Single-Port Triple-Speed Ethernet and On-Board PHY Chip Reference Design

2015.12.14

AN-647 Subscribe Send Feedback

The Single-Port Triple-Speed Ethernet and On-Board PHY Chip reference designs provide flexible test and demonstration platforms on which you can control, test, and monitor the Ethernet operations using system loopbacks.

The Single-Port Triple-Speed Ethernet and On-Board PHY Chip reference designs demonstrate Ethernet operations of the Altera® Triple-Speed Ethernet IP core with on-board Marvell 88E1111 PHY chips. In these reference designs, the Triple-Speed Ethernet IP core connects to the on-board PHY chip through either the Reduce Gigabit Media Independent Interface (RGMII) or the Serial Gigabit Media Independent Interface (SGMII).

The reference designs offer the following features:

- Minimal hardware requirement for a complete test.
- Implementation of one Triple-Speed Ethernet IP core instance supporting 10/100/1000-Mbps Ethernets operations with RGMII or SGMII with auto-negotiation.
- Support for programmable test parameters such as number of packets, packet length, source and destination MAC addresses, and payload-data type.
- Support for sequential random bursts test that enables the configuration of each burst for the number of packets, payload-data type, and payload size.
- Demonstration of Ethernet packets transmission and reception through internal loopback path at the maximum theoretical data rates without errors.
- Support for gathering throughput statistics.
- Support for System Console user interface.

System Architecture

The reference designs demonstrate fully operational subsystems that integrate the Triple-Speed Ethernet IP core for Ethernet applications.

Figure 1: Reference Design with SGMII Interface—Arria 10 Devices

The figure shows a high-level block diagram of the reference design implementing the SGMII interface with Arria 10 devices.

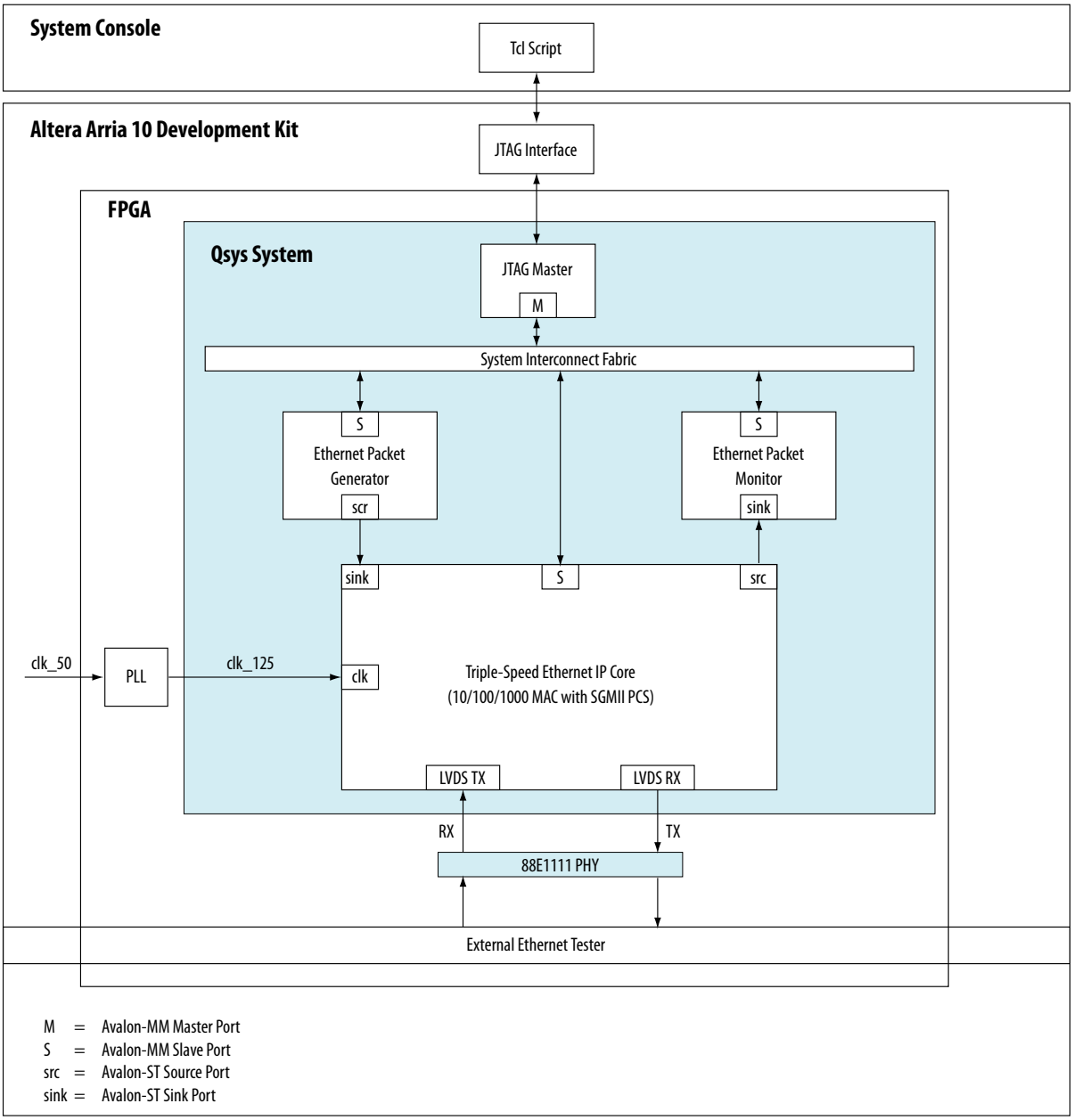


Figure 2: Reference Design with SGMII Interface—Stratix V GX Devices

The figure shows a high-level block diagram of the reference design implementing the SGMII interface with Stratix V GX devices.

