

# Application Note

## PCIe Link Training Overview

---



Jae Byeok Yoon and Nicholas Malone

### ABSTRACT

This application note explores the basic history, concept, link training and link equalization processes of the PCIe interface. This document is based on TI Precision Labs' "What is PCIe?" video. For a video version of this content, see [What is PCIe?](#).

---

### Table of Contents

<b>1 Introduction</b>	<b>2</b>
<b>2 History</b>	<b>2</b>
<b>3 Components of PCIe Communication</b>	<b>3</b>
3.1 Root Complex	3
3.2 Repeater	3
3.3 Endpoints	3
<b>4 Signaling</b>	<b>4</b>
4.1 PERST	4
4.2 WAKE and CLKREQ	4
4.3 REFCLK	4
<b>5 Link Training</b>	<b>4</b>
5.1 Receiver Detect (Rx Detect)	5
5.2 Polling	5
5.3 Configuration	6
<b>6 Link Equalization</b>	<b>7</b>
6.1 Phase 0 and 1	8
6.2 Phase 2 and 3	8
<b>7 Summary</b>	<b>9</b>
<b>8 References</b>	<b>9</b>

### List of Figures

Figure 2-1. Timeline of PCIe Interface	2
Figure 3-1. PCIe Topology	3
Figure 4-1. PCIe Control Signals	4
Figure 5-1. PCIe Link Receiver Detection	5
Figure 5-2. PCIe Link Polling	5
Figure 5-3. PCIe Link Configuration	6
Figure 6-1. Link Equalization Preset Values and Signals	7
Figure 6-2. Link Equalization Phase 0 and 1	8
Figure 6-3. L0 State of PCIe Devices After Link Equalization	8

### Trademarks

All trademarks are the property of their respective owners.

## 1 Introduction

Peripheral Component Interconnect Express (PCIe) is a motherboard expansion bus standard introduced in 2003 to enable high-speed serial communication between the Central Processing Unit (CPU) and its peripheral components. Today, it has become the primary motherboard expansion bus standard and a popular communication method for many other onboard applications. PCIe is often used for Graphics Processing Unit (GPU) and solid-state drives (SSD) to send and receive data with the CPU.

## 2 History

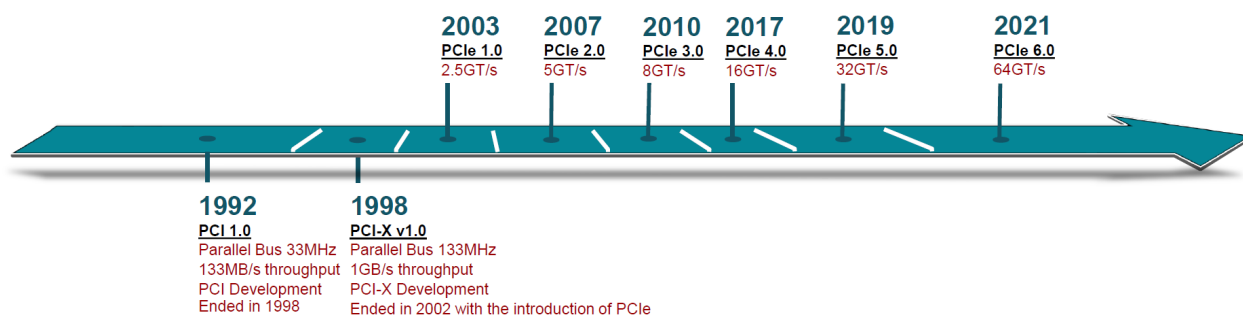
PCIe is based on the predecessor PCI. The PCI bus existed on many motherboards in the 1990s, along with a few other expansion bus technologies. Motherboard expansion bus standards are designed for communication between the CPU and devices plugged into the motherboard's expansion slots. Initially, all expansion bus standard used a parallel bus, which means that data is sent and received over multiple channels.

