ARM® CoreLink™ TLX-400 Network Interconnect Thin Links

Revision: r0p2

Supplement to ARM® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual

ARM DSU 0028C (ID121113)

ARM CoreLink TLX-400 Network Interconnect Thin Links
Supplement to ARM CoreLink NIC-400 Network Interconnect Technical Reference Manual

Release Information

The following changes have been made to this book.

Change history

Date	Issue	Confidentiality	Change
02 July 2012	A	Non-Confidential	First release for r0p0
07 May 2013	B	Non-Confidential	First release for r0p1
11 December 2013	C	Non-Confidential	First release for r0p2

Product Status

The information in this document is final, that is for a developed product.

Web Address

`http://www.arm.com`

Contents
ARM CoreLink TLX-400 Network Interconnect Thin Links Supplement to ARM CoreLink NIC-400 Network Interconnect Technical Reference Manual

Chapter 1
Introduction

This chapter introduces the CoreLink TLX-400 Network Interconnect Thin Links. For convenience Thin Links is referred to as TLX. This chapter contains the following sections:

1.1 About the product

The CoreLink TLX-400 Network Interconnect Thin Links is an extension to the CoreLink NIC-400 Network Interconnect base product and provides a mechanism to reduce the number of signals in an AXI point-to-point connection and enable it to be routed over a longer distance.

1.2 Key features

The CoreLink TLX-400 Network Interconnect Thin Links has the following features:

- TLX reduces routing congestion and aids timing closure of point-to-point connections. Point-to-point connections are implemented as forward and reverse links. Each link can be independently configured to reduce the number of wires the connection requires.
- TLX supports clock domain crossing to aid physical implementation:
  - The end points of the TLX are always specified to be in a different clock domain. The relationship of the clocks must be defined as asynchronous.
- TLX can incorporate other NIC-400 functions. For example:
  - A connection between components of different data widths.
  - A connection between components of different protocols.
- TLX can be used in conjunction with *Quality of Service for Virtual Networks* (QVN-400). For more information, see the *ARM® CoreLink™ QVN-400 Network Interconnect Advanced Quality of Service for Virtual Networks, Supplement to ARM® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual*.
- TLX can be used in conjunction with *Advanced Quality of service* (QoS-400). For more information, see the *ARM® CoreLink™ QoS-400 Network Interconnect Advanced Quality of Service, Supplement to ARM® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual*.
- TLX is implemented as a forward and reverse link. Each link is partitioned into three functional sections:
  - *Interface Layer* (IL).
  - *Data Link Layer* (DLL).
  - *Physical Layer* (PL).
- The IL presents the AMBA protocol compliant interface and performs:
  - Channel identification of transfer packets across the link.
  - Arbitration between transfer packets on to the link.
  - Packing of transfer packets on to the link.
  - Flow control across the link.
- The DLL performs buffering of transfer packets at the destination end of each link.
- You can modify or replace the PL to enable different physical implementations.
- You can enable hierarchical clock gating support for the destination domain.
- You can enable hierarchical clock gating signalling for the source domain.
- You can configure support for power domain crossing.

1.3 Product revisions

This section describes the differences in functionality between product revisions:

r0p0 First release.

r0p1 Second release. No technical updates.

r0p2 Third release. Updated Figure 2-2 on page 2-7 and Figure 2-3 on page 2-8.

Chapter 2
Functional Description

This chapter provides a functional description of the CoreLink TLX-400 Network Interconnect Thin Links and how it works. It contains the following sections:

2.1 Interfaces

TLX-400 enables TLX protocol functionality to be added to NIC-400 slave and master interfaces in a larger system, as shown in Figure 2-1.

