Circuit Emulation Service over Packets

Data Sheet

Applications

September 2003

- Circuit Emulation Service over packets
  - Leased Line support over packet network
  - Multi-Tenant Unit access concentration
- Packet switched backplane applications
- TDM backplane extension / expansion

Ordering Information

ZL50110	552 PBGA
ZL50111	552 PBGA
ZL50114	552 PBGA

-40°C to +85°C

Features

Circuit Emulation Functions

Supports the following circuit emulation services (CES) over the packet domain:

- structured, synchronous CES
- unstructured, asynchronous CES, with integral per stream clock recovery
- complies with standards for native TDM circuit emulation proposed by the IETF's PWE3 working group

TDM Interfaces

- TDM Access interface, consisting of up to 32 T1, 32 E1, 8 J2, 2 T3 or 2 E3 streams (depending on variant)
- Up to 1024 bi-directional 64 Kbit/s channels
- Interfaces either directly to LIU, via a framer, or via a TDM backplane
- Dual reference Stratum 3, 4 and 4E PLL for synchronous operation
- TDM-to-TDM loopback of TDM streams

Network Interfaces

- 3 x 100 Mbit/s MII or Dual Redundant 1000 Mbit/s GMII/PCS(TBI) Interfaces

System Interfaces

- Flexible 32 bit host CPU interface (Motorola PowerQUICC™ II compatible)
- On-chip packet memory for self-contained operation, with buffer depths of over 16 ms
- Up to 8 Mbytes of off-chip packet memory, supporting buffer depths of over 128ms

Packet Processing Functions

- Flexible, multi-protocol packet encapsulation
- Packet sequencing to allow lost packet detection
- N x 64 Kbit/s trunking of channels
- Four classes of service with programmable priority mechanisms (WFQ and SP)
- Flexible classification of incoming packets at layers 2, 3, 4, and 5
- Packet-to-packet loopback

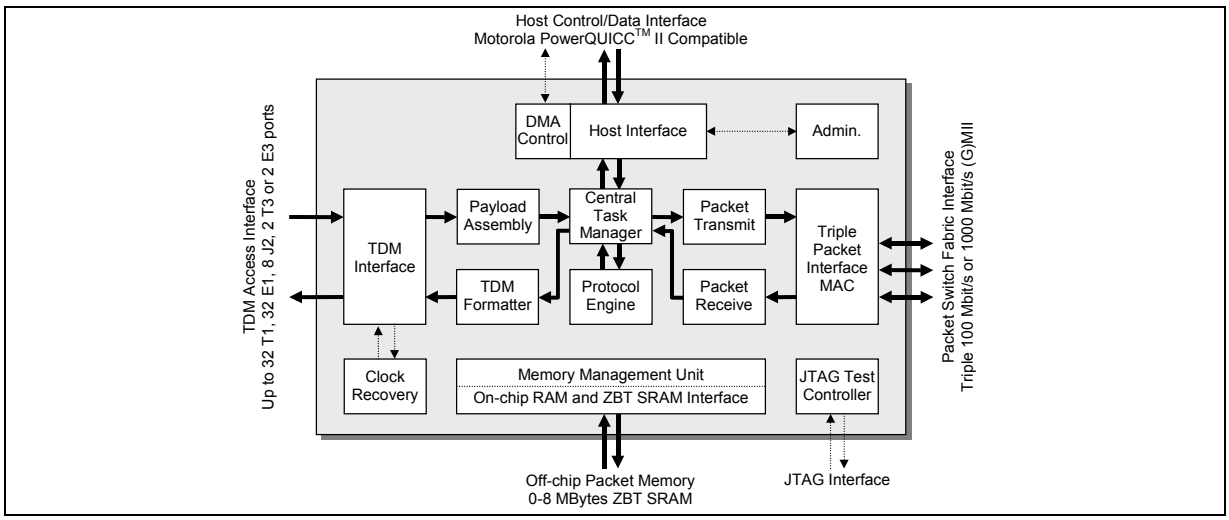


Figure 1 - ZL50110/1/4 High Level Overview

Zarlink Semiconductor Inc.
Zarlink, ZL and the Zarlink Semiconductor logo are trademarks of Zarlink Semiconductor Inc.
Copyright 2003, Zarlink Semiconductor Inc. All Rights Reserved.

Description

The ZL50110/1/4 family is a range of highly functional TDM to Packet bridging devices. It provides both structured and unstructured circuit emulation services (CES) for both T1 and E1 streams across a packet network based on Ethernet technology. In addition, it provides unstructured circuit emulation for T3 and E3 streams over the same packet network. The circuit emulation features in the ZL50110/1/4 comply with the relevant standards currently being developed within the IETF's PWE3 working group.

The ZL50110/1/4 incorporates a range of powerful clock recovery mechanisms on each TDM stream, allowing the frequency of the original clock to be regenerated. Timing is carried using RTP or similar protocols, and both adaptive and differential clock recovery schemes are included, allowing the customer to choose the correct scheme for the application.

Another application for the ZL50110/1/4 family is the provision of packet backplane interconnection. It can be used to either replace a TDM backplane entirely with an Ethernet switched backplane, or to extend the reach of a TDM backplane by carrying the TDM traffic across the Ethernet network. In conjunction with Zarlink's Ethernet switch family (MVTX2600 and MVTX2800) it provides a flexible and powerful means of distributing either individual or groups of TDM channels across a distributed system.

The ZL50110/1/4 is capable of assembling user-defined packets of TDM traffic from the TDM access interface and transmitting them from the Ethernet interfaces using a variety of protocols. It supports a range of different packet switched networks, including Ethernet VLAN’s, IP (both versions 4 and 6) and MPLS. The device also supports four different classes of service on packet egress, allowing priority treatment of TDM-based traffic. This can be used to help minimise latency variation in the TDM data.

Packets received from the Ethernet interfaces are parsed to determine the egress destination, and are appropriately queued to the TDM access interface, the host interface, or back toward the packet interface. Packets queued to the TDM interface can be re-ordered based on sequence number, and lost packets filled in to maintain timing integrity.

The ZL50110/1/4 family includes sufficient on-chip memory that external memory is not required in most applications. This reduces system costs and simplifies the design. For applications that do require more memory (e.g. high stream count or high latency), the device supports up to 8 Mbytes of SSRAM.

Device Line Up

There are three products within the ZL50110/1/4 family, with capacity as shown in the following table:

Device	TDM Interfaces	Ethernet Packet I/F
ZL50114	4 T1, 4 E1, or 1 J2 streams or 4 ST-BUS/MVIP streams at 2.048 Mbit/s or 1 ST-BUS/H.110/H-MVIP streams at 8.192 Mbit/s	Dual 100Mbit/s or Dual Gbit/s
ZL50110	8 T1, 8 E1 or 2 J2 streams or 8 ST-BUS/MVIP streams at 2.048 Mbit/s or 2 ST-BUS/H.110/H-MVIP streams at 8.192 Mbit/s	Dual 100Mbit/s or Dual Gbit/s
ZL50111	32 T1, 32 E1, 8 J2, 2 T3 or 2 E3 streams or 32 ST-BUS/MVIP streams at 2.048 Mbit/s or 8 ST-BUS/H.110/H-MVIP streams at 8.192 Mbit/s	Triple 100Mbit/s or Dual Gbit/s

Table 1 - Capacity of devices in the ZL50110/1/4 family

Table of Contents

1.0 Introduction . . . 10
1.1 Overview . . . 10
1.2 Latency . . . 11
2.0 Typical Applications . . . 11
2.1 Circuit Emulation Services over Packet Switched Networks (CESoPSN) . . . 11
2.1.1 Leased Line Provision . . . 11
2.1.2 Remote Concentration Unit . . . 12
2.1.3 Clock Recovery . . . 13
2.1.3.1 Adaptive Clock Recovery . . . 14
2.1.3.2 Differential Clock Recovery . . . 14
2.2 Packet Backplane Interconnection . . . 14
2.3 TDM Backplane Extension . . . 15
2.4 TDM Backplane Expansion . . . 16
2.5 Loopback Modes . . . 17
3.0 Physical Specification . . . 18
4.0 External Interface Description . . . 28
4.1 TDM Interface . . . 28
4.1.1 ZL50111 Variant TDM stream connection . . . 28
4.1.2 ZL50110 Variant TDM stream connection . . . 30
4.1.3 ZL50114 Variant TDM stream connection . . . 31
4.1.4 TDM Signals common to ZL50111, ZL50110 and ZL50114 . . . 32
4.2 PAC Interface . . . 33
4.3 Packet Interfaces . . . 34
4.4 External Memory Interface . . . 45
4.5 CPU Interface . . . 47
4.6 System Function Interface . . . 50
4.7 Test Facilities . . . 51
4.7.1 Administration, Control and Test Interface . . . 51
4.7.2 JTAG Interface . . . 51
4.8 Miscellaneous Inputs . . . 52
4.9 Power and Ground Connections . . . 52
5.0 Basic Operation . . . 53
5.1 TDM Access Interface . . . 53
5.2 TDM Payload Assembly . . . 53
5.3 Higher Layer Protocol Support . . . 54
5.4 Packet Transmission . . . 54
5.5 Host Packet Generation . . . 54
5.6 Packet Reception . . . 54
5.7 TDM Re-Formatting . . . 55
5.8 Data and Control Flows . . . 55
5.8.1 TM to TDM Flow . . . 56
5.8.2 TDM to TM Flow . . . 57
5.8.3 PKT to TM Flow . . . 58
5.8.4 TM to PKT Flow . . . 59
5.9 Assembling a Packet . . . 60
5.9.1 Structured Operation . . . 60
5.9.2 Unstructured Operation . . . 61
5.10 TDM Port Data Formats . . . 62
5.11 External Memory Requirement . . . 63
5.12 TDM Clock Structure . . . 64
5.12.1 Synchronous TDM Clock Generation . . . 64
5.12.2 Asynchronous TDM Clock Generation . . . 64

Table of Contents

5.13 GIGABIT Ethernet - Recommended Configurations. . . . 66
5.13.1 Central Ethernet Switch 66
5.13.2 Redundant Ethernet Switch 67
5.14 Loss of Service (LOS) 67
6.0 Power Up sequence 68
7.0 DPLL Specification 68
7.1 Modes of operation 68
7.1.1 Locking Mode (normal operation) 69
7.1.2 Holdover Mode. 70
7.1.3 Freerun Mode. 70
7.1.4 Powerdown Mode 70
7.2 Reference Monitor Circuit 70
7.3 Locking Mode Reference Switching 70
7.4 Locking Range. 71
7.5 Locking Time 71
7.6 Lock Status 71
7.7 Jitter. 72
7.7.1 Acceptance of input wander. 72
7.7.2 Intrinsic Jitter 72
7.7.3 Jitter Tolerance 72
7.7.4 Jitter Transfer. 72
7.8 Maximum Time Interval Error (MTIE) 73
8.0 Miscellaneous 74
8.1 JTAG Interface and Board Level Test Features. 74
8.2 External Component Requirements 74
8.3 Miscellaneous Features. 74
9.0 Memory Map and Register definitions 75
10.0 Test Modes Operation 75
10.1 Overview 75
10.1.1 System Normal Mode. 75
10.1.2 System Tri-State Mode 75
10.2 Test Mode Control. 75
10.3 System Normal Mode 75
10.4 System Tri-state Mode 75
11.0 DC Characteristics 76
11.1 Absolute Maximum Ratings. 76
11.2 Recommended Operating Conditions 76
11.3 DC Characteristics. 77
11.4 Input Levels 77
11.5 Output Levels 77
12.0 AC Characteristics 78
12.1 TDM Interface Timing - ST-BUS 78
12.1.1 ST-BUS Slave Clock Mode 78
12.1.2 ST-BUS Master Clock Mode 80
12.2 TDM Interface Timing - H.110 Mode 81
12.3 TDM Interface Timing - H-MVIP 82
12.4 TDM LIU Interface Timing 84
12.5 PAC Interface Timing 85
12.6 Packet Interface Timing. 85
12.6.1 MII Transmit Timing 85
12.6.2 MII Receive Timing 86

Table of Contents

12.6.3 GMII Transmit Timing . . . 87
12.6.4 GMII Receive Timing . . . 88
12.6.5 PCS Interface Timing . . . 89
12.6.6 Management Interface Timing . . . 90
12.7 External Memory Interface Timing . . . 91
12.8 CPU Interface Timing . . . 92
12.9 System Function Port . . . 95
12.10 JTAG Interface Timing . . . 96
13.0 Power Characteristics . . . 98
14.0 Design and Layout Guidelines . . . 99
14.1 High Speed Clock & Data Interfaces . . . 99
14.1.1 External Memory Interface - special considerations during layout . . . 100
14.1.2 GMAC Interface - special considerations during layout . . . 100
14.1.3 TDM Interface - special considerations during layout . . . 100
14.1.4 Summary . . . 100
14.2 CPU TA Output . . . 100
15.0 Reference Documents . . . 102
15.1 External Standards/Specifications . . . 102
15.2 Zarlink Standards . . . 102
15.3 Zarlink ZL50110/1/4 Product Related Documentation . . . 102
16.0 Related Products . . . 103
17.0 Glossary . . . 104

List of Figures

Figure 1 - ZL50110/1/4 High Level Overview . . . 1
Figure 2 - ZL50110/1/4 Family Operation . . . 10
Figure 3 - Leased Line services over a Circuit Emulation link . . . 12
Figure 4 - Metropolitan Network using Circuit Emulation Services . . . 12
Figure 5 - Remote Concentrator Unit for Circuit Emulation Services . . . 13
Figure 6 - Multiservice Access Platform using the ZL50110/1/4 . . . 15
Figure 7 - H.100/H.110 Extension over Ethernet link . . . 16
Figure 8 - H.100/H.110 Expansion using Ethernet Switching Fabric . . . 17
Figure 9 - ZL50111 Package View and Ball Positions . . . 19
Figure 10 - ZL50110 Package View and Ball Positions . . . 20
Figure 11 - ZL50114 Package View and Ball Positions . . . 21
Figure 12 - ZL50110/1/4 Data and Control Flows . . . 53
Figure 13 - TDM to TM Flow Diagram . . . 56
Figure 14 - TDM to TM Flow Diagram . . . 57
Figure 15 - PKT to TM Flow . . . 58
Figure 16 - TM to PKT flow . . . 59
Figure 17 - ZL50110/1/4 Packet Format - Structured Mode . . . 60
Figure 18 - ZL50110/1/4 Packet Format - Unstructured Mode . . . 61
Figure 19 - External Memory Requirement for ZL50111 . . . 63
Figure 20 - External Memory Requirement for ZL50110 . . . 63
Figure 21 - Synchronous TDM Clock Generation . . . 64
Figure 22 - Differential Clock Recovery . . . 65
Figure 23 - Adaptive Clock Recovery . . . 65
Figure 24 - Gigabit Ethernet Connection - Central Ethernet Switch . . . 66
Figure 25 - Gigabit Ethernet Connection - Redundant Ethernet Switch . . . 67
Figure 26 - Powering Up the ZL50110/1/4 . . . 68
Figure 27 - Jitter Transfer Function . . . 73
Figure 28 - Jitter Transfer Function - Detail . . . 74
Figure 29 - TDM ST-BUS Slave Mode Timing at 8.192Mbit/s . . . 79
Figure 30 - TDM ST-BUS Slave Mode Timing at 2.048Mbit/s . . . 79
Figure 31 - TDM Bus Master Mode Timing at 8.192Mbit/s . . . 80
Figure 32 - TDM Bus Master Mode Timing at 2.048Mbit/s . . . 81
Figure 33 - H.110 Timing Diagram . . . 82
Figure 34 - TDM - H-MVIP Timing Diagram for 16MHz clock (8.192Mbit/s) . . . 83
Figure 35 - TDM-LIU Structured Transmission/Reception . . . 84
Figure 36 - MII Transmit Timing Diagram . . . 85
Figure 37 - MII Receive Timing Diagram . . . 86
Figure 38 - GMII Transmit Timing Diagram . . . 87
Figure 39 - GMII Receive Timing Diagram . . . 88
Figure 40 - PCS Transmit Timing Diagram . . . 89
Figure 41 - PCS Receive Timing Diagram . . . 90
Figure 42 - Management Interface Timing for Ethernet Port - Read . . . 90
Figure 43 - Management Interface Timing for Ethernet Port - Write . . . 90
Figure 44 - External RAM Read and Write timing . . . 91
Figure 45 - CPU Read - MPC8260 . . . 93
Figure 46 - CPU Write - MPC8260 . . . 93
Figure 47 - CPU DMA Read - MPC8260 . . . 94
Figure 48 - CPU DMA Write - MPC8260 . . . 94

List of Figures

Figure 49 - JTAG Signal Timing 97
Figure 50 - JTAG Clock and Reset Timing. 97
Figure 51 - ZL50110/1/4 Power Consumption Plot 98
Figure 52 - CPU_TA Board Circuit. 101

List of Tables

Table 1 - Capacity of devices in the ZL50110/1/4 family . . . 3
Table 2 - TDM services offered by the ZL50110/1/4 family . . . 10
Table 3 - TDM Interface ZL50111 Stream Pin Definition . . . 29
Table 4 - TDM Interface ZL50110 Stream Pin Definition . . . 30
Table 5 - TDM Interface ZL50110 Stream Pin Definition . . . 31
Table 6 - TDM Interface Common Pin Definition . . . 32
Table 7 - PAC Interface Package Ball Definition . . . 33
Table 8 - Packet Interface Signal Mapping - MII to GMII/PCS . . . 34
Table 9 - MII Management Interface Package Ball Definition . . . 35
Table 10 - MII Port 0 Interface Package Ball Definition . . . 35
Table 11 - MII Port 1 Interface Package Ball Definition . . . 38
Table 12 - MII Port 2 Interface Package Ball Definition . . . 41
Table 13 - MII Port 3 Interface Package Ball Definition . . . 43
Table 14 - External Memory Interface Package Ball Definition . . . 45
Table 15 - CPU Interface Package Ball Definition . . . 47
Table 16 - System Function Interface Package Ball Definition. . . . 50
Table 17 - Administration/Control Interface Package Ball Definition . . . 51
Table 18 - JTAG Interface Package Ball Definition . . . 51
Table 19 - Miscellaneous Inputs Package Ball Definitions . . . 52
Table 20 - Power and Ground Package Ball Definition . . . 52
Table 21 - DMA Maximum Bandwidths . . . 54
Table 22 - Standard Device Flows . . . 55
Table 23 - Some of the TDM Port Formats accepted by the ZL50110/1/4 Family . . . 62
Table 24 - DPLL Input Reference Frequencies . . . 69
Table 25 - Test Mode Control . . . 75
Table 26 - Absolute Maximum Ratings . . . 76
Table 27 - Recommended Operating Conditions . . . 76
Table 28 - DC Characteristics . . . 77
Table 29 - Input Levels . . . 77
Table 30 - Output Levels . . . 77
Table 31 - TDM ST-BUS Slave Timing Specification . . . 78
Table 32 - TDM ST-BUS Master Timing Specification . . . 80
Table 33 - TDM H.110 Timing Specification . . . 81
Table 34 - TDM H-MVIP Timing Specification . . . 83
Table 35 - TDM - LIU Structured Transmission/Reception . . . 84
Table 36 - PAC Timing Specification . . . 85
Table 37 - MII Transmit Timing - 100Mbit/s . . . 85
Table 38 - MII Receive Timing - 100Mbit/s . . . 86
Table 39 - GMII Transmit Timing - 1000Mbit/s . . . 87
Table 40 - GMII Receive Timing - 1000Mbit/s . . . 88
Table 41 - PCS Timing - 1000Mbit/s . . . 89
Table 42 - MAC Management Timing Specification . . . 90
Table 43 - External Memory Timing . . . 91
Table 44 - CPU Timing Specification . . . 92
Table 45 - System Clock Timing. . . . 95
Table 46 - JTAG Interface Timing. . . . 96

1.0 Introduction

1.1 Overview

The ZL50110/1/4 family provides the data-plane processing to enable constant bit rate TDM services to be carried over a packet switched network, such as an Ethernet, IP or MPLS network. The device segments the TDM data into user-defined packets, and passes it transparently over the packet network to be reconstructed at the far end. This has a number of applications, including emulation of TDM circuits and packet backplanes for TDM-based equipment.

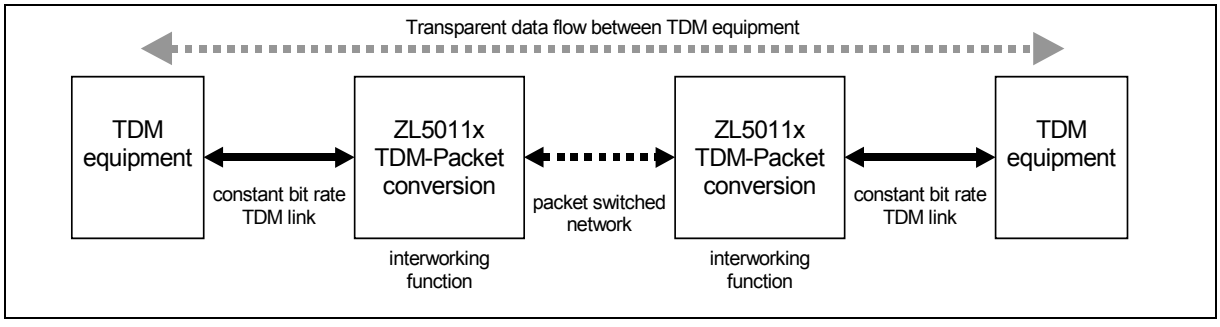


Figure 2 - ZL50110/1/4 Family Operation

The ZL50110/1/4 family offers the following types of TDM service across the packet network

Service type	TDM interface	Interface type	Interfaces to
Unstructured asynchronous	T1, E1, J2, E3 and T3	Bit clock in and out Data in and out	Line interface unit
Structured synchronous (N x 64 Kbit/s)	T1, E1 and J2 Framed TDM data streams at 2.048 and 8.192 Mbit/s	Bit clock out Frame pulse out Data in and out	Framers TDM backplane (master)
		Bit clock in Frame in Data in and out	Framers TDM backplane (slave)

Table 2 - TDM services offered by the ZL50110/1/4 family

Unstructured services are fully asynchronous, and include full support for clock recovery on a per stream basis. Both adaptive and differential clock recovery mechanisms can be used. Structured services are synchronous, with all streams driven by a common clock and frame reference. These services can be offered in two ways:

- **Synchronous master mode** - the ZL50110/1/4 provides a common clock and frame pulse to all streams, which may be locked to an incoming clock or frame reference
- **Synchronous slave mode** - the ZL50110/1/4 accepts a common external clock and frame pulse to be used by all streams

In either mode, N x 64Kbit/s trunking is supported across all streams and channels, but limited to a single input stream.

In addition, it can be used with a variety of different protocols. It includes full support for the CESoPSN (Circuit Emulation Services over Packet Switched Networks) protocol currently in development by the IETF's PWE3 (Pseudo-Wire Edge to Edge Emulation) working group. It is also fully backwards compatible with the Context Descriptor Protocol used by Zarlink's MT9088x family of devices

1.2 Latency

The following lists the intrinsic processing latency of the ZL50110/1/4, regardless of the number of active channels or contexts.

- TDM to Packet transmission processing latency less than 125 μs
- Packet to TDM transmission processing latency less than 250 μs (unstructured)
- Packet to TDM transmission processing latency less than 250 μs (structured, more than 16 channels in context)
- Packet to TDM transmission processing latency less than 375 μs (structured, 16 or less channels in context)

End-to-end latency may be estimated as the transmit latency + packet network latency + receive latency. The transmit latency is the sum of the transmit processing and the number of frames per packet x 125 μs. The receive latency is the sum of the receive processing and the delay through the jitter buffer which is programmed to compensate for packet network PDV.

The ZL50110/1/4 is capable of creating an extremely low latency connection, with end to end delays of less than 0.5 ms, depending on user configuration.

2.0 Typical Applications

2.1 Circuit Emulation Services over Packet Switched Networks (CESoPSN)

The ZL50110/1/4 family can be used to transport TDM links across the packet domain and transparently reconstruct the links at the far end. This is similar to the circuit emulation services defined by the ATM Forum, with the ZL50110/1/4 providing the core of the circuit emulation inter-working function. The ZL50110/1/4 supports circuit emulation of the following TDM circuits: E1, T1, J2, E3 and T3.

The devices conform to the draft standards for circuit emulation across a packet switched network being developed in the IETF's PWE3 (Pseudo Wire Edge-to-Edge Emulation) working group. They are capable of interfacing to several different types of packet switched networks, including Ethernet using VLAN tags, IP (both version 4 and 6) and MPLS. This is described in more detail in the ZL50110/1/4 Programmers Model.

2.1.1 Leased Line Provision

Circuit emulation is typically used to support the provision of leased line services to customers using legacy TDM equipment. For example, Figure 3 shows a leased line TDM service being carried across a packet network. The advantages are that a carrier can upgrade to a packet switched network, whilst still maintaining their existing TDM business.

The ZL50110/1/4 is capable of handling circuit emulation of both structured T1, E1, and J2 links (e.g for support of fractional circuits) and unstructured (or clear channel) T1, E1, J2, T3 and E3 links. The device handles the data-plane requirements of the provider edge inter-working function (with the exception of the physical interfaces and line interface units). Control plane functions are forwarded to the host processor controlling the ZL50110/1/4 device.

The ZL50110/1/4 provides a per-stream clock recovery function to reproduce the TDM service frequency at the egress of the packet network. This is required otherwise the queue at the egress of the packet network will either fill up or empty, depending on whether the regenerated clock is slower or faster than the original.

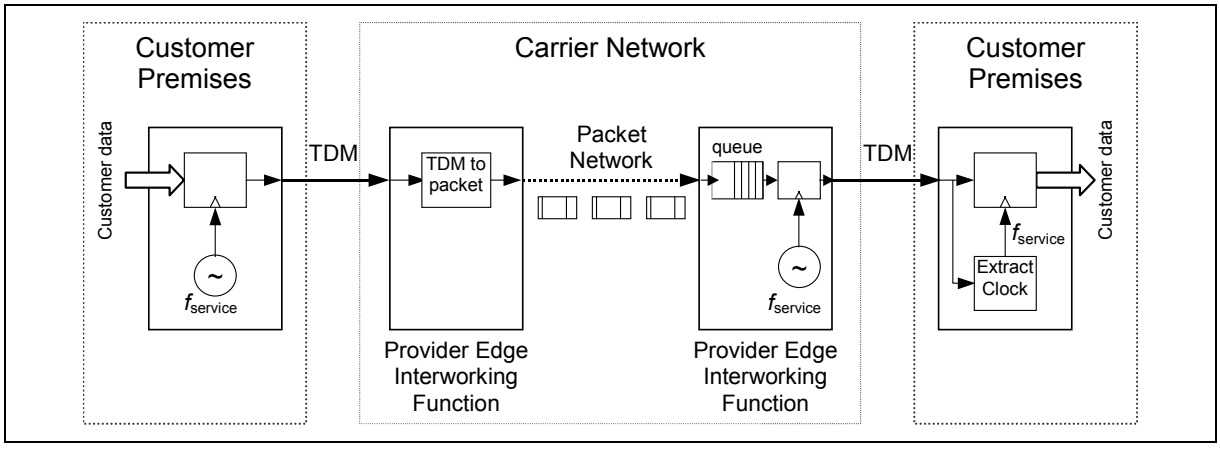


Figure 3 - Leased Line services over a Circuit Emulation link

2.1.2 Remote Concentration Unit

Another application of circuit emulation services is for a remote concentration unit, either in a customer located unit or a roadside pedestal. This allows the metropolitan network to be upgraded or extended without the expense of changing the infrastructure all the way out to the customer.

A remote unit can be used to terminate existing connections near to the customer, and carry the circuits back to the central office over a high speed metropolitan packet network (e.g. using Gigabit Ethernet or Resilient Packet Ring). For example, in Figure 4, TDM services can be terminated in a customer premise unit, using a single Gigabit Ethernet link to an access point on the Metropolitan network, or they can be brought directly into the access point itself.

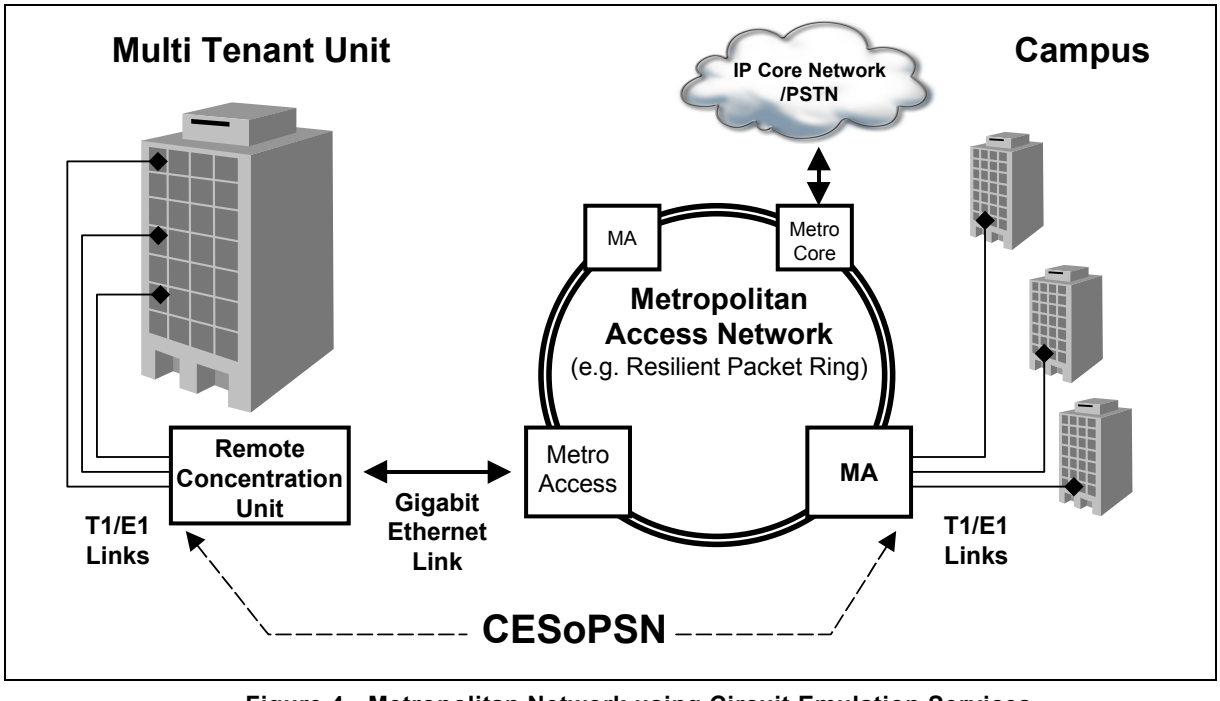


Figure 4 - Metropolitan Network using Circuit Emulation Services

Figure 5 shows the structure of a typical remote concentrator unit for such a system. Legacy TDM links for existing customers are brought into the ZL50110/1/4 device via line interface units, and assembled into packets. Network links for new customers can be brought in via a building LAN. A switch fabric using the MVTX2600 or 2800 series devices is then used to aggregate the packet streams onto a single Gigabit Ethernet link for transmission up to the access point of the Metropolitan network.

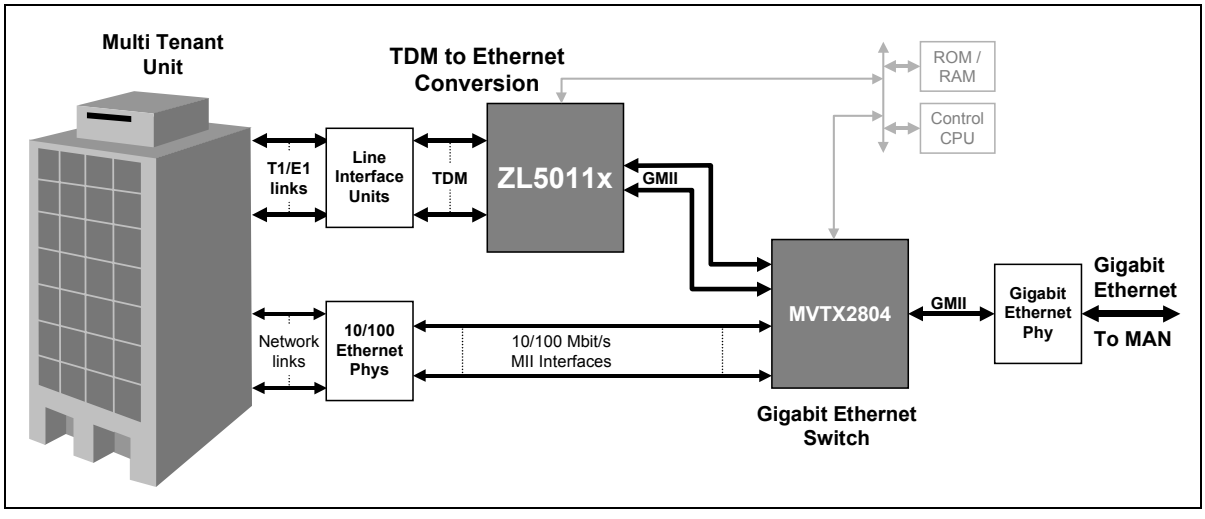


Figure 5 - Remote Concentrator Unit for Circuit Emulation Services

2.1.3 Clock Recovery

One of the main issues with circuit emulation is that the clock used to drive the TDM link is not necessarily linked into the central office reference clock, and hence may be any value within the tolerance defined for that service. The reverse link may also be independently timed, and operating at a slightly different frequency. In the plesiochronous digital hierarchy the difference in clock frequencies between TDM links is compensated for using bit stuffing techniques, allowing the clock to be accurately regenerated at the remote end of the carrier network.

With a packet network, that connection between the ingress and egress frequency is broken, since packets are discontinuous in time. From Figure 3, the TDM service frequency $f_{service}$ at the customer premises must be exactly reproduced at the egress of the packet network. The consequence of a long-term mismatch in frequency is that the queue at the egress of the packet network will either fill up or empty, depending on whether the regenerated clock is slower or faster than the original. This will cause loss of data and degradation of the service.

The ZL50110/1/4 provides a per-stream clock recovery function to reproduce the TDM service frequency at the egress of the packet network. Two schemes are employed, depending on the availability of a common reference clock at each provider edge unit.

2.1.3.1 Adaptive Clock Recovery

For applications where there is no common reference clock between provider edge units, an adaptive clock recovery technique is provided. This infers the clock rate of the original TDM service clock from the mean arrival rate of packets at the packet egress point.

The disadvantage of this type of scheme is that, depending on the characteristics of the packet network, it may prove difficult to regenerate a clock that stays within the wander requirements of the plesiochronous digital hierarchy (specifically MTIE). The reason for this is that any variation in delay between packets will feed through as a variation in the frequency of the recovered clock. High frequency jitter can be filtered out, but any low frequency variation or wander is more difficult to remove without a very long time constant. This will in turn affect the ability of the system to lock to the original clock within an acceptable time.

2.1.3.2 Differential Clock Recovery

For applications where the wander characteristics of the recovered clock are very important, such as when the emulated circuit must be connected into the plesiochronous digital hierarchy (PDH), the ZL50110/1/4 also offers a differential clock recovery technique. This relies on having a common reference clock available at each provider edge point.

In a differential technique, the timing of data packet formation is sent relative to the common reference clock. Since the same reference is available at the packet egress point and the packet size is fixed, the original service clock frequency can be recovered. This technique is unaffected by any low frequency components in the packet delay variation. The disadvantage is the requirement for a common reference clock at each end of the packet network, which could either be the central office TDM clock, or provided by a global position system (GPS) receiver.

2.2 Packet Backplane Interconnection

The ZL50110/1/4 can be used to entirely replace the TDM backplane infrastructure in a conventional computer telephony system with a packet backplane. This has several advantages: it is easily scalable, eliminates the timing problems in passing large TDM buses around the system, uses readily available and low cost network hardware, and simplifies the provision of new services to customers. This type of structure can be used in applications as diverse as telephone switches, multi-service access platforms and voice over IP gateways. Figure 6 shows a multi-service access platform based on an Ethernet backplane using the ZL50110/1/4.

The Zarlink MT9088x family is also designed to address the packet backplane market, and is fully compatible with the ZL50110/1/4. This means that traffic generated by an MT9088x can be read by a ZL50110/1/4 and vice versa.