At the time of its introduction in 2003, the PCIe serial bus standard was meant to replace these older parallel buses to enable a higher data rate and to simplify system design. In 2003, the PCIe standard was defined by the PCI-SIG organization. Since then, the PCIe standard has iteratively improved over time to accommodate the latest bandwidth needs of modern computers. In 2021, the PCIe 6.0 specification was introduced, enabling 64 GT/s, or 64 Gbps per link. One unique feature of the PCIe standard is the ability to increase the number of lanes from 1 to 32 lanes to increase its throughput, a feature inspired by its parallel bus predecessor. A PCIe 6.0 link that is 16 lanes wide would have a data rate of 128 GB/s, which is extremely fast by today's standards. [Figure 2-1](#) shows the timeline of PCIe development and its data rate over the generations along with its predecessors' data rate.

Using the parallel bus feature, PCIe can establish link with other PCIe devices in link width of 1, 2, 4, 8, 16, and even 32 lanes as defined in the PCIe standard. Different link width allows PCIe devices to transmit more data by using more lanes or vice versa as needed by the machine. Having different link width also allows you to bifurcate, or split, the lanes that are used for the device. In doing so, you can attach multiple PCIe devices onto one PCIe slot.

Since the development, PCIe standard has been used for variety of form factors. One of the more popular form factors besides CEM (Card Electromechanical) is M.2. M.2 is a replacement interface for mSATA and Mini PCIe and is mainly used for SSD connection to the CPU.

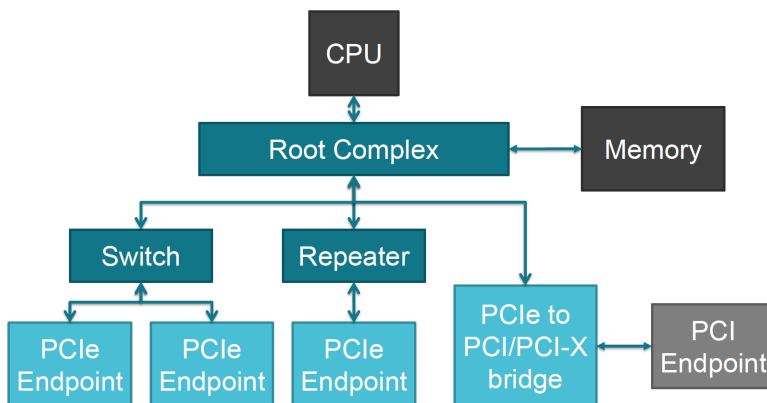
# Peripheral Component Interconnect Express



**Figure 2-1. Timeline of PCIe Interface**

### 3 Components of PCIe Communication

PCIe communication consists of three main components: root complex, repeaters, and PCIe endpoints. PCIe communication is hierarchical so there is a single source, which is the root complex, through which all the data passes. The data goes to the root complex from multiple PCIe endpoints and vice versa. This hierarchy is shown in [Figure 3-1](#).



**Figure 3-1. PCIe Topology**

#### 3.1 Root Complex

A root complex is the interface between the system CPU, memory, and the rest of the PCIe interface. The root complex is either integrated into the CPU directly, or is external to the CPU as a discrete component. This interface also acts as a single source where all the data from various PCIe endpoints pass through. [Figure 3-1](#) shows it as the dark blue box that connects CPU, memory, and PCIe components and is referred to as "Root Complex."