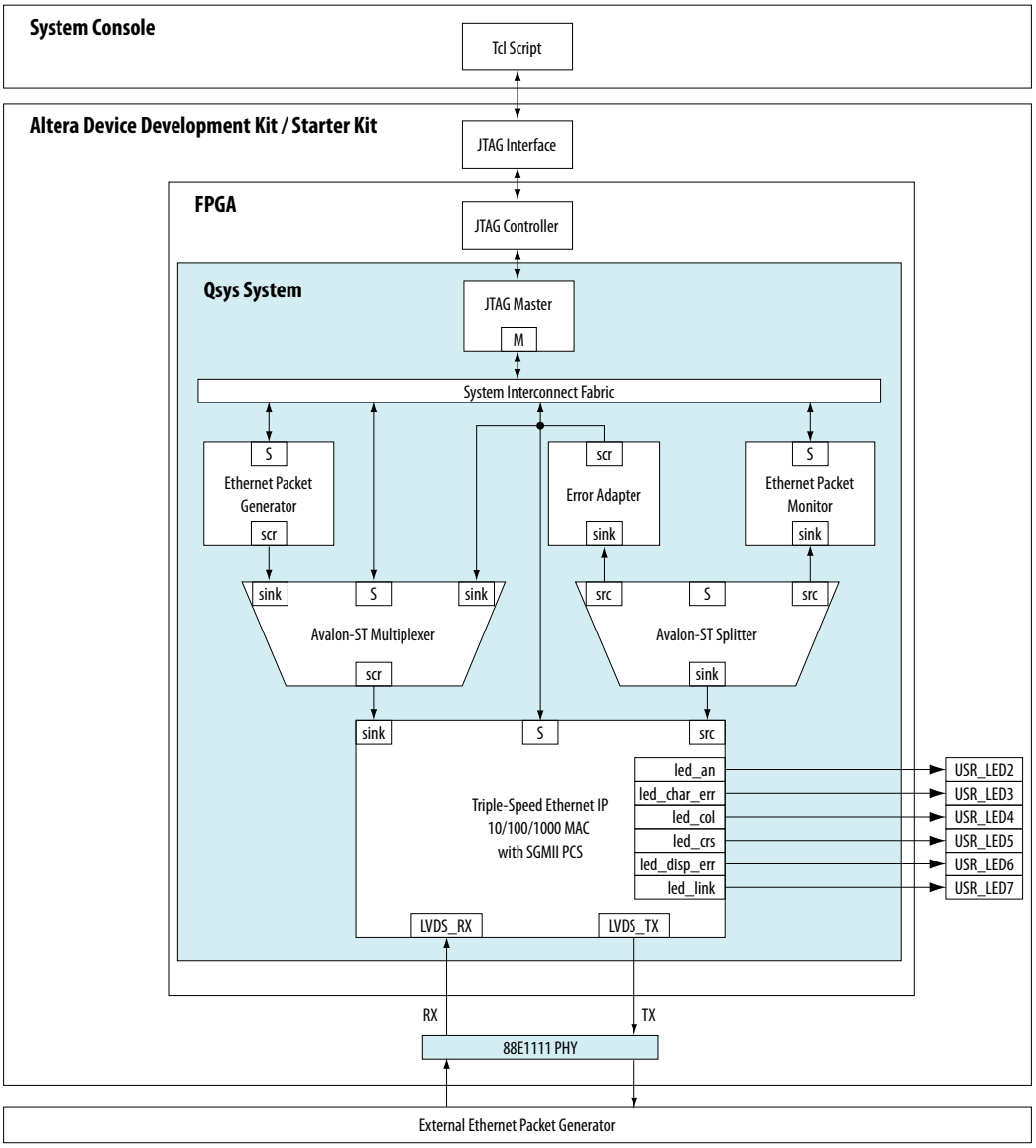
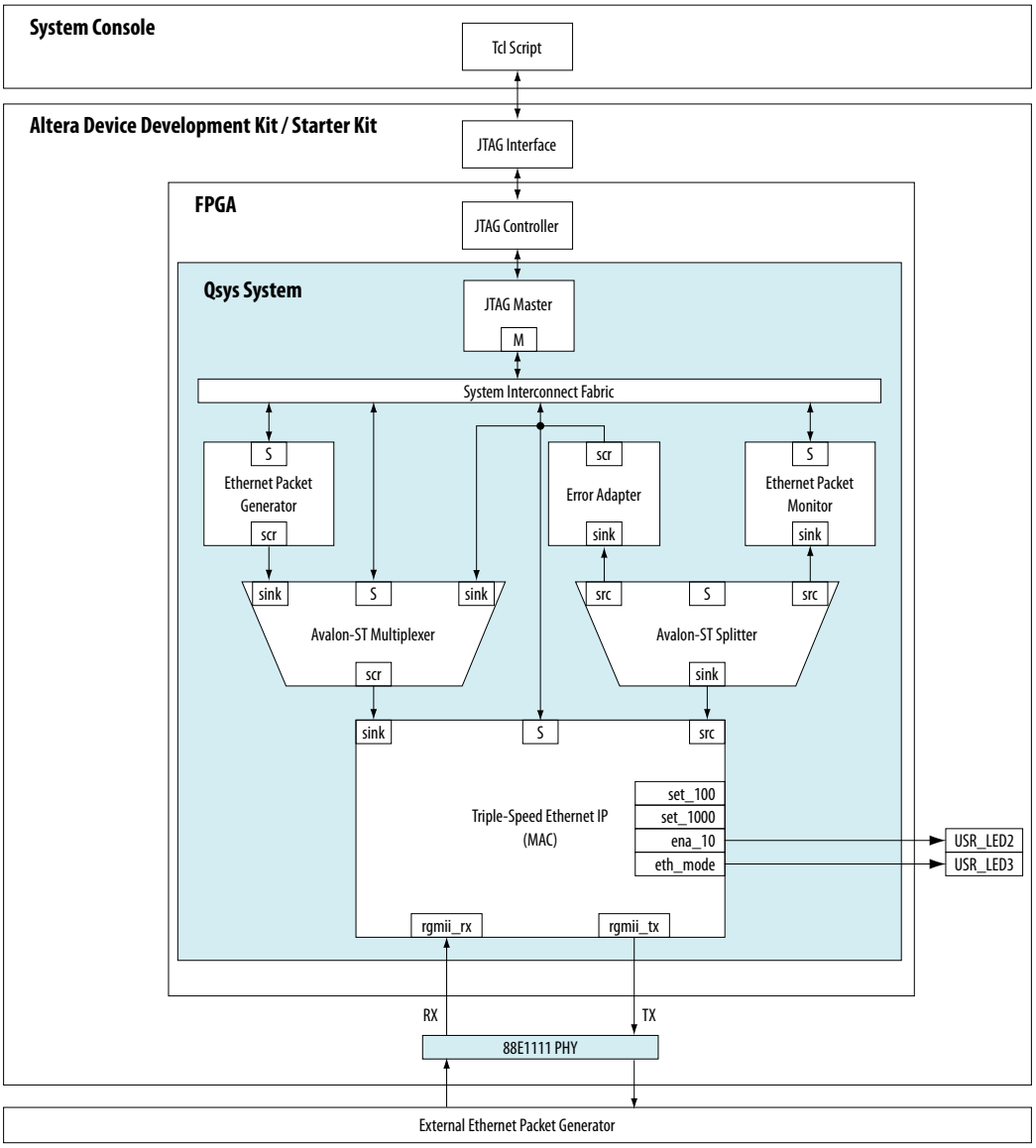


