XAPP955 (v1.3) September 19, 2008

10-Gigabit Ethernet Hardware Demonstration Platform

Summary

This 10-Gigabit Ethernet Hardware Demonstration Platform application note describes the functionality of the LogiCORE™ IP 10-Gigabit Ethernet and XAUI cores in Xilinx FPGA hardware. Development board requirements, setup instructions, MAC core-specific design components, and a description of the Graphical User Interface (GUI) used to control the demonstration platform are provided.

Introduction

The 10-Gigabit Ethernet Hardware Demonstration Platform shows the functionality of the following LogiCORE IP products:

- 10-Gigabit Ethernet MAC (10GEMAC) core
- XAUI core (as part of the XGMAC demonstration)

The demonstration platform uses the following:

- A development board with a Xilinx FPGA loaded with the hardware design
- A PC to control the hardware design using a RS-232 serial cable

The demonstration platform shows how to integrate these cores into a system, generate the required clock resources, handle Ethernet data flow using packet FIFOs and flow control, and connect to a physical interface.

A GUI that uses a standard RS-232 serial cable is provided, allowing the user to experiment with different MAC configurations, monitor frame information, and observe the MAC statistics, all of which are defined in this document.

Requirements

Development Boards

- The demonstration platform supports the ML421 Rev C, ML523 Rev. D LX110T and FX100T development boards.
- XAUI connections use SMA cables.
  - 2 cables are needed if running in internal loopback mode
  - 10 cables are needed if running in external cable loopback
  - 18 cables are needed if connecting to another XAUI device

PC

The demonstration platform requires a PC running Windows® XP for the GUI, and a spare RS232 serial port is required to connect the GUI to the development board.

Demonstration Platform System Design

10-Gigabit Ethernet MAC Hardware Design

Figure 1 illustrates the design for the 10-Gigabit Ethernet MAC hardware. The design, including the microprocessor system, uses approximately 9500 slices for the Virtex™-4 FPGA design and 4600 slices for the Virtex-5 FPGA design. A XAUI core provides a physical interface to the MAC, and a FIFO is used on the client side of the MAC to provide frame buffering. The microprocessor system connects to the MAC host interface to control the MAC configuration

and connects to the FIFO to read/ write frame data. The microprocessor is a MicroBlaze™ processor.

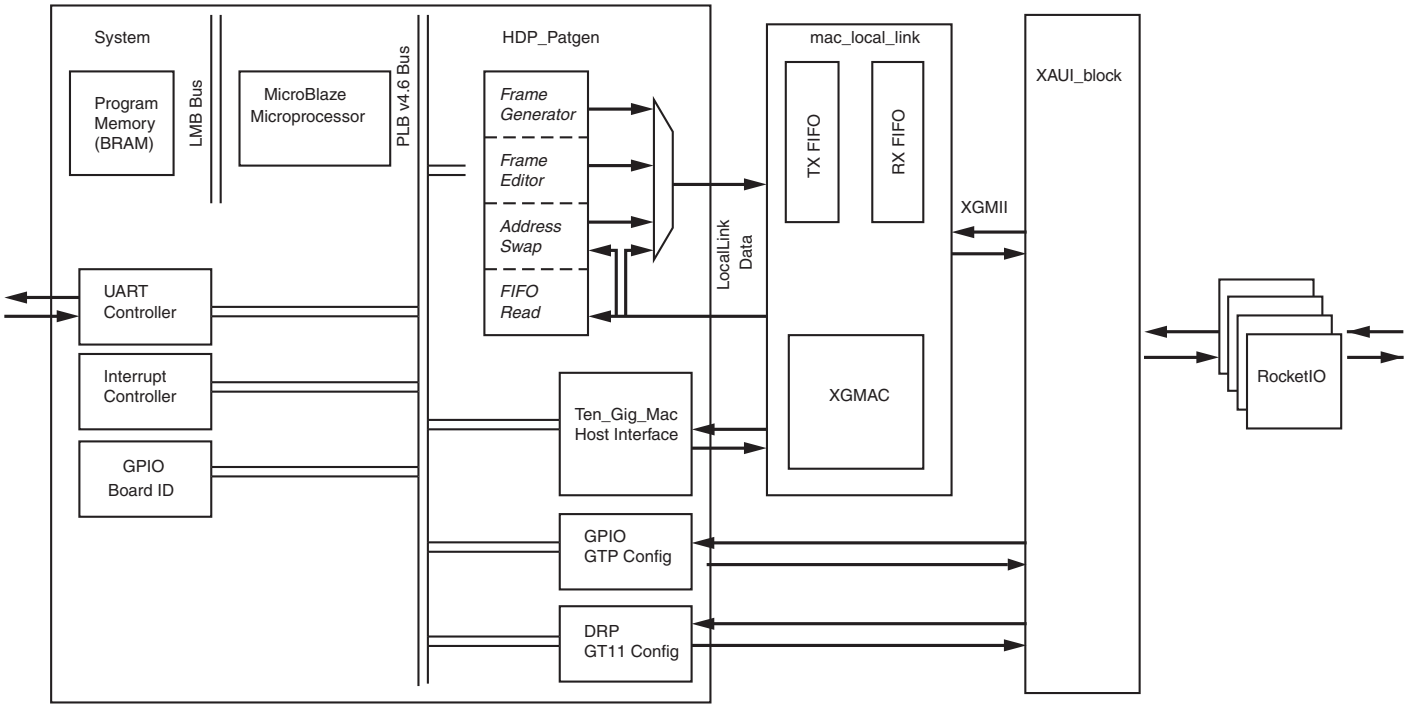

Figure 1: **10-Gigabit Ethernet Hardware Demonstration Platform**

10-Gigabit Ethernet MAC Core

The 10GEMAC core v8.6 is a CORE Generator™ system instance generated with the following XCO parameters:

```
component_name = ten_gig_eth_mac_v8_6
physical_interface = internal
management_interface = true
statistics_gathering = true
simplex_split = none
```

The core is generated with a full-system hardware evaluation license, which operates for approximately eight hours, after which the core ceases to function and the FPGA must be reconfigured to resume operation.

XAUI Core

The XAUI core v7.4 is a CORE Generator system instance used to connect to a PHY or an external device with a XAUI interface, and is generated with the following XCO parameters:

```
component_name = xaui_v7_4_v4 (Virtex-4)
component_name = xaui_v7_4_v5 (Virtex-5 LXT)
component_name = xaui_v7_4_v5_gtx (Virtex-5 FXT)
802_3ae_state_machines=true
mdio_management=true
simplex_split=None
use_tx_elastic_buffer=false
xgmii_interface=Internal
```

There are three instantiations of the XAUI block level in the HDL, as the pinout differs between architectures.

Ethernet FIFO

The FIFO used is taken from the 10-Gigabit Ethernet MAC example design and is provided with the core.

Physical Interface

The XAUI signals all connect to SMAs on the board.