#### 3.2 Repeater

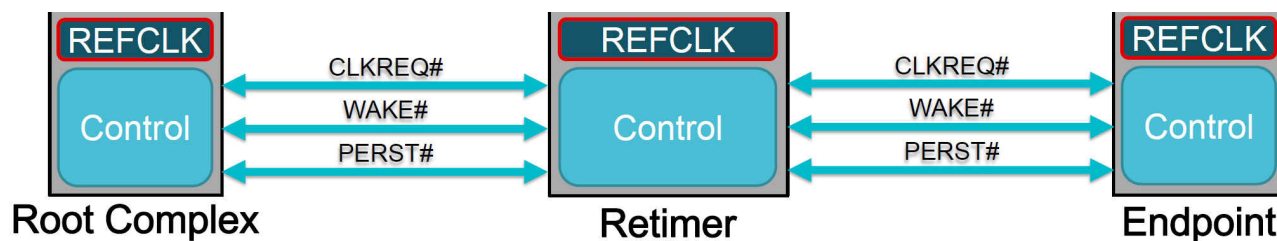
A repeater is a signal conditioning device that ensures a good signal to reach to and from the root complex and PCIe endpoints. Repeaters can fall into two categories: retimers and redrivers. Both are common PCIe components used to maintain signal quality of high speed links and compensate for the loss of signal quality over the traces. [Figure 3-1](#) shows it as the dark blue box that connects the root complex and PCIe endpoint and is referred to as "Repeater."

#### 3.3 Endpoints

An endpoint is a general term for a PCIe end component. This could represent many different types of PCIe devices such as M.2 solid state drive (SSD) or graphics processing unit (GPU). It could be either a PCIe component or a PCI component with PCIe to PCI/PCI-X bridge. [Figure 3-1](#) shows it as light blue box and gray box that is connected to either bridge, switch, or repeater and is referred to as either "PCIe Endpoint" or "PCI Endpoint."

## 4 Signaling

Each component of PCIe communication (except for redrivers) have the following control signals: PERST, WAKE, CLKREQ, and REFCLK. These signals work to generate high-speed signals and communicate with other PCIe devices. [Figure 4-1](#) shows the diagram of PCIe devices with the control signals. This diagram shows that all of the control signals except REFCLK are active low signals.



**Figure 4-1. PCIe Control Signals**

### 4.1 PERST

PERST is referred to as a fundamental reset. PERST should be held low until all the power rails in the system and the reference clock are stable. A transition from low to high in this signal usually indicates the beginning of link initialization. In [Figure 4-1](#), it is referred as "PERST#."

### 4.2 WAKE and CLKREQ

WAKE and CLKREQ signals are both used for transitioning to and from low power states. WAKE signal is an active-low signal that is used to return the PCIe interface to an active state when in a low-power state. CLKREQ signal is also an active-low signal and is used to request the reference clock. In [Figure 4-1](#), these are referred as "WAKE#" and "CLKREQ#", respectively.

### 4.3 REFCLK

A REFCLK, or reference clock signal, is a prerequisite for a PCIe device to begin data transmission. This 100 MHz reference clock signal is used by the PCIe device to generate the high-speed PCIe data within the link and is shared by the PCIe devices within the link. In [Figure 4-1](#), it is referred as "REFCLK."

## 5 Link Training

When all devices are powered and have a reference clock provided, a PCIe device starts the link training process. The link training process consists of receiver detection (Rx detect), polling, and configuration. After this process, PCIe devices are connected from the endpoint to the root complex.

## 5.1 Receiver Detect (Rx Detect)

The first step in link training is receiver detection (Rx detect). Once all the devices are powered and have a reference clock provided, the devices start the Rx detect circuit on each lane that allows the device to determine if the device has a link partner to pair with. Figure 5-1 shows the diagram of PCIe devices trying to identify other PCIe devices to connect and send or receive data. Assuming that the PCIe Rx detect circuit sees the other device, each individual lane begins to transmit serial data at 2.5 Gbps. This is the lowest and most fundamental PCIe data rate, which was specified in the original PCIe Gen 1 specification. PCIe 1.0, also called PCIe Gen 1, is compatible with any PCIe device. So, every PCIe link begins with the same link initialization process. This also means that any PCIe device can transfer data at 2.5 Gbps if they can only form link at PCIe 1.0 data rate.

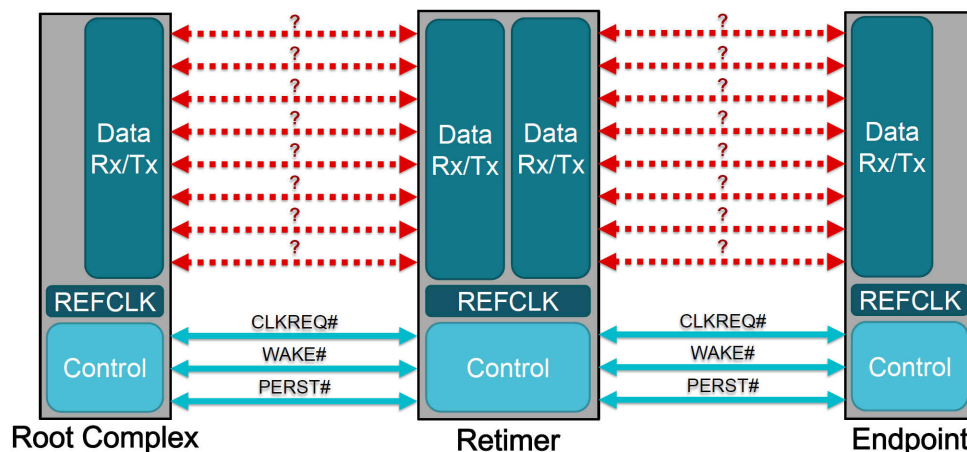


Figure 5-1. PCIe Link Receiver Detection

## 5.2 Polling

After Rx detect stage is done and each lane is transmitting data, PCIe link will start polling. Polling is a stage in which the root complex, repeater (referred as retimer in Figure 5-2), and the endpoint all begin transmitting ordered sets of data called training sequences at PCIe Gen 1 speeds in order to establish bit and symbol lock. Bit lock refers to when the receiver locks the clock frequency of the transmitter. Symbol lock refers to when the receiver is able to decode the valid 10-bit symbol coming from the transmitter. Figure 5-2 shows polling as red arrows with square signals pointing at PCIe devices. At the end of this process, each device is able to interpret the received data and respond accordingly and then proceed into the configuration stage.

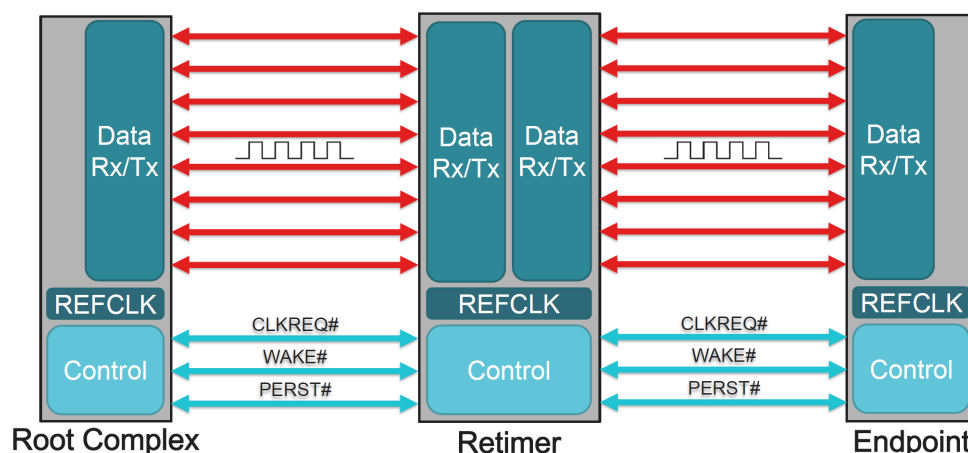


Figure 5-2. PCIe Link Polling

## 5.3 Configuration

In the configuration state, a lane-to-lane deskew process takes place in which any misalignment in the data due to varying channel length is compensated for. The PCIe link width is also determined at this stage. At the end of this process, each lane is associated with a specific link number, and a lane number within that link. Figure 5-3 represents the configuration stage with red arrows showing the exchange of signals in order to align data signals. If there are multiple links, the PCIe connection as referred to as bifurcated. Because Figure 5-3 shows a single non-bifurcated connection, all lanes are assigned to link number 0. With bifurcation, the number of links will increase. For example, in the case of x8x8 (2 x 8 lane endpoints) bifurcation, the link number is 0 for the first 8 lanes and 1 for the next 8 lanes. In addition, the link can be split in two parts due to the PCIe retimer: root complex to the retimer and retimer to the endpoint as shown in Figure 5-3 with the retimer in the middle. The link on both sides of the retimer undergo link initialization separately.

After determining link width and lane numbers, the PCIe link can move into a number of states. The system in Figure 5-3 moves into what is called the L0 state, which is the normal operational state where data and packets are sent and received. Once the L0 state is reached, the root complex and endpoint can successfully communicate between each other. Alternatively, the PCIe link could transition into a number of low-power states or into another link training state called recovery.

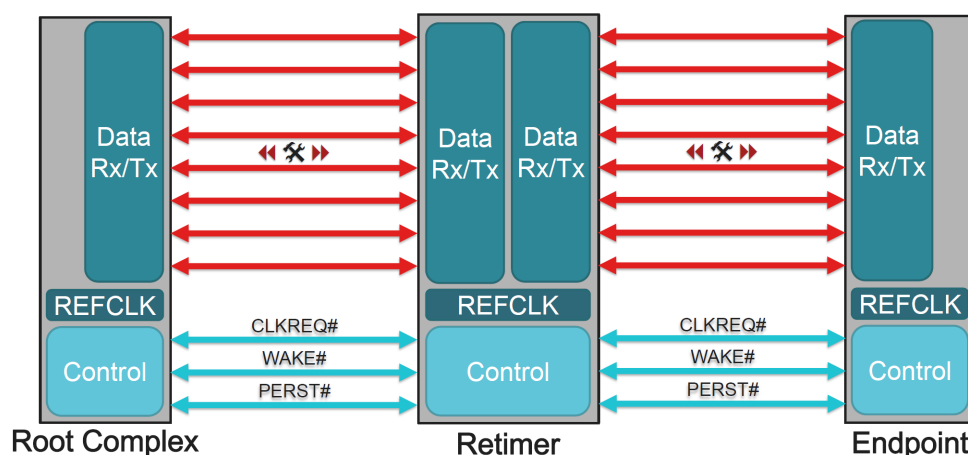
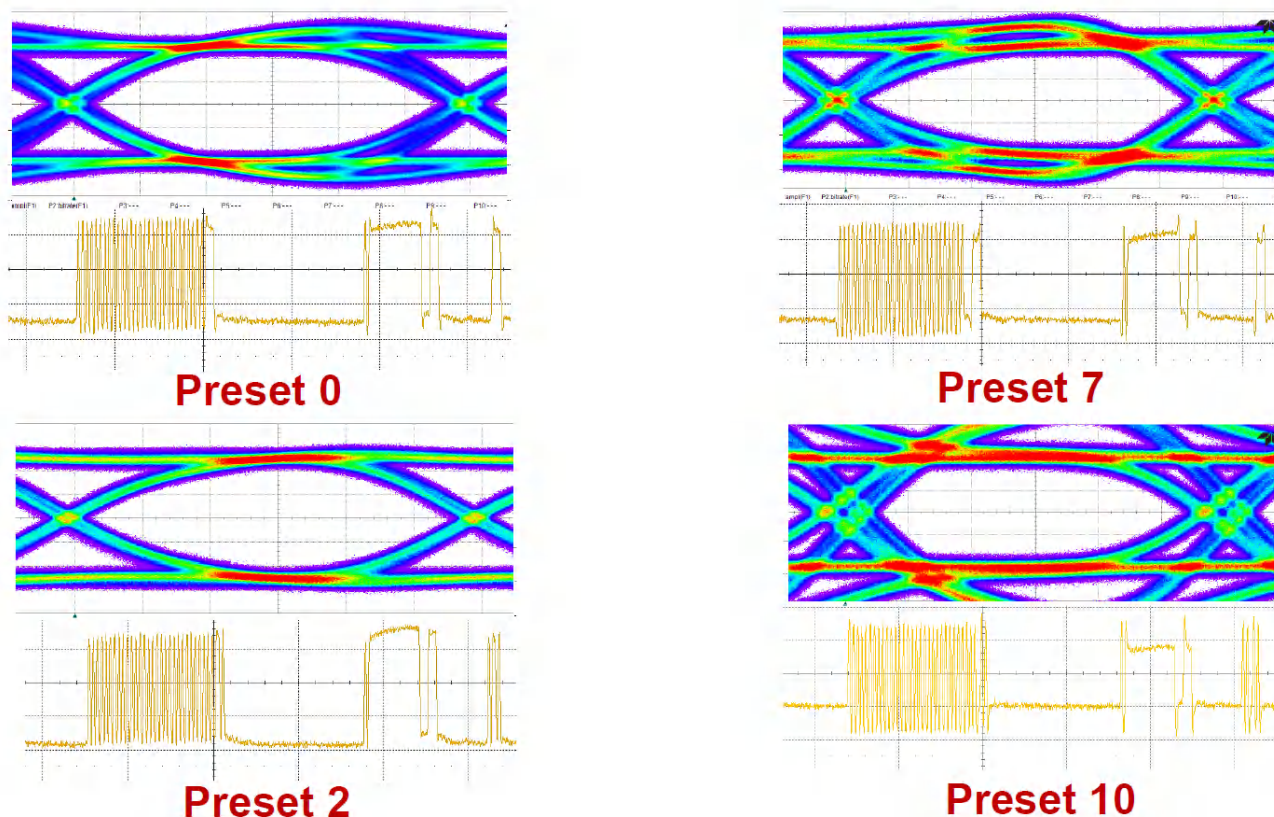


Figure 5-3. PCIe Link Configuration



## 6 Link Equalization

After link training, all PCIe devices may go through additional **link equalization processes** to establish stable connection among the devices. Link equalization is a link optimization process that modifies the characteristics of the transmitted data waveform for each part so that it results in the most stable PCIe link at a higher data rate. It happens when all devices in PCIe link can support data rates of PCIe Gen 3 or higher. Link equalization may happen multiple times since PCIe connection has to optimize the connection at every generation of PCIe above Gen 3. For example, if all PCIe devices are Gen 5, there will be three link equalization processes: first to Gen 3, from Gen 3 to Gen 4, and from Gen 4 to Gen 5. Link equalization is achieved by using the preset values defined in PCIe specification. Preset values are configurations that can modify the characteristics of the transmitted data wave form. Figure 6-1 shows select preset values in waveform and eye diagram.

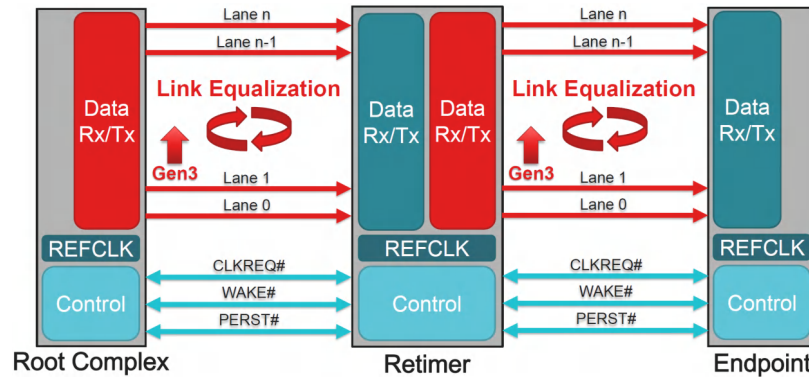


**Figure 6-1. Link Equalization Preset Values and Signals**

For Gen 3 and Gen 4, there are 11 presets numbered from 0 to 10 that may be used, each with its own unique signal characteristics. The preset values for each port are negotiated through link equalization until the ideal preset is chosen via phases 0, 1, 2, and 3 for all link equalization processes.

## 6.1 Phase 0 and 1

Phase 0 is the first phase of link equalization. This phase starts when the downstream port sends desired transmitter preset values for each lane to the upstream device. Shortly after receiving the downstream port's request, the upstream port increases the data rate of the link to Gen 3 data rate and begins transmitting training sequences back to the downstream port using the desired presets. Link equalization moves to phase 1 once the connection with Gen 3 is achieved. Figure 6-2 shows the phase 0 of link equalization from Gen 1 to Gen 3 connection with red arrows of each lane pointing the downstream device to represent the upstream device transmitting desired preset values. The link's data rate increases to the data rate of PCIe Gen 3.

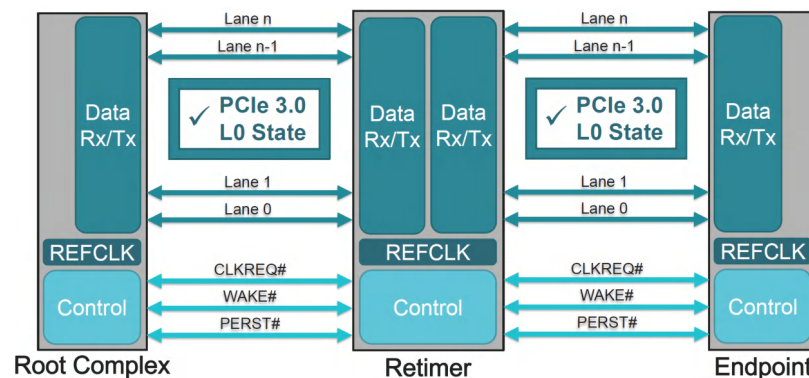


**Figure 6-2. Link Equalization Phase 0 and 1**

In phase 1, identical training sequences are sent repeatedly to ensure the correct presets are received, despite the possibility of poor link quality. This is done in order to optimize the link enough to be able to exchange training sequences and complete the remaining link equalization phases for fine tuning. Link equalization moves to phase 2 when the link has achieved a link with a bit error rate (BER) of less than  $10^{-4}$ .

## 6.2 Phase 2 and 3

In phase 2 and 3, link equalization conducts fine tuning on the link. This further optimizes the preset values for the upstream port. Then, in phase 3, same optimization happens for the downstream ports. After completing Phase 3 of the link equalization process, link equalization is completed, and the PCIe link should have BER less than  $10^{-12}$ . In some motherboard designs, particularly those with long channel links, this level of signal quality is not possible. Additional signal conditioning may be required. In this case, repeaters like redrivers and retimers are used to conduct signal conditioning and provide high-quality signal between the endpoints and the root complex. The link now moves into L0 state in Gen 3 and can communicate reliably at that speed. Figure 6-3 shows the diagram of PCIe devices connected in PCIe Gen 3 data rate after link equalization that started with Gen 1 connection. For connection at higher data rate, PCIe devices have to go through more link equalization processes.



**Figure 6-3. L0 State of PCIe Devices After Link Equalization**



## 7 Summary

PCIe is an expansion bus that can communicate between CPU and various PCIe devices. It is a high-speed signal interface that can communicate up to 128 GT/s in PCIe 7.0. PCIe devices go through the link training process to establish connection among the root complex and the PCIe endpoints. This allows PCIe devices to send and receive data at PCIe Gen 1 data rate. If all connected PCIe devices are higher than Gen 3, PCIe devices will conduct link equalization processes to establish PCIe link at faster rates. Link equalization goes through initial tuning and fine tuning to allow bit error rate of less than  $10^{-12}$  and send or receive data at the fastest rate it can stably support.

## 8 References

- Texas Instruments TIPL video: [What is PCIe?](#)

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](#) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
Copyright © 2022, Texas Instruments Incorporated