Figure 3: Reference Design with RGMII Interface—Arria V GX and Cyclone V GX Devices

The figure shows a high-level block diagram of the reference design implementing the RGMII interface with Arria V GX and Cyclone V GX devices.



Design Components

This topic describes each component of the Single-Port Triple-Speed Ethernet reference designs.

Table 1: Components of the Single-Port Triple-Speed Ethernet Reference Designs

Component	Description
Phase-Locked Loop (PLL) Core	• This IP core generates a 125-MHz PLL output clock (`clk_125M`). • This output clock is the system-wide clock source for the Qsys system. • All the components in these reference designs use the 125-MHz clock from the PLL core.
JTAG to Avalon Master Bridge Core	• This IP core provides a connection between the System Console and Qsys system through the physical interfaces. • The System Console can initiate Avalon Memory-Mapped (Avalon-MM) transactions by sending encoded streams of bytes through the bridge's physical interfaces.
Triple-Speed Ethernet IP Core	• This IP core provides an integrated Ethernet MAC, PCS, and PMA solution for Ethernet applications. • The Triple Speed Ethernet IP core transmits Ethernet packets from Avalon Streaming (Avalon-ST) interface to a 1.25-Gbps serial transceiver interface and receives packets from the opposite direction.
Ethernet Packet Generator	This Qsys custom component generates Ethernet packets. Refer to **Ethernet Packet Generator** on page 6 for more information.
Ethernet Packet Monitor	This Qsys custom component verifies the payload of all receive packets, indicates the validity of the packets, and collects statistics about each packet, such as the number of bytes received. Refer to **Ethernet Packet Monitor** on page 8 for more information.
Error Adapter	• This Qsys custom component connects mismatched Avalon-ST source and sink interfaces. • The adapter allows you to connect a data source to a data sink of differing byte sizes. For TX-to-RX Avalon-ST reverse loopback in these design examples, `ff_tx_err` is a 1-bit error signal and `rx_err` is a 6-bit error signal. The adapter ensures that the per-bit error information provided by `ff_tx_err` at the source interface connects correctly to the `rx_err` signal. • The adapter connects matching error conditions that are handled by the source and the sink. **Note:** Not applicable for Arria 10 designs.

