Tri-Speed Ethernet MAC IP Core - Lattice Radiant Software

User Guide

FPGA-IPUG-02084-1.2

June 2020

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS and with all faults, and all risk associated with such information is entirely with Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
APB	Advanced Peripheral Bus
AHB	Advanced High-Performance Bus
FIFO	First In First Out
LMMI	Lattice Memory Mapped Interface
MIIM	Media Independent Interface Management Module
RTL	Register Transfer Language
SGMII	Serial Gigabit Media Independent Interface
TSEMAC	Tri-Speed Ethernet Media Access Controller

1. Introduction

Tri-Speed Ethernet Media Access Controller (TSEMAC) IP core is a complex core containing all necessary logic, interfacing and clocking infrastructure necessary to integrate an external industry-standard Ethernet PHY with an internal processor efficiently and with minimal overhead.

The TSEMAC IP core supports the ability to transmit and receive data between the standard interfaces, such as APB or AHB-Lite, and an Ethernet network. The main function of TSEMAC IP is to ensure that the Media Access rules specified in the 802.3 IEEE standard are met while transmitting a frame of data over Ethernet. On the receiving side, the TSEMAC extracts different components of a frame and transfers them to higher applications through the FIFO interface.

Table 1.1 presents a summary of the Tri-Speed Ethernet MAC IP.

1.1. Quick Facts

Table 1.1. TSEMAC IP Quick Facts

IP Requirements	Supported FPGA Families	CrossLink™-NX, Certus™-NX
Resource Utilization	Targeted Devices	LIFCL-40, LIFCL-17, LFD2NX-40
	Supported User Interfaces	APB, AHB-Lite, AXI4-Stream
	Resources	See Table A.1. Resource Utilization.
Design Tool Support	Lattice Implementation	Lattice Radiant Software 2.1
	Synthesis	Lattice Synthesis Engine
		Synopsys® Synplify Pro for Lattice
	Simulation	For a list of supported simulators, see the Lattice Radiant Software 2.1 User Guide.

1.2. Features

Key features of the Tri-Speed Ethernet Media Access Controller (TSEMAC) IP include:

- Compliant to IEEE 802.3-2005 standard
- 8-bit wide internal data path
- Full-duplex operation in 1G mode
- Full- and half-duplex operation in 10/100 mode
- Transmit and receive statistics vector
- Programmable Inter-Packet Gap (IPG)
- Multicast address filtering
- Selectable MAC operating options:
  - Classic TSEMAC with G/MII-*/
  - Gigabit MAC with GMII
  - SGMII Easy Connect MAC with GMII
- Host control interface configurable to either APB or AHB-Lite
- Interrupt interface
- Compliant to IEEE 802.1AE MACsec Standard (connectionless data confidentiality, data integrity, and data origin authenticity by media access independent protocols)

The Tri-Speed Ethernet Media Access Controller IP supports:

- Full-duplex control using PAUSE frames
- VLAN tagged frames
- Automatic re-transmission on collision
- Automatic padding of short frames
- Multicast and Broadcast frames
- Optional FCS transmission and reception
- Optional MII management interface module
- Jumbo frames of any length

1.3. Conventions

1.3.1. Nomenclature

The nomenclature used in this document is based on Verilog HDL.

1.3.2. Signal Names

Signal names that end with:

- _*n* are active low (asserted when value is logic 0)
- _*i* are input signals
- _*o* are output signals

1.3.3. Attribute Names

Attribute names in this document are formatted in title case and italicized (*Attribute Name*).

2. Functional Descriptions

2.1. Overview

TSEMAC IP transmits and receives data between a client application and an Ethernet network. The main function of the Ethernet MAC is to ensure that the Media Access rules specified in the 802.3 IEEE standard are met, while transmitting and receiving Ethernet frames.

The TSEMAC IP core is a fully synchronous machine composed of Transmit and Receive MAC sections that operate independently to support full duplex operation.

In the 1G mode, the 125 MHz system clock is supplied to the Transmit MAC. The system clock is used to clock the GMII interface for data transmission. When receiving data, an external PHY device provides the 125 MHz clock to the GMII receive section.

In the 10/100 mode, an external PHY device supplies the 25 MHz clock to the Transmit MAC and the Receive MAC.

Figure 2-1, Figure 2-2 and Figure 2-3 show select frame formats of data transmitted and received on the Ethernet network that TSEMAC IP core supports.

