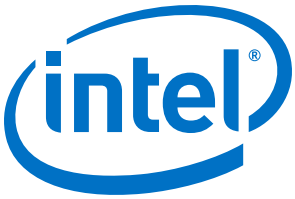


PCI Express*: Migrating to Intel® Stratix® 10 Devices for the Avalon® Streaming Interface

Updated for Intel® Quartus® Prime Design Suite: **Quartus Prime Pro v17.1 Stratix 10 Editions**

AN791 | 2017.05.08
Latest document on the web: **PDF** | **HTML**

Contents

1. Introduction

This design migration reference guide describes differences in PCI Express implementations between the Arria® 10 (or Stratix® V) and Stratix 10 device families. It should ease migration from those earlier device families to Stratix 10 devices.

The Stratix 10 implementation provides many new features. In addition, the interfaces and functions are different than the Arria 10 and Stratix V IP cores.

This design migration reference guide covers migration for the Avalon® Streaming (Avalon-ST) interface. A separate design migration reference guide covers the Avalon Memory-Mapped (Avalon-MM) and Avalon-MM with DMA interfaces.

Related Information

PCI Express: Migrating to Stratix 10 from Arria 10 for the Avalon-MM and Avalon-MM DMA Interfaces

2. Configuration Options

The parameters available to configure the PCI Express IP core in Stratix 10 devices differ from those available for Stratix V and Arria 10 devices.

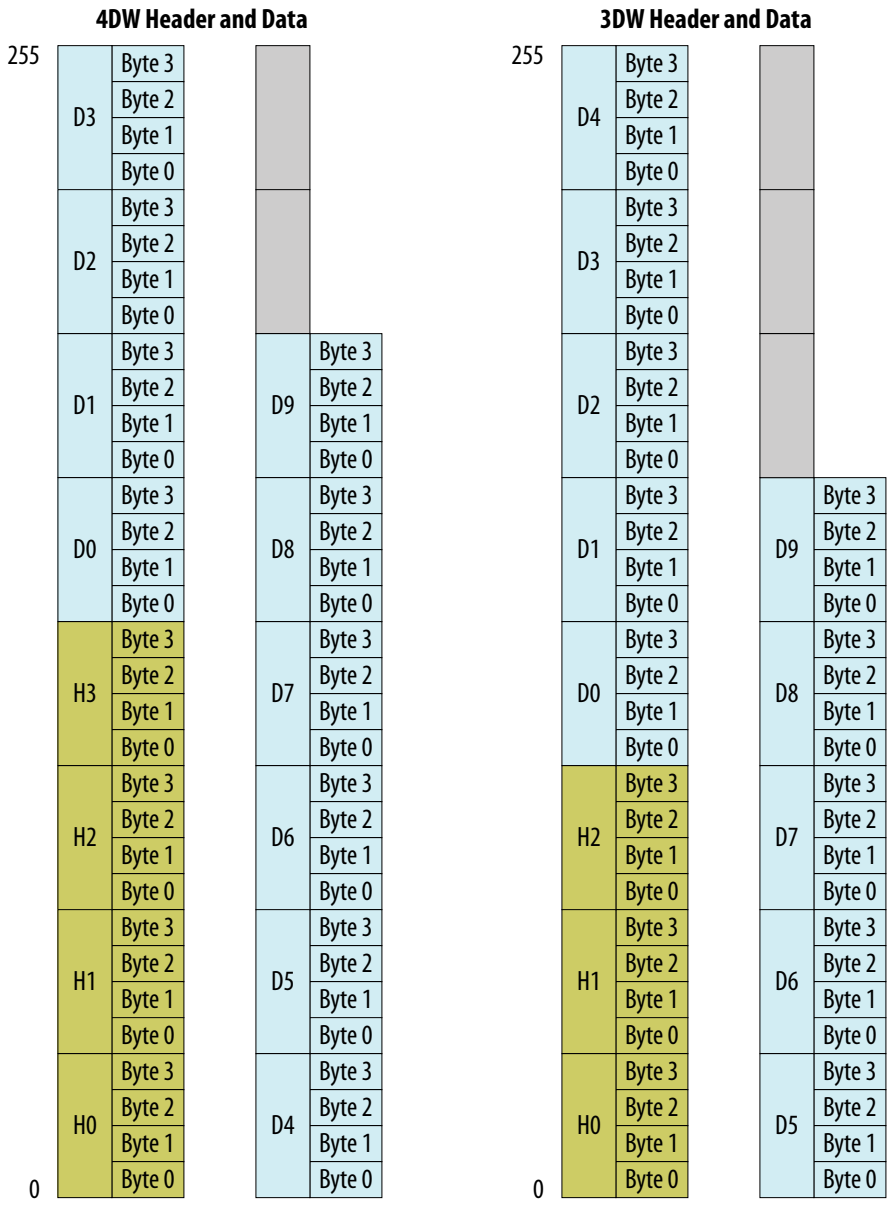
Table 1. Comparison of Stratix 10 and Stratix V or Arria 10 Parameters