Component	Description
Avalon-ST Multiplexer	• This Qsys custom component accepts data on its two Avalon-ST sink interfaces and multiplexes the data for transmission on its Avalon-ST source interface. • One Avalon-ST sink interface connects to the source of the Ethernet Packet Generator for forward loopback while the other sink interface connects to the source of the Error Adapter for reverse loopback. • The Avalon-ST source interface sends Ethernet packets to the Triple-Speed Ethernet IP Core. **Note:** Not applicable for Arria 10 designs.
Avalon-ST Splitter	• This Qsys custom component accepts data on its Avalon-ST sink interface and splits the data for transmission on its two Avalon-ST source interfaces. • The Avalon-ST sink interface receives Ethernet packets from the Triple-Speed Ethernet IP Core. One Avalon-ST source interface connects to the sink of the Ethernet Packet Generator for forward loopback while the other source interface connects to the sink of the Error Adapter for reverse loopback. **Note:** Not applicable for Arria 10 designs.

Related Information

- **SPI Slave/JTAG to Avalon Master Bridge Cores**
  Provides more information about JTAG to Avalon Master Bridge Core.
- **Triple-Speed Ethernet Interface Signals**
  Provides more information about the `ff_tx_err` and `rx_err` error signals.

Ethernet Packet Generator

Figure 4: Ethernet Packet Generator Block Diagram

This figure shows a high-level block diagram of the Ethernet Packet Generator module.

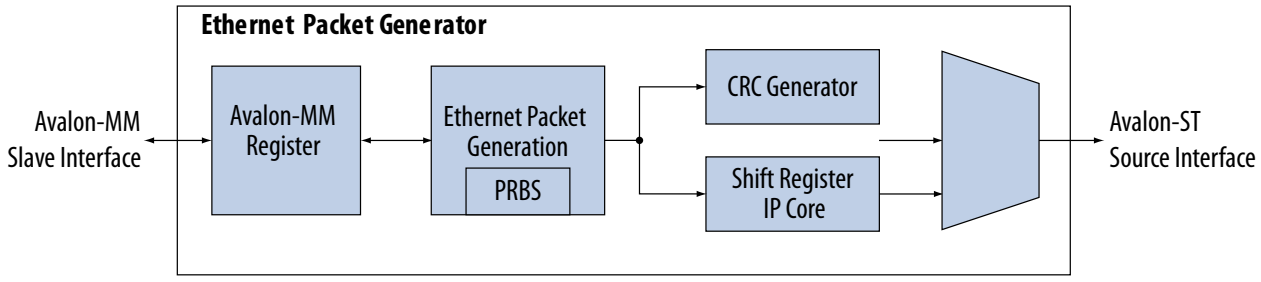
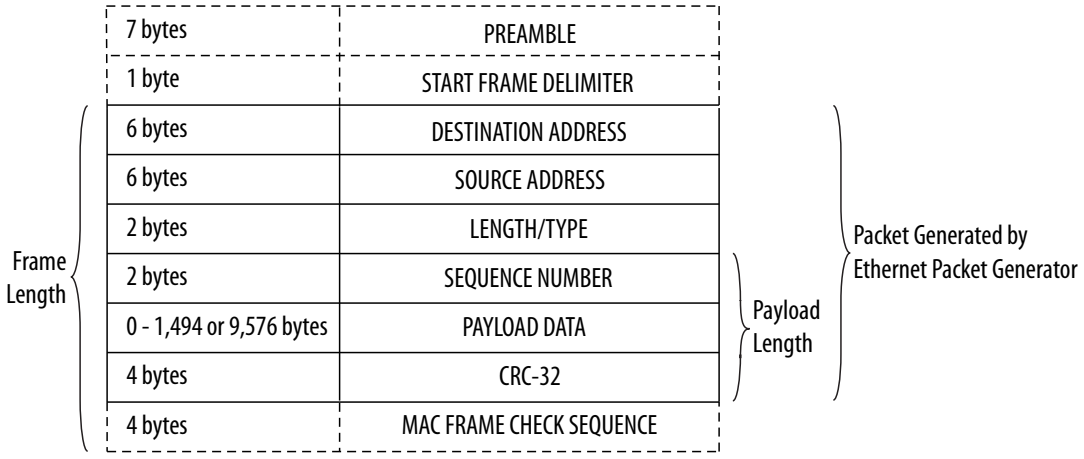


Table 2: Components of the Ethernet Packet Generator

Component	Description
Ethernet Packet Generation Block	• The Ethernet packet generation block generates an Ethernet packet header, data payload and running sequence number for each packet. • The Ethernet packet generation block sends the packets to the CRC Generator and the RAM-based Shift Register IP core.
CRC Generator	• The CRC Generator calculates the CRC-32 checksum for the packet and the RAM-based shift register megafunction stores the packet until the checksum is available. • After the generator merges the valid CRC-32 checksum with the packet stream, it sends the complete packet to the Avalon-ST source interface.
Avalon-MM Registers	• The Avalon-MM slave interface provides access to the Avalon-MM register interface. Using a Tcl script, you can configure the Avalon-MM configuration registers. • The Avalon-MM status registers provide the status of the transmit operation and report the number of packets that were successfully transmitted. Refer to **Ethernet Packet Generator Configuration Registers** on page 10 for more information.
Shift Register (RAM-based) IP Core	The Shift Register IP core implements a shift register with taps. Refer to the *RAM-Based Shift Register IP Core User Guide* for more information.