PREAMBLE	SFD	DESTINATION ADDRESS	SOURCE ADDRESS	LENGTH/ TYPE	DATA/PAD	FRAME CHECK SEQUENCE
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	46-1500 bytes	4 bytes

Figure 2-1. Un-Tagged Ethernet Frame Format

PREAMBLE	SFD	DESTINATION ADDRESS	SOURCE ADDRESS	VLAN TAG HEADER	LENGTH/ TYPE	DATA/PAD	FRAME CHECK SEQUENCE
7 bytes	1 byte	6 bytes	6 bytes	4 bytes	2 bytes	46-1500 bytes	4 bytes

Figure 2-2. VLAN-Tagged Ethernet Frame Format

PREAMBLE	SFD	DESTINATION ADDRESS 01-80-C2-00-00-01	SOURCE ADDRESS	LENGTH/ TYPE 88-08	MAC CTL OP_CODE 00-01	OP_CODE PARAMS/RSV	FRAME CHECK SEQUENCE
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	2 bytes	60 bytes	4 bytes

Figure 2-3. Ethernet Control Pause Frame Format

On the receiving side, the Ethernet MAC extracts various components of a frame and transfers them to higher applications through LMMI interface. Data received from the G/MII interface is first buffered until sufficient data is available to be processed by the Receive MAC (Rx MAC). The Preamble and the Start-of-Frame Delimiter (SFD) information are then extracted from the incoming frame to determine the start of a valid frame. The Receive MAC checks the address of the received packet and validates whether the frame can be received before transferring it into the FIFO. Only valid frames are transferred into the FIFO (runts and fragments are discarded). The Rx MAC also provides a statistics vector on a 'per packet' basis that can be used by the application. The TSEMAC IP core always calculates CRC to check whether the frame was received error-free.

On the transmit side, the Tx MAC is responsible for controlling access to the physical medium. The Tx MAC reads data from Tx FIFO, formats this data into an Ethernet packet and passes it to the G/MII module. The Tx MAC reads data from the Tx FIFO when a packet is available, and the Tx MAC is in its appropriate state. The Tx MAC pre-fixes the Preamble and the Start-of-Frame Delimiter information to the data and appends the Frame Check Sequence at the end of the data. In half-duplex operation, the Tx MAC stores the first 64 bytes of data from FIFO in an internal buffer, to be used in re-transmitting data on collisions.

The SGMII Easy Connect configuration option adds pins and logic for seamless connection to the Lattice Gigabit Ethernet PCS IP core.

A Tagged frame includes a 4-byte VLAN Tag field, which is located between the Source Address field and the Length/Type field. The VLAN Tag field includes the VLAN Identifier and other control information needed when operating with Virtual Bridged LANs as described in IEEE P802.1Q.

2.1.1. Configuration Options

2.1.1.1. Classic TSEMAC Option

When the Classic TSEMAC option is selected, the TSEMAC IP core can be configured to operate in either the 1G mode (1000Mbps data rate) or the Fast Ethernet mode (10/100 Mbps data rate) by setting an internal register bit. A block diagram of the Classic TSMAC IP core option is shown in Figure 2.4.