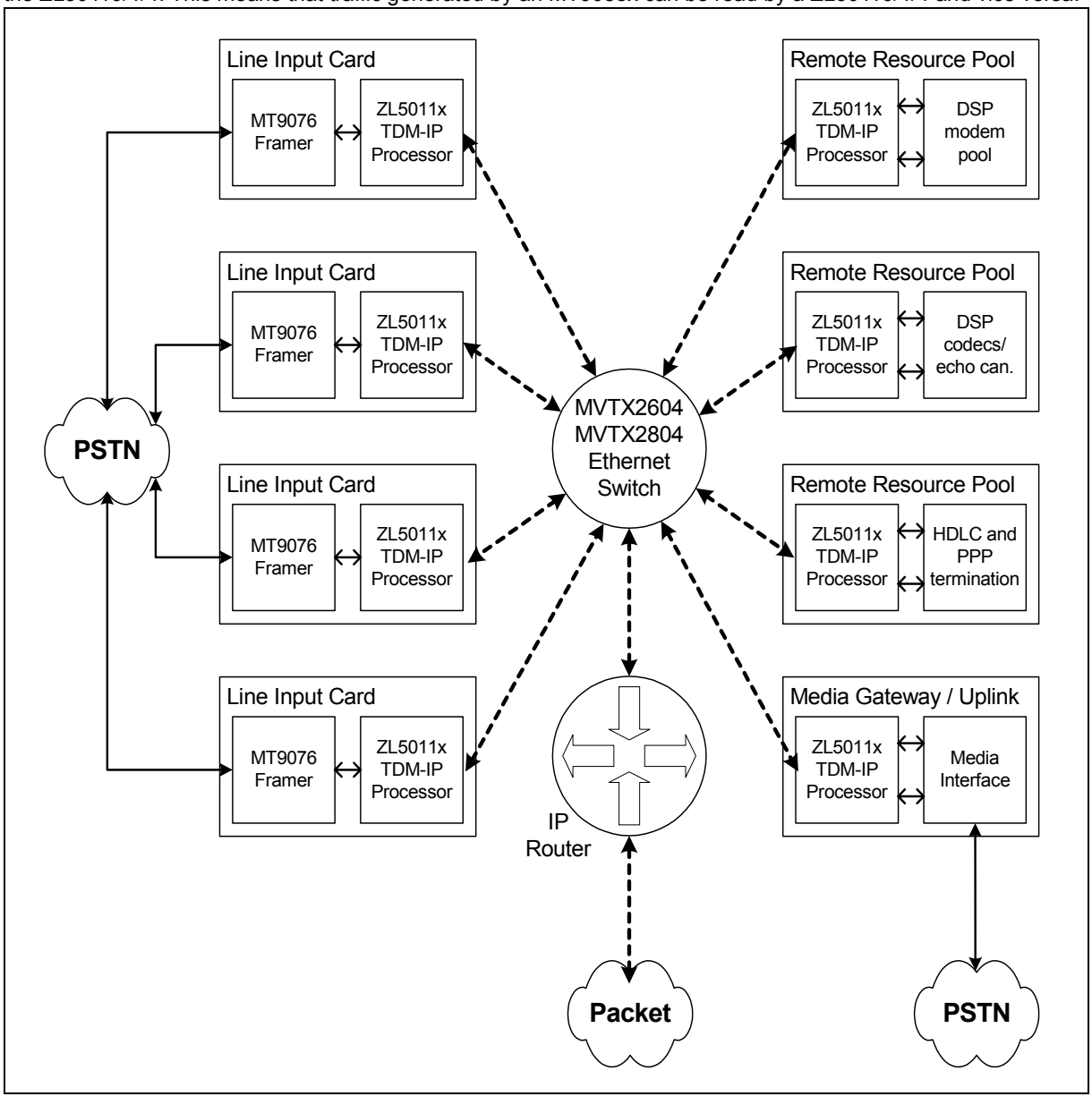


Figure 6 - Multiservice Access Platform using the ZL50110/1/4

2.3 TDM Backplane Extension

TDM backplanes, such as the H.100/H.110 bus commonly used in today's computer telephony systems, are typically based on a physical backplane the width of a single telecom rack. Extending the reach of the bus is expensive using traditional TDM infrastructure. Such links are not easily scalable, and require accurate and stable clock generation.

The ZL50110/1/4 enables the bus to be simply and easily extended beyond confines of the rack using an Ethernet connection. This enables the entire bus to be replicated in another physical location using a low cost, flexible and easily managed connection medium (see Figure 7).

This diagram shows the ZL50110/1/4 device connected to a TDM backplane via a TDM switch device (e.g. the Zarlink MT90866). This switch is used to concentrate the backplane onto the interface of the ZL50110/1/4. The configuration allows any of the TDM channels on the original backplane to be switched onto any channel on the extension backplane, and vice versa.

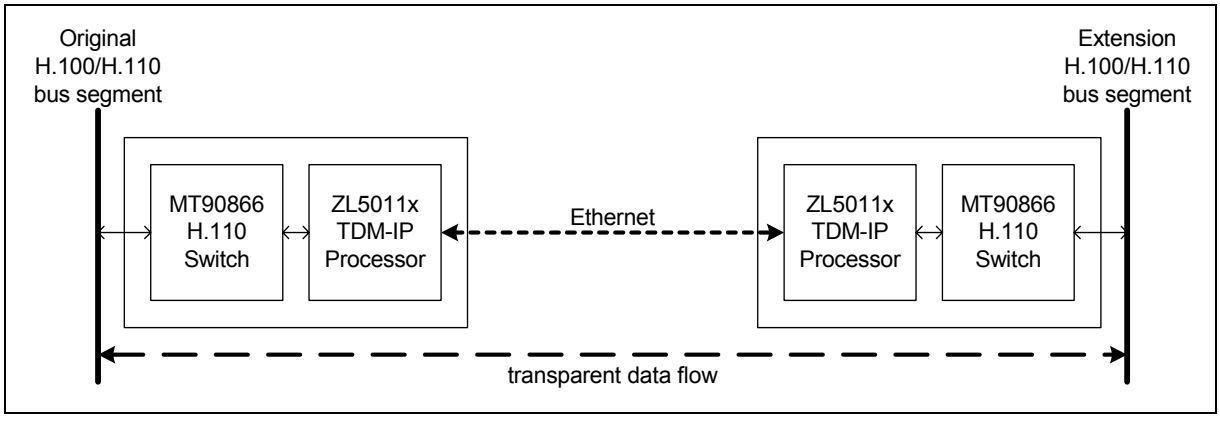


Figure 7 - H.100/H.110 Extension over Ethernet link

2.4 TDM Backplane Expansion

One of the issues faced by medium and high-end telecommunication systems is scalability. The H.100/H.110 TDM bus is limited to 4096 concurrent timeslots, or 2048 full duplex links. The ZL50110/1/4 can be used to expand the capacity of a system by switching timeslots between multiple separate TDM backplane segments.

This application is shown in Figure 8. Unlike parts based on expensive or proprietary infrastructure, the use of an Ethernet switch fabric enables common, readily available and low cost hardware to be employed, reducing both installation costs and operational expenses. The combination of existing TDM infrastructure and an Ethernet-based packet backplane enables systems to be built up using off-the-shelf components, and to be quickly expanded as required.

As in Figure 7, the MT90866 TDM switch is used to concentrate the TDM backplane onto the ZL50110/1/4 TDM interface. This allows any channel on a bus segment to be switched onto any channel on any other bus segment.

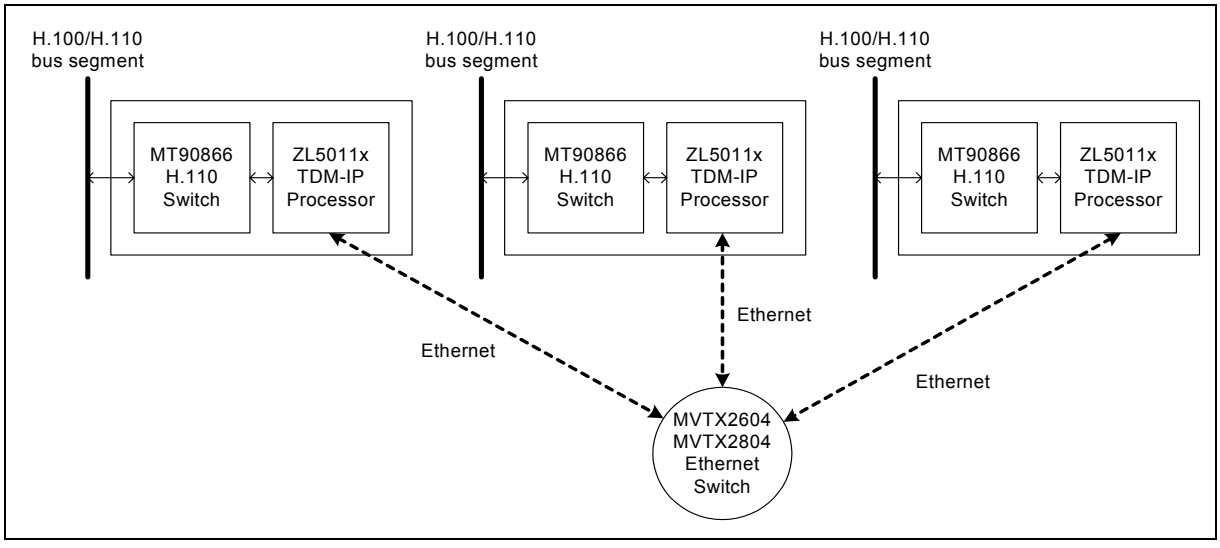


Figure 8 - H.100/H.110 Expansion using Ethernet Switching Fabric

2.5 Loopback Modes

The ZL50110/1/4 devices support loopback of the TDM circuits and the circuit emulation packets.

TDM loopback is achieved by first packetising the TDM circuit as normal via the TDM Interface and Payload Assembly blocks. The packetised data is then routed by the Task Manager back to the same TDM port via the TDM Formatter and TDM Interface.

Loopback of the emulated services is achieved by redirecting classified packets from the Packet Receive blocks, back to the packet network. The Packet Transmit blocks are setup to strip the original header and add a new header directing the packets back to the source.

3.0 Physical Specification

The ZL50111 will be packaged in a PBGA device.

Features:

- Body Size: 35mm x 35mm (typ)
- Ball Count: 552
- Ball Pitch: 1.27mm (typ)
- Ball Matrix: 26 x 26
- Ball Diameter: 0.75 mm (typ)
- Total Package Thickness: 2.33 mm (typ)

The ZL50110 will be packaged in a PBGA device.

Features:

- Body Size: 35mm x 35mm (typ)
- Ball Count: 552
- Ball Pitch: 1.27mm (typ)
- Ball Matrix: 26 x 26
- Ball Diameter: 0.75 mm (typ)
- Total Package Thickness: 2.33 mm (typ)

The ZL50114 will be packaged in a PBGA device.

Features:

- Body Size: 35mm x 35mm (typ)
- Ball Count: 552
- Ball Pitch: 1.27mm (typ)
- Ball Matrix: 26 x 26
- Ball Diameter: 0.75 mm (typ)
- Total Package Thickness: 2.33 mm (typ)

ZL50111 Package view from TOP side. Note that ball A1 is non-chamfered corner.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
A	GND	TDM_STo[1]	TDM_CLKo[3]	TDM_STo[4]	TDM_STo[5]	TDM_STi[6]	TDM_STo[7]	TDM_STi[7]	TDM_CLKo[10]	TDM_CLK[10]	TDM_CLK[11]	TDM_CLKo[13]	GND	TDM_STo[13]	TDM_STo[14]	TDM_CLKo[15]	TDM_STo[16]	TDM_CLKo[18]	TDM_STi[18]	TDM_CLKo[20]	TDM_STi[20]	TDM_STo[21]	TDM_STi[21]	TDM_CLKo[24]	TDM_CLKo[25]	GND	A
B	TDM_FRMo_REF	TDM_STo[0]	TDM_STi[2]	TDM_CLK[3]	TDM_STi[4]	TDM_CLKo[6]	TDM_STo[6]	TDM_CLKo[8]	TDM_CLKo[9]	TDM_STo[10]	TDM_STi[10]	TDM_CLK[12]	TDM_STo[12]	TDM_STi[13]	TDM_CLK[15]	TDM_STi[15]	TDM_STi[17]	TDM_CLK[18]	TDM_CLKo[20]	TDM_STo[19]	TDM_CLKo[22]	TDM_CLKo[23]	TDM_STo[24]	TDM_CLKo[26]	TDM_STi[24]	TDM_CLKo[27]	B
C	TDM_CLKP	TDM_FRMi_REF	TDM_CLK_REF	TDM_CLKo[1]	TDM_STi[3]	TDM_CLKo[2]	TDM_CLK[6]	TDM_CLK[7]	TDM_CLKo[9]	TDM_STo[9]	TDM_STi[9]	TDM_STi[11]	TDM_CLK[13]	TDM_CLKo[14]	TDM_CLKo[16]	TDM_STi[16]	TDM_CLKo[17]	TDM_STi[19]	TDM_CLKo[21]	TDM_CLK[21]	TDM_CLK[24]	TDM_STi[22]	TDM_STo[26]	TDM_CLK[27]	TDM_STi[27]	TDM_STi[28]	C
D	RAM_DATA[3]	RAM_DATA[1]	TDM_CLKS	RAM_DATA[0]	TDM_STi[0]	TDM_CLK[1]	TDM_STo[3]	TDM_STi[5]	TDM_CLK[5]	TDM_CLKo[7]	TDM_STi[8]	TDM_CLKo[11]	TDM_STi[12]	TDM_STi[14]	TDM_CLK[16]	TDM_CLKo[19]	TDM_STo[18]	TDM_STo[20]	TDM_CLKo[22]	TDM_STo[27]	TDM_STo[25]	TDM_CLKo[26]	TDM_CLKo[28]	TDM_CLKo[29]	TDM_STi[29]	TDM_STi[31]	D
E	RAM_DATA[10]	RAM_DATA[9]	RAM_DATA[5]	RAM_DATA[4]	RAM_DATA[2]	TDM_CLKo_REF	TDM_CLK[0]	TDM_CLKo[4]	TDM_STi[1]	TDM_CLK[4]	TDM_STo[8]	TDM_CLK[8]	TDM_CLK[12]	TDM_STo[15]	TDM_CLK[17]	TDM_CLKo[19]	TDM_STo[23]	TDM_STi[23]	TDM_CLKo[25]	TDM_STi[26]	TDM_CLK[28]	GND	TDM_CLKo[30]	TDM_CLKo[30]	TDM_STi[30]	TDM_STo[29]	E
F	RAM_DATA[15]	RAM_DATA[13]	RAM_DATA[12]	RAM_DATA[6]	RAM_DATA[7]	GND	VDD_CORE	TDM_STo[2]	TDM_CLKo[0]	TDM_CLK[2]	TDM_CLKo[5]	VDD_CORE	TDM_STo[11]	TDM_CLK[14]	VDD_CORE	TDM_STo[17]	TDM_CLK[22]	TDM_STi[25]	TDM_CLKo[23]	VDD_CORE	GND	TDM_CLK[31]	TDM_CLKo[29]	TDM_STo[28]	TDM_CLKo[31]	M2_LINKUP_LED	F
G	RAM_DATA[21]	RAM_DATA[18]	RAM_DATA[16]	RAM_DATA[14]	RAM_DATA[11]	RAM_DATA[8]															TDM_STo[31]	TDM_STo[30]	M1_LINKUP_LED	M0_LINKUP_LED	M1_GIGABIT_LED	M_MDIO	G
H	RAM_DATA[25]	RAM_DATA[24]	RAM_DATA[23]	RAM_DATA[19]	RAM_DATA[17]	VDD_CORE															VDD_CORE	M0_GIGABIT_LED	M_MDC	M3_CRS	M3_TXCLK	M3_RXER	H
J	RAM_DATA[29]	RAM_DATA[28]	RAM_DATA[27]	RAM_DATA[26]	RAM_DATA[22]	RAM_DATA[20]			VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO			M3_RXDV	M3_RXD[3]	M3_RXD[2]	M3_RXD[1]	M3_RXD[0]	M3_COL	J
K	RAM_PARITY[1]	RAM_PARITY[0]	RAM_DATA[31]	RAM_DATA[30]	GND	VDD_CORE			VDD_IO									VDD_IO			VDD_CORE	GND	M3_TXD[3]	M3_TXEN	M3_TXER	M3_RXCLK	K
L	RAM_PARITY[7]	RAM_PARITY[6]	RAM_PARITY[5]	RAM_PARITY[4]	RAM_PARITY[3]	RAM_PARITY[2]			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_RXER	M1_TXCLK	M1_CRS	M3_TXD[0]	M3_TXD[1]	M3_TXD[2]	L
M	RAM_ADDR[5]	RAM_ADDR[4]	RAM_ADDR[2]	RAM_ADDR[3]	RAM_ADDR[0]	RAM_ADDR[1]			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			VDD_CORE	M1_REFCLK	M1_RXCLK	M1_RXD[5]	M1_RXD[7]	M1_RXDV	M
N	GND	RAM_ADDR[6]	RAM_ADDR[7]	RAM_ADDR[8]	GND	VDD_CORE			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_GTX_CLK	GND	M1_TXER	M1_RXD[2]	M1_RXD[3]	GND	N
P	RAM_ADDR[9]	RAM_ADDR[10]	RAM_ADDR[11]	RAM_ADDR[13]	RAM_ADDR[16]	GND			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_TXD[2]	M1_TXD[6]	M1_TXEN	GND	M1_RXD[4]	M1_RXD[6]	P
R	RAM_ADDR[12]	RAM_ADDR[14]	RAM_ADDR[15]	RAM_ADDR[19]	N/C	N/C			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_TXD[0]	M1_TXD[3]	M1_TXD[5]	M1_TXD[7]	M1_COL	M1_RXD[1]	R
T	RAM_ADDR[17]	RAM_ADDR[18]	RAM_BW_B	N/C	GND	A1VDD			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			VDD_CORE	M1_TXD[1]	M1_TXD[4]	GND	M1_RBC1	M1_RXD[0]	T
U	PLL_PRI	RAM_BW_A	RAM_BW_C	RAM_RW	SYSTEM_DEBUG	SYSTEM_CLK			VDD_IO									VDD_IO			M0_GTX_CLK	M0_RXD[2]	M0_RXD[5]	M0_TXCLK	M0_CRS	M1_RBC0	U
V	PLL_SEC	RAM_BW_D	RAM_BW_F	SYSTEM_RST	GPIO[2]	VDD_CORE			VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO			M0_TXD[7]	M0_TXER	M0_TXEN	M0_RXD[4]	M0_RXDV	M0_RXER	V
W	RAM_BW_E	RAM_BW_G	GPIO[0]	GPIO[3]	GPIO[9]	RAM_DATA[33]															M0_TXD[2]	M0_TXD[5]	M0_TXD[6]	M0_RXD[6]	M0_RXD[7]	M0_RXD[3]	W
Y	RAM_BW_H	GPIO[4]	GPIO[6]	GPIO[10]	RAM_DATA[32]	VDD_CORE															VDD_CORE	M0_TXD[1]	M0_TXD[4]	M0_RBC0	M0_COL	M0_RXD[1]	Y
AA	GPIO[1]	GPIO[7]	GPIO[8]	GPIO[15]	RAM_DATA[39]	GND	RAM_DATA[45]	RAM_DATA[52]	VDD_CORE	JTAG_TMS	CPU_ADDR[2]	CPU_ADDR[12]	VDD_CORE	VDD_CORE	CPU_DATA[8]	CPU_DATA[15]	CPU_DATA[23]	VDD_CORE	M2_RXCLK	M2_RXDV	GND	M0_TXD[0]	M0_TXD[3]	M0_REFCLK	M0_RBC1	M0_RXD[0]	AA
AB	GPIO[5]	GPIO[11]	GPIO[14]	RAM_DATA[38]	RAM_DATA[43]	RAM_DATA[44]	RAM_DATA[51]	RAM_DATA[60]	TEST_MODE[1]	GND	CPU_ADDR[6]	CPU_ADDR[14]	CPU_ADDR[23]	CPU_TA	CPU_DATA[1]	CPU_DATA[7]	CPU_DATA[12]	CPU_DATA[22]	CPU_DATA[30]	M2_TXER	M2_RXD[1]	M0_RXCLK	M3_LINKUP_LED	M2_ACTIVE_LED	M1_ACTIVE_LED	M3_ACTIVE_LED	AB
AC	GPIO[12]	GPIO[13]	RAM_DATA[37]	RAM_DATA[42]	RAM_DATA[46]	RAM_DATA[49]	RAM_DATA[59]	TEST_MODE[0]	JTAG_TDO	CPU_ADDR[4]	CPU_ADDR[9]	CPU_ADDR[16]	CPU_ADDR[22]	CPU_CLK	CPU_DREQ0	N/C	CPU_DATA[10]	CPU_DATA[16]	CPU_DATA[21]	CPU_DATA[27]	M2_TXD[1]	M2_TXEN	M2_RXD[2]	M2_RXER	M2_CRS	M0_ACTIVE_LED	AC
AD	RAM_DATA[34]	RAM_DATA[36]	RAM_DATA[41]	RAM_DATA[47]	RAM_DATA[53]	RAM_DATA[58]	RAM_DATA[63]	JTAG_TCK	PULL_LO	CPU_ADDR[7]	CPU_ADDR[11]	CPU_ADDR[17]	CPU_ADDR[21]	CPU_WE	CPU_SDACK2	CPU_IREQ1	CPU_DATA[3]	CPU_DATA[6]	CPU_DATA[14]	CPU_DATA[20]	CPU_DATA[24]	CPU_DATA[29]	M2_TXD[2]	M2_RXD[0]	M2_RXD[3]	M2_TXCLK	AD
AE	RAM_DATA[35]	RAM_DATA[40]	RAM_DATA[48]	RAM_DATA[54]	RAM_DATA[57]	RAM_DATA[62]	JTAG_TRST	N/C	CPU_ADDR[3]	CPU_ADDR[8]	CPU_ADDR[13]	CPU_ADDR[18]	CPU_ADDR[20]	CPU_OE	CPU_TS_ALE	CPU_DREQ1	N/C	CPU_DATA[4]	CPU_DATA[9]	CPU_DATA[13]	CPU_DATA[18]	CPU_DATA[25]	CPU_DATA[28]	M2_TXD[0]	M2_TXD[3]	M2_COL	AE
AF	GND	RAM_DATA[50]	RAM_DATA[55]	RAM_DATA[56]	RAM_DATA[61]	TEST_MODE[2]	JTAG_TDI	PULL_LO	CPU_ADDR[5]	CPU_ADDR[10]	CPU_ADDR[15]	CPU_ADDR[19]	GND	CPU_CS	CPU_SDACK1	PULL_HI	CPU_IREQ0	CPU_DATA[0]	CPU_DATA[5]	CPU_DATA[2]	CPU_DATA[11]	CPU_DATA[17]	CPU_DATA[19]	CPU_DATA[26]	CPU_DATA[31]	GND	AF
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	

Figure 9 - ZL50111 Package View and Ball Positions

ZL50110 Package view from TOP side. Note that ball A1 is non-chamfered corner.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
A	GND	TDM_STo[1]	TDM_CLKo[3]	TDM_STo[4]	TDM_STo[5]	TDM_STi[6]	TDM_STo[7]	TDM_STi[7]	N/C	N/C	N/C	N/C	GND	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	GND	A
B	TDM_FRMo_REF	TDM_STo[0]	TDM_STi[2]	TDM_CLKi[3]	TDM_STi[4]	TDM_CLKo[6]	TDM_STo[6]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	B
C	TDM_CLKiP	TDM_FRMi_REF	TDM_CLKi_REF	TDM_CLKo[1]	TDM_STi[3]	TDM_CLKo[2]	TDM_CLKi[6]	TDM_CLKi[7]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	C
D	RAM_DATA[3]	RAM_DATA[1]	TDM_CLKiS	RAM_DATA[0]	TDM_STi[0]	TDM_CLKi[1]	TDM_STo[3]	TDM_STi[5]	TDM_CLKi[5]	TDM_CLKo[7]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	D
E	RAM_DATA[10]	RAM_DATA[9]	RAM_DATA[5]	RAM_DATA[4]	RAM_DATA[2]	TDM_CLKo_REF	TDM_CLKi[0]	TDM_CLKo[4]	TDM_STi[1]	TDM_CLKi[4]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	GND	N/C	N/C	N/C	N/C	E
F	RAM_DATA[15]	RAM_DATA[13]	RAM_DATA[12]	RAM_DATA[8]	RAM_DATA[7]	GND	VDD_CORE	TDM_STo[2]	TDM_CLKo[0]	TDM_CLKi[2]	TDM_CLKo[5]	VDD_CORE	N/C	N/C	VDD_CORE	N/C	N/C	N/C	N/C	VDD_CORE	GND	N/C	N/C	N/C	N/C	N/C	F
G	RAM_DATA[21]	RAM_DATA[18]	RAM_DATA[16]	RAM_DATA[14]	RAM_DATA[11]	RAM_DATA[8]															N/C	N/C	M1_LINKUP_LED	M0_LINKUP_LED	M1_GIGABIT_LED	M_MDIO	G
H	RAM_DATA[25]	RAM_DATA[24]	RAM_DATA[23]	RAM_DATA[19]	RAM_DATA[17]	VDD_CORE															VDD_CORE	M0_GIGABIT_LED	M_MDC	N/C	N/C	N/C	H
J	RAM_DATA[29]	RAM_DATA[28]	RAM_DATA[27]	RAM_DATA[26]	RAM_DATA[22]	RAM_DATA[20]			VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO			N/C	N/C	N/C	N/C	N/C	N/C	J
K	RAM_PARITY[1]	RAM_PARITY[0]	RAM_DATA[31]	RAM_DATA[30]	GND	VDD_CORE			VDD_IO									VDD_IO			VDD_CORE	GND	N/C	N/C	N/C	N/C	K
L	RAM_PARITY[7]	RAM_PARITY[6]	RAM_PARITY[5]	RAM_PARITY[4]	RAM_PARITY[3]	RAM_PARITY[2]			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_RXER	M1_TXCLK	M1_CRS	N/C	N/C	N/C	L
M	RAM_ADDR[5]	RAM_ADDR[4]	RAM_ADDR[2]	RAM_ADDR[3]	RAM_ADDR[0]	RAM_ADDR[1]			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			VDD_CORE	M1_REFCLK	M1_RXCLK	M1_RXD[5]	M1_RXD[7]	M1_RXDV	M
N	GND	RAM_ADDR[6]	RAM_ADDR[7]	RAM_ADDR[8]	GND	VDD_CORE			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_GTX_CLK	GND	M1_TXER	M1_RXD[2]	M1_RXD[3]	GND	N
P	RAM_ADDR[9]	RAM_ADDR[10]	RAM_ADDR[11]	RAM_ADDR[13]	RAM_ADDR[16]	GND			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_TXD[2]	M1_TXD[6]	M1_TXEN	GND	M1_RXD[4]	M1_RXD[6]	P
R	RAM_ADDR[12]	RAM_ADDR[14]	RAM_ADDR[15]	RAM_ADDR[19]	N/C	N/C			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_TXD[0]	M1_TXD[3]	M1_TXD[5]	M1_TXD[7]	M1_COL	M1_RXD[1]	R
T	RAM_ADDR[17]	RAM_ADDR[18]	RAM_BW_B	N/C	GND	A1VDD			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			VDD_CORE	M1_TXD[1]	M1_TXD[4]	GND	M1_RBC1	M1_RXD[0]	T
U	PLL_PRI	RAM_BW_A	RAM_BW_C	RAM_RW	SYSTEM_DEBUG	SYSTEM_CLK			VDD_IO									VDD_IO			M0_GTX_CLK	M0_RXD[2]	M0_RXD[5]	M0_TXCLK	M0_CRS	M1_RBC0	U
V	PLL_SEC	RAM_BW_D	RAM_BW_F	SYSTEM_RST	GPIO[2]	VDD_CORE			VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO			M0_TXD[7]	M0_TXER	M0_TXEN	M0_RXD[4]	M0_RXDV	M0_RXER	V
W	RAM_BW_E	RAM_BW_G	GPIO[0]	GPIO[3]	GPIO[9]	RAM_DATA[33]															M0_TXD[2]	M0_TXD[5]	M0_TXD[6]	M0_RXD[6]	M0_RXD[7]	M0_RXD[3]	W
Y	RAM_BW_H	GPIO[4]	GPIO[6]	GPIO[10]	RAM_DATA[32]	VDD_CORE															VDD_CORE	M0_TXD[1]	M0_TXD[4]	M0_RBC0	M0_COL	M0_RXD[1]	Y
AA	GPIO[1]	GPIO[7]	GPIO[8]	GPIO[15]	RAM_DATA[39]	GND	RAM_DATA[45]	RAM_DATA[52]	VDD_CORE	JTAG_TMS	CPU_ADDR[2]	CPU_ADDR[12]	VDD_CORE	VDD_CORE	CPU_DATA[8]	CPU_DATA[15]	CPU_DATA[23]	VDD_CORE	N/C	N/C	GND	M0_TXD[0]	M0_TXD[3]	M0_REFCLK	M0_RBC1	M0_RXD[0]	AA
AB	GPIO[5]	GPIO[11]	GPIO[14]	RAM_DATA[38]	RAM_DATA[43]	RAM_DATA[44]	RAM_DATA[51]	RAM_DATA[60]	TEST_MODE[1]	GND	CPU_ADDR[6]	CPU_ADDR[14]	CPU_ADDR[23]	CPU_TA	CPU_DATA[1]	CPU_DATA[7]	CPU_DATA[12]	CPU_DATA[22]	CPU_DATA[30]	N/C	N/C	M0_RXCLK	N/C	N/C	M1_ACTIVE_LED	N/C	AB
AC	GPIO[12]	GPIO[13]	RAM_DATA[37]	RAM_DATA[42]	RAM_DATA[46]	RAM_DATA[49]	RAM_DATA[59]	TEST_MODE[0]	JTAG_TDO	CPU_ADDR[4]	CPU_ADDR[9]	CPU_ADDR[16]	CPU_ADDR[22]	CPU_CLK	CPU_DREQ0	N/C	CPU_DATA[10]	CPU_DATA[16]	CPU_DATA[21]	CPU_DATA[27]	N/C	N/C	N/C	N/C	N/C	M0_ACTIVE_LED	AC
AD	RAM_DATA[34]	RAM_DATA[36]	RAM_DATA[41]	RAM_DATA[47]	RAM_DATA[53]	RAM_DATA[58]	RAM_DATA[63]	JTAG_TCK	PULL_LO	CPU_ADDR[7]	CPU_ADDR[11]	CPU_ADDR[17]	CPU_ADDR[21]	CPU_WE	CPU_SDACK2	CPU_IREQ1	CPU_DATA[3]	CPU_DATA[6]	CPU_DATA[14]	CPU_DATA[20]	CPU_DATA[24]	CPU_DATA[29]	N/C	N/C	N/C	N/C	AD
AE	RAM_DATA[35]	RAM_DATA[40]	RAM_DATA[48]	RAM_DATA[54]	RAM_DATA[57]	RAM_DATA[62]	JTAG_TRST	N/C	CPU_ADDR[3]	CPU_ADDR[8]	CPU_ADDR[13]	CPU_ADDR[18]	CPU_ADDR[20]	CPU_OE	CPU_TSALE	CPU_DREQ1	N/C	CPU_DATA[4]	CPU_DATA[9]	CPU_DATA[13]	CPU_DATA[18]	CPU_DATA[25]	CPU_DATA[28]	N/C	N/C	N/C	AE
AF	GND	RAM_DATA[50]	RAM_DATA[55]	RAM_DATA[56]	RAM_DATA[61]	TEST_MODE[2]	JTAG_TDI	PULL_LO	CPU_ADDR[5]	CPU_ADDR[10]	CPU_ADDR[15]	CPU_ADDR[19]	GND	CPU_CS	CPU_SDACK1	PULL_HI	CPU_IREQ0	CPU_DATA[0]	CPU_DATA[5]	CPU_DATA[2]	CPU_DATA[11]	CPU_DATA[17]	CPU_DATA[19]	CPU_DATA[26]	CPU_DATA[31]	GND	AF
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	

Figure 10 - ZL50110 Package View and Ball Positions

ZL50114 Package view from TOP side. Note that ball A1 is non-chamfered corner.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
A	GND	TDM_STo[1]	TDM_CLKo[3]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	GND	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	GND	A
B	TDM_FRMo_REF	TDM_STo[0]	TDM_STi[2]	TDM_CLKi[3]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	B
C	TDM_CLKiP	TDM_FRMi_REF	TDM_CLKi_REF	TDM_CLKo[1]	TDM_STi[3]	TDM_CLKo[2]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	C
D	RAM_DATA[3]	RAM_DATA[1]	TDM_CLKiS	RAM_DATA[0]	TDM_STi[0]	TDM_CLKi[1]	TDM_STo[3]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	D
E	RAM_DATA[10]	RAM_DATA[9]	RAM_DATA[5]	RAM_DATA[4]	RAM_DATA[2]	TDM_CLKo_REF	TDM_CLKi[0]	N/C	TDM_STi[1]	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	N/C	GND	N/C	N/C	N/C	N/C	E
F	RAM_DATA[15]	RAM_DATA[13]	RAM_DATA[12]	RAM_DATA[6]	RAM_DATA[7]	GND	VDD_CORE	TDM_STo[2]	TDM_CLKo[0]	TDM_CLKi[2]	N/C	VDD_CORE	N/C	N/C	VDD_CORE	N/C	N/C	N/C	N/C	VDD_CORE	GND	N/C	N/C	N/C	N/C	N/C	F
G	RAM_DATA[21]	RAM_DATA[18]	RAM_DATA[16]	RAM_DATA[14]	RAM_DATA[11]	RAM_DATA[8]															N/C	N/C	M1_LINK_UP_LED	M0_LINK_UP_LED	M1_GIGA_BIT_LED	M_MDIO	G
H	RAM_DATA[25]	RAM_DATA[24]	RAM_DATA[23]	RAM_DATA[19]	RAM_DATA[17]	VDD_CORE															VDD_CORE	M0_GIGA_BIT_LED	M_MDC	N/C	N/C	N/C	H
J	RAM_DATA[29]	RAM_DATA[28]	RAM_DATA[27]	RAM_DATA[26]	RAM_DATA[22]	RAM_DATA[20]			VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO			N/C	N/C	N/C	N/C	N/C	N/C	J
K	RAM_PARITY[1]	RAM_PARITY[0]	RAM_DATA[31]	RAM_DATA[30]	GND	VDD_CORE			VDD_IO									VDD_IO			VDD_CORE	GND	N/C	N/C	N/C	N/C	K
L	RAM_PARITY[7]	RAM_PARITY[6]	RAM_PARITY[5]	RAM_PARITY[4]	RAM_PARITY[3]	RAM_PARITY[2]			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_RXER	M1_TXCLK	M1_CRS	N/C	N/C	N/C	L
M	RAM_ADDR[5]	RAM_ADDR[4]	RAM_ADDR[2]	RAM_ADDR[3]	RAM_ADDR[0]	RAM_ADDR[1]			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			VDD_CORE	M1_REFCLK	M1_RXCLK	M1_RXD[5]	M1_RXD[7]	M1_RXDV	M
N	GND	RAM_ADDR[6]	RAM_ADDR[7]	RAM_ADDR[8]	GND	VDD_CORE			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_GTX_CLK	GND	M1_TXER	M1_RXD[2]	M1_RXD[3]	GND	N
P	RAM_ADDR[9]	RAM_ADDR[10]	RAM_ADDR[11]	RAM_ADDR[13]	RAM_ADDR[16]	GND			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_TXD[2]	M1_TXD[6]	M1_TXEN	GND	M1_RXD[4]	M1_RXD[6]	P
R	RAM_ADDR[12]	RAM_ADDR[14]	RAM_ADDR[15]	RAM_ADDR[19]	N/C	N/C			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			M1_TXD[0]	M1_TXD[3]	M1_TXD[5]	M1_TXD[7]	M1_COL	M1_RXD[1]	R
T	RAM_ADDR[17]	RAM_ADDR[18]	RAM_BW_B	N/C	GND	A1VDD			VDD_IO		GND	GND	GND	GND	GND	GND		VDD_IO			VDD_CORE	M1_TXD[1]	M1_TXD[4]	GND	M1_RBC1	M1_RXD[0]	T
U	PLL_PRI	RAM_BW_A	RAM_BW_C	RAM_RW	SYSTEM_DEBUG	SYSTEM_CLK			VDD_IO									VDD_IO			M0_GTX_CLK	M0_RXD[2]	M0_RXD[5]	M0_TXCLK	M0_CRS	M1_RBC0	U
V	PLL_SEC	RAM_BW_D	RAM_BW_F	SYSTEM_RST	GPIO[2]	VDD_CORE			VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO	VDD_IO			M0_TXD[7]	M0_TXER	M0_TXEN	M0_RXD[4]	M0_RXDV	M0_RXER	V
W	RAM_BW_E	RAM_BW_G	GPIO[0]	GPIO[3]	GPIO[9]	RAM_DATA[33]															M0_TXD[2]	M0_TXD[5]	M0_TXD[6]	M0_RXD[6]	M0_RXD[7]	M0_RXD[3]	W
Y	RAM_BW_H	GPIO[4]	GPIO[6]	GPIO[10]	RAM_DATA[32]	VDD_CORE															VDD_CORE	M0_TXD[1]	M0_TXD[4]	M0_RBC0	M0_COL	M0_RXD[1]	Y
AA	GPIO[1]	GPIO[7]	GPIO[8]	GPIO[15]	RAM_DATA[39]	GND	RAM_DATA[45]	RAM_DATA[52]	VDD_CORE	JTAG_TMS	CPU_ADDR[2]	CPU_ADDR[12]	VDD_CORE	VDD_CORE	CPU_DATA[8]	CPU_DATA[15]	CPU_DATA[23]	VDD_CORE	N/C	N/C	GND	M0_TXD[0]	M0_TXD[3]	M0_REFCLK	M0_RBC1	M0_RXD[0]	AA
AB	GPIO[5]	GPIO[11]	GPIO[14]	RAM_DATA[38]	RAM_DATA[43]	RAM_DATA[44]	RAM_DATA[51]	RAM_DATA[60]	TEST_MODE[1]	GND	CPU_ADDR[6]	CPU_ADDR[14]	CPU_ADDR[23]	CPU_TA	CPU_DATA[1]	CPU_DATA[7]	CPU_DATA[12]	CPU_DATA[22]	CPU_DATA[30]	N/C	N/C	M0_RXCLK	N/C	N/C	M1_ACTIVE_LED	N/C	AB
AC	GPIO[12]	GPIO[13]	RAM_DATA[37]	RAM_DATA[42]	RAM_DATA[46]	RAM_DATA[49]	RAM_DATA[59]	TEST_MODE[0]	JTAG_TDO	CPU_ADDR[4]	CPU_ADDR[9]	CPU_ADDR[16]	CPU_ADDR[22]	CPU_CLK	CPU_DREQ0	N/C	CPU_DATA[10]	CPU_DATA[16]	CPU_DATA[21]	CPU_DATA[27]	N/C	N/C	N/C	N/C	N/C	M0_ACTIVE_LED	AC
AD	RAM_DATA[34]	RAM_DATA[36]	RAM_DATA[41]	RAM_DATA[47]	RAM_DATA[53]	RAM_DATA[58]	RAM_DATA[63]	JTAG_TCK	PULL_LO	CPU_ADDR[7]	CPU_ADDR[11]	CPU_ADDR[17]	CPU_ADDR[21]	CPU_WE	CPU_SDACK2	CPU_IREQ1	CPU_DATA[3]	CPU_DATA[6]	CPU_DATA[14]	CPU_DATA[20]	CPU_DATA[24]	CPU_DATA[29]	N/C	N/C	N/C	N/C	AD
AE	RAM_DATA[35]	RAM_DATA[40]	RAM_DATA[48]	RAM_DATA[54]	RAM_DATA[57]	RAM_DATA[62]	JTAG_TRST	N/C	CPU_ADDR[3]	CPU_ADDR[8]	CPU_ADDR[13]	CPU_ADDR[18]	CPU_ADDR[20]	CPU_OE	CPU_TSALE	CPU_DREQ1	N/C	CPU_DATA[4]	CPU_DATA[9]	CPU_DATA[13]	CPU_DATA[18]	CPU_DATA[25]	CPU_DATA[28]	N/C	N/C	N/C	AE
AF	GND	RAM_DATA[50]	RAM_DATA[55]	RAM_DATA[56]	RAM_DATA[61]	TEST_MODE[2]	JTAG_TDI	PULL_LO	CPU_ADDR[5]	CPU_ADDR[10]	CPU_ADDR[15]	CPU_ADDR[19]	GND	CPU_CS	CPU_SDACK1	PULL_HI	CPU_IREQ0	CPU_DATA[0]	CPU_DATA[5]	CPU_DATA[2]	CPU_DATA[11]	CPU_DATA[17]	CPU_DATA[19]	CPU_DATA[26]	CPU_DATA[31]	GND	AF
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	