Feature	Stratix 10	Stratix V or Arria 10	Comments
Supported Link Widths	1,2,4,8,16	1,2,4,8	
Interface widths	256 bits only	Avalon-ST: 64-, 128-, or 256-bit	You can create 64- or 128-bit adapters.
Packets per cycle	Single packet per cycle on the Avalon-ST interface	One or two packets per cycle on the Avalon-ST	Arria 10 and Stratix V 256-bit interface support the start of one packet and the end of the previous packet in the same cycle. Stratix 10 supports only one packet's data per cycle.
Data layout	Data packed on the Avalon-ST interface. No gaps between header and data. Refer to the *Figure 1. Three and Four double word (dword) Header and Data on the TX and RX Interfaces* below.	Data is quad word (qword) aligned. If the header presents an address that is not qword aligned, the Hard IP block shifts the data within the qword to achieve the qword alignment.	Unaligned data increases throughput. However, it is not backwards compatible with the previous implementations for Arria 10 and Stratix V devices.
Completion Timeout	Supports all Completion Timeout ranges. However, the parameter editor does not include a GUI parameter to select the desired range.	The parameter editor includes a GUI parameter to select the timeout.	You can specify the Completion Timeout Value in the `Device Control 2` register. Refer to *Section 2.2.9* of the *PCI Express Base Specification Revision 3.0*.
Completion Timeout disable	A **Disable Completion Timeout** parameter is not available.	The GUI includes a **Disable Completion Timeout** parameter on the **PCI Express/PCI Capabilities** tab to disable the Completion Timeout capability.	Completion Timeouts are always enabled. You may choose to ignore this information.
Error reporting	The GUI does not include an **Error Reporting** tab.	The GUI includes the following parameters to enable error checking: • **Enable Advanced Error Reporting (AER)** • **Enable ECRC Checking** • **Enable ECRC Generation**	The Stratix 10 implementation enables the following error reporting capabilities by default: • Advanced Error Reporting • ECRC checking • ECRC generation You can turn off optional error conditions in the the PCI Express Advanced Error Reporting Extended

continued...

