ZL30301

Timing over Packet (ToP) Technology

Data Sheet

September 2005

Features

- Recovers and transmits network synchronization over Ethernet, IP and MPLS Networks
- Output clocks meet ITU-T G.823 and G.824 traffic interface specifications, and ANSI T1.403 timing requirements
- Fully configurable, enabling performance to be tailored to application and network requirements
- Generates outgoing packet reference locked to the TS_CLKi electrical reference clock
- Recovers up to 4 independent clock frequencies from packet streams, in the frequency range 1.544 MHz to 10 MHz
- Average frequency accuracy better than ± 15 ppb
- Supports Master, Slave and Repeater modes of operation
- Supports user defined timing recovery algorithms
- Dual configurable packet interface:
  - Two MII interfaces
  - One MII and one GMII/TBI
- Flexible 32 bit host CPU interface (Motorola PowerQUICC™ 1 and 2 compatible)

Ordering Information

ZL30301GAG 324 PBGA Trays

-40°C to +85°C

- Flexible classification of incoming packets at layers 2, 3, 4 and 5
- Flexible, multi-protocol packet encapsulation, with support for Ethernet, VLAN, IPv4/6, MPLS, L2TPv3, UDP and RTP
- JTAG (IEEE 1149) boundary-scan interface

Applications

- GSM, UMTS air interface synchronization over a packet network
- Circuit Emulation Service over Packets (CESoP), TDM over IP (TDMoIP)
- IP-PBX
- VoIP Gateways
- Video Conferencing
- Broadband Video Distribution

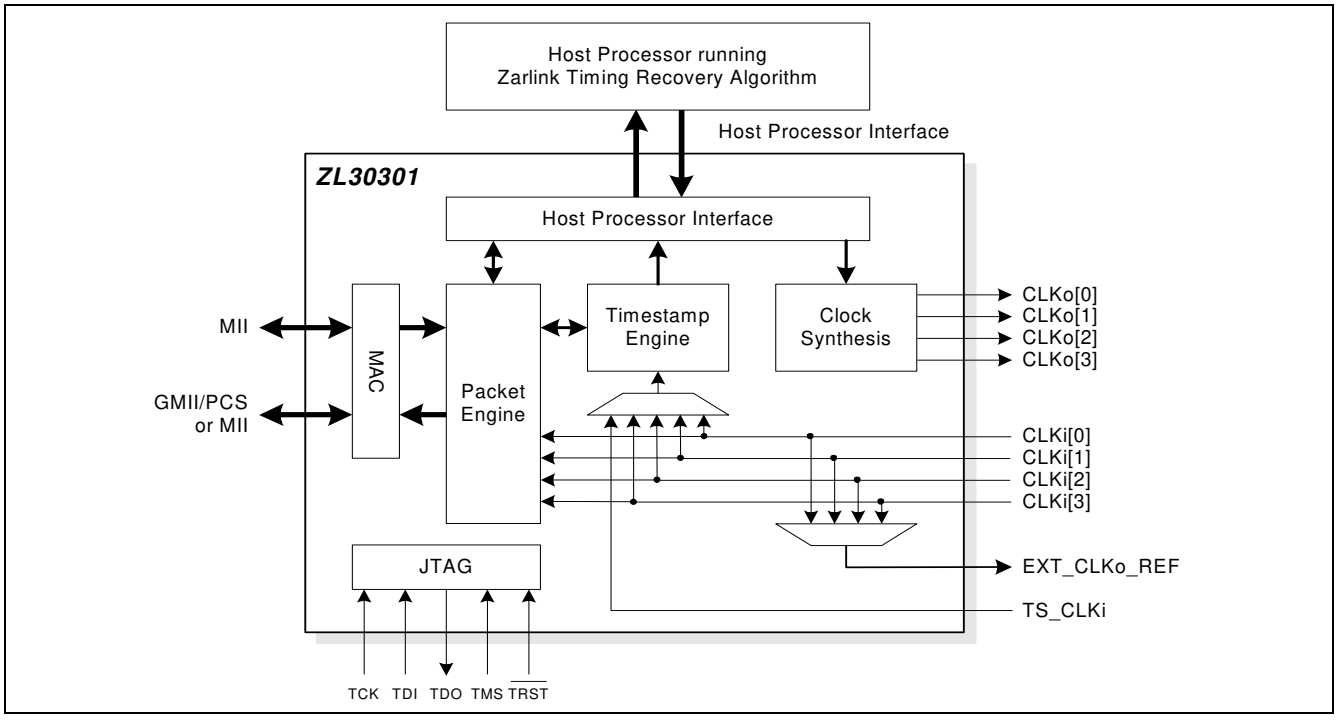


Figure 1 - ZL30301 Functional Block Diagram

Table of Contents

Table of Contents

List of Figures

List of Tables

1.0 Description

Network infrastructures are gradually converging onto an asynchronous packet-based architecture. With this convergence, there are an increasing number of synchronous applications that require accurate timing to be distributed over the packet network. Examples of precision timing sensitive applications that need to transport synchronization over asynchronous packet networks include transport of TDM over packet networks, connections to 2 G and 3 G wireless base stations, Voice over IP, IP PBXs, videoconferencing and broadband video.

Zarlink's Timing over Packet (ToP) technology enables accurate timing and synchronization to be distributed across an asynchronous packet network. This patent-pending technology is implemented in the ZL3030x family of devices, which in combination with the line card microprocessor provide a complete solution for high performance clock synchronization over an asynchronous packet network. The family supports synchronization transfer across both layer 2 and layer 3 networks, using a range of standard protocols including Ethernet, VLAN, MPLS, IP, L2TPv3, UDP and RTP.

The ZL30301 recovers up to 4 independent clocks that are locked to 4 independent references. It receives synchronization information in the form of numbered and time-stamped packets, whose arrival time is cross-referenced to the local clock source. This information is transmitted to the microprocessor, which in turn controls synthesis of the recovered clock.

The ZL30301 algorithm continuously tracks the frequency offset (phase drift) between the clocks located at the master and the slave nodes connected via the packet switched network. This algorithm is tolerant of packet delay variation caused by packet queuing; the precision of the timing recovery depends on statistical properties of the propagation delay of timing packets through the network.

The device is highly configurable to ensure that in the presence of jitter and wander of the reference signals, and short network interruptions, the generated clocks meet the appropriate international standards.

The ZL30301 is designed to maintain average frequency accuracy better than +/-15 ppb with a Stratum 3 quality TCXO system clock and it is tolerant of packet network impairments. However network effects, and the behavior of the sending side of the synchronization link can degrade clock frequency accuracy.

In the event of a failure in the packet network, or the advent of severe congestion preventing or seriously delaying the delivery of timing packets, the ZL30301 will put the recovered clocks into holdover until the flow of timing packets is restored. When the device is in holdover mode the drift of the system clock directly affects the accuracy of the holdover.

The ZL30301 provides the JTAG (Joint Test Action Group) interface.

2.0 Physical Specification

The package for the ZL30301 is a 324-ball PBGA.

Features:

- Body Size: 23 mm x 23 mm (typ)
- Ball Count: 324
- Ball Pitch: 1.00 mm (typ)
- Ball Matrix: 22 x 22
- Ball Diameter: 0.60 mm (typ)
- Total Package Thickness: 2.03 mm (typ)

ZL30301 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
A	VDD_IO	M1_TXEN	M0_TXCLK	M0_RXD[7]	M0_RXD[6]	M0_RXD[4]	M0_COL	M0_GTX_CLK	M0_TXEN	DEVICE_ID[2]	CPU_DATA[28]	CPU_DATA[24]	GND	CPU_DATA[23]	GND	CPU_DATA[19]	CPU_DATA[12]	CPU_DATA[9]	CPU_DATA[8]	CPU_DATA[7]	CPU_SDACK1	VDD_IO
B	M1_TXD[2]	VDD_IO	GND	M1_TXD[0]	M1_TXD[1]	M0_CRS	M0_RXD[0]	M0_RBC1	M0_RBC0	M0_TXER	GND	M0_TXD[5]	M0_TXD[3]	M0_TXD[2]	M1_ACTIVE_LED	CPU_DATA[27]	CPU_DATA[22]	CPU_DATA[20]	CPU_DATA[13]	GND	VDD_IO	CPU_TA
C	M1_TXD[3]	GND	VDD_IO	M1_RXCLK	M1_COL	M1_TXER	M0_RXDV	M0_RXD[3]	M0_RXD[1]	M0_RXCLK	M0_TXD[7]	M0_TXD[4]	M0_TXD[0]	VDD_IO	VDD_IO	CPU_DATA[31]	M1_LINKUP_LED	CPU_DATA[29]	CPU_DATA[26]	VDD_IO	GND	CPU_DREQ1
D	M1_RXD[1]	M1_RXD[0]	M1_RXD[2]	VDD_IO	M1_RXDV	M0_RXER	VDD_IO	M0_RXD[5]	VDD_CORE	M0_RXD[2]	M0_REFCLK	M0_TXD[6]	M0_TXD[1]	VDD_CORE	VDD_CORE	VDD_IO	M0_ACTIVE_LED	VDD_CORE	VDD_IO	CPU_DATA[25]	CPU_ADDR[23]	CPU_DATA[6]
E	M1_RXD[3]	M0_GIGABIT_LED	M1_TXCLK	M1_RXER															CPU_DATA[30]	CPU_DATA[21]	CPU_DATA[15]	CPU_DATA[14]
F	NC	M1_CRS	DEVICE_ID[1]	VDD_CORE															VDD_CORE	CPU_DATA[18]	CPU_DATA[17]	CPU_DATA[16]
G	M_MDIO	DEVICE_ID[0]	M0_LINKUP_LED	VDD_IO															VDD_IO	CPU_IREQ1	CPU_DATA[11]	CPU_DATA[0]
H	M_MDC	GND	NC	VDD_CORE															CPU_DATA[10]	CPU_DATA[1]	CPU_DATA[4]	IC
J	NC	NC	NC	VDD_CORE					GND	GND	GND	GND	GND	GND					VDD_CORE	CPU_DATA[5]	CPU_DATA[3]	CPU_IREQ0
K	NC	NC	NC	VDD_IO					GND	GND	GND	GND	GND	GND					GND	CPU_DATA[2]	IC	CPU_DREQ0
L	GND	CON_L3	CON_L2	VDD_CORE					GND	GND	GND	GND	GND	GND					CPU_CLK	GND	CPU_SDACK2	IC_VDD_IO
M	CLKi[3]	IC	IC	VDD_IO					GND	GND	GND	GND	GND	GND					GND	CPU_TS_ALE	CPU_WE	CPU_OE
N	IC	CLKo[3]	IC	VDD_CORE					GND	GND	GND	GND	GND	GND					VDD_IO	CPU_ADDR[22]	CPU_CS	CPU_ADDR[19]
P	CLKi[2]	GND	VDD_IO	VDD_CORE					GND	GND	GND	GND	GND	GND					VDD_CORE	CPU_ADDR[17]	CPU_ADDR[18]	CPU_ADDR[21]
R	CLKo[2]	IC	CLKi[0]	IC															GND	CPU_ADDR[11]	CPU_ADDR[13]	CPU_ADDR[20]
T	CLKi[1]	CLKo[1]	IC	VDD_IO															VDD_IO	VDD_IO	CPU_ADDR[14]	CPU_ADDR[16]
U	IC	VDD_IO	GND	IC															VDD_CORE	JTAG_TMS	CPU_ADDR[15]	CPU_ADDR[12]
V	IC	CLKo[0]	IC	TS_CLKI															DEVICE_ID[3]	JTAG_TCK	CPU_ADDR[10]	CPU_ADDR[9]
W	IC	IC	IC	VDD_IO	VDD_IO	VDD_CORE	VDD_IO	VDD_IO	VDD_CORE	EXT_CLKo REF	IC_GND	GND	SYSTEM_CLK	VDD_CORE	IC	VDD_IO	IC	DEVICE_ID[4]	VDD_IO	JTAG_TDO	CPU_ADDR[4]	CPU_ADDR[8]
Y	IC	GND	VDD_IO	IC	IC	VDD_CORE	IC	IC	IC	IC	IC_GND	IC	GND	GND	IC	IC	TEST_MODE[1]	JTAG_TRST	IC_GND	VDD_IO	GND	CPU_ADDR[7]
AA	IC	VDD_IO	GND	VDD_IO	VDD_IO	IC	GND	A1VDD_PLL1	IC	IC	SYSTEM_DEBUG	SYSTEM_RST	GPIO[1]	GPIO[2]	GPIO[7]	IC	TEST_MODE[0]	JTAG_TDI	IC_GND	GND	VDD_IO	CPU_ADDR[6]
AB	VDD_IO	IC	IC	IC	GND	IC	IC	IC	GPIO[0]	GPIO[3]	GPIO[4]	GPIO[5]	GPIO[6]	IC	IC	IC	TEST_MODE[2]	IC_GND	CPU_ADDR[2]	CPU_ADDR[3]	CPU_ADDR[5]	VDD_IO

Figure 2 - ZL30301 Package View and Ball Positions

Ball #	ZL30301 Signal Name
A1	VDD_IO
A10	DEVICE_ID[2]
A11	CPU_DATA[28]
A12	CPU_DATA[24]
A13	GND
A14	CPU_DATA[23]
A15	GND
A16	CPU_DATA[19]
A17	CPU_DATA[12]
A18	CPU_DATA[9]
A19	CPU_DATA[8]
A20	CPU_DATA[7]
A21	CPU_SDACK1
A22	VDD_IO
A2	M1_TXEN
A3	M0_TXCLK
A4	M0_RXD[7]
A5	M0_RXD[6]
A6	M0_RXD[4]
A7	M0_COL
A8	M0_GTX_CLK
A9	M0_TXEN
B1	M1_TXD[2]
B10	M0_TXER
B11	GND
B12	M0_TXD[5]
B13	M0_TXD[3]
B14	M0_TXD[2]
B15	M1_ACTIVE_LED
B16	CPU_DATA[27]
B17	CPU_DATA[22]
B18	CPU_DATA[20]
B19	CPU_DATA[13]
B20	GND
B21	VDD_IO
B22	CPU_TA
B2	VDD_IO
B3	GND
B4	M1_TXD[0]
B5	M1_TXD[1]

Table 1 - ZL30301 Ball Signal Assignment

Ball #	ZL30301 Signal Name
B6	M0_CRS
B7	M0_RXD[0]
B8	M0_RBC1
B9	M0_RBC0
C1	M1_TXD[3]
C10	M0_RXCLK
C11	M0_TXD[7]
C12	M0_TXD[4]
C13	M0_TXD[0]
C14	VDD_IO
C15	VDD_IO
C16	CPU_DATA[31]
C17	M1_LINKUP_LED
C18	CPU_DATA[29]
C19	CPU_DATA[26]
C20	VDD_IO
C21	GND
C22	CPU_DREQ1
C2	GND
C3	VDD_IO
C4	M1_RXCLK
C5	M1_COL
C6	M1_TXER
C7	M0_RXDV
C8	M0_RXD[3]
C9	M0_RXD[1]
D1	M1_RXD[1]
D10	M0_RXD[2]
D11	M0_REFCLK
D12	M0_TXD[6]
D13	M0_TXD[1]
D14	VDD_CORE
D15	VDD_CORE
D16	VDD_IO
D17	M0_ACTIVE_LED
D18	VDD_CORE
D19	VDD_IO
D20	CPU_DATA[25]
D21	CPU_ADDR[23]
D22	CPU_DATA[6]

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
D2	M1_RXD[0]
D3	M1_RXD[2]
D4	VDD_IO
D5	M1_RXDV
D6	M0_RXER
D7	VDD_IO
D8	M0_RXD[5]
D9	VDD_CORE
E1	M1_RXD[3]
E19	CPU_DATA[30]
E20	CPU_DATA[21]
E21	CPU_DATA[15]
E22	CPU_DATA[14]
E2	M0_GIGABIT_LED
E3	M1_TXCLK
E4	M1_RXER
F1	NC
F19	VDD_CORE
F20	CPU_DATA[18]
F21	CPU_DATA[17]
F22	CPU_DATA[16]
F2	M1_CRS
F3	DEVICE_ID[1]
F4	VDD_CORE
G1	M_MDIO
G19	VDD_IO
G20	CPU_IREQ1
G21	CPU_DATA[11]
G22	CPU_DATA[0]
G2	DEVICE_ID[0]
G3	M0_LINKUP_LED
G4	VDD_IO
H1	M_MDC
H19	CPU_DATA[10]
H20	CPU_DATA[1]
H21	CPU_DATA[4]
H22	IC
H2	GND
H3	NC
H4	VDD_CORE

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
J1	NC
J10	GND
J11	GND
J12	GND
J13	GND
J14	GND
J19	VDD_CORE
J20	CPU_DATA[5]
J21	CPU_DATA[3]
J22	CPU_IREQ0
J2	NC
J3	NC
J4	VDD_CORE
J9	GND
K1	NC
K10	GND
K11	GND
K12	GND
K13	GND
K14	GND
K19	GND
K20	CPU_DATA[2]
K21	IC
K22	CPU_DREQ0
K2	NC
K3	NC
K4	VDD_IO
K9	GND
L1	GND
L10	GND
L11	GND
L12	GND
L13	GND
L14	GND
L19	CPU_CLK
L20	GND
L21	CPU_SDACK2
L22	IC_VDD_IO
L2	CON_L3
L3	CON_L2

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
L4	VDD_CORE
L9	GND
M1	CLKi[3]
M10	GND
M11	GND
M12	GND
M13	GND
M14	GND
M19	GND
M20	CPU_TS_ALE
M21	CPU_WE
M22	CPU_OE
M2	IC
M3	IC
M4	VDD_IO
M9	GND
N1	IC
N10	GND
N11	GND
N12	GND
N13	GND
N14	GND
N19	VDD_IO
N20	CPU_ADDR[22]
N21	CPU_CS
N22	CPU_ADDR[19]
N2	CLKo[3]
N3	IC
N4	VDD_CORE
N9	GND
P1	CLKi[2]
P10	GND
P11	GND
P12	GND
P13	GND
P14	GND
P19	VDD_CORE
P20	CPU_ADDR[17]
P21	CPU_ADDR[18]
P22	CPU_ADDR[21]