Figure 11 - ZL50114 Package View and Ball Positions

Ball Signal Assignment

Ball Number	Signal Name
A1	GND
A2	TDM_STo[1]
A3	TDM_CLKo[3]
A4‡	TDM_STo[4]
A5‡	TDM_STo[5]
A6‡	TDM_STi[6]
A7‡	TDM_STo[7]
A8‡	TDM_STi[7]
A9†	TDM_CLKo[10]
A10†	TDM_CLKi[10]
A11†	TDM_CLKi[11]
A12†	TDM_CLKo[13]
A13	GND
A14†	TDM_STo[13]
A15†	TDM_STo[14]
A16†	TDM_CLKo[15]
A17†	TDM_STo[16]
A18†	TDM_CLKo[18]
A19†	TDM_STi[18]
A20†	TDM_CLKi[20]
A21†	TDM_STi[20]
A22†	TDM_STo[21]
A23†	TDM_STi[21]
A24†	TDM_CLKo[24]
A25†	TDM_CLKo[25]
A26	GND
B1	TDM_FRMo_REF
B2	TDM_STo[0]
B3	TDM_STi[2]
B4	TDM_CLKi[3]
B5‡	TDM_STi[4]
B6‡	TDM_CLKo[6]
B7‡	TDM_STo[6]

Ball Number	Signal Name
B8†	TDM_CLKo[8]
B9†	TDM_CLKi[9]
B10†	TDM_STo[10]
B11†	TDM_STi[10]
B12†	TDM_CLKi[12]
B13†	TDM_STo[12]
B14†	TDM_STi[13]
B15†	TDM_CLKi[15]
B16†	TDM_STi[15]
B17†	TDM_STi[17]
B18†	TDM_CLKi[18]
B19†	TDM_CLKo[20]
B20†	TDM_STo[19]
B21†	TDM_STo[22]
B22†	TDM_CLKo[23]
B23†	TDM_STo[24]
B24†	TDM_CLKo[26]
B25†	TDM_STi[24]
B26†	TDM_CLKo[27]
C1	TDM_CLKiP
C2	TDM_FRMi_REF
C3	TDM_CLKi_REF
C4	TDM_CLKo[1]
C5	TDM_STi[3]
C6	TDM_CLKo[2]
C7‡	TDM_CLKi[6]
C8‡	TDM_CLKi[7]
C9†	TDM_CLKo[9]
C10†	TDM_STo[9]
C11†	TDM_STi[9]
C12†	TDM_STi[11]
C13†	TDM_CLKi[13]
C14†	TDM_CLKo[14]
C15†	TDM_CLKo[16]

Ball Number	Signal Name
C16†	TDM_STi[16]
C17†	TDM_CLKo[17]
C18†	TDM_STi[19]
C19†	TDM_CLKo[21]
C20†	TDM_CLKi[21]
C21†	TDM_CLKi[24]
C22†	TDM_STi[22]
C23†	TDM_STo[26]
C24†	TDM_CLKi[27]
C25†	TDM_STi[27]
C26†	TDM_STi[28]
D1	RAM_DATA[3]
D2	RAM_DATA[1]
D3	TDM_CLKiS
D4	RAM_DATA[0]
D5	TDM_STi[0]
D6	TDM_CLKi[1]
D7	TDM_STo[3]
D8‡	TDM_STi[5]
D9‡	TDM_CLKi[5]
D10‡	TDM_CLKo[7]
D11†	TDM_STi[8]
D12†	TDM_CLKo[11]
D13†	TDM_STi[12]
D14†	TDM_STi[14]
D15†	TDM_CLKi[16]
D16†	TDM_CLKo[19]
D17†	TDM_STo[18]
D18†	TDM_STo[20]
D19†	TDM_CLKo[22]
D20†	TDM_STo[27]
D21†	TDM_STo[25]
D22†	TDM_CLKi[26]
D23†	TDM_CLKo[28]

Ball Number	Signal Name
D24†	TDM_CLKi[29]
D25†	TDM_STi[29]
D26†	TDM_STi[31]
E1	RAM_DATA[10]
E2	RAM_DATA[9]
E3	RAM_DATA[5]
E4	RAM_DATA[4]
E5	RAM_DATA[2]
E6	TDM_CLKo_REF
E7	TDM_CLKi[0]
E8‡	TDM_CLKo[4]
E9	TDM_STi[1]
E10‡	TDM_CLKi[4]
E11†	TDM_STo[8]
E12†	TDM_CLKi[8]
E13†	TDM_CLKo[12]
E14†	TDM_STo[15]
E15†	TDM_CLKi[17]
E16†	TDM_CLKi[19]
E17†	TDM_STo[23]
E18†	TDM_STi[23]
E19†	TDM_CLKi[25]
E20†	TDM_STi[26]
E21†	TDM_CLKi[28]
E22	GND
E23†	TDM_CLKo[30]
E24†	TDM_CLKi[30]
E25†	TDM_STi[30]
E26†	TDM_STo[29]
F1	RAM_DATA[15]
F2	RAM_DATA[13]
F3	RAM_DATA[12]
F4	RAM_DATA[6]
F5	RAM_DATA[7]

Ball Number	Signal Name
F6	GND
F7	VDD_CORE
F8	TDM_STo[2]
F9	TDM_CLKo[0]
F10	TDM_CLKi[2]
F11‡	TDM_CLKo[5]
F12	VDD_CORE
F13†	TDM_STo[11]
F14†	TDM_CLKi[14]
F15	VDD_CORE
F16†	TDM_STo[17]
F17†	TDM_CLKi[22]
F18†	TDM_STi[25]
F19†	TDM_CLKi[23]
F20	VDD_CORE
F21	GND
F22†	TDM_CLKi[31]
F23†	TDM_CLKo[29]
F24†	TDM_STo[28]
F25†	TDM_CLKo[31]
F26†	M2_LINKUP_LED
G1	RAM_DATA[21]
G2	RAM_DATA[18]
G3	RAM_DATA[16]
G4	RAM_DATA[14]
G5	RAM_DATA[11]
G6	RAM_DATA[8]
G21†	TDM_STo[31]
G22†	TDM_STo[30]
G23	M1_LINKUP_LED
G24	M0_LINKUP_LED
G25	M1_GIGABIT_LED
G26	M_MDIO
H1	RAM_DATA[25]

Ball Number	Signal Name
H2	RAM_DATA[24]
H3	RAM_DATA[23]
H4	RAM_DATA[19]
H5	RAM_DATA[17]
H6	VDD_CORE
H21	VDD_CORE
H22	M0_GIGABIT_LED
H23	M_MDC
H24†	M3_CRS
H25†	M3_TXCLK
H26†	M3_RXER
J1	RAM_DATA[29]
J2	RAM_DATA[28]
J3	RAM_DATA[27]
J4	RAM_DATA[26]
J5	RAM_DATA[22]
J6	RAM_DATA[20]
J9	VDD_IO
J10	VDD_IO
J11	VDD_IO
J12	VDD_IO
J13	VDD_IO
J14	VDD_IO
J15	VDD_IO
J16	VDD_IO
J17	VDD_IO
J18	VDD_IO
J21†	M3_RXDV
J22†	M3_RXD[3]
J23†	M3_RXD[2]
J24†	M3_RXD[1]
J25†	M3_RXD[0]
J26†	M3_COL
K1	RAM_PARITY[1]

Ball Number	Signal Name
K2	RAM_PARITY[0]
K3	RAM_DATA[31]
K4	RAM_DATA[30]
K5	GND
K6	VDD_CORE
K9	VDD_IO
K18	VDD_IO
K21	VDD_CORE
K22	GND
K23†	M3_TXD[3]
K24†	M3_TXEN
K25†	M3_TXER
K26†	M3_RXCLK
L1	RAM_PARITY[7]
L2	RAM_PARITY[6]
L3	RAM_PARITY[5]
L4	RAM_PARITY[4]
L5	RAM_PARITY[3]
L6	RAM_PARITY[2]
L9	VDD_IO
L11	GND
L12	GND
L13	GND
L14	GND
L15	GND
L16	GND
L18	VDD_IO
L21	M1_RXER
L22	M1_TXCLK
L23	M1_CRS
L24†	M3_TXD[0]
L25†	M3_TXD[1]
L26†	M3_TXD[2]
M1	RAM_ADDR[5]

Ball Number	Signal Name
M2	RAM_ADDR[4]
M3	RAM_ADDR[2]
M4	RAM_ADDR[3]
M5	RAM_ADDR[0]
M6	RAM_ADDR[1]
M9	VDD_IO
M11	GND
M12	GND
M13	GND
M14	GND
M15	GND
M16	GND
M18	VDD_IO
M21	VDD_CORE
M22	M1_REFCLK
M23	M1_RXCLK
M24	M1_RXD[5]
M25	M1_RXD[7]
M26	M1_RXDV
N1	GND
N2	RAM_ADDR[6]
N3	RAM_ADDR[7]
N4	RAM_ADDR[8]
N5	GND
N6	VDD_CORE
N9	VDD_IO
N11	GND
N12	GND
N13	GND
N14	GND
N15	GND
N16	GND
N18	VDD_IO
N21	M1_GTX_CLK

Ball Number	Signal Name
N22	GND
N23	M1_TXER
N24	M1_RXD[2]
N25	M1_RXD[3]
N26	GND
P1	RAM_ADDR[9]
P2	RAM_ADDR[10]
P3	RAM_ADDR[11]
P4	RAM_ADDR[13]
P5	RAM_ADDR[16]
P6	GND
P9	VDD_IO
P11	GND
P12	GND
P13	GND
P14	GND
P15	GND
P16	GND
P18	VDD_IO
P21	M1_TXD[2]
P22	M1_TXD[6]
P23	M1_TXEN
P24	GND
P25	M1_RXD[4]
P26	M1_RXD[6]
R1	RAM_ADDR[12]
R2	RAM_ADDR[14]
R3	RAM_ADDR[15]
R4	RAM_ADDR[19]
R5	N/C
R6	N/C
R9	VDD_IO
R11	GND
R12	GND

Ball Number	Signal Name
R13	GND
R14	GND
R15	GND
R16	GND
R18	VDD_IO
R21	M1_TXD[0]
R22	M1_TXD[3]
R23	M1_TXD[5]
R24	M1_TXD[7]
R25	M1_COL
R26	M1_RXD[1]
T1	RAM_ADDR[17]
T2	RAM_ADDR[18]
T3	RAM_BW_B
T4	N/C
T5	GND
T6	A1VDD
T9	VDD_IO
T11	GND
T12	GND
T13	GND
T14	GND
T15	GND
T16	GND
T18	VDD_IO
T21	VDD_CORE
T22	M1_TXD[1]
T23	M1_TXD[4]
T24	GND
T25	M1_RBC1
T26	M1_RXD[0]
U1	PLL_PRI
U2	RAM_BW_A
U3	RAM_BW_C

Ball Number	Signal Name
U4	RAM_RW
U5	SYSTEM_DEBUG
U6	SYSTEM_CLK
U9	VDD_IO
U18	VDD_IO
U21	M0_GTX_CLK
U22	M0_RXD[2]
U23	M0_RXD[5]
U24	M0_TXCLK
U25	M0_CRS
U26	M1_RBC0
V1	PLL_SEC
V2	RAM_BW_D
V3	RAM_BW_F
V4	SYSTEM_RST
V5	GPIO[2]
V6	VDD_CORE
V9	VDD_IO
V10	VDD_IO
V11	VDD_IO
V12	VDD_IO
V13	VDD_IO
V14	VDD_IO
V15	VDD_IO
V16	VDD_IO
V17	VDD_IO
V18	VDD_IO
V21	M0_TXD[7]
V22	M0_TXER
V23	M0_TXEN
V24	M0_RXD[4]
V25	M0_RXDV
V26	M0_RXER
W1	RAM_BW_E

Ball Number	Signal Name
W2	RAM_BW_G
W3	GPIO[0]
W4	GPIO[3]
W5	GPIO[9]
W6	RAM_DATA[33]
W21	M0_TXD[2]
W22	M0_TXD[5]
W23	M0_TXD[6]
W24	M0_RXD[6]
W25	M0_RXD[7]
W26	M0_RXD[3]
Y1	RAM_BW_H
Y2	GPIO[4]
Y3	GPIO[6]
Y4	GPIO[10]
Y5	RAM_DATA[32]
Y6	VDD_CORE
Y21	VDD_CORE
Y22	M0_TXD[1]
Y23	M0_TXD[4]
Y24	M0_RBC0
Y25	M0_COL
Y26	M0_RXD[1]
AA1	GPIO[1]
AA2	GPIO[7]
AA3	GPIO[8]
AA4	GPIO[15]
AA5	RAM_DATA[39]
AA6	GND
AA7	RAM_DATA[45]
AA8	RAM_DATA[52]
AA9	VDD_CORE
AA10	JTAG_TMS
AA11	CPU_ADDR[2]

Ball Number	Signal Name
AA12	CPU_ADDR[12]
AA13	VDD_CORE
AA14	VDD_CORE
AA15	CPU_DATA[8]
AA16	CPU_DATA[15]
AA17	CPU_DATA[23]
AA18	VDD_CORE
AA19†	M2_RXCLK
AA20†	M2_RXDV
AA21	GND
AA22	M0_TXD[0]
AA23	M0_TXD[3]
AA24	M0_REFCLK
AA25	M0_RBC1
AA26	M0_RXD[0]
AB1	GPIO[5]
AB2	GPIO[11]
AB3	GPIO[14]
AB4	RAM_DATA[38]
AB5	RAM_DATA[43]
AB6	RAM_DATA[44]
AB7	RAM_DATA[51]
AB8	RAM_DATA[60]
AB9	TEST_MODE[1]
AB10	GND
AB11	CPU_ADDR[6]
AB12	CPU_ADDR[14]
AB13	CPU_ADDR[23]
AB14	CPU_TA
AB15	CPU_DATA[1]
AB16	CPU_DATA[7]
AB17	CPU_DATA[12]
AB18	CPU_DATA[22]
AB19	CPU_DATA[30]

Ball Number	Signal Name
AB20†	M2_TXER
AB21†	M2_RXD[1]
AB22	M0_RXCLK
AB23†	M3_LINKUP_LED
AB24†	M2_ACTIVE_LED
AB25	M1_ACTIVE_LED
AB26†	M3_ACTIVE_LED
AC1	GPIO[12]
AC2	GPIO[13]
AC3	RAM_DATA[37]
AC4	RAM_DATA[42]
AC5	RAM_DATA[46]
AC6	RAM_DATA[49]
AC7	RAM_DATA[59]
AC8	TEST_MODE[0]
AC9	JTAG_TDO
AC10	CPU_ADDR[4]
AC11	CPU_ADDR[9]
AC12	CPU_ADDR[16]
AC13	CPU_ADDR[22]
AC14	CPU_CLK
AC15	CPU_DREQ0
AC16	N/C
AC17	CPU_DATA[10]
AC18	CPU_DATA[16]
AC19	CPU_DATA[21]
AC20	CPU_DATA[27]
AC21†	M2_TXD[1]
AC22†	M2_TXEN
AC23†	M2_RXD[2]
AC24†	M2_RXER
AC25†	M2_CRS
AC26	M0_ACTIVE_LED
AD1	RAM_DATA[34]

Ball Number	Signal Name
AD2	RAM_DATA[36]
AD3	RAM_DATA[41]
AD4	RAM_DATA[47]
AD5	RAM_DATA[53]
AD6	RAM_DATA[58]
AD7	RAM_DATA[63]
AD8	JTAG_TCK
AD9	PULL_LO
AD10	CPU_ADDR[7]
AD11	CPU_ADDR[11]
AD12	CPU_ADDR[17]
AD13	CPU_ADDR[21]
AD14	CPU_WE
AD15	CPU_SDACK2
AD16	CPU_IREQ1
AD17	CPU_DATA[3]
AD18	CPU_DATA[6]
AD19	CPU_DATA[14]
AD20	CPU_DATA[20]
AD21	CPU_DATA[24]
AD22	CPU_DATA[29]
AD23†	M2_TXD[2]
AD24†	M2_RXD[0]
AD25†	M2_RXD[3]
AD26†	M2_TXCLK
AE1	RAM_DATA[35]
AE2	RAM_DATA[40]
AE3	RAM_DATA[48]
AE4	RAM_DATA[54]
AE5	RAM_DATA[57]
AE6	RAM_DATA[62]
AE7	JTAG_TRST
AE8	N/C
AE9	CPU_ADDR[3]

Ball Number	Signal Name
AE10	CPU_ADDR[8]
AE11	CPU_ADDR[13]
AE12	CPU_ADDR[18]
AE13	CPU_ADDR[20]
AE14	CPU_OE
AE15	CPU_TS_ALE
AE16	CPU_DREQ1
AE17	N/C
AE18	CPU_DATA[4]
AE19	CPU_DATA[9]
AE20	CPU_DATA[13]
AE21	CPU_DATA[18]
AE22	CPU_DATA[25]
AE23	CPU_DATA[28]
AE24†	M2_TXD[0]
AE25†	M2_TXD[3]
AE26†	M2_COL
AF1	GND
AF2	RAM_DATA[50]
AF3	RAM_DATA[55]
AF4	RAM_DATA[56]
AF5	RAM_DATA[61]
AF6	TEST_MODE[2]
AF7	JTAG_TDI
AF8	PULL_LO
AF9	CPU_ADDR[5]
AF10	CPU_ADDR[10]
AF11	CPU_ADDR[15]
AF12	CPU_ADDR[19]
AF13	GND
AF14	CPU_CS
AF15	CPU_SDACK1
AF16	PULL_HI
AF17	CPU_IREQ0

Ball Number	Signal Name
AF18	CPU_DATA[0]
AF19	CPU_DATA[5]
AF20	CPU_DATA[2]
AF21	CPU_DATA[11]
AF22	CPU_DATA[17]
AF23	CPU_DATA[19]
AF24	CPU_DATA[26]
AF25	CPU_DATA[31]
AF26	GND

† Not Connected on ZL50110 and ZL50114 - leave open circuit.
‡ Not Connected on ZL50114 - leave open circuit.
N/C - Not Connected - leave open circuit.

4.0 External Interface Description

The following key applies to all tables:

I Input

O Output

D Internal 100kΩ pull-down resistor present

U Internal 100kΩ pull-up resistor present

T Tri-state Output

4.1 TDM Interface

All TDM Interface signals are 5V tolerant.

All TDM Interface outputs are high impedance while System Reset is LOW.

All TDM Interface inputs (including data, clock and frame pulse) have internal pull-down resistors so they can be safely left unconnected if not used.

4.1.1 ZL50111 Variant TDM stream connection

Signal	I/O	Package Balls				Description
TDM_STi[31:0]	I D	[31]	D26	[15]	B16	TDM port serial data input streams. For different standards these pins are given different identities: ST-BUS: TDM_STi[31:0] H.110: TDM_D[31:0] H-MVIP: TDM_HDS[31:0] Triggered on rising edge or falling edge depending on standard. At 8.192Mbit/s only streams [7:0] are used, with 128 channels per stream. Streams [7:0] are used for J2, and streams [1:0] are used for T3 or E3.
		[30]	E25	[14]	D14	
		[29]	D25	[13]	B14	
		[28]	C26	[12]	D13	
		[27]	C25	[11]	C12	
		[26]	E20	[10]	B11	
		[25]	F18	[9]	C11	
		[24]	B25	[8]	D11	
		[23]	E18	[7]	A8	
		[22]	C22	[6]	A6	
		[21]	A23	[5]	D8	
		[20]	A21	[4]	B5	
		[19]	C18	[3]	C5	
		[18]	A19	[2]	B3	
		[17]	B17	[1]	E9	
		[16]	C16	[0]	D5	
TDM_STo[31:0]	OT	[31]	G21	[15]	E14	TDM port serial data output streams. For different standards these pins are given different identities: ST-BUS: TDM_STo[31:0] H.110: TDM_D[31:0] H-MVIP: TDM_HDS[31:0] Triggered on rising edge or falling edge depending on standard. At 8.192Mbit/s only streams [7:0] are used, with 128 channels per stream. Streams [7:0] are used for J2, and streams [1:0] are used for T3 or E3.
		[30]	G22	[14]	A15	
		[29]	E26	[13]	A14	
		[28]	F24	[12]	B13	
		[27]	D20	[11]	F13	
		[26]	C23	[10]	B10	
		[25]	D21	[9]	C10	
		[24]	B23	[8]	E11	
		[23]	E17	[7]	A7	
		[22]	B21	[6]	B7	
		[21]	A22	[5]	A5	
		[20]	D18	[4]	A4	
		[19]	B20	[3]	D7	
		[18]	D17	[2]	F8	
		[17]	F16	[1]	A2	
		[16]	A17	[0]	B2	

Signal	I/O	Package Balls				Description
TDM_CLKi[31:0]	I D	[31]	F22	[15]	B15	TDM port clock inputs programmable as active high or low. Can accept frequencies of 1.544MHz, 2.048MHz, 4.096MHz, 6.312MHz, 8.192MHz, 16.384MHz, 34.368MHz or 44.736MHz depending on standard used. At 8.192Mbit/s only streams [7:0] are used. Streams [7:0] are used for J2, and streams [1:0] are used for T3 or E3.
		[30]	E24	[14]	F14	
		[29]	D24	[13]	C13	
		[28]	E21	[12]	B12	
		[27]	C24	[11]	A11	
		[26]	D22	[10]	A10	
		[25]	E19	[9]	B9	
		[24]	C21	[8]	E12	
		[23]	F19	[7]	C8	
		[22]	F17	[6]	C7	
		[21]	C20	[5]	D9	
		[20]	A20	[4]	E10	
		[19]	E16	[3]	B4	
		[18]	B18	[2]	F10	
		[17]	E15	[1]	D6	
		[16]	D15	[0]	E7	
TDM_CLKo[31:0]	O	[31]	F25	[15]	A16	TDM port clock outputs. Will generate 1.544MHz, 2.048MHz, 4.096MHz, 6.312MHz, 8.192MHz, 16.384MHz, 34.368MHz or 44.736MHz depending on standard used. At 8.192Mbit/s only streams [7:0] are used. Streams [7:0] are used for J2, and streams [1:0] are used for T3 or E3.
		[30]	E23	[14]	C14	
		[29]	F23	[13]	A12	
		[28]	D23	[12]	E13	
		[27]	B26	[11]	D12	
		[26]	B24	[10]	A9	
		[25]	A25	[9]	C9	
		[24]	A24	[8]	B8	
		[23]	B22	[7]	D10	
		[22]	D19	[6]	B6	
		[21]	C19	[5]	F11	
		[20]	B19	[4]	E8	
		[19]	D16	[3]	A3	
		[18]	A18	[2]	C6	
		[17]	C17	[1]	C4	
		[16]	C15	[0]	F9	

Table 3 - TDM Interface ZL50111 Stream Pin Definition

Speed modes:

- 2.048Mbits/s - all 32 streams active (bits [31:0]), with 32 channels per stream - 1024 total channels.
- 8.192Mbits/s - 8 streams active (bits [7:0]), with 128 channels per stream - 1024 total channels.
- J2 - 8 streams active (bits [7:0]), with 98 channels per stream - 784 total channels
- E3 - 2 streams active (bits [1:0]), with 537 channels per stream - 1074 total channels
- T3 - 2 streams active (bits [1:0]), with 699 channels per stream - 1398 total channels

All TDM Interface inputs (including data, clock and frame pulse) have internal pull-down resistors so they can be safely left unconnected if not used.

4.1.2 ZL50110 Variant TDM stream connection

Signal	I/O	Package Balls	Description
TDM_STi[7:0]	I D	[7] A8 [6] A6 [5] D8 [4] B5 [3] C5 [2] B3 [1] E9 [0] D5	TDM port serial data input streams. For different standards these pins are given different identities: ST-BUS: TDM_STi[7:0] H.110: TDM_D[7:0] H-MVIP: TDM_HDS[7:0] Triggered on rising edge or falling edge depending on standard. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.
TDM_STo[7:0]	OT	[7] A7 [6] B7 [5] A5 [4] A4 [3] D7 [2] F8 [1] A2 [0] B2	TDM port serial data output streams. For different standards these pins are given different identities: ST-BUS: TDM_STo[7:0] H.110: TDM_D[7:0] H-MVIP: TDM_HDS[7:0] Triggered on rising edge or falling edge depending on standard. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.
TDM_CLKi[7:0]	I D	[7] C8 [6] C7 [5] D9 [4] E10 [3] B4 [2] F10 [1] D6 [0] E7	TDM port clock inputs programmable as active high or low. Can accept frequencies of 1.544MHz, 2.048MHz, 4.096MHz, 8.192MHz, 6.312MHz or 16.384MHz depending on standard used. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.
TDM_CLKo[7:0]	OT	[7] D10 [6] B6 [5] F11 [4] E8 [3] A3 [2] C6 [1] C4 [0] F9	TDM port clock outputs. Will generate 1.544MHz, 2.048MHz, 4.096MHz, 6.312MHz, 8.192MHz or 16.384MHz depending on standard used. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.

Table 4 - TDM Interface ZL50110 Stream Pin Definition

Note: Speed modes:
2.048Mbits/s - all 8 streams active (bits [7:0]), with 32 channels per stream - 256 total channels.
8.192Mbits/s - 2 streams active (bits [1:0]), with 128 channels per stream - 256 total channels
J2 - 2 streams active (bits [1:0]), with 98 channels per stream - 196 total channels
Note: All TDM Interface inputs (including data, clock and frame pulse) have internal pull-down resistors so they can be safely left unconnected if not used.

4.1.3 ZL50114 Variant TDM stream connection

Signal	I/O	Package Balls	Description
TDM_STi[3:0]	I D	[3] C5 [2] B3 [1] E9 [0] D5	TDM port serial data input streams. For different standards these pins are given different identities: ST-BUS: TDM_STi[3:0] H.110: TDM_D[3:0] H-MVIP: TDM_HDS[3:0] Triggered on rising edge or falling edge depending on standard. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.
TDM_STo[3:0]	OT	[3] D7 [2] F8 [1] A2 [0] B2	TDM port serial data output streams. For different standards these pins are given different identities: ST-BUS: TDM_STo[3:0] H.110: TDM_D[3:0] H-MVIP: TDM_HDS[3:0] Triggered on rising edge or falling edge depending on standard. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.
TDM_CLKi[3:0]	I D	[3] B4 [2] F10 [1] D6 [0] E7	TDM port clock inputs programmable as active high or low. Can accept frequencies of 1.544MHz, 2.048MHz, 4.096MHz, 8.192MHz, 6.312MHz or 16.384MHz depending on standard used. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.
TDM_CLKo[3:0]	OT	[3] A3 [2] C6 [1] C4 [0] F9	TDM port clock outputs. Will generate 1.544MHz, 2.048MHz, 4.096MHz, 6.312MHz, 8.192MHz or 16.384MHz depending on standard used. At 8.192Mbit/s only streams [1:0] are used. Streams [1:0] are used for J2.

Table 5 - TDM Interface ZL50110 Stream Pin Definition

Note: Speed modes:
2.048Mbits/s - all 4 streams active (bits [3:0]), with 32 channels per stream - 128 total channels.
8.192Mbits/s - 2 streams active (bits [1:0]), with 128 channels per stream - 256 total channels.
J2 - 2 streams active (bits [1:0]), with 98 channels per stream - 196 total channels

Note: All TDM Interface inputs (including data, clock and frame pulse) have internal pull-down resistors so they can be safely left unconnected if not used.

4.1.4 TDM Signals common to ZL50111, ZL50110 and ZL50114

Signal	I/O	Package Balls	Description
TDM_CLKi_REF	I D	C3	TDM port reference clock input for backplane operation.
TDM_CLKo_REF	O	E6	TDM port reference clock output for backplane operation.
TDM_FRMi_REF	I D	C2	TDM port reference frame input. For different standards this pin is given a different identity: ST-BUS: TDM_F0i H.110: TDM_FRAME H-MVIP: TDM_F0 Signal is normally active low, but can be active high depending on standard. Indicates the start of a TDM frame by pulsing every 125µs. Normally will straddle rising edge or falling edge of clock pulse, depending on standard and clock frequency.
TDM_FRMo_REF	O	B1	TDM port reference frame output. For different standards this pin is given a different identity: ST-BUS: TDM_F0o H.110: TDM_FRAME H-MVIP: TDM_F0 Signal is normally active low, but can be active high depending on standard. Indicates the start of a TDM frame by pulsing every 125µs. Normally will straddle rising edge or falling edge of clock pulse, depending on standard and clock frequency.

Table 6 - TDM Interface Common Pin Definition

4.2 PAC Interface

All PAC Interface signals are 5V tolerant

All PAC Interface outputs are high impedance while System Reset is LOW.

Signal	I/O	Package Balls	Description
TDM_CLKiP	I D	C1	Primary reference clock input. Should be driven by external clock source to provide locking reference to internal / optional external DPLL in TDM master mode. Also provides PRS clock for RTP timestamps in synchronous modes. Acceptable frequency range: 8kHz - 34.368MHz.
TDM_CLKiS	I D	D3	Secondary reference clock input. Backup external reference for automatic switch-over in case of failure of TDM_CLKiP source.
PLL_PRI	OT	U1	Primary reference output to optional external DPLL. Multiplexed & frequency divided reference output for support of optional external DPLL. Expected frequency range: 8kHz - 16.384MHz.
PLL_SEC	OT	V1	Secondary reference output to optional external DPLL Multiplexed & frequency divided reference output for support of optional external DPLL. Expected frequency range: 8kHz - 16.384MHz.

Table 7 - PAC Interface Package Ball Definition

4.3 Packet Interfaces

For the ZL50111 variant the packet interface is capable of either 3 MII interfaces, 2 GMII interfaces or 2 PCS (1000Mbit/s) interfaces. The PCS interface is a TBI interface supported by an integrated 1000BASE-X PCS module. The ZL50110 variant has either 2 MII interfaces, 2 GMII interfaces or 2 PCS (1000Mbit/s) interfaces. Ports 2 and 3 are not available on the ZL50110 device.

NOTE: In GMII/PCS mode only 1 GMAC port may be used to receive data. The second GMAC port is for redundancy purposes only.

Data for all three types of packet switching is based on Specification IEEE Std. 802.3 - 2000. For the ZL50111 variant, only Ports 0 and 1 have the 1000 Mbit/s capability necessary for the GMII/PCS interface. In either GMII or PCS mode Ports 2 and 3 are disabled. Alternatively 3 ports can be used as 100Mbit/s MII interfaces, either Ports 0, 1 and 2 or Ports 0, 1 and 3.

Note: Port 2 and Port 3 can not be used to receive data simultaneously, they are mutually exclusive for packet reception. They may both be used for packet transmission if required.

Table 8 maps the signal pins used in the MII interface to those used in the GMII and PCS interface. Table 9 shows all the pins and their related package ball, but is based on the GMII/MII configuration.

All Packet Interface signals are 5V tolerant, and all outputs are high impedance while System Reset is LOW.

MII	GMII	PCS (TBI)
M*n*_LINKUP_LED	M*n*_LINKUP_LED	M*n*_LINKUP_LED
M*n*_ACTIVE_LED	M*n*_ACTIVE_LED	M*n*_ACTIVE_LED
-	M*n*_GIGABIT_LED	M*n*_GIGABIT_LED
-	M*n*_REFCLK	M*n*_REFCLK
M*n*_RXCLK	M*n*_RXCLK	M*n*_RBC0
M*n*_COL	M*n*_COL	M*n*_RBC1
M*n*_RXD[3:0]	M*n*_RXD[7:0]	M*n*_RXD[7:0]
M*n*_RXDV	M*n*_RXDV	M*n*_RXD[8]
M*n*_RXER	M*n*_RXER	M*n*_RXD[9]
M*n*_CRS	M*n*_CRS	M*n*_Signal_Detect
M*n*_TXCLK	-	-
M*n*_TXD[3:0]	M*n*_TXD[7:0]	M*n*_TXD[7:0]
M*n*_TXEN	M*n*_TXEN	M*n*_TXD[8]
M*n*_TXER	M*n*_TXER	M*n*_TXD[9]
-	M*n*_GTX_CLK	M*n*_GTX_CLK

Table 8 - Packet Interface Signal Mapping - MII to GMII/PCS

Note: M*n* can be either M0, M1, M2, or M3 for ZL50111 variant; and M0 or M1 for ZL50110 variant

Signal	I/O	Package Balls	Description
M_MDC	O	H23	MII management data clock. Common for all four MII ports. It has a minimum period of 400ns (maximum freq. 2.5MHz), and is independent of the TXCLK and RXCLK.
M_MDIO	ID/ OT	G26	MII management data I/O. Common for all four MII ports at up to 2.5 MHz. It is bi-directional between the ZL50110/1/4 and the Ethernet station management entity. Data is passed synchronously with respect to M_MDC.

Table 9 - MII Management Interface Package Ball Definition

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_LINKUP_LED	O	G24	LED drive for MAC 0 to indicate port is linked up. Logic 0 output = LED on Logic 1 output = LED off
M0_ACTIVE_LED	O	AC26	LED drive for MAC 0 to indicate port is transmitting or receiving packet data. Logic 0 output = LED on Logic 1 output = LED off
M0_GIGABIT_LED	O	H22	LED drive for MAC 0 to indicate operation at Gigabit/s Logic 0 output = LED on Logic 1 output = LED off
M0_REFCLK	I D	AA24	**GMII/PCS** - Reference Clock input at 125MHz. Can be used to lock receive circuitry (RX) to M0_GTXCLK rather than recovering the RXCLK (or RBC0 and RBC1). Useful, for example, in the absence of valid serial data. **NOTE:** In **MII** mode this pin must be driven with the same clock as M0_RXCLK.

Table 10 - MII Port 0 Interface Package Ball Definition

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_RXCLK	I U	AB22	GMII/MII - M0_RXCLK. Accepts the following frequencies: 25.0 MHz MII 100Mbit/s 125.0 MHz GMII 1Gbit/s
M0_RBC0	I U	Y24	PCS - M0_RBC0. Used as a clock when in PCS mode. Accepts 62.5MHz, and is 180° out of phase with M0_RBC1. Receive data is clocked at each rising edge of M1_RBC1 and M1_RBC0, resulting in 125MHz sample rate.
M0_RBC1	I U	AA25	PCS - M0_RBC1 Used as a clock when in PCS mode. Accepts 62.5MHz, and is 180° out of phase with M0_RBC0. Receive data is clocked at each rising edge of M0_RBC1 and M0_RBC0, resulting in 125MHz sample rate.
M0_COL	I D	Y25	GMII/MII - M0_COL. Collision Detection. This signal is independent of M0_TXCLK and M0_RXCLK, and is asserted when a collision is detected on an attempted transmission. It is active high, and only specified for half-duplex operation.
M0_RXD[7:0]	I U	[7] W25 [3] W26 [6] W24 [2] U22 [5] U23 [1] Y26 [4] V24 [0] AA26	Receive Data. Only half the bus (bits [3:0]) are used in MII mode. Clocked on rising edge of M0_RXCLK (GMII/MII) or the rising edges of M0_RBC0 and M0_RBC1 (PCS).

Table 10 - MII Port 0 Interface Package Ball Definition

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_RXDV / M0_RXD[8]	I D	V25	GMII/MII - M0_RXDV Receive Data Valid. Active high. This signal is clocked on the rising edge of M0_RXCLK. It is asserted when valid data is on the M0_RXD bus. PCS - M0_RXD[8] Receive Data. Clocked on the rising edges of M0_RBC0 and M0_RBC1.
M0_RXER / M0_RXD[9]	I D	V26	GMII/MII - M0_RXER Receive Error. Active high signal indicating an error has been detected. Normally valid when M0_RXDV is asserted. Can be used in conjunction with M0_RXD when M0_RXDV signal is de-asserted to indicate a False Carrier. PCS - M0_RXD[9] Receive Data. Clocked on the rising edges of M0_RBC0 and M0_RBC1
M0_CRS / M0_Signal_Detect	I D	U25	GMII/MII - M0_CRS Carrier Sense. This asynchronous signal is asserted when either the transmission or reception device is non-idle. It is active high. PCS - M0_Signal Detect Similar function to M0_CRS.
M0_TXCLK	I U	U24	**MII only** - Transmit Clock Accepts the following frequencies: 25.0 MHz MII 100Mbit/s
M0_TXD[7:0]	O	[7] V21 [3] AA23 [6] W23 [2] W21 [5] W22 [1] Y22 [4] Y23 [0] AA22	Transmit Data. Only half the bus (bits [3:0]) are used in MII mode. Clocked on rising edge of M0_TXCLK (MII) or the rising edge of M0_GTXCLK (GMII/PCS).