Feature	Stratix 10	Stratix V or Arria 10	Comments
			Capability Structure. For more information refer to *Section 7.10 Advanced Error Reporting Extended Capability Header* of the *PCI Express Base Specification Revision 3.0*.
ECRC forwarding	Not available	The GUI includes the **Enable ECRC forwarding on the Avalon-ST interface** to forward the ECRC.	Not available.
RX Completion buffer overflow tracking	Not available	The GUI includes the **Track RX completion buffer overflow on the Avalon-ST interface** to track Completion buffer overflow.	No direct way to track how close the RX buffer is to overflow. Overflows are logged in the AER status register.
DLL Active Reporting	Not available	The GUI includes an option for **Data Link Layer (DLL) Active Reporting**.	
Surprise Down Reporting	Not available	GUI includes option for **Surprise down reporting**.	
Configuration via Protocol (CvP)	Not available	GUI includes option to **Enable configuration via protocol (CvP)**.	CvP will be available in a future release.
Reset control	Includes hard reset controller	Arria 10: includes hard reset controller. Stratix V: Gen 1 has a hard reset controller. Gen2 and Gen3 have a soft reset controller.	
Dynamic design example generation	Dynamic example Qsys system connects target design to a downstream DUT	Arria 10: Dynamic example Qsys system connects target design to a downstream DUT. Stratix V: Provides many static example designs.	Dynamic example designs reflect the parameters you choose in the parameter editor. Static example designs have fixed parameters.

Figure 1. Three and Four Dword Header and Data on the TX and RX Interfaces



3. Interface and Signal Differences

3.1. Clock Interface

Table 2. Clock Interface

Stratix 10	Arria 10, Stratix V	Comments
Not available	`pld_clk`	Stratix 10: There is no `pld_clk` input to the hard IP core. The hard IP drives a clock to the application. You must provide clock crossing logic if it is required. Arria 10 and Stratix V: `coreclkout_hip`, the fixed frequency clock for the DLL and Transaction Layer (TL) can drive `pld_clk`. If a different clock source drives the application logic, it must be equal to or faster than `coreclkout_hip`.

3.2. Avalon-ST TX Interface

Table 3. Avalon-ST TX Interface

Stratix 10	Arria 10, Stratix V	Comments
`tx_st_data[255:0]`	`tx_st_data[`*n*`-1:0]`	Stratix 10: Support 256 bits only. Arria 10 and Stratix V: Support 64, 128, and 256 bits.
`tx_st_sop`	`tx_st_sop[`*n*`-1:0]`	Stratix 10: Supports a single packet per cycle. Arria 10 and Stratix V: When using a 256-bit Avalon-ST interface with multiple packets per cycle enabled, the following encodings apply for `tx_st_sop[`*n*`-1:0]`: • 2'b01: the TLP starts with `tx_st_data[127:0]` • 2'b10: the TLP starts with `tx_st_data[255: 128]`
`tx_st_eop`	`tx_st_eop[`*n*`-1:0]`	Stratix 10: Supports a single packet per cycle. Arria 10 and Stratix V: When using a 256-bit Avalon-ST interface with multiple packets per cycle, the following encodings apply for `tx_st_eop[`*n*`-1:0]`: • 2'b01: the TLP ends with `tx_st_data[127:0]` • 2'b10: the TLP ends with `tx_st_data[255: 128]`
`tx_st_ready` `tx_st_valid`	`tx_st_ready` `tx_st_valid`	Stratix 10: The ready latency is 3 clock cycles. Arria 10 and Stratix V: The ready latency is 2 clock cycles.

continued...

Stratix 10	Arria 10, Stratix V	Comments
tx_st_err	tx_st_err	Stratix 10: The application asserts tx_st_err to nullify the currently transmitting TLP. This signal must be asserted along with tx_st_eop. The erroneous TLP is ignored if tx_st_sop, tx_st_eop, and tx_st_err assert simultaneously. Arria 10 and Stratix V: The application asserts tx_st_err to nullify a TLP. tx_st_err must be asserted after the tx_st_sop cycle and before the tx_st_eop cycle. Consequently, tx_st_err is not available for 1-2 cycle TLPs. For 256-bit data, when you turn on Enable multiple packets per cycle, bit 0 applies to the entire bus tx_st_data[255:0]. Bit 1 is not used.
tx_par_err	tx_par_err[1:0]	Stratix 10: The TL or DLL asserts the tx_par_err signal to indicate a parity error. Arria 10 and Stratix V: The following encodings apply for the tx_par_err[1:0] signal: • 2'b01: The TL detects a parity error • 2'b10: The DLL detects a parity error
Not available.	tx_st_empty[1:0]	Stratix 10: Not supported. Arria 10 and Stratix V: Specifies the number of empty quad words (qwords) during cycles that contain the tx_st_eop[<*n*>-1:0] signal.