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
P2	GND
P3	VDD_IO
P4	VDD_CORE
P9	GND
R1	CLKo[2]
R19	GND
R20	CPU_ADDR[11]
R21	CPU_ADDR[13]
R22	CPU_ADDR[20]
R2	IC
R3	CLKi[0]
R4	IC
T1	CLKi[1]
T19	VDD_IO
T20	VDD_IO
T21	CPU_ADDR[14]
T22	CPU_ADDR[16]
T2	CLKo[1]
T3	IC
T4	VDD_IO
U1	IC
U19	VDD_CORE
U20	JTAG_TMS
U21	CPU_ADDR[15]
U22	CPU_ADDR[12]
U2	VDD_IO
U3	GND
U4	IC
V1	IC
V19	DEVICE_ID[3]
V20	JTAG_TCK
V21	CPU_ADDR[10]
V22	CPU_ADDR[9]
V2	CLKo[0]
V3	IC
V4	TS_CLKI
W1	IC
W10	EXT_CLKo_REF
W11	IC_GND
W12	GND

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
W13	SYSTEM_CLK
W14	VDD_CORE
W15	IC
W16	VDD_IO
W17	IC
W18	DEVICE_ID[4]
W19	VDD_IO
W20	JTAG_TDO
W21	CPU_ADDR[4]
W22	CPU_ADDR[8]
W2	IC
W3	IC
W4	VDD_IO
W5	VDD_IO
W6	VDD_CORE
W7	VDD_IO
W8	VDD_IO
W9	VDD_CORE
Y1	IC
Y10	IC
Y11	IC_GND
Y12	IC
Y13	GND
Y14	GND
Y15	IC
Y16	IC
Y17	TEST_MODE[1]
Y18	JTAG_TRST
Y19	IC_GND
Y20	VDD_IO
Y21	GND
Y22	CPU_ADDR[7]
Y2	GND
Y3	VDD_IO
Y4	IC
Y5	IC
Y6	VDD_CORE
Y7	IC
Y8	IC
Y9	IC

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
AA1	IC
AA10	IC
AA11	SYSTEM_DEBUG
AA12	SYSTEM_RST
AA13	GPIO[1]
AA14	GPIO[2]
AA15	GPIO[7]
AA16	IC
AA17	TEST_MODE[0]
AA18	JTAG_TDI
AA19	IC_GND
AA20	GND
AA21	VDD_IO
AA22	CPU_ADDR[6]
AA2	VDD_IO
AA3	GND
AA4	VDD_IO
AA5	VDD_IO
AA6	IC
AA7	GND
AA8	A1VDD_PLL1
AA9	IC
AB1	VDD_IO
AB10	GPIO[3]
AB11	GPIO[4]
AB12	GPIO[5]
AB13	GPIO[6]
AB14	IC
AB15	IC
AB16	IC
AB17	TEST_MODE[2]
AB18	IC_GND
AB19	CPU_ADDR[2]
AB20	CPU_ADDR[3]
AB21	CPU_ADDR[5]
AB22	VDD_IO
AB2	IC
AB3	IC
AB4	IC
AB5	GND

Table 1 - ZL30301 Ball Signal Assignment (continued)

Ball #	ZL30301 Signal Name
AB6	IC
AB7	IC
AB8	IC
AB9	GPIO[0]

Table 1 - ZL30301 Ball Signal Assignment (continued)

NC - Not Connected - leave open circuit.
IC - Internally Connected - leave open circuit.
IC_GND - Internally Connected - tie to ground
IC_VDD_IO - Internally Connected - tie to VDD_IO
CON_L2 - connect to ball L2
CON_L3 - connect to ball L3

3.0 External Interface Description

The following key applies to all tables:

I	Input
O	Output
D	Internal 100 kΩ pull-down resistor present
U	Internal 100 kΩ pull-up resistor present
T	Tri-state Output

3.1 Clock Interface

All Clock Interface signals are 5 V tolerant.

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

All Clock Interface inputs have internal pull-down resistors so they can be safely left unconnected if not used.

Signal	I/O	Package Balls	Description
CLKi[3:0]	I D	[3] M1 [2] P1 [1] T1 [0] R3	Clock inputs. In master mode these input are connected to CLKo[3:0], respectively. In slave mode these inputs can be connected to CLKo[3:0] for some applications. These inputs accepts frequencies from 8 kHz up to 34.368 MHz
CLKo[3:0]	OT	[3] N2 [2] R1 [1] T2 [0] V2	Clock outputs. In slave mode, these are the recovered clocks from the packet network. These clocks may be connected to CLKi[3:0] for some applications, e.g., selection of a single output clock, or selection of a timestamp reference for a repeater function. In master mode these outputs are connected to CLKi[3:0], respectively. The recovered clocks are 8 kHz multiples in the range 1.544 MHz to 10 MHz.
TS_CLKi	I D	V4	Time Stamp clock input. In master mode and/or repeater mode, this input is multiplexed with the four input clock signals CLKi[3:0] to select the clock to be distributed over the packet network. It is used as the timestamp clock, or as a reference clock input for differential clock recovery. Acceptable frequency range: 1.544 MHz - 10 MHz (timestamp clock) or 4 MHz to 40 MHz (differential reference clock).

Table 2 - Clock Interface Pin Definition

Signal	I/O	Package Balls	Description
EXT_CLKo_REF	OT	W10	Multiplexed output signal, selecting one of the clock inputs, CLKi[3:0]. The source clock may be frequency divided internally prior to output on EXT_CLKo_REF pin. Expected frequency range: 8 kHz - 10 MHz.

Table 2 - Clock Interface Pin Definition

Note: All Clock Interface inputs have internal pull-down resistors so they can be safely left unconnected if not used.

3.2 Packet Interfaces

The ZL30301 packet interface features either 2 MII interfaces, or 1 MII and 1 GMII interfaces, or 1 MII and 1 TBI (1000 Mbps) interfaces. The TBI interface is a PCS interface supported by an integrated 1000BASE-X PCS module.

Data for all three types of packet switching is based on Specification IEEE Std. 802.3 - 2000. Only Port 0 has the 1000 Mbps capability necessary for the GMII/TBI interface.

Table 3 maps the signal pins used in the MII interface to those used in the GMII and TBI interface. Table 4 through Table 6 show all the pins and their related package balls, based on the GMII/MII configuration.

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

MII	GMII	TBI (PCS)
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 3 - Packet Interface Signal Mapping - MII to GMII/TBI

Note: M*n* can be either M0 or M1 for ZL30301.

Signal	I/O	Package Balls	Description
M_MDC	O	H1	MII management data clock. Common for both MII ports. It has a minimum period of 400 ns (maximum frequency 2.5 MHz), and is independent of the TXCLK and RXCLK.
M_MDIO	ID/ OT	G1	MII management data I/O. Common for both MII ports at up to 2.5 MHz. It is bi-directional between the ZL30301 and the Ethernet station management entity. Data is passed synchronously with respect to M_MDC.

Table 4 - MII Management Interface Package Ball Definition

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_LINKUP_LED	O	G3	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	D17	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	E2	LED drive for MAC 0 to indicate operation at Gbps. Logic 0 output = LED on Logic 1 output = LED off
M0_REFCLK	I D	D11	**GMII/TBI** - Reference Clock input at 125 MHz. 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.
M0_RXCLK	I U	C10	GMII/MII - M0_RXCLK. Accepts the following frequencies: 25.0 MHz MII 100 Mbps 125.0 MHz GMII 1 Gbps
M0_RBC0	I U	B9	TBI - M0_RBC0. Used as a clock when in TBI mode. Accepts 62.5 MHz and is 180°C out of phase with M0_RBC1. Receive data is clocked at each rising edge of M0_RBC1 and M0_RBC0, resulting in 125 MHz sample rate.

Table 5 - MII Port 0 Interface Package Ball Definition

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_RBC1	I U	B8	TBI - M0_RBC1 Used as a clock when in TBI mode. Accepts 62.5 MHz, 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 125 MHz sample rate.
M0_COL	I D	A7	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] A4 [3] C8 [6] A5 [2] D10 [5] D8 [1] C9 [4] A6 [0] B7	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 (TBI).
M0_RXDV / M0_RXD[8]	I D	C7	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. TBI - M0_RXD[8] Receive Data. Clocked on the rising edges of M0_RBC0 and M0_RBC1.
M0_RXER / M0_RXD[9]	I D	D6	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. TBI - M0_RXD[9] Receive Data. Clocked on the rising edges of M0_RBC0 and M0_RBC1.
M0_CRS / M0_Signal_Detect	I D	B6	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. TBI - M0_Signal Detect Similar function to M0_CRS.
M0_TXCLK	I U	A3	**MII only** - Transmit Clock Accepts the following frequencies: 25.0 MHz MII 100 Mbps

Table 5 - MII Port 0 Interface Package Ball Definition (continued)

MII Port 0			
Signal	**I/O**	**Package Balls**	**Description**
M0_TXD[7:0]	O	[7] C11 [3] B13 [6] D12 [2] B14 [5] B12 [1] D13 [4] C12 [0] C13	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/TBI).
M0_TXEN / M0_TXD[8]	O	A9	GMII/MII - M0_TXEN Transmit Enable. Asserted when the MAC has data to transmit, synchronously to M0_TXCLK with the first preamble of the packet to be sent. Remains asserted until the end of the packet transmission. Active high. TBI - M0_TXD[8] Transmit Data. Clocked on rising edge of M0_GTXCLK.
M0_TXER / M0_TXD[9]	O	B10	GMII/MII - M0_TXER Transmit Error. Transmitted synchronously with respect to M0_TXCLK, and active high. When asserted (with M0_TXEN also asserted) the ZL30301 will transmit a non-valid symbol, somewhere in the transmitted frame. TBI - M0_TXD[9] Transmit Data. Clocked on rising edge of M0_GTXCLK.
M0_GTX_CLK	O	A8	**GMII/TBI only** - Gigabit Transmit Clock Output of a clock for Gigabit operation at 125 MHz.

Table 5 - MII Port 0 Interface Package Ball Definition (continued)

MII Port 1			
Signal	**I/O**	**Package Balls**	**Description**
M1_LINKUP_LED	O	C17	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	B15	LED drive for MAC 1 to indicate port is transmitting or receiving packet data. Logic 0 output = LED on Logic 1 output = LED off
M1_RXCLK	I U	C4	MII only - Receive Clock. Accepts the following frequencies: 25.0 MHz MII 100 Mbps

Table 6 - MII Port 1 Interface Package Ball Definition

MII Port 1			
Signal	**I/O**	**Package Balls**	**Description**
M1_COL	I D	C5	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.
M1_RXD[3:0]	I U	[3] E1 [1] D1 [2] D3 [0] D2	Receive Data. Clocked on rising edge of M1_RXCLK.
M1_RXDV	I D	D5	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.
M1_RXER	I D	E4	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.
M1_CRS	I D	F2	Carrier Sense. This asynchronous signal is asserted when either the transmission or reception device is non-idle. It is active high.
M1_TXCLK	I U	E3	MII only - Transmit Clock Accepts the following frequencies: 25.0 MHz MII 100 Mbps
M1_TXD[3:0]	O	[3] C1 [1] B5 [2] B1 [0] B4	Transmit Data. Clocked on rising edge of M1_TXCLK.
M1_TXEN	O	A2	Transmit Enable. Asserted when the MAC has data to transmit, synchronously to M1_TXCLK with the first preamble of the packet to be sent. Remains asserted until the end of the packet transmission. Active high.
M1_TXER	O	C6	Transmit Error. Transmitted synchronously with respect to M1_TXCLK, and active high. When asserted (with M1_TXEN also asserted) the ZL30301 will transmit a non-valid symbol, somewhere in the transmitted frame.

Table 6 - MII Port 1 Interface Package Ball Definition (continued)

3.3 CPU Interface

All CPU Interface signals are 5 V 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] C16 [30] E19 [29] C18 [28] A11 [27] B16 [26] C19 [25] D20 [24] A12 [23] A14 [22] B17 [21] E20 [20] B18 [19] A16 [18] F20 [17] F21 [16] F22 [15] E21 [14] E22 [13] B19 [12] A17 [11] G21 [10] H19 [9] A18 [8] A19 [7] A20 [6] D22 [5] J20 [4] H21 [3] J21 [2] K20 [1] H20 [0] G22	CPU Data Bus. Bi-directional data bus, synchronously transmitted with CPU_CLK rising edge. NOTE: as with all ports in the ZL30301 device, CPU_DATA[0] is the least significant bit (lsb).
CPU_ADDR[23:2]	I	[23] D21 [22] N20 [21] P22 [20] R22 [19] N22 [18] P21 [17] P20 [16] T22 [15] U21 [14] T21 [13] R21 [12] U22 [11] R20 [10] V21 [9] V22 [8] W22 [7] Y22 [6] AA22 [5] AB21 [4] W21 [3] AB20 [2] AB19	CPU Address Bus. Address input from processor to ZL30301, synchronously transmitted with CPU_CLK rising edge. NOTE: as with all ports in the ZL30301 device, CPU_ADDR[2] is the least significant bit (lsb).
$\overline{\text{CPU\_CS}}$	I U	N21	CPU Chip Select. Synchronous to rising edge of CPU_CLK and active low. Is asserted with CPU_TS_ALE. Must be asserted with $\overline{\text{CPU\_OE}}$ to asynchronously enable the CPU_DATA output during a read, including DMA read.
$\overline{\text{CPU\_WE}}$	I	M21	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 ZL30301. Asserted one clock cycle after CPU_TS_ALE.

Table 7 - CPU Interface Package Ball Definition

Signal	I/O	Package Balls	Description
CPU_OE	I	M22	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 ZL30301. 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	M20	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	A21	CPU/DMA 1 Acknowledge Input. Active low synchronous to CPU_CLK rising edge. Used to acknowledge request from ZL30301 for a DMA write transaction. Only used for DMA transfers, not for normal register access.
CPU_SDACK2	I	L21	CPU/DMA 2 Acknowledge Input Active low synchronous to CPU_CLK rising edge. Used to acknowledge request from ZL30301 for a DMA read transaction. Only used for DMA transfers, not for normal register access.
CPU_CLK	I	L19	CPU PowerQUICC™ II Bus Interface clock input. 66 MHz clock, with minimum of 6 ns high/low time. Used to time all host interface signals into and out of ZL30301 device.
CPU_TA	OT	B22	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 ZL30301 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.

Table 7 - CPU Interface Package Ball Definition (continued)

Signal	I/O	Package Balls	Description
$\overline{\text{CPU\_DREQ0}}$	OT	K22	CPU DMA 0 Request Output Active low synchronous to CPU_CLK rising edge. Asserted by ZL30301 to request the host initiates a DMA write. Only used for DMA transfers, not for normal register access.
$\overline{\text{CPU\_DREQ1}}$	OT	C22	CPU DMA 1 Request Active low synchronous to CPU_CLK rising edge. Asserted by ZL30301 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	J22	CPU Interrupt 0 Request (Active Low)
$\overline{\text{CPU\_IREQ1}}$	O	G20	CPU Interrupt 1 Request (Active Low)

Table 7 - CPU Interface Package Ball Definition (continued)

3.4 System Function Interface

All System Function Interface signals are 5 V 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	W13	System Clock Input. The system clock frequency is 100 MHz.
SYSTEM_RST	I	AA12	System Reset Input. Active low. The system reset is asynchronous, and causes all registers within the ZL30301 to be reset to their default state.
SYSTEM_DEBUG	I	AA11	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 8 - System Function Interface Package Ball Definition

3.5 Test Facilities

3.5.1 Administration, Control and Test Interface

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

Signal	I/O	Package Balls	Description
GPIO[7:0]	ID/ OT	[7] AA15 [6] AB13 [5] AB12 [4] AB11 [3] AB10 [2] AA14 [1] AA13 [0] AB9	General Purpose I/O pins. Connected to an internal register, so customer can set user-defined parameters.
TEST_MODE[2:0]	I D	[2] AB17 [1] Y17 [0] AA17	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 9 - Administration/Control Interface Package Ball Definition

3.5.2 JTAG Interface

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

Signal	I/O	Package Balls	Description
$\overline{\text{JTAG\_TRST}}$	I U	Y18	JTAG Reset. Asynchronous reset. In normal operation this pin should be pulled low.
JTAG_TCK	I	V20	JTAG Clock - maximum frequency is 25 MHz, typically run at 10 MHz. In normal operation this pin should be pulled either high or low.
JTAG_TMS	I U	U20	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	AA18	JTAG test data input. Synchronous to JTAG_TCK.
JTAG_TDO	O	W20	JTAG test data output. Synchronous to JTAG_TCK.

Table 10 - JTAG Interface Package Ball Definition

3.6 Miscellaneous Inputs

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

Signal	Package Balls	Description
IC_GND	W11, Y11, Y19, AA19, AB18	Internally connected. Tie to GND.
IC_VDD_IO	L22	Internally connected. Tie to VDD_IO.

Table 11 - Miscellaneous Inputs Package Ball Definitions

3.7 Power and Ground Connections

Signal	Package Balls				Description
VDD_IO	A1	A22	AA2	AA21	3.3 V VDD Power Supply for IO Ring
	AA4	AA5	AB1	AB22	
	B2	B21	C14	C15	
	C20	C3	D16	D19	
	D4	D7	G19	G4	
	K4	M4	N19	P3	
	T19	T20	T4	U2	
	W16	W19	W4	W5	
	W7	W8	Y20	Y3	
GND	A13	A15	AA20	AA3	0 V Ground Supply
	AA7	AB5	B11	B20	
	B3	C2	C21	H2	
	J10	J11	J12	J13	
	J14	J9	K10	K11	
	K12	K13	K14	K19	
	K9	L1	L10	L11	
	L12	L13	L14	L20	
	L9	M10	M11	M12	
	M13	M14	M19	M9	
	N10	N11	N12	N13	
	N14	N9	P10	P11	
	P12	P13	P14	P2	
	P9	R19	U3	W12	
	Y13	Y14	Y2	Y21	
VDD_CORE	D14	D15	D18	D9	1.8 V VDD Power Supply for Core Region
	F19	F4	H4	J19	
	J4	L4	N4	P19	
	P4	U19	W14	W6	
	W9	Y6			
A1VDD	AA8				1.8 V PLL Power Supply

Table 12 - Power and Ground Package Ball Definition

3.8 Internal Connections

The following pins are connected internally, and must be left open circuit.