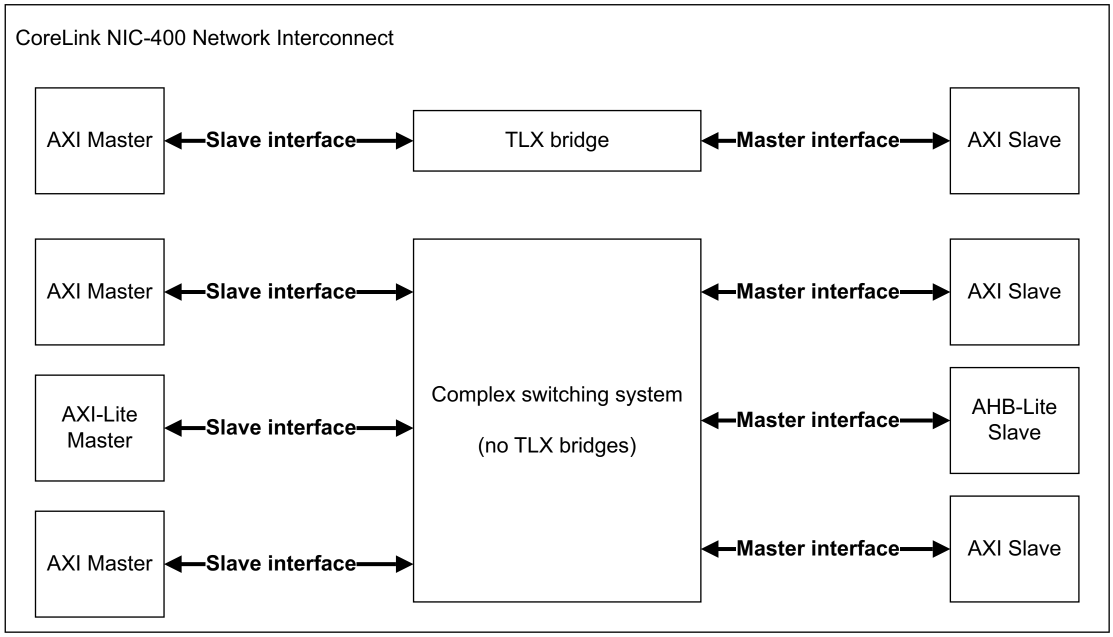

Figure 2-1 TLX in a larger NIC-400 configuration

This section describes:

- *Slave interfaces*.
- *Master interfaces* on page 2-3.
- *Low-power interface* on page 2-3.
- *Physical layer interfaces* on page 2-4.
- *Signal descriptions* on page 2-4.

2.1.1 Slave interfaces

Within NIC-400, you can only configure TLX as a bridge, that is, TLX can only support a single slave interface. However, it is possible for one or more TLX bridges to be configured within a larger NIC, as shown in Figure 2-1.

The TLX supports all the slave interfaces that the base NIC-400 product supports. These are:

- AXI3.
- AXI4.
- AHB-Lite slave.
- AHB-Lite mirrored master.

You can only configure an AHB slave interface if the master interface is not of type AHB, that is, neither an AHB to AHB bridge nor an AHB to AHB TLX bridge is supported.

You can configure a slave interface to support QVN if:

- The QVN product license is installed.
- The slave interface type is AXI3 or AXI4.

See the *ARM® CoreLink™ QVN-400 Network Interconnect Advanced Quality of Service for Virtual Networks Supplement to ARM® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual*.

2.1.2 Master interfaces

Within NIC-400, you can only configure TLX as a bridge, that is, TLX can only support one master interface. However, it is possible for the TLX bridge to be configured within a larger NIC, as Figure 2-1 on page 2-2 shows.

The TLX supports all the master interfaces that the base NIC-400 product does. These are:

- AXI3.
- AXI4.
- AHB-Lite slave.
- AHB-Lite mirrored master.

You can only configure an AHB master interface provided the slave interface is not type AHB, that is, neither an AHB to AHB bridge nor an AHB to AHB TLX bridge is supported.

You can configure a master interface to support QVN if:

- The QVN product license is installed.
- The slave interface type is AXI3 or AXI4.
- The slave interface is configured to support QVN.

See the *ARM® CoreLink™ QVN-400 Network Interconnect Advanced Quality of Service for Virtual Networks Supplement to ARM® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual*.

2.1.3 Low-power interface

This section describes:

- *Hierarchical clock gating interfaces*.
- *Power domain crossing interfaces*.

Hierarchical clock gating interfaces

When the TLX bridge is configured to support hierarchical clock gating there is a *Low-power Interface* (LPI) to enable clock gating of the master interface clock domain. See the *ARM® AMBA®* AXI™ *and* ACE *Protocol Specification* for more information on the LPI.

