer many more benefits than the obvious space savings. Some of the newer small packages offer better capacitance, resistance, and inductance characteristics needed for the high speed signals that are now so common in today’s designs. New packages such as the Quad Flatpack No-lead (QFN) are leading the industry in all aspects of performance, including minimal footprint and profile, as well as superior electrical and thermal characteristics. Proper pad layout, reflow profile, and thermal loading during operation are all packaging parameters that are critical to the board designer to guarantee a successful design. With careful attention to detail, you can get the most out of your IC packaging.
Introduction

In previous chapters of this book, we have focused on various detailed elements of clock design. This chapter takes a look at clocks at a system level and how these clocks are validated against a given specification. PCs traditionally use many frequencies and have in many instances advanced the timing technologies, so we’ll use this platform as a model system to show testing and validation techniques. Since motherboards have single-ended and differential clocks, as well as SERDES interfaces, there are many parallels between this system and many others.

Computer motherboards for PCs, laptops and servers have many different systems and components requiring a variety of frequencies and voltage signaling levels for proper operation. Motherboard clock generators are a specific type of clock device designed to produce all the necessary clocks from a single, inexpensive 14.318-MHz crystal. This chapter will explain the different types of clocks that are produced by motherboard clock generators, what part of the computer they are intended for, and how validation is used to ensure proper operation.

PC Clock Overview

Before diving into the specific clocks of a PC, a discussion of PC architecture seems appropriate. The simple exterior of a desktop computer hides the complicated structure of the internal motherboard. The motherboard contains various subsystems, including the CPU (central processing unit), memory, the graphics/memory controller hub (MCH, formerly known as the “Northbridge”), the I/O controller hub (ICH, formerly known as the “Southbridge”), and the BIOS. In the former Northbridge/Southbridge architecture, the two bridge devices used a 133-MBps PCI (Peripheral Component Interconnect) bus to communicate with each other. In the MCH/ICH hub architecture, the devices have a direct
266-MBps interface, which doubles the communication bandwidth between the devices. PCI, if implemented, is now only used on the ICH. All of these components work together to accomplish the wide range of functions performed by the PC. Clock references are used to mediate data traffic between the different subsystems.

Many of the terms and references, such as MCH and ICH, apply to Intel®-based processor architectures. However, there are other processor architectures and chipsets, such as AMD and IBM’s PowerPC. Each is slightly different, but overall the concepts and system architecture are very similar and the methods shown here apply to all. The remainder of this chapter will continue to use the Intel architecture as the reference platform.

The clock references come in two varieties: main clock generator and clock buffer. The main clock generates most of the outputs required for the system to function. The main outputs are the clock signals used by the CPU, MCH and ICH to communicate with one another. Additional outputs include reference clocks for USB, PCI bus slots and the ATA disk controller.

Clock frequencies must increase as the motherboard performance increases. These higher frequency requirements also change the clock signal characteristics. For example, at higher frequencies, clock voltage amplitudes will have to decrease to meet the slew rate requirements. Also, with higher frequencies and lower amplitude, noise coupling or interference can more easily affect the signals, thus requiring a migration from single-ended signaling to differential signaling.

CPU reference clock frequencies in the current generation of motherboards may run from 266 MHz to 333 MHz, which are then internally multiplied to achieve advertised CPU frequencies. PCI reference-clock frequencies and hard drive reference-clock frequencies will both migrate from 33-MHz single-ended LVTTL to 100-MHz differential at 0.7V. With these new standards emerging, new test methodologies must be applied to guarantee that these clocks will work in the system. Similar standards exist today in the realm of communications, but due to the nature of PC motherboard clocks, where cost is driving the market, cost-effective test methodologies need to be developed. A Digital Storage Oscilloscope (DSO) is a common piece of test equipment, so test methodologies will be developed around it.

PCs continue to evolve in order to increase performance in many areas. This evolution goes well beyond merely the number of gigahertz and megabytes a processor may have. For example, bus architectures continue to change, clock speeds increase, and new standards emerge, such as Serial ATA (SATA) and Serial Attached SCSI (SAS) for disk drives and PCI Express (PCIe) for I/O. As new specifications emerge, new clock requirements often are needed. All of the standards required for a given PC are documented in some specification such as the Intel “Yellow Book”.

The Clock Architecture

Figure 16.1 shows the typical outputs of a motherboard clock and the components each output drives.
The CPU clock outputs are used by the CPU and MCH in two ways. The first is to create a common reference clock to transfer information between them. This frequency is multiplied internally, with the resultant frequency known as the Front Side Bus. The second usage of the CPU outputs from the motherboard clock is to use it as a reference and multiply internally, which results in the raw CPU speed.

Notice that the diagram labels the MCH as “GMCH”. The “G” simply means that a graphics controller is part of the device. A dot clock is only used for the Intel chipset with on-board graphics as a video clock. This clock is used by the memory controller hub to generate the horizontal synchronization (hsync) signal used by CRT monitors.

The SRC (serial reference clock) is used in several locations. First, it replaces the reference clock used by the AGP (Accelerated Graphics Port) interface. The MCH takes this clock and multiplies it to 2.5 GHz, which is the PCIe standard frequency. This clock is also used by the ICH to communicate with Serial ATA drives, as well as other PCIe peripherals.