Signal	Package Balls				Description
IC	AA1	AA10	AA16	AA6	Internally connected. Leave open circuit
	AA9	AB14	AB15	AB16	
	AB2	AB3	AB4	AB6	
	AB7	AB8	H22	K21	
	M2	M3	N1	N3	
	R2	R4	T3	U1	
	U4	V1	V3	W1	
	W17	W2	W3	W15	
	Y1	Y9	Y10	Y12	
	Y15	Y16	Y4	Y5	
	Y7	Y8			

Table 13 - Internal Connections Package Ball Definitions

The following pins must be connected together.

Signal	Package Balls	Description
CON_L3	L2	Connect to ball L3. Balls L2 and L3 perform a loopback, and should be connected only to each other
CON_L2	L3	Connect to ball L2. See L2.

Table 14 - Internal Connections Package Ball Definitions

3.9 No Connections

The following pins are not connected internally, and should be left open circuit.

Signal	Package Balls				Description
NC	F1	H3	J1	J2	No connection. Leave open circuit.
	J3	K1	K2	K3	

Table 15 - Miscellaneous Inputs Package Ball Definitions

3.10 Device ID

Signal	I/O	Package Balls	Description
DEVICE_ID[4:0]	O	[4] W18 [3] V19 [2] A10 [1] F3 [0] G2	Device ID. ZL30300 = 00100 ZL30301 = 10001 ZL30302 = 00110

Table 16 - Device ID Ball Definition

4.0 Typical Applications

Many carriers are now beginning the process of moving their networks over to a packet-based structure. This breaks the circuit-switched nature of the telecommunications network, and divorces the delivery of data from the delivery of timing and synchronization. However, there are many applications which still require accurate timing and synchronization, including:

- Circuit Emulation Service over packets, TDM over IP
- GSM, UMTS air interface synchronization over a packet network
- IP-PBX
- VoIP Gateways
- Video Conferencing
- Broadband Video Distribution

4.1 Edge of the PSN

There are a wide variety of applications and equipment that require synchronization, whether for voice, video or data. Figure 3 is a representation of a few different situations where synchronization from a PRS (primary reference source) is required to be carried across a packet network to its outer edges. At the PRS a ZL30301 is used to encode timing information. This timing is routed through the packet network to the boundaries at the outer edges of the PSN (packet-switched network). At the edge of the PSN another ZL30301 is used to regenerate or recover the timing.

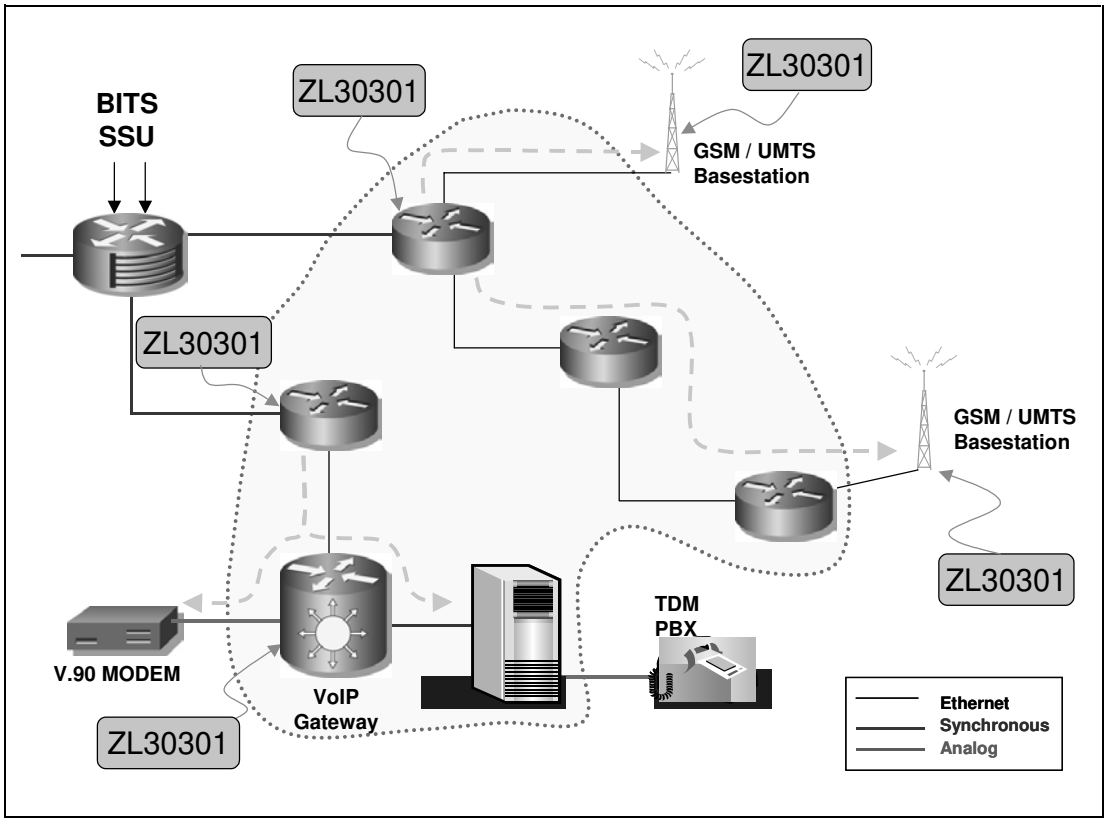


Figure 3 - Edge of the PSN

Unlike VoIP, fax and modem connections are not tolerant of buffer slips or a large number of data errors. A ZL30301 is used to synchronize the fax/modem inter-working functions to ensure no buffer slips.

A second application is legacy PBX support. Using TDM pseudo-wires or Circuit Emulation Services over Packet (CESoP) the T1/E1 interface trunk to the PBX may be carried across a PSN. A ZL30301 may be used in this case to synchronize both ends of the CESoP connection to ensure the T1/E1 line meets the required ITU-T and ANSI timing and synchronization specifications.

A third application is for VoIP. Traditionally VoIP did not put much emphasis on timing and synchronization in the gateway. It is becoming more important, for good voice quality, to reduce buffer slips by synchronizing the VoIP gateway. A ZL30301 is a perfect fit here, as well.

4.2 Wireless Access Applications

Traditionally within the UMTS Terrestrial Radio Access Network the Node B (basestation) is connected to the Radio Network Controller (RNC) through a T1/E1 link. The remote base stations must remain synchronized to the network, and synchronization is also derived from the T1/E1 link. When this link is replaced by a packet network, an alternative means of synchronization must be provided. Current generation wireless base stations often meet synchronization requirements through GPS clocks when PRS traceable network references are not available.

The ZL30301 can replace expensive parts such as GPS, distributing the clock over the packet network from the RNC. The transmit frequency must be maintained at a highly reliable frequency, within +/- 15 ppb from the master clock. This is because the clock is used to generate the radio signals for the air interface, and frequency deviations will cause interference with adjacent cell sites. The master clock can be distributed to the wireless network to maintain all nodes in complete synchronicity. Figure 2 depicts an example of such Wireless Infrastructure.

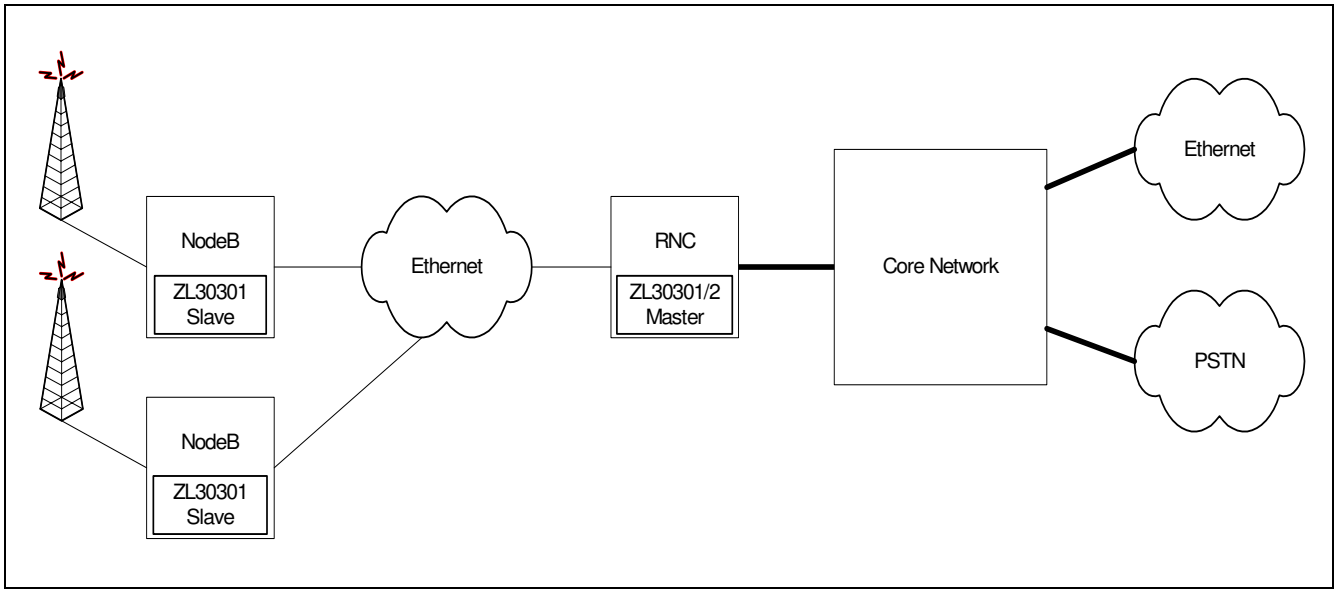


Figure 4 - Example of Wireless Infrastructure

5.0 Functional Description

5.1 Modes of Operation

The ZL30301 can operate in three primary modes:

- as a timing master
- as a timing slave
- as a timing repeater

Figure 5 shows an application diagram of the ZL30301 operating in Master, Slave or in a Timing Repeater Mode.

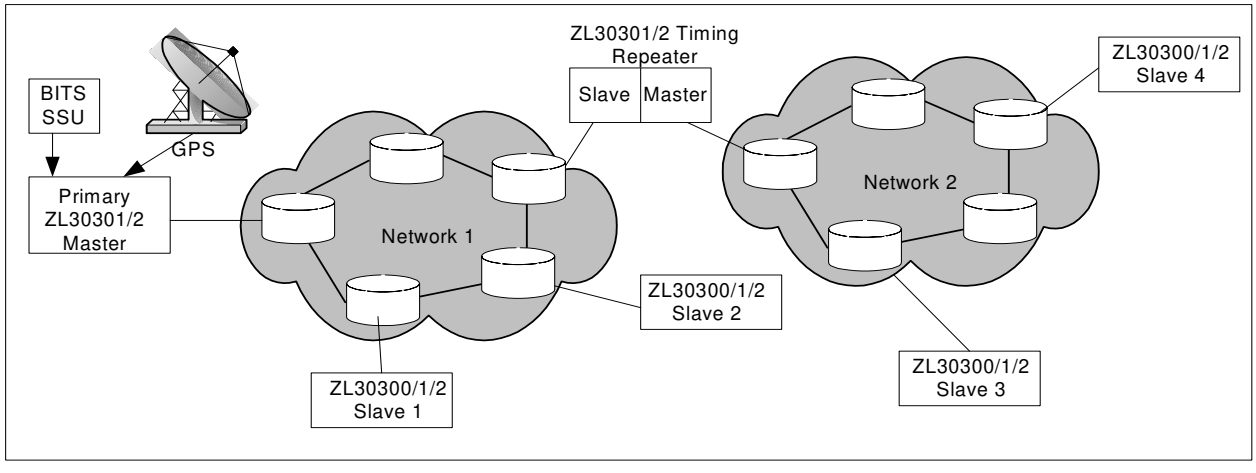


Figure 5 - ZL30301 Operating Modes

5.1.1 Master Mode of Operation

The ZL30301 is capable of transmitting network synchronization over Ethernet, IP and MPLS Networks. It may generate streams of packets in the required format, referenced to a master clock in the frequency range 1.544 MHz to 10 MHz. These packets may be either broadcast to all devices in the network, multicast to a number of selected devices (i.e., those in the addressed multicast group), or unicast to up to five separate slave devices. In unicast mode, the connections can be differentiated by address or port number, e.g., IP destination address (Unicast or Multicast), MPLS inner label, UDP port number, VLAN ID, etc.

The master reference clock is connected to the timestamp input, TS_CLKi. All outgoing timing packets are timestamped from this clock. TS_CLKi accepts clocks from 1.544 MHz up to 10 MHz, but for better resolution the higher clock rates are recommended.

In master operation, the DCOs are used to govern the rate of the packet streams. Therefore the DCO outputs need to be looped back into clock inputs of the device as follows: CLKo[0] into CLKi[0], CLKo[1] into CLKi[1], CLKo[2] into CLKi[2], and CLKo[3] into CLKi[3] (see Figure 6). Note that a further DCO is available internally to the device, to drive the fifth unicast packet stream. Typical packet rates are in the range 30 - 100 packets per second.

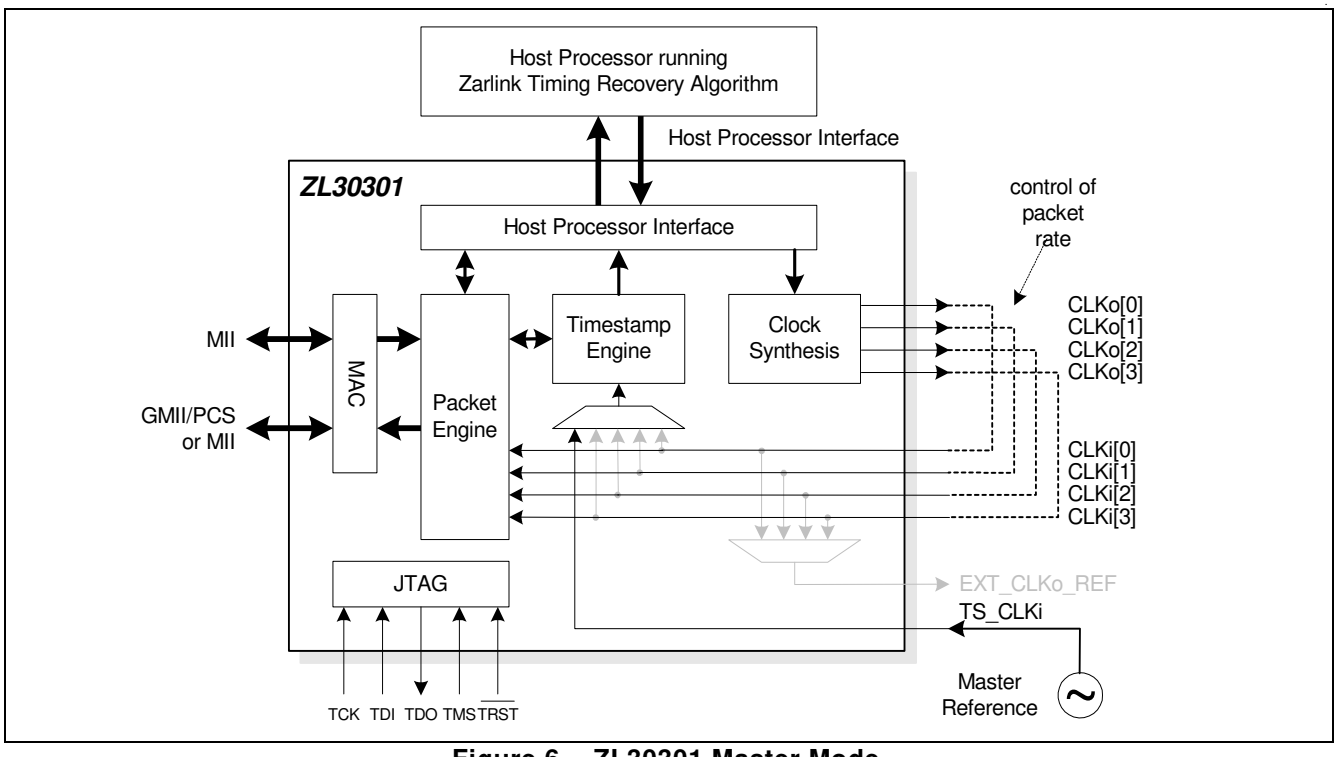


Figure 6 - ZL30301 Master Mode

5.1.2 Slave Mode of Operation

In slave mode, the ZL30301 can recover up to 4 independent clocks locked to 4 separate master clocks in the frequency range 1.544 MHz to 10 MHz. This may be utilized as part of a redundancy strategy, to minimize the effect of failure of a master clock or its distribution path. It is designed to maintain average frequency accuracy better than +/-15 ppb with a Stratum 3 quality TCXO system clock and it is tolerant of packet network impairments. However network effects, and the behavior of the sending side of the synchronization link can degrade clock frequency accuracy.

The device is able to recover clocks from packet streams encoded in any of the following standardized formats:

1. “Timing over Packet” (ToP) streams with the packet header format Ethernet/IP/UDP/RTP[1].
2. Standard CES data streams in one of the following formats:
   - ITU-T Recommendation Y.1413, March 2004
   - IETF, draft-ietf-pwe3-satop-02, work in progress, July 2005
   - IETF, draft-ietf-pwe3-cesopsn-03, work in progress, July 2005
   - IETF, draft-iettf-pwe3-tdmoip-03, work in progress, February 2005 (unstructured data transfer only)
   - Metro Ethernet Forum, MEF 8, November 2004
   - MPLS Forum, MFA 8.0.0, November 2004

1. Format could also be Ethernet/IP/L2TP/RTP, or Ethernet/MPLS/MPLS/RTP to suit different types of packet switched networks.

One of the following methods is used to recover the clock. These methods are described in Section 5.3:

- Adaptive clock recovery based on RTP timestamp (standard method for ToP streams)
- Adaptive clock recovery based on sequence number (used for CES data streams where there may not be a timestamp. Since the packets are constant length, an effective timestamp is calculated based on the sequence number and the length of the constant-bit-rate payload)
- Differential clock recovery based on RTP timestamp (used for CES data streams where a timestamp is provided relative to a known reference clock which is available at both master and slave devices)

5.1.3 Timing Repeater Mode of Operation

The ZL30301 can also function as a timing repeater. This feature is very useful if there is a need for the synchronization information to be transmitted over a large network. Not only is the volume of timing packets reduced around the timing master device, reducing congestion, but the quality of the recovered clock is improved by breaking the trail through the packet network into two shorter segments.