3.3. Avalon-ST TX Credit Interface

Table 4. Avalon-ST TX Interface

Stratix 10	Arria 10, Stratix V	Comments
tx_ph_cdts[7:0]	tx_cred_hdr_fc[7:0] tx_cred_fc_sel [1:0] = 2'b00	Stratix 10: Posted header credit limit value for posted requests. Arria 10 and Stratix V: tx_cred_hdr_fc[7:0]: specifies the posted data credit limit value when the application logic sets tx_cred_fc_sel [1:0] = 2'b00.
tx_pd_cdts[11:0]	tx_cred_data_fc[7:0] tx_cred_fc_sel [1:0] = 2'b00	Stratix 10: Posted data credit limit value for posted requests. Arria 10 and Stratix V: tx_cred_data_fc[7:0]: Specifies the posted data credit limit value when the user application logic sets tx_cred_fc_sel [1:0] = 2'b00.
tx_nph_cdts[7:0]	tx_cred_hdr_fc[7:0] tx_cred_fc_sel [1:0] = 2'b01	Stratix 10: Non-posted header credit limit value for non-posted requests. Arria 10 and Stratix V: tx_cred_hdr_fc[7:0]: Specifies the non-posted header credit limit value when application logic sets tx_cred_fc_sel = 2'b01.
tx_npd_cdts[11:0] tx_st_valid	tx_cred_data_fc[7:0] tx_cred_fc_sel [1:0] = 2'b01	Stratix 10: Non-posted data credit limit value for non-posted requests. Arria 10 and Stratix V: tx_cred_hdr_fc[7:0]: Specifies the non-posted data credit limit value when application logic sets tx_cred_fc_sel = 2'b01 .

continued...

Stratix 10	Arria 10, Stratix V	Comments
`tx_cplh_cdts[7:0]`	`tx_cred_hdr_fc[7:0]` `tx_cred_fc_sel [1:0] = 2'b10`	Stratix 10: Specifies the Completion header credit limit value for Completions. A value of 0 indicates infinite Completion header credits for Endpoints. Arria 10 and Stratix V: `tx_cred_hdr_fc[7:0]`: Specifies the Completion header credit limit value when the application logic sets `tx_cred_fc_sel = 2'b10` .

`tx_hdr_cdts_consumed`
`tx_data_cdts_consumed`
`tx_cdts_data_value`
`tx_cdts_type[1:0]`
	`tx_cred_fc_hip_cons[5:0]`	Stratix 10: The hard IP asserts the `tx_hdr_cdts_consumed` signal to indicate header credit consumption by the user logic (not by the hard IP). The `tx_cdts_type` signal specifies the credit type.
		The hard IP asserts the `tx_data_cdts_consumed` signal to indicate data credit consumption by the user logic (not by the hard IP). The data credits consumed equals `tx_cdts_data_value`+1. The `tx_cdts_type` signal specifies the credit type.
		To optimize performance, the user application logic can track its own consumed credits using this interface. Requesting TLP transmission based on the credits available, calculated from credit limit and credit consumed, allows the application logic to minimize the latency due to `tx_st_ready` deassertion.
		For more information, refer to *TX Credit Interface* in the *Intel Stratix 10 Avalon-ST Interface for PCIe Solutions User Guide*.
		Stratix V, Arria 10: The hard IP asserts `tx_cred_fc_hip_cons` for 1 cycle each time it consumes a credit. These credits are from Messages it generates for 2 purposes:
		• To respond to Configuration Requests
		• To send error messages
		This signal is not asserted when application logic consumes a credit.
		The six bits of this vector correspond to the following six types of credit types:
		• [5]: Posted headers
		• [4]: Posted data
		• [3]: Non-posted header
		• [2]: Non-posted
		• [1]: Completion header
		• [0]: Completion data