The PCI clock is used by the ICH to communicate with PCI devices connected to PCI slots or legacy drives; the USB clock is used by the ICH to communicate with USB devices; and the REF clock is used by the ICH to communicate with legacy audio devices. The REF clock is often used to support miscellaneous devices on the motherboard, such as ASICs and FPGAs.
As defined by Intel’s Yellow Book, these clocks must meet certain AC specifications. The more critical specifications include:

- Period, clock accuracy (ppm)
- Cycle-to-cycle jitter
- Rise time
- Fall time
- Duty cycle

**Measurement Procedures and Equipment**

Period measurement is taken from an average period over any 1 us interval of time. This can be done in two ways—by using a DSO (digital storage oscilloscope) or by using a more specialized piece of equipment. Using a DSO requires that the entire 1-µs interval be stored in the oscilloscope’s memory. For example, if the output is 100 MHz with a period of 10 ns, to capture 1 µs worth of the time period, the scope needs to store 100 cycles. The following calculations show how much memory is needed. If the oscilloscope has a sampling rate of 20 Gsps, this translates to 50 ps/sample that would give us 500 points (pt) per cycle. To capture 100 cycles, this would require 500 pt x 100 cycles, which equates to 50K samples (or 50 KB of memory with an 8-bit sample size).

A second method to measure the time period is with the use of a modulation domain analyzer or an accurate frequency counter. These two types of specialized test equipment have the capability to provide an average frequency taken over a specified interval.

Taking accurate measurements with any of these pieces of equipment requires that the measuring device and probe have adequate bandwidth to handle the rise times of the clock signal. For more information, see Chapter 11, “Probing High-Speed Clocks.” Also, the measurements taken are relative to the reference output. Before a reading can be judged to be in violation of the clock accuracy specifications, a measurement of the reference output must be taken to ensure that the reference is within the specification. The outputs are simply multiples of the reference and will directly reflect the accuracy of the reference.

In the past, cycle-to-cycle jitter has been measured with an 8-10 Gsps sampling rate oscilloscope. Recently, the specifications have become tighter, requiring test equipment with more performance. The new standard for determining whether a clock meets or fails the cycle-to-cycle jitter test is done with a sampling scope with at least 10 Gsps sampling and accumulating at least 100K cycles. If we take the above example, a 10-ns period would require at least 20 MB of memory.

**Cycle-to-Cycle Jitter Measurements**

Software is available for DSOs that makes measuring jitter quick and simple. Here is the step-by-step procedure for measuring cycle-to-cycle jitter using a digital storage scope, running the Tektronix TDSJIT3 measurement software:
1. Set up the scope to show a few cycles of the waveform as in Figure 16.2.

2. Configure the reference signal. This will allow the software to correctly determine the signal level. Figure 16.3 shows the default settings.

3. Select the active channel and click on the Setup button under Autoset to configure TDSJIT3. The result is shown in Figure 16.4.

4. Set the memory to its maximum. To set the memory depth, go to the “Horiz/Acq” menu and select “Horiz Setup.” In the example in Figure 16.5, the maximum is 32 MB in record length.

5. Select the “Go to Results” tab on the left side, click on “Single” to acquire the data, and the screen should look like Figure 16.6.

Cycle-to-cycle jitter can be read off of the statistics results lists in Figure 16.6. The parameter for cycle-to-cycle jitter is “max + delta” and “max – delta.”

Figure 16.2 First Step in Cycle-to-Cycle Jitter Measurement
Figure 16.3 Default Settings to Set Reference Level for Cycle-to-Cycle Jitter Measurement

Figure 16.4 Screen after Autoset Command, Cycle-to-Cycle Jitter Measurement
Rise/ Fall Time and Duty Cycle Measurement Procedures

TDSJIT3 can also take rise/fall time and duty cycle measurements. Rise and fall time measurements are taken on the single-ended waveform with a TDSJIT3 screen, as in Figure 16.7. Figure 16.8 shows the duty cycle measurement, which is taken on the differential waveform.
Serial Signaling Overview

The latest generation of PC motherboards uses high-speed serial signaling standards to overcome the bottlenecks created by existing standard parallel buses. The first serial standards to be adopted in the PC space are for Serial ATA, which replaces IDE parallel buses for communication between chipsets and hard disks, and PCI Express (PCIe), which
is a more general-purpose serial standard. PCIe will initially replace the parallel AGP standard and will eventually replace most of the peripheral interconnect currently serviced by the regular PCI bus.

Serial links utilize devices with embedded SERDES (SERializer/ DESerializer) to transmit and receive serial data. The SERDES portion of the device converts parallel data into serial data for transmission and converts the received serial datastream back into parallel for processing. The SERDES also performs other functions necessary to maintaining a serial link, such as scrambling and encoding the transmitted data, and framing, decoding and descrambling the received data. PCIe is implemented by a full-duplex link, which means that there are separate differential pairs for transmitting and receiving data.

A simplified diagram of a SERDES is shown in Figure 16.9. A reference clock is used in both the transmit and the receive sections. This reference clock is usually defined to be the same rate as the parallel data. For example, in a SERDES that converts parallel data from 10 bits at 250 Mbps to a serial stream of 2.5 Gbps, the reference clock is usually specified at 250 MHz, or 125 MHz if DDR clocking is used on the parallel input. The reference clock is sometimes used to latch the parallel data into the serializer.

![Simplified SERDES Block Diagram](image)

**Figure 16.9 Simplified SERDES Block Diagram**

On the transmit side, the reference clock is multiplied by a PLL to the serial output rate. The output of this PLL is used as the bit-rate clock to time the output serial data. This clock is effectively embedded into the serial stream.