When the LPI indicates that the low-power state has been entered then the clock for the master interface domain can be clock gated.

When the TLX bridge is configured to support hierarchical clock gating, there is a **cactive** signal output from the slave clock domain to assist in clock gating of the slave interface domain. It is driven HIGH when the TLX requires the clock, and must be incorporated in the slave interface clock domain clock controller.

Power domain crossing interfaces

When a bridge is configured to support *Power Domain Crossing* (PDC), hierarchical clock gating support is also included.

An additional PDC LPI is output from the slave domain to support the power gating sequence. When the PDC LPI is changing mode of operation, the slave domain clock must be operational. The bridge slave domain asserts an additional LPI **cactive** signal, to indicate that a clock is required and must not be turned off.

If the PDC LPI indicates that the low-power state has been entered, either power domain can be power gated.

Note

If the master domain is powered down, any transactions issued to the slave domain are stalled at the interface.

2.1.4 Physical layer interfaces

You can replace the physical layer to enable flexibility in the implementation.

There are two interfaces in each direction. These are AXI stream compliant. See the *ARM® AMBA® 4 AXI4-Stream Protocol Specification* for more information.

Forward direction

These interfaces consist of:

AXI stream data interface

AXI stream data interface. The forward link width you define in the AMBA Designer GUI defines the data width of the interface.

AXI stream flow control interface

The number of response channels that are R and B for an AXI to AXI bridge defines the data width of the interface.

Reverse direction

These interfaces consist of:

AXI stream data interface

The reverse link width you define in the AMBA Designer GUI defines the data width of the interface.

AXI stream flow control interface

Provided QVN is enabled, the number of response channels, that is, AW, AR, and W multiplied by the number of virtual networks, defines the data width of the interface.

2.1.5 Signal descriptions

The interface signals are listed in Table 2-1 on page 2-5. For additional information on these signals see the *ARM® AMBA® 4 AXI4-Stream Protocol Specification*.

Table 2-1 on page 2-5 uses the following parameters to define the signal widths:

Where:

n Data bus width in bits.

m Flow control bus width in bits. For a forward *Physical Layer* (PL):

For an AHB TLX-400 bridge:
The flow control bus width = 1.

For an AXI TLX-400 bridge:
The flow control bus width = 2.

For an AXI TLX-400 bridge with QVN AMBA quality of service using virtual networks:
The flow control bus width = 2 * number of virtual networks.

Table 2-1 Forward PL interface signal descriptions

Signal Name	Description
pl_fwd_x_tlxclk	This is the clock source and the global clock signal. All signals are sampled on the rising edge of **pl_fwd_x_tlxclk**.
pl_fwd_x_tresetn	This is the reset source and the global reset signal. **pl_fwd_x_tresetn** is active-LOW.
tvalid_pl_fwd_x_s_stream	Indicates a valid data transfer is being presented to the forward PL.
tdata_pl_fwd_x_s_stream[(n-1):0]	This is the data payload input.
tready_pl_fwd_x_s_stream	Indicates that the slave can accept a transfer in the current cycle.
tvalid_pl_fwd_x_m_stream	Indicates that the forward physical layer is presenting a valid data transfer to the forward *Data Link Layer* (DLL).
tdata_pl_fwd_x_m_stream[(n-1)]:0]	This is the data payload output.
tready_pl_fwd_x_m_stream	Indicates that the forward data link layer can accept a transfer in the current cycle.
tvalid_pl_fwd_x_s_flow	Indicates a valid flow control transfer is being presented to the forward PL.
tdata_pl_fwd_x_s_flow[(m-1):0]	This is the flow control payload input.
tready_pl_fwd_x_s_flow	Indicates that the slave can accept a transfer in the current cycle.
tvalid_pl_fwd_x_m_flow	Indicates that the forward physical layer is presenting a valid flow control transfer to the forward DLL.
tdata_pl_fwd_x_m_flow[(m-1):0]	This is the flow control payload output.
tready_pl_fwd_x_m_flow	Indicates that the forward data link layer can accept a transfer in the current cycle.