Not available.	`tx_cred_fc_infinite[5:0]`	Arria 10 and Stratix V: When asserted indicates infinite credit for the corresponding credit types. The six bits of this vector correspond to the following six credit types: • [5]: Posted headers • [4]: Posted data • [3]: Non-posted header • [2]: Non-posted • [1]: Completion header • [0]: Completion data

Related Information

TX Credit Interface
For more comprehensive information about the TX credit interface in Stratix 10 devices.

3.4. Avalon-ST RX Interface

Table 5. Avalon-ST RX Interface

Stratix 10	Arria 10, Stratix V	Comments
rx_st_data[255:0]	rx_st_data[<*n*>-1:0]	Stratix 10: Supports 256 bits only. Arria 10 and Stratix V: Support 64, 128, and 256 bits.
rx_st_parity[31:0]	rx_st_parity[<*n*>-1:0]	Stratix 10: Only supports 256-bit data bus. Arria 10, Stratix V: <*n*> = 8, 16, or 32. Supports 64, 128, and 256-bit data bus.
rx_par_err	Not available	Stratix 10: Asserted for 1 cycle to indicate a parity error on the RX data bus. Arria 10, Stratix V: Not supported.
rx_st_sop	rx_st_sop[<*n*>-1:0]	Stratix 10: Supports a single packet per cycle. Arria 10 and Stratix V: When using a 256-bit Avalon-ST interface with multiple packets per cycle, the following encodings apply for rx_st_sop[<*n*>-1:0]: • 2'b01: The TLP starts with rx_st_data[127:0] • 2'b10: The TLP starts with rx_st_data[255: 128]
rx_st_eop	rx_st_eop[<*n*>-1:0]	Stratix 10: Supports a single packet per cycle. Arria 10 and Stratix V: When using a 256-bit Avalon-ST interface with multiple packets per cycle, the following encodings apply for rx_st_eop[<*n*>-1:0]: • 2'b01: The TLP ends with rx_st_data[127:0] • 2'b10: The TLP ends with rx_st_data[255: 128]
rx_st_empty[2:0]	rs_st_empty[1:0]	Stratix 10: Indicates number of empty dwords. Arria 10, Stratix V: Indicates number of empty qwords.
rx_st_bar_range[2:0]	rx_st_bar[7:0]	Stratix 10: Uses binary encoding. Indicates the bar range for the current request. The following encodings are for Endpoints: • 3'b000: Memory Bar 0 • 3'b001: Memory Bar 1 • 3'b010: Memory Bar 2 • 3'b011: Memory Bar 3 • 3'b100: Memory Bar 4 • 3'b101: Memory Bar 5 • 3'b110: Unused • 3'b111: Expansion ROM Bar 0 Root Ports are not supported in the current release. Arria 10 and Stratix V: U bit encoding. The following encodings are defined for Endpoints: • Bit 0: BAR 0 • Bit 1: BAR 1 • Bit 2: BAR 2 • Bit 3: BAR 3 • Bit 4: BAR 4

continued...

Stratix 10	Arria 10, Stratix V	Comments
		• Bit 5: BAR 5 • Bit 6: Expansion ROM • Bit 7: Reserved The following encodings are defined for Root Ports: • Bit 0: BAR 0 • Bit 1: BAR 1 • Bit 2: Primary Bus number • Bit 3: Secondary Bus number • Bit 4: Secondary Bus number to Subordinate Bus number window
Not available	`rx_st_mask`	Stratix 10: You cannot stall non-posted RX TLPs. Consequently, you must implement an RX buffer.
`rx_st_ready` `rx_st_valid`	`rx_st_ready` `rx_st_valid`	Stratix 10: The ready latency is 17 clock cycles. To achieve optimal performance, the application logic must include a receive buffer large enough to avoid the deassertion of `rx_st_ready`. Arria 10 and Stratix V: The ready latency is 2 clock cycles.
Not available	`rx_st_err`	Stratix 10: Not supported. Use the error interface signals such as `derr_uncor_ext_rcv` to determine error status. Arria 10 and Stratix V: `rx_st_err` is an optional signal that indicates an uncorrectable error correction code (ECC) error in the internal RX buffer.