At the receiver end of the link, the reference clock is used to set the approximate frequency for the internal PLL to track the incoming datastream and recover the original clock and data. SERDES devices specify a small maximum allowable difference, such as ± 300 ppm, between the clock used on the transmit side of the link and the clock used as the receiver reference clock on the other side of the link. If the difference between the transmitter clock
and receiver reference clock is outside this tolerance, the receiver will be unable to correctly recover the data and clock from the incoming serial stream. In a full-duplex link where the same clock is used for both the transmit and receive portions of the SERDES, the reference clock tolerances also ensure that the signaling rates in either direction are closely matched to prevent buffer overrun or underrun conditions.

The easiest and most reliable way of matching signaling rates in either direction is to send a copy of the transmit reference clock to the receiver. This ensures that the transmitter and receiver of both sides of the link are using a clock with no frequency difference (0 ppm).

In PCIe systems utilizing spread-spectrum clocking, this is the only way to ensure the accuracy requirement is met under conditions where the clock can be modulating by up to 0.5%. The PCIe connector specification demands that a reference clock be sent over the connector along with the serial bitstream.

Figure 16.10 shows the typical topology used in a PCIe system. The motherboard clock generator generates multiple versions of the serial reference clocks, with current generations producing about eight copies. One copy of the reference clock goes to the root complex (possibly a Northbridge sort of chipset device that connects the CPU/memory subsystem to I/O devices), and one copy goes to the end of each peripheral that is connected to the root complex via a serial link. This figure shows one of the endpoints using a local oscillator as a reference clock, so the clock transmitted over the connector is not used. However, it will be difficult to ensure that this local oscillator stays within the required ppm difference, and this connection will not be able to utilize spread-spectrum clocking.
The reference clock source needs to be a stable and low-jitter source with the required frequency stability. Instability in the reference clock can lead to data errors in the serial link. On the transmit side, the output jitter in the transmitted serial data will be influenced by the jitter present in the reference clock. The PLL will filter out any high frequency jitter that is well above its bandwidth, but any jitter close to or below its bandwidth will be passed directly on to the transmitted serial stream.

Figures 16.11 and 16.12 demonstrate how jitter is transferred from the reference clock to the serial data stream. Figure 16.11 shows the serial output of a SERDES device running at 1.25 Gbps. In this case, the 125 MHz reference clock came from a clean clock generator. The output eye diagram is wide open with very low jitter. Figure 16.12 shows the same device running at the same rate, but using a lower-quality, jittery clock source as the transmit reference clock. The clock source used for Figure 16.12 has about 200 ps of cycle-to-cycle jitter. It can be easily seen that this jitter is now present on the serial output. The scope readings show that the peak-to-peak jitter on the output in Figure 16.12 is 342 ps, compared with 95 ps when using the low-jitter clock source in Figure 16.11.
The reference clocks used by PCIe and SATA are known as Serial Reference Clocks (SRC). These are defined as low-voltage differential swing clocks running at 100 MHz. The voltage swing is from 0V to 0.7V single-ended, which gives a differential swing of 1.4V peak-to-peak. Some of the important specifications pertaining to these clocks are listed in Table 16.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Condition</th>
<th>Min.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>tDC</td>
<td>SRCT and SRCC Duty Cycle</td>
<td>Measured at crossing point V'OX</td>
<td>45</td>
<td>55</td>
<td>%</td>
</tr>
<tr>
<td>tPERIOD</td>
<td>100-MHz SRCT and SRCC Period</td>
<td>Measured at crossing point V'OX</td>
<td>9.997001</td>
<td>10.003</td>
<td>ns</td>
</tr>
<tr>
<td>tPERIODSS</td>
<td>100-MHz SRCT and SRCC Period, SSC</td>
<td>Measured at crossing point V'OX</td>
<td>9.997001</td>
<td>10.05327</td>
<td>ns</td>
</tr>
<tr>
<td>tPERIODABS</td>
<td>100-MHz SRCT and SRCC Absolute Period</td>
<td>Measured at crossing point V'OX</td>
<td>9.872001</td>
<td>10.128</td>
<td>ns</td>
</tr>
<tr>
<td>tPERIODSSABS</td>
<td>100-MHz SRCT and SRCC Absolute Period, SSC</td>
<td>Measured at crossing point V'OX</td>
<td>9.872001</td>
<td>10.17827</td>
<td>ns</td>
</tr>
<tr>
<td>tSKEW</td>
<td>Any SRCT/C to SRCT/C Clock Skew</td>
<td>Measured at crossing point V'OX</td>
<td>—</td>
<td>100</td>
<td>ps</td>
</tr>
<tr>
<td>tCCJ</td>
<td>SRCT/C Cycle-to-Cycle Jitter</td>
<td>Measured at crossing point V'OX</td>
<td>—</td>
<td>125</td>
<td>ps</td>
</tr>
<tr>
<td>LACC</td>
<td>SRCT/C Long Term Accuracy</td>
<td>Measured at crossing point V'OX</td>
<td>—</td>
<td>300</td>
<td>ppm</td>
</tr>
<tr>
<td>tR/tF</td>
<td>SRCT and SRCC Rise and Fall Times</td>
<td>Measured from VO_L = 0.175 to VO_H = 0.525V</td>
<td>175</td>
<td>700</td>
<td>ps</td>
</tr>
<tr>
<td>tRFM</td>
<td>Rise/Fall Matching</td>
<td>Determined as a Fraction of 2* (tR−tF)/(tR + tF)</td>
<td>—</td>
<td>20</td>
<td>%</td>
</tr>
<tr>
<td>V_HIGH</td>
<td>Voltage High</td>
<td>Math Averages</td>
<td>660</td>
<td>850</td>
<td>mV</td>
</tr>
<tr>
<td>V_LOW</td>
<td>Voltage Low</td>
<td>Math Averages</td>
<td>-150</td>
<td>—</td>
<td>mV</td>
</tr>
<tr>
<td>V'OX</td>
<td>Crossing Point Voltage at 0.7V Swing</td>
<td>Math Averages</td>
<td>250</td>
<td>550</td>
<td>mV</td>
</tr>
</tbody>
</table>