For example, in Figure 5, at the repeater node the ZL30301 will work as a slave node for the Network 1 and as a master node for the Network 2. The device is simultaneously operates in both Master and Slave modes to achieve the repeater function. Figure 7 shows the detailed connection for the ZL30301 operating in Repeater Timing mode. CLKo[0] is the clock recovered from the incoming packet stream. This is used to timestamp the outgoing packets sent to the subsequent slave devices. CLKo[1] - CLKo[3] are used to control the rate of the outgoing packet streams.

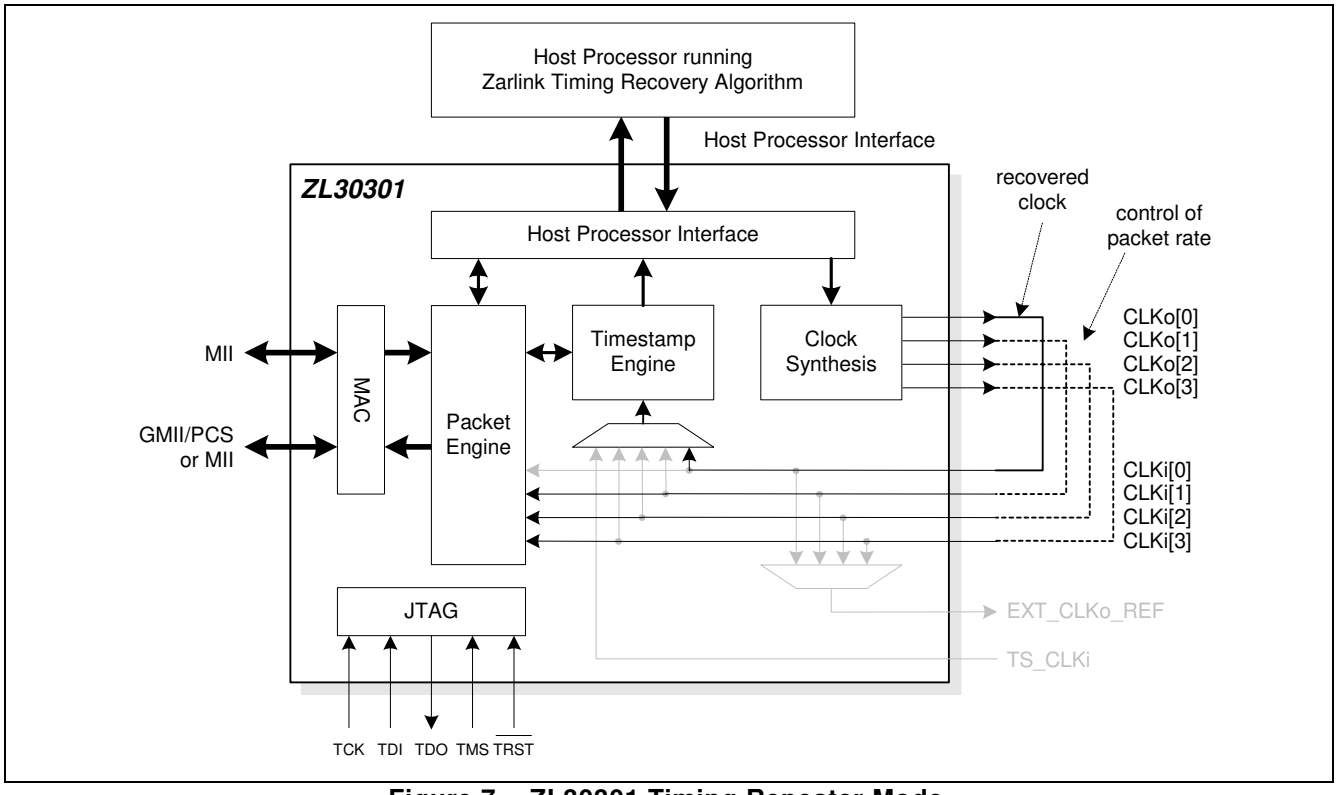


Figure 7 - ZL30301 Timing Repeater Mode

5.2 Timing Redundancy

The ZL30301 can recover clocks from up to four separate packet timing sources. The on-chip multiplexer can be used to select the required clock, routing this to the common “EXT_CLKo_REF” pin. There is also a second internal multiplexer which can select the clock to be used as the timestamp source in the case of a repeater function.

Various statistics on the status or the quality of the recovered clocks are available on which to base the choice of clock (see Section 5.7). However, it should be noted that a phase transient may be generated when switching over between recovered clocks. An external PLL with hitless reference switching should be used if it is important to avoid the phase hit.

5.3 Clock Recovery

The ZL30301 supports clock recovery from up to four individual timing packet streams. There are two clock recovery schemes which can employed, depending on the availability of a common reference clock at both master and slave nodes - adaptive and differential. In some applications these schemes may be used in combination, where a ToP stream itself is used to distribute the common reference clock to the slave node. The clock recovery algorithm is performed by software in the external processor, with support from on-chip hardware to gather the required statistics.

5.3.1 Adaptive Clock Recovery

For applications where there is no common reference clock available at both master and slave nodes, an adaptive clock recovery technique is used. This infers the clock rate of the original service clock from information about the arrival times of the timing packets at the slave, and the original launch times of the timing packets from the master. The advantage of this scheme is that there is no need for a common reference clock at both end of the network.

Typical adaptive clock recovery algorithms use averaging to calculate the frequency of the original clock source. However, low frequency variations in the delay of packets through the network may be fed through as wander in the recovered clock. Zarlink has developed a superior method of adaptive clock recovery using patent-pending algorithms and heuristics to overcome the issue, and identify other disruptive events seen in typical packet networks.

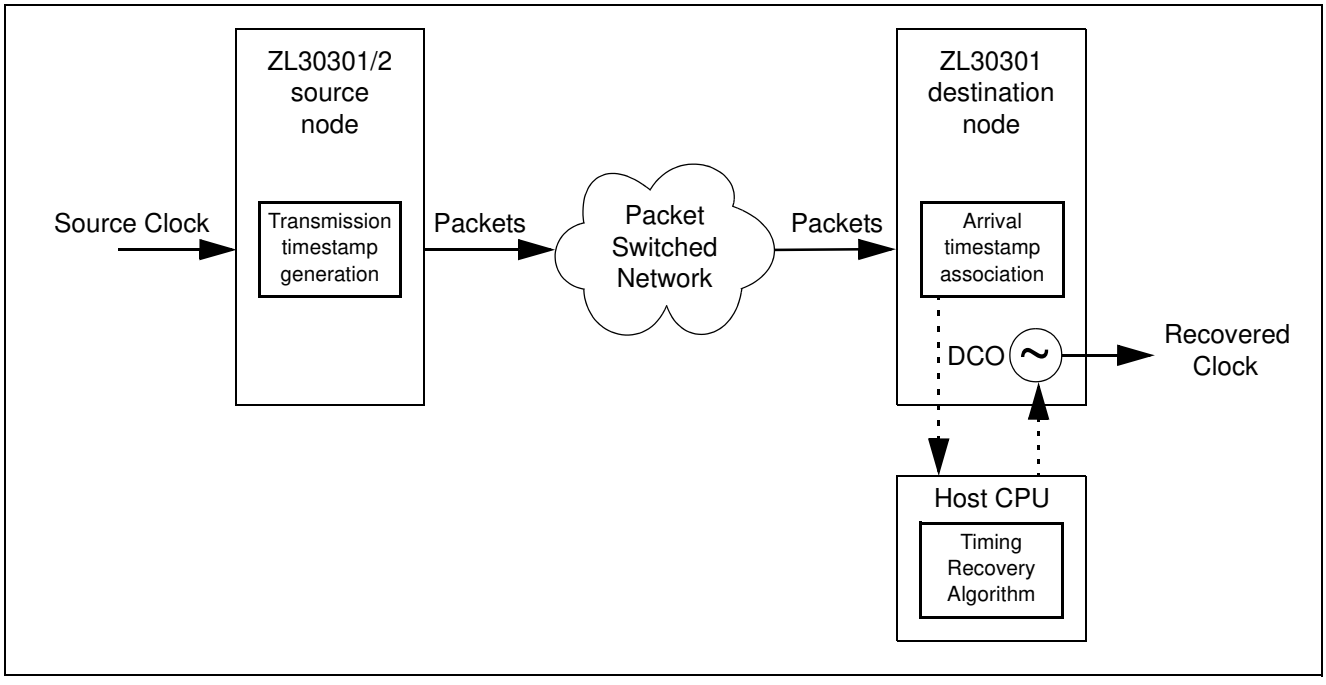


Figure 8 - Adaptive Clock Recovery

Incoming data traffic on the packet interface is received by the MACs, and forwarded to the packet classifier to determine the destination. Those packets identified as timing packets are timestamped on arrival, and this is compared to the timestamp data in the received packet. Where there is no explicit timestamp in the packet (such as in the case of CES data streams), an “effective timestamp” from the sequence number may be calculated, provided the packets are generated at a known, stable and unchanging rate. For CES data packets this is normally the case, since they are made up of a fixed number of bits from a constant-bit-rate data stream.

The host processor filters the results to determine the frequency and phase of the master time-stamping clock, and to compare it to the frequency of the recovered clock. The filtering uses Zarlink’s patent-pending intelligent algorithms and heuristics to take into account any disruptive events in the packet network such as changes in routing, congestion and packet loss. It is also able to compensate for long-term changes in packet delay variation, such as may be exhibited by change in network usage patterns over a 24 hour period. The output of the filter is used to control the frequency of the output clock, which is generated using Zarlink’s precision, low-jitter DCO technology.

The algorithm follows a simple state-machine design, shown in Figure 9. When the device starts up, the clock is in “free run” mode, and may be set to a pre-determined frequency. When it starts to receive timing packets, the algorithm will attempt to lock onto the stream of packets, and moves into the “acquiring” state. The “acquired” state is obtained when the device is locked to the frequency and phase of the source.

Should a network event compromise the delivery of timing packets such that the slave is not able to make a valid assessment of the master clock frequency or phase for a period of time, it will drop back to the “acquiring” state. During this period, it will cease updating the clock frequency to avoid making an adjustment based on bad information, and sending the clock out of specification. If it is still unable to make a good estimation of the master clock then it will move into the “holdover” mode, and generate an interrupt to indicate to the management system that it has lost lock to the master source.

The “holdover” state is typically entered for short durations while network is temporarily disrupted.While in holdover, the drift of the system clock directly affects the accuracy of the clock frequency. The device continues to monitor the incoming packets while in holdover, and on receipt of good packets will move back into the “acquiring” state and attempt to lock back onto the master clock source.

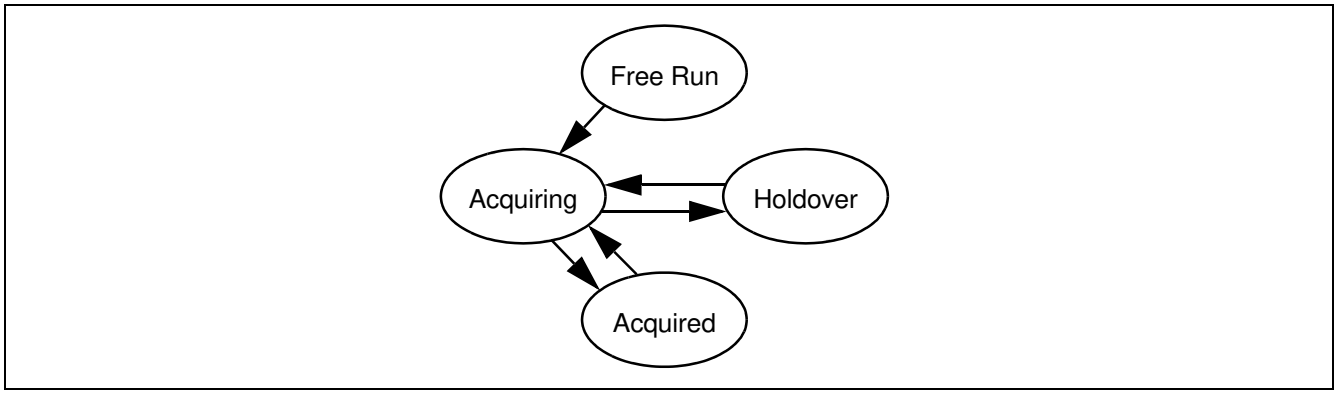


Figure 9 - Adaptive Clock Recovery State Machine

5.3.2 Differential Clock Recovery

For applications where the wander characteristics of the recovered clock are very important, such as when an emulated circuit must be connected into the plesiochronous digital hierarchy (PDH), the ZL30301 also offers a differential clock recovery technique. This relies on having a common reference clock available at each provider edge point. Figure 10 illustrates this concept with a common Primary Reference Source (PRS) clock being present at both the source and destination equipment.

In a differential technique, the timing of the service clock 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.

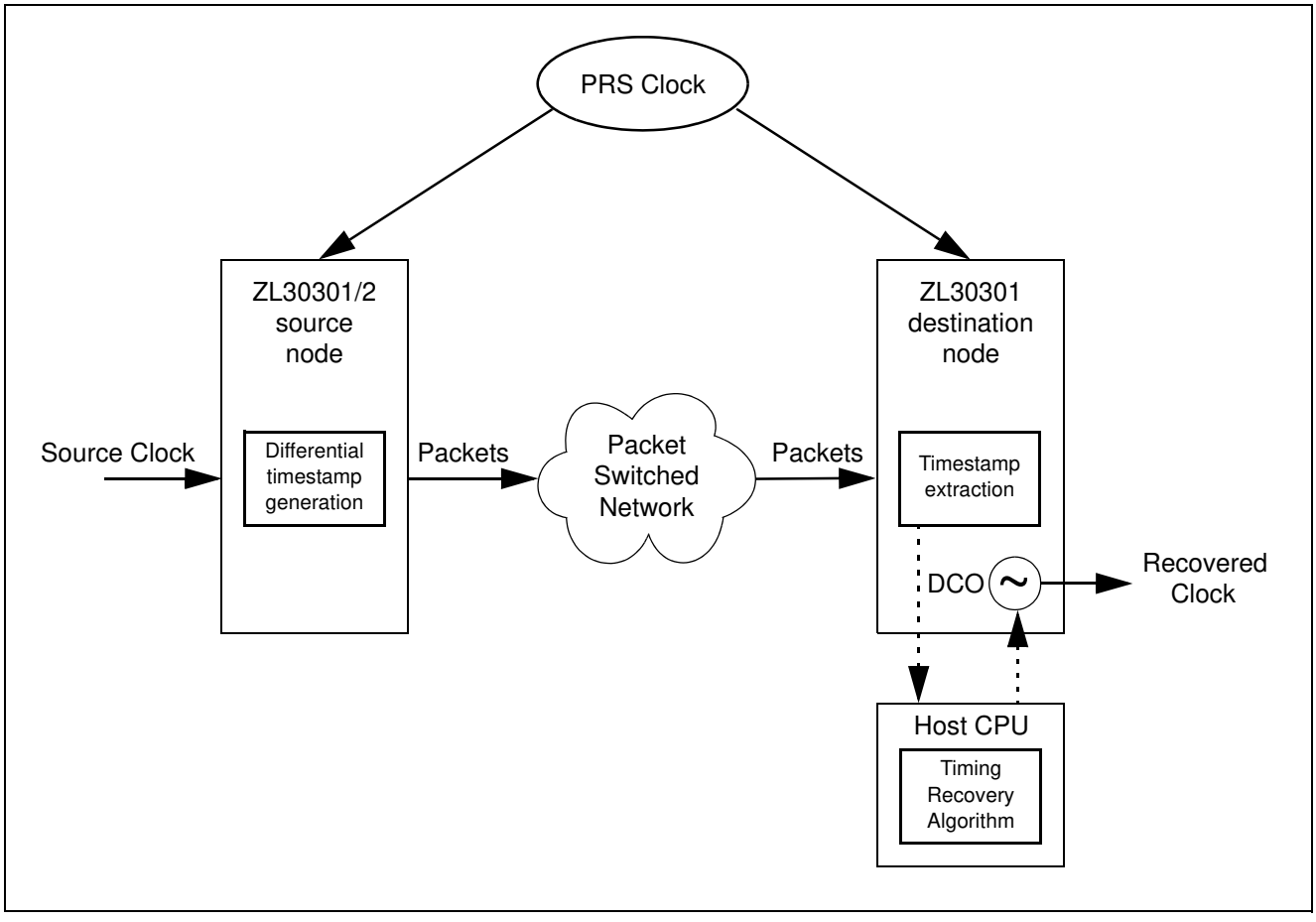


Figure 10 - Differential Clock Recovery

5.3.3 Combination of Adaptive and Differential Clock Recovery

It is possible to combine the two techniques to gain the advantages of differential recovery without the need for a common reference clock. In this scenario, the reference clock is distributed between the two (or more) nodes using the “ToP” adaptive technique, and then several further clocks may be recovered differentially by reference to it. Figure 11 shows how this works in practice.

An example of this kind of application is circuit emulation, where the central office reference may be distributed to several slave nodes, and the plesiochronous clocks associated with the TDM streams differentially encoded with respect to that reference (see Figure 12). The advantage of this is that the single “ToP” stream may achieve much better quality than adaptive recovery from the circuit emulation streams. This is because the packet formation process is freed from the necessity to transport regular, constant bit rate data, and can be optimized for timing transfer only. The result is a more robust, reliable and accurate recovery of the reference clock.