Figure 5: Ethernet Packet Generator Output Frame Format

The figure shows the format the Avalon-ST source interface streams Ethernet packets. The generated packets do not include the 7-byte preamble, 1-byte start frame delimiter (SFD) and 4-byte MAC-calculated Frame Check Sequence (FCS) fields.



Related Information

RAM-Based Shift Register (ALTSHIFT_TAPS) IP Core User Guide

Ethernet Packet Monitor

Figure 6: Ethernet Packet Monitor Block Diagram

This figure shows a high-level block diagram of the Ethernet Packet Monitor module.

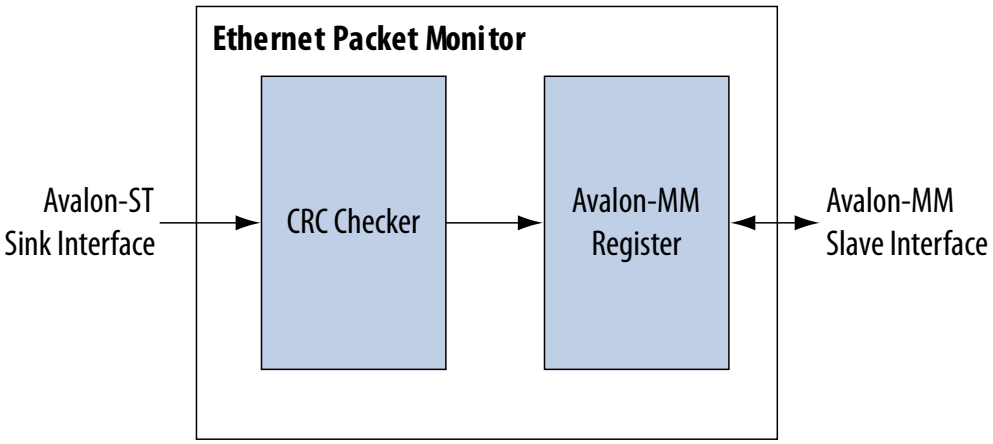


Table 3: Components of the Ethernet Packet Monitor

Component	Description
CRC Checker	• The Avalon-ST sink interface accepts Ethernet packets and sends the packets to the CRC Checker. • The CRC Checker computes the CRC-32 checksum of the receive packet and verifies it against the CRC-32 checksum field in the packet. • The checker then outputs a status signal that identifies whether the packet received is good or corrupted, and updates the statistics counters accordingly.
Avalon-MM Registers	• The Avalon-MM slave interface provides access to the Avalon-MM register interface. • Using a Tcl script, you can configure the Avalon-MM configuration registers to specify the number of packets the monitor expects to receive. • The Avalon-MM status registers provide the status of the receive operation and report the number of good and bad packets received, the number of bytes received, and the number of clock cycles. This information is used to calculate the performance and throughput rate of the reference designs. Refer to **Ethernet Packet Monitor Configuration Registers** on page 13 for more information.

Interface Signals

The top-level signals of the reference designs show the behavior of the specific operations.

Clock and Reset Signals

Table 4: RGMII Clock and Reset Signals

Signal	Description
clk_clk	Reference design clock. The clock is derived from the PLL.
triple_speed_ethernet_0_pcs_mac_rx_clock_connection_clk	RGMII receive clock. The clock is sourced from the on-board PHY chip.
triple_speed_ethernet_0_pcs_mac_tx_clock_connection_clk	RGMII transmit clock. The clock is sourced from the clock multiplexer which is sourced from the PLL.
reset_reset_n	Single reset signal for all logic in the reference design.

Table 5: SGMII Clock and Reset Signals

Signal	Description
clk_clk	Reference design clock. The clock is derived from the PLL.
triple_speed_ethernet_0_pcs_ref_clk_clock_connection_clk	Reference clock for the transceiver. The clock is sourced from the 125-MHz oscillator.
reset_reset_n	Single reset signal for all logic in the reference design. Connect this reset signal to the RESET push button (USER_PB0).

Triple-Speed Ethernet Component Signals

Table 6: RGMII Reference Design Signals

Signal	Description
triple_speed_ethernet_0_mac_rgmii_connection_rgmii_in	RGMII receive data bus. Connect this bus to the on-board PHY chip.
triple_speed_ethernet_0_mac_rgmii_connection_rx_control	RGMII receive control output signal. Connect this signal to the on-board PHY chip.
triple_speed_ethernet_0_mac_rgmii_connection_rgmii_out	RGMII transmit data bus. Connect this bus to the on-board PHY chip.
triple_speed_ethernet_0_mac_rgmii_connection_tx_control	RGMII transmit control output signal. Connect this signal to the on-board PHY chip.

Table 7: SGMII Reference Design Signals

Signal	Description
triple_speed_ethernet_0_serial_connection_rxp_0	SGMII receive serial data bus. Connect this bus to the on-board PHY chip.