The performance of these clocks can be measured using the techniques of period, cycle-to-cycle jitter, rise/fall time, and duty cycle measurements, as described earlier in this chapter. Because these clocks are feeding a downstream PLL, we are also concerned with the spectra of the clock signal, the phase noise, and the cycle-to-cycle jitter. The reason for this is explained more thoroughly in Chapter 13, “Cascading PLLs.” We are especially interested in jitter components that fall below the transmit PLL’s loop bandwidth. For a PCIe
SERDES, the PLL loop bandwidth is typically between 7 MHz and 22 MHz. Excessive jitter components present on the reference clock signal below these frequencies will be multiplied up by the transmit PLL and will degrade the quality of the transmitted serial bit-stream.

As with the cycle-to-cycle jitter measurements, there are two options for analyzing the spectrum of jitter present in the SRC clock signal—using a piece of specialized equipment known as a spectrum analyzer, or using software on a DSO. As before, we will focus on the DSO method because of its convenience. Analyzing the jitter spectrum with a DSO involves taking the Fast Fourier Transform (FFT) of the period measurements of a clock signal to produce a spectral plot. An example is shown in Figures 16.13 and 16.14 using a Tektronix 7404 scope with the TDSJIT3 jitter analysis package. To analyze the jitter spectrum, first a period jitter measurement over a large number of cycles is taken. This is a measurement of period time differences over consecutive periods of the clock after a trigger point. The more clock cycles that can be captured, the more accurate the result will be. In the example shown, we have captured 1.6-ms worth of clock periods, 800 µs after the trigger point and 800 µs before the trigger point. This equates to 160,000 cycles of a 100-MHz clock.

From these period jitter measurements, the maximum cycle-to-cycle jitter can be derived by finding the largest deviation in period of consecutive clock cycles. The FFT of this measurement provides a plot of jitter component vs. frequency, as shown in Figure 16.13. In this plot, we see the spectrum from DC to 50 MHz. A jitter component of 33 MHz can be seen, attributable to coupling from the 33-MHz PCI outputs on the clock generator. In Figure 16.14, we have zoomed in to look at the spectrum from DC to 15 MHz. Two jitter components at 3.592 MHz and 13.83 MHz can be seen with rms values of 236.7 fs and 170.7 fs, respectively.

![Figure 16.13 Jitter Component vs. Frequency (FFT), DC to 50 MHz](image1)

![Figure 16.14 Jitter Component vs. Frequency (FFT), DC to 15 MHz](image2)
System Measurements

The measurements previously described are useful for estimating system performance and ensuring the reference clocks meet the data sheet specifications. However, the best way to evaluate performance is to measure the performance of the actual SERDES on the motherboard using the actual clock, connectors and traces. To measure a PCIe serial link, we use a compliance load board and the Tektronix Real-Time Eye (RT-Eye) software running on a TEK7404 scope.

The PCIe compliance load board plugs into the PCIe connector on the motherboard, where it brings out the serial signal (2.5 Gbaud) to SMA connectors. Connecting these to the TEK7404 scope provides a 50Ω termination. The PCIe PHY wakes up when it senses the termination and sends out a request to the other side of the link. When it receives no reply, it assumes it has been connected to a piece of test equipment and starts sending out the PCIe compliance pattern. This pattern is shown in Figure 16.15.

The RT-Eye PCIe module uses clock recovery software to recover the serial stream from this pattern and create an eye diagram. It then compares this eye diagram to the eye mask called out in the PCIe specification. The PCIe PHY is driven by the SRC clock from the motherboard clock generator, so the SRC clock will directly influence the width of the recovered eye and the jitter that is present.

To create the eye, the software takes 3500 consecutive unit intervals (UI) of the serial data stream, with a UI defined as a single bit time. The software then takes the middle 250 UI to create the eye diagram. This is done to ensure that the presence of any spread-spectrum modulation does not influence the eye. The compliance pattern ensures that all types of transitions and non-transitions are captured.

There are two specifications and eye masks, one for the transmitter and one for the receiver. These dictate what the quality of the serial signal should be at different points along the serial link, and are defined to ensure an error-free link. The measurements described here are transmitter measurements, which is defined as probing the link after the serial signal has traveled over the PCIe connector.

![Figure 16.15 PCIe Compliance Test Pattern](image)
The software creates two different eyes, one for transition bits and one for non-transition bits. The software tests the following specs on the serial link: rise/fall time, eye height, eye width, TIE (time interval error) jitter, unit interval, differential average voltage, differential peak voltage, common mode voltage, and de-emphasis. The results of the eye mask test for a passing device are shown in Figure 16.16. Table 16.2 shows the results of each of the other PCIe compliance tests.

Figure 16.16 PCIe Eye Mask Compliance Test, Transition Bits (Left), Non-Transition Bits (Right)
Of all the PCIe compliance specifications, the ones that are influenced by the performance of the SRC clock are eye width, TIE jitter, and unit interval. Eye width is the opening of the eye, defined as 70 UI for the transmitter. TIE jitter refers to the measurement variation of the serial signal crossing point in relation to the average unit interval. This refers to the width of the transition region of the eye. The specification for the transmitter is maximum 0.15 UI, which translates to 60 ps at the 2.5 Gbaud signaling rate (1 UI = 400 ps). The unit interval measurement is specified without spread spectrum enabled, since enabling the ~0.5% spread will create a unit interval on the serial link 0.5% higher than without spread. The spec is ±300 ppm, or 0.12 ps for a UI of 400 ps.