Table 2-2 on page 2-6 uses the following parameters to define the signal widths:

Where:

n Data bus width in bits.

m Flow control bus width in bits. For a reverse physical layer:

For an AHB TLX-400 bridge:
The flow control bus width = 2.

For an AXI TLX-400 bridge:
The flow control bus width = 3.

For an AXI TLX-400 bridge with QVN AMBA quality of service using virtual networks:
The flow control bus width = 3 * number of virtual networks.

Table 2-2 Reverse PL interface signal descriptions

Signal Name	Description
pl_rev_x_tlxclk	This is the clock source and the global clock signal. All signals are sampled on the rising edge of **pl_rev_x_tlxclk**.
pl_rev_x_tresetn	This is the reset source and the global reset signal. **pl_rev_x_tresetn** is active-LOW.
tvalid_pl_rev_x_s_stream	Indicates a valid data transfer is being presented to the reverse PL.
tdata_pl_rev_x_s_stream[(n-1):0]	This is the data payload input.
tready_pl_rev_x_s_stream	Indicates that the slave can accept a transfer in the current cycle.
tvalid_pl_rev_x_m_stream	Indicates that the reverse physical layer is presenting a valid data transfer to the reverse DLL.
tdata_pl_rev_x_s_flow[(m-1):0]	This is the data payload output.
tready_pl_rev_x_m_stream	Indicates that the reverse data link layer can accept a transfer in the current cycle.
tvalid_pl_rev_x_s_flow	Indicates a valid flow control transfer is being presented to the reverse PL.
tdata_pl_rev_x_s_flow[(m-1):0]	This is the flow control payload input.
tready_pl_rev_x_s_flow	Indicates that the slave can accept a transfer in the current cycle.
tvalid_pl_rev_x_m_flow	Indicates that the reverse physical layer is presenting a valid flow control transfer to the reverse DLL.
tdata_pl_rev_x_m_flow[(m-1):0]	This is the flow control payload output.
tready_pl_rev_x_m_flow	Indicates that the reverse data link layer can accept a transfer in the current cycle.

2.2 Operation

Figure 2-2 shows the TLX hierarchy.

