CΩreEl

MicroSystems

CoreEl MicroSystems

4046 Clipper Court
Fremont, CA 94538 USA
Phone: +1 510-770-2277
Fax: +1 510-770-2288
E-mail: sales@coreel.com
URL: www.coreel.com

Features

- Individual transmitter and receiver cores, available separate or together
- Fully synchronous logic design
- Fully meets IEEE 802.3 specification
- Supports half and full duplex operations
- Supports full duplex flow control feature (802.3x)
- Flexible frame retransmission or abort feature
- Short frame transmission by padding
- Programmable inter-packet gap
- Optional long frame transmission and reception feature
- MAC address match feature
- Pause control frame detection
- Multicast and broadcast frame detection
- Extensive statistics information on transmit frames for RMON and MIBs
- Media Independent Interface (MII)
- Simple host data transfer interface
- Two host interface data path width options: 8- or 16-bits
- Meets Virtual Socket Interface (VSI) specification for a Soft Virtual Component

AllianceCORE™ Facts

Core Specifics		
Supported Family	4000EX	Virtex
Device Tested	4028EX-2	V150-4, V200-4, V300-4
CLBs - Transmitter CLBs - Receiver:	649 283	440[1] 332[1]
Clk IOB - Transmitter: Clk IOB - Receiver:	1 1	1 1
IOBs[2] - Transmitter: IOBs[2] - Receiver:	99 261	99 261
Performance	25 MHz	50 Mhz[4]
Xilinx Tools	M1.4	M1.5i
Special Features	SelectRAM	None
Provided with Core		
Documentation		Product Brief Datasheet
Design File Formats		VHDL compiled XNF netlist, VHDL compiled EDIF netlist[3]
Constraints File		Transmitter-fmac_tx.ucf Receiver-fmac_rx.ucf[3]
Verification		Test vectors in TSSI format (.do,.def) Script based behavioral VHDL testbench available
Instantiation Templates		None
Reference Designs & Application Notes		Hardware reference board for 4000EX series
Additional Items		None
Simulation Tool Used		
Model Tech V-System, version 4.6f		
Support		
Support provided by CoreEl MicroSystems		

Notes:
1. Utilization numbers for Virtex are in CLB slices. Implementation statistics provided are for Transmitter and Receiver cores configured with 16-bit host data busses.
2. I/O count assumes all core signals routed off-chip. Typical applications will incorporate many of these signals internal to the FPGA.
3. Virtex implementation files are in EDIF.
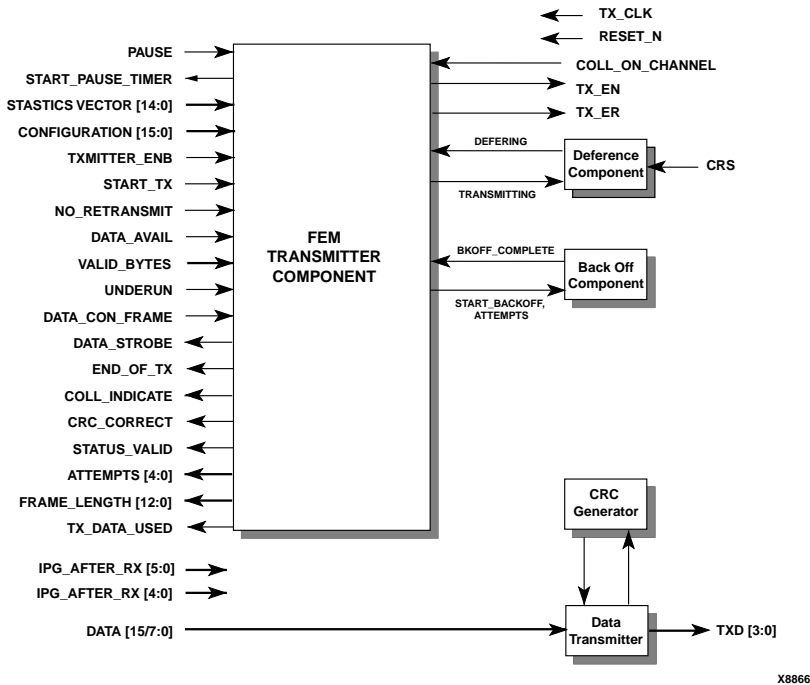4. IEEE 802.3 specification requires only 25 MHz.