3.5. Status and Link Training Interface

Table 6. Avalon-ST TX Interface

Stratix 10	Arria 10, Stratix V	Comments
Not available	`rxfc_cplbuf_ovf`	Stratix 10: This signal is not supported. RX buffer overflow is logged in the AER status register. Arria 10, Stratix V: Asserted to indicate the internal RX buffer has overflowed.
Not available	`ko_cpl_spc_header` `ko_cpl_spc_data`	Stratix 10: The RX buffer is larger. Allocation is fixed as shown in following table. • Posted: — PH: 127 — PD: 750 • Non-Posted: — NPH: 115 — NPD: 230 • Completions: — CPLH: 770 — CPLD: 2400 These are equivalent credits. Advertised credits are infinite. Arria 10, Stratix V: RX buffer allocation varies. Signals indicate the total number of 16-byte Completions that can be stored in the RX buffer.

continued...

Stratix 10	Arria 10, Stratix V	Comments
		Application logic can use this information to build circuitry to prevent RX buffer overflow for Completions.
ltssmstate[5:0]	ltssmstate[4:0]	Stratix 10: Provides finer granularity and a different encoding. For example, L0 is 0x11 instead of 0x0F.

3.6. Configuration Status Interface

Table 7. Avalon-ST TX Interface

Stratix 10	Arria 10, Stratix V	Comments
tl_cfg*	tl_cfg*	Stratix 10: Data is valid every clock cycle. You do not need to use multi-cycle sampling to capture data. Stratix V, Arria 10: The tl_cfg_* interface includes multi-cycle paths. Depending on the parameterization, the tl_cfg_add and tl_cfg_ctl signals update every 4 or 8 coreclkout_hip cycles. Refer to the *Configuration Space Register Access Timing* section in the *Stratix V Avalon-ST Interface for PCIe Solutions User Guide* or *Intel Arria 10 Avalon-ST Interface for PCIe Solutions User Guide* for more information.
tl_cfg_add[4:0]	tl_cfg_add[3:0]	Stratix 10: 24 register entries available. Stratix V, Arria 10: 16 register entries available. Register contents varies between devices. Refer to the *Stratix 10 Avalon-ST Interface for PCIe Solutions User Guide* Section 6.1.10 for more information.
tl_cfg_func[1:0]	Not available	Stratix 10: Specifies the PF or VF to which tl_cfg* applies, providing more visibility than Arria 10 or Stratix V
Not available	tl_cfg_sts[52:0]	Stratix 10: No fixed cfg status port. You can use an Avalon-MM interface to access the full Configuration Space.
tl_cfg_ctl	tl_cfg_ctl	The mapping of Stratix 10 is different than the mapping of Stratix V and Arria 10. Refer to *Transaction Layer Configuration Space Interface* in *Stratix 10 Avalon-ST Interface for PCIe Solutions User Guide* for more information.

Related Information

- Configuration Space Register Access Timing in Arria 10 Devices
- Configuration Space Register Access Timing in Stratix V Devices
- Transaction Layer Configuration Space Interface
  Defines the tl_cfg interface for Stratix 10 devices.