Signal	Description
triple_speed_ethernet_0_serial_connection_txp_0	SGMII transmit serial data bus. Connect this bus to the on-board PHY chip.

Base Addresses and Configuration Registers

To access the configuration registers of the reference design components, use the base address of the component and the register offset.

Table 8: Base Addresses of Reference Design Components

Base Address	Name	Description
0x00000000	triple_speed_ethernet_0	Triple-Speed Ethernet
0x00000400	st_mux_2_to_1_0	Avalon-ST Multiplexer **Note:** Not applicable for Arria 10 designs.
0x00000800	eth_mon_0	Ethernet Packet Monitor
0x00000C00	eth_gen_0	Ethernet Packet Generator

Ethernet Packet Generator Configuration Registers

Table 9: Ethernet Packet Generator Configuration Registers

Byte Offset	Register	Bit Number	Bit Name	R/W	H/W Reset	Description
0x00	number_packet	31:0	–	RW	0x00	Specifies the total number of packets to be generated.

Byte Offset	Register	Bit Number	Bit Name	R/W	H/W Reset	Description
0x04	config_setting	0	LENGTH_SEL	RW	0x00	0: Fixed packet length 1: Random packet length
		14:1	PKT_LENGTH	RW	0x00	Specifies the fixed packet length. Valid values are between 24 to 9,600. Applicable only when you set bit 0 of this register to 0.
		15	PATTERN_SEL	RW	0x00	Specifies the data pattern for the random packet length. 0: Incremental—data starts from zero and increments by 1 in subsequent bytes. 1: Random.
		31:16	–	–	–	Reserved.
0x08	operation	0	START	RW	0x00	Set this bit to 1 to trigger packet generation. This bit clears as soon as packet generation starts.
		1	STOP	RW	0x00	Set this bit to 1 to stop packet generation. The generator completes the current packet before termination packet generation.
		2	TX_DONE	RO	0x00	A value of 1 indicates that the packet generator completes generating the total number of packets specified in the number_packet register. This bit clears each time packet generation triggers.
		31:3	–	–	–	Reserved.

Byte Offset	Register	Bit Number	Bit Name	R/W	H/W Reset	Description
0x10	source_addr0	31:0	–	RW	0x00	6-byte MAC address. • source_addr0/destination_addr0 = last four bytes of the address • Bits 0 to 15 of source_addr1/destination_addr1 = first two bytes of the address • Bits 16 to 31 of source_addr1/destination_addr1 = unused For example, if the source MAC address is 00-1C-23-17-4A-CB, you get the following assignments: • source_addr0 = 0x17231C00 • source_addr1 = 0x0000CB4A
0x14	source addr1	31:0	–	RW	0x00	
0x18	destination_addr0	31:0	–	RW	0x00	
0x1C	destination_addr1	31:0	–	RW	0x00	
0x24	packet_tx_count	31:0	–	–	–	Keeps track of the number of packets the generator successfully transmits. This register clears each time packet generation triggers.
0x30	rand_seed0	31:0	–	RW	0x00	The lower 32 bits of the random seed. Occupies bits 31:0 of the PBRS generator when you set the data pattern to random (bit 15 of the configuration register).
0x34	rand_seed1	31:0	–	RW	0x00	The middle 32 bits of the random seed. Occupies bits 63:32 of the PBRS generator when you set the data pattern to random (bit 15 of the configuration register).

Byte Offset	Register	Bit Number	Bit Name	R/W	H/W Reset	Description
0x38	rand_seed2	31:0	–	RW	0x00	The upper 32 bits of the random seed. Occupies bits 91:64 of the PBRS generator when you set the data pattern to random (bit 15 of the configuration register).

Ethernet Packet Monitor Configuration Registers

Table 10: Ethernet Packet Monitor Configuration Registers

Byte Offset	Register	Bit Number	Bit Name	R/W	H/W Reset	Description
0x00	number_packet	31:0	–	RO	0x00	Total number of packets the monitor expects to receive.
0x04	packet_rx_ok	31:0	–	RO	0x00	Total number of good packets received.
0x08	packet_rx_error	31:0	–	RO	0x00	Total number of packets received with error.
0x0C	byte_rx_count_0	31:0	–	RO	0x00	64-bit counter that keeps track of the total number of bytes received. • byte_rx_count_0 represents the lower 32 bits. • byte_rx_count_1 represents the upper 32 bits. Read byte_rx_count_0 followed by byte_rx_count_1 in the subsequent cycle to get an accurate count.
0x10	byte_rx_count_1	31:0	–	RO	0x00	

Byte Offset	Register	Bit Number	Bit Name	R/W	H/W Reset	Description
0x14	cycle_rx_count_0	31:0	–	RO	0x00	64-bit counter that keeps track of the total number of cycles the monitor takes to receive all packets.. • cycle_rx_count_0 represents the lower 32 bits. • cycle_rx_count_1 represents the upper 32 bits. Read byte_rx_count_0 followed by byte_rx_count_1 in the subsequent cycle to get an accurate count.
0x18	cycle_rx_count_1	31:0	–	RO	0x00	
0x1C	rx_control_status	0	START	RW	0x00	Set this bit to 1 to start packet reception. This bit clears when packet reception starts.
		1	STOP	RW	0x00	Set this bit to 1 to stop packet reception. This bit clears each time packet reception starts.
		2	RX_DONE	RO	0x00	A value of 1 indicates that the packet monitor has received the total number of packets specified in the number_packet register.
		3	CRCBAD	RO	0x00	A value of 1 indicates CRC error in the current packet received by the monitor.
		9:4	RX_ERR	RO	0x00	Receive error status. The rx_err[] signal of the Triple-Speed Ethernet IP core maps to this register.
		31:10	–	–	–	Reserved.