Figure 1: FEM Transmitter Block Diagram

Applications

The Fast Ethernet Media Access Controller cores can be used to implement a multi-channel MAC chip with other common functions like a linked list buffer manager and a DMA control function. This MAC can be used for any applications requiring CSMA-CD protocol for media access. It can be used for Ethernet switches, hubs and network interface cards (NICs). It is also possible to purchase and use either core separately, for systems that only require a transmit or receive function which would allow them fit in a Xilinx Spartan FPGA for a very low-cost implementation.

General Description

The cores are suited to be combined with other higher level functions in a single Xilinx FPGA. Each core been designed according to the recommendations of the Virtual Socket Interface (VSI) Alliance.

Functional Description

The FEM provides a detailed and comprehensive statistics vector reporting regarding frame transmission and reception. The frame statistics vector information can be used to implement Ethernet Management Information Base (MIB) counters to facilitate support for RMON and SNMP management. The cores have been implemented and tested in

Xilinx FPGAs in a evaluation NIC board that is available for purchase separately from CoreEl MicroSystems.

The FEM transmitter and receiver modules are fully compliant to the 802.3 specification. The designs are fully synchronous to the Transmit Clock (TX_CLK) and Receive Clock (RX_CLK) signals from the MII interface, respectively.

FEM Transmitter Core

One side of the FEM transmitter provides a media independent interface (MII) to a PHY interface core. On the other side a host interface provides for easy interfacing to a FIFO or RAM.

Frame transmission is initiated from the host. Frame data is provided to the transmitter core via a data bus that can be specified at either 8- or 16-bits at the time of purchase. In the source code version of this core, users can configure this interface themselves. The user of the core then designs a simple control function to read data out of a FIFO and initiate a frame transfer through the transmitter core.

The functional block diagram of the transmitter core is shown in Figure 1, and described below.
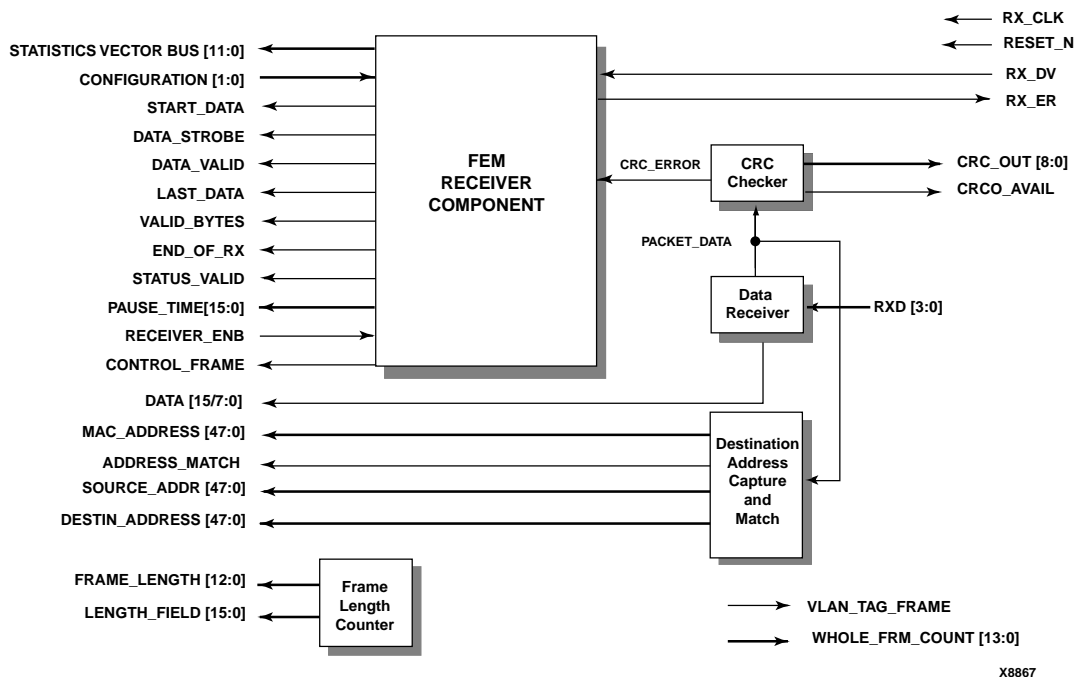
Figure 2: FEM Receiver Block Diagram

FEM Transmitter Component

The main IEEE 802.3 Media Access Control Transmit function has been implemented in this component. This block is responsible for the control of various blocks and the interfaces on either side of the core, i.e. the host interface and the MII interface.