3.7. Interrupt Interface

Table 8. Interrupt Interface

Stratix 10	Arria 10, Stratix V	Comments
Not available	`app_int_ack`	Stratix 10: Legacy interrupts do not generate an acknowledge. You must parse the TLPs.
`int_status[7:0]`	`int_status[3:0]`	Stratix 10: Provides 4 additional interrupt status signals. These signals specify the following interrupts: • [4]: RC AER error interrupt status • [5]: Root complex PME interrupt status • [6]: Asserted when hot plug event occurs and PME is enabled • [7]: Hot plug event interrupt status
`int_status_common[2:0]`	Not available	Stratix 10: Provides 3 additional status indications: • [0]: Interrupt status for the `Link Autonomous Bandwidth Status` register • [1]: Interrupt status for the `Link Bandwidth Management Status` register • [2]: Interrupt status for `Link Equalization Request` bit in the `Link Status` register

3.8. Error Interface

Table 9. Error Interface

Stratix 10	Arria 10, Stratix V	Comments
`serr_out[1:0]`	`serr_out`	Stratix 10: Additional signal specifies the PF that detected a system error. More PFs are available in Stratix 10. Note: This signal is valid only for Root Ports.
`app_err*`	`cpl_err[6:0]`	Stratix 10: Signals specify different error types to the Stratix 10 Hard IP for PCIe IP core. Refer to *Transaction Layer Configuration Space Interface* in the *Stratix 10 Avalon-ST Interface for PCIe Solutions User Guide* for more information.
`derr_uncor_ext_rcv`	Not available	Stratix 10: Indicates an uncorrectable 2-bit ECC error in the RX buffer. Stratix V, Arria 10: Not supported.
Not available	`cpl_pending`	Stratix 10: There is no equivalent signal in Stratix 10. You may need to track pending Completions in your application.

Related Information

Transaction Layer Configuration Space Interface
Use the Transaction Layer Configuration Space interface signals in conjunction with the `app_err*` signal to debug TLP transmission problems.

3.9. Reset

Table 10. Reset Interface

Stratix 10	Arria 10, Stratix V	Comments
app_init_rst	Not available	Stratix 10: Request from the application layer for a hot reset to downstream devices. Stratix V, Arria 10: Not supported.

3.10. PHY Interface for PCI Express (PIPE) Interface

Table 11. PIPE Interface

Stratix 10	Arria 10, Stratix V	Comments
rxeqeval rxeqinprogress invalidreq dirfeedback[5:0] (H-Tile only) sim_pipe_mask_tx_pll_lock	Not available (all)	Stratix 10: Adding these signals and removing others from prior device generations, provides a PIPE 3.0 compliant interface.
sim_ltssmstate[5:0]	sim_ltssmstate[4:0]	Stratix 10: Provides finer granularity and different encoding. For example, L0 is now 0x11 instead of 0x0F.
Not available	eidleinfersel0[2:0]	Stratix 10: This signal is not required because Stratix 10 is PIPE compliant. Arria 10, Stratix V: Indicates Electrical Idle inference mechanism selection.

3.11. Test Interface

Table 12. Test Interface

Stratix 10		Arria 10, Stratix V	Comments
x16 test_in[66:0] aux_test_out[6:0] test_out[66:0]	x8 or smaller test_in[66:0] aux_test_out[66:0] Not available	test_in[63:0] diag_ctrl_bus[2:0]	Stratix 10: test_out bus is available only in x16 configurations. aux_test_out[6:0] is available only in x8 or smaller configurations. Stratix 10 has a different mapping for the test interface. Contact Intel for mapping information.
Not available		LMI Interface	Stratix 10: Root Ports can use an Avalon-MM interface to write error log descriptor information to the TLP header log registers. Endpoints do not have direct access. (Root Ports are not supported in the current release.)
Message Signaled Interface (MSI) Interface		MSI interface	The tl_cfg interface no longer provides access to the information you may need to create MSI packets. You may use the Hard IP Reconfiguration interface to access this information.

. Document Revision History for AN-791

Date	Version	Changes
May 2017	2017.05.08	Initial release.