Clock Management

The XAUI core requires a 312.5 MHz reference clock for Virtex-4 FPGAs, and a 156.25 MHz reference clock for Virtex-5 FPGAs. A lower-speed clock (50 MHz) should be present as an auxiliary clock to control the transceivers.

ChipScope

ChipScope™ Integrated Logic Analyzers (ILAs) in the hardware design allow for viewing the data as it passes through the demonstration design. The ChipScope probes are placed on the transmitter XGMII and receiver XGMII interfaces, the transmitter client and receiver client interfaces, and the MAC statistics vector. A ChipScope project file is provided for the 10-Gigabit Ethernet MAC demonstration.

Microprocessor System

The microprocessor system uses a MicroBlaze processor. This was created using the Xilinx Platform Studio as a submodule of the top-level design. This is a MicroBlaze v7.1 CPU subsystem containing RAM, UART (xps_uartlite), interrupt controller, LMB, and PLBv46 buses.

Pattern Generator

The Pattern Generator (HDP_Patgen) peripheral holds all the logic required to access the client-side data stream of the MAC core, allowing data to be written to the transmitter side FIFO or to be read from the receiver side FIFO. This peripheral is comprised of the following components:

- **Frame Editor**. Uses block RAM to store user-defined frames for the transmitter. 16 kBytes of RAM are available for frame storage.
- **Frame Generator**. Generates fixed pattern test frames and transmits them individually or continuously.
- **Loopback Multiplexer**. Included in the pattern generator. The loopback multiplexer connects the transmitter (Tx) and receiver (Rx) FIFOs, allowing external test equipment or Ethernet devices to stimulate the MAC core and receive frames. There is an option to switch the source address and destination address fields of each frame passing through the multiplexer, allowing external equipment to stimulate the design and receive valid addressed frames from the demonstration platform.
- **Receive FIFO Reader**. Reads the contents of the receiver-side FIFO. The captured frame data can be read from the peripheral by the microprocessor, which formats the frames and prepares them for display on GUI by transmitting them over the serial link.

Host

The Host peripheral (ten_gig_mac) allows the Management Interface of the MAC core to be accessed. Through this interface, the MAC configuration can be read and modified and the statistics counters can be read.

DRP Controller (Virtex-4 Only)

The DRP controller (drp_config) allows full control over the Dynamic Reconfiguration Port on the RocketIO™ transceiver. The pre-emphasis and receive equalization on the RocketIO transceiver for Virtex-4 are controlled through this module.

Microprocessor Firmware

The microprocessor runs firmware to monitor the serial interface and respond to commands issued by the GUI. On startup, the microprocessor detects which design is operating on the FPGA, configures the logic on the FPGA appropriately, and then processes any commands sent from the PC.

Setting up the Demonstration Platform

ML421 Configuration for 10-Gigabit Ethernet MAC Demo

Power Supply

The ML421 should be powered from the 5V jack input. See the ML421 documentation for details about connecting the supply.

Clock Selection

The demonstration platform uses the Xilinx Superclock Module to generate the 312.5 MHz differential oscillator as a master reference to generate all required clocks required. This differential clock signal from the superclock module should be connected to MGTCLK 102 on the ML421 using two SMA cables. A 25 MHz crystal should be present in the XTAL0 socket on the Superclock Module.

The DIP switches should be set as follows:

- N0: 1
- N1: 0
- N2: 0
- M0: 1
- M1: 1
- M2: 0
- SEL0: 0
- SEL1: 0

An oscillator (X2) with a frequency of 50 MHz should be plugged into socket X2.

Jumper Settings

The setting on the DIP switches (other than the System ACE™ address) does not affect the operation of the designs.

System Reset

The system reset is connected to push-button SW1.

RS-232

Connect a null modem cable between the RS-232 port on the board and a RS-232 port on the PC. A null modem cable is an RS-232 serial cable where the transmit and receive wires are cross-connected.

Table 1: **Null-Modem Wiring**

Signal Name	DB-9 Pin	DB-9 Pin
TD (Transmit Data)	3	2
RD (Receive Data)	2	3
RTS (Request To Send)	7	8
CTS (Clear To Send)	8	7

Table 1: **Null-Modem Wiring**

Signal Name	DB-9 Pin	DB-9 Pin
SG (Signal Ground)	5	5
DSR (Data Set Ready)	6	4
CD (Carrier Detect)	1	4
DTR (Data Terminal Ready)	4	1
DTR (Data Terminal Ready)	4	6

LEDs

The LEDs on the boards indicate the basic status of the platform. More detailed status information is available through the GUI. Table 2 defines the LED functions.

Table 2: **LED Indications for ML421 10-Gigabit MAC Demo**

LED	10-Gigabit Function
DS11	Lane0 Sync OK
DS12	Lane1 Sync OK
DS13	Lane2 Sync OK
DS14	Lane3 Sync OK
DS15	Lanes Aligned
DS16	Fifo Loopback ON
DS17	Address Swap ON
DS18	DCM Locked

Physical Interface

SMA cables must be used to connect the MGT inputs and outputs to a suitable XAUI-equipped PHY. Sixteen SMA cables are required for the 4-lane differential XAUI interface. The board connections are defined in Table 3.

Table 3: **XAUI SMA Connections**

XAUI Lane	SMA Connector
0	103B
1	103A
2	102B
3	102A

Cabled Loopback

Eight SMA cables are required for the external cabled loopback. The cables should be connected from the transmit side to the receive side of the same lane, as shown in the following example: TXP -> RXP, TXN-> RXN.

FPGA Configuration

The FPGA must be configured with the bit file included with the Application Note source files. The bit file can be found in the Implement folder.

xgm_hdp_421.bit	:	10-Gigabit design for V4 ML421 board

The FPGA can be configured with the System ACE or a programming cable.

FPGA Configuration Using System ACE

System ACE files for the bit file listed above are included in the implement folder. Select the correct folder for the design using the same naming convention as above. For example, implement/systemace_xgm_ml421 contains the design for the ML421. Copy the contents of this folder onto a compact flash card and then insert the card into the CF card slot. The System ACE address DIP switches should be set to "000." The device is then configured from the System ACE file upon power-up.

FPGA Configuration Using a Programming Cable

After connecting the programming cable to the JTAG header (see development board documentation), the appropriate bit file should be loaded onto the FPGA using iMPACT or ChipScope.

ML523 Configuration for 10-Gigabit Ethernet MAC Demo

Power Supply

The ML523 should be powered from the 5V jack input. See the ML523 documentation for instructions for connecting the power supply.

Clock Selection

The demonstration platform uses the Xilinx Superclock Module to generate the 156.25 MHz differential oscillator as a master reference to generate all required clocks. The differential clock should be connected to REFCLK_118. A 25 MHz crystal should be present in the XTAL0 socket.

Set the DIP switches as follows:

- N0: 1
- N1: 1
- N2: 0
- M0: 1
- M1: 1
- M2: 0
- SEL0: 0
- SEL1: 0

An oscillator with a frequency of 50 MHz should be plugged into socket X2.