Data Transmitter

This component acts as the data path for the transmitter. It is responsible for transmitting the host frame data received on the 8/16 bit DATA[15/7:0] bus onto the MII TXD[3:0] bus under control of the FEM Transmitter Component. Under control of the Transmitter component it also generates and transmits preamble, start of frame, jam, pad data onto the TXD bus.

Back Off Component

When a transmission attempt has been terminated due to a collision it is retried. The scheduling of the retransmission is determined by a controlled randomized process referred to as the truncated binary exponential backoff. A maximal length pseudo random sequence generator is seeded and is used to generate the integer.

Deference Component

Even when there is nothing to transmit the deference machine monitors the medium by watching the carrier sense signal. Whenever the medium is busy the transmitter defers to the passing frame by delaying its transmission. When the medium is free the deference machine continues to defer until the proper inter frame gap time has been reached. At this point the deference machine removes its deference on signal and the transmitter is free to start transmission.

CRC Generator

This component calculates a running CRC-32 on all transmitted octets. The CRC-32 calculator is based on a parallel 8 bit implementation. At the end of the data phase of the frame the 32-bit CRC is bit-reversed and the 4 octets are transmitted at the end of the frame.

FEM Receiver Core

The FEM receiver core implements the 802.3 Fast Media Access Control layer function. It provides a simple interface to transfer frame data from the MAC layer to the host. The width of the data transfer bus can be specified at either 8- or 16-bits at the time of purchase. In the source code version of this core, users can configure this interface themselves. The core interfaces to the PHY via a standard MII.

Frame reception starts if the receiver has been enabled and receive data valid (RX_DV) becomes true on the MII interface. The FEM receiver then looks for the preamble and start of frame delimiter pattern. The FEM receiver then switches to receive frame data and transfer it to the host via a 8- or 16-bit interface.

The functional block diagram of the receiver core is shown in Figure 2 and described below.

FEM Receiver Component

The main IEEE 802.3 Media Access Control Receive function has been implemented in this component. This block is responsible for the control of various blocks and the interfaces on either side of the FEM, i.e. the host interface and the MII interface.

Data Receiver

This block performs the function of serial to parallel conversion. According to the BITS_16 input to the receiver core, this block converts the incoming 4 bit data to either 8- or 16-bit data.

CRC Checker

This block calculates the CRC for the incoming data and compares it with the CRC appended to the incoming frame at the end. If the two do not match, an error is reported on the statistics output.

Destination Address Capture and Match

This block extracts the destination address from the incoming frame and compares it with the DESTN_ADDRESS at the input of the FEM receiver. If they match, the ADDRESS_MATCH signal is asserted.

Frame Length Counter

This block captures the length field from the incoming frame. It also calculates the number of data bytes received for the received frame. The value of both these frame lengths are available as the output.

Table 1: FEM Transmitter Core Signal Pinout

Signal	Signal Direction	Description
Flow Control Support (802.3x) Signals		
PAUSE	Input	TRUE tells Tx to stop frame transmission; FALSE tell Tx to start new frame transmission; control frames can transmit even when Tx paused.
START_PAUSE_TIMER	Output	Starts Pause Timer count down; when count=0, input signal PAUSE=FALSE and Tx can start a new frame transmission. Pause time implemented externally.
Statistics Vector [14:0] Signals (not broken out in Figure 1)		
STS_TX_SUCCESS	Output	Valid when STATUS_VALID asserted by Tx; indicates a successful frame transmission.
STS_TX_COLL	Output	Valid when STATUS_VALID asserted by Tx; indicates a collision was encountered on medium. Other collision status bits further detail type of collision.
STS_BROADCAST_FRAME	Output	Valid when STATUS_VALID asserted by Tx; indicates a broadcast frame was transmitted.
STS_MULTICAST_FRAME	Output	Valid when STATUS_VALID asserted by Tx; indicates a multicast frame was transmitted.
STS_LATE_COLL	Output	Valid when STATUS_VALID asserted by Tx; indicates a collision was encountered on channel later than slot time.
STS_FRAME_PADED	Output	Valid when STATUS_VALID asserted by Tx; indicates frame had to be padded in order to meet minimum frame size requirement.
STS_MAX_FRAME_ERR	Output	Valid when STATUS_VALID asserted by Tx; indicates host attempted to send a frame longer than maximum frame size if LONG_FRAME_ENABLE=FALSE.
STS_LONG_FRAME_ERR	Output	Valid when STATUS_VALID asserted by Tx; indicates host attempted to send a frame greater than long frame size if LONG_FRAME_ENABLE=TRUE.