Table 10 - MII Port 0 Interface Package Ball Definition

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_TXEN / M0_TXD[8]	O	V23	GMII/MII - M0_TXEN Transmit Enable. Asserted when the MAC has data to transmit, synchronously to M0_TXCLK with the first pre-amble of the packet to be sent. Remains asserted until the end of the packet transmission. Active high. PCS - M0_TXD[8] Transmit Data. Clocked on rising edge of M0_GTXCLK
M0_TXER / M0_TXD[9]	O	V22	GMII/MII - M0_TXER Transmit Error. Transmitted synchronously with respect to M0_TXCLK, and active high. When asserted (with M0_TXEN also asserted) the ZL50110/1/4 will transmit a non-valid symbol, somewhere in the transmitted frame. PCS - M0_TXD[9] Transmit Data. Clocked on rising edge of M0_GTXCLK
M0_GTX_CLK	O	U21	**GMII/PCS only** - Gigabit Transmit Clock Output of a clock for Gigabit operation at 125MHz.

Table 10 - MII Port 0 Interface Package Ball Definition

MII Port 1			
Signal	**I/O**	**Package Balls**	**Description**
M1_LINKUP_LED	O	G23	LED drive for MAC 1 to indicate port is linked up. Logic 0 output = LED on Logic 1 output = LED off
M1_ACTIVE_LED	O	AB25	LED drive for MAC 1 to indicate port is transmitting or receiving packet data. Logic 0 output = LED on Logic 1 output = LED off

Table 11 - MII Port 1 Interface Package Ball Definition

MII Port 1			
Signal	**I/O**	**Package Balls**	**Description**
M1_GIGABIT_LED	O	G25	LED drive for MAC 1 to indicate operation at Gigabit/s Logic 0 output = LED on Logic 1 output = LED off
M1_REFCLK	I D	M22	**GMII/PCS** - Reference Clock input at 125MHz. Can be used to lock receive circuitry (RX) to M1_GTXCLK rather than recovering the RXCLK (or RBC0 and RBC1). Useful, for example, in the absence of valid serial data. **NOTE:** In **MII** mode this pin must be driven with the same clock as M1_RXCLK.
M1_RXCLK	I U	M23	GMII/MII - M1_RXCLK. Accepts the following frequencies: 25.0 MHz MII 100Mbit/s 125.0 MHz GMII 1Gbit/s
M1_RBC0	I U	U26	PCS - M1_RBC0. Used as a clock when in PCS mode. Accepts 62.5MHz, and is 180° out of phase with M1_RBC1. Receive data is clocked at each rising edge of M1_RBC1 and M1_RBC0, resulting in 125MHz sample rate.
M1_RBC1	I U	T25	PCS - M1_RBC1 Used as a clock when in PCS mode. Accepts 62.5MHz, and is 180° out of phase with M1_RBC0. Receive data is clocked at each rising edge of M1_RBC1 and M1_RBC0, resulting in 125MHz sample rate.
M1_COL	I D	R25	GMII/MII - M1_COL. Collision Detection. This signal is independent of M1_TXCLK and M1_RXCLK, and is asserted when a collision is detected on an attempted transmission. It is active high, and only specified for half-duplex operation.

Table 11 - MII Port 1 Interface Package Ball Definition

MII Port 1			
Signal	**I/O**	**Package Balls**	**Description**
M1_RXD[7:0]	I U	[7] M25 [3] N25 [6] P26 [2] N24 [5] M24 [1] R26 [4] P25 [0] T26	Receive Data. Only half the bus (bits [3:0]) are used in MII mode. Clocked on rising edge of M1_RXCLK (GMII/MII) or the rising edges of M1_RBC0 and M1_RBC1 (PCS).
M1_RXDV / M1_RXD[8]	I D	M26	GMII/MII - M1_RXDV Receive Data Valid. Active high. This signal is clocked on the rising edge of M1_RXCLK. It is asserted when valid data is on the M1_RXD bus. PCS - M1_RXD[8] Receive Data. Clocked on the rising edges of M1_RBC0 and M1_RBC1.
M1_RXER / M1_RXD[9]	I D	L21	GMII/MII - M1_RXER Receive Error. Active high signal indicating an error has been detected. Normally valid when M1_RXDV is asserted. Can be used in conjunction with M1_RXD when M1_RXDV signal is de-asserted to indicate a False Carrier. PCS - M1_RXD[9] Receive Data. Clocked on the rising edges of M1_RBC0 and M1_RBC1
M1_CRS / M1_Signal_Detect	I D	L23	GMII/MII - M1_CRS Carrier Sense. This asynchronous signal is asserted when either the transmission or reception device is non-idle. It is active high. PCS - M1_Signal Detect Similar function to M1_CRS.
M1_TXCLK	I U	L22	**MII only** - Transmit Clock Accepts the following frequencies: 25.0 MHz MII 100Mbit/s
M1_TXD[7:0]	O	[7] R24 [3] R22 [6] P22 [2] P21 [5] R23 [1] T22 [4] T23 [0] R21	Transmit Data. Only half the bus (bits [3:0]) are used in MII mode. Clocked on rising edge of M1_TXCLK (MII) or the rising edge of M1_GTXCLK (GMII/PCS).

Table 11 - MII Port 1 Interface Package Ball Definition

MII Port 1			
Signal	**I/O**	**Package Balls**	**Description**
M1_TXEN / M1_TXD[8]	O	P23	GMII/MII - M1_TXEN Transmit Enable. Asserted when the MAC has data to transmit, synchronously to M1_TXCLK with the first pre-amble of the packet to be sent. Remains asserted until the end of the packet transmission. Active high. PCS - M1_TXD[8] Transmit Data. Clocked on rising edge of M1_GTXCLK
M1_TXER / M1_TXD[9]	O	N23	GMII/MII - M1_TXER Transmit Error. Transmitted synchronously with respect to M1_TXCLK, and active high. When asserted (with M1_TXEN also asserted) the ZL50110/1/4 will transmit a non-valid symbol, somewhere in the transmitted frame. PCS - M1_TXD[9] Transmit Data. Clocked on rising edge of M1_GTXCLK
M1_GTX_CLK	O	N21	**GMII/PCS only** - Gigabit Transmit Clock Output of a clock for Gigabit operation at 125MHz.

Table 11 - MII Port 1 Interface Package Ball Definition

MII Port 2 - ZL50111 variant only. Note: This port must not be used to receive data at the same time as port 3, they are mutually exclusive.			
Signal	**I/O**	**Package Balls**	**Description**
M2_LINKUP_LED	O	F26	LED drive for MAC 2 to indicate port is linked up. Logic 0 output = LED on Logic 1 output = LED off
M2_ACTIVE_LED	O	AB24	LED drive for MAC 2 to indicate port is transmitting or receiving packet data. Logic 0 output = LED on Logic 1 output = LED off

Table 12 - MII Port 2 Interface Package Ball Definition

MII Port 2 - ZL50111 variant only. Note: This port must not be used to receive data at the same time as port 3, they are mutually exclusive.			
Signal	**I/O**	**Package Balls**	**Description**
M2_RXCLK	I U	AA19	MII only - Receive Clock. Accepts the following frequencies: 25.0 MHz MII 100Mbit/s
M2_COL	I D	AE26	Collision Detection. This signal is independent of M2_TXCLK and M2_RXCLK, and is asserted when a collision is detected on an attempted transmission. It is active high, and only specified for half-duplex operation.
M2_RXD[3:0]	I U	[3] AD25 [1] AB21 [2] AC23 [0] AD24	Receive Data. Clocked on rising edge of M2_RXCLK.
M2_RXDV	I D	AA20	Receive Data Valid. Active high. This signal is clocked on the rising edge of M2_RXCLK. It is asserted when valid data is on the M2_RXD bus.
M2_RXER	I D	AC24	Receive Error. Active high signal indicating an error has been detected. Normally valid when M2_RXDV is asserted. Can be used in conjunction with M2_RXD when M2_RXDV signal is de-asserted to indicate a False Carrier.
M2_CRS	I D	AC25	Carrier Sense. This asynchronous signal is asserted when either the transmission or reception device is non-idle. It is active high.
M2_TXCLK	I U	AD26	MII only - Transmit Clock Accepts the following frequencies: 25.0 MHz MII 100Mbit/s
M2_TXD[3:0]	O	[3] AE25 [1] AC21 [2] AD23 [0] AE24	Transmit Data. Clocked on rising edge of M2_TXCLK

Table 12 - MII Port 2 Interface Package Ball Definition

MII Port 2 - ZL50111 variant only. Note: This port must not be used to receive data at the same time as port 3, they are mutually exclusive.			
Signal	**I/O**	**Package Balls**	**Description**
M2_TXEN	O	AC22	Transmit Enable. Asserted when the MAC has data to transmit, synchronously to M2_TXCLK with the first pre-amble of the packet to be sent. Remains asserted until the end of the packet transmission. Active high.
M2_TXER	O	AB20	Transmit Error. Transmitted synchronously with respect to M2_TXCLK, and active high. When asserted (with M2_TXEN also asserted) the ZL50110/1/4 will transmit a non-valid symbol, somewhere in the transmitted frame.

Table 12 - MII Port 2 Interface Package Ball Definition

MII Port 3 - ZL50111 variant only Note: This port must not be used to receive data at the same time as port 2, they are mutually exclusive.			
Signal	**I/O**	**Package Balls**	**Description**
M3_LINKUP_LED	O	AB23	LED drive for MAC 3 to indicate port is linked up. Logic 0 output = LED on Logic 1 output = LED off
M3_ACTIVE_LED	O	AB26	LED drive for MAC 3 to indicate port is transmitting or receiving packet data. Logic 0 output = LED on Logic 1 output = LED off
M3_RXCLK	I U	K26	MII only - Receive Clock. Accepts the following frequencies: 25.0 MHz MII 100Mbit/s
M3_COL	I D	J26	Collision Detection. This signal is independent of M3_TXCLK and M3_RXCLK, and is asserted when a collision is detected on an attempted transmission. It is active high, and only specified for half-duplex operation.

Table 13 - MII Port 3 Interface Package Ball Definition

MII Port 3 - ZL50111 variant only Note: This port must not be used to receive data at the same time as port 2, they are mutually exclusive.			
Signal	**I/O**	**Package Balls**	**Description**
M3_RXD[3:0]	I U	[3] J22 [1] J24 [2] J23 [0] J25	Receive Data. Clocked on rising edge of M3_RXCLK.
M3_RXDV	I D	J21	Receive Data Valid. Active high. This signal is clocked on the rising edge of M3_RXCLK. It is asserted when valid data is on the M3_RXD bus.
M3_RXER	I D	H26	Receive Error. Active high signal indicating an error has been detected. Normally valid when M3_RXDV is asserted. Can be used in conjunction with M3_RXD when M3_RXDV signal is de-asserted to indicate a False Carrier.
M3_CRS	I D	H24	Carrier Sense. This asynchronous signal is asserted when either the transmission or reception device is non-idle. It is active high.
M3_TXCLK	I U	H25	MII only - Transmit Clock Accepts the following frequencies: 25.0 MHz MII 100Mbit/s
M3_TXD[3:0]	O	[3] K23 [1] L25 [2] L26 [0] L24	Transmit Data. Clocked on rising edge of M3_TXCLK
M3_TXEN	O	K24	Transmit Enable. Asserted when the MAC has data to transmit, synchronously to M3_TXCLK with the first pre-amble of the packet to be sent. Remains asserted until the end of the packet transmission. Active high.
M3_TXER	O	K25	Transmit Error. Transmitted synchronously with respect to M3_TXCLK, and active high. When asserted (with M3_TXEN also asserted) the ZL50110/1/4 will transmit a non-valid symbol, somewhere in the transmitted frame.

Table 13 - MII Port 3 Interface Package Ball Definition

4.4 External Memory Interface

All External Memory Interface signals are 5V tolerant.

All External Memory Interface outputs are high impedance while System Reset is LOW.

If the External Memory Interface is unused, all input pins may be left unconnected.

Active low signals are designated by a # suffix, in accordance with the convention used in common memory data sheets.

Signal	I/O	Package Balls				Description
RAM_DATA[63:0]	IU/ OT	[63]	AD7	[31]	K3	Buffer memory data. Synchronous to rising edge of SYSTEM_CLK.
		[62]	AE6	[30]	K4	
		[61]	AF5	[29]	J1	
		[60]	AB8	[28]	J2	
		[59]	AC7	[27]	J3	
		[58]	AD6	[26]	J4	
		[57]	AE5	[25]	H1	
		[56]	AF4	[24]	H2	
		[55]	AF3	[23]	H3	
		[54]	AE4	[22]	J5	
		[53]	AD5	[21]	G1	
		[52]	AA8	[20]	J6	
		[51]	AB7	[19]	H4	
		[50]	AF2	[18]	G2	
		[49]	AC6	[17]	H5	
		[48]	AE3	[16]	G3	
		[47]	AD4	[15]	F1	
		[46]	AC5	[14]	G4	
		[45]	AA7	[13]	F2	
		[44]	AB6	[12]	F3	
		[43]	AB5	[11]	G5	
		[42]	AC4	[10]	E1	
		[41]	AD3	[9]	E2	
		[40]	AE2	[8]	G6	
		[39]	AA5	[7]	F5	
		[38]	AB4	[6]	F4	
		[37]	AC3	[5]	E3	
		[36]	AD2	[4]	E4	
		[35	AE1	[3]	D1	
		[34]	AD1	[2]	E5	
		[33]	W6	[1]	D2	
		[32]	Y5	[0]	D4	
RAM_PARITY[7:0]	IU/ OT	[7]	L1	[3]	L5	Buffer memory parity. Synchronous to rising edge of SYSTEM_CLK. Bit [7] is parity for data byte [63:56], bit [0] is parity for data byte [7:0].
		[6]	L2	[2]	L6	
		[5]	L3	[1]	K1	
		[4]	L4	[0]	K2	

Table 14 - External Memory Interface Package Ball Definition

Signal	I/O	Package Balls	Description
RAM_ADDR[19:0]	O	[19] R4 [18] T2 [17] T1 [16] P5 [15] R3 [14] R2 [13] P4 [12] R1 [11] P3 [10] P2 [9] P1 [8] N4 [7] N3 [6] N2 [5] M1 [4] M2 [3] M4 [2] M3 [1] M6 [0] M5	Buffer memory address output. Synchronous to rising edge of SYSTEM_CLK.
RAM_BW_A#	O	U2	Synchronous Byte Write Enable A (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[7:0].
RAM_BW_B#	O	T3	Synchronous Byte Write Enable B (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[15:8].
RAM_BW_C#	O	U3	Synchronous Byte Write Enable C (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[23:16].
RAM_BW_D#	O	V2	Synchronous Byte Write Enable D (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[31:24].
RAM_BW_E#	O	W1	Synchronous Byte Write Enable E (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[39:32].
RAM_BW_F#	O	V3	Synchronous Byte Write Enable F (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[47:40].
RAM_BW_G#	O	W2	Synchronous Byte Write Enable G (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[55:48]

Table 14 - External Memory Interface Package Ball Definition

Signal	I/O	Package Balls	Description
RAM_BW_H#	O	Y1	Synchronous Byte Write Enable H (Active Low). Must be asserted same clock cycle as RAM_ADDR. Enables RAM_DATA[63:56].
RAM_RW#	O	U4	Read/Write Enable output Read = high Write = low

Table 14 - External Memory Interface Package Ball Definition

4.5 CPU Interface

All CPU Interface signals are 5V tolerant.

All CPU Interface outputs are high impedance while System Reset is LOW.

Signal	I/O	Package Balls	Description
CPU_DATA[31:0]	I/OT	[31] AF25 [30] AB19 [29] AD22 [28] AE23 [27] AC20 [26] AF24 [25] AE22 [24] AD21 [23] AA17 [22] AB18 [21] AC19 [20] AD20 [19] AF23 [18] AE21 [17] AF22 [16] AC18 [15] AA16 [14] AD19 [13] AE20 [12] AB17 [11] AF21 [10] AC17 [9] AE19 [8] AA15 [7] AB16 [6] AD18 [5] AF19 [4] AE18 [3] AD17 [2] AF20 [1] AB15 [0] AF18	CPU Data Bus. Bi-directional data bus, synchronously transmitted with CPU_CLK rising edge. NOTE: as with all ports in the ZL50110/1/4 device, CPU_DATA[0] is the least significant bit (lsb).
CPU_ADDR[23:2]	I	[23] AB13 [22] AC13 [21] AD13 [20] AE13 [19] AF12 [18] AE12 [17] AD12 [16] AC12 [15] AF11 [14] AB12 [13] AE11 [12] AA12 [11] AD11 [10] AF10 [9] AC11 [8] AE10 [7] AD10 [6] AB11 [5] AF9 [4] AC10 [3] AE9 [2] AA11	CPU Address Bus. Address input from processor to ZL50110/1/4, synchronously transmitted with CPU_CLK rising edge. NOTE: as with all ports in the ZL50110/1/4 device, CPU_ADDR[2] is the least significant bit (lsb).

Table 15 - CPU Interface Package Ball Definition

Signal	I/O	Package Balls	Description
CPU_CS	I U	AF14	CPU Chip Select. Synchronous to rising edge of CPU_CLK and active low. Is asserted with CPU_TS_ALE. Must be asserted with CPU_OE to asynchronously enable the CPU_DATA output during a read, including DMA read.
CPU_WE	I	AD14	CPU Write Enable. Synchronously asserted with respect to CPU_CLK rising edge, and active low. Used for CPU writes from the processor to registers within the ZL50110/1/4. Asserted one clock cycle after CPU_TS_ALE
CPU_OE	I	AE14	CPU Output Enable. Synchronously asserted with respect to CPU_CLK rising edge, and active low. Used for CPU reads from the processor to registers within the ZL50110/1/4. Asserted one clock cycle after CPU_TS_ALE. Must be asserted with CPU_CS to asynchronously enable the CPU_DATA output during a read, including DMA read.
CPU_TS_ALE	I	AE15	Synchronous input with rising edge of CPU_CLK. Latch Enable (ALE), active high signal. Asserted with CPU_CS, for a single clock cycle.
CPU_SDACK1	I	AF15	CPU/DMA 1 Acknowledge Input. Active low synchronous to CPU_CLK rising edge. Used to acknowledge request from ZL50110/1/4 for a DMA write transaction. Only used for DMA transfers, not for normal register access.

Table 15 - CPU Interface Package Ball Definition

Signal	I/O	Package Balls	Description
$\overline{\text{CPU_SDACK2}}$	I	AD15	CPU/DMA 2 Acknowledge Input Active low synchronous to CPU_CLK rising edge. Used to acknowledge request from ZL50110/1/4 for a DMA read transaction. Only used for DMA transfers, not for normal register access.
CPU_CLK	I	AC14	CPU PowerQUICC™ II Bus Interface clock input. 66MHz clock, with minimum of 6ns high/low time. Used to time all host interface signals into and out of ZL50110/1/4 device.
$\overline{\text{CPU_TA}}$	OT	AB14	CPU Transfer Acknowledge. Driven from tri-state condition on the negative clock edge of CPU_CLK following the assertion of CPU_CS. Active low, asserted from the rising edge of CPU_CLK. For a read, asserted when valid data is available at CPU_DATA. The data is then read by the host on the following rising edge of CPU_CLK. For a write, is asserted when the ZL50110/1/4 is ready to accept data from the host. The data is written on the rising edge of CPU_CLK following the assertion. Returns to tri-state from the negative clock edge of CPU_CLK following the de-assertion of CPU_CS.
$\overline{\text{CPU_DREQ0}}$	OT	AC15	CPU DMA 0 Request Output Active low synchronous to CPU_CLK rising edge. Asserted by ZL50110/1/4 to request the host initiates a DMA write. Only used for DMA transfers, not for normal register access.

Table 15 - CPU Interface Package Ball Definition

Signal	I/O	Package Balls	Description
$\overline{\text{CPU_DREQ1}}$	OT	AE16	CPU DMA 1 Request Active low synchronous to CPU_CLK rising edge. Asserted by ZL50110/1/4 to indicate packet data is ready for transmission to the CPU, and request the host initiates a DMA read. Only used for DMA transfers, not for normal register access.
$\overline{\text{CPU_IREQO}}$	O	AF17	CPU Interrupt 0 Request (Active Low)
$\overline{\text{CPU_IREQ1}}$	O	AD16	CPU Interrupt 1 Request (Active Low)

Table 15 - CPU Interface Package Ball Definition

4.6 System Function Interface

All System Function Interface signals are 5V tolerant.

The core of the chip will be held in reset for 16383 SYSTEM_CLK cycles after SYSTEM_RST has gone HIGH to allow the PLL's to lock.

Signal	I/O	Package Balls	Description
SYSTEM_CLK	I	U6	System Clock Input. The system clock frequency is 100MHz. The frequency must be accurate to within ±32ppm in synchronous mode.
SYSTEM_RST	I	V4	System Reset Input. Active low. The system reset is asynchronous, and causes all registers within the ZL50110/1/4 to be reset to their default state.
SYSTEM_DEBUG	I	U5	System Debug Enable. This is an asynchronous signal that, when de-asserted, prevents the software assertion of the debug-freeze command, regardless of the internal state of registers, or any error conditions. Active high.

Table 16 - System Function Interface Package Ball Definition

4.7 Test Facilities

4.7.1 Administration, Control and Test Interface

All Administration, Control and Test Interface signals are 5V tolerant.

Signal	I/O	Package Balls	Description
GPIO[15:0]	ID/ OT	[15] AA4 [14] AB3 [13] AC2 [12] AC1 [11] AB2 [10] Y4 [9] W5 [8] AA3 [7] AA2 [6] Y3 [5] AB1 [4] Y2 [3] W4 [2] V5 [1] AA1 [0] W3	General Purpose I/O pins. Connected to an internal register, so customer can set user-defined parameters. Bits [4:0] reserved at startup or reset for memory TDL setup. See the ZL50110/1/4 Programmers Model for more details.
TEST_MODE[2:0]	I D	[2] AF6 [1] AB9 [0] AC8	Test Mode input - ensure these pins are tied to ground for normal operation. 000 SYS_NORMAL_MODE 001-010 RESERVED 011 SYS_TRISTATE_MODE 100-111 RESERVED

Table 17 - Administration/Control Interface Package Ball Definition

4.7.2 JTAG Interface

All JTAG Interface signals are 5V tolerant, and conform to the requirements of IEEE1149.1 (2001).

Signal	I/O	Package Balls	Description
$\overline{\text{JTAG_TRST}}$	I U	AE7	JTAG Reset. Asynchronous reset. In normal operation this pin should be pulled low.
JTAG_TCK	I	AD8	JTAG Clock - maximum frequency is 25MHz, typically run at 10MHz. In normal operation this pin should be pulled either high or low.
JTAG_TMS	I U	AA10	JTAG test mode select. Synchronous to JTAG_TCK rising edge. Used by the Test Access Port controller to set certain test modes.
JTAG_TDI	I U	AF7	JTAG test data input. Synchronous to JTAG_TCK.
JTAG_TDO	O	AC9	JTAG test data output. Synchronous to JTAG_TCK.

Table 18 - JTAG Interface Package Ball Definition

4.8 Miscellaneous Inputs

The following unused inputs must be held low or high as appropriate.

Signal	I/O	Package Balls	Description
PULL_LO	I	AF8 AD9	Reserved inputs, must be pulled low.
PULL_HI	I	AF16	Reserved input, must be pulled high.

Table 19 - Miscellaneous Inputs Package Ball Definitions

4.9 Power and Ground Connections

Signal	Package Balls				Description
VDD_IO	J9 J13 J17 L9 N9 R9 U9 V11 V15	J10 J14 J18 L18 N18 R18 U18 V12 V16	J11 J15 K9 M9 P9 T9 V9 V13 V17	J12 J16 K18 M18 P18 T18 V10 V14 V18	3.3V VDD Power Supply for IO Ring
GND	A1 F6 L11 L15 M13 N1 N13 N22 P12 P16 R13 T5 T14 AA6 AF13	A13 F21 L12 L16 M14 N5 N14 N26 P13 P24 R14 T11 T15 AA21 AF26	A26 K5 L13 M11 M15 N11 N15 P6 P14 R11 R15 T12 T16 AB10	E22 K22 L14 M12 M16 N12 N16 P11 P15 R12 R16 T13 T24 AF1	0V Ground Supply
VDD_CORE	F7 F20 K6 N6 Y6 AA13	F12 H6 K21 T21 Y21 AA14	F15 H21 M21 V6 AA9 AA18		1.8V VDD Power Supply for Core Region
A1VDD	T6				1.8V PLL Power Supply

Table 20 - Power and Ground Package Ball Definition

5.0 Basic Operation

A diagram of the ZL50110/1/4 device is given in Figure 12, which shows the major data flows between functional components.

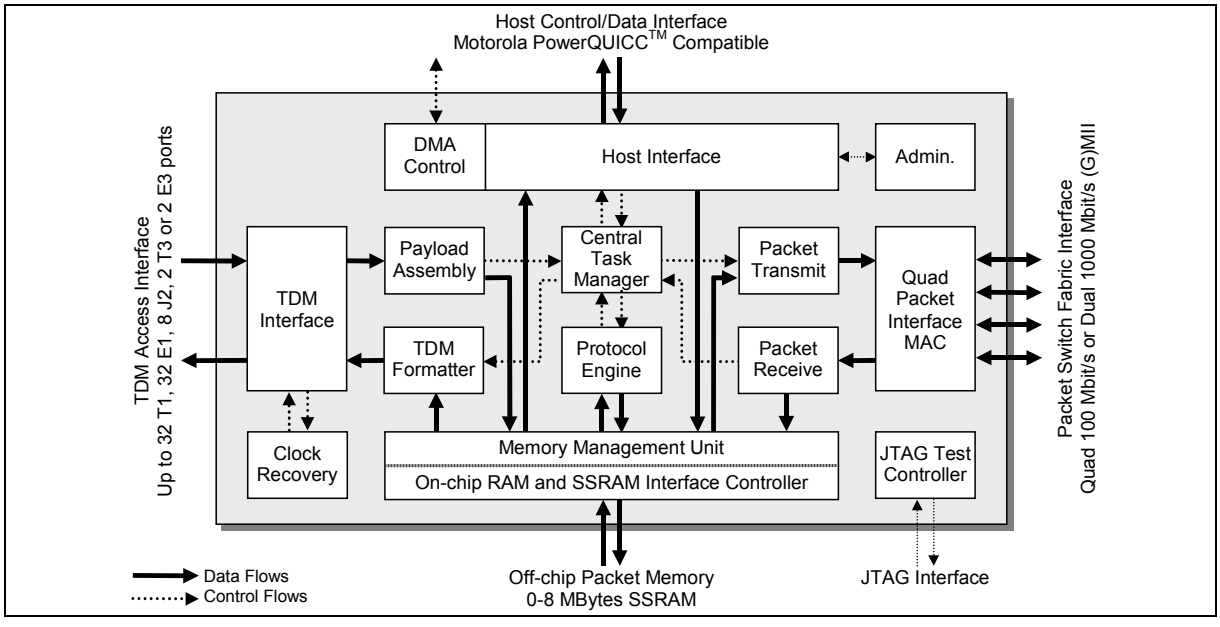


Figure 12 - ZL50110/1/4 Data and Control Flows

5.1 TDM Access Interface

The TDM Access Interface consists of up to 32 streams (depending on variant), each with an input and an output data stream operating at either 1.544 Mbit/s or 2.048 Mbit/s. It contains two basic types of interface: unstructured clock and data, for interfacing directly to a line interface unit; or structured, framed data, for interfacing to a framer or TDM backplane.

Unstructured data is treated asynchronously, with every stream using its own clock. Clock recovery is provided on each output stream, to reproduce the TDM service frequency at the egress of the packet network. Structured data is treated synchronously, i.e. all data streams are timed by the same clock and frame references. These can either be supplied from an external source (slave mode) or generated internally using the on-chip stratum 3/4/4E PLL (master mode).

5.2 TDM Payload Assembly

Data traffic received on the TDM Access Interface is sampled in the TDM Interface block, and synchronized to the internal clock. It is then forwarded to the payload assembly process. The ZL50110/1/4 Payload Assembler can handle up to 128 active packet streams or “contexts” simultaneously. Packet payloads are assembled in the format shown in Figure 17 on page 60. This meets the requirements of the CESoPSN standard under development in the IETF.

When the payload has been assembled it is written into the centrally managed memory, and a task message is passed to the Task Manager. This acts as a “router” in the centre of the chip, directing packets to the appropriate blocks for further processing. The task message contains a pointer to the relevant data, instructions as to what to do with the data, and ancillary information about the packet. Effectively this means the flow of data through the device can be programmed, by setting the task message contents appropriately.

5.3 Higher Layer Protocol Support

In general, the next processing block for TDM packets is the Protocol Engine. This handles the data-plane requirements of the main higher level protocols (layers 4 and 5) expected to be used in typical applications of the ZL50110/1/4 family: UDP, RTP, L2TP, CESoPSN and CDP. The Protocol Engine can add a header to the datagram containing up to 24 bytes. This header is largely static information, and is programmed directly by the CPU. It may contain a number of dynamic fields, including a length field, checksum, sequence number and a timestamp. The location, and in some cases the length of these fields is also programmable, allowing the various protocols to be placed at variable locations within the header.

5.4 Packet Transmission

Packets ready for transmission are queued to the switch fabric interface by the Queue Manager. Four classes of service are provided, allowing some packet streams to be prioritized over others. On transmission, the Packet Transmit block appends a programmable header, which has been set up in advance by the control processor. Typically this contains the data-link and network layer headers (layers 2 and 3), such as Ethernet, IP (versions 4 and 6) and MPLS.

5.5 Host Packet Generation

The control processor can generate packets directly, allowing it to use the network for out-of-band communications. This can be used for transmission of control data or network setup information, e.g. routing information. The host interface can also be used by a local resource for network transmission of processed data.

The device supports dual address DMA transfers of packets to and from the CPU memory, using the host's own DMA controller. Table 21 illustrates the maximum bandwidths achievable by an external DMA master.

DMA Path	Packet Size	Max Bandwidth Mbps[1]
ZL50110/1/4 to CPU only	>1000 bytes	50
ZL50110/1/4 to CPU only	60 bytes	6.7
CPU to ZL50110/1/4 only	>1000 bytes	60
CPU to ZL50110/1/4 only	60 bytes	43
Combined[2]	>1000 bytes	58 (29 each way)
Combined[2]	60 bytes	11 (5.5 each way)

Table 21 - DMA Maximum Bandwidths

Note 1: Maximum bandwidths are the maximum the ZL50110/1/4 devices can transfer under host control, and assumes only minimal packet processing by the host.

Note 2: Combined figures assume the same amount of data is to be transferred each way.

5.6 Packet Reception

Incoming data traffic on the packet interface is received by the MACs. The well-formed packets are forwarded to a packet classifier to determine the destination. When a packet is successfully classified the destination can be the TDM interface, the LAN interface or the host interface. TDM traffic is then further classified to determine the context it is intended for.

Each TDM interface context has an individual queue, and the TDM re-formatting process re-creates the TDM streams from the incoming packet streams. This queue is used as a jitter buffer, to absorb variation in packet delay across the network. The size of the jitter buffer can be programmed in units of TDM frames (i.e. steps of 125 μs).

There is also a queue to the host interface, allowing a traffic flow to the host CPU for processing. Again the host's DMA controller can be used to retrieve packet data and write it out into the CPU's own memory.

5.7 TDM Re-Formatting

At the receiving end of the packet network, the original TDM data must be re-constructed from the packets received. This is known as re-formatting, and follows the reverse process from the Payload Assembler. The TDM Formatter plays out the packets in the correct sequence, directing each octet to the selected timeslot on the output TDM interface.

When lost or late packets are detected, the TDM Formatter plays out underrun data for the same number of TDM frames as were included in the missing packet. Underrun data can either be the last value played out on that timeslot, or a pre-programmed value (e.g. 0xFF). If the packet subsequently turns up it is discarded. In this way, the end-to-end latency through the system is maintained at a constant value.

5.8 Data and Control Flows

There are numerous combinations that can be implemented to pass data through the ZL50110/1/4 device depending on the application requirements. The Task Manager can be considered the central pivot, through which all flows must operate.

The flow is determined by the Type field in the Task Message (see ZL50110/1/4 Programmers Model).

Flow Number	Flow Through Device
1	TDM to (TM) to PE to (TM) to PKT
2	PKT to (TM) to PE to (TM) to TDM
3	TDM to (TM) to PKT
4	PKT to (TM) to TDM
5	TDM to (TM) to CPU
6	TDM to (TM) to PE to (TM) to CPU
7	CPU to (TM) to TDM
8	PKT to (TM) to CPU
9	CPU to (TM) to PKT
10[1]	TDM to (TM) to TDM
11[1]	PKT to (TM) to PKT

Table 22 - Standard Device Flows

1. This flow is for loopback test purposes only

Each of the 11 data flows uses the Task Manager to route packet information to the next block or interface for onward transmission. This section describes the flows between the TDM interface, the packet interface and the Task Manager which are the main flow routes used in the ZL50110/1/4 family. For example, the TDM->TM flow is used in flow types 1, 3, 5, and 6, and the TM->PKT flow is used in flow types 1, 3, and 9.

Figure 12 can be used as a reference for each flow described above. Figure 13 to Figure 16 describe TDM and Packet flows in detail and illustrate data transmission through the ZL50110/1/4 device.