**Measuring Long-Term Jitter on the USB and Dot Clocks**

Like the SRC clocks used for PCIe and SATA, the USB 48-MHz clock is used as a reference clock for a serial link, this time for the 480-Mbps USB2.0. Serial link electrical performance is evaluated by an eye pattern test, which will be impacted by the reference clock source. Tolerances that do not meet the specification can lead to USB failures that will cause devices attached to the PC to malfunction.
The 96-MHz dot clock is used to control the external monitor VGA hsync and vsync signals. The quality of this clock will impact VGA output quality with a bad clock causing screen ripple issues and other video problems.

To measure long-term jitter on the 48-MHz USB clock, we can use a DSO such as the Tektronix 694, 7254, or the HP 54845A. The probe bandwidth needs to be $\geq 1.5$ GHz. Find the end point of USB clock trace on the motherboard, and then connect the probe tip to the endpoint, as shown in Figure 16.17.

Set up the scope using the following instructions:
1. Adjust the scale to maximum. Let the clock edge show in the whole scope screen.
2. Set the delay trigger to 125 $\mu$s.
3. Set the histogram range on the delay trigger point.
4. Let the scope trigger for more than 10K cycles.
5. Record the p-p (peak-to-peak) histogram result. This result is 48-MHz USB clock long-term jitter at 125 $\mu$s delay (Figure 16.18).

In Figure 16.19, the same measurement is shown from a clock generator that suffers from excessive long-term jitter. In this waveform, the long-term jitter at 125 $\mu$s is measured as much greater than 1 ns. This kind of clock performance will cause a USB 2.0 eye pattern test failure, as shown in Figure 16.20.
Figure 16.18 48-MHz USB Clock LTJ at 125 µs (Normal)

Figure 16.19 48-MHz USB Clock LTJ at 125 µs (Failure)
Figure 16.20 USB 2.0 Failing Eye Pattern

**Dot Clock Long-Term Jitter Measurement with Histogram**

For motherboards that do not use separate add-in graphic cards, the hsync output from the VGA connector is derived by the Northbridge/MCH directly from the 96 MHz dot clock. This section will show the correlation between the long-term jitter on the dot clock and the jitter present on the hsync output.

The VESA Video Signal Standard specification Version 1, Rev. 1 describes the method for testing the hsync output at the VGA connector output of the motherboard. The measurement is officially defined by accumulating 100,000 samples of rising edge to rising edge period (or falling edge to falling edge), with the edge defined as a transition through 1.5V. The peak-to-peak value is obtained by subtracting the minimum period from the maximum period of all the samples. Divide this value by two to get the specification value, which is essentially “peak.” The maximum value allowed for this is 15% of the pixel frequency, a frequency that changes depending on the screen resolution and refresh rate.

The highest resolution and refresh rate currently used is 1600 x 1200 at 85 Hz. This is the test target used for many major manufacturers. This resolution and refresh rate correspond to a maximum jitter of 650 ps on the hsync output. Note that this is the long-term jitter divided by 2, so it actually equates to a long-term jitter of 1.3 ns on the hsync output. The frequency of the hsync output at this resolution setting is 106.3 KHz. Measuring at the next edge from the trigger point equates to a delay of 9.65 µs. We have verified experimentally that the cycle-to-cycle jitter on hsync directly correlates with long-term jitter on the dot
clock over a measurement period equal to the hsync period. Since the most stringent hsync jitter requirement comes at a period of 9.65 µs, we will use this hsync jitter requirement at an long-term jitter measurement period of 10 µs (rounded up for simplicity and margin).

The measurement process of long-term jitter on the 96-MHz dot clock is similar to the long-term jitter measurement on the USB clock. Suggested oscilloscopes are the Tektronix 694, Tektronix 77254, or the HP 54845A with a probe bandwidth ≥1.5 GHz. First we need to find the endpoint of the 96-MHz dot clock traces on the motherboard. Because the 96-MHz dot clock is a differential pair, either two single-ended probes (Figure 16.21) or a single differential probe (Figure 16.22) can be used. When using two single-ended probes, they both need to be calibrated and deskewed. Once this is done, a mathematic subtraction is used to obtain the differential measurement from the two single-ended probes.
The process for measuring dot clock long-term jitter is as follows:

1. Adjust the scale to the maximum. Let the waveform (differential or subtracted single-ended) show in the whole scope screen.

2. Set the delay trigger to 10 µs.

3. Set the histogram range on the delay trigger point.

4. Wait until the scope triggers >100K cycles.

5. Record the peak-to-peak histogram result. This result is 96-MHz dot clock long-term jitter at 10 µs delay.

A good clock waveform is shown in Figure 16.23, while Figure 16.24 shows a waveform that fails the specification for long-term jitter. In this waveform, the long-term jitter at 10 µs is measured at about 2 ns. If we measure the jitter between adjacent hsync pulses on the VGA connector in a system using this clock, we will see VESA defined jitter of about 2 ns accordingly. This not only violates the VESA specification, but can also cause screen ripple noise on the CRT.