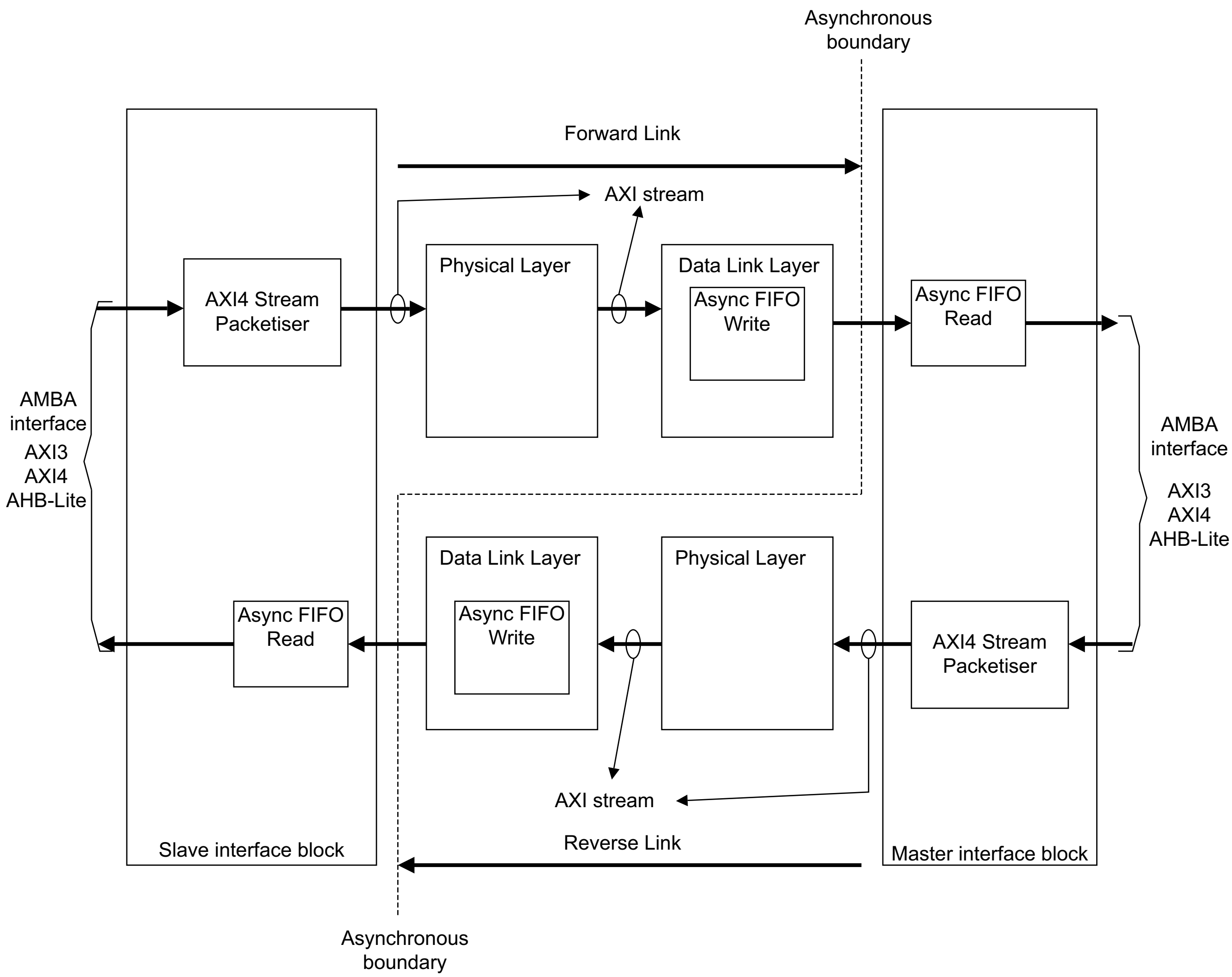

Figure 2-2 TLX hierarchy

2.2.1 TLX

The TLX consists of two AXI stream interfaces for each direction:

- Forward data.
- Reverse data.
- Forward flow.
- Reverse flow.

Figure 2-3 on page 2-8 shows the TLX hierarchical structure.

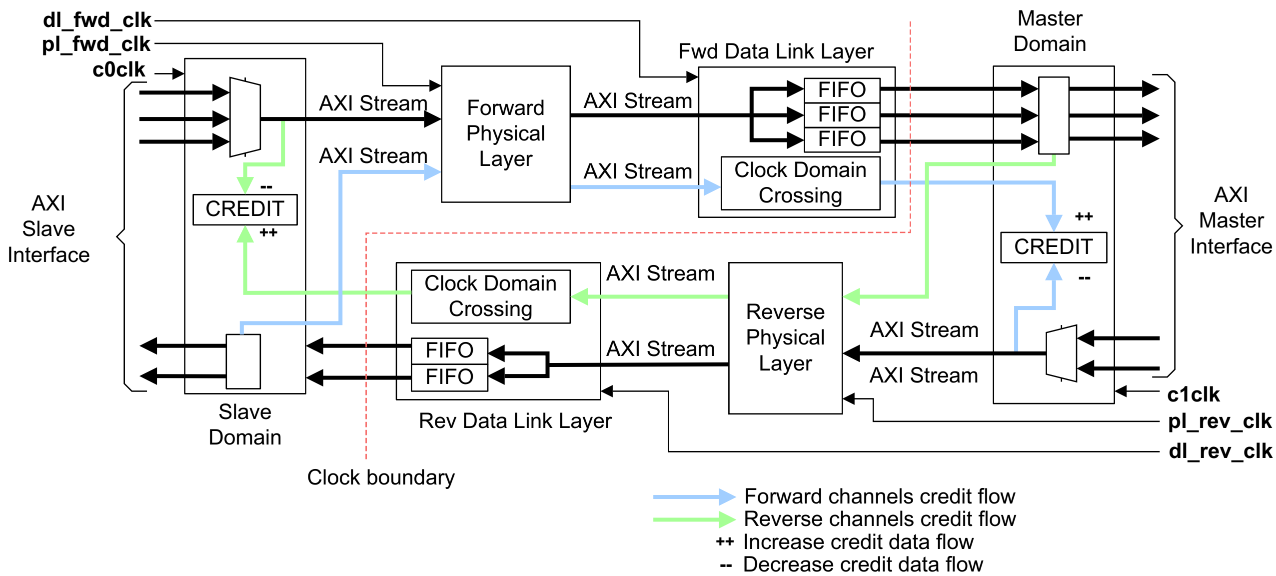

Figure 2-3 TLX hierarchical structure

The data stream links transport the AMBA forward and reverse channel beats. The flow stream interfaces are used for replenishment of credit tokens.