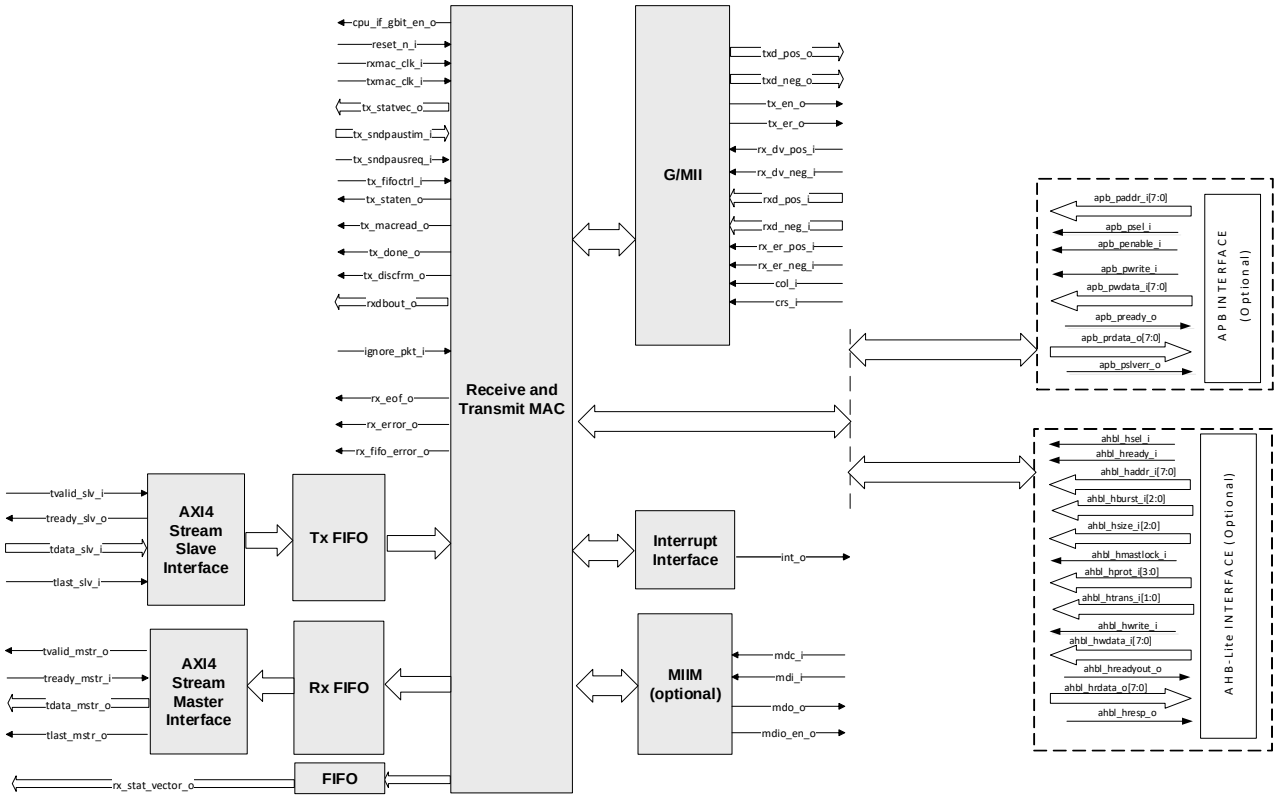

Figure 2.4 Top-Level Block Diagram for the Classic TSMAC IP Option

2.1.1.2. Gigabit MAC or SGMII Configurable Options

For the Gigabit MAC configuration option, the TSEMAC always operates at the Gigabit data rate and is effectively configured as a full-duplex Gigabit MAC only. For the SGMII Easy Connect configuration option, the TSEMAC operates at the Gigabit data rate and uses clock enables provided by the SGMII PCS IP core to work at the three different speeds.

A block diagram of the Gigabit MAC / SGMII Easy Connect configurations is shown in Figure 2.5.