---

**Figure 16.23** 96-MHz Dot Clock LTJ at 10 µs (Normal)

**Figure 16.24** 96-MHz Dot Clock LTJ at 10 µs (Failure)
Conclusion

The clocks on a motherboard help it perform complicated tasks seamlessly between many different components. The many increases in the external performance of motherboard functions demand proportional increases in clock performance. The specifications for the various clocks (CPU, SRC, USB, and dot clocks) are important, not only for qualification purposes, but for the proper functioning of all the systems that the clocks supply. The measurement techniques discussed in this chapter can be used to ensure that your motherboard or system that uses similar clocks are of sufficient quality to guarantee adequate performance.
### Glossary

<table>
<thead>
<tr>
<th>Word</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anechoic</td>
<td>Without echo</td>
</tr>
<tr>
<td>Anechoic chamber</td>
<td>A test room designed to have minimal reflections by using highly absorbent materials. Used as the facility to detect and measure electromagnetic emission.</td>
</tr>
<tr>
<td>ANSI</td>
<td>American National Standards Institute</td>
</tr>
<tr>
<td>Attenuation</td>
<td>Process to reduce the amplitude of a signal.</td>
</tr>
<tr>
<td>B</td>
<td>Byte; a sequence of adjacent bits, usually eight.</td>
</tr>
<tr>
<td>b</td>
<td>Bit; a fundamental unit of information having just two possible values, as either of the binary digits 0 or 1.</td>
</tr>
<tr>
<td>baud</td>
<td>The unit that measures the information carrying capacity or “signaling rate” of a communication channel. One baud is one symbol per second. This coincides with bits per second only for two-level modulation with no framing or stop bits.</td>
</tr>
<tr>
<td>Bimodal</td>
<td>Having two modes. In statistics refers to having two peaks in a distribution.</td>
</tr>
<tr>
<td>Blind via</td>
<td>A plated through-hole connecting an outer layer to one or more internal layers of a multilayer printed circuit board. It does not extend fully through all of the layers of base material of the board.</td>
</tr>
<tr>
<td>Buried Microstrip</td>
<td>Type of PCB construction; The trace signal is on an inner layer surrounded by a dielectric with a reference plane below and air above.</td>
</tr>
<tr>
<td>Buried via</td>
<td>A plated through-hole connecting an inner layer to one or more internal layers of a multilayer printed circuit board.</td>
</tr>
<tr>
<td>C</td>
<td>See Celsius</td>
</tr>
<tr>
<td>c</td>
<td>The speed at which light travels in a vacuum, which is equal to 299,792,458 meters per second.</td>
</tr>
<tr>
<td>Term</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Celsius</td>
<td>A unit of temperature similar to the centigrade scale. Temperature on the Celsius scale is the temperature on the Kelvin scale minus 273.15. This definition makes values on the Celsius and centigrade scale agree to within less than 0.1 degree. The current official definition of Celsius sets 0.01°C to be at the triple point (ice, liquid, vapor) of water and a degree to be 1/273.16 of the difference in temperature between the triple point of water and absolute zero (absence of heat energy). Named after Swedish astronomer Anders Celsius.</td>
</tr>
<tr>
<td>Centigrade</td>
<td>A unit of temperature defined as having a thermometric scale having as fixed points the temperature of melting ice (0°) and the vapor of distilled water boiling (100°) at standard atmospheric pressure. Now an obsolete term as Celsius is the accepted standard and a more precise measurement.</td>
</tr>
<tr>
<td>CISPR</td>
<td>International Special Committee On Radio Interference; EMI standards body for Europe</td>
</tr>
<tr>
<td>CMR</td>
<td>Common Mode Rejection</td>
</tr>
<tr>
<td>Common Mode</td>
<td>Voltage or current that is common to both sides of a differential pair signal</td>
</tr>
<tr>
<td>dB</td>
<td>Decibel; a unit for representing the ratio of the magnitudes of two voltages or currents equal to 20 times the logarithm of the voltage or current ratio.</td>
</tr>
<tr>
<td>Dielectric</td>
<td>An insulating material that lies between two or more conducting materials.</td>
</tr>
<tr>
<td>Dielectric Constant</td>
<td>Sometimes called relative permittivity; A constant that is the ratio of a material’s permittivity to that of a vacuum. See permittivity.</td>
</tr>
<tr>
<td>Differential Mode</td>
<td>Voltage or current that is unique to one signal of a differential pair signal.</td>
</tr>
<tr>
<td>Droop</td>
<td>A momentary drop in supply voltage typically due to switching signals.</td>
</tr>
<tr>
<td>DSO</td>
<td>Digital Storage Oscilloscope; Test Equipment.</td>
</tr>
<tr>
<td>DUT</td>
<td>Device Under Test; refers to any element on which a measurement is taken.</td>
</tr>
<tr>
<td>Duty Cycle</td>
<td>A ratio of the high time and the low time of a digital signal. Usually expressed in terms of a percentage. A 50% duty cycle waveform has one-half the period high and one-half low.</td>
</tr>
<tr>
<td>EBD</td>
<td>Electronic Board Description EIA Electronic Industries Alliance</td>
</tr>
<tr>
<td><strong>EMC</strong></td>
<td>Electromagnetic Compatibility; the ability a device can exist within its environment.</td>
</tr>
<tr>
<td><strong>EMI</strong></td>
<td>Electromagnetic Interference; the process by which energy is transmitted.</td>
</tr>
<tr>
<td><strong>ESL</strong></td>
<td>Effective Series Inductance; the total inductance of a capacitor including the DC and AC components.</td>
</tr>
<tr>
<td><strong>ESR</strong></td>
<td>Effective Series Resistance; the total resistance of a capacitor including the DC and AC components.</td>
</tr>
<tr>
<td><strong>Extrinsic Skew</strong></td>
<td>Skew generated outside a device usually from layout or design parameters.</td>
</tr>
<tr>
<td><strong>F</strong></td>
<td>Farad; The unit of capacitance in the ‘meter-kilogram-second’ system equal to the capacitance of a capacitor having a charge of 1 coulomb when a potential difference of 1 volt is applied. Named after British physicist Michael Faraday.</td>
</tr>
<tr>
<td><strong>FCC</strong></td>
<td>Federal Communications Commission</td>
</tr>
<tr>
<td><strong>FET</strong></td>
<td>Field Effect Transistor</td>
</tr>
<tr>
<td><strong>FFT</strong></td>
<td>Fast Fourier Transform; given a finite set of data points, for example a periodic sampling taken from a real-world signal, the FFT expresses the data in terms of its component frequencies. Named after J. B. Joseph Fourier.</td>
</tr>
<tr>
<td><strong>fs</strong></td>
<td>Femtosecond; one quadrillionth (10^-12) of a second.</td>
</tr>
<tr>
<td><strong>FTG</strong></td>
<td>Frequency Timing Generator; a Device that generates seemingly unrelated output frequencies.</td>
</tr>
<tr>
<td><strong>Gaussian Distribution</strong></td>
<td>A bell shaped curve representing a statistical distribution. Named after German mathematician and astronomer Karl Friedrich Gauss.</td>
</tr>
<tr>
<td><strong>GHz</strong></td>
<td>Gigahertz; a unit of frequency equal to one billion (10^9) cycles per second.</td>
</tr>
<tr>
<td><strong>GTL</strong></td>
<td>Gunning Transceiver Logic; a low-level, high-speed interface standard for digital integrated circuits.</td>
</tr>
<tr>
<td><strong>H</strong></td>
<td>Henry; The unit of inductance in which an induced electromotive force of one volt is produced when the current is varied at the rate of one ampere per second. Named after physicist Joseph Henry.</td>
</tr>
<tr>
<td><strong>Histogram</strong></td>
<td>A graph representing the frequency of an occurrence.</td>
</tr>
<tr>
<td><strong>HSTL</strong></td>
<td>High Speed Transceiver Logic; a JEDEC specification for an electrical interface.</td>
</tr>
</tbody>
</table>
**Hz**
Hertz; a unit of frequency equal to one cycle per second. Named after German physicist Heinrich Hertz.