This architecture means that the TLX bridge operation is independent of the physical layer latency.

This section describes:

- *Forward AMBA channels*.
- *Response AMBA channels*.
- *Data packing* on page 2-9.
- *Arbitration* on page 2-9.

Forward AMBA channels

To guarantee that the data link does not stall and therefore cause blocking between channels, a forward channel beat is only issued onto the forward data stream link when that channel has credit, indicating that there is space at the destination to accept that beat. Therefore, a channel credit is consumed when a beat is issued into the physical layer data stream link. The number of credits is equal to the number of buffer slots available for that channel at the destination. When the credits are all consumed for a channel, then no more beats are issued onto the data stream interface.

When a destination buffer slot is emptied through a beat being issued downstream out of the thin link bridge, a flow control credit for that channel is returned back across the reverse flow control stream link.

Response AMBA channels

To guarantee that the data link does not stall and therefore cause blocking between channels, a reverse channel beat is issued onto the reverse data stream link when that channel has credit, indicating that there is space at the destination to accept that beat. Therefore, the credit of a channel is consumed when a beat is issued into the physical layer data stream link. The number of credits is equal to the number of buffer slots available for that channel at the destination. When the credits are all consumed for a channel then no more beats are issued onto the data stream interface.

When a destination buffer slot is emptied through a beat being issued back upstream out of the TLX bridge, a flow control credit for that channel is then returned back across the forward flow control stream link.

Data packing

Five different packing strategies are available. There is no requirement to use the same strategy in both forward and reverse directions:

- Widest Width.
- Widest Width / 2.
- Widest Width / 4.
- Forward or Reverse channels:
  - Address Width + Data Width, for forward channels only.
  - Read Data Width + Response, for reverse channels only.
- User Defined.

Widest Width

This is the width of the widest channel in the direction under consideration.

Widest Width / 2

This is half the width of the widest channel in the direction under consideration.

Widest Width / 4

This is a quarter of the width of the widest channel in the direction under consideration.

Forward or Reverse channels:

- Address Width + Data Width.

  This is the widest address channel plus the widest data width in the forward link direction.
- Read Data Width + Response.

  This is the width of the read data plus response in the reverse link direction.

User Defined (in Bytes)

This is any multiple byte width up to a maximum of the widest channel / 2.

Note

See *Configuring Thin Links in the Configuring the Network chapter of the ARM® CoreLink™ NIC-400 Network Interconnect Supplement to ARM® CoreLink™ ADR-400 AMBA® Designer User Guide*.

Arbitration

This entails arbitration of which channel to issue to the link. When there is a choice of channels available, it is achieved by using the **awqos** value, **arqos** value, or the stored **awqos** value associated with the current W channel traffic. The reverse link selection uses a round robin arbitration.

Note

For more information on **axqos** values, see the *ARM® CoreLink™ QoS-400 Network Interconnect Advanced Quality of Service, Supplement to ARM® CoreLink™ NIC-400 Network Interconnect Technical Reference Manual*.

Appendix A
Revisions

This appendix describes the technical changes between released issues of this book.

Table A-1 Issue A

Change	Location	Affects
First release	-	-

Table A-2 Differences between issue A and issue B

Change	Location	Affects
Diagram added to Thin Link section for clarity, that is a subsection of Operation	*TLX* on page 2-7	All revisions

Table A-3 Differences between issue B and issue C

Change	Location	Affects
Updated key features	*Key features* on page 1-3	All revisions
Added paragraph for clarification	*Low-power interface* on page 2-3	All revisions
Signal descriptions section added	*Signal descriptions* on page 2-4	All revisions
Updated TLX hierarchy diagram	*TLX hierarchy* on page 2-7	All revisions
Updated TLX hierarchy structure diagram	*TLX hierarchical structure* on page 2-8	All revisions