5.8.1 TM to TDM Flow

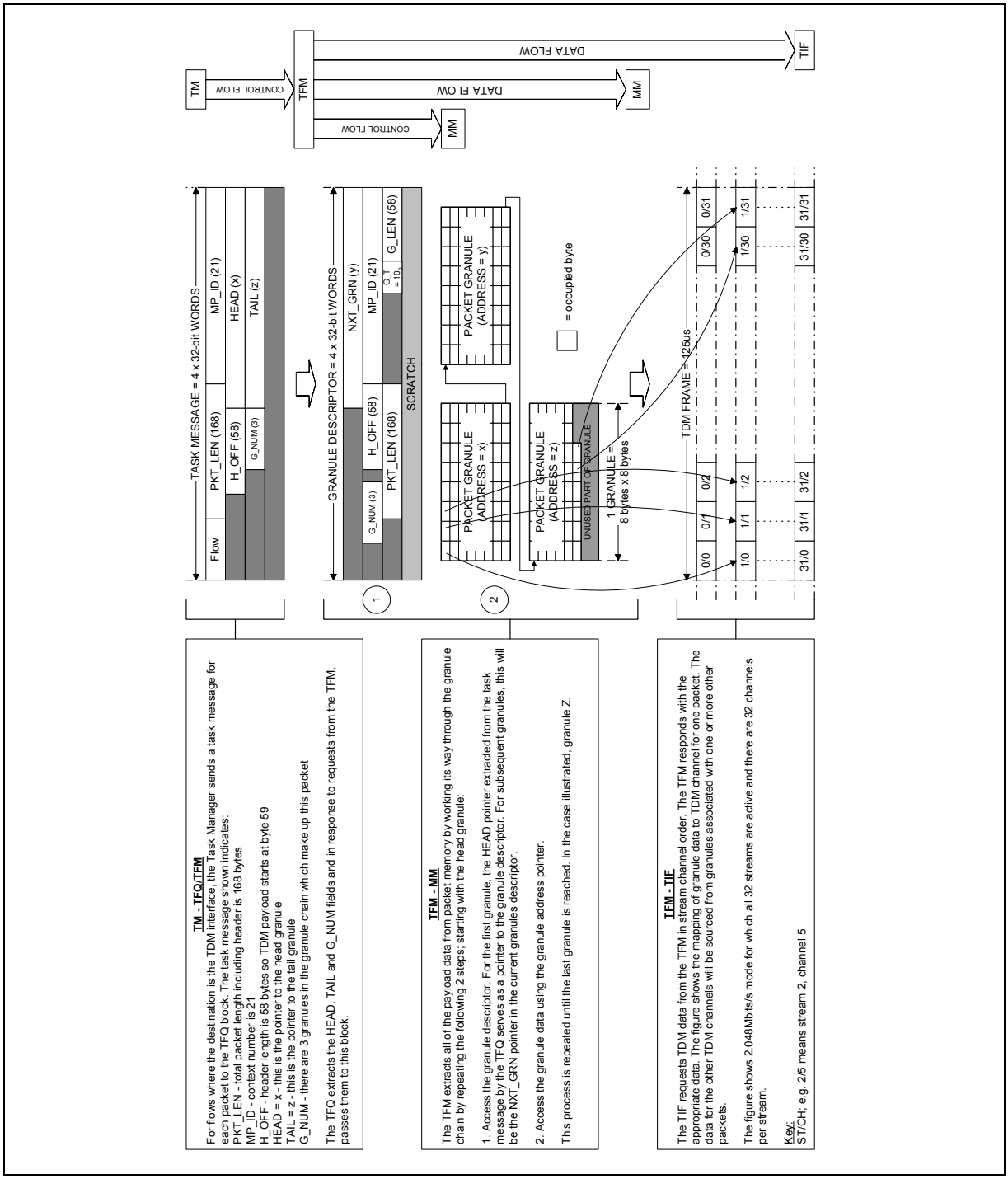


Figure 13 - TDM to TM Flow Diagram

5.8.2 TDM to TM Flow

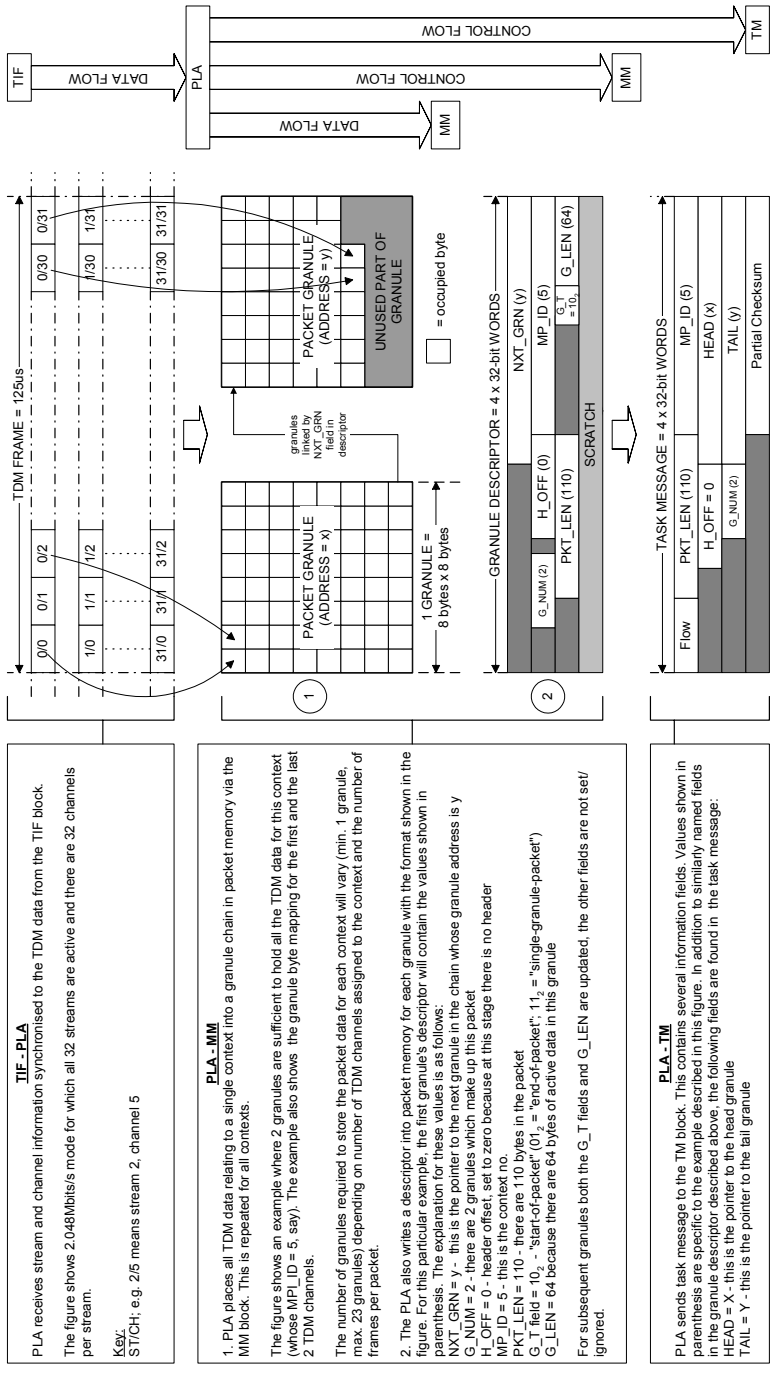


Figure 14 - TDM to TM Flow Diagram

5.8.3 PKT to TM Flow

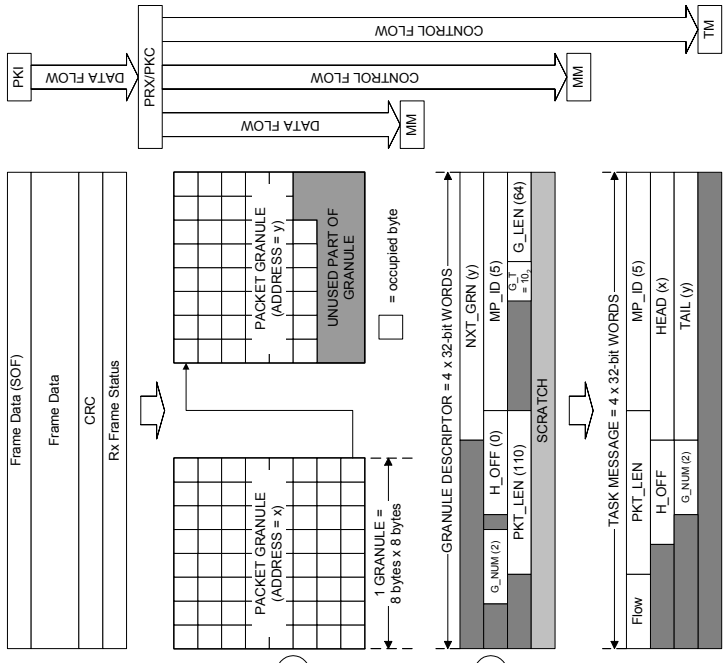


Figure 15 - PKT to TM Flow

5.8.4 TM to PKT Flow

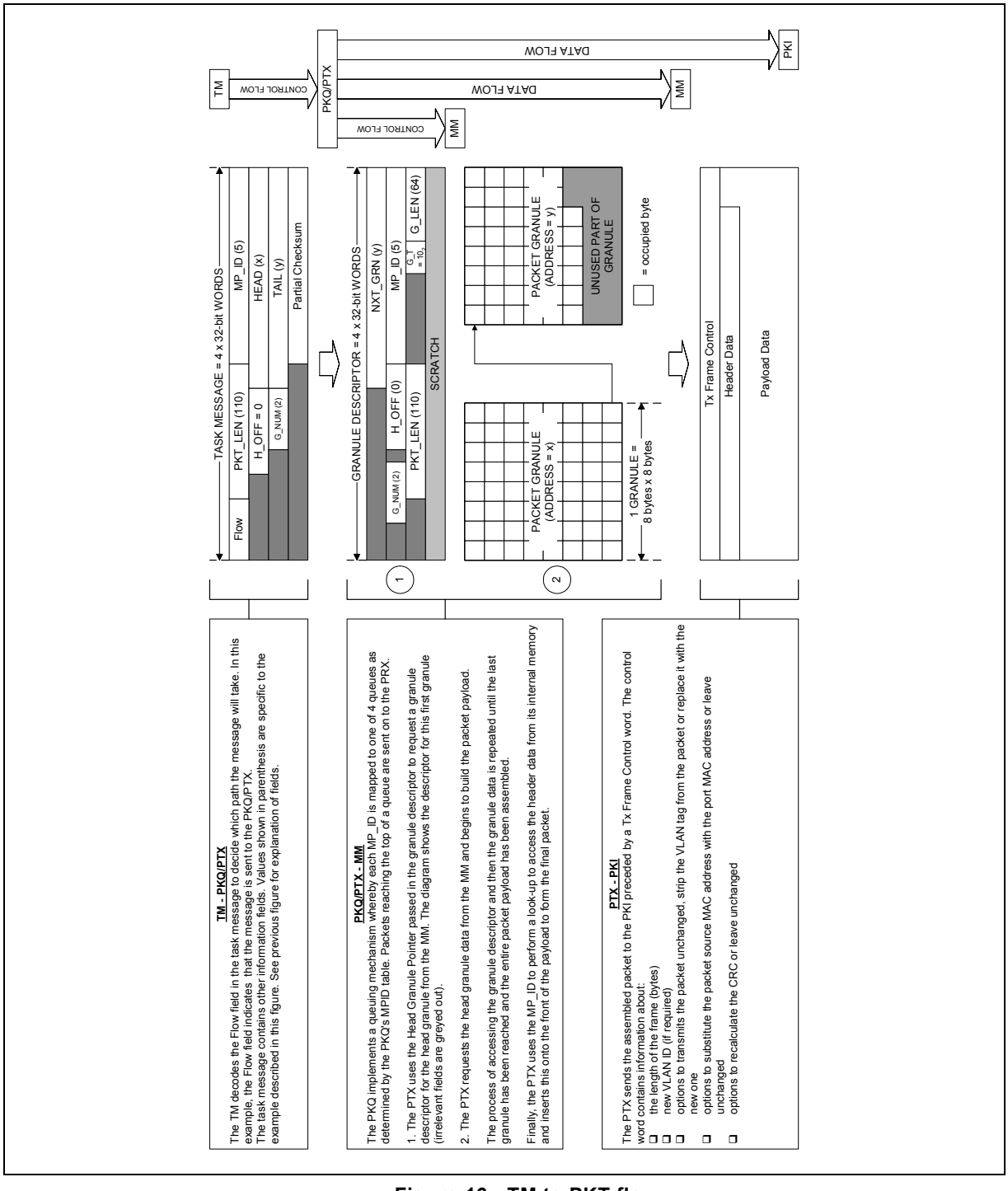


Figure 16 - TM to PKT flow

5.9 Assembling a Packet

The incoming TDM data streams are assembled into packet payloads by the Payload Assembly block. This can handle up to 128 active contexts at a time, where each context represents a "virtual channel connection" in CES terms. Each context generates a single stream of packets identified by a label in the packet header known as the "context ID".

5.9.1 Structured Operation

In structured mode a context may contain any number of 64 kbit/s channels. These channels need not be contiguous but must be from a single input stream.

Channels may be added or deleted dynamically from a context. This feature can be used to optimise bandwidth utilisation. Modifications to the context are synchronised with the start of a new packet.

The fixed header at the start of each packet is added by the Packet Transmit block. This consists of up to 64 bytes, containing the Ethernet header, any upper layer protocol headers, and the two byte context descriptor field (see section below). The header is entirely user programmable, enabling the use of any protocol.

The payload header and size must be chosen so that the overall packet size is not less than 64 bytes, the Ethernet standard minimum packet size. Where this is likely to be the case, the header or data must be padded (as shown in Figure 17 and Figure 18) to ensure the packet is large enough. This padding is added by the ZL50110/1/4 for most applications. If interfacing to the MT9088x family, padding must be added using the same method as described in the MT9088x documentation.

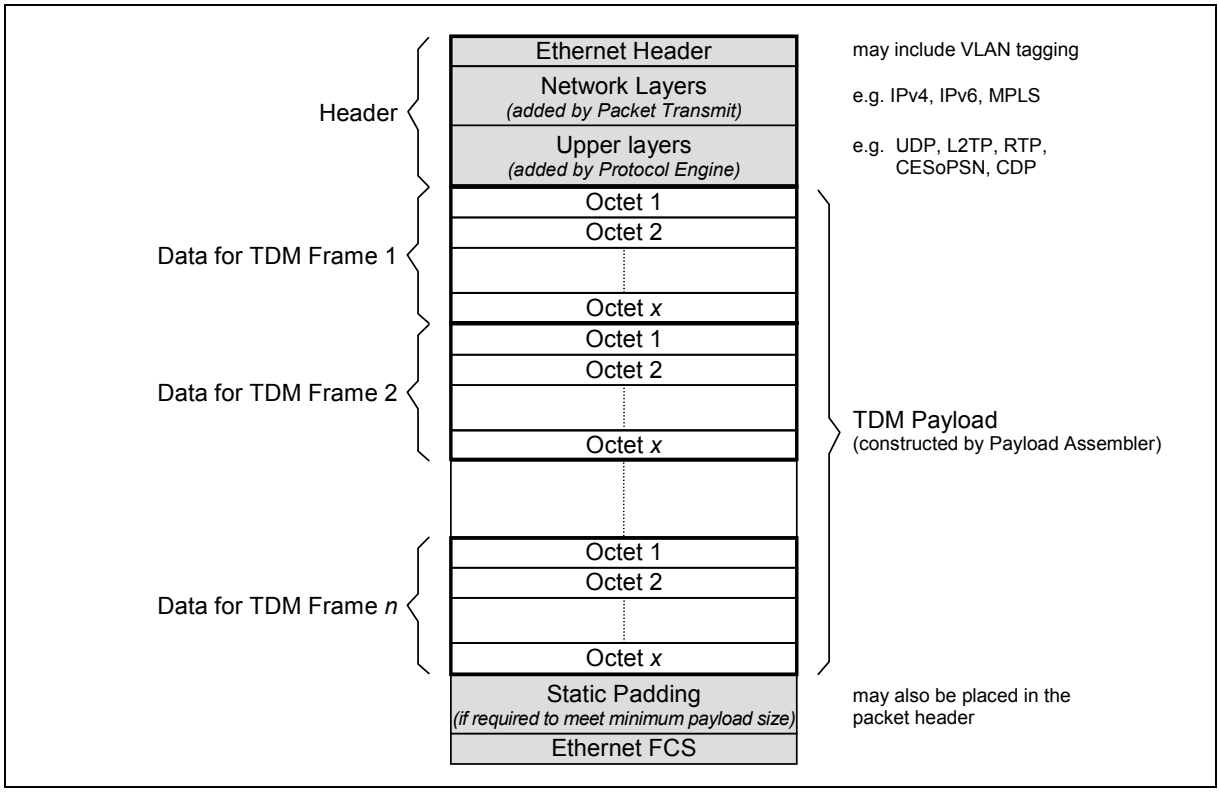


Figure 17 - ZL50110/1/4 Packet Format - Structured Mode

In applications where large payloads are being used, the payload size must be chosen such that the overall packet size does not exceed the maximum Ethernet packet size of 1518 bytes (1522 bytes with VLAN tags). Figure 17 shows the packet format for structured TDM data, where the payload is split into frames, and each frame concatenated to form the packet.

5.9.2 Unstructured Operation

In unstructured mode, the payload is not split by defined frames or timeslots, so the packet consists of a continuous stream of data. Each packet consists of a number of octets, as shown in Figure 18. For example, consider mapping the unstructured data of a 25 timeslot DS0 stream. The data for each T1 frame would normally consist of 193 bits, 192 data bits and 1 framing bit. If the payload consists of 24 octets it will be 1 bit short of a complete frames worth of data, if the payload consists of 25 octets it will be 7 bits over a complete frames worth of data. **NOTE:** No alignment of the octets with the T1 framing structure can be assumed.

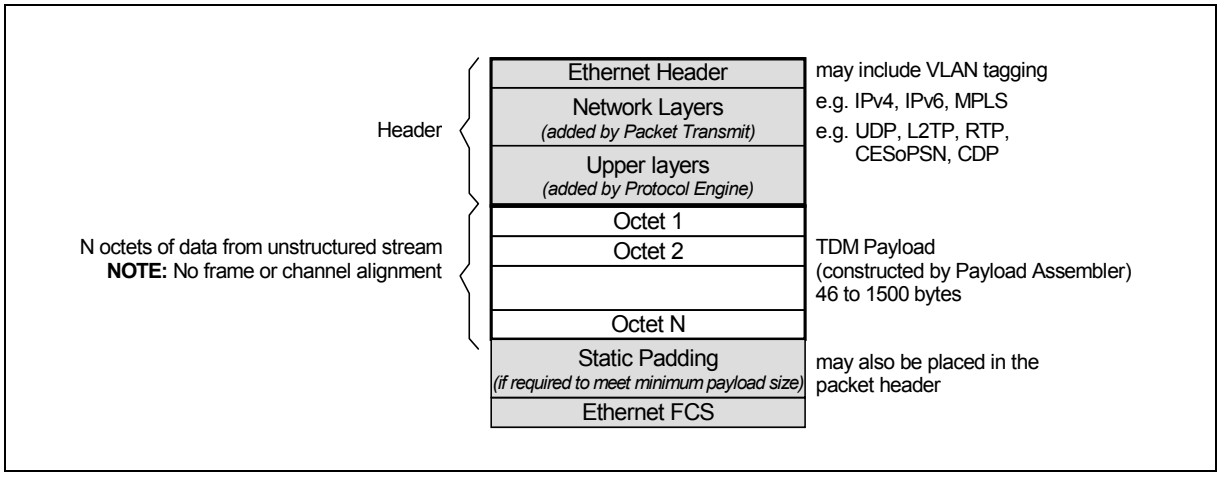


Figure 18 - ZL50110/1/4 Packet Format - Unstructured Mode

5.10 TDM Port Data Formats

The ZL50110/1/4 is programmable such that the frame/clock polarity and clock alignment can be set to any desired combination. Table 23 shows a brief summary of four different TDM formats; ST-BUS, H.110, H-MVIP, and Generic (synchronous mode only), for more information see the relevant specifications shown. There are many additional formats for TDM transmission not depicted in Table 23, but the flexibility of the port will cover almost any scenario. The overall data format is set for the entire TDM Interface device, rather than on a per stream basis. It is possible to control the polarity of the master clock and frame pulse outputs, independent of the chosen data format (used when operating in synchronous master mode).

Data Format	Data Rate (Mbit/s)	Number of channels per frame	Clock Freq. (MHz)	Nominal Frame Pulse Width (ns)	Frame Pulse Polarity	Frame Boundary Alignment		Standard
						clock	frame pulse	
ST-bus	2.048	32	2.048	244	Negative	Rising Edge	Straddles boundary	MSAN-126 Rev B (Issue 4) Zarlink
	2.048	32	4.096	244	Negative	Falling Edge	Straddles boundary	
	8.192	128	16.384	61	Negative	Falling Edge	Straddles boundary	
H.110	8.192	128	8.192	122	Negative	Rising edge	Straddles boundary	ECTF H.110
H-MVIP	2.048	32	2.048	244	Negative	Rising Edge	Straddles boundary	H-MVIP Release 1.1a
	2.048	32	4.096	244	Negative	Falling Edge	Straddles boundary	
	8.192	128	16.384	244	Negative	Falling Edge	Straddles boundary	
Generic	2.048	32	2.048	488	Positive	Rising Edge	Rising edge of clock	
	8.192	128	8.192	122	Positive	Rising Edge	Rising edge of clock	

Table 23 - Some of the TDM Port Formats accepted by the ZL50110/1/4 Family

5.11 External Memory Requirement

The ZL50110/1/4 family includes a large amount of on-chip memory, such that for most applications, external memory will not be required. However, for certain combinations of header size, packet size and jitter buffer size, there may be a requirement for external memory. Therefore the device allows the connection of up to 8Mbytes of synchronous ZBT-SRAM.

The following charts show how much memory is required by the ZL50111 (32 T1 streams) and the ZL50110 (8 T1 streams) for a variety of packet sizes (expressed in number of frames of TDM data) and jitter buffer sizes. It is assumed that each packet contains a full Ethernet/MPLS/MPLS/RTP/CESoPSN header.

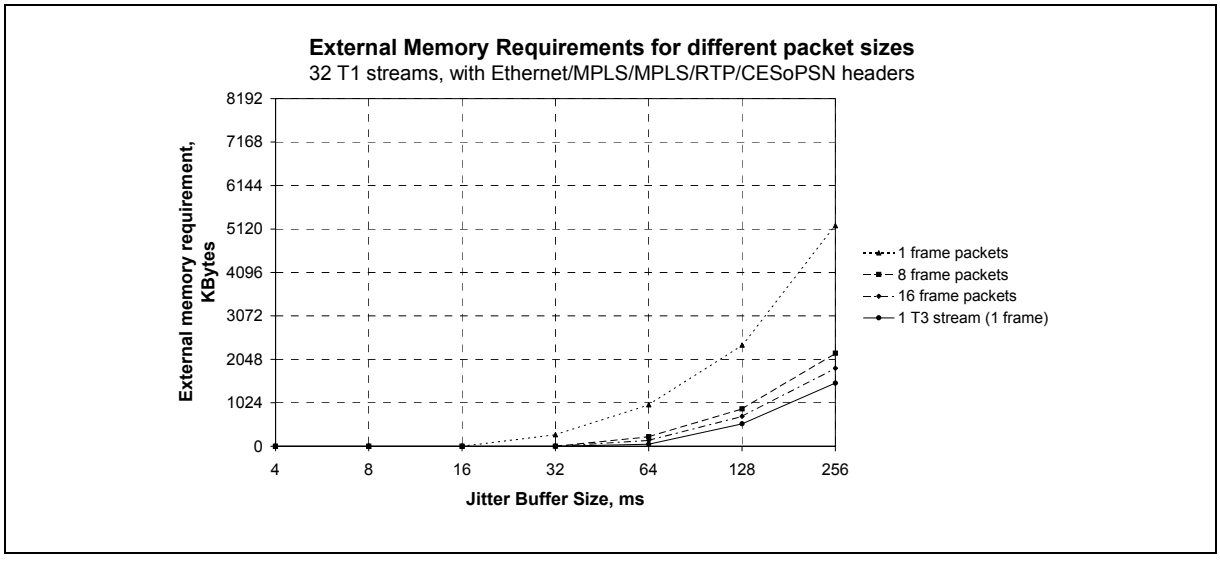


Figure 19 - External Memory Requirement for ZL50111

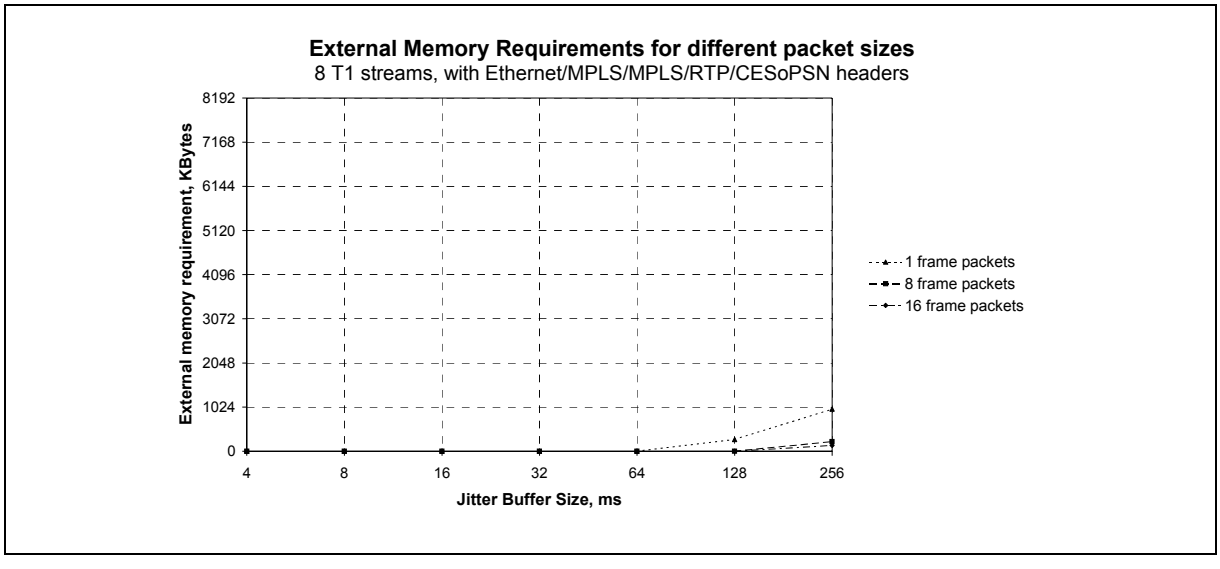


Figure 20 - External Memory Requirement for ZL50110

5.12 TDM Clock Structure

The TDM interface can operate in two modes, synchronous for structured TDM data, and asynchronous for unstructured TDM data. The ZL50110/1/4 is capable of providing the TDM clock for either of the modes, but clock recovery is only possible in asynchronous mode, where the timing for each stream is controlled independently.

5.12.1 Synchronous TDM Clock Generation

In synchronous mode all 32 streams will be driven by a common clock source. When the ZL50110/1/4 is acting as a master device, the source can either be the internal DPLL or an external PLL. In both cases, the primary and secondary reference clocks are taken from either two TDM input clocks, or two external clock sources driven to the chip. The input clocks are then divided down where necessary and sent either to the internal DPLL or to the output pins for connection to an external DPLL. The DPLL then provides the common clock and frame pulse required to drive the TDM streams. See “DPLL Specification” on page 68 for further details.

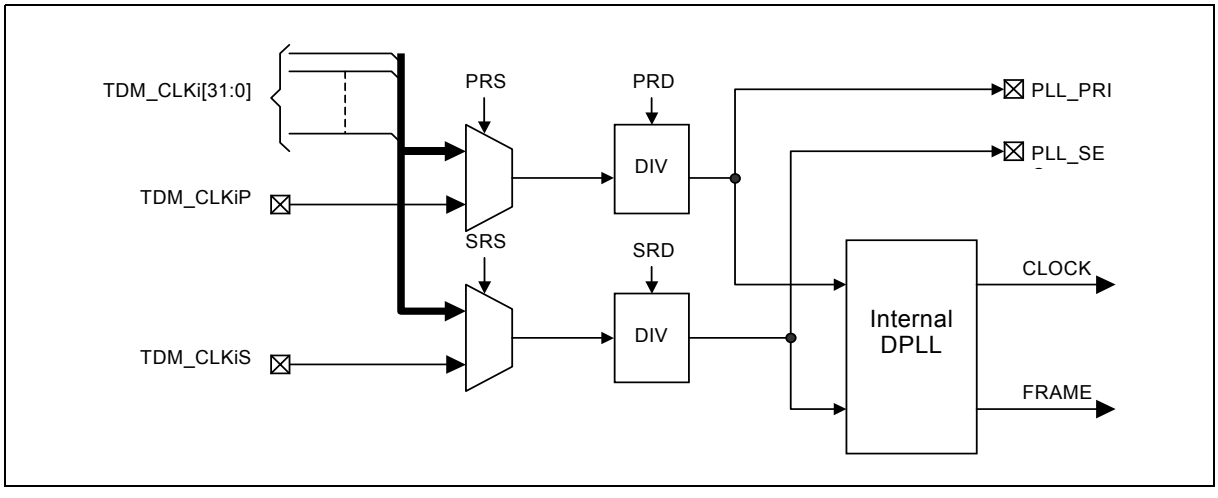


Figure 21 - Synchronous TDM Clock Generation

When the ZL50110/1/4 is acting as a slave device, the common clock and frame pulse signals are taken from an external device providing the TDM master function.

5.12.2 Asynchronous TDM Clock Generation

Each stream uses a separate internal DCO to provide an asynchronous TDM clock output. The DCO can be controlled to recover the clock from the original TDM source depending on the timing algorithm used.

There are two algorithms provided to assist clock recovery within the ZL50110/1/4; differential, and adaptive. The clock recovery itself is performed by software in the external processor, with support from on-chip hardware to gather the required statistics.

The Differential Clock recovery method is used with a common reference clock at each end of the packet network. The relationship between the TDM service clock and the common reference clock at the source node is used to control the DCO at the destination node. This ensures that the TDM data is re-generated at the destination with the same timing as at the source node. Figure 22 illustrates this concept with a common Primary Reference Source (PRS) clock being present at both the source and destination equipment.

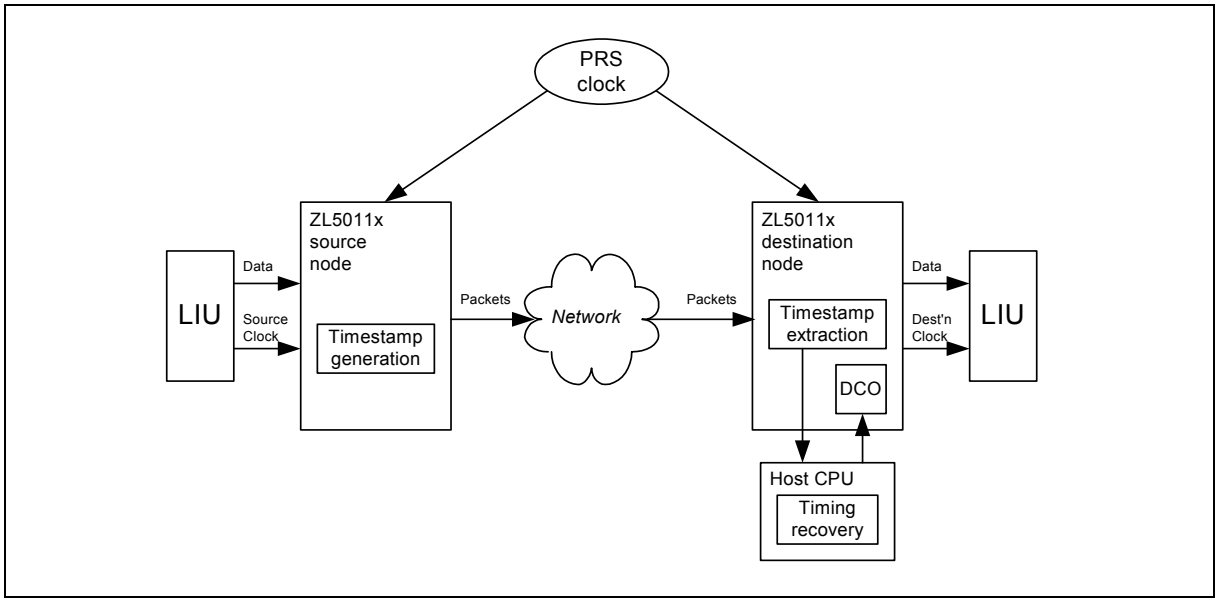


Figure 22 - Differential Clock Recovery

With no PRS clock the only information available to determine the TDM transmission speed is the average arrival rate of the packets, as shown in Figure 23. This is known as Adaptive Clock recovery. Timestamps representing the number of elapsed source clock periods may be included in the packet header, or information can be inferred from a known payload size at the destination. It is possible to maintain average buffer-fill levels at the destination, where an increase or decrease in the fill level of the buffer would require a change in transmission clock speed to maintain the average. Additionally, the buffer-fill depth can be altered independently, with no relation to the recovered clock frequency, to control TDM transmission latency.

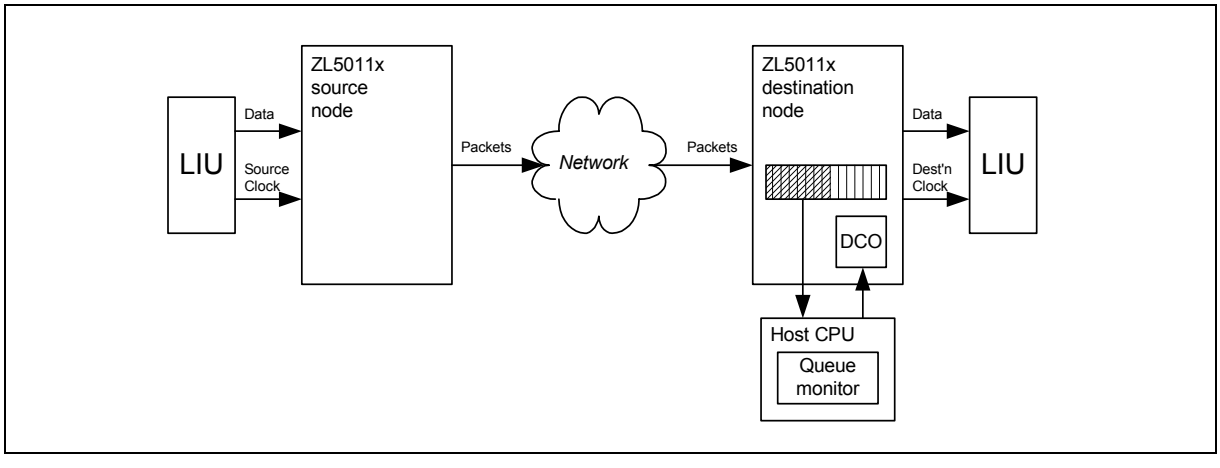


Figure 23 - Adaptive Clock Recovery

5.13 GIGABIT Ethernet - Recommended Configurations.

NOTE: In GMII/PCS mode only 1 GMAC port may be used. The second GMAC port is for redundancy purposes only.

This section outlines connection methods for the ZL50110/1/4 in a Gigabit Ethernet environment recommended to ensure optimum performance. Two areas are covered

- Central Ethernet Switch
- Redundant Ethernet Switch

5.13.1 Central Ethernet Switch

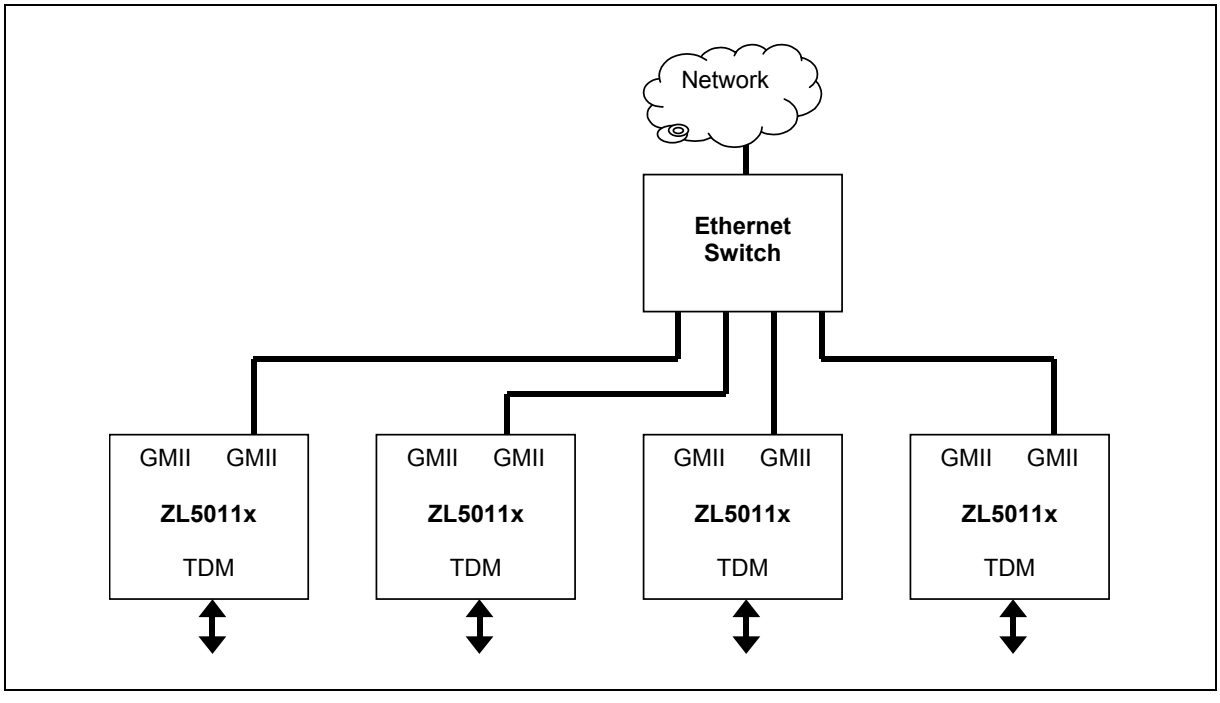


Figure 24 - Gigabit Ethernet Connection - Central Ethernet Switch

TDM data and control packets are directed to the appropriate ZL50110/1/4 device through the Ethernet Switch. There is no limit on the number of ZL50110/1/4 devices that can be connected in this configuration.

5.13.2 Redundant Ethernet Switch

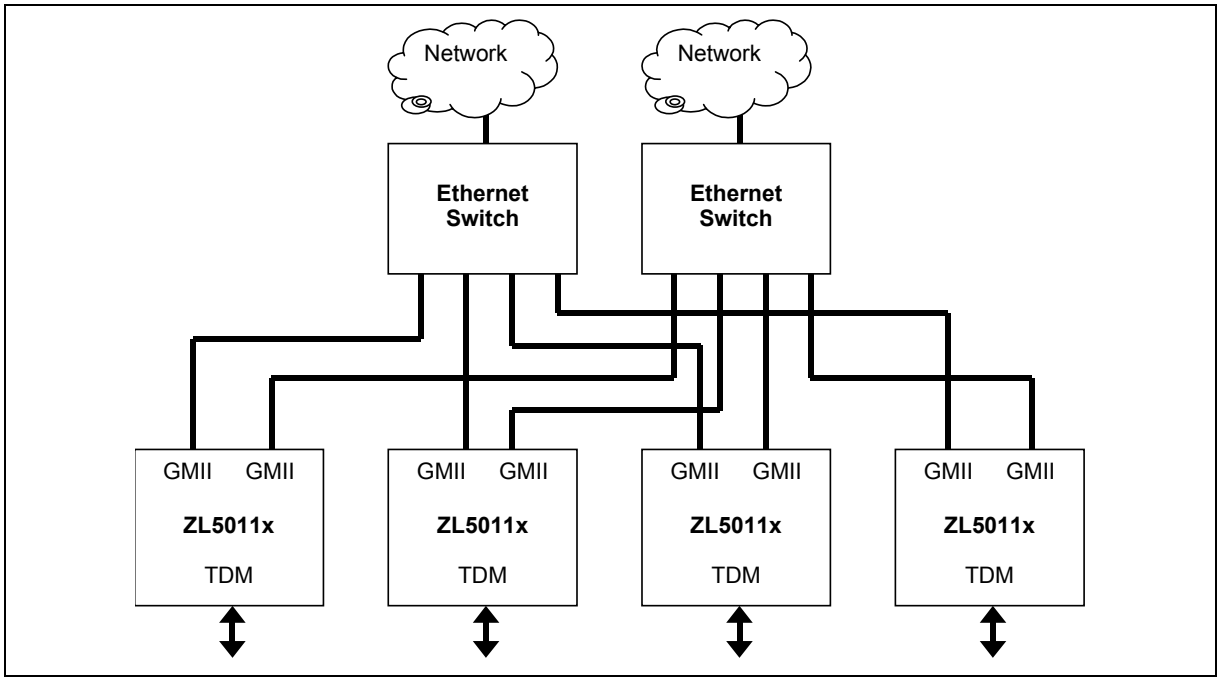


Figure 25 - Gigabit Ethernet Connection - Redundant Ethernet Switch

The central Ethernet Switch configuration can be extended to include a redundant switch connected to the second ZL50110/1/4 GMII port. One port should be used for all the TDM-to-Packet and Packet-to-TDM data with the other port idle. If the current port fails then data must be transferred to the spare port.

5.14 Loss of Service (LOS)

During normal transmission a situation may arise where a Loss of Service occurs, caused by a disruption in the transmission line due to engineering works or cable disconnection for example. This results in the loss of a TDM signal, including the associated TDM clock, from the LIU.

With no TDM signal or clock, no packets can be assembled by the transmitting ZL50110/1/4 device, and the flow of packets will cease. The absence of packets at the receiving ZL50110/1/4 device will cause underrun data to be generated at the TDM output, normally an “all-ones” pattern, with the exception of DS3 which alternates ones and zeros. The LOS condition is detected by the receive ZL50110/1/4 device.

Additionally, when the LIU detects LOS, it can notify the CPU. The CPU can set a control bit in the packet header (bit A in the Vainshtein draft), which is then transmitted. The receiving ZL50110/1/4 device recognises the control bit, and transmits an AIS (all-ones) pattern on the appropriate TDM stream.

Using both mechanisms provides a robust method of indicating an LOS condition to the downstream TDM equipment.

6.0 Power Up sequence

To power up the ZL50110/1/4 the following procedure must be used:

- The Core supply must never exceed the I/O supply by more than $0.5V_{DC}$.
- Both the Core supply and the I/O supply must be brought up together
- The System Reset and, if used, the JTAG Reset must remain low until at least 100µs after the 100MHz system clock has stabilised. Note that if JTAG Reset is not used it must be tied low.

This is illustrated in the diagram shown in Figure 26.

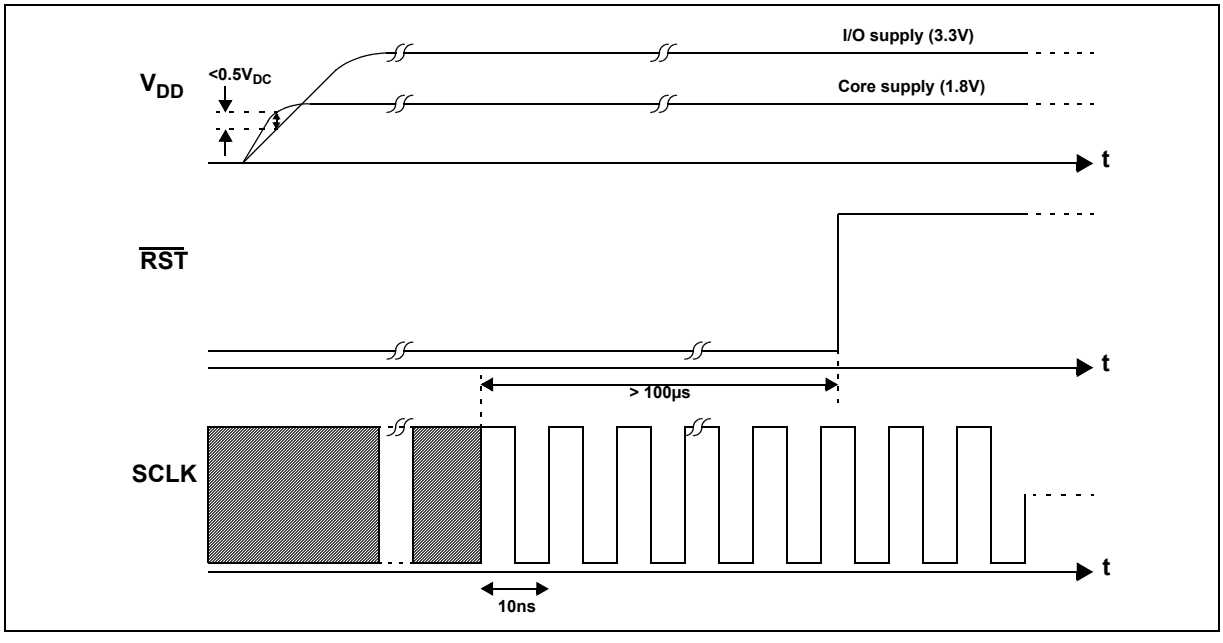


Figure 26 - Powering Up the ZL50110/1/4

7.0 DPLL Specification

The ZL50110/1/4 family incorporates an internal DPLL that meets Telcordia GR-1244-CORE Stratum 3 and Stratum 4/4E requirements, assuming an appropriate clock oscillator is connected to the system clock pin. It will meet the jitter/wander tolerance, jitter/wander transfer, intrinsic jitter/wander, frequency accuracy, capture range, phase change slope, holdover frequency and MTIE requirements for these specifications. In structured mode with the ZL50110/1/4 device operating as a master the DPLL is used to provide clock and frame reference signals to the internal and external TDM infrastructure. In structured mode, with the ZL50110/1/4 device operating as a slave, the DPLL is not used. All TDM clock generation is performed externally and the input streams are synchronised to the system clock by the TDM interface. The DPLL is not required in unstructured mode, where TDM clock and frame signals are generated by internal DCO's assigned to each individual stream.

7.1 Modes of operation

It can be set into one of four operating modes: Locking mode, Holdover mode, Freerun mode and Powerdown mode.

7.1.1 Locking Mode (normal operation)

The DPLL accepts a reference signal from either a primary or secondary source, providing redundancy in the event of a failure. These references should have the same nominal frequencies but do not need to be identical as long as their frequency offsets meet the appropriate Stratum requirements. Each source is selected from any one of the available TDM input stream clocks (up to 32 on the ZL50111 variant), or from the external TDM_CLKiP (primary) or TDM_CLKiS (secondary) input pins, as illustrated in Figure 21 - on page 64. It is possible to supply a range of input frequencies as the DPLL reference source, depicted in Table 24. The PRD register Value is the number (in hexadecimal) that must be programmed into the PRD register within the DPLL to obtain the divided down frequency at PLL_PRI or PLL_SEC.

Source Input Frequency (MHz)	Tolerance (±ppm)	Divider Ratio	PRD/SRD Register Value (Hex) (Note 1)	Frequency at PLL_PRI or PLL_SEC (MHz)	Maximum Acceptable Input Wander tolerance (UI) (Note 2)
0.008	30	1	1	0.008	±1
1.544	130	1	1	1.544	±1023
2.048	50	1	1	2.048	±1023
4.096	50	1	1	4.096	±1023
8.192	50	1	1	8.192	±1023
16.384	50	1	1	16.384	±1023
6.312	30	1	1	6.312	±1023
22.368	20	2796	AEC	0.008	±1 (on 64kHz)
34.368	20	537	219	0.064	±1 (on 64kHz)
44.736 (Note 3)	20	699	2BB	0.064	±1 (on 64kHz)

Table 24 - DPLL Input Reference Frequencies

Note 1: A PRD/SRD value of 0 will suppress the clock, and prevent it from reaching the DPLL.

Note 2: UI means Unit Interval - in this case periods of the time signal. So ±1UI on a 64kHz signal means ±15.625µs, the period of the reference frequency. Similarly ±1023UI on a 4.096MHz signal means ±250µs.

Note 3: This input frequency is supported with the use of an external divide by 2.

The maximum lock-in range can be programmed up to ±372ppm regardless of the input frequency. The DPLL will fail to lock if the source input frequency is absent, if it is not of approximately the correct frequency or if it is too jittery. See Section 7.7 for further details. Limitations depend on the users programmed values, so the DPLL must be programmed properly to meet Stratum 3, or Stratum 4/4E. The Application Program Interface (API) software that accompanies the ZL50110/1/4 family can be used to automatically set up the DPLL for the appropriate standard requirement.

The DPLL lock-in range can be programmed using the Lock Range register (see ZL50110/1/4 Programmers Model document) in order to extend or reduce the capture envelope. The DPLL provides bit-error-free reference switching, meeting the specification limits in the Telcordia GR-1244-CORE standard. If Stratum 3 or Stratum 4/4E accuracy is not required, it is possible to use a more relaxed system clock tolerance.

The DPLL output consists of three signals; a common clock (comclk), a double-rate common clock (comclkx2), and a frame reference (8kHz). These are used to time the internal TDM Interface, and hence the corresponding TDM infrastructure attached to the interface. The output clock options are either 2.048Mbit/s (comclkx2 at 4.096Mbit/s) or 8.192Mbit/s (comclkx2 at 16.384Mbit/s), determined by setup in the DPLL control register. The frame pulse is programmable for polarity and width.

7.1.2 Holdover Mode

In the event of a reference failure resulting in an absence of both the primary and secondary source, the DPLL automatically reverts to Holdover mode. The last valid frequency value recorded before failure can be maintained within the Stratum 3 limits of ±0.05ppm. The hold value is wholly dependent on the drift and temperature performance of the system clock. For example, a ±32ppm oscillator may have a temperature coefficient of ±0.1ppm/°C. Thus a 10°C ambient change since the DPLL was last in the Locking mode will change the holdover frequency by an additional ±1ppm, which is much greater than the ±0.05ppm Stratum 3 specification. If the strict target of Stratum 3 is not required, a less restrictive oscillator can be used for the system clock.

Holdover mode is typically used for a short period of time until network synchronisation is re-established.

7.1.3 Freerun Mode

In freerun mode the DPLL is programmed with a centre frequency, and can output that frequency within the Stratum 3 limits of ±4.6ppm. To achieve this the 100MHz system clock must have an absolute frequency accuracy of ±4.6ppm. The centre frequency is programmed as a fraction of the system clock frequency.

7.1.4 Powerdown Mode

It is possible to “power down” the DPLL when it is not in use. For example, an unstructured TDM system, or use of an external DPLL would mean the internal DPLL could be switched off, saving power. The internal registers can still be accessed while the DPLL is powered down.

7.2 Reference Monitor Circuit

There are two identical reference monitor circuits, one for the primary and one for the secondary source. Each circuit will continually monitor its reference, and report the references validity. The validity criteria depends on the frequency programmed for the reference. A reference must meet all the following criteria to maintain validity:

- The “period in specified range” check is performed regardless of the programmed frequency. Each period must be within a range, which is programmable for the application. Refer to the ZL50110/1/4 programmers model for details.
- If the programmed frequency is 1.544MHz or 2.048MHz, the “n periods in specified range” check will be performed. The time taken for n cycles must be within a programmed range, typically with n at 64, the time taken for consecutive cycles must be between 62 and 66 periods of the programmed frequency.

The fail flags are independent of the preferred option for primary or secondary operation, will be asserted in the event of an invalid signal regardless of mode.

7.3 Locking Mode Reference Switching

When the reference source the DPLL is currently locking to becomes invalid, the DPLL’s response depends on which one of the failure detect modes has been chosen: autodetect, forced primary, or forced secondary. One of these failure detect modes must be chosen via the FDM1:0 bits of the DOM register. After a device reset via the SYSTEM_RESET pin, the autodetect mode is selected.

In autodetect mode (automatic reference switching) if both references are valid the DPLL will synchronise to the preferred reference. If the preferred reference becomes unreliable, the DPLL continues driving its output clock in a stable holdover state until it makes a switch to the backup reference. If the preferred reference recovers, the DPLL makes a switch back to the preferred reference. If necessary, the switch back can be prevented by changing the preferred reference using the REFSEL bit in the DOM register, after the switch to the backup reference has occurred.

If both references are unreliable, the DPLL will drive its output clock using the stable holdover values until one of the references becomes valid.

In forced primary mode, the DPLL will synchronise to the primary reference only. The DPLL will not switch to the secondary reference under any circumstances including the loss of the primary reference. In this condition, the DPLL remains in holdover mode until the primary reference recovers. Similarly in forced secondary mode, the DPLL will synchronise to the secondary reference only, and will not switch to the primary reference. Again, a failure of the secondary reference will cause the DPLL to enter holdover mode, until such time as the secondary reference recovers. The choice of preferred reference has no effect in these modes.

When a conventional PLL is locked to its reference, there is no phase difference between the input reference and the PLL output. For the DPLL, the input references can have any phase relationship between them. During a reference switch, if the DPLL output follows the phase of the new reference, a large phase jump could occur. The phase jump would be transferred to the TDM outputs. The DPLL's MTIE (Maximum Time Interval Error) feature preserves the continuity of the DPLL output so that it appears no reference switch had occurred. The MTIE circuit is not perfect however, and a small Time Interval Error is still incurred per reference switch. To align the DPLL output clock to the nearest edge of the selected input reference, the MTIE reset bit (MRST bit in the DOM register) can be used.

Unlike some designs, switching between references which are at different nominal frequencies do not require intervention such as a system reset.

7.4 Locking Range

The locking range is the input frequency range over which the DPLL must be able to pull into synchronisation and to maintain the synchronisation. The locking range is programmable up to ±372ppm.

Note that the locking range relates to the system clock frequency. If the external oscillator has a tolerance of -100ppm, and the locking range is programmed to ±200ppm, the actual locking range is the programmed value shifted by the system clock tolerance to become -300ppm to +100ppm.

7.5 Locking Time

The Locking Time is the time it takes the synchroniser to phase lock to the input signal. Phase lock occurs when the input and output signals are not changing in phase with respect to each other (not including jitter).

Locking time is very difficult to determine because it is affected by many factors including:

- initial input to output phase difference
- initial input to output frequency difference
- DPLL Loop Filter
- DPLL Limiter (phase slope)

Although a short phase lock time is desirable, it is not always achievable due to other synchroniser requirements. For instance, better jitter transfer performance is obtained with a lower frequency loop filter which increases locking time; and a better (smaller) phase slope performance will increase locking time. Additionally, the locking time is dependent on the p_shift value.

The DPLL Loop Filter and Limiter have been optimised to meet the Telcordia GR-1244-CORE jitter transfer and phase alignment speed requirements. The phase lock time is guaranteed to be no greater than 30 seconds when using the recommended Stratum 3 and Stratum 4/4E register settings.

7.6 Lock Status

The DPLL has a Lock Status Indicator and a corresponding Lock Change Interrupt. The response of the Lock Status Indicator is a function of the programmed Lock Detect Interval (LDI) and Lock Detect Threshold (LDT) values in the dpll_ldetect register. The LDT register can be programmed to set the jitter tolerance level of the Lock Status Indicator. To determine if the DPLL has achieved lock the Lock Status Indicator must be high for a period of at least 30 seconds. When the DPLL loses lock the Lock Status Indicator will go low after LDI x 125us.

7.7 Jitter

The DPLL is designed to withstand, and improve inherent jitter in the TDM clock domain.

7.7.1 Acceptance of input wander

For T1(1.544MHz), E1(2.048MHz) and J2(6.312MHz) input frequencies, the DPLL will accept a wander of up to $\pm 1023UI_{pp}$ at 0.1Hz to conform with the relevant specifications. For the 8kHz (frame rate) and 64kHz (the divided down output for T3/E3) input frequencies, the wander acceptance is limited to ±1 UI (0.1Hz). This principle is illustrated in Table 24.

7.7.2 Intrinsic Jitter

Intrinsic jitter is the jitter produced by a synchronizer and measured at its output. It is measured by applying a jitter free reference signal to the input of the device, and measuring its output jitter. Intrinsic jitter may also be measured when the device is in a non synchronizing mode such as free running or holdover, by measuring the output jitter of the device. Intrinsic jitter is usually measured with various band-limiting filters, depending on the applicable standards.

The intrinsic jitter in the DPLL is reduced to less than 1ns p-p[1] by an internal Tapped Delay Line (TDL). The DPLL can be programmed so that the output clock meets all the Stratum 3 requirements of Telcordia GR-1244-CORE. Stratum 4/4E is also supported.

7.7.3 Jitter Tolerance

Jitter tolerance is a measure of the ability of a PLL to operate properly without cycle slips (i.e. remain in lock and/or regain lock in the presence of large jitter magnitudes at various jitter frequencies) when jitter is applied to its reference. The applied jitter magnitude and the jitter frequency depends on the applicable standards.

The DPLL's jitter tolerance can be programmed to meet Telcordia GR-1244-CORE DS1 reference input jitter tolerance requirements.

7.7.4 Jitter Transfer

Jitter transfer or jitter attenuation refers to the magnitude of jitter at the output of a device for a given amount of jitter at the input of the device. Input jitter is applied at various amplitudes and frequencies, and output jitter is measured with various filters depending on the applicable standards.

Since intrinsic jitter is always present, jitter attenuation will appear to be lower for small input jitter signals than larger ones. Consequently, accurate jitter transfer function measurements are usually made with large input jitter signals (e.g. 75% of the specified maximum jitter tolerance).

The internal DPLL is a first order type 2 component, so a frequency offset doesn't result in a phase offset. Stratum 3 requires a -3dB frequency of less than 3Hz. The nature of the filter results in some peaking, resulting in a -3dB frequency of 1.9Hz and a 0.08dB peak with a system clock frequency of 100MHz assuming a p_shift value of 2. The transfer function is illustrated in Figure 27 and in more detail in Figure 28. Increasing the p_shift value increases the speed the DPLL will lock to the required frequency and reduces the peak, but also reduces the tolerance to jitter - so the p_shift value must be programmed correctly to meet Stratum 3 or Stratum 4/4E jitter transfer characteristics. This is done automatically in the API.

1. There are 2 exceptions to this. a) When reference is 8kHz, and reference frequency offset relative to the master is small, jitter up to 1 master clock period is possible, i.e. 10ns p-p. b) In holdover mode, if a huge amount of jitter had been present prior to entering holdover, then an additional 2ns p-p is possible.

7.8 Maximum Time Interval Error (MTIE)

In order to meet several standards requirements, the phase shift of the DPLL output must be controlled. A potential phase shift occurs every time the DPLL is re-arranged by changing reference source signal, or the mode. In order to meet the requirements of Stratum 3, the DPLL will shift phase by no more than 20ns per re-arrangement.

Additionally the speed at which the change occurs is also critical. A large step change in output frequency is undesirable. The rate of change is programmable using the skew register, up to a maximum of 15.4ns / 125 µs (124ppm).

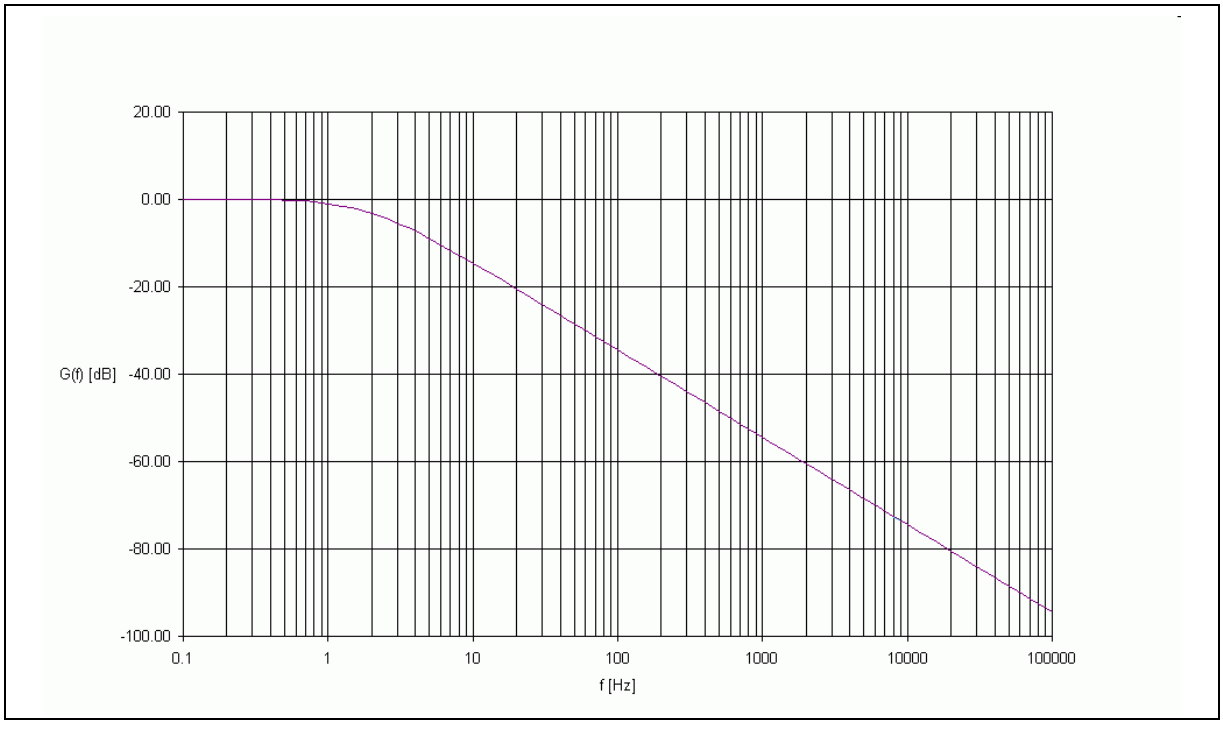


Figure 27 - Jitter Transfer Function

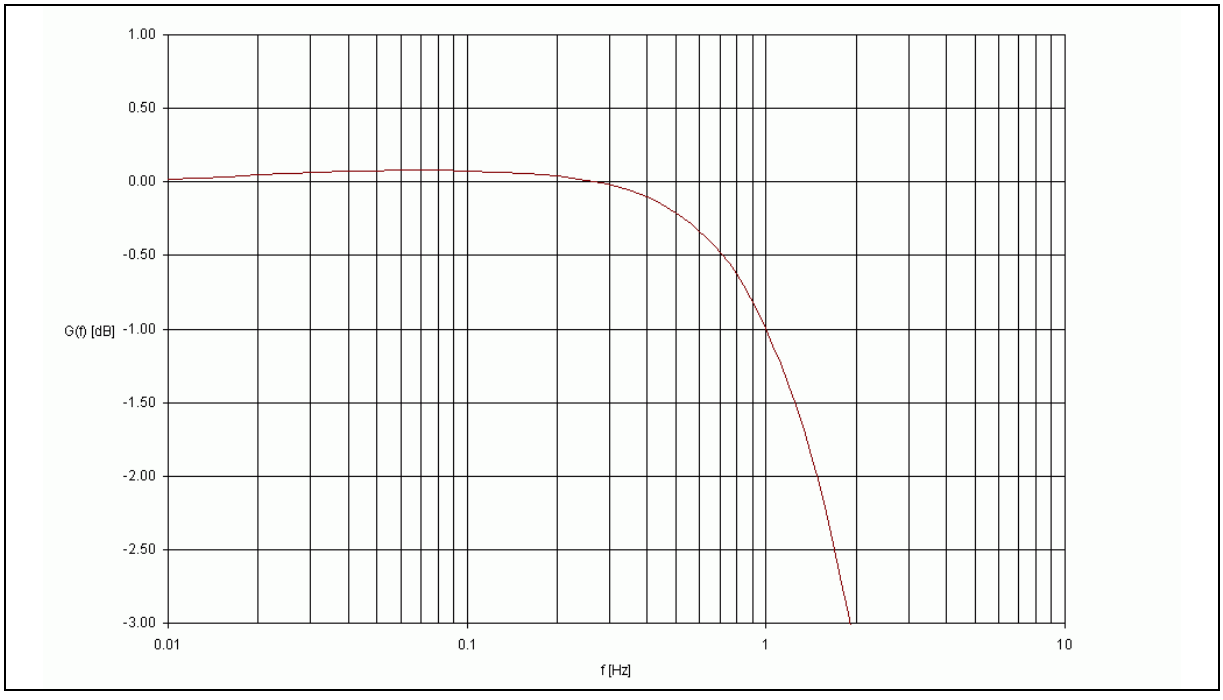


Figure 28 - Jitter Transfer Function - Detail

8.0 Miscellaneous

8.1 JTAG Interface and Board Level Test Features.

The JTAG interface is used to access the boundary scan logic for board level production testing.

8.2 External Component Requirements

- Direct connection to PowerQUICC™ II (MPC8260) host processor and associated memory, but can support other processors with appropriate glue logic.
- TDM Framers and/or Line Interface Units
- Ethernet PHY for each MAC port
- Optional ZBT-SRAM for extended packet memory buffer depth

8.3 Miscellaneous Features

- System clock speed of 100MHz
- Host clock speed of up to 66MHz
- Debug option to freeze all internal state machines
- JTAG (IEEE1149) Test Access Port
- 3.3V I/O Supply rail with 5V tolerance
- 1.8V Core Supply rail
- Fully compatible with MT9088x Zarlink product line

9.0 Memory Map and Register definitions

All memory map and register definitions are included in the ZL50110/1/4 Programmers Model document.

10.0 Test Modes Operation

10.1 Overview

The ZL50110/1/4 family supports the following modes of operation.

10.1.1 System Normal Mode

This mode is the device's normal operating mode. Boundary scan testing of the peripheral ring is accessible in this mode via the dedicated JTAG pins. The JTAG interface is compliant with the IEEE Std. 1149.1-2001; Test Access Port and Boundary Scan Architecture.

Each variant has it's own dedicated.bsdl file which fully describes it's boundary scan architecture.

10.1.2 System Tri-State Mode

All output and I/O output drivers are tri-stated allowing the device to be isolated when testing or debugging the development board.

10.2 Test Mode Control

The System Test Mode is selected using the dedicated device input bus TEST_MODE[2:0] as follows in Table 25.

System Test Mode	test_mode[2:0]
SYS_NORMAL_MODE	3’b000
SYS_TRI_STATE_MODE	3’b011

Table 25 - Test Mode Control

10.3 System Normal Mode

Selected by TEST_MODE[2:0] = 3'b000. As the test_mode[2:0] inputs have internal pull-downs this is the default mode of operation if no external pull-up/downs are connected. The GPIO[15:0] bus is captured on the rising edge of the external reset to provide internal bootstrap options. After the internal reset has been de-asserted the GPIO pins may be configured by the ADM module as either inputs or outputs.

10.4 System Tri-state Mode

Selected by TEST_MODE[2:0] = 3'b011. All device output and I/O output drivers are tri-stated.

11.0 DC Characteristics

11.1 Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Units
I/O Supply Voltage	V_{DD_IO}	-0.5	5.0	V
Core Supply Voltage	V_{DD_CORE}	-0.5	2.5	V
PLL Supply Voltage	V_{DD_PLL}	-0.5	2.5	V
Input Voltage	V_I	-0.5	V_{DD} + 0.5	V
Input Voltage (5V tolerant inputs)	V_{I_5V}	-0.5	7.0	V
Continuous current at digital inputs	I_{IN}	-	±10	mA
Continuous current at digital outputs	I_O	-	±15	mA
Package power dissipation	PD	-	4	W
Storage Temperature	TS	-55	+125	°C

Table 26 - Absolute Maximum Ratings

Exceeding these figures may cause permanent damage. Functional operation under these conditions is not guaranteed. Voltage measurements are with respect to ground (V_{SS}) unless otherwise stated.

The core and PLL supply voltages must never be allowed to exceed the I/O supply voltage by more than 0.5V during power-up. Failure to observe this rule could lead to a high-current latch-up state, possibly leading to chip failure, if sufficient cross-supply current is available. To be safe ensure the I/O supply voltage supply always rises earlier than the core and PLL supply voltages.

11.2 Recommended Operating Conditions

Characteristics	Symbol	Min	Typ	Max	Units	Test Condition
Operating Temperature	T_{OP}	-40	25	+85	°C	
Junction temperature	T_J	-40	-	125	°C	
Positive Supply Voltage, I/O	V_{DD_IO}	3.0	3.3	3.6	V	
Positive Supply Voltage, Core	V_{DD_CORE}	1.65	1.8	1.95	V	
Positive Supply Voltage, Core	V_{DD_PLL}	1.65	1.8	1.95	V	
Input Voltage Low - all inputs	V_{IL}	-	-	0.8	V	
Input Voltage High	V_{IH}	2.0	-	V_{DD_IO}	V	
Input Voltage High, 5V tolerant inputs	V_{IH_5V}	2.0	-	5.5	V	

Table 27 - Recommended Operating Conditions

Typical figures are at 25°C and are for design aid only, they are not guaranteed and not subject to production testing. Voltage measurements are with respect to ground (V_{SS}) unless otherwise stated

11.3 DC Characteristics

Typical characteristics are at 1.8V core, 3.3V I/O, 25°C and typical processing. The min and max values are defined over all process conditions, from -40 to 125°C junction temperature, core voltage 1.65 to 1.95V and I/O voltage 3.0 and 3.6V unless otherwise stated.

Characteristics	Symbol	Min	Typ	Max	Units	Test Condition
Input Leakage	I_{LEIP}			±1	µA	No pull up/down V_{DD} = 3.6V
Output (High impedance) Leakage	I_{LEOP}			1	µA	No pull up/down V_{DD} = 3.6V
Input Capacitance	C_{IP}		2		pF	
Output Capacitance	C_{OP}		4		pF	
Pullup Current	I_{PU}		-33		µA	Input at 0V
Pulldown Current	I_{PD}		33		µA	Input at V_{DD}
Core 1.8V supply current	I_{DD_CORE}			890	mA	Note 1,2
PLL 1.8V supply current	I_{DD_PLL}			1.30	mA	
I/O 3.3V supply current	I_{DD_IO}			180	mA	Note 1,2

Table 28 - DC Characteristics

Note 1: The IO and Core supply current worst case figures apply to different scenarios, e.g. internal or external memory and can not simply be summed for a total figure. For a clearer indication of power consumption, please refer to Section 13.0.

Note 2: Worst case assumes the maximum number of active contexts and channels, i.e. 128 contexts/1024 channels. Figures are for the ZL50111. For an indication of power consumption by the ZL50110 and ZL50114, please refer to Section 13.0 and choose the appropriate memory configuration and number of contexts.

11.4 Input Levels

Characteristics	Symbol	Min	Typ	Max	Units	Test Condition
Input Low Voltage	V_{IL}			0.8	V	
Input High Voltage	V_{IH}	2.0			V	
Positive Schmitt Threshold	V_{T+}		1.6		V	
Negative Schmitt Threshold	V_{T-}		1.2		V	

Table 29 - Input Levels

11.5 Output Levels

Characteristics	Symbol	Min	Typ	Max	Units	Test Condition
Output Low Voltage	V_{OL}			0.4	V	
Output High Voltage	V_{OH}	2.4			V	
Output Low Current	I_{OL}		1.6		mA	
Output High Current	I_{OH}		1.2		mA	

Table 30 - Output Levels

12.0 AC Characteristics

12.1 TDM Interface Timing - ST-BUS

The TDM Bus either operates in Slave mode, where the TDM clocks for each stream are provided by the device sourcing the data, or Master mode, where the TDM clocks are generated from the ZL50110/1/4.

12.1.1 ST-BUS Slave Clock Mode

Data Format	Parameter	Symbol	Min	Typ	Max	Units	Notes
ST-BUS 8.192Mbit/s mode	TDM_CLKi Period	t_{C16IP}	54	60	66	ns	
	TDM_CLKi High	t_{C16IH}	27	-	33	ns	
	TDM_CLKi Low	t_{C16IL}	27	-	33	ns	
ST-BUS 2.048Mbit/s mode	TDM_CLKi Period	t_{C4IP}	-	244.1	-	ns	
	TDM_CLKi High	t_{C4IH}	110	-	134	ns	
	TDM_CLKi Low	t_{C4IL}	110	-	134	ns	
All Modes	$\overline{\text{TDM_F0i}}$ Width 8.192Mbit/s 2.048Mbit/s	t_{FOIW}	 50 200	 - -	 - 300	ns	
	$\overline{\text{TDM_F0i}}$ Setup Time	t_{FOIS}	5	-	-	ns	With respect to TDM_CLKi falling edge
	$\overline{\text{TDM_F0i}}$ Hold Time	t_{FOIH}	5	-	-	ns	With respect to TDM_CLKi falling edge
	TDM_STo Delay	t_{STOD}	1	-	20	ns	With respect to TDM_CLKi Load C_L = 50pF
	TDM_STi Setup Time	t_{STIS}	5	-	-	ns	With respect to TDM_CLKi
	TDM_STi Hold Time	t_{STIH}	5	-	-	ns	With respect to TDM_CLKi

Table 31 - TDM ST-BUS Slave Timing Specification

In synchronous mode the clock must be within the locking range of the DPLL to function correctly (± 245ppm). In asynchronous mode, the clock may be any frequency.

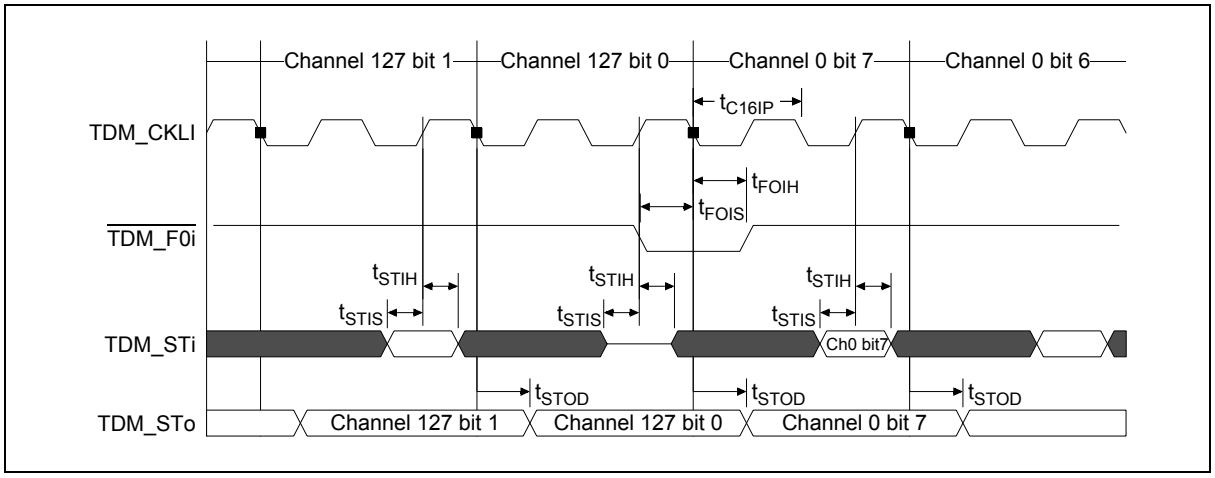


Figure 29 - TDM ST-BUS Slave Mode Timing at 8.192Mbit/s

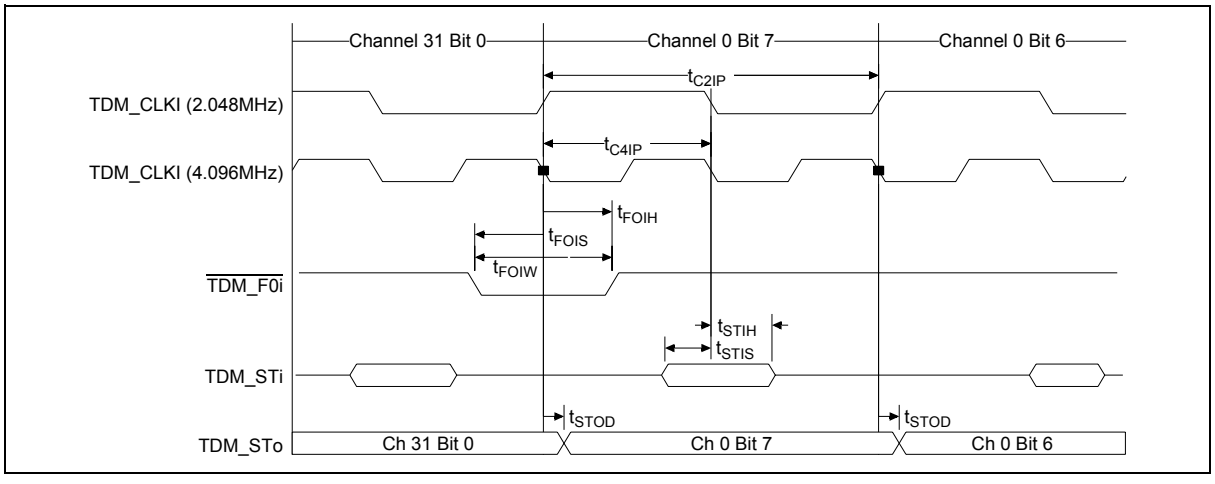


Figure 30 - TDM ST-BUS Slave Mode Timing at 2.048Mbit/s

12.1.2 ST-BUS Master Clock Mode

Data Format	Parameter	Symbol	Min	Typ	Max	Units	Notes
ST-BUS 8.192Mbit/s mode	TDM_CLKo Period	t_{C16OP}	54.0	61.0	68.0	ns	
	TDM_CLKo High	t_{C16OH}	23.0	-	37.0	ns	
	TDM_CLKo Low	t_{C16OL}	23.0	-	37.0	ns	
ST-BUS 2.048Mbit/s mode	TDM_CLKo Period	t_{C4OP}	237.0	244.1	251.0	ns	
	TDM_CLKo High	t_{C4OH}	115.0	-	129.0	ns	
	TDM_CLKo Low	t_{C4OL}	115.0	-	129.0	ns	
All Modes	$\overline{\text{TDM_F0o}}$ Delay	t_{FOD}	-	-	25	ns	With respect to TDM_CLKo falling edge
	TDM_STo Delay Active-Active	t_{STOD}	-	-	5	ns	With respect to TDM_CLKo falling edge
	TDM_STo Delay Active to HiZ and HiZ to Active	t_{DZ}, t_{ZD}	-	-	33	ns	With respect to TDM_CLKo falling edge
	TDM_STi Setup Time	t_{STIS}	5	-	-	ns	With respect to TDM_CLKo
	TDM_STi Hold Time	t_{STIH}	5	-	-	ns	With respect to TDM_CLKo

Table 32 - TDM ST-BUS Master Timing Specification

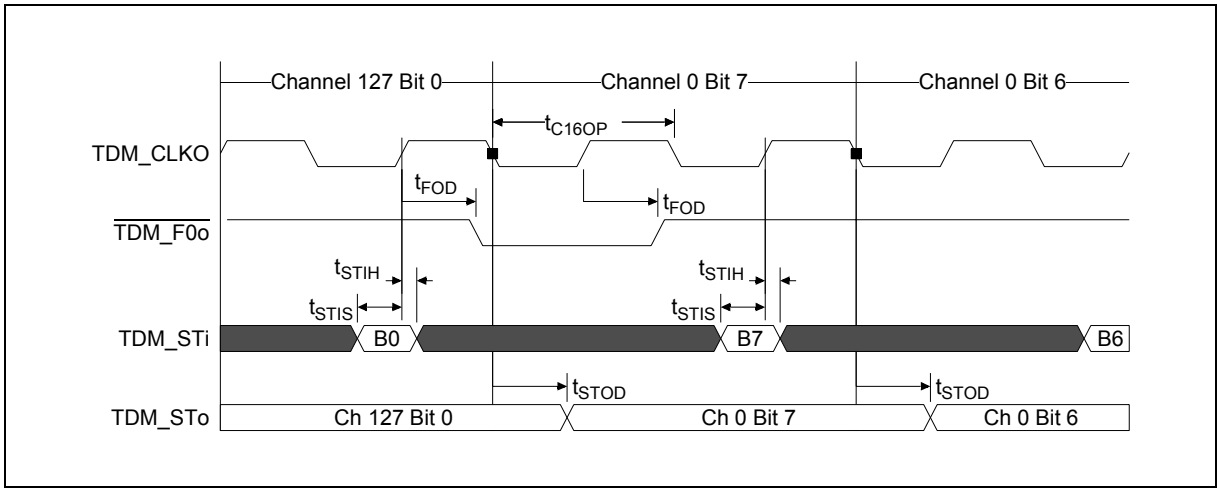


Figure 31 - TDM Bus Master Mode Timing at 8.192Mbit/s

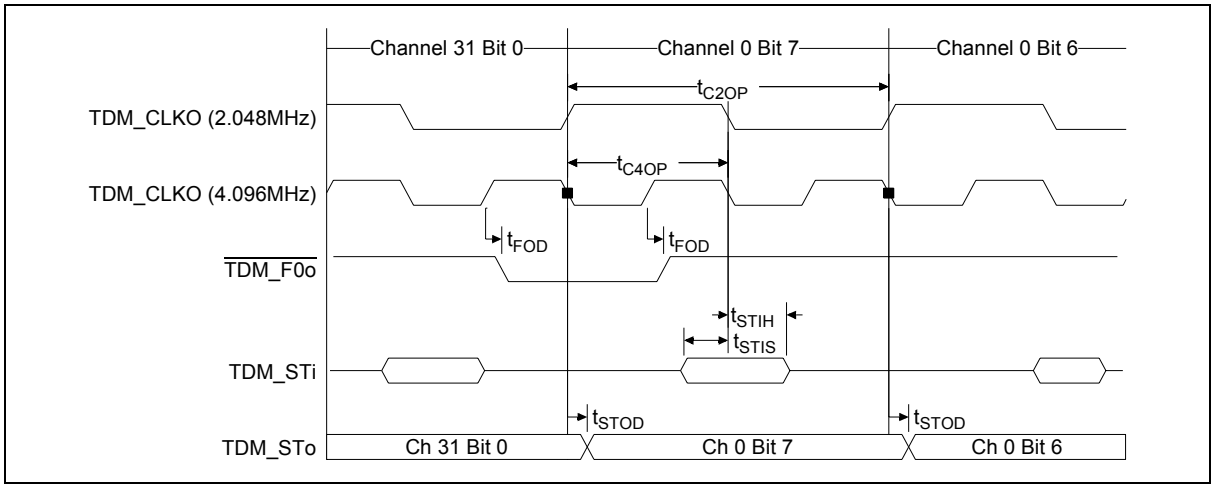


Figure 32 - TDM Bus Master Mode Timing at 2.048Mbit/s

12.2 TDM Interface Timing - H.110 Mode

These parameters are based on the H.110 Specification from the Enterprise Computer Telephony Forum (ECTF) 1997.