Running the Reference Designs

When the board is set up and the FPGA is configured, you can start running the design tests.

To run the tests, download and unzip the reference design files to your local directory.

Table 11: Reference Design Files

This table lists the reference design files.

Components	File	Description
Top Level Design File	**top.v**	Top-level entity file of the reference design for verification in the hardware.
	top_out.sdc	Quartus® Prime SDC constraint file for use with the TimeQuest timing analyzer.
	qsys_top.qsys	Qsys file of the reference design.
	pll.v	The IP-generated PLL file.
Ethernet Generator	**eth_gen_hw.tcl**	Qsys custom component which generates Ethernet packets.
	eth_gen.v	The top level file of the Ethernet Packet Generator.
	prbs23.v	Module to generate Pseudo-Random Bit Sequence 23 data.
	shiftreg_data.v	RAM-based shift register to delay transmit packet payload sending to Triple-Speed Ethernet TX FIFO interface for CRC checksum merging at EOP.
	shiftreg_ctrl.v	RAM-based shift register to store and delay control signals.
	crcgen_dat32.v	Module to generate checksum CRC32.
Ethernet Monitor	**eth_mon_hw.cl**	Qsys custom component which verifies the payload of all receive packets.
	eth_mon.v	Top level file of the Ethernet Packet Monitor.
	crcchk_dat32.v	Module to verify the data payload of the received packets.

Components	File	Description
CRC Generator/ Checke	**crc32.sdc**	Quartus Prime SDC constraint file for use with the TimeQuest timing analyzer.
	crc32_chk.v	Top level file of the CRC 32 checker.
	crc32_gen.v	Top level file of the CRC 32 generator.
	avalon_st_to_crc_if_ bridge.v	Module to convert the Avalon-ST signal to the CRC 32 calculator input signal.
	byte_endian_converter.v	Module to convert the input data byte to big or little endian.
	crc32_calculator.v	Module to calculate the crc32 checksum for the incoming data.
	crc_checksum_aligner.v	Module to align the checksum with crc_valid signal.
	crc_comparator	Module to compare the checksum of the receiving packet with the Ethernet CRC 32 residue value.
Avalon-St Multiplexer (not applicable for Arria 10 designs)	**st_mux_2_to_1_hw.tcl**	The Qsys custom component accepts data on its two Avalon-ST sink interfaces, and multiplexes the data for transmission on its Avalon-St source interface.
	st_mux_2_to_1.v	
	st_mux.v	LPM_MUX IP core file.
Avalon St-Splitter (not applicable for Arria 10 designs)	**aso_splitter_hw.tcl**	The Qsys custom component accepts data on its Avalon-ST sink interface and splits the data for transmission on its two Avalon-ST source interfaces.
	aso_splitter.v	
Error Adaptor (not applicable for Arria 10 designs)	**error_adapter2_hw.tcl**	The Qsys custom component connects mismatched Avalon-ST source and sink interfaces.
	error_adapter2.tcl	

Related Information

- **TSE Single Port SGMII Arria 10 GX**
  Download the design file from the design store.
- **TSE Single Port RGMII Arria V GX**
  Click to download the design file.
- **TSE Single Port RGMII Cyclone V GX**
  Click to download the design file.
- **TSE Single Port SGMII Stratix V GX**
  Click to download the design file.
- **TSE Single Port SGMII Stratix IV GX**
  Click to download the design file.

Hardware and Software Requirements

The reference designs require the following hardware and software:

- Arria 10 GX, Arria V GX, or Stratix V GX, or Stratix IV GX FPGA Development Kit
- USB-Blaster or ByteBlaster download cable
- External Ethernet packet generator (only for Avalon-ST reverse loopback test)
- Ethernet cable assembly (only for Avalon-ST reverse loopback test)
- Quartus Prime version 15.0 or later
  - USB-Blaster or ByteBlaster driver
  - Qsys system
  - System Console

Related Information

Analyzing and Debugging Designs with the System Console
Provides more information about analyzing and debugging designs.

Internal MAC Loopback Test

Complete the tasks to run the internal MAC loopback test.

Before you begin

Set up the development board.

1. Connect the programming cable to the JTAG connection port.
2. Connect the board to the power supply input.

To run the internal MAC loopback test:

1. Set up the System Console.
   a. Open Qsys.
   b. On the Tools menu, click **System Console**.

      The System Console is a debugging tool that provides you with Tcl scripts to perform low-level hardware debugging and run tests on your reference designs. The console communicates to the hardware components instantiated into your Qsys system reference designs through the JTAG to Avalon Master Bridge.
2. Open the **config.tcl** script and set `LOOP_ENA` to 1 to enable the MAC loopback mode.
3. Type the following command to start the MAC and PHY configurations in the System Console:

   ```
   source config.tcl
   ```

   For more information, refer to **Configuration Script** on page 19.