**IBIS**
Input/Output Buffer Information Specification; a behavioral model of a device representing its input and output structures.

**IEC**
National Committees of the International Electro Technical Commissions

**Impedance**
Similar to resistance, but a measurement of the AC opposition to current flow.

**Intrinsic Skew**
Skew belonging to the device by its very nature

**JEDEC**
Joint Electron Device Engineering Council; the semiconductor standardization body of the EIA.

**Jitter**
The deviation of a clock's output from its ideal position.

**Jitter Generation**
The effect of a PLL added noise to the output clock signal.

**Jitter Transfer**
Amount of jitter passed from the input to the output of a device or PLL.

**Kelvin**
A temperature scale where zero is the complete absence of heat energy. The Kelvin is the same size as the Celsius degree. Note that “degree” is omitted when using the Kelvin scale. Named after the British physicist William Thomson Kelvin.

**KHz**
Kilohertz; a unit of frequency equal to one thousand cycles per second.

**LVCMOS**
Low Voltage Complementary Metal Oxide Semiconductor

**LVDS**
Low Voltage Differential Signaling

**LVPECL**
Low Voltage Positive Emitter Coupled Logic;

**LVTTL**
Low Voltage Transistor Transistor Logic;

**Mean**
For statistics, it is short for arithmetic mean. Also referred to as the average.

**MHz**
Megahertz; a unit of frequency equal to one million ($10^6$) cycles per second.

**Micron**
Unit of measure equal to one millionth of a meter (metric).

**Microstrip**
Type of PCB construction; the trace signal is on an outer layer with a dielectric separating the trace from a reference plane.