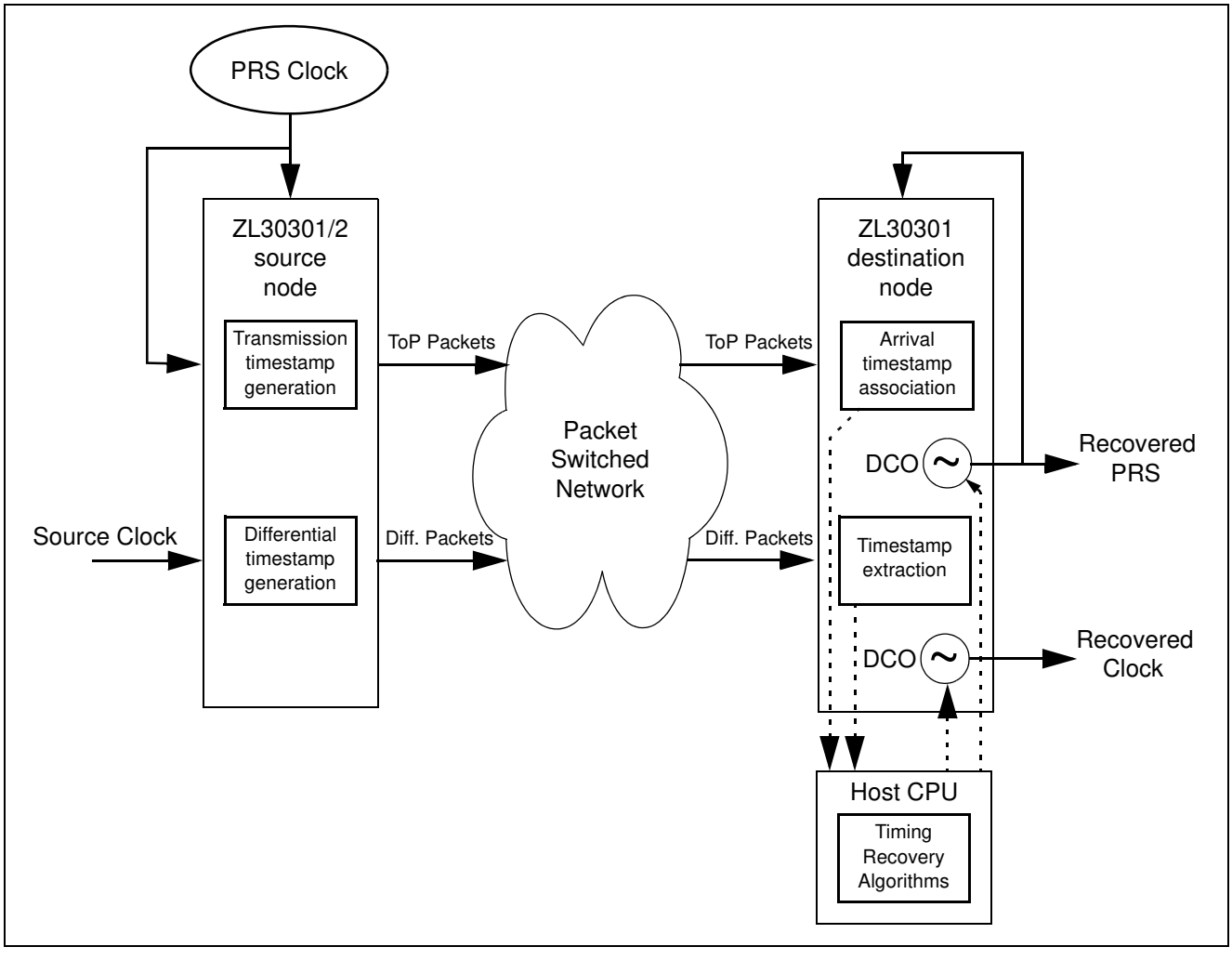


Figure 11 - Combination of Adaptive and Differential Clock Recovery

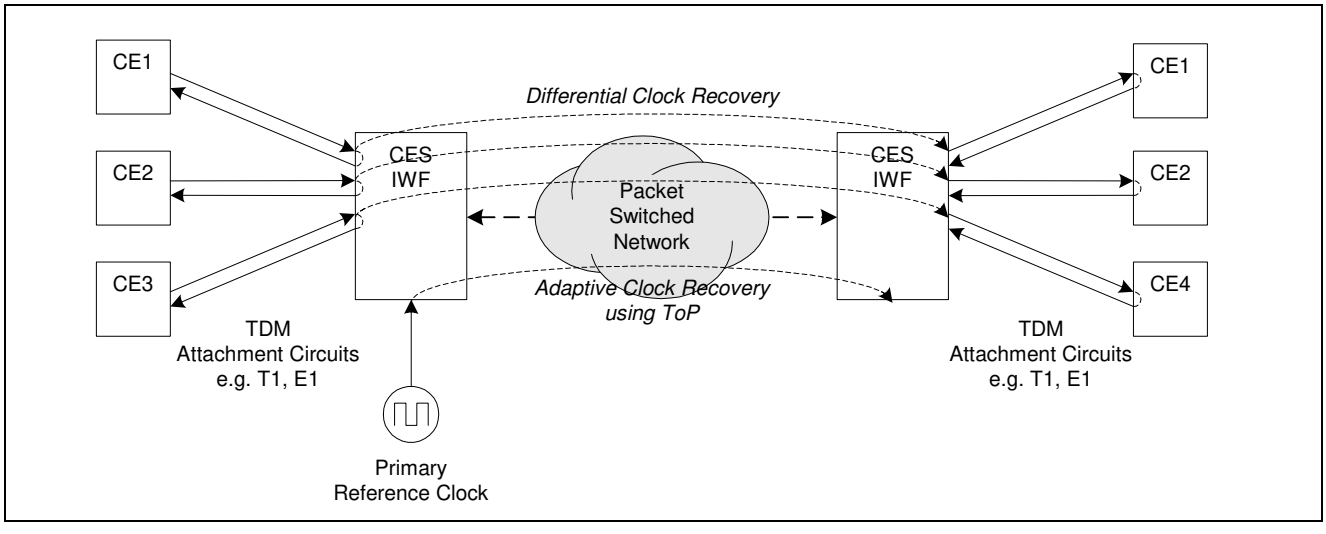


Figure 12 - Application to Circuit Emulation

5.4 Handling of Non-Timing Packets

Typically, a ZL30301 sits as close as possible to the customer interface, to avoid degradation of timing through the customer LAN. The devices may be connected in one of three ways:

1. snoop mode
2. pass-through mode
3. standalone mode

5.4.1 Snoop Mode

This is where the device “listens” as packets fly past on the MII interface, ignoring all non-timing packets, as shown in Figure 13. Timing packets are passed to the clock synthesis function, and the embedded clocks recovered. This techniques is useful in a CES application, where packets are simultaneously passed to a CES interworking function to recover the data, while the timing is recovered by the ZL30301. The technique prevents any transmission by the ZL30301, and hence by implication, the use of a repeater stage.

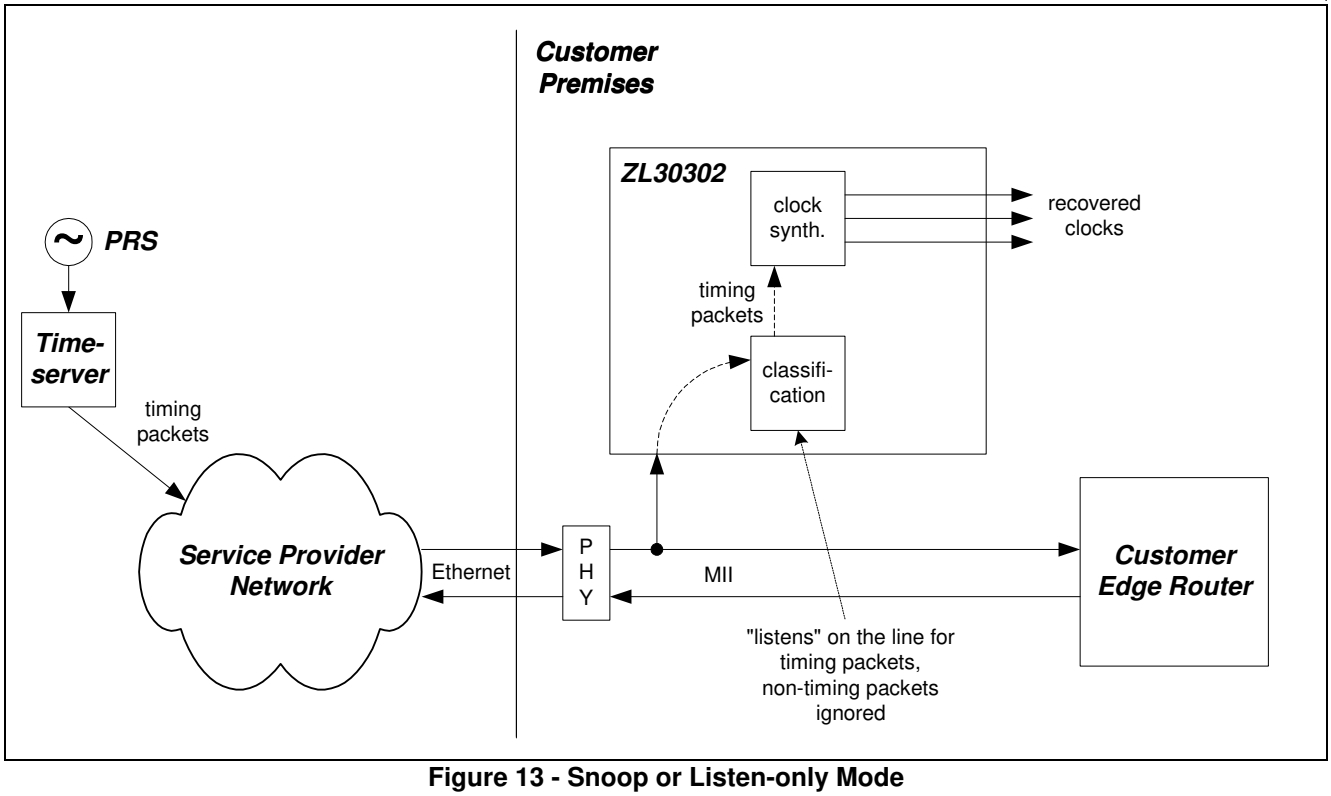


Figure 13 - Snoop or Listen-only Mode

5.4.2 Pass-through Mode

Pass-through mode is where the device forwards all non-timing packets onto the opposite packet interface (e.g. packets from MII1 to MII2 and vice versa). The devices on either side take care of any standard IP protocol control messages. This is the typical expected usage mode for the ZL30301 in slave and repeater applications.

In this mode, the device is situated as near the customer/provider interface as possible (see Figure 14), with one port connected to the service provider network, and the second port connected to the customer network (normally the customer’s edge router or switch). All packets intended for the customer pass through the ZL30301 device. The ZL30301 classifies packets as they arrive to determine if they are timing packets or not. Timing packets are stripped out and processed, while all non-timing packets are forwarded to the customer edge router.

In the reverse direction, packets from the customer edge router into the network are forwarded straight on into the service provider network. Timing packets (such as those generated by a repeater function) may also be forwarded into the network.

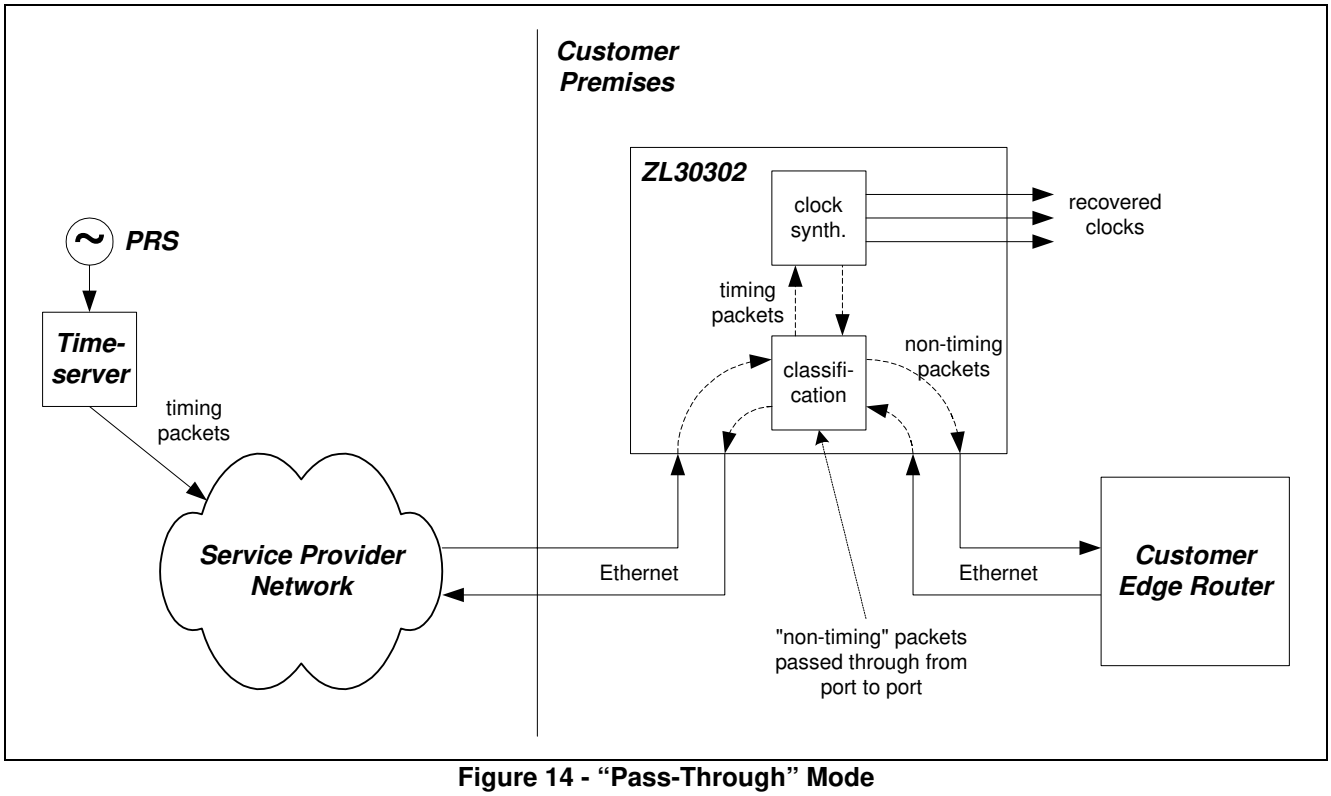


Figure 14 - “Pass-Through” Mode

5.4.3 Standalone Mode

In standalone mode, the ZL30301 is connected on its own port to a switch or router. For example, a standalone time-server or timing master device may use this configuration. The classifier must still identify timing packets for processing, but in this mode all non-timing packets are forwarded to the host CPU controlling the device. Packet forwarding may be done in two ways:

- via the ZL30301 CPU interface, using the DMA queues internal to the device
- forwarding non-timing packets to the second Ethernet port, and connecting this to the CPU’s own Ethernet port

The second case is essentially identical to pass-through mode.

5.5 Contribution of the Network and Local Oscillator on the Performance

The ZL30301 uses a local oscillator to feed the system clock. The ZL30301 uses the system clock for internal operations and relies on the system clock during holdover of the recovered reference clock.

The precision of timing recovery depends on statistical properties of the propagation delay of timing packets through the network and the stability of the local oscillator. The precision timing recovery through a switched network depends on several factors, including:

- The accuracy and stability of the Local Oscillator
- The timing packet rate
- The delay profile of the timing packet stream
  This is in turn dependent on:
  - The length of the timing packet
  - The number of switches or routers in the network
  - The relative data load at the inter-switch links
  - The internal timing granularity of the switches or routers

The accuracy and stability of the regenerated clock at the slaves depends on the combination of the clock recovery methodology and the accuracy and stability of the LO's in the system. The stability of the LO at the slave determines the packet rate that is required to sample its wander fast enough. The target precision for the average frequency accuracy at the slave is within 15 parts per billion.

The performance that the ZL30301 achieves is dependent of the network that connects the master to the slave nodes. A sustained high network load affects the ZL30301 performance. The packet delay profile and the stability of the Local Oscillator are critical factors that are related to each other and suggest that several combinations are possible.

A trade off must be made for the application between:

1. The accuracy and the stability of the LO
2. The maximum allowable timing packet rate
3. The dimensions of the network (number of inter-switch links and long-term average network load)

5.6 CPU Interface

Figure 15 on page 37 gives an example on how to connect the CPU interface pin son the ZL30301 to an MPC8260. The intention is to help board designers understand the function of each of the CPU interface signals. Timing and other important issues are not considered in these examples. For a real interface design, it may be useful to have all CPU control signals connected through a PLD or FPGA logic, so that any tweaking on signals or timing can be easily implemented. As most of the microprocessors have multiple interrupts and DMA controllers, the choices made are discretionary.

A “TA Stretch Circuit” is recommended for host interfaces operating above 40 MHz. Refer to section “CPU TA Output” on page 60 for more details.

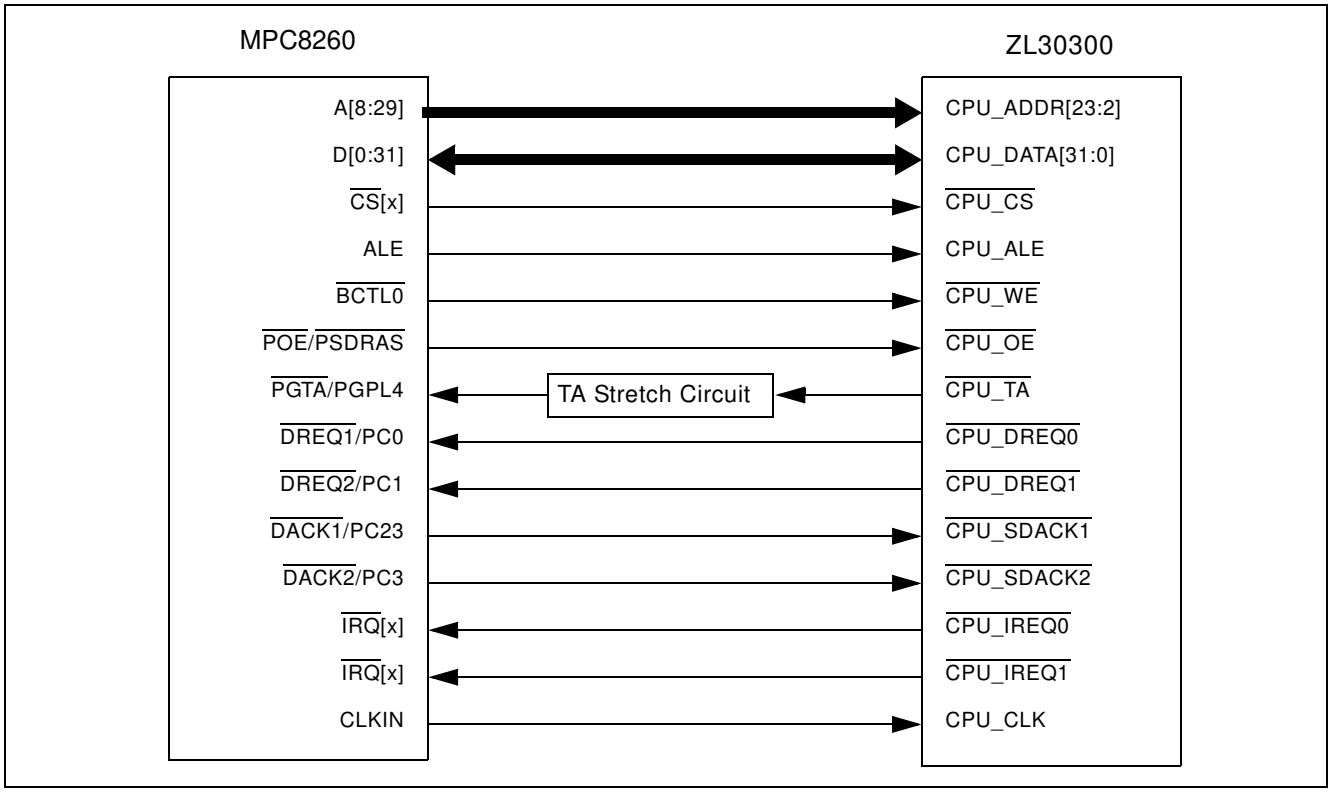


Figure 15 - Block Diagram of ZL30301 to MPC8260 Connection

5.7 Management and Clock Quality Statistics

The ZL30301 can generate both network management and clock quality statistics, partially satisfying the requirements of the following protocols:

- RTCP (RFC3550, section 6) - RTP is used as the protocol for the transfer of timing information
- Pseudo-wire MIB (draft-ietf-pwe3-pw-mib-05) - this is important because timing packets could be distributed via a “timing pseudo-wire”, similar to that used for circuit emulation packets.