The System ACE oscillator (X1) should be enabled (J4 set to **ON**) if the FPGA is configured using System ACE.

Jumper Settings

The setting on the DIP switches (other than the System ACE address) does not affect the operation of the designs.

System Reset

The system reset is connected to push-button SW8.

RS-232

Connect a null modem cable between the RS-232 port on the board and a RS-232 port on the PC. A null modem cable is an RS-232 serial cable where the transmit and receive wires are cross-connected. Table 4 lists the null modem wiring attributes for RS-232.

Table 4: **Null-Modem Wiring**

Signal Name	DB-9 Pin	DB-9 Pin
TD (Transmit Data)	3	2
RD (Receive Data)	2	3
RTS (Request To Send)	7	8
CTS (Clear To Send)	8	7
SG (Signal Ground)	5	5
DSR (Data Set Ready)	6	4
CD (Carrier Detect)	1	4
DTR (Data Terminal Ready)	4	1
DTR (Data Terminal Ready)	4	6

LEDs

The board LEDs indicate the basic status of the platform at a glance. Detailed status information is available from the GUI. Table 5 defines the LED functions.

Table 5: **LED Indications for ML523 10-Gigabit MAC Demo**

LED	10-Gigabit Function
DS23	Lane0 Sync OK
DS22	Lane1 Sync OK
DS21	Lane2 Sync OK
DS20	Lane3 Sync OK
DS19	Lanes Aligned
DS18	Fifo Loopback ON
DS17	Address Swap ON
DS16	DCM Locked

Physical Interface

SMA cables must be used to connect the MGT inputs and outputs to a suitable XAUI-equipped PHY. Sixteen SMA cables are required for the 4-lane differential XAUI interface. The board connections are described in Table 6.

Table 6: **XAUI SMA connections**

XAUI Lane	SMA Connector
0	122_0
1	122_1
2	118_0
3	118_1

Cabled Loopback

Eight SMA cables are required for external cabled loopback. The cables should be connected from the transmit side to the receive side of the same lane; for example:

TXP -> RXP, TXN-> RXN

FPGA Configuration

The FPGA must be configured with one of the bit files included with the Application Note source files.The following bit file is located in the implement folder.

xgm_hdp_lx110t.bit	:	10-Gigabit design for ML523 LX110T board
xgm_hdp_fx100t.bit		10-Gigabit design for ML523 FX100T board

The FPGA can be configured using the System ACE or programming cable.

FPGA Configuration Using System ACE

System ACE files for the bit file listed above are also located in the implement folder. Select the correct folder for the design using the same naming convention as above. For example, implement/systemace_xgm_ML523 contains the design for the for the LX110T ML523. Copy the contents of this folder to a compact flash card and insert the card into the CF card slot. The System ACE address DIP switches should be set to **000**. The device is then configured from the System ACE file upon power-up.

FPGA Configuration Using a Programming Cable

After connecting the programming cable to the JTAG header (see board documentation), the xgm_hdp_523.bit file should be loaded onto the FPGA using iMPACT or ChipScope.

Graphical User Interface

The GUI application allows for control of the Ethernet MAC core for demonstration and other components within the design. When the GUI is executed, the Serial Port Selection dialog appears. When a selection is made, the Main screen appears, and the MAC Statistics screen can be accessed from the Main screen.

Installation

Extract the project files from the Application Note ZIP file into a new directory (for example, c:\xapp955).

Note: Do not download the demonstration platform to a directory path containing spaces; doing so causes errors when running the GUI.

After loading the hardware design onto the FPGA, launch the GUI by navigating to the directory c:\xapp955\gui\ (assuming you saved the zip file contents to c:\xapp955) and double-click on the xapp955.exe file.

Serial (COM) Port Selection

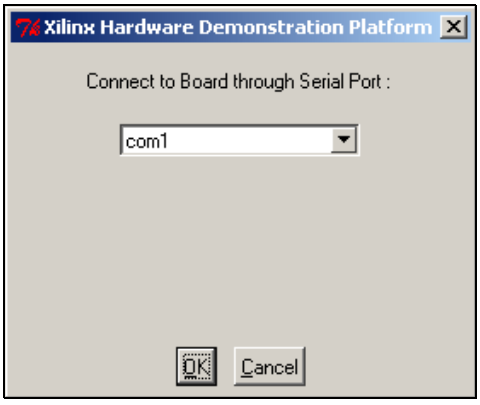

Figure 2: **Serial (COM) Port Selection Screen**

The Serial Port selection dialog box (Figure 2) appears when the GUI is launched. Use this dialog to select the COM port to be used to communicate with the board. Click OK to establish the link with the board. When the link is established, the main GUI screen appears (Figure 3).

When a link is established with the board, the board type and MAC type are displayed at the bottom of the screen. If a connection to the board cannot be established, an error message appears at the bottom of the main screen.

To establish a connection, do the following:

1. Ensure that the board is switched on.
2. Ensure that the FPGA is configured with the correct bit file.
3. Ensure that the serial cable is connected and is the correct type.
4. Restart the GUI.

Main Screen

Figure 3 illustrates the main screen of the demonstration platform, consisting of four sections:

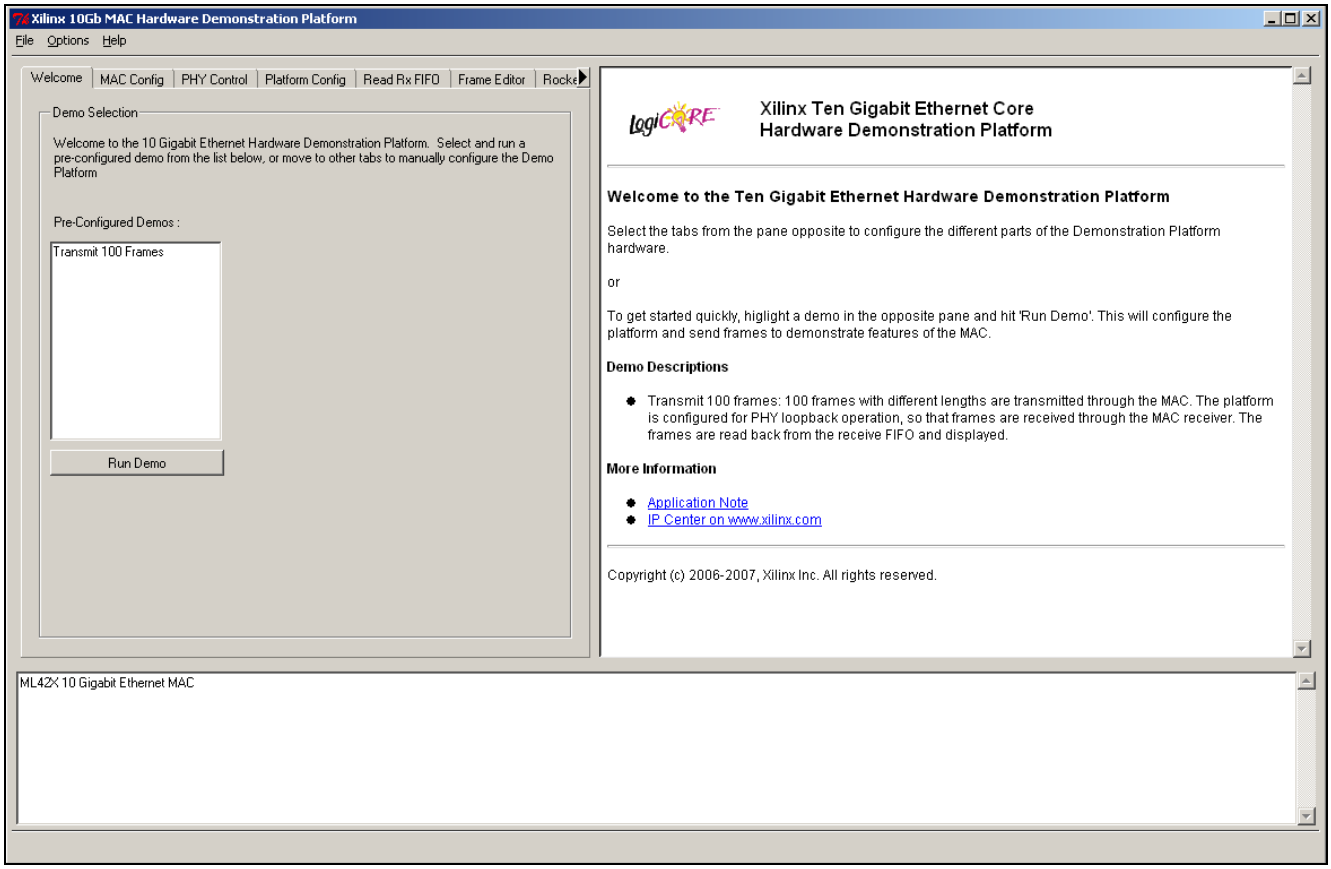

Figure 3: **Demonstration Platform Main Screen**

Main Menu Bar

A standard Windows menu that allows you to exit the client and open the statistics menu.

Control Pane

The Control Pane provides several tabs for controlling the different configuration settings of the design.

Help Pane (Rightmost Pane)

Displays context-sensitive help.

Message Pane (Bottom Pane)

Displays debug and status messages.

Welcome

The Welcome tab (Figure 4) is displayed when the application starts (after the COM port is selected). Use this screen to run the pre-configured demo of Transmitting 100 Frames by selecting it from the list of Pre-Configured Demos, and click **Run Demo**. When the demo is complete, an informational message appears in the message pane (bottom of screen).

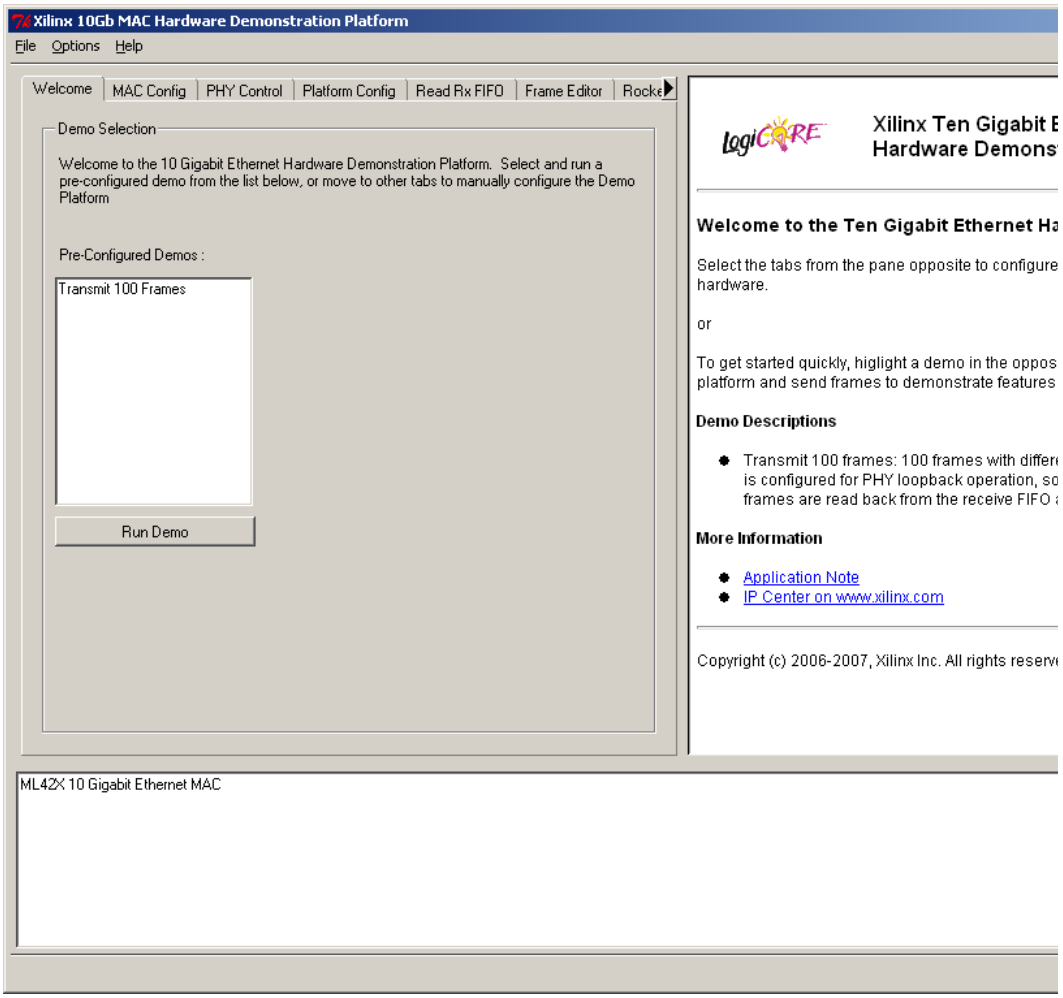

Figure 4: **Welcome Tab**

MAC Configuration

The MAC Configuration tab (Figure 5) provides access to the dynamically configureable features of the MAC cores. When the FPGA is initialized, this screen displays the default configuration for the MAC core.