Parameter	Symbol	Min	Typ	Max	Units	Notes
TDM_C8 Period	t_{C8P}	122.066-Φ	122	122.074+Φ	ns	Note 1 Note 2
TDM_C8 High	t_{C8H}	63-Φ	-	69+Φ	ns	
TDM_C8 Low	t_{C8L}	63-Φ	-	69+Φ	ns	
TDM_D Output Delay	t_{DOD}	0	-	11	ns	Load - 12pF
TDM_D Output to HiZ	t_{DOZ}	-	-	33	ns	Load - 12pF Note 3
TDM_D HiZ to Output	t_{ZDO}	0	-	11	ns	Load - 12pF Note 3
TDM_D Input Delay to Valid	t_{DV}	0	-	83	ns	Note 4
TDM_D Input Delay to Invalid	t_{DIV}	102	-	112	ns	Note 4
$\overline{\text{TDM_FRAME}}$ width	t_{FP}	90	122	180	ns	Note 5
$\overline{\text{TDM_FRAME}}$ setup	t_{FS}	45	-	90	ns	
$\overline{\text{TDM_FRAME}}$ hold	t_{FH}	45	-	90	ns	
Phase Correction	F	0	-	10	ns	Note 6

Table 33 - TDM H.110 Timing Specification

Note 1: TDM_C8 and $\overline{\text{TDM_FRAME}}$ signals are required to meet the same timing standards and so are not defined independently.

Note 2: TDM_C8 corresponds to pin TDM_CLKi

Note 3: t_{DOZ} and t_{ZDO} apply at every time-slot boundary.

Note 4: Refer to H.110 Standard from Enterprise Computer Telephony Forum (ECTF) for the source of these numbers

Note 5: The $\overline{\text{TDM_FRAME}}$ signal is centred on the rising edge of TDM_C8. All timing measurements are based on this rising edge point; $\overline{\text{TDM_FRAME}}$ corresponds to pin TDM_F0i

Note 6: Phase correction (Φ) results from DPLL timing corrections

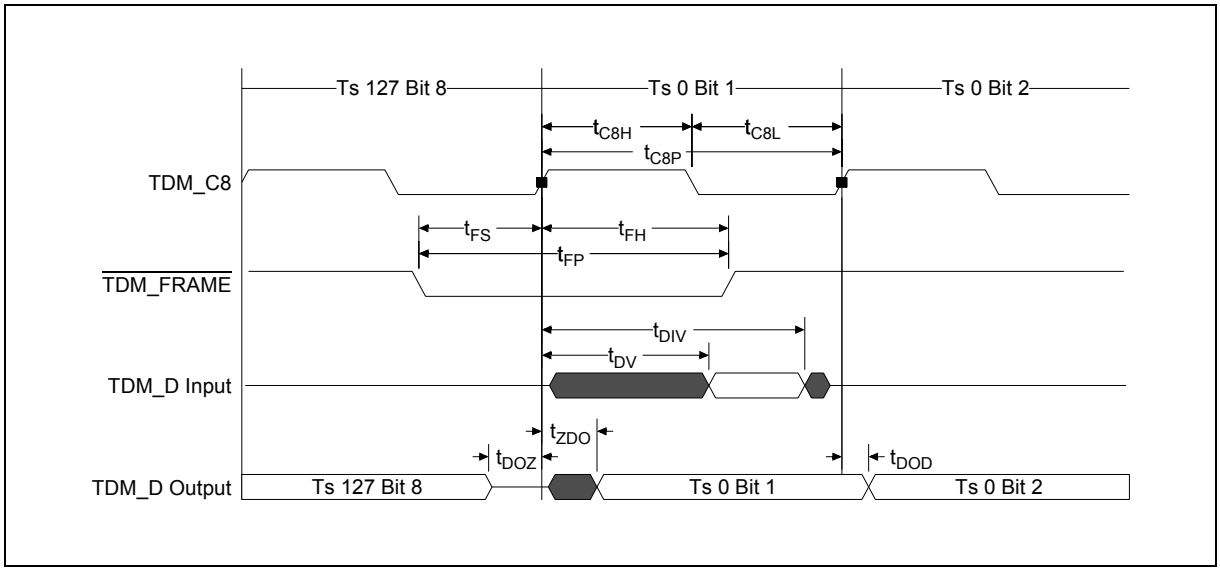


Figure 33 - H.110 Timing Diagram

12.3 TDM Interface Timing - H-MVIP

These parameters are based on the Multi-Vendor Integration Protocol (MVIP) specification for an H-MVIP Bus, Release 1.1a (1997).

Positive transitions of TDM_C2 are synchronous with the falling edges of $\overline{\text{TDM_C4}}$ and $\overline{\text{TDM_C16}}$. The signals TDM_C2, $\overline{\text{TDM_C4}}$ and $\overline{\text{TDM_C16}}$ correspond with pins TDM_CLKi. The signals $\overline{\text{TDM_F0}}$ correspond with pins $\overline{\text{TDM_F0i}}$. The signals TDM_HDS correspond with pins TDM_STi and TDM_STo.

Parameter	Symbol	Min	Typ	Max	Units	Notes
TDM_C2 Period	t_{C2P}	487.8	488.3	488.8	ns	
TDM_C2 High	t_{C2H}	220	-	268	ns	
TDM_C2 Low	t_{C2L}	220	-	268	ns	
$\overline{\text{TDM_C4}}$ Period	t_{C4P}	243.9	244.1	244.4	ns	
$\overline{\text{TDM_C4}}$ High	t_{C4H}	110	-	134	ns	
$\overline{\text{TDM_C4}}$ Low	t_{C4L}	110	-	134	ns	
$\overline{\text{TDM_C16}}$ Period	t_{C16P}	60.9	61.0	61.1	ns	
$\overline{\text{TDM_C16}}$ High	t_{C16H}	30	-	31	ns	
$\overline{\text{TDM_C16}}$ Low	t_{C16L}	30	-	31	ns	
TDM_HDS Output Delay	t_{PD}	-	-	30	ns	At 8.192Mb/s
TDM_HDS Output Delay	t_{PD}	-	-	100	ns	At 2.048Mb/s
TDM_HDS Output to HiZ	t_{HZD}	-	-	30	ns	
TDM_HDS Input Setup	t_S	30	-	0	ns	
TDM_HDS Input Hold	t_H	30	-	0	ns	

Parameter	Symbol	Min	Typ	Max	Units	Notes
$\overline{\text{TDM_F0}}$ width	t_{FW}	200	244	300	ns	
$\overline{\text{TDM_F0}}$ setup	t_{FS}	50	-	150	ns	
$\overline{\text{TDM_F0}}$ hold	t_{FH}	50	-	150	ns	

Table 34 - TDM H-MVIP Timing Specification

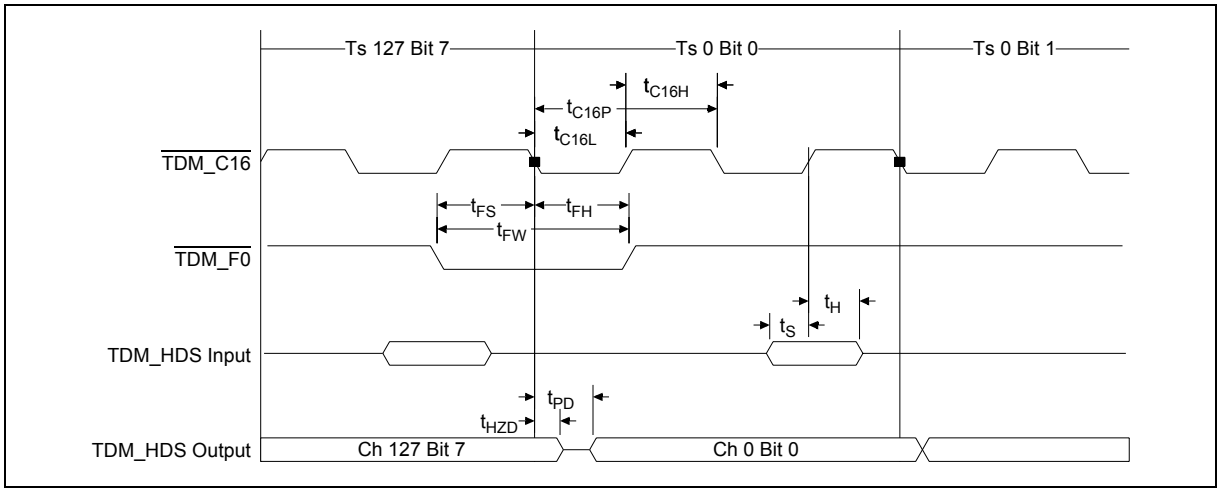


Figure 34 - TDM - H-MVIP Timing Diagram for 16MHz clock (8.192Mbit/s)

12.4 TDM LIU Interface Timing

The TDM Interface can be used to directly drive into a Line Interface Unit (LIU). The interface can work in this mode with E1, DS1, J2, E3 and DS3. The frame pulse is not present, just data and clock is transmitted and received. Table 35 shows timing for DS3, which would be the most stringent requirement.

Parameter	Symbol	Min	Typ	Max	Units	Notes
TDM_TXCLK Period	t_{CTP}		22.353		ns	DS3 clock
TDM_TXCLK High	t_{CTH}	6.7			ns	
TDM_TXCLK Low	t_{CTL}	6.7			ns	
TDM_RXCLK Period	t_{CRP}		22.353		ns	DS3 clock
TDM_RXCLK High	t_{CRH}	9.0			ns	
TDM_RXCLK Low	t_{CRL}	9.0			ns	
TDM_TXDATA Output Delay	t_{PD}	3	-	10	ns	
TDM_RXDATA Input Setup	t_S	6			ns	
TDM_RXDATA Input Hold	t_H	3			ns	

Table 35 - TDM - LIU Structured Transmission/Reception

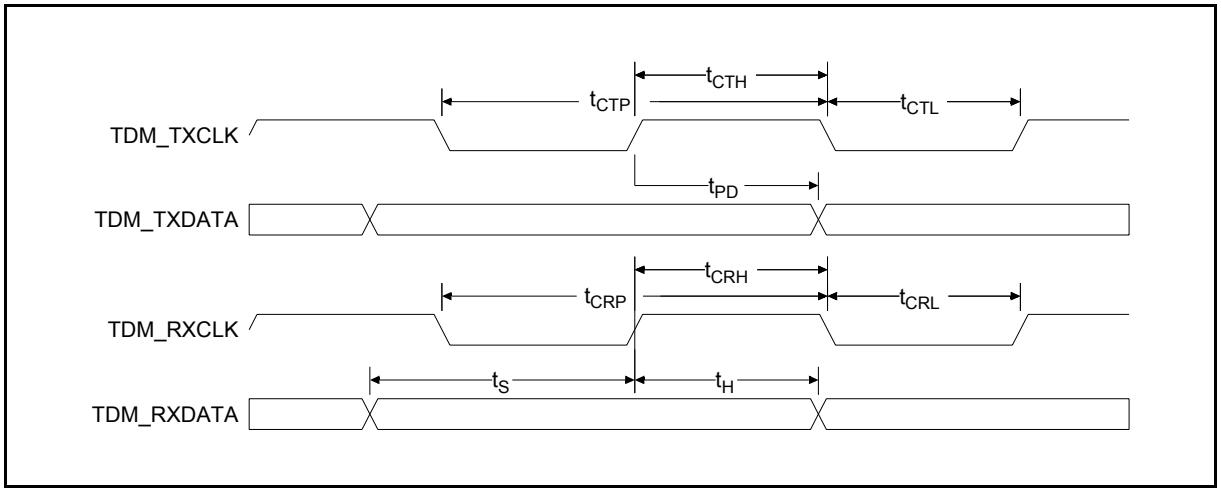


Figure 35 - TDM-LIU Structured Transmission/Reception

12.5 PAC Interface Timing

Parameter	Symbol	Min	Typ	Max	Units	Notes
TDM_CLKiP High / Low Pulsewidth	t_{CPP}	10	-	-	ns	
TDM_CLKiS High / Low Pulsewidth	t_{CSP}	10	-	-	ns	

Table 36 - PAC Timing Specification

12.6 Packet Interface Timing

Data for the MII/GMII/PCS packet switching is based on Specification IEEE Std. 802.3 - 2000.

12.6.1 MII Transmit Timing

Parameter	Symbol	100Mbit/s			Units	Notes
		Min	Typ	Max		
TXCLK period	t_{CC}	-	40	-	ns	
TXCLK high time	t_{CHI}	14	-	26	ns	
TXCLK low time	t_{CLO}	14	-	26	ns	
TXCLK rise time	t_{CR}	-	-	5	ns	
TXCLK fall time	t_{CF}	-	-	5	ns	
TXCLK rise to TXD[3:0] active delay (TXCLK rising edge)	t_{DV}	1	-	25	ns	Load = 25pF
TXCLK to TXEN active delay (TXCLK rising edge)	t_{EV}	1	-	25	ns	Load = 25pF
TXCLK to TXER active delay (TXCLK rising edge)	t_{ER}	1	-	25	ns	Load = 25pF

Table 37 - MII Transmit Timing - 100Mbit/s

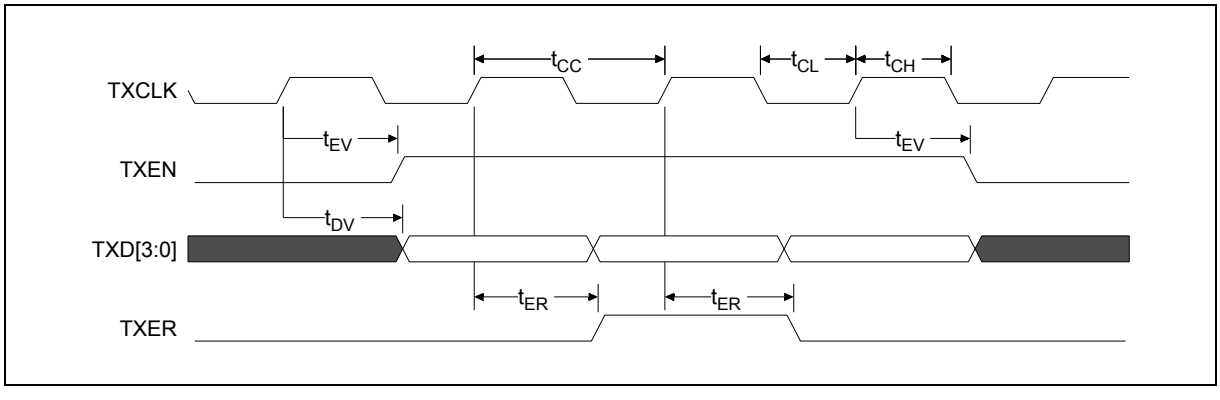


Figure 36 - MII Transmit Timing Diagram

12.6.2 MII Receive Timing

Parameter	Symbol	100Mbit/s			Units	Notes
		Min	Typ	Max		
RXCLK period	t_{CC}	-	40	-	ns	
RXCLK high wide time	t_{CH}	14	20	26	ns	
RXCLK low wide time	t_{CL}	14	20	26	ns	
RXCLK rise time	t_{CR}	-	-	5	ns	
RXCLK fall time	t_{CF}	-	-	5	ns	
RXD[3:0] setup time (RXCLK rising edge)	t_{DS}	10	-	-	ns	
RXD[3:0] hold time (RXCLK rising edge)	t_{DH}	5	-	-	ns	
RXDV input setup time (RXCLK rising edge)	t_{DVS}	10	-	-	ns	
RXDV input hold time (RXCLK rising edge)	t_{DVH}	5	-	-	ns	
RXER input setup time (RXCL edge)	t_{ERS}	10	-	-	ns	
RXER input hold time (RXCLK rising edge)	t_{ERH}	5	-	-	ns	

Table 38 - MII Receive Timing - 100Mbit/s

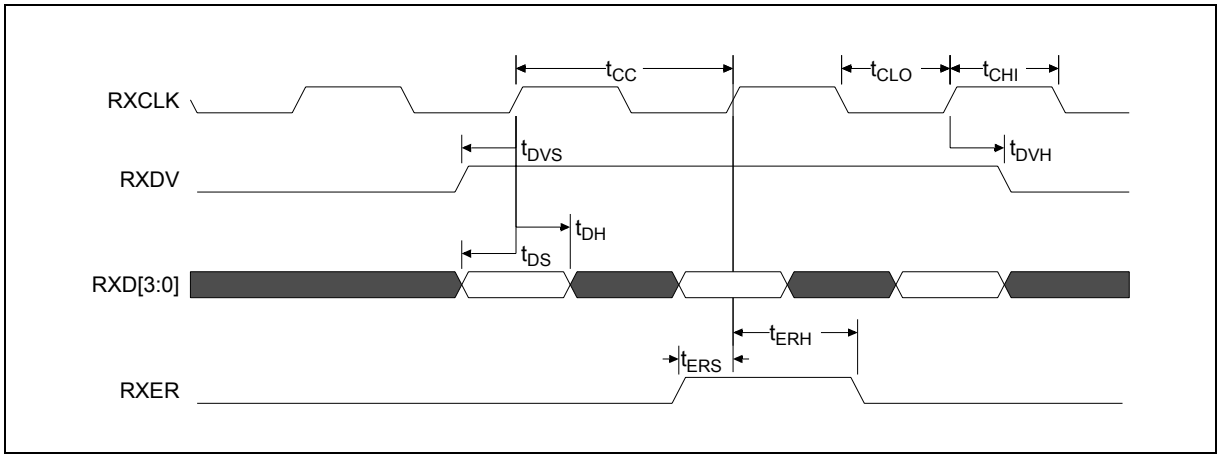


Figure 37 - MII Receive Timing Diagram

12.6.3 GMII Transmit Timing

Parameter	Symbol	1000Mbit/s			Units	Notes
		Min	Typ	Max		
GTXCLK period	t_{GC}	7.5	-	8.5	ns	
GTXCLK high time	t_{GCH}	2.5	-	-	ns	
GTXCLK low time	t_{GCL}	2.5	-	-	ns	
GTXCLK rise time	t_{GCR}	-	-	1	ns	
GTXCLK fall time	t_{GCF}	-	-	1	ns	
GTXCLK rise to TXD[7:0] active delay	t_{DV}	1.5	-	6	ns	Load = 25pF
GTXCLK rise to TXEN active delay	t_{EV}	2	-	6	ns	Load = 25pF
GTXCLK rise to TXER active delay	t_{ER}	1	-	6	ns	Load = 25pF

Table 39 - GMII Transmit Timing - 1000Mbit/s

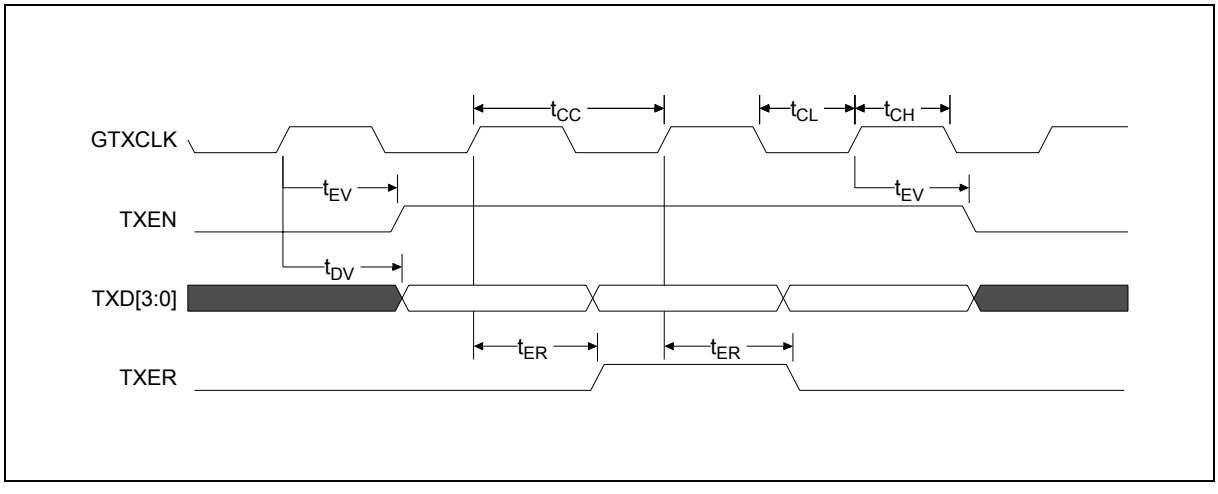


Figure 38 - GMII Transmit Timing Diagram

12.6.4 GMII Receive Timing

Parameter	Symbol	1000Mbit/s			Units	Notes
		Min	Typ	Max		
RXCLK period	t_{CC}	7.5	-	8.5	ns	
RXCLK high wide time	t_{CH}	2.5	-	-	ns	
RXCLK low wide time	t_{CL}	2.5	-	-	ns	
RXCLK rise time	t_{CR}	-	-	1	ns	
RXCLK fall time	t_{CF}	-	-	1	ns	
RXD[7:0] setup time (RXCLK rising edge)	t_{DS}	2	-	-	ns	
RXD[7:0] hold time (RXCLK rising edge)	t_{DH}	1	-	-	ns	
RXDV setup time (RXCLK rising edge)	t_{DVS}	2	-	-	ns	
RXDV hold time (RXCLK rising edge)	t_{DVH}	1	-	-	ns	
RXER setup time (RXCLK rising edge)	t_{ERS}	2	-	-	ns	
RXER hold time (RXCLK rising edge)	t_{ERH}	1	-	-	ns	

Table 40 - GMII Receive Timing - 1000Mbit/s

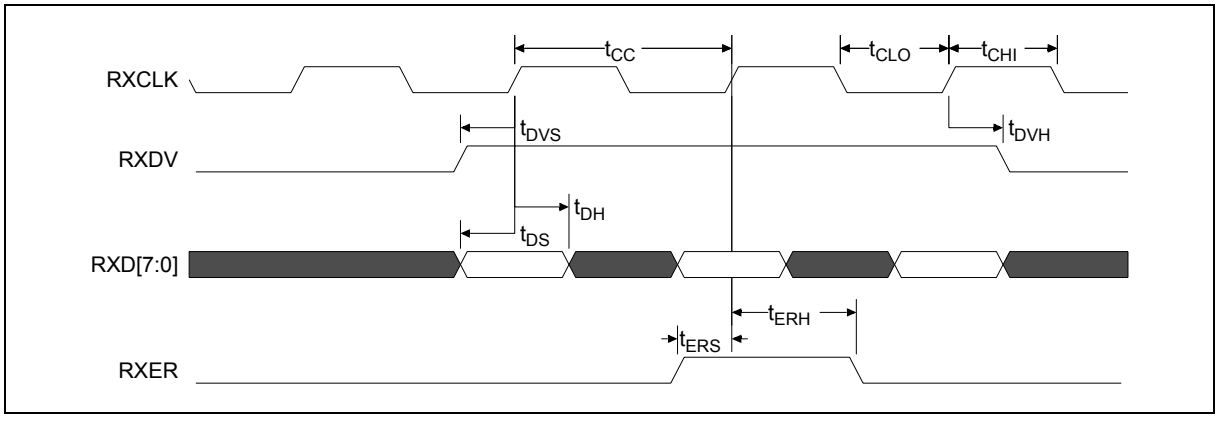


Figure 39 - GMII Receive Timing Diagram

12.6.5 PCS Interface Timing

Parameter	Symbol	1000Mbit/s			Units	Notes
		Min	Typ	Max		
GTXCLK period	t_{GC}	7.5	-	8.5	ns	
GTXCLK high wide time	t_{GH}	2.5	-	-	ns	
GTXCLK low wide time	t_{GL}	2.5	-	-	ns	
TXD[9:0] Output Delay (GTXCLK rising edge)	t_{DV}	1	-	6		Load = 25pF
RCB0/RBC1 period	t_{RC}	15	16	17	ns	
RCB0/RBC1 high wide time	t_{RH}	5	-	-	ns	
RCB0/RBC1 low wide time	t_{RL}	5	-	-	ns	
RCB0/RBC1 rise time	t_{RR}	-	-	2	ns	
RCB0/RBC1 fall time	t_{RF}	-	-	2	ns	
RXD[9:0] setup time (RCB0 rising edge)	t_{DS}	2	-	-	ns	
RXD[9:0] hold time (RCB0 rising edge)	t_{DH}	1	-	-	ns	
REFCLK period	t_{FC}	7.5	-	8.5	ns	
REFCLK high wide time	t_{FH}	2.5	-	-	ns	
REFCLK low wide time	t_{FL}	2.5	-	-	ns	

Table 41 - PCS Timing - 1000Mbit/s

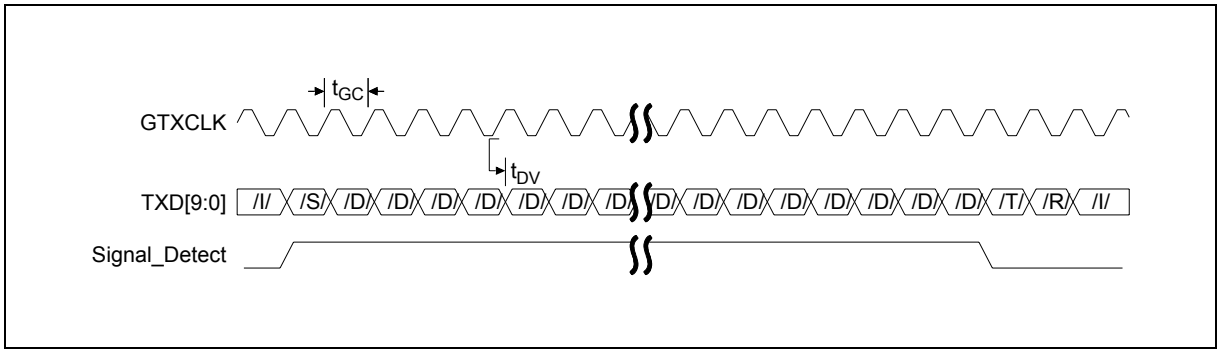


Figure 40 - PCS Transmit Timing Diagram

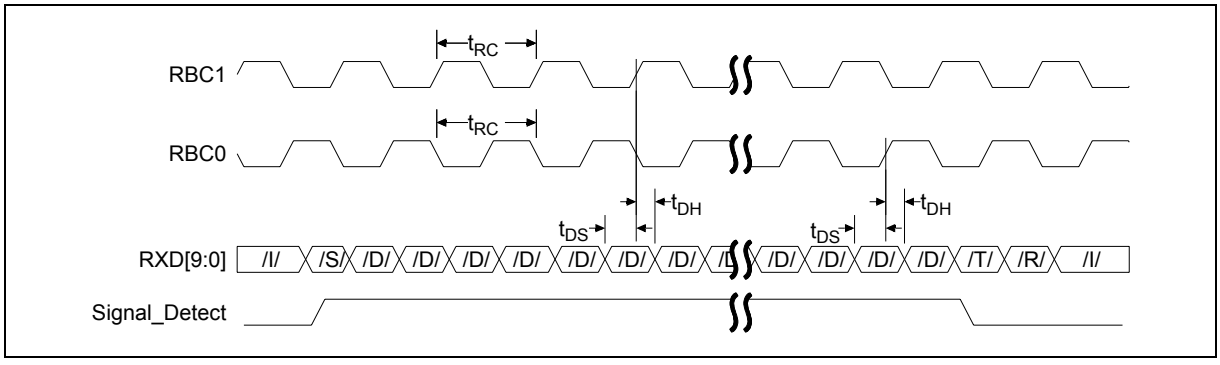


Figure 41 - PCS Receive Timing Diagram

12.6.6 Management Interface Timing

The management interface is common for all inputs and consists of a serial data I/O line and a clock line.

Parameter	Symbol	Min	Typ	Max	Units	Notes
M_MDC Clock Output period	t_{MP}	1990	2000	2010	ns	Note 1
M_MDC high	t_{MHI}	900	1000	1100	ns	
M_MDC low	t_{MLO}	900	1000	1100	ns	
M_MDC rise time	tMR	-	-	5	ns	
M_MDC fall time	t_{MF}	-	-	5	ns	
M_MDIO setup time (MDC rising edge)	t_{MS}	10	-	-	ns	Note 1
M_MDIO hold time (M_MDC rising edge)	t_{MH}	10	-	-	ns	Note 1
M_MDIO Output Delay (M_MDC rising edge)	t_{MD}	1	-	300	ns	Note 2

Table 42 - MAC Management Timing Specification

Note 1: Refer to Clause 22 in IEEE802.3 (2000) Standard for input/output signal timing characteristics

Note 2: Refer to Clause 22C.4 in IEEE802.3 (2000) Standard for output load description of MDIO

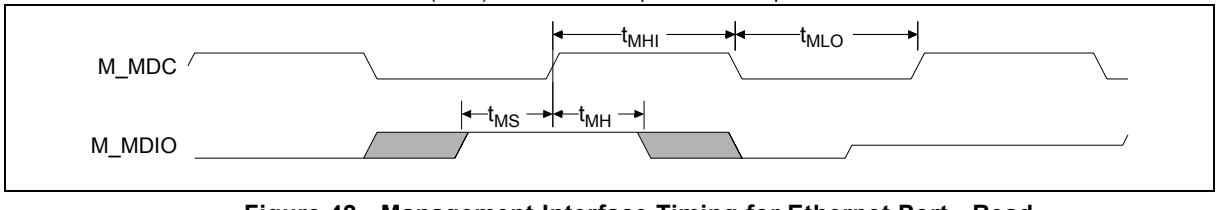


Figure 42 - Management Interface Timing for Ethernet Port - Read

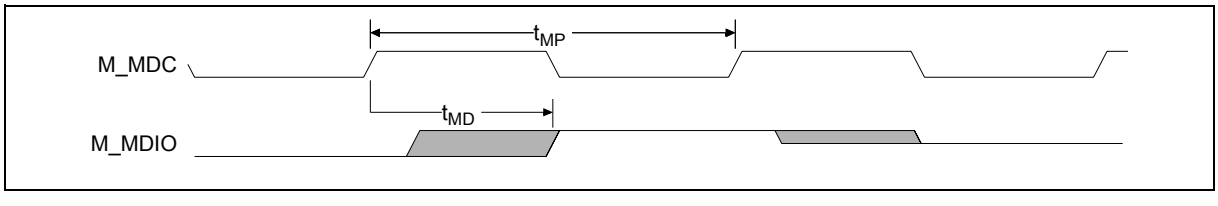


Figure 43 - Management Interface Timing for Ethernet Port - Write

12.7 External Memory Interface Timing

The timings for the External Memory Interface are based on the requirements of a ZBT-SRAM device, with the system clock speed at 100MHz.

Parameter	Symbol	Min	Typ	Max	Units	Notes
RAM_DATA[63:0] Output Valid Delay	t_{RDV}	-	-	4	ns	Load C_L = 30pF
RAM_RW/RAM_ADDR[19:0] Delay	t_{RAV}	-	-	4	ns	Load C_L = 30pF Note 1
RAM_BW[7:0]# Delay	t_{RBW}	-	-	4	ns	Load C_L = 30pF
RAM_DATA[63:0] Setup Time	t_{RDS}	2	-	-	ns	
RAM_DATA[63:0] Hold Time	t_{RDH}	0.5	-	-	ns	
RAM_PARITY[7:0] Output Valid Delay	t_{RPV}	-	-	4	ns	Load C_L = 30pF
RAM_PARITY[7:0] Setup Time	t_{RPS}	2	-	-	ns	
RAM_PARITY[7:0] Hold Time	t_{RPS}	0.5	-	-	ns	

Table 43 - External Memory Timing

Note 1: Must be capable of driving TWO separate RAM loads simultaneously

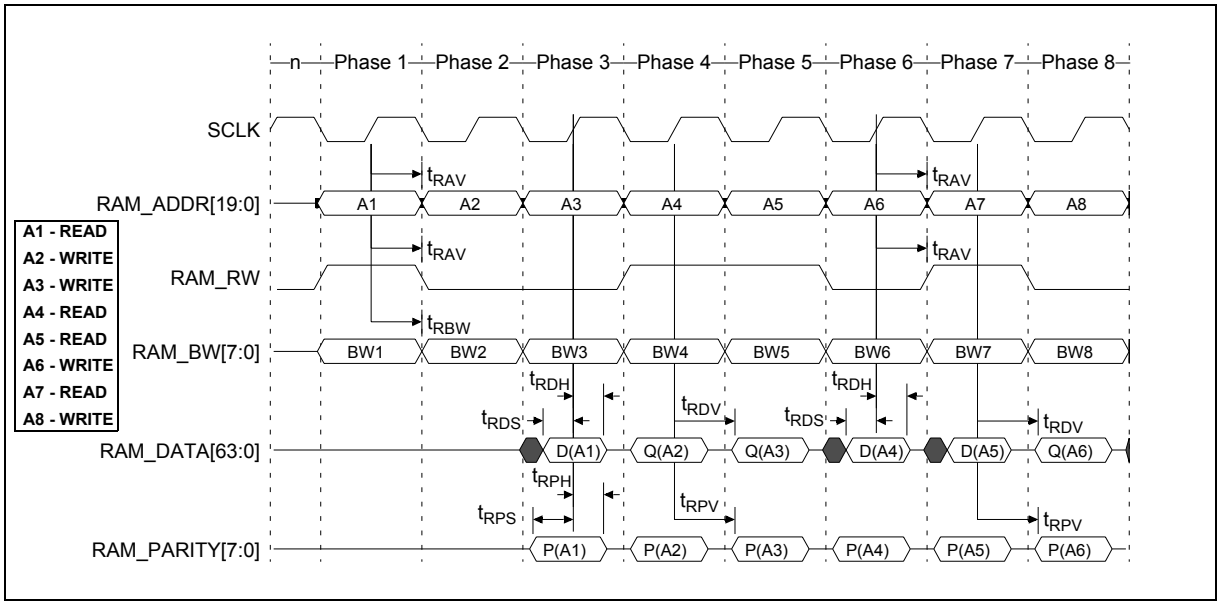


Figure 44 - External RAM Read and Write timing

12.8 CPU Interface Timing

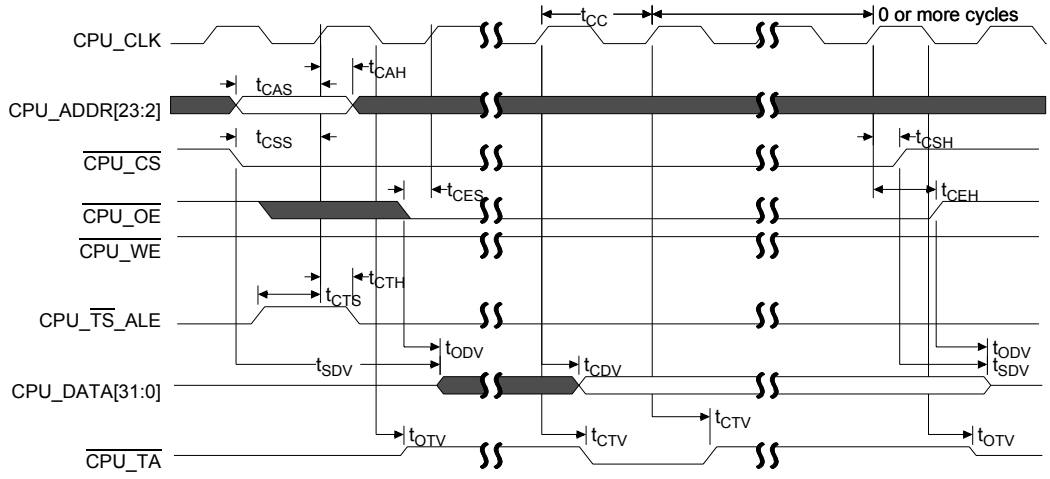
Parameter	Symbol	Min	Typ	Max	Units	Notes
CPU_CLK Period	t_{CC}		15.152		ns	
CPU_CLK High Time	t_{CCH}	6			ns	
CPU_CLK Low Time	t_{CCL}	6			ns	
CPU_CLK Rise Time	t_{CCR}			4	ns	
CPU_CLK Fall Time	t_{CCF}			4	ns	
CPU_ADDR[23:2] Setup Time	t_{CAS}	4			ns	
CPU_ADDR[23:2] Hold Time	t_{CAH}	2			ns	
CPU_DATA[31:0] Setup Time	t_{CDS}	4			ns	
CPU_DATA[31:0] Hold Time	t_{CDH}	2			ns	
$\overline{\text{CPU_CS}}$ Setup Time	t_{CSS}	4			ns	
$\overline{\text{CPU_CS}}$ Hold Time	t_{CSH}	2			ns	
$\overline{\text{CPU_WE}}$/$\overline{\text{CPU_OE}}$ Setup Time	t_{CES}	5			ns	
$\overline{\text{CPU_WE}}$/$\overline{\text{CPU_OE}}$ Hold Time	t_{CEH}	2			ns	
CPU_$\overline{\text{TS}}$_ALE Setup Time	t_{CTS}	4			ns	
CPU_$\overline{\text{TS}}$_ALE Hold Time	t_{CTH}	2			ns	
$\overline{\text{CPU_SDACK1}}$/$\overline{\text{CPU_SDACK2}}$ Setup Time	t_{CKS}	2			ns	
$\overline{\text{CPU_SDACK1}}$/$\overline{\text{CPU_SDACK2}}$ Hold Time	t_{CKH}	2			ns	Note 1
$\overline{\text{CPU_TA}}$ Output Valid Delay	t_{CTV}	2		11.3	ns	Note 1,2
$\overline{\text{CPU_DREQ0}}$/$\overline{\text{CPU_DREQ1}}$ Output Valid Delay	t_{CWV}	2		6	ns	Note 1
$\overline{\text{CPU_IREQ0}}$/$\overline{\text{CPU_IREQ1}}$ Output Valid Delay	t_{CRV}	2		6	ns	Note 1
CPU_DATA[31:0] Output Valid Delay	t_{CDV}	2		7	ns	Note 1
$\overline{\text{CPU_CS}}$ to Output Data Valid	t_{SDV}	3.2		10.4	ns	
$\overline{\text{CPU_OE}}$ to Output Data Valid	t_{ODV}	3.3		10.4	ns	
CPU_CLK(falling) to $\overline{\text{CPU_TA}}$ Valid	t_{OTV}	3.2		9.5	ns	

Table 44 - CPU Timing Specification

Note 1: Load = 50pF maximum

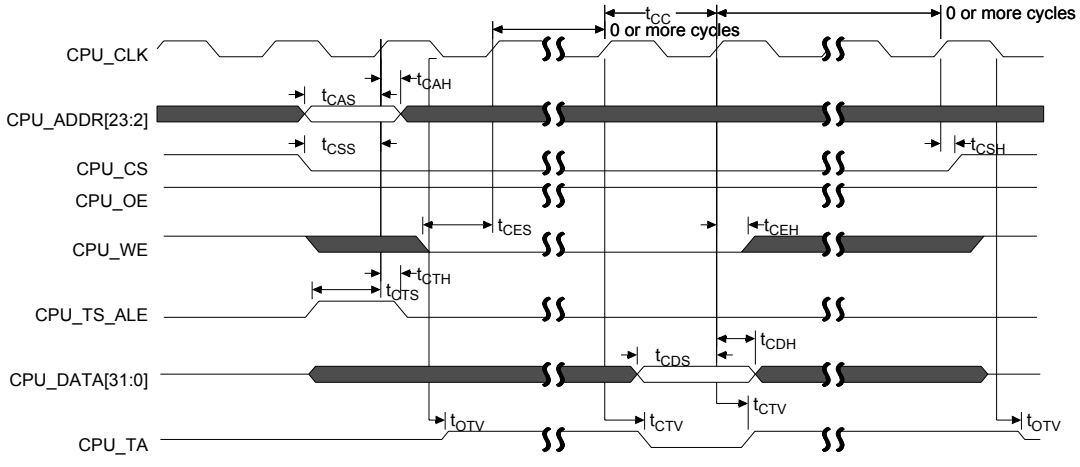
Note 2: The maximum value of t_{CTV} may cause setup violations if directly connected to the MPC8260. See Section 14.2 for details of how to accommodate this during board design.

The actual point where read/write data is transferred occurs at the positive clock edge following the assertion of CPU_TA, not at the positive clock edge during the assertion of CPU_TA.



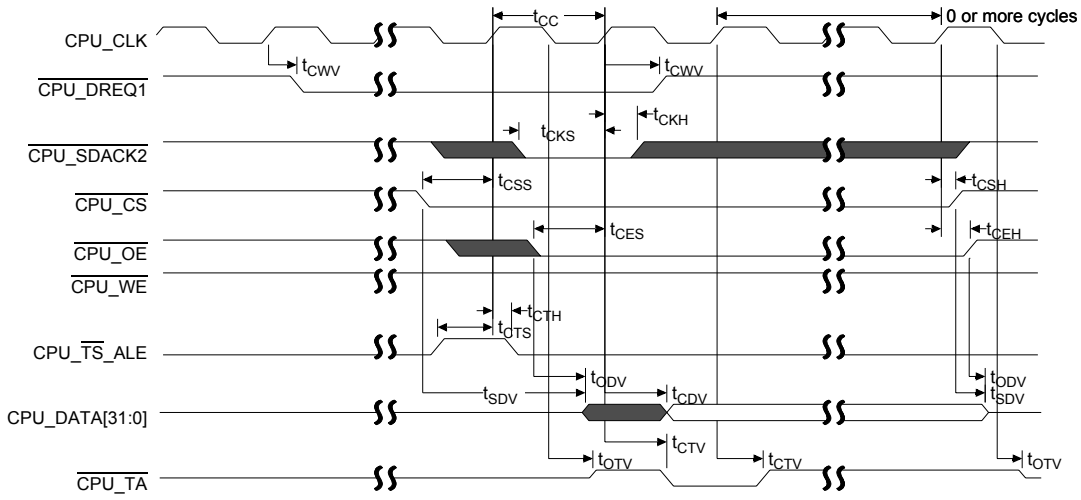
NOTE: CPU_DATA is valid when CPU_TA is asserted. CPU_DATA will remain valid while both CPU_CS and CPU_OE are asserted. CPU_TA will continue to be driven until CPU_CS is deasserted. CPU_CS and CPU_OE must BOTH be asserted to enable the CPU_DATA output.

Figure 45 - CPU Read - MPC8260



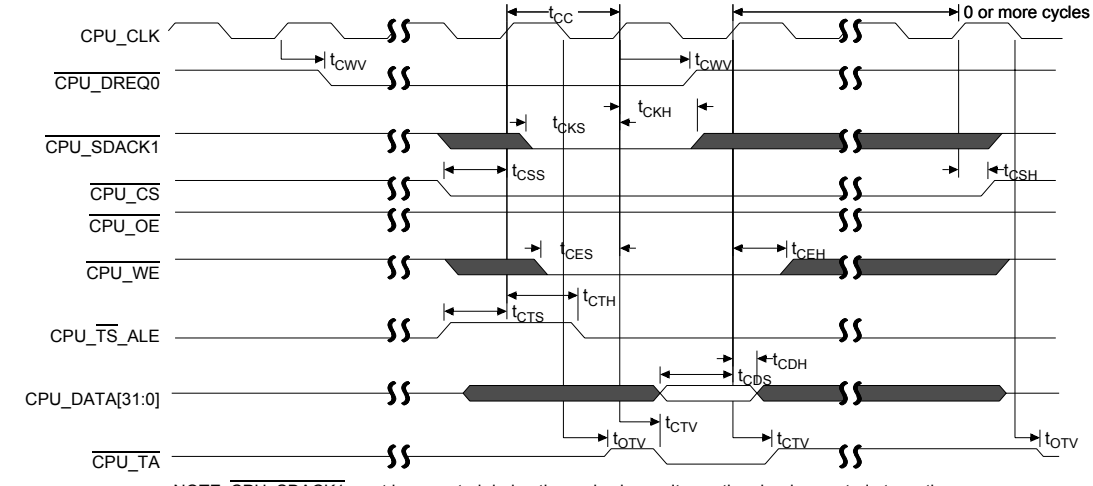
NOTE: Following assertion of CPU_TA, CPU_CS may be deasserted. The MPC8260 will continue to assert CPU_CS until CPU_TA has been synchronized internally. CPU_TA will continue to be driven until CPU_CS is finally deasserted. During continued assertion of CPU_CS, CPU_WE and CPU_DATA may be removed.

Figure 46 - CPU Write - MPC8260



NOTE: CPU_SDACK2 must be asserted during the cycle shown. It may then be deasserted at any time. CPU_DATA is valid when CPU_TA is asserted (always timed as shown). CPU_DATA will remain valid while CPU_CS and CPU_OE are asserted. CPU_TA will continue to be driven until CPU_CS is deasserted. CPU_CS and CPU_OE must BOTH be asserted to enable the CPU_DATA output.

Figure 47 - CPU DMA Read - MPC8260



NOTE: CPU_SDACK1 must be asserted during the cycle shown. It may then be deasserted at any time. Following assertion of CPU_TA (always timed as shown), CPU_CS may be deasserted. The MPC8260 will continue to assert CPU_CS until CPU_TA has been synchronized internally. CPU_TA will continue to be driven until CPU_CS is finally deasserted. During continued assertion of CPU_CS, CPU_WE and CPU_DATA may be removed.

Figure 48 - CPU DMA Write - MPC8260

12.9 System Function Port

Parameter	Symbol	Min	Typ	Max	Units	Notes
SYSTEM_CLK Frequency	CLK_{FR}	-	100	-	MHz	Note 1 and Note 2
SYSTEM_CLK accuracy (synchronous master mode)	CLK_{ACS}	-	-	±30	ppm	Note 3
SYSTEM_CLK accuracy (synchronous slave mode and asynchronous mode)	CLK_{ACA}	-	-	±200	ppm	Note 4

Table 45 - System Clock Timing

Note 1: The system clock frequency stability affects the holdover-operating mode of the DPLL. Holdover Mode is typically used for a short duration while network synchronisation is temporarily disrupted. Drift on the system clock directly affects the Holdover Mode accuracy. Note that the absolute system clock accuracy does not affect the Holdover accuracy, only the change in the system clock (SYSTEM_CLK) accuracy while in Holdover. For example, if the system clock oscillator has a temperature coefficient of 0.1ppm/ºC, a 10ºC change in temperature while the DPLL is in will result in a frequency accuracy offset of 1ppm. The intrinsic frequency accuracy of the DPLL Holdover Mode is 0.06 ppm, excluding the system clock drift.

Note 2: The system clock frequency affects the operation of the DPLL in free-run mode. In this mode, the DPLL provides timing and synchronisation signals which are based on the frequency of the accuracy of the master clock (i.e. frequency of clock output equals 8.192MHz ± SYSTEM_CLK accuracy ± 0.005ppm)

Note 3: The absolute SYSTEM_CLK accuracy must be controlled to ± 30 ppm in synchronous master mode to enable the internal DPLL to function correctly

Note 4: In asynchronous mode and in synchronous slave mode the DPLL is not used. Therefore the tolerance on SYSTEM_CLK may be relaxed slightly.

12.10 JTAG Interface Timing

Parameter	Symbol	Min	Typ	Max	Units	Notes
JTAG_CLK period	t_{JCP}	40	100		ns	
JTAG_CLK clock pulse width	t_{LOW}, t_{HIGH}	20	-	-	ns	
JTAG_CLK rise and fall time	t_{JRF}	0	-	3	ns	
$\overline{\text{JTAG_TRST}}$ setup time	t_{RSTSU}	10	-	-	ns	With respect to JTAG_CLK falling edge. Note 1
$\overline{\text{JTAG_TRST}}$ assert time	t_{RST}	10	-	-	ns	
Input data setup time	t_{JSU}	5	-	-	ns	Note 2
Input Data hold time	t_{JH}	15	-	-	ns	Note 2
JTAG_CLK to Output data valid	t_{JDV}	0	-	20	ns	Note 3
JTAG_CLK to Output data high impedance	t_{JZ}	0	-	20	ns	Note 3
JTAG_TMS, JTAG_TDI setup time	t_{TPSU}	5	-	-	ns	
JTAG_TMS, JTAG_TDI hold time	t_{TPH}	15	-	-	ns	
JTAG_TDO delay	t_{TOPDV}	0	-	15	ns	
JTAG_TDO delay to high impedance	t_{TPZ}	0	-	15	ns	

Table 46 - JTAG Interface Timing

Note 1: $\overline{\text{JTAG_TRST}}$ is an asynchronous signal. The setup time is for test purposes only.

Note 2: Non Test (other than JTAG_TDI and JTAG_TMS) signal input timing with respect to JTAG_CLK

Note 3: Non Test (other than JTAG_TDO) signal output with respect to JTAG_CLK

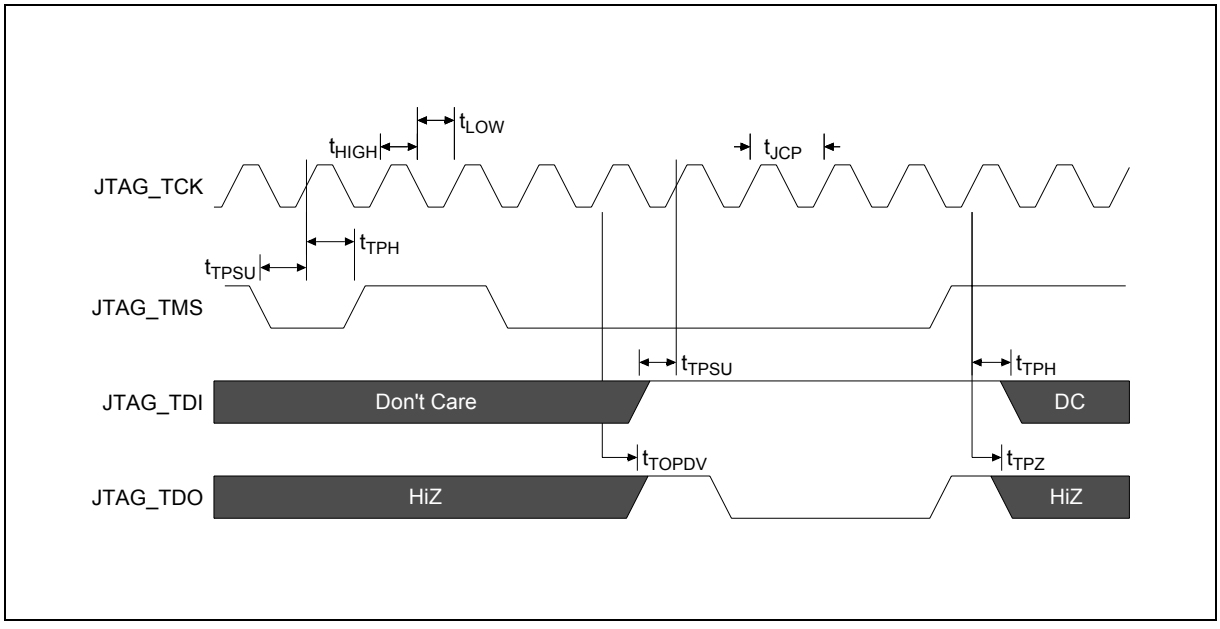


Figure 49 - JTAG Signal Timing

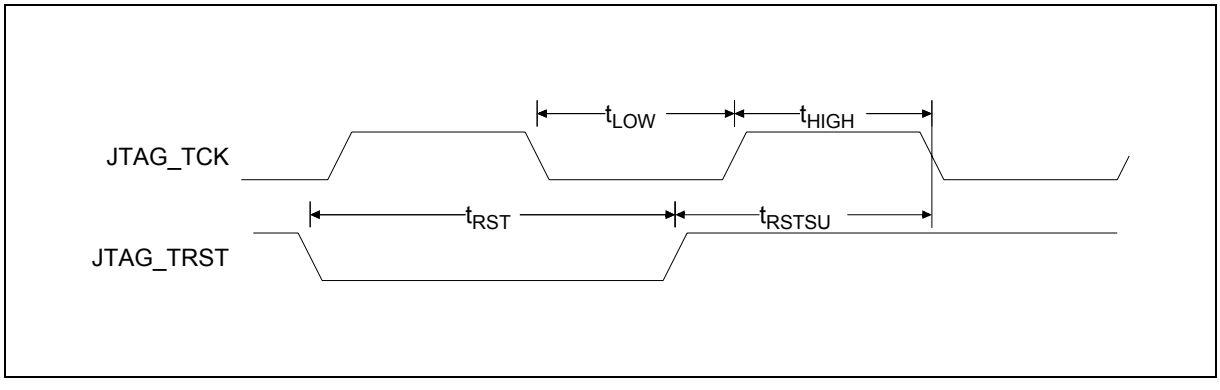


Figure 50 - JTAG Clock and Reset Timing

13.0 Power Characteristics

The following graph in Figure 51 illustrates typical power consumption figures for the ZL50110/1/4 family. Typical characteristics are at 1.8V core, 3.3V I/O, 25°C and typical processing. Power is plotted against the number of active contexts, which is the dominant factor for power consumption.

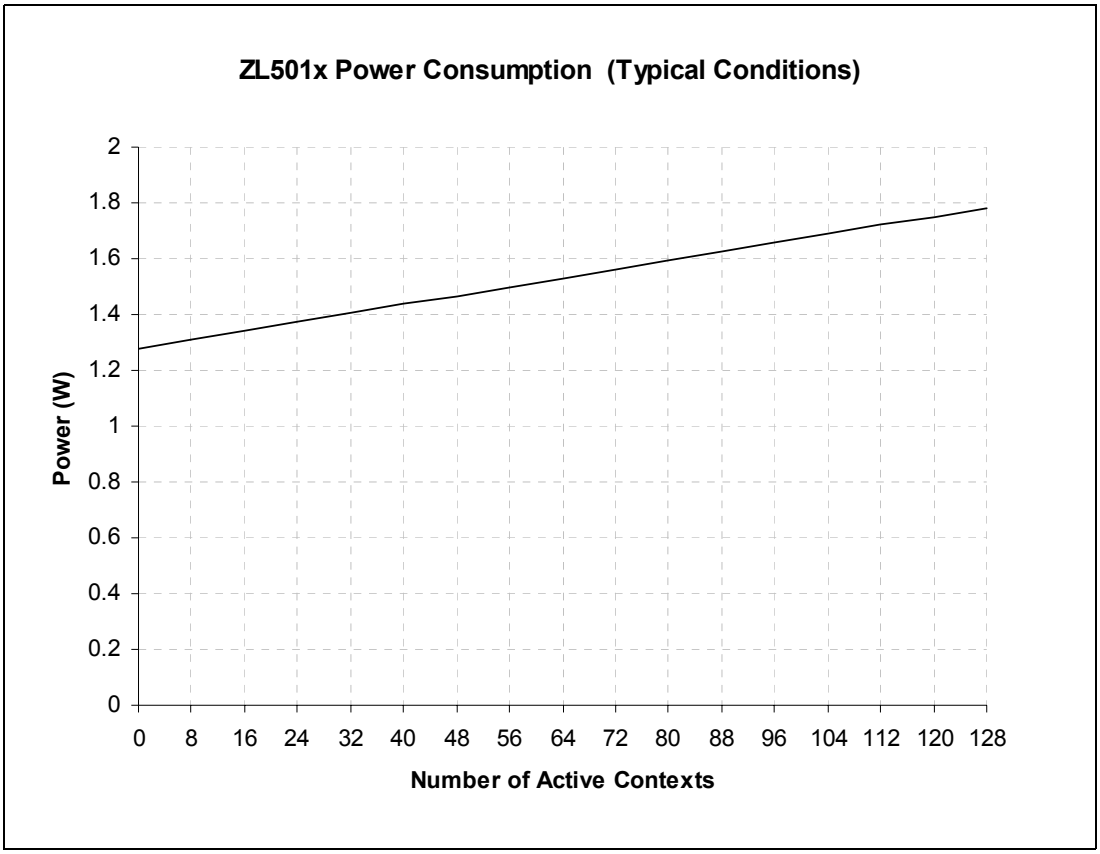


Figure 51 - ZL50110/1/4 Power Consumption Plot

14.0 Design and Layout Guidelines

This guide will provide information and guidance for PCB layouts when using the ZL50110/1/4. Specific areas of guidance are:

- High Speed Clock and Data, Outputs and Inputs
- CPU_TA Output

14.1 High Speed Clock & Data Interfaces

On the ZL50110/1/4 series of devices there are four high-speed data interfaces that need consideration when laying out a PCB to ensure correct termination of traces and the reduction of crosstalk noise. The interfaces being:

- External Memory Interface
- GMAC Interfaces
- TDM Interface
- CPU Interface

It is recommended that the outputs are suitably terminated using a series termination through a resistor as close to the output pin as possible. The purpose of the series termination resistor is to reduce reflections on the line. The value of the series termination and the length of trace the output can drive will depend on the driver output impedance, the characteristic impedance of the PCB trace (recommend 50 ohm), the distributed trace capacitance and the load capacitance. As a general rule of thumb, if the trace length is less than 1/6th of the equivalent length of the rise and fall times, then a series termination may not be required.

the equivalent length of rise time = rise time (ps) / delay (ps/mm)

For example:

Typical FR4 board delay = 6.8ps/mm
Typical rise/fall time for a ZL50110/1/4 output = 2.5ns

critical track length = (1/6) x (2500/6.8) = 61mm

Therefore tracks longer than 61mm will require termination.

As a signal travels along a trace it creates a magnetic field, which induces noise voltages in adjacent traces, this is crosstalk. If the crosstalk is of sufficiently strong amplitude, false data can be induced in the trace and therefore it should be minimised in the layout. The voltage that the external fields cause is proportional to the strength of the field and the length of the trace exposed to the field. Therefore to minimise the effect of crosstalk some basic guidelines should be followed.

First, increase separation of sensitive signals, a rough rule of thumb is that doubling the separation reduces the coupling by a factor of four. Alternatively, shield the victim traces from the aggressor by either routing on another layer separated by a power plane (in a correctly decoupled design the power planes have the same AC potential) or by placing guard traces between the signals usually held ground potential.

Particular effort should be made to minimise crosstalk from ZL50110/1/4 outputs and ensuring fast rise time to these inputs.

In Summary:

- Place series termination resistors as close to the pins as possible.
- Minimise output capacitance.
- Keep common interface traces close to the same length to avoid skew.
- Protect input clocks and signals from crosstalk.

14.1.1 External Memory Interface - special considerations during layout

The timing of address, data and control are all related to the system clock which is also used by the external SSRAM to clock these signals. Therefore the propagation delay of the clock to the ZL50110/1/4 and the SSRAM must be matched to within 250ps, worst case conditions. Trace lengths of theses signals must also be minimised (<100mm) and matched to ensure correct operation under all conditions.

14.1.2 GMAC Interface - special considerations during layout

The GMII interface passes data to and from the ZL50110/1/4 with their related transmit and receive clocks. It is therefore recommended that the trace lengths for transmit related signals and their clock and the receive related signals and their clock are kept to the same length. By doing this the skew between individual signals and their related clock will be minimised.

14.1.3 TDM Interface - special considerations during layout

Although the data rate of this interface is low the outputs edge speeds share the characteristics of the higher data rate outputs and therefore must be treated with the same care extended to the other interfaces with particular reference to the lower stream numbers which support the higher data rates. The TDM interface has numerous clocking schemes and as a result of this the input clock traces to the ZL50110/1/4 devices should be treated with care.

14.1.4 Summary

Particular effort should be made to minimise crosstalk from ZL50110/1/4 outputs and ensuring fast rise time to these inputs.

In Summary:

- Place series termination resistors as close to the pins as possible.
- Minimise output capacitance.
- Keep common interface traces close to the same length to avoid skew.
- Protect input clocks and signals from crosstalk.

14.2 CPU TA Output

The CPU_TA output signal from the ZL50110/1/4 is a critical handshake signal to the CPU that ensures the correct completion of a bus transaction between the two devices. As the signal is critical, it is recommend that the circuit shown in Figure 52 is implemented in systems operating above 40MHz bus frequency to ensure robust operation under all conditions.

- The following external logic is required to implement the circuit:
- 74LCX74 dual D-type flip-flop (one section of two)
- 74LCX08 quad AND gate (one section of four)
- 74LCX125 quad tri-state buffer (one section of four)
- 4K7 resistor x2

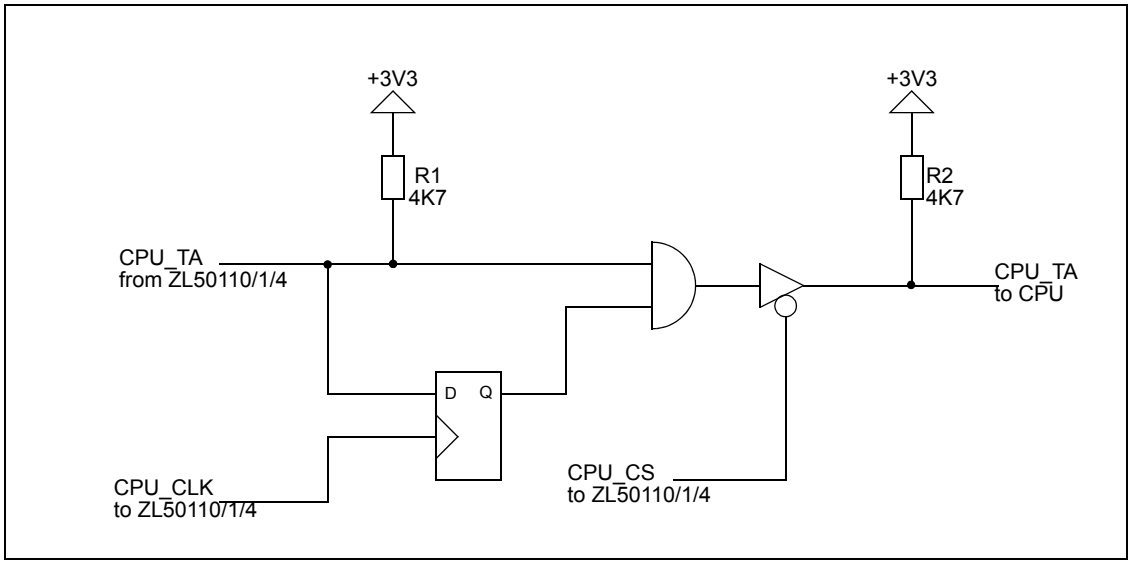


Figure 52 - CPU_TA Board Circuit

The function of the circuit is to extend the TA signal, to ensure the CPU correctly registers it. Resistor R2 must be fitted to ensure correct operation of the TA input to the processor. It is recommended that the logic is fitted close to the ZL50110/1/4 and that the clock to the 74LCX74 is derived from the same clock source as that input to the ZL50110/1/4.

15.0 Reference Documents

15.1 External Standards/Specifications

- IEEE Standard 1149.1-2001; Test Access Port and Boundary Scan Architecture
- IEEE Standard 802.3-2000; Local and Metropolitan Networks CSMA/CD Access Method and Physical Layer
- ECTF H.110 Revision 1.0; Hardware Compatibility Specification
- H-MVIP (GO-MVIP) Standard Release 1.1a; Multi-Vendor Integration Protocol
- MPC8260AEC/D Revision 0.7; Motorola MPC8260 Family Hardware Specification
- RFC 768; UDP
- RFC 791; IPv4
- RFC2460; IPv6
- RFC 1889; RTP
- RFC 2661; L2TP
- RFC 1213; MIB II
- RFC 1757; Remote Network Monitoring MIB (for SMIv1)
- RFC 2819; Remote Network Monitoring MIB (for SMIv2)
- RFC 2863; Interfaces Group MIB
- CCITT G.712; TDM Timing Specification (Method 2)
- G.823; Control of Jitter/Wander with digital networks based on the 2.048Mbit/s hierarchy
- G.824; Control of Jitter/Wander with digital networks based on the 1.544Mbit/s hierarchy
- ANSI T1.101 Stratum 3/4
- Telcordia GR-1244-CORE Stratum 3/4/4e
- IETF's PWE3 draft-ietf-l2tpext-l2tp-base-02
- IETF's PWE3 draft-vainshtein-cesopsn-03
- *Optional Packet Memory Device* - Micron MT55L128L32P1 8Mb ZBT-SRAM

15.2 Zarlink Standards

- MSAN-126 Revision B, Issue 4; ST-BUS Generic Device Specification

15.3 Zarlink ZL50110/1/4 Product Related Documentation

- ZL50110/1/4 Application Note
- ZL50110/1/4 Programmers Model

16.0 Related Products

- MT907x T1/E1/J2 Framer Products (some with combined LIU)
- MH89792/3 E1 Line Interface Unit (LIU)
- ZL50211 Voice Echo Canceller (G.168)
- ZL5001x 512x512 channel Flexible TDM Digital Switch
- MT90866/8/9 MT90870/1 Full range of Flexible Digital Switches
- MVTX260x 24 Port 10/100Mbit/s Ethernet Switch
- MVTX280x 4/8 Port Gigabit Ethernet Switch
- ZL50130 Ethernet Pseudo-Wire across a PSN
- MT9088x TDM to Packet Processor
- MT904x DPLL products (up to Stratum 4 / 4E)
- ZL50418 Ethernet Switches

17.0 Glossary

API Application Program Interface

ATM Asynchronous Transfer Mode

CDP Context Descriptor Protocol (the protocol used by Zarlink's MT9088x family of TDM-Packet devices)

CES Circuit Emulation Services

CESoPSN Circuit Emulation Services over Packet Switched Networks (draft-vainshtein-cesopsn)

CONTEXT A programmed connection of a number of TDM timeslots assembled into a unique packet stream.

CPU Central Processing Unit

DMA Direct Memory Access

DPLL Digital Phase Locked Loop

DSP Digital Signal Processor

GMII Gigabit Media Independent Interface

H.100/H.110 High capacity TDM backplane standards

H-MVIP High-performance Multi-Vendor Integration Protocol (a TDM bus standard)

IETF Internet Engineering Task Force

IP Internet Protocol (version 4, RFC 791, version 6, RFC 2460)

JTAG Joint Test Algorithms Group (generally used to refer to a standard way of providing a board-level test facility)

L2TP Layer 2 Tunneling Protocol (RFC 2661)

LAN Local Area Network

LIU Line Interface Unit

MAC Media Access Control

MII Media Independent Interface

MIB Management Information Base

MPLS Multi Protocol Label Switching

MTIE Maximum Time Interval Error

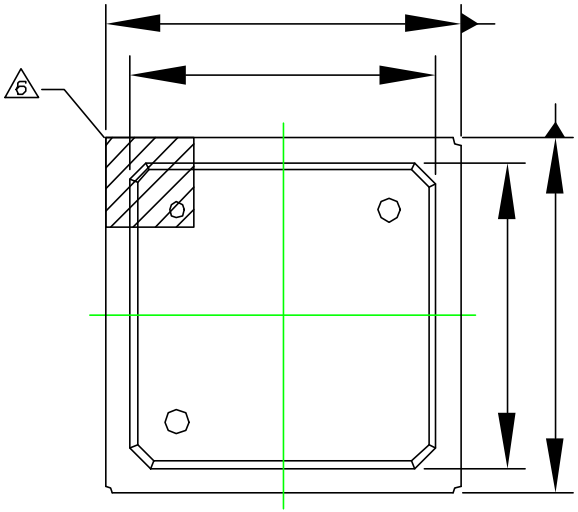
MVIP Multi-Vendor Integration Protocol (a TDM bus standard)

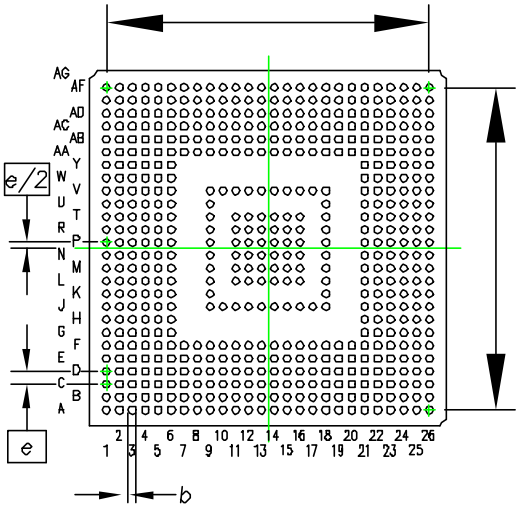
OC3 Optical Carrier 3 - 155.52 Mbit/s leased line

PDH Plesiochronous Digital Hierarchy

PLL Phase Locked Loop

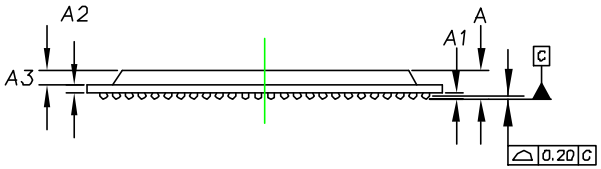
PRS	Primary Reference Source
PRX	Packet Receive
PSTN	Public Switched Telephone Circuit
PTX	Packet Transmit
PWE3	Pseudo-Wire End-to-End Emulation (a working group of the IETF)
QOS	Quality of Service
RTP	Real Time Protocol (RFC 1889)
PE	Protocol Engine
SSRAM	Synchronous Static Random Access Memory
ST BUS	Standard Telecom Bus, a standard interface for TDM data streams
TDL	Tapped Delay Line
TDM	Time Division Multiplexing
UDP	User Datagram Protocol (RFC 768)
UI	Unit Interval
VLAN	Virtual Local Area Network
WFQ	Weighted Fair Queuing
ZBT	Zero Bus Turnaround, a type of synchronous SRAM





SYMBOL	MILLIMETER			INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.13	2.33	2.53	.084	.092	.100
A1	0.50	0.60	0.70	.020	.024	.028
A2	0.51	0.56	0.61	.020	.022	.024
A3	1.12	1.17	1.22	.044	.046	.048
b	0.60	0.75	0.90	.024	.030	.035
D	34.80	35.00	35.20	1.370	1.378	1.386
D1	31.75 BSC			1.250 BSC		
D2	29.90	30.00	30.10	1.177	1.181	1.185
E	34.80	35.00	35.20	1.370	1.378	1.386
E1	31.75 BSC			1.250 BSC		
E2	29.90	30.00	30.10	1.177	1.181	1.185
e	1.27 BSC			.050 BSC		

Confirms to JEDEC MS-034
BAR-2 iss. A



NOTE:
1. ALL DIMENSIONS AND TOLERANCES CONFORM TO ANSI Y14.5M-1982.
2. DIMENSION "b" IS MEASURED AT THE MAXIMUM SOLDER BALL DIAMETER
3. PRIMARY DATUM -C- AND SEATING PLANE ARE DEFINED BY THE SPHERICAL CROWNS OF THE SOLDER BALLS.
4. ALL DIMENSIONS ARE IN MILLIMETERS.
5. NOT TO SCALE.
6. DETAILS OF A1 CORNER ARE OPTIONAL, AND MAY CONSIST OF INK DOT, LASER MARK OR METALISED MARKING, BUT MUST BE LOCATED WITHIN ZONE INDICATED.

© Zarlink Semiconductor 2002 All rights reserved.

ISSUE	1			
ACN	213837			
DATE	12Dec02			
APPRD.				

ZARLINK SEMICONDUCTOR

Previous package codes

BP

Package Code GA

Package Outline for 552 PBGA (35x35)mm, 1.27mm pitch

GPD00809

For more information about all Zarlink products visit our Web Site at

www.zarlink.com

Information relating to products and services furnished herein by Zarlink Semiconductor Inc. trading as Zarlink Semiconductor or its subsidiaries (collectively "Zarlink") is believed to be reliable. However, Zarlink assumes no liability for errors that may appear in this publication, or for liability otherwise arising from the application or use of any such information, product or service or for any infringement of patents or other intellectual property rights owned by third parties which may result from such application or use. Neither the supply of such information or purchase of product or service conveys any license, either express or implied, under patents or other intellectual property rights owned by Zarlink or licensed from third parties by Zarlink, whatsoever. Purchasers of products are also hereby notified that the use of product in certain ways or in combination with Zarlink, or non-Zarlink furnished goods or services may infringe patents or other intellectual property rights owned by Zarlink.

This publication is issued to provide information only and (unless agreed by Zarlink in writing) may not be used, applied or reproduced for any purpose nor form part of any order or contract nor to be regarded as a representation relating to the products or services concerned. The products, their specifications, services and other information appearing in this publication are subject to change by Zarlink without notice. No warranty or guarantee express or implied is made regarding the capability, performance or suitability of any product or service. Information concerning possible methods of use is provided as a guide only and does not constitute any guarantee that such methods of use will be satisfactory in a specific piece of equipment. It is the user's responsibility to fully determine the performance and suitability of any equipment using such information and to ensure that any publication or data used is up to date and has not been superseded. Manufacturing does not necessarily include testing of all functions or parameters. These products are not suitable for use in any medical products whose failure to perform may result in significant injury or death to the user. All products and materials are sold and services provided subject to Zarlink's conditions of sale which are available on request.

Purchase of Zarlink's I^2C components conveys a licence under the Philips I^2C Patent rights to use these components in an I^2C System, provided that the system conforms to the I^2C Standard Specification as defined by Philips.

Zarlink, ZL and the Zarlink Semiconductor logo are trademarks of Zarlink Semiconductor Inc.

Copyright 2003, Zarlink Semiconductor Inc. All Rights Reserved.

TECHNICAL DOCUMENTATION - NOT FOR RESALE