5.7.1 Statistics on Received Timing Packets (Slave mode)

Statistic	Value	Required by	Notes
Number of packets received	32 bit count	PW MIB	PW MIB specifies both a 32 and 64 bit counter. For typical timing packet rates the 32 bit counter will never overflow within the 15 minute reporting interval.
Number of payload octets received	32 bit count	PW MIB	PW MIB specifies both a 32 and 64 bit counter. For typical timing packet rates the 32 bit counter will never overflow within the 15 minute reporting interval.

Table 17 - Management Statistics on Received Timing Packets

Statistic	Value	Required by	Notes
Cumulative number of packets lost	32 bit count	RTCP	RTCP specifies an 8-bit “fraction lost since last report” parameter. If required, this may be implemented by periodically monitoring the cumulative value.
Extended highest sequence number received	32 bit value	RTCP	The 16-bit extension may lose accuracy during extended periods of packet loss due to multiple wraparounds.
Estimate of Packet Delay Variation	Minimum and maximum values over a period	RTCP *(inter-arrival jitter)*	Reported in units of 125 us periods, although accuracy is approximately 1 ms. RTCP requests a measure of inter-arrival jitter rather than packet delay variation. The PDV figure may be used to obtain a rough estimate of inter-arrival jitter.

Table 17 - Management Statistics on Received Timing Packets (continued)

5.7.2 Statistics on Transmitted Timing Packets (Master mode)

Statistic	Value	Required by	Notes
Number of packets transmitted	32 bit count	PW MIB, RTCP	PW MIB specifies both a 32 and 64 bit counter. For typical timing packet rates the 32 bit counter will never overflow within the 15 minute reporting interval.
Number of payload octets transmitted	32 bit count	PW MIB, RTCP	PW MIB specifies both a 32 and 64 bit counter. For typical timing packet rates the 32 bit counter will never overflow within the 15 minute reporting interval.

Table 18 - Management Statistics on Transmitted Timing Packets

5.7.3 Status Information on Recovered Clocks

Status	Value	Notes
Slave state information	Free running, Acquiring, Acquired, Holdover	Indication of the current state of the clock recovery algorithm. An interrupt is generated on entry into/exit from holdover.
Master state information	Boolean	Boolean value indicating loss of signal or reference at the master device. This may indicate either an absence of a master clock, or that the master clock has gone into holdover. An interrupt is generated on change of state.
Clock Quality factor	8 bit value	Value indicates a notional “quality value” of the recovered clock, based on how well disciplined the output frequency is. Values range from 0 (poor) - 255 (good), and are both normalised depending on the local oscillator type, and smoothed with a configurable time constant.

Table 19 - Status Information on the Recovered Clocks

5.8 Processing of Incoming Packets

The incoming packets are classified into different packet timing connections. A connection is a mechanism used by the device to keep track of each data is extracted from each packet.

The contents of the packet header of the incoming packets are examined to differentiate which connection should receive the packets. This is achieved using a multi-stage filter/comparator engine called the Packet Classifier.

The first stage of the Packet Classifier is the Rx Filter or Pre-processor which looks at the destination MAC field and the ethertype field to allow packet to be quickly discarded. Additionally this stage also converts any IEEE 802.3 frames into standard Ethernet II frames. Packets that are not discarded will be passed on to the second stage.

The second stage of the Packet Classifier is the Protocol Match or Pre-Classifier stage. This looks for specified fixed bytes in particular byte positions in order to confirm that the incoming packet is of the required protocol. There are four of these matches so allowing four different protocols to be identified. Packets that do not match any of the protocols are discarded. For packets that satisfy one of these protocol matches, certain bytes will then be extracted from the packet header and passed onto various other processing blocks.

The third stage of interest is the Connection Match or Classifier stage. This receives from the protocol match 12 bytes from specified positions assembled into a contiguous array. The Connection Match compares this array of 12 bytes against user definable reference arrays contained within a number of “rules”. If no match is found the packet will be discarded. If however, a match is declared then the matched rule specifies which connection will be used to receive and process the packet as well as specifying the route for the packet through the device.

The Protocol Match also extracts several fields of interest from the packet header. These are the sequence number, timestamp and length fields which are all passed on to other device blocks. One final field that is extracted is a number of “check bytes”. For packets which pass the Connection Match stage these check bytes provide a final check on the authenticity of the required packet.

The output of all these stages is that packets that are accepted will be directed to the appropriate packet timing connection for further processing.

The API provides facilities to program each of these stages independently (see API User Guide for details).

6.0 System Features

6.1 Loopback Modes

The ZL30301 devices support loopback of the clocks from CLKi[3:0] to its respectively CLKo[3:0] on a per port basis. That is to say that CLKo[3:0] may either be sourced from the packet network clock recovery or from CLKi[3:0].

Loopback of the ingress packets on the packet interface 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.

6.2 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 20 illustrates the maximum bandwidths achievable by an external DMA master.

DMA Path	Packet Size	Max Bandwidth Mbps[1]
ZL30301 to CPU only	>1000 bytes	50
ZL30301 to CPU only	60 bytes	6.7
CPU to ZL30301 only	>1000 bytes	60
CPU to ZL30301 only	60 bytes	43
Combined[2]	>1000 bytes	58 (29 each way)
Combined[2]	60 bytes	11 (5.5 each way)

Table 20 - DMA Maximum Bandwidths

Note 1: Maximum bandwidths are the maximum the ZL30301 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.

6.3 Power Up Sequence

To power up the ZL30301 the following procedure must be used:

- The Core supply must never exceed the I/O supply by more than 0.5 V_{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 100 µs after the 100 MHz 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 16.

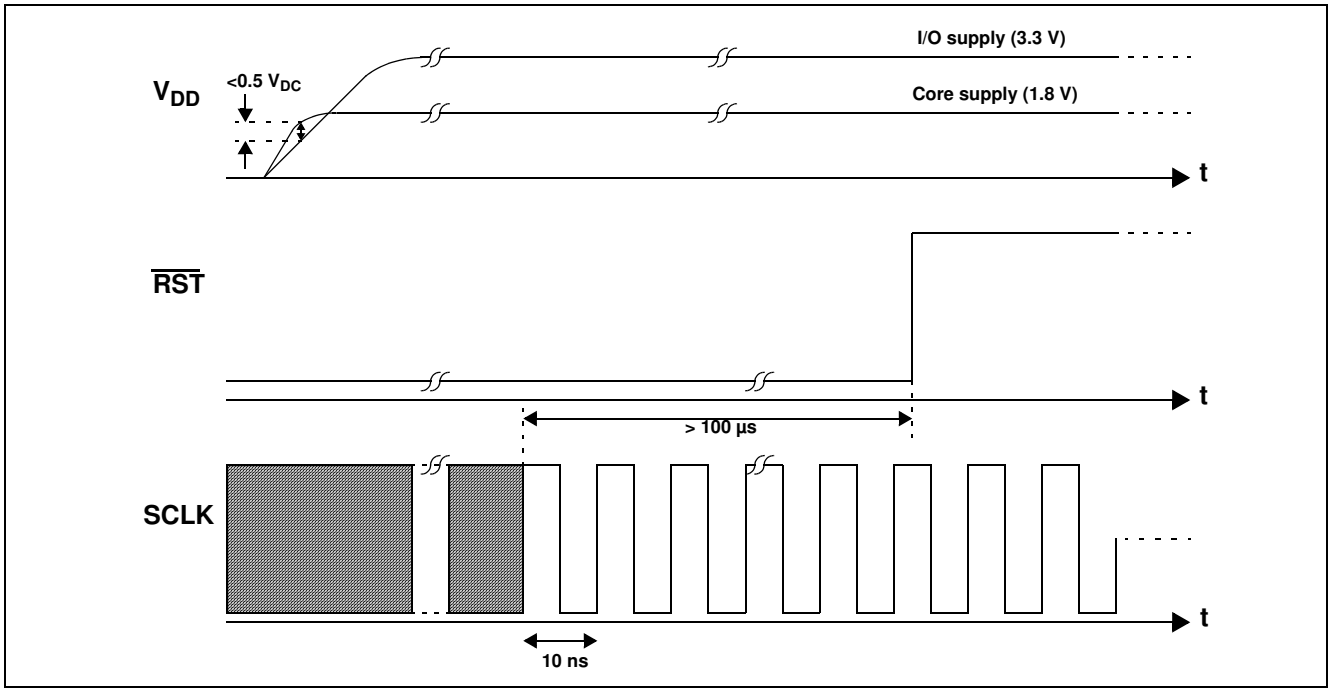


Figure 16 - Powering Up the ZL30301

6.4 JTAG Interface and Board Level Test Features

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

6.5 External Component Requirements

- Direct connection to PowerQUICC™ II (MPC8260) host processor and associated memory, but can support other processors with appropriate glue logic
- Ethernet PHY for each MAC port

6.6 Miscellaneous Features

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

6.7 Test Modes Operation

6.7.1 Overview

The ZL30301 supports the following modes of operation.

6.7.1.1 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.

6.7.1.2 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.

6.7.2 Test Mode Control

The System Test Mode is selected using the dedicated device input bus TEST_MODE[2:0] as follows in Table 21.

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

Table 21 - Test Mode Control

6.7.3 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.

6.7.4 System Tri-state Mode

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

7.0 DC Characteristics

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 (5 V 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

* 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 0.5 V 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.

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, 5 V tolerant inputs	V_{IH_5V}	2.0	-	5.5	V	

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.

DC Electrical Characteristics- Typical characteristics are at 1.8 V core, 3.3 V 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 1.95 V and I/O voltage 3.0 and 3.6 V unless otherwise stated.

Characteristics	Symbol	Min.	Typ.	Max.	Units.	Test Condition
Input Leakage	I_{LEIP}			±1	µA	No pull up/down V_{DD_IO} = 3.6 V
Output (High impedance) Leakage	I_{LEOP}			2	µA	No pull up/down V_{DD_IO} = 3.6 V
Input Capacitance	C_{IP}		1		pF	
Output Capacitance	C_{OP}		4		pF	
Pullup Current	I_{PU}		-27		µA	Input at 0 V
Pullup Current, 5 V tolerant inputs	I_{PU_5V}		-110		µA	Input at 0 V
Pulldown Current	I_{PD}		27		µA	Input at V_{DD_IO}
Pulldown Current, 5 V tolerant inputs	I_{PD_5V}		110		µA	Input at V_{DD_IO}
Core 1.8 V supply current	I_{DD_CORE}		720	950	mA	Note 1,2
PLL 1.8 V supply current	I_{DD_PLL}			1.30	mA	
I/O 3.3 V supply current	I_{DD_IO}		32	120	mA	Note 1,2

Note 1: Worst case assumes the maximum number of active connections.

Note 2: Typical assumes four active E1 connections, and two 100 Mbps MII ports.

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	

Output Levels

Characteristics	Symbol	Min.	Typ.	Max.	Units	Test Condition
Output Low Voltage	V_{OL}			0.4	V	I_{OL} = 6 mA I_{OL} = 12 mA for packet interface (m*) pins and GPIO pins I_{OL} = 24 mA for LED pins
Output High Voltage	V_{OH}	2.4			V	I_{OH} = 6 mA I_{OH} = 12 mA for packet interface (m*) pins and GPIO pins I_{OH} = 24 mA for LED pins

8.0 AC Characteristics

8.1 Clock Interface Timing

The clock signals can generate a wide range of clock frequencies including standard Telecom frequencies for E1, DS1, J2, E3 and DS3. Table 22 shows timing for DS3, which would be the most stringent requirement.

Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
CLKo Period	t_{CTP}		22.353		ns	DS3 clock
CLKo High	t_{CTH}	6.7			ns	
CLKo Low	t_{CTL}	6.7			ns	
CLKi Period	t_{CRP}		22.353		ns	DS3 clock
CLKi High	t_{CRH}	9.0			ns	
CLKi Low	t_{CRL}	9.0			ns	

Table 22 - Clock Interface Timing

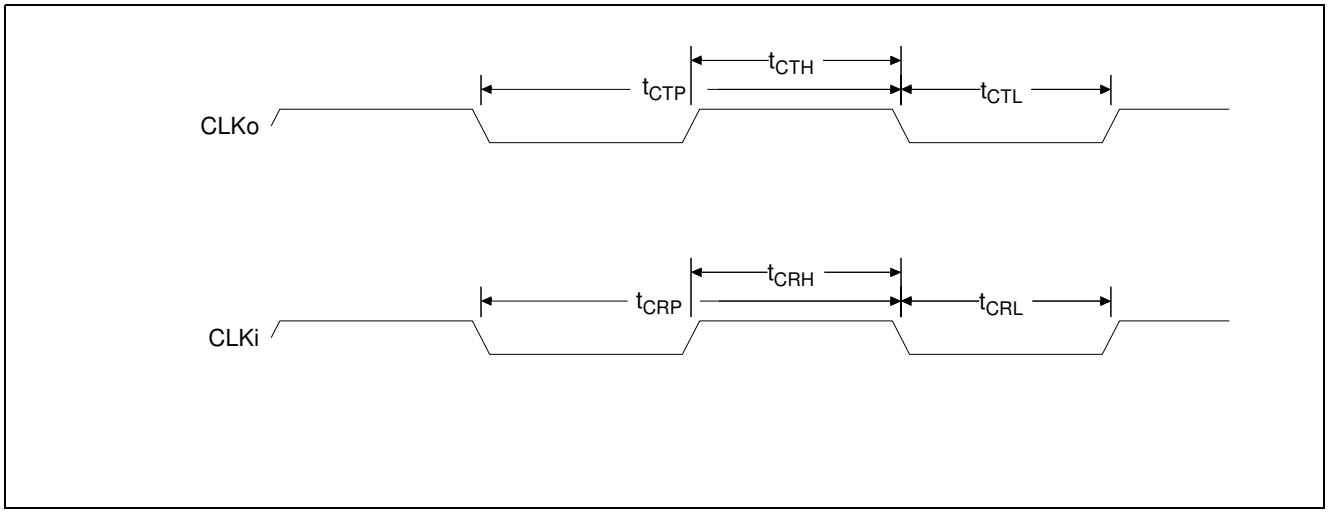


Figure 17 - Clock Interface Timing

8.2 Timestamp Reference Timing

Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
TS_CLKi High / Low Pulsewidth	t_{CPP}	10	-	-	ns	

Table 23 - Timestamp Reference Timing Specification

8.3 Packet Interface Timing

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

8.3.1 MII Transmit Timing

Parameter	Symbol	100 Mbps			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 = 25 pF
TXCLK to TXEN active delay (TXCLK rising edge)	t_{EV}	1	-	25	ns	Load = 25 pF
TXCLK to TXER active delay (TXCLK rising edge)	t_{ER}	1	-	25	ns	Load = 25 pF

Table 24 - MII Transmit Timing - 100 Mbps

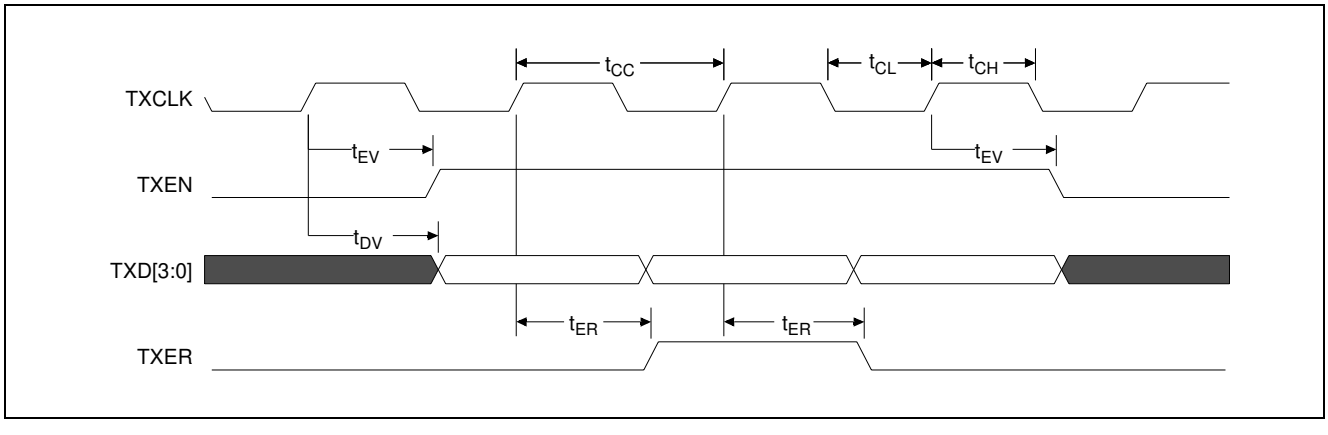


Figure 18 - MII Transmit Timing Diagram

8.3.2 MII Receive Timing

Parameter	Symbol	100 Mbps			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 25 - MII Receive Timing - 100 Mbps

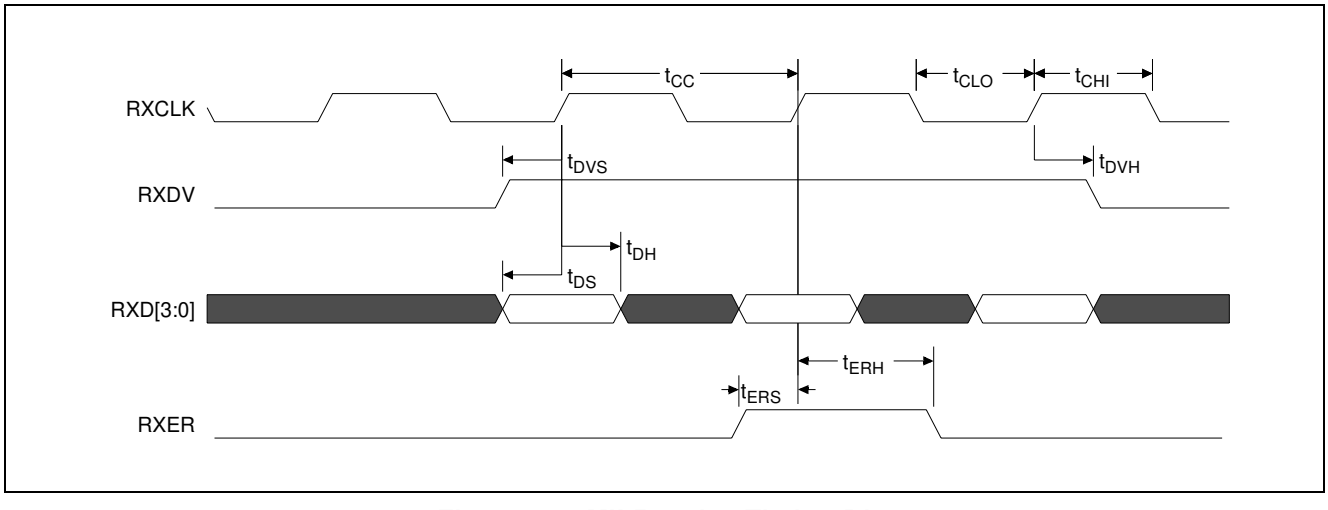


Figure 19 - MII Receive Timing Diagram

8.3.3 GMII Transmit Timing

Parameter	Symbol	1000 Mbps			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 = 25 pF
GTXCLK rise to TXEN active delay	t_{EV}	2	-	6	ns	Load = 25 pF
GTXCLK rise to TXER active delay	t_{ER}	1	-	6	ns	Load = 25 pF

Table 26 - GMII Transmit Timing - 1000 Mbps

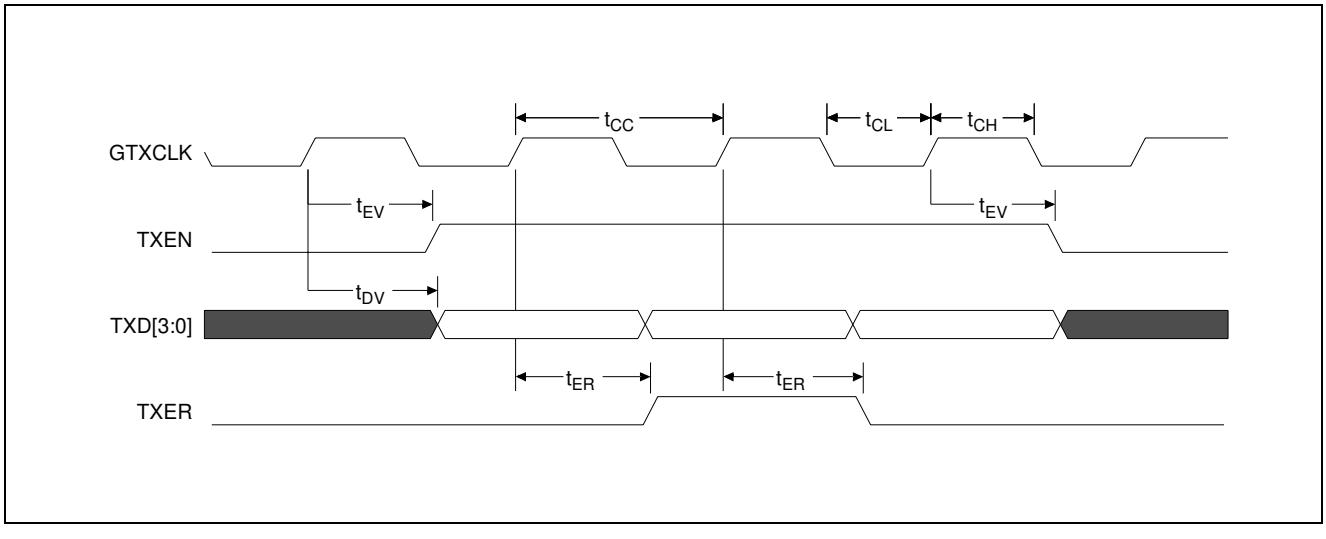


Figure 20 - GMII Transmit Timing Diagram

8.3.4 GMII Receive Timing

Parameter	Symbol	1000 Mbps			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 27 - GMII Receive Timing - 1000 Mbps

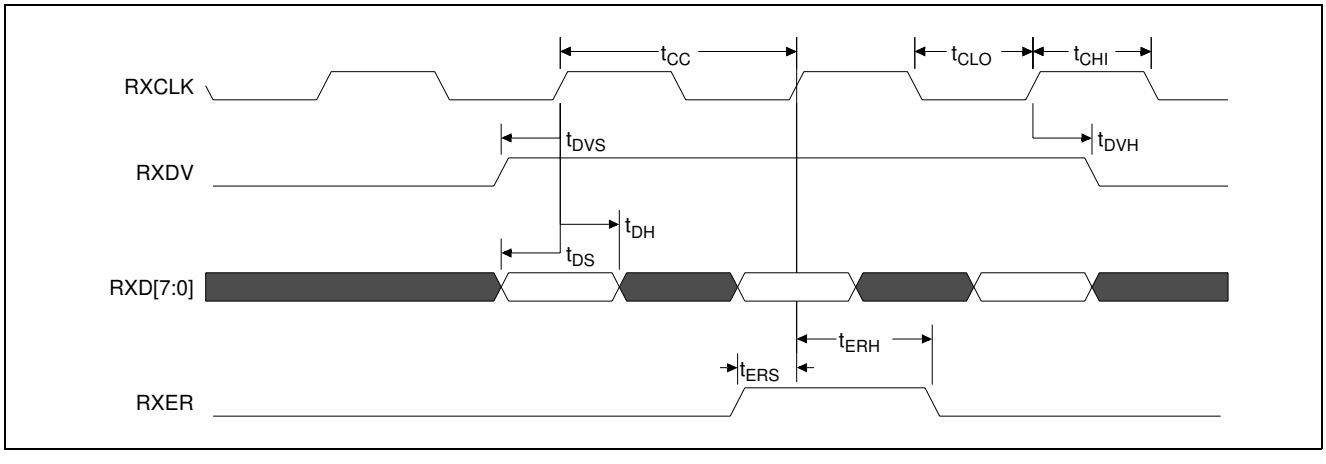


Figure 21 - GMII Receive Timing Diagram

8.3.5 TBI Interface Timing

Parameter	Symbol	1000 Mbps			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 = 25 pF
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 28 - TBI Timing - 1000 Mbps

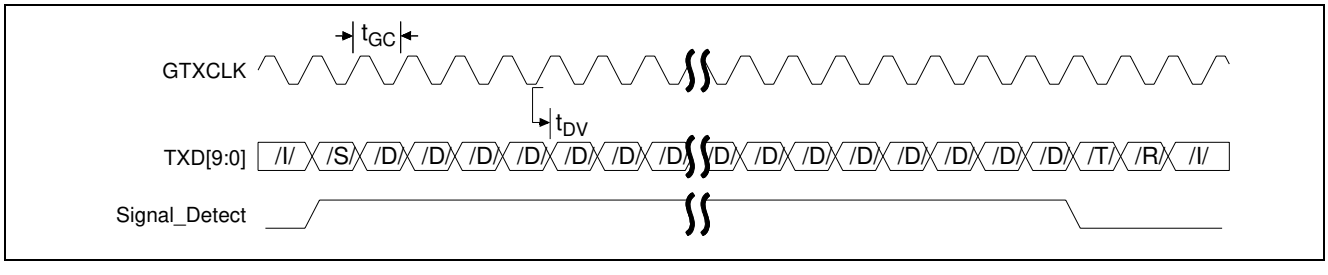


Figure 22 - TBI Transmit Timing Diagram

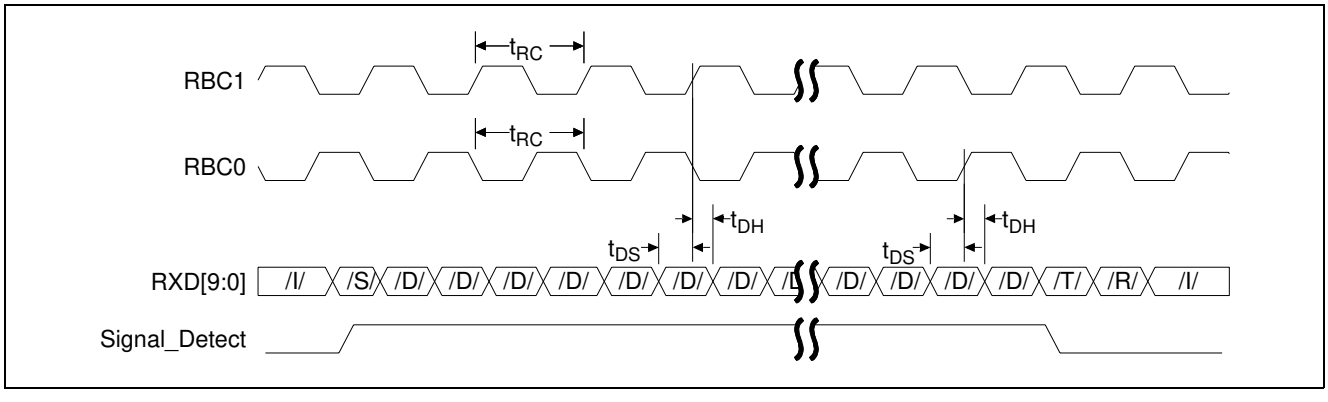


Figure 23 - TBI Receive Timing Diagram

8.3.6 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 29 - 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.

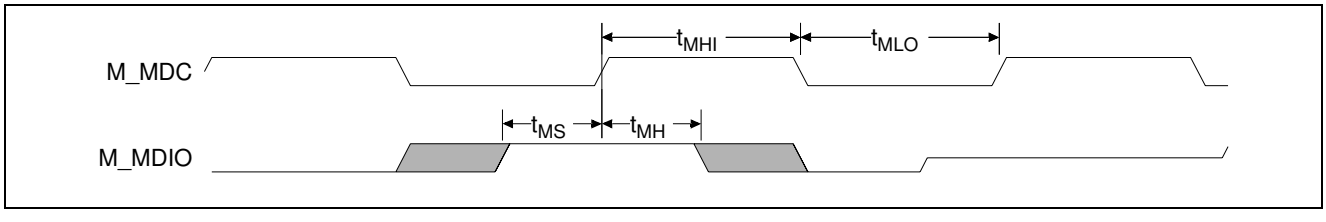


Figure 24 - Management Interface Timing for Ethernet Port - Read

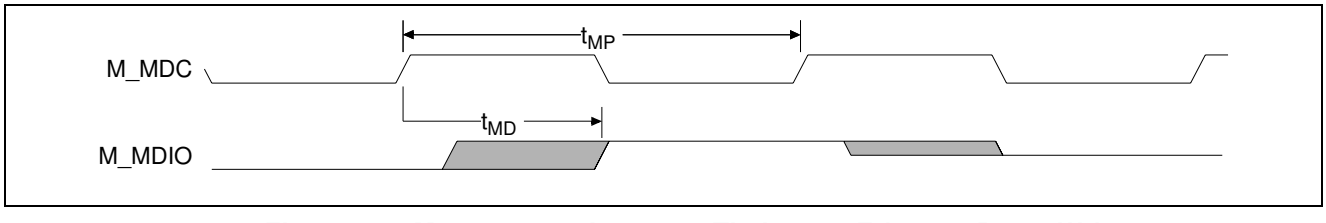


Figure 25 - Management Interface Timing for Ethernet Port - Write

8.4 CPU Interface Timing

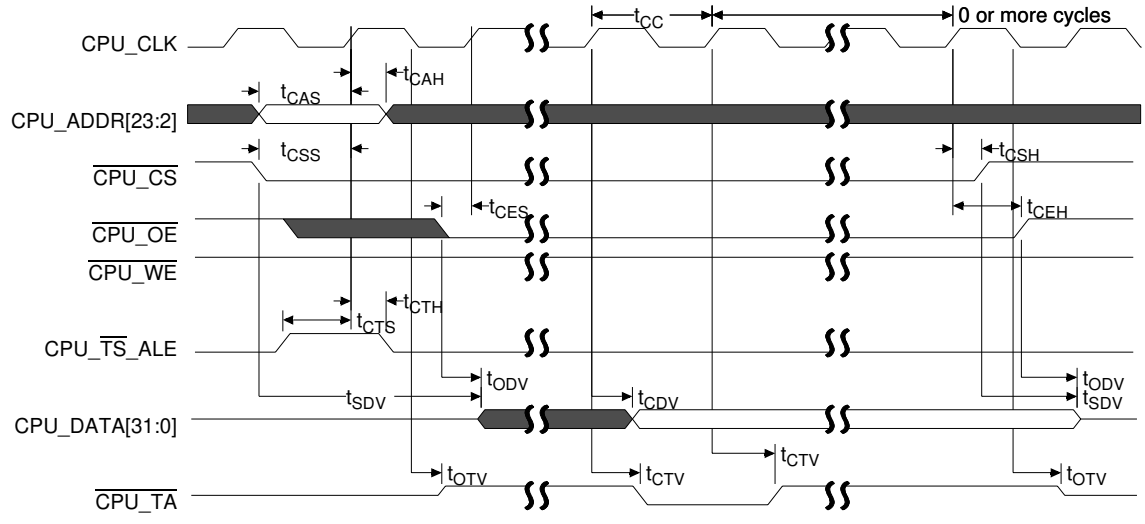
Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
CPU_CLK Period	t_{CC}		15.152		ns	
CPU_CLK High Time	t_{CCH}	6			ns	
CPU_CLK Low Time	t_{CCL}	6			ns	
CPU_CLK Rise Time	t_{CCR}			4	ns	
CPU_CLK Fall Time	t_{CCF}			4	ns	
CPU_ADDR[23:2] Setup Time	t_{CAS}	4			ns	
CPU_ADDR[23:2] Hold Time	t_{CAH}	2			ns	
CPU_DATA[31:0] 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 30 - CPU Timing Specification

Note 1: Load = 50 pF maximum

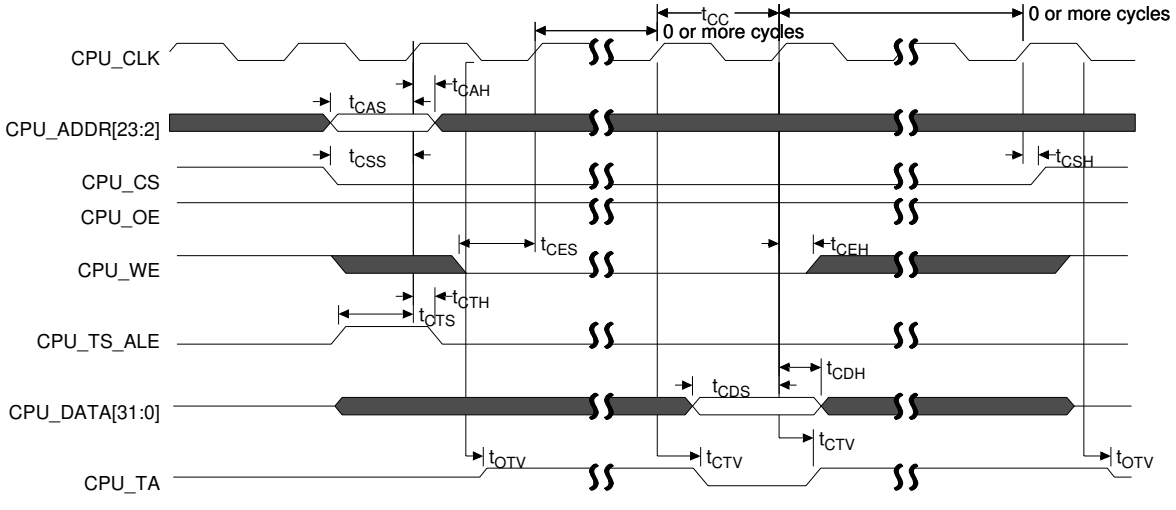
Note 2: The maximum value of t_{CTV} may cause setup violations if directly connected to the MPC8260. See Section 9.2 for details of how to accommodate this during board design.

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.



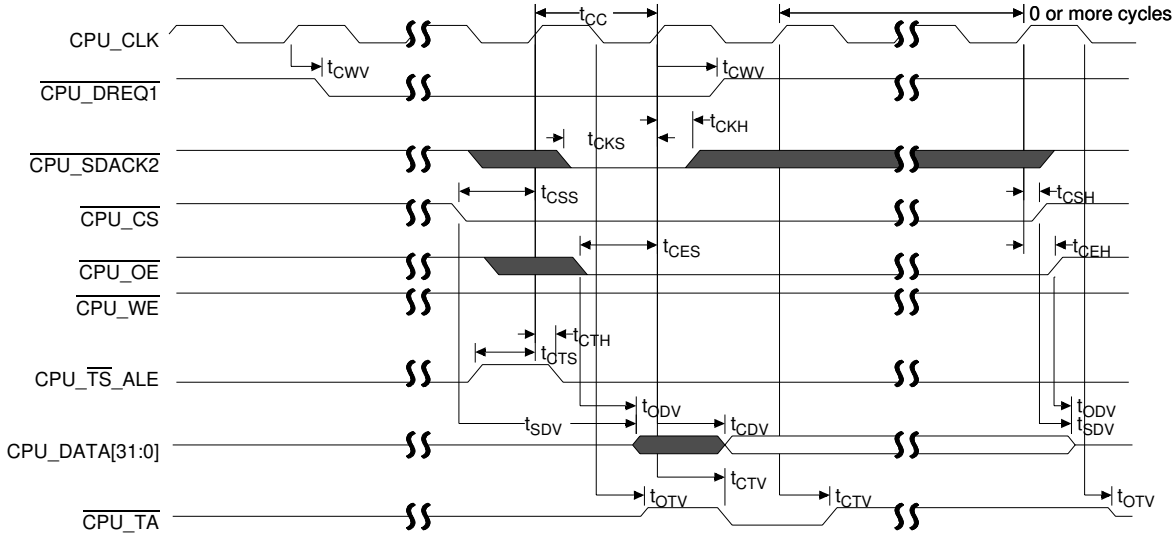
NOTE: CPU_DATA is valid when CPU_TA is asserted. CPU_DATA will remain valid while both CPU_CS and CPU_OE are asserted. CPU_TA will continue to be driven until CPU_CS is deasserted. CPU_CS and CPU_OE must BOTH be asserted to enable the CPU_DATA output.

Figure 26 - CPU Read - MPC8260



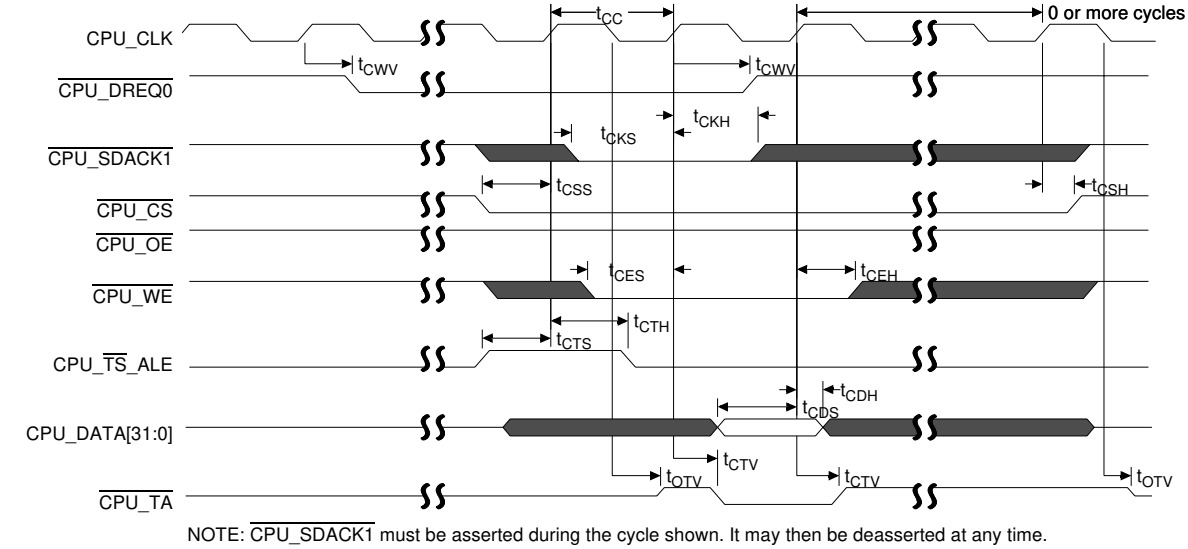
NOTE: Following assertion of CPU_TA, CPU_CS may be deasserted. The MPC8260 will continue to assert CPU_CS until CPU_TA has been synchronized internally. CPU_TA will continue to be driven until CPU_CS is finally deasserted. During continued assertion of CPU_CS, CPU_WE and CPU_DATA may be removed.

Figure 27 - 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 28 - 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 29 - CPU DMA Write - MPC8260

8.5 System Function Port

Parameter	Symbol	Min.	Typ.	Max.	Units	Notes
SYSTEM_CLK Frequency	CLK_{FR}	-	100	-	MHz	Note 1, Note 2 and Note 3
SYSTEM_CLK accuracy (	CLK_{ACA}	-	-	±200	ppm	Note 4

Table 31 - System Clock Timing

Note 1: The system clock frequency stability affects the holdover-operating mode. 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 0.1 ppm/ºC, a 10ºC change in temperature while the DCO is in holdover will result in a frequency accuracy offset of 1 ppm.

Note 2: The system clock frequency affects the operation of the DCO in free-run mode. In this mode, the DCO 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 8.192 MHz ± SYSTEM_CLK accuracy ± 0.005 ppm).

Note 3: The absolute SYSTEM_CLK accuracy must be controlled to ± 30 ppm in to enable the internal DCO to meet T1/E1 specification.

Note 4: Maximum system clock accuracy for the proper operation of the device.

8.6 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 32 - 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.

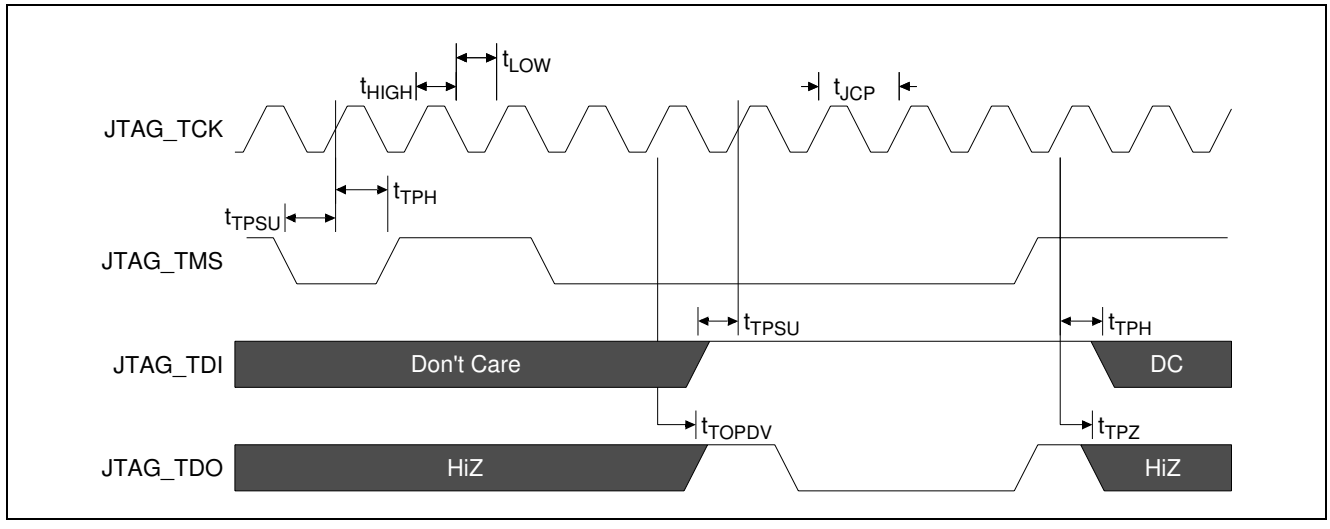


Figure 30 - JTAG Signal Timing

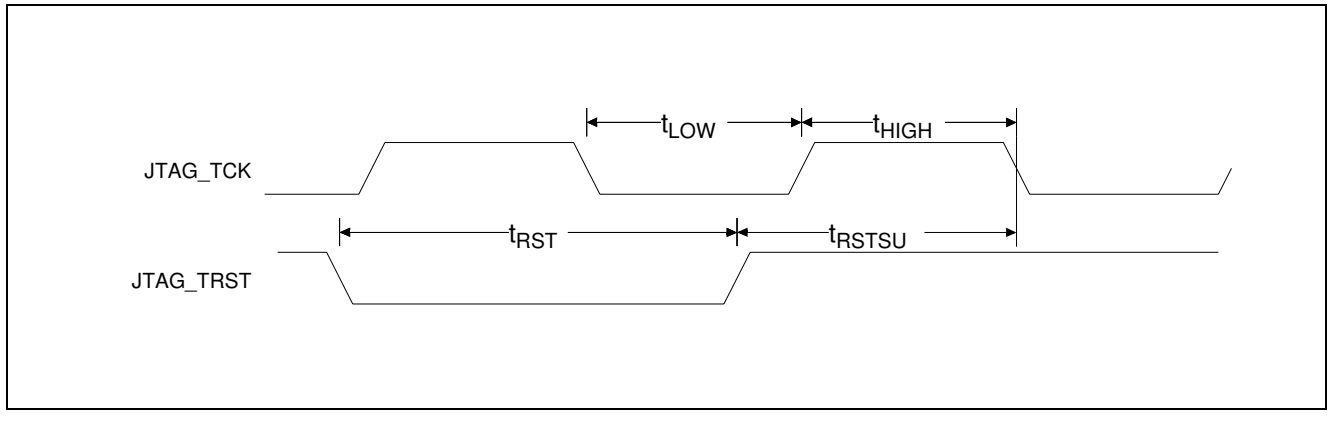


Figure 31 - JTAG Clock and Reset Timing

9.0 Design and Layout Guidelines

This guide will provide information and guidance for PCB layouts when using the ZL30301. Specific areas of guidance are:

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

9.1 High Speed Clock & Data Interfaces

On the ZL30301 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:

- GMAC Interfaces
- Clock 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 = 6.8 ps/mm
Typical rise/fall time for a ZL30301 output = 2.5 ns

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

Therefore tracks longer than 61 mm 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 minimized 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 minimize 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.

9.1.1 GMAC Interface - Special Considerations During Layout

The GMII interface passes data to and from the ZL30301 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 minimized.

9.1.2 Clock Interface - Special Considerations during Layout

Although the clock rates at the clock interface are typically low (1.544 MHz to 10 MHz) the outputs' edge speeds share the characteristics of the higher data rate outputs. Therefore they should be treated with the same care extended to the other interfaces. In particular, the input clock traces to the ZL30301 devices should be treated with care.

9.1.3 Summary

Particular effort should be made to minimize crosstalk from ZL30301 outputs and ensuring fast rise time to these inputs.

In summary:

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

9.2 CPU TA Output

The CPU_TA output signal from the ZL30301 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 32 is implemented in systems operating above 40 MHz 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

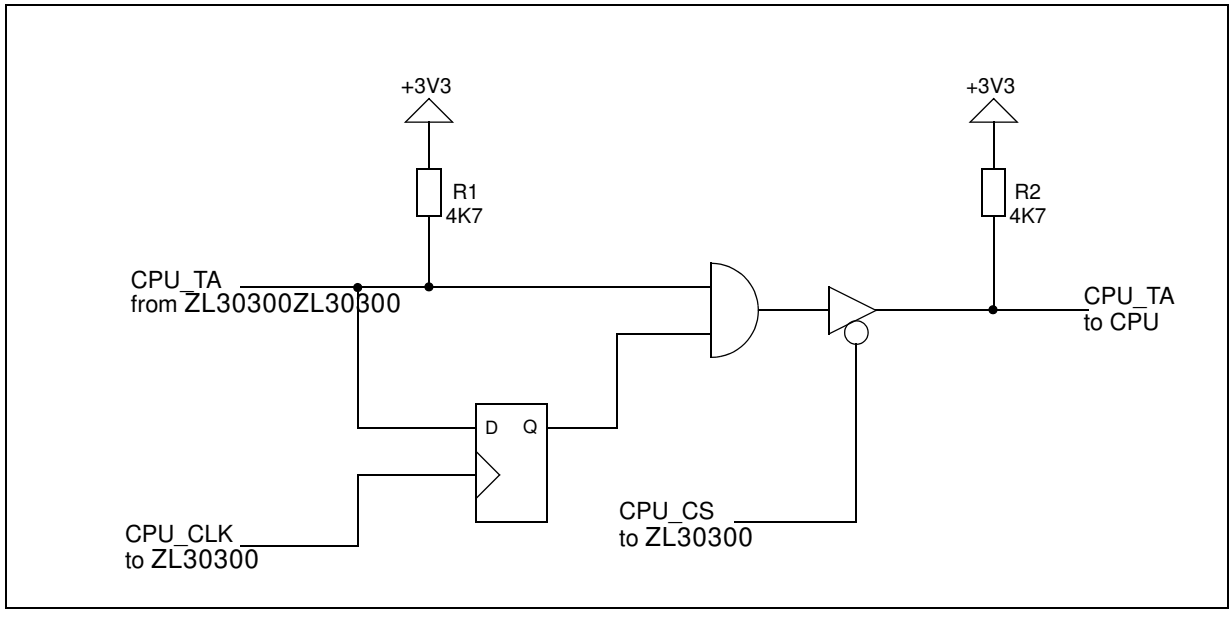


Figure 32 - CPU_TA Board Circuit

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 ZL30301 and that the clock to the 74LCX74 is derived from the same clock source as that input to the ZL30301.

10.0 Reference Documents

10.1 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
- MPC8260AEC/D Revision 0.7; Motorola MPC8260 Family Hardware Specification
- RFC 768; UDP
- RFC 791; IPv4
- RFC2460; IPv6
- RFC 2661; L2TP
- RFC 3550; RTP
- 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
- G.712; TDM Timing Specification (Method 2)
- G.823; Control of Jitter/Wander with digital networks based on the 2.048 Mbps hierarchy
- G.824; Control of Jitter/Wander with digital networks based on the 1.544 Mbps hierarchy
- ANSI T1.101 Stratum 3/4
- Telcordia GR-1244-CORE Stratum 3/4/4e
- RFC 3931; L2TP Version 3
- IETF PWE3 draft-ietf-pwe3-cesopsn
- IETF PWE3 draft-ietf-pwe3-satop
- IETF PWE3 draft-ietf-pwe3-tdmoip
- ITU-T Y.1413; TDM-MPLS Network Interworking
- MEF 8; Implementation Agreement for the Emulation of PDH Circuits over Metro Ethernet Networks
- MFA 8.0.0; Emulation of TDM Circuits over MPLS Using Raw Encapsulation Implementation Agreement

11.0 Glossary

API Application Program Interface

CESoP Circuit Emulation Services over Packet

CESoPSN Circuit Emulation Services over Packet Switched Networks

CPU Central Processing Unit

DCO Digital Controlled Oscillator

DMA Direct Memory Access

GMII Gigabit Media Independent Interface

IETF Internet Engineering Task Force

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

JTAG Joint Test Algorithms Group (refers to a boundary-scan architecture providing a board-level test facility - see IEEE1149)

L2TP Layer 2 Tunneling Protocol (RFC 2661 and RFC 3931)

LAN Local Area Network

MAC Media Access Control

MEF Metro Ethernet Forum

MFA MPLS and Frame Relay Alliance

MII Media Independent Interface

MIB Management Information Base

MPLS Multi Protocol Label Switching

PDH Plesiochronous Digital Hierarchy

PLL Phase Locked Loop

PRS Primary Reference Source

PWE3 Pseudo-Wire Emulation Edge to Edge (a working group of the IETF)

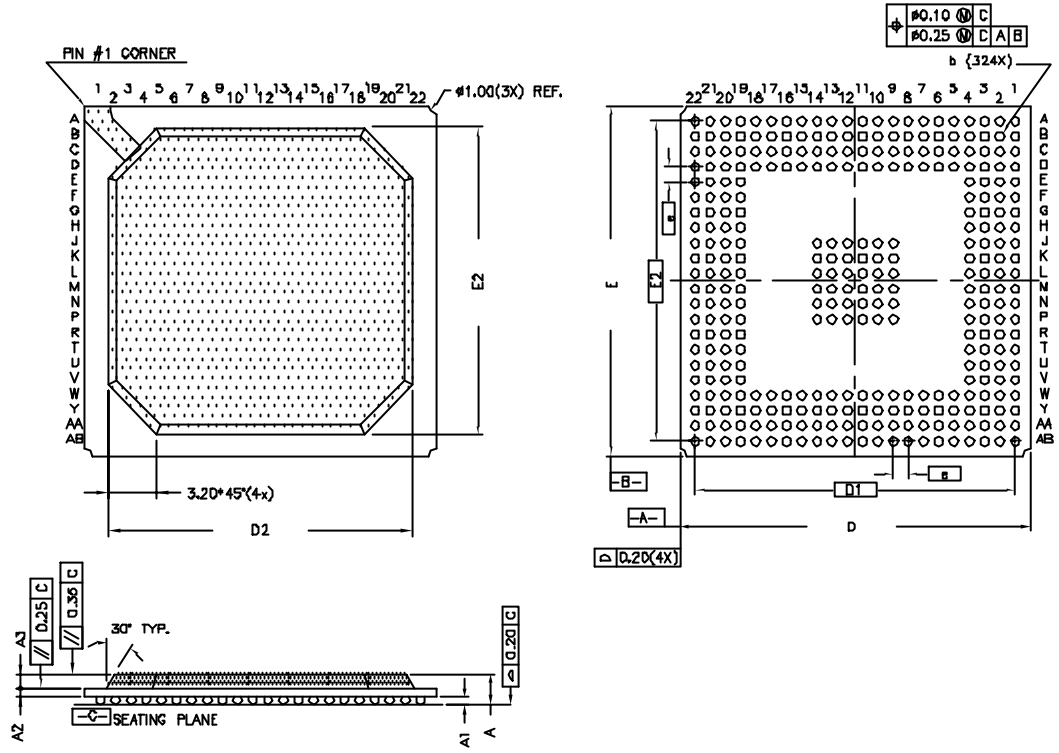
RTP Real Time Protocol (RFC 3550)

SAToP Structure-Agnostic TDM over Packet

TDM Time Division Multiplexing

UDP User Datagram Protocol (RFC 768)

VLAN Virtual Local Area Network



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	1.90	2.03	2.16
A1	0.40	0.50	0.60
A2		0.56 Ref.	
A3		0.97 Ref.	
b	0.50	0.60	0.70
D	22.80	23.00	23.20
D1		21.00 Ref.	
D2		20.00 Ref.	
E	22.80	23.00	23.20
E1		21.00 Ref.	
E2		20.00 Ref.	
e		1.00 Ref.	

Confirms to JEDEC MS-034
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.

ISSUE	1			
ACN				
DATE	02 June 04			
APPRD.				

ZARLINK SEMICONDUCTOR

Package Code GA

Previous package codes

Package Outline for 324, PBGA (23x23x2.03)mm, 1.00 Pitch

105072

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

www.zarlink.com

TECHNICAL DOCUMENTATION - NOT FOR RESALE

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

www.zarlink.com

TECHNICAL DOCUMENTATION - NOT FOR RESALE