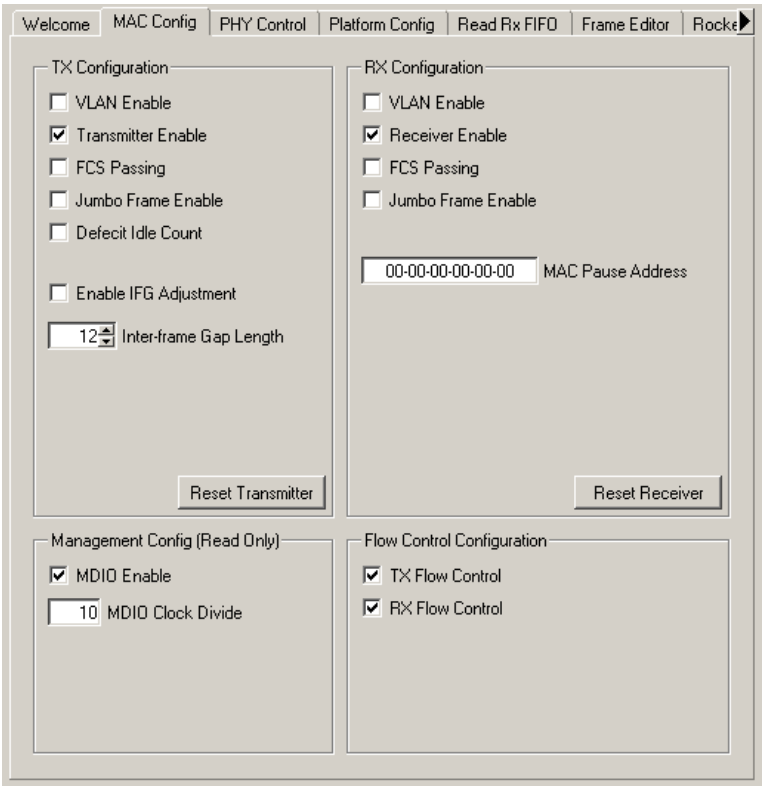

Figure 5: **MAC Configuration Tab**

The MAC configuration for transmitter and receiver can be reset to the default values by using the **Reset Transmitter** and **Reset Receiver** buttons.

The management configuration for the MDIO interface cannot be modified from the GUI, as these settings are configured correctly to allow the hardware to control the PHY or MGT. The management configuration is set by the embedded processor software, which writes to the management configuration of the MAC upon start-up.

PHY Control

The PHY control tab (Figure 6) displays the information gathered by the MAC core about the PHY (XAUI) using MDIO. The only items in this list that can be user-modified are loopback and powerdown. Loopback sets the XAUI loopback register and disables the TXPOST_TAP_PD bit

via the DRP in Virtex-4. Powerdown sets the XAUI powerdown bit, which places the transceivers into low power mode. This allows the demo to be used without a PHY card.

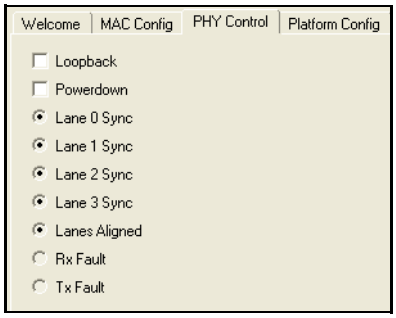

Figure 6: **PHY Control Tab**

The other items provide status information about the four XAUI lanes. All lanes should be synchronized and aligned. When no SMA XAUI device is connected, the lanes will not synchronize and align until loopback is enabled. (The bit will remain high until it is cleared by software.) An Rx Fault condition is indicated if the lanes do not align, and is only cleared when the tab is brought up. You may see an indication that there has been a Rx Fault between the time the tab was viewed and the present time, and therefore the current status of the lanes is not always immediately reflected.

Platform Configuration

The Platform Configuration tab (Figure 7) controls the data sources in the demonstration platform.

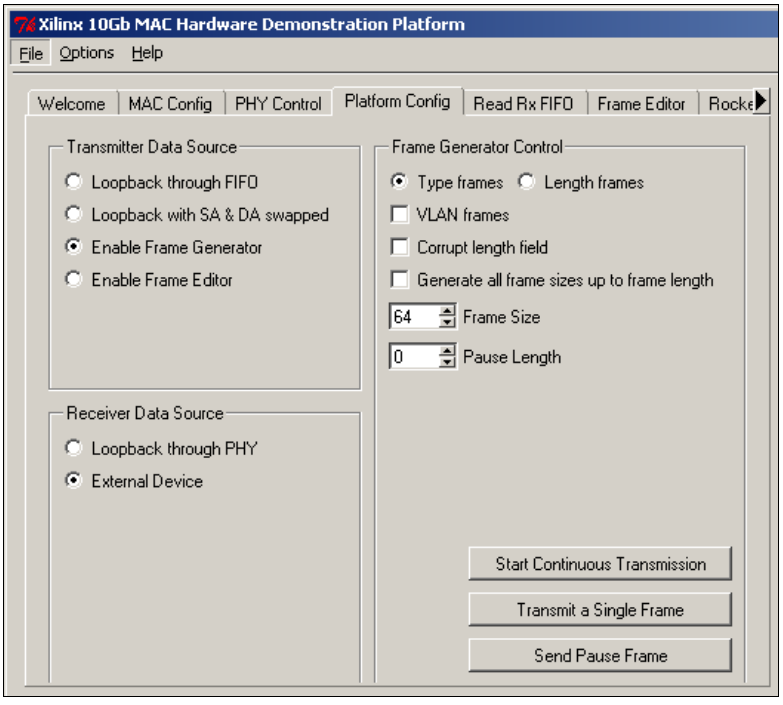

Figure 7: **Platform Configuration Tab**

Transmitter Data Source

The client side transmitter interface of the MAC can be configured in one of the following ways:

- **Loopback Through Fifo**. Connects the data path from the MAC client side receiver to the transmit data path. Data passes from the MAC receiver to the receive FIFO, then directly to the transmit FIFO and finally to the MAC transmitter.
- **Loopback with Address Swap**. Similar to the above mode, but the first six bytes of each frame are switched with the following six bytes; switching the source and destination address. This allows an external device to send frames to the demonstration platform and receive correctly addressed frames.
- **Frame Generator**. Connects the transmit FIFO to the hardware pattern generator. Options for controlling the pattern generator are also set here and are described in "Frame Generator Control."
- **Frame Editor**. Connects the transmit FIFO to the RAM based frame editor, which allows entering and transmitting any frame using the GUI. The Frame Editor is controlled in the Frame Editor tab. See "Frame Editor," page 16.

Receiver Data Source

The receiver can be configured in either of the following two ways:

- **Loopback through PHY**. The MDIO is used to request loopback at the MGTs used by the XAUI.
- **External Device**. Loopback is disabled and all data is driven onto the SMA cables.

Frame Generator Control

The Frame Generator can generate various frame types and lengths, as described below.