**Mil**
Unit of length equal to 0.001 of an inch.

**MLF**
Micro Lead-Frame
<p>| <strong>mm</strong> | Millimeter; a unit of length equal to one thousandth ($10^{-3}$) of a meter (0.0394 inch). |
| <strong>mW</strong> | Milliwatt; a unit of power equal to one thousandth ($10^{-3}$) of a Watt. |
| <strong>Modulate</strong> | To vary the amplitude, frequency, phase, or intensity of a signal with another signal |
| <strong>MOSFET</strong> | Metal Oxide Semiconductor Field Effect Transistor |
| <strong>ms</strong> | Millisecond; one thousandths ($10^{-3}$) of a second. |
| <strong>Multimodal</strong> | Having more than one mode. In statistics, refers to having more than one peak in a distribution. |
| <strong>nH</strong> | Nanohenry; one billionth ($10^{-9}$) of a Henry. See H. |
| <strong>ns</strong> | Nanosecond; one billionth ($10^{-9}$) of a second. |
| <strong>Normal Distribution</strong> | See Gaussian Distribution |
| <strong>PCB</strong> | Printed Circuit Board |
| <strong>Peak-to-peak</strong> | Includes measurement data for the maximum values. Used in measuring jitter. |
| <strong>Permittivity</strong> | A measure of the ability of a material to resist the formation of an electric field within it. Also called dielectric constant. |
| <strong>Phase Error</strong> | Skew between the edges of two clock signals. |
| <strong>Phase Noise</strong> | Short term frequency fluctuation of a signal. |
| <strong>PLCC</strong> | Plastic Leaded Chip Carrier or Plastic Leadless Chip Carrier |
| <strong>PLL</strong> | Phase-locked Loop, a function that locks a reference frequency to the phase of an input. |
| <strong>Polyimide</strong> | Thermoplastic used with glass to produce printed circuit laminates. |
| <strong>ppm</strong> | Parts per million |
| <strong>Prepeg</strong> | Material used in PCBs consisting of a base material impregnated with a synthetic resin, such as epoxy or Polyimide. |
| <strong>ps</strong> | Picosecond; one trillionth ($10^{-12}$) of a second. |
| <strong>PWB</strong> | Printed Wiring Board |
| <strong>QFN</strong> | Quad Flatpack No-lead; a type of semiconductor device package characterized by a small footprint and small connecting pins. |
| <strong>QFP</strong> | Quad Flat Pack |
| <strong>RF</strong> | Radio Frequency, the frequency range, roughly from 10 kHz to 100 GHz, used in communications. |
| <strong>RMS</strong> | Root Mean Square |
| <strong>SAS</strong> | Serial Attached SCSI |
| <strong>SATA</strong> | Serial Advanced Technology Attachment; a hard disk interface. |
| <strong>SCSI</strong> | Small Computer System Interface |
| <strong>Shunt</strong> | To divert part or all of a current by connecting a circuit in parallel with another. |
| <strong>Sigma</strong> | See Standard Deviation |
| <strong>Skew</strong> | The variation in the arrival time of two signals specified to occur at the same time. |
| <strong>Slew Rate</strong> | The rate of change of a signal. |
| <strong>SOIC</strong> | Small-Outline Integrated Circuit |
| <strong>SPICE</strong> | Simulation Program with Integrated Circuit Emphasis; detailed simulation model of a circuit or device. |
| <strong>SSOP</strong> | Shrink Small Outline Package |
| <strong>Spread Spectrum</strong> | A technique that spreads a signal bandwidth over a wide range of frequencies. |
| <strong>SST</strong> | Spread Spectrum Technology; also known as Clock Modulation. |
| <strong>Standard Deviation</strong> | Measurement of the spread of data about the mean value. One Standard Deviation contains 68.26% of the data on one side of the mean. |
| <strong>Stripline</strong> | Type of PCB construction; the trace signal is on an inner layer surrounded by a dielectric with a reference plane above and below. |
| <strong>Thevenin</strong> | Theorem by M.L. Thevenin, a French engineer, to rearrange any linear circuit into two networks connected together. Often used to describe a termination technique that is rearranged to provide an equivalent circuit. |
| <strong>TIA</strong> | Timing Interval Analyzer; test equipment. |
| <strong>TQFP</strong> | Thin Quad Flat Pack |
| <strong>Tracking</strong> | Ability for a PLL to follow the phase variations of an input signal. |
| <strong>TSSOP</strong> | Thin Shrink Small Outline Package |
| <strong>TTB</strong> | Total Timing Budget |</p>
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>UI</td>
<td>Unit Interval; equal to one period of a clock waveform.</td>
</tr>
<tr>
<td>µF</td>
<td>Microfarad; one millionth (10^-6) of a Farad. See F.</td>
</tr>
<tr>
<td>µs</td>
<td>Microsecond; one millionth (10^-6) of a second.</td>
</tr>
<tr>
<td>VCC</td>
<td>Collector supply voltage used in Bipolar technology. Often misused for VDD.</td>
</tr>
<tr>
<td>VCCI</td>
<td>Voluntary Control Council for Interference by Information Technology Equipment; EMI standards body for Japan.</td>
</tr>
<tr>
<td>VCO</td>
<td>Voltage Controlled Oscillator; frequency output is proportional to the value of a control voltage.</td>
</tr>
<tr>
<td>VDD</td>
<td>Drain supply voltage; power supply (Vd is the drain voltage of a FET).</td>
</tr>
<tr>
<td>VESA</td>
<td>Video Electronics Standards Association</td>
</tr>
<tr>
<td>via</td>
<td>A plated through-hole connecting an outer layer to one or more internal layers of a multilayer printed circuit board.</td>
</tr>
<tr>
<td>VSS</td>
<td>Source Supply Voltage—Usually 0 volts (ground) in digital circuits (Vs is the source voltage of a FET).</td>
</tr>
<tr>
<td>Vtt</td>
<td>Termination Voltage</td>
</tr>
<tr>
<td>W</td>
<td>See Watt: Watt A unit of power equal to one joule per second. Named after British engineer James Watt.</td>
</tr>
<tr>
<td>ZDB</td>
<td>Zero Delay Buffer; a PLL-based clock buffer that has a nominal zero delay from the input reference clock to the output clock.</td>
</tr>
</tbody>
</table>
References

Application notes for Surface Mount Assembly of Amkor’s MicroLeadFrame (MLF) package, Amkor.

George Chien and Paul R. Gray, “TP 12.4: A 900-MHz Local Oscillator Using a DLL-based Frequency Multiplier Technique for PCS Applications,” University of California, Berkeley, CA.


IPC IPC-SM-782: Surface Mount Design and Land Pattern Standard

JEDEC J-STD-020B: Moisture/Reflow Sensitivity Classification for Nonhermetic Solid State Surface Mount Device


“Power distribution system design methodology and capacitor selection for modern CMOS technology,” Smith, et. al., Published by Sun Microsystems, 1999.

Right the First Time: A practical handbook on high speed PCB and system design, Lee W. Ritchey Vol. 1, Speeding Edge, 2003


“T11.2/Project 1316-DT,” a proposed draft of ASC of the National Committee for Information Technology Standardization (NCITS), reference work done by and for the Secretariat of the NCITS.