Signal	Signal Direction	Description
STS_MAXATTEMPT_ERR	Output	Valid when STATUS_VALID asserted by Tx; indicates it has reached its maximum number of attempts to transmit a frame. Transmit attempt for this frame will be aborted due to multiple collisions.
STS_FRAME_DEFERED	Output	Valid when STATUS_VALID asserted by Tx; indicates frame deferred to a passing frame.
STS_EXCESS_DEFER	Output	Valid when STATUS_VALID asserted by Tx; indicates frame transmission aborted due to deference exceeding maximum limit.
STS_LOSS_OF_CRS	Output	Valid when STATUS_VALID asserted by Tx; indicates CRS never became true or CRS disappeared once or multiple times during frame transmission.
STS_UNDERUN_ERR	Output	Valid when STATUS_VALID asserted by Tx; indicates host under run during frame transmission. Tx aborted transmission of cell in middle.
STS_CONTROL_FRAME	Output	Valid when STATUS_VALID asserted by Tx; indicates host transmitted a Control Frame.
STS_PAUSE_FRAME	Output	Valid when STATUS_VALID asserted by Tx; indicates host transmitted a Pause Frame.
Configuration [15:0] Signals (not broken out in Figure 1)		
FULL_DUPLEX	Input	TRUE enables full duplex mode of operation.
LONG_FRAME_ENABLE	Input	TRUE enables long frame transmission greater than maximum frame size. Long frame size can be specified.
DISABLE_CRC_GEN	Input	TRUE disables CRC generation by Tx; switchable on a frame-by-frame basis.
BITS_16	Input	TRUE=DATA[15:0]; FALSE=DATA[7:0]; bus is pre-configured for Xilinx netlist version of core.
ENABLE_PAD	Input	TRUE enables short frame padding.
RANDOM_SEED[9:0]	Input	Seed value for random number generator used to compute backoff
LOAD_RANDOM_SEED	Input	Goes high for a period more than one clock in order to load RANDOM_SEED into backoff random number generator.
Host Interface		
TXMITTER_ENB	Input	Enables Tx.
START_TX	Input	Single clock pulse input starts a frame transmission.
NO_RETRANSMIT	Input	If TRUE then Tx aborts retransmission after a collision.
DATA_AVAIL	Input	TRUE signals to Tx that DATA[15/7:0] is valid; end of frame data signaled by de-asserting this signal.
VALID_BYTES	Input	Specifies number of octets to be transmitted from last DATA.
UNDERRUN	Input	TRUE signals Tx host is incapable of providing more data for frame because of a transmit buffer underrun.
DATA_CON_FRAME	Input	TRUE: Tx interprets frame to be transmitted as a data frame; FALSE: frame interpreted as a control frame; Tx does not inhibit transmission of a control frame even when pausing for flow control purposes.
DATA_STROBE	Output	TRUE signals host that current data on DATA[15/7:0] has been accepted and next data to be made available in next clock cycle.
END_OF_TX	Output	Asserted for one clock cycle by Tx indicating completion of frame transmission attempt.
COLL_INDICATE	Output	Valid when STATUS_VALID asserted by Tx; indicates early collision.
CRC_CORRECT	Output	Indicates CRC received from host was correct.
STATUS_VALID	Output	Asserted for one clock cycle by Tx signifying transmit frame statistics vector valid.
ATTEMPTS[4:0]	Output	Bundle valid when STATUS_VALID asserted by Tx and indicates number of attempts made to transmit frame.