- **Type Frames**. When checked, the pattern generator inserts hexadecimal 0x0800 in the length/type field of each frame.
- **Length Frames**. When checked, the pattern generator inserts the length of the data contained in the frame into the length/type field of each frame.
- **VLAN Frames**. When enabled, the pattern generator inserts a VLAN tag into each frame. The total length of each frame is unchanged but the length field is adjusted to indicate less data in each frame.
- **Corrupt Length Field**. When selected, the length field is set incorrectly, giving an errored frame.
- **Generate All Frame Sizes Up To Frame Length**. When checked, the pattern generator will start transmitting 19-byte frames (padded by the MAC core) and increases the length of the frame by 1-byte. When the frame length equals the length selected, the length is set back to 19 and the process repeats. Use this option for continuous frame transmission.
- **Frame Size**. Set the total size for frames sent to the transmitter. If less than 60, the MAC core pads the frame, unless configured for FCS passing.
- **Pause Length**. Set the 16-bit pause length input to the MAC core, which is used if **Send Pause Frame** is selected. See the specific MAC user guide for a description of the pause transmission mechanism.

The buttons are used to send frames to the MAC transmitter and trigger transmission by the pattern generator. The Transmitter Data Source must be set to **Enable Frame Generator** to enable these buttons.

- **Transmit a Single Frame.** Enables one-shot mode in which a single frame is sent.
- **Start Continuous Transmission.** Sends frames to the MAC until the same button, now displaying **Stop Transmission** is clicked.
- **Send Pause Frame**. Sends a single pause frame from the MAC which asserts the pause request input to the MAC.

Read Rx FIFO

The Read Rx FIFO tab (Figure 8) is used to read frames from the receive side FIFO that have been received by the MAC core. The FIFO cannot be read if the platform is configured for loopback through the FIFO.

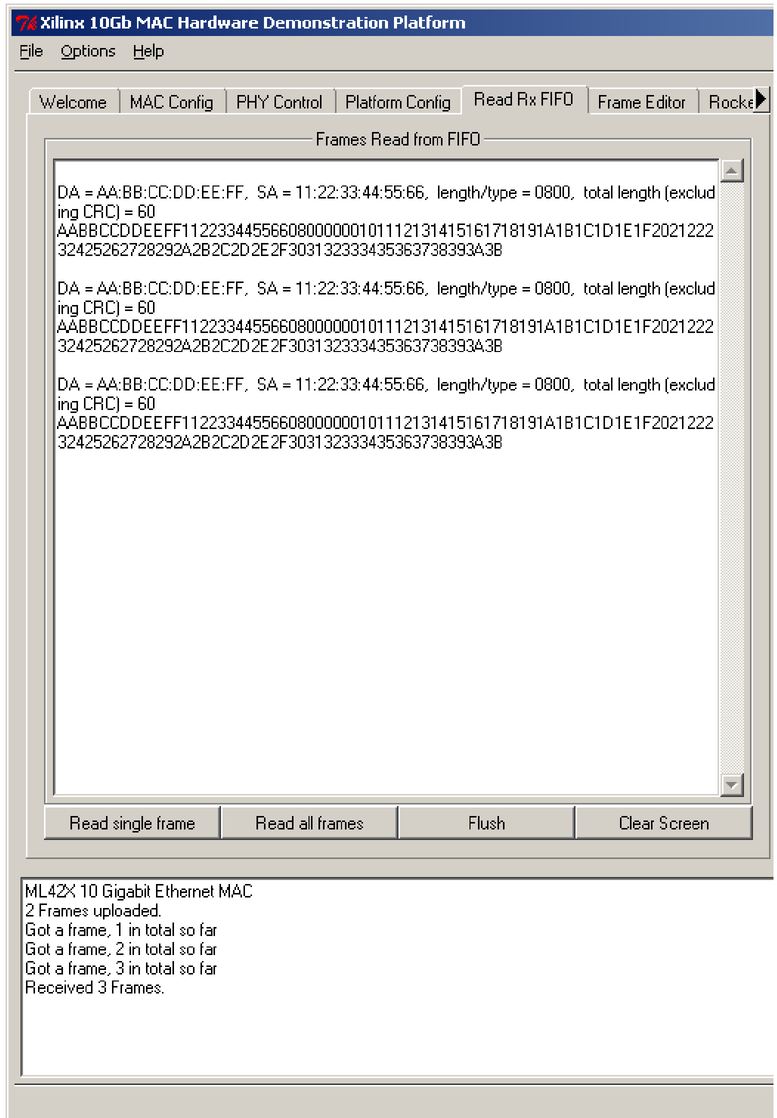

Figure 8: **Read Receive FIFO Tab**

The FIFO stores all frames received from the MAC core until it is full. Once full, the FIFO discards further incoming frames. The FIFO is 16 Kbytes deep, and can store approximately 10 frames of 1518 bytes.

Care should be taken when mixing frame sizes. If the FIFO is nearing full and a frame larger than the remaining space is received, it will be discarded. If the next frame is smaller than the remaining space, it will be stored in the FIFO. This should not be confused with frame loss.

To read frames from the receive FIFO, it is advisable to flush the FIFO before initiating transmission. This ensures that the FIFO has space to store the new frames. After transmission is started, the FIFO can be read. During continuous reception of frames, a snapshot of the data can be captured by flushing the FIFO, causing it to discard any stored frames, and to store the next 16 kBytes of frame data received. You can read a single frame or all available frames.

During continuous reception, the FIFO is unlikely to empty because data can enter the FIFO at a much faster rate than can be transmitted across the serial link to the GUI. To prevent an endless upload of frames to the GUI, a maximum of 64 frames can be read at once. To read more frames, continue to use the **Read all frames** button.

Frame Editor

The Frame Editor tab (Figure 9) is used to load frames into the RAM in the Pattern Generator, which can then be sent to the MAC core for transmission. To enable this functionality, set the Transmitter Data Source (located in the Platform Config tab) to Enable Frame Editor.

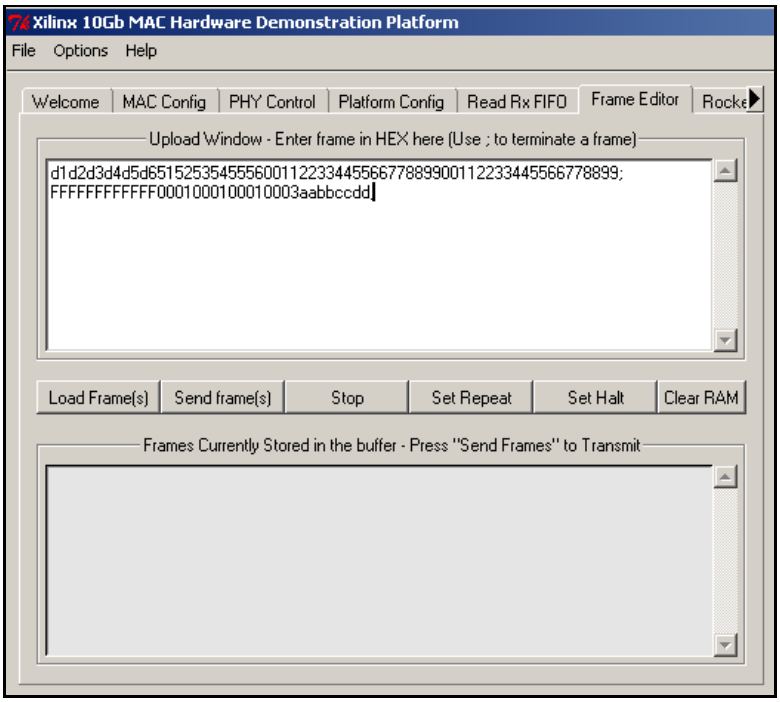

Figure 9: **Frame Edit Tab: Data Entry**

Frame data should be entered as hexadecimal in the top window. Multiple frames can be entered and uploaded at the same time.

The syntax for entering frames is as follows:

1. White space and CR/LF characters are ignored.
2. Data must only contain the characters 01234567890ABCDEFabcdef.
3. Each frame must be terminated with a semicolon.
4. Each frame must have an even number of characters (nibbles).

   Example valid frames:

   ```
   1122334455667788990OAAbbCCddEE;
   1 2 3 4 5 6 7 8;11223344; 2 35556;
   ```

   Example invalid frames:

   ```
   123456789; - uneven number of nibbles will give a warning
   11223344rt56; - invalid hex characters
   ```

   Example mistakes:

   ```
   11223344556677889900
   12345678123456789012;
   ```

The input will be treated as a single frame. The carriage return character is not treated as a frame terminate character.

Click Load Frames(s) to load frames into the RAM (Figure 10). If an error is detected on an entered frame, it is highlighted and the frame remains in the upload window. If three frames are entered and the first two are valid but the third is invalid, the first two are uploaded and the third remains in the edit window. If an error occurs in the first frame, all three frames remain in the upload window.

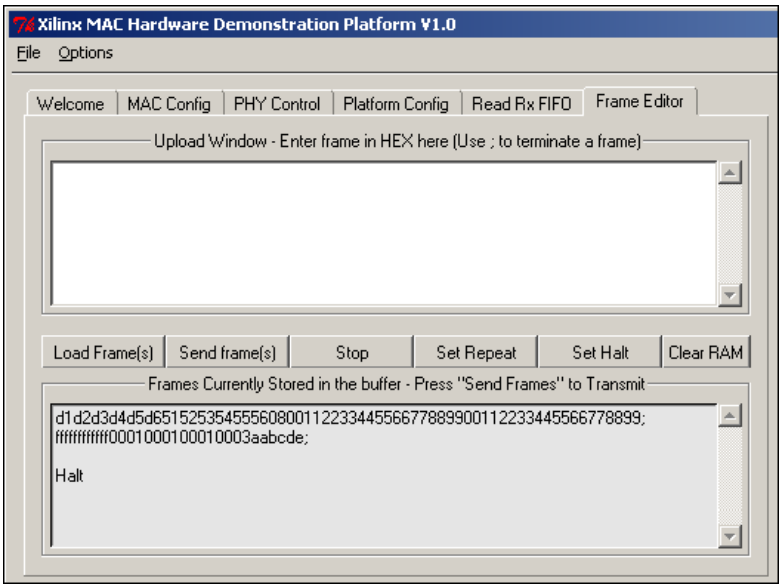

Figure 10: **Frame Edit Tab: Buffer Display**

- **Set Repeat** can be used to request continuous transmission of frames by continuously looping through all the frames in the RAM.
- **Set Halt** can be used to request that frames in the RAM are only sent once.
- **Send frames** is used to start sending frames to the MAC core.
- **Stop** is used to stop transmission of frames to the MAC core.
- **Load CJPAT** is used to load the IEEE 802.3-2005 Continuous Jitter Test Pattern.

RocketIO Transceiver Configuration

The RocketIO transceiver configuration tab (Figure 11) allows precise control over the Pre-emphasis, Tx Driver, and Receive Equalization settings. For more information see the appropriate RocketIO Transceiver User Guide.

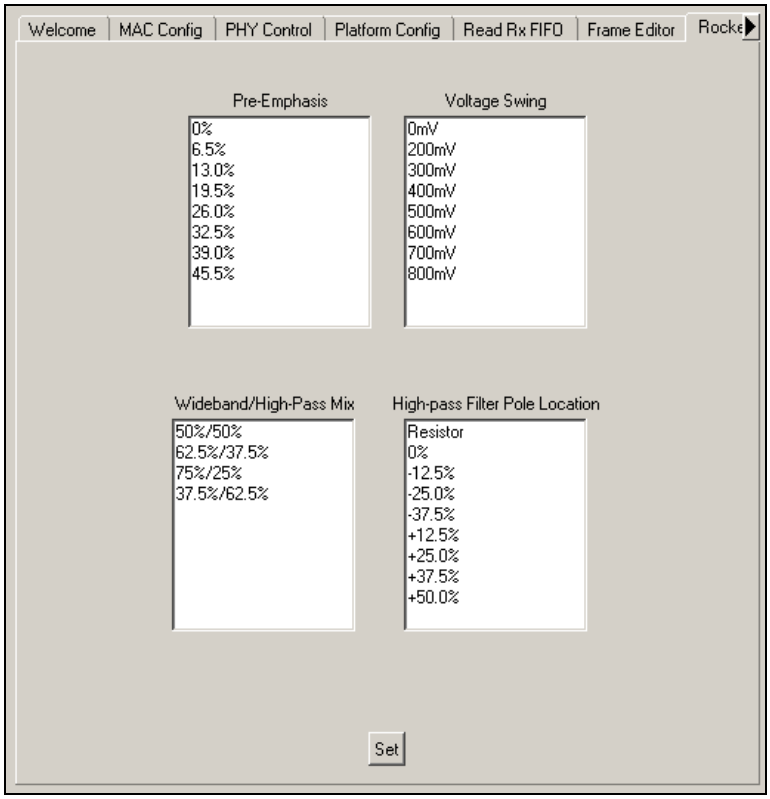

Figure 11: **RocketIO Transceiver Tab**

MAC Statistics

Choose Options > Show Stats from the top menu to view the statistics collected by the demonstration platform. This opens a new window (Figure 12) containing statistic counter values, collected either by the 10-Gigabit Ethernet MAC statistic counters.