   The System Console displays the copper link connection status and the PHY's operating speed and mode. Verify that the console displays the correct configurations
4. Open and edit the **eth_gen_start.tcl** script.

   For more information, refer to **Ethernet Packet Generator Script** on page 19.
5. Type the following command to start generating Ethernet packets:

   ```
   source eth_gen_start.tcl
   ```

   The Ethernet Packet Monitor automatically starts when you start the Ethernet Packet Generator.

- When the monitor receives all the Ethernet packets, the System Console displays the loopback test result.
- If the monitor receives packets with error, the console displays the total number of packets received with error and the type of error for each packet.

6. Type the following command to view the MAC statistic counters:

```
source tse_stat_read.tcl
```

Avalon-ST Reverse Loopback Test

Complete the tasks to run the Avalon-ST reverse loopback test.

Before you begin

Set up the development board. The Avalon-ST reverse loopback test requires an external Ethernet packet generator.

1. Using the Ethernet cable assembly, connect the external generator to the RJ-45 port of the FPGA development board.
2. Connect the programming cable to the JTAG connection port.
3. Connect the board to the power supply input (J4).

To run the internal MAC loopback test:

1. Set up the System Console.
   a. Open Qsys.
   b. On the Tools menu, click **System Console**.

      The System Console is a debugging tool that provides you with Tcl scripts to perform low-level hardware debugging and run tests on your reference designs. The console communicates to the hardware components instantiated into your Qsys system reference designs through the JTAG to Avalon Master Bridge.
2. Open the **config.tcl** script and set `LOOP_ENA` to 0 to disable the MAC loopback mode.

   For more information, refer to **Configuration Script** on page 19.
3. Type the following command to start the MAC and PHY configurations in the System Console:

   ```
   source config.tcl
   ```

   The System Console displays the copper link connection status and the PHY's operating speed and mode. Verify that the console displays the correct configurations
4. Start sending Ethernet packets from the external packet generator to the FPGA development board and verify that the packets are correctly looped back to the external packet generator.
5. Type the following command to view the MAC statistic counters:

   ```
   source tse_stat_read.tcl
   ```

Tcl Script

You can use any text editor to edit the Tcl scripts inside the **sc_tcl** folder of the reference designs.

Note: Altera recommends that you do not modify the **tse_mac_config.tcl**, **tse_marvel_phy.tcl**, **eth_gen_mon.tcl**, and **tse_stat_read.tcl** scripts inside the **sc_tcl** folder.

Configuration Script

The **config.tcl** configuration script contains the parameters to configure the MAC, PCS, and Marvell PHY registers in the reference designs.

You can configure the following settings in the Tcl script:

- MAC configuration setting to configure the MAC registers.
- PCS configuration setting to configure the PCS registers.
- Marvell PHY configuration setting to configure the on-board PHY chip registers.

Parameter	Description
PHY_ENABLE	To enable or disable the on-board PHY chip.
PHY_ETH_SPEED	To select the PHY's operating speed.
PHY_ENABLE_AN	To enable or disable auto-negotiation on the PHY.
PHY_COPPER_DUPLEX	To select the PHY's operating mode.Statistic Counter Script
PHY_LOOPBACK	To enable or disable the on-board PHY serial loopback.

The **tse_stat_read.tcl** script reads the values of the MAC statistic counters after you execute the reference designs.

Related Information

Configuration Register Space
Provides more information about Triple-Speed Ethernet configuration registers and statistic counters.

Ethernet Packet Generator Script

The **eth_gen_start.tcl** configuration script contains the parameters to configure the Ethernet Packet Generator registers in the reference designs.

You can use any text editor to configure the following registers in **eth_gen_start.tcl** script.

Table 12: Ethernet Packet Generator Script Parameters

number_packet	To set the total number of packets to be generated by the packet generator.
eth_gen	To enable or disable the packet generator.
length_sel	To select fixed or random packet length.
pkt_length	To set the fixed packet length. The packet length can be a value between 24 to 9,600 bytes.
pattern_sel	To select the data pattern for the random packet length.
rand_seed	To set the initial random seed for the PRBS generator. This parameter is only valid when you select random packet length.
source_addr	To set the source MAC address.
destination_addr	To set the destination MAC address.

JTAG Connection Port for Altera Devices

The JTAG connection port varies for different Altera devices.

Table 13: JTAG Connection Port for Altera Devices

Device	JTAG Connection Port
Arria 10 GX	J3
Arria V GX	J14
Stratix V GX	J7
Stratix IV GX	J7

Document Revision History

Date	Version	Changes
December 2015	2015.12.14	• Added information about Arria 10 reference design. • Removed information about the obsoleted Arria II GX reference design.
December 2012	2012.12.03	Added device support and restructured reference design topic.
Sept 2011	2011.09.23	Updated the link to the design example download page.
June 2011	2011.06.17	Initial release.