Signal	Signal Direction	Description
FRAME_LENGTH[12:0]	Output	Bundle valid when STATUS_VALID asserted by Tx and indicates length of frame transmitted.
TX_DATA_USED	Output	Indicates that data transmission started on the M11 Interface
IPG_AFTER_TX[4:0]	Input	Programmable Inter Packet Gap value for full IPG timeout; value specified in nibbles.
IPG_AFTER_RX[4:0]	Input	Programmable Inter Packet Gap value for part1[2/3] of IPG timeout; value specified in nibbles.
DATA[15/7:0]	Input	Frame data provided on this bus in 8- or 16-bit format depending on configuration of core.
MII Interface		
TX_CLK	Input	Transmit Clock; uses 1 CLKIOB pin.
RESET_N	Input	Asynchronous Reset for core.
COLL_ON_CHANNEL	Input	Collision on MII.
TX_EN	Output	Transmit Enable on MII.
TX_ER	Output	Transmit Error on MII.
CRS	Input	Carrier Sense on MII.
TXD[3:0]	Output	Transmit Data on MII.

Table 2: FEM Receiver Core Signal Pinout

Signal	Signal Direction	Description
Statistics Vector Bus [11:0] Signals (not broken out in Figure 2)		
STS_RX_SUCCESS	Output	Valid when STATUS_VALID asserted by Rx; indicates successful frame reception.
STS_SHORT_FRAME	Output	Valid when STATUS_VALID asserted by Rx; indicates an individual frame which less than minimum frame size (64 octets) has been received.
STS_CRC_ERROR	Output	Valid when STATUS_VALID asserted by Rx; indicates frame received has a CRC error.
STS_MAXFRAME_ERR	Output	Valid when STATUS_VALID asserted by Rx; indicates frame being received has exceeded max limit.
STS_LONGFRAME_ERR	Output	Valid when STATUS_VALID asserted by Rx; indicates frame being received has exceeded max limit.
STS_DRIBBLE_NIBBLE	Output	Valid when STATUS_VALID asserted by Rx; indicates a one nibble, not full octet was not received at end of frame.
STS_IPG_SHRINK	Output	Valid when STATUS_VALID asserted by Rx; indicates insufficient gap between frames.
STS_PREAM_SHRINK	Output	Valid when STATUS_VALID asserted by Rx; indicates a preamble shrinkage in received frame.
STS_PAUSE_FRAME	Output	Valid when STATUS_VALID asserted by Rx; indicates a PAUSE frame (as specified IEEE 802.3x full duplex mode of operation) received.
STS_CARR_EVNT_SEEN	Output	Valid when STATUS_VALID asserted by Rx; indicates a carrier event was observed (before current frame) - i.e, carrier was valid but no valid preamble or SFD (start of frame delimiter) was detected.
BROADCAST_FRAME	Output	Asserted when a broadcast frame received.
MULTICAST_FRAME	Output	Asserted when a multicast frame received.
Configuration [1:0] Bus Signals (not broken out in Figure 2)		
LONG_FRAME_ENABLE	Input	TRUE enables long frame reception greater than maximum frame size; long frame size can be specified.

Signal	Signal Direction	Description
BITS_16	Input	TRUE=DATA[15:0]; FALSE=DATA[7:0]; bus is pre-configured for Xilinx netlist version of core.
Host Interface		
START_DATA	Output	Indicates first data strobe for frame.
DATA_STROBE	Output	Indicates frame data available on DATA[15/7:0].
DATA_VALID	Output	Asserted by Rx with START_DATA until end of last data.
LAST_DATA	Output	Indicates last data strobe by asserting for one clock cycle.
VALID_BYTES	Output	Specifies number of valid octets on DATA[15/7:0] during last data strobe.
END_OF_RX	Output	Asserted for one clock cycle indication frame reception completed.
STATUS_VALID	Output	Asserted for one clock cycle indicating valid receive frame status signals.
PAUSE_TIME[15:0]	Output	Valid when STATUS_VALID asserted by Rx; specifies pause time for Tx in slot times.
RECEIVER_ENB	Input	Enables Rx.
CONTROL_FRAME	Output	Asserted whenever control frame is received
DATA[15/7:0]	Output	Frame data provided on this bus in 8- or 16-bit format depending on configuration of core.
MAC_ADDRESS[47:0]	Input	MAC address; Rx compares incoming frame destination address with MAC address.
ADDRESS_MATCH	Output	TRUE if match occurs between MAC_ADDRESS and DESTIN_ADDRESS of incoming frame.
SOURCE_ADDR[47:0]	Output	Source address in received frame.
DESTIN_ADDRESS[47:0]	Output	Destination address in received frame.
FRAME_LENGTH[12:0]	Output	Valid when STATUS_VALID asserted by Rx; indicates length of frame received.
LENGTH_FIELD[15:0]	Output	Frame length in LENGTH/TYPE field of received frame.
CRCOUT[8:0]	Output	9 MS bits of 32-bit CRC checker after frame destination address field received; can be used as key for hash address table lookup.
CRCO_AVAIL	Output	TRUE during frame indicates valid value on CRCOUT for frame; also indicates destination address field received by Rx.
VLAN_TAG_FRAME	Output	TRUE indicates reception of VLAN tagged frame.
WHOLE_FRM_COUNT[13:0]	Output	Nibble Count of whole frames.
MII Interface		
RX_CLK	Input	Receive Clock; uses 1 CLKIOB pin.
RESET_N	Input	Asynchronous Reset for core.
RX_DV	Input	Receive data valid on MII.
RX_ER	Input	Receive Error on MII.
RXD [3:0]	Input	Receive Data on MII.