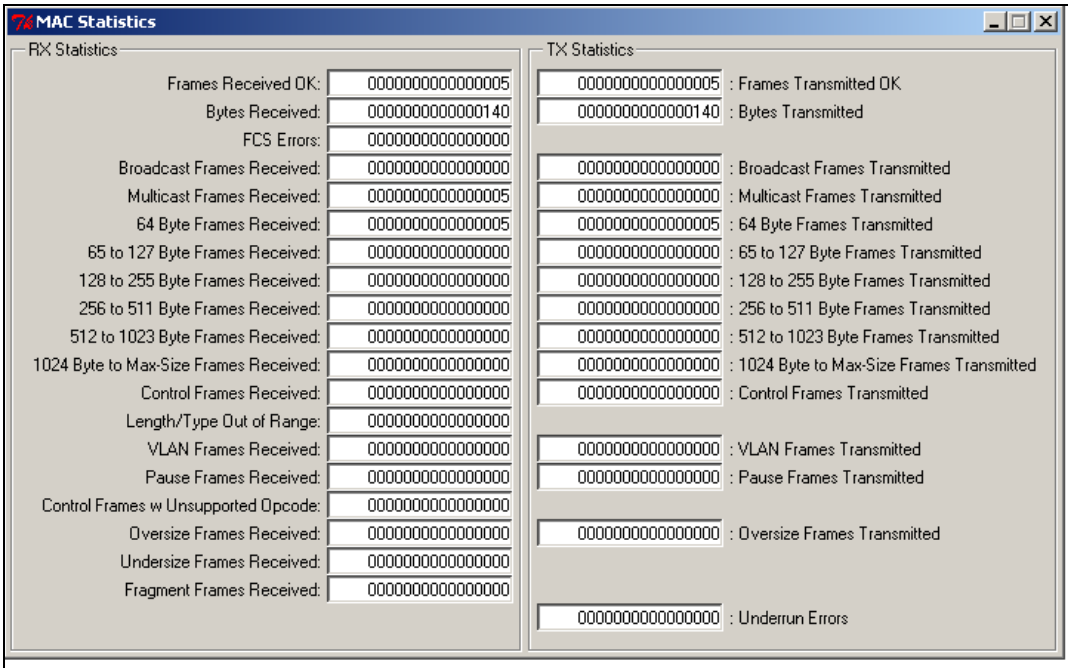

Figure 12: **MAC Statistics**

Building the Hardware

Ready-to-use bit files and System ACE files are provided. HDL source code and support files are also provided for the project. This permits regeneration of programming files if changes are made to the hardware design. This section describes how to recreate the programming files.

Source Code Directory Structure

The two main components of the hardware design are:

- An EDK subsystem containing the MicroBlaze peripherals and software.
- An RTL-based project containing the MAC and XAUI cores and board specific logic.

Figure 13 shows the organization of the source code directories. The EDK system is built using RTL from the edk_microblaze directory. The libraries which are provided by the EDK installation are not included in the ZIP file. The top-level RTL is contained in the hdl directory and the MAC, XAUI and ChipScope netlists are in the netlist directory.

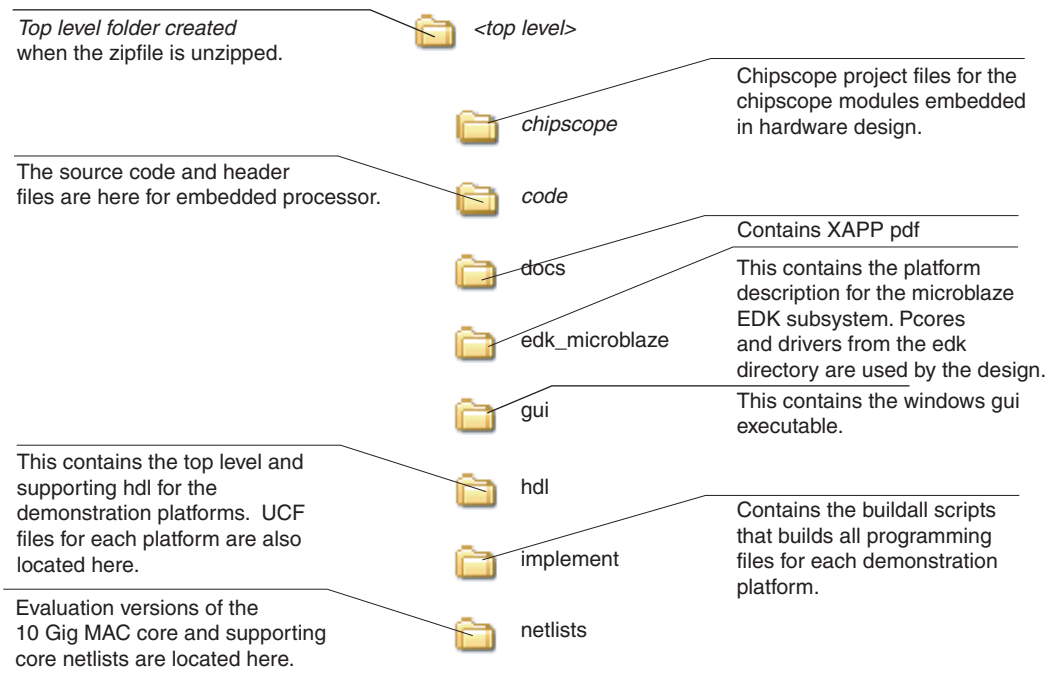

Figure 13: **Source Code Directories**

Building Demonstration Platforms

Two scripts are provided to build all the demonstration bit files and System ACE™ files automatically. First, the EDK system is built and then the Xilinx implementation tools are run. The output is created in the implement directory along with a log file for each project. The System ACE folders for each MAC/board combination are also created in the implement directory.

The script must be run with Xilinx ISE/EDK v10.1 SP3 or later installed. Hardware Evaluation or purchased licenses must be installed on the machine for each of the cores in the designs.

To run the script on Windows:

1. Open a Command Prompt and navigate to the folder created when the ZIP file was unzipped.
2. Change to the implement folder (c:\xapp955> cd implement).
3. Run `buildallv5.bat mb lx110t / buildallv5.bat mb fx100t` (If you have already built the EDK sub-system, you can substitute the 'mb' option with 'noedk'.)

To run the script on Solaris / Linux:

1. Open a shell and navigate to the folder created when the ZIP file was unzipped.
2. Change to the implement folder ([xapp955] > cd implement).
3. Run `uildallv5.bat mb lx110t / buildallv5.bat mb fx100t` (If you have already built the EDK sub-system, you can substitute the 'mb' option with 'noedk.')

References

[1] *LogiCORE IP 10-Gigabit Ethernet MAC User Guide*, UG148, September 19, 2008

[2] *LogiCORE IP XAUI User Guide*, UG150, September 19, 2008

[4] *ML42X User Guide*, UG087 (v1.1), November 6, 2005

[5] *ML52x User Guide*, UG225 (v1.1), August 6, 2007

[6] *Xilinx Generic Interface (XGI) SuperClock Module User Guide,* UG091 (v1.0.1), July 28, 2005

[7] *Virtex-4 RocketIO Multi-Gigabit Transceiver User Guide,* UG076 (v4.0), August 17, 2007

[8] *Virtex-5 RocketIO GTP Transceiver User Guide,* UG196 (v1.6), February 11, 2008

[9] *Virtex-5 RocketIO GTX Transceiver User Guide,* UG198 (v1.1), May 8, 2008

Revision History

Date	Version	Revision
3/01/07	1.1	Initial Xilinx release.
11/27/07	1.2	Update to LX110T.
9/18/08	1.3	Update subsystem to PLBv46. Add support for FX100T.

Notice of Disclaimer