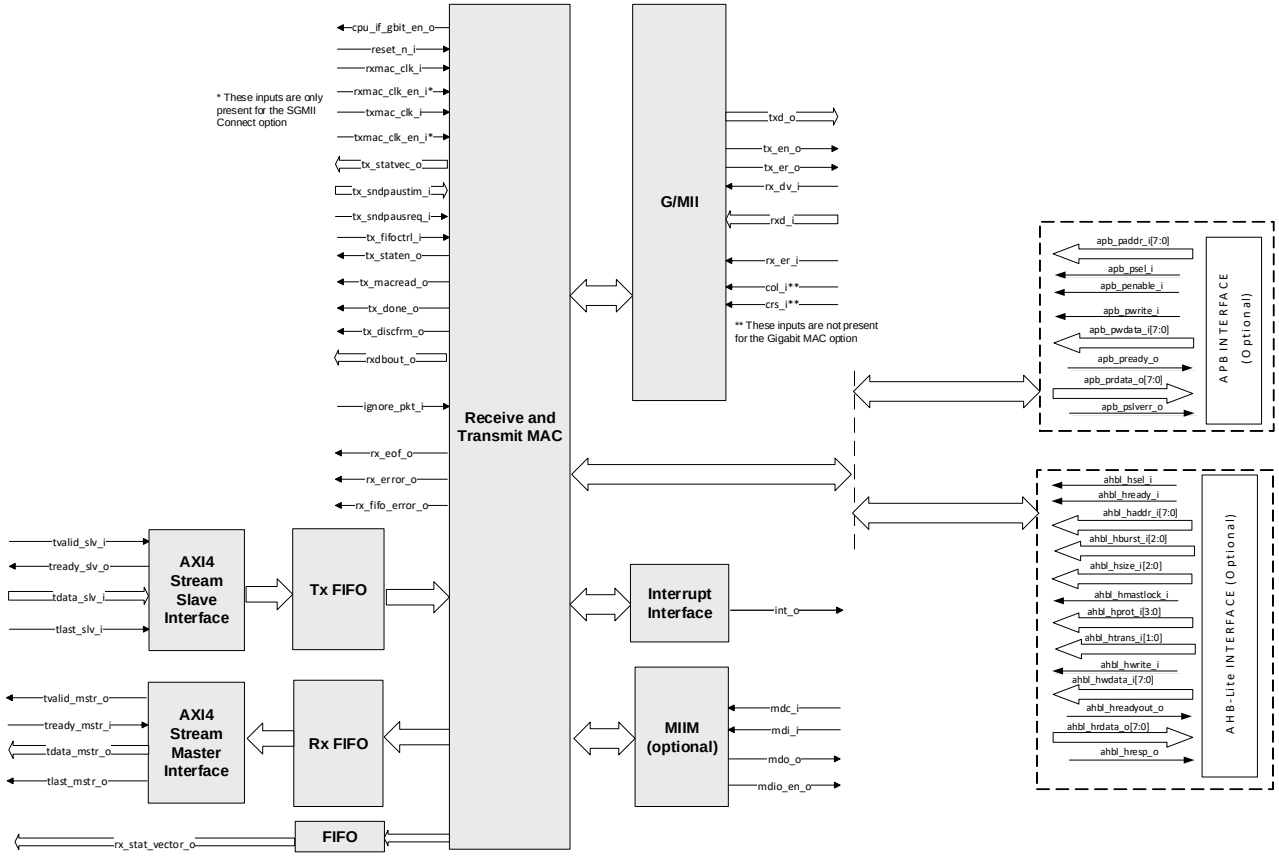

Figure 2.5. Top-Level Block Diagram for the One Gigabit and SGMII Options

2.1.2. Receive MAC (Rx MAC)

The main function of Rx MAC is to accept formatted data from G/MII interface and pass it to LMMI interface through FIFO. During this, Rx MAC performs following functions:

- Detect the start of the frame
- Compare MAC address
- Re-calculate CRC
- Process the control frame and pass it to the flow control module.

Rx MAC operation is determined by programming Mode and Transmit and Receive Control Registers (register definitions and bit descriptions can be found in the Register Description section) Note that setting the Gbit_en bit in the Mode Register to High sets TSEMAC IP to operate in 1G mode, whereas setting the Gbit_en bit to Low sets the TSEMAC IP to operate in 10/100 Mode. Events that occur during reception of a frame are logged into the rx_stat_vector_o signal and the Transmit and Receive Status Register. TSEMAC IP can report information about miscellaneous events like:

- FIFO overflow
- CRC error
- Receive error
- Short frame reception
- Long frame reception
- IPG violation

By default, the entire frame (except Preamble and SFD bytes) is sent to FIFO via Rx MAC application interface signals. If a user does not want to receive the FCS, the core can be programmed to strip the FCS field as well as any PAD bytes in the frame and send the rest to the FIFO. Rx MAC section operates on the rxmac_clk_i derived from the rx_clk sourced from the PHY. All the signals on Receive MAC FIFO interface are synchronous to this clock. Rx MAC is disabled while Rx_en is Low (Bit_2 of the Mode Register) and should be enabled only after the associated registers are properly initialized.

2.1.2.1. Receiving Frames

The frames received by Rx MAC are analyzed and the Preamble and SFD bytes are stripped off the frame before it is transferred to the Rx FIFO. LMMI interface which is used to read Rx data from Rx FIFO is eight bits wide. Default operation for Rx MAC is to transfer the unmodified frame after stripping off the Preamble and SFD bytes. This behavior can be changed by setting bit [1] of the Transmit and Receive Control Register. When bit [1] is set, Rx MAC strips the Preamble, SFD, FCS bytes and the PAD bytes, if any. Note that Rx MAC assumes that received frame has PAD bytes if a 64-byte packet is received with its Length/Type field set to the value of less than 46 bytes.

2.1.2.2. Address Filtering

Rx MAC offers several address filtering methods users can utilize to effectively block unwanted frames. It also provides a Promiscuous mode in which all supported filtering schemes are abandoned and Rx MAC transfers all the frames irrespective of the address they contain. By default, Rx MAC is configured to filter and discard Broadcast frames (i.e. all bits of the received DA == 1) and multicast frames (i.e. bit[0] of the received DA == 1). The MAC can be configured to receive broadcast frames by setting bit [7] of the Transmit and Receive Control Register. Multicast frames are received only when bit [4] of the Transmit and Receive Control Register is set. When set, multicast frames are subject to filtering that is dependent on a 64-bit hash table lookup. The 64-bit hash table is organized as eight 8-bit registers. The six middle bits of the most significant byte of the CRC calculated for the destination address field of the frame, are used to address one of the 64 bits of the hash table. The three most significant bits of the calculated CRC select one of the eight tables, and the three least significant bits select a bit. The frame is received only if the retrieved bit is set. The IP registers specifying the hash tables contents are described in the Register Description section. All other regular frames are filtered based on the Rx MAC address programmed into the MAC Address (0,1,2) Registers.

2.1.2.3. Filtering Based on Frame Length

The default minimum Ethernet frame size is 64 bytes. Any frame smaller than 64 bytes could be a collision fragment. By default, Rx MAC is configured to ignore bytes shorter than 64 bytes. You can configure the MAC to receive shorter frames by setting bit [8] of the Transmit and Receive Control Register. Whenever a short frame is received, the appropriate bit is set in the statistics vector, marking it as a Short frame. Rx MAC has been designed to receive frames larger than the standard specified maximum as easily as any other frame, thus, it may be used in environments that generate jumbo frames. However, for statistics purposes, users can set the maximum length of the frame in the Maximum Packet Size Register. When the received frame is larger than the number in this register, bit [31] of the Receive Statistics Vector bus is set, marking it as a Long frame.

2.1.2.4. Receiving a Pause Frame

When Rx MAC receives a Pause frame, the Tx MAC continues with the current transmission, then pauses for the duration indicated in the Pause time. During this time, the Tx MAC can transmit Control frames. Although Pause frames may contain the Multicast Address, Multicast filtering rules do not apply to them. If bit [3] of the Transmit and Receive Control Register is set, the Rx MAC signals the Tx MAC to stop transmitting for the duration specified in the frame. If this bit is reset, the Rx MAC assumes the Tx MAC does not have the Pause capability and/or does not wish to be paused and so Rx MAC does not signal it to stop transmitting. If the drop control, bit[6] in the Transmit and Receive Control Register, is set, the Pause frame is received but dropped internal to the MAC and is not transferred to the FIFO. Otherwise, the Pause frame is received and transferred to FIFO.

2.1.2.5. Statistics Vector

By default, Statistics Vector is generated for all received frames transferred to the FIFO. If users want the Rx MAC to ignore all incoming frames, the input signal ignore_next_pkt_i must be asserted. The frame that should have been received is consequently ignored, and Rx MAC sets the Packet Ignored bit (bit 26) of the Statistics Vector. The Maximum Packet Size Register is programmed by the user as a threshold for setting the Long Frame bit of the Statistics Vector. This value is only used for the un-tagged frames. Receive MAC adds “4” to the value specified in this register for all VLAN Tagged frames when checking against the number of bytes received in the frame. This is because all VLAN Tagged frames have additional four bytes of data. When a tagged frame is received, the entire VLAN Tag field is stored in the VLAN_TAG Register. Additionally, every time a Statistics Vector is generated, some of the bits are written into the corresponding bit locations [9:1] of the Transmit and Receive Status Register. This is done so that you can get this information via the Host interface. The description of the bits in the Statistics Vector bus is shown in Table 2.1.

Table 2.1. Receive Statistics Vector Description

Bit	Description
31	Long Frame. This bit is set when a frame longer than specified in the MAX_PKT_SIZE Register is received
30	Short Frame. This bit is set when a frame shorter than 64 bytes is received
29	IPG Violation. This bit is set when a frame is received before the IPG timer runs out (96 bit times)
28	Not Used. This bit always returns a zero
27	Carrier Event Previously Seen. When asserted, indicates that a carrier event was detected since the last frame
26	Packet Ignored. When set, this bit indicates the incoming packet is to be ignored
25	CRC Error. This bit is set when a frame is received with an error in the CRC field
24	Length Check Error. This bit is set if the number of data bytes in the incoming frame do not match the value in the length field of the frame
23	Receive OK. This bit is set if the frame is received without any error
22	Multicast Address. This bit is set to indicate that the received frame contains a Multicast Address
21	Broadcast Address. This bit is set to indicate that the received frame contains a Broadcast Address
20	Dribble Nibble. This bit is set when only four bits of the data presented on the RS interface are valid
19	Unsupported Opcode. This bit is set if the received control frame has an unsupported opcode. In this version of the IP, only the opcode for Pause frame is supported
18	Control Frame. This bit is set to indicate that a Control frame was received
17	Pause Frame. This bit is set when the received Control frame contains a valid Pause opcode
16	VLAN Tag Detected. This bit is set when TSEMAC IP receives a VLAN Tagged frame

Bit	Description
15:0	Frame Byte Count. This bus contains the length of the frame that is received. The frame length includes the DA, SA, L/T, TAG, DATA, PAD and FCS fields

2.1.3. Transmit MAC (Tx MAC)

Transmit MAC (Tx MAC) is responsible for controlling access to the physical medium. Tx MAC reads data from a Tx FIFO when the FIFO is not empty and