Core Modifications

Normally, modifications are not possible by the user since the cores are provided in a Xilinx netlist format. CoreEl can perform special modifications for additional charge. However source code is available for each core for additional cost where the customer can make modifications. Contact CoreEl MicroSystems for more information.

Eight and 16-bit host interface datapath versions of the Xilinx netlist version of each core are available. Customers will need to specify which is preferred at the time of purchase.

Pinout

The pinout is not fixed to any specific device I/O. Signal names for the transmitter and receiver cores are provided in the block diagrams shown in Figures 1 and 2, and described in Tables 1 and 2, respectively.

Verification Methods

The FEM core has been used in larger ASICs and is silicon proven. The FPGA verification was done by back annotating the implementation and simulating in a Model technology V-System environment.

In addition, CoreEl has built and tested a Xilinx-based hardware evaluation board for the FEM core. This board is available for purchase separately from CoreEl (see Available Support Products).

Recommended Design Experience

To implement a complete design with the FEM core, the user should have experience in the following areas:

1. Complete understanding of the IEEE specs for MAC

2. Understanding of Xilinx-FPGA architecture

3. Familiarity with simulation, synthesis and Xilinx tool environments

4. Knowledge of Ethernet and networking standards will be an added advantage.

Available Support Products

CoreEl provides a complete line of support products to help you evaluate and integrate this core into your FPGA design. These products are listed below and must be purchased separately.

Apart from those listed below, RMON, MIB counters and full/half duplex flow control modules are some of the products that can be incorporated with the FEM core for feature enhancement. The Xilinx FPGA versions of these modules will be available in the near future.

FEM Test Bench

The FEM test bench facilitates simple, flexible and thorough testing of FEM cores, conforming to IEEE 802.3u specifications. This script based test bench allows user to test all the possible fault conditions of the FEM using simple commands supported by the test bench. The test bench maintains a record of the status vectors generated for each received or transmitted packet through the FEM. Corresponding report files are generated.

Evaluation Network Interface Card (NIC)

The 10/100 Ethernet network interface adapter is an ISA plug-in card for personal computers (PC-AT). The card is developed to test the FEM core of CoreEl MicroSystems. It is 100% compatible with industry standard Ethernet NICs.

The card comes with a device driver for Windows NT 4.0, so that it can be tested in a normal LAN environment. The card is designed using 2 Xilinx 4028EX FPGAs, one containing the FEM core and the other containing the buffer controller and bus interface logic.

Ordering Information

When inquiring about or ordering these cores, please specify which product(s) you are interested in:

- Transmitter core (8- or 16-bit host data bus)
- Receiver core (8- or 16-bit host data bus)
- Netlist version of cores must be specified prior to purchase

For more information on this or other products mentioned in this datasheet, contact CoreEl MicroSystems directly.

Related Information

Xilinx Programmable Logic

For information on Xilinx programmable logic or development system software, contact your local Xilinx sales office, or:

Xilinx, Inc.
2100 Logic Drive
San Jose, CA 95124
Phone: +1 408-559-7778
Fax: +1 408-559-7114
URL: www.xilinx.com

For general Xilinx literature, contact:

Phone: +1 800-231-3386 (inside the US)
            +1 408-879-5017 (outside the US)
E-mail: literature@xilinx.com

For AllianceCORE™ specific information, contact:

Phone: +1 408-879-5381
E-mail: alliancecore@xilinx.com
URL: www.xilinx.com/products/logicore/alliance/
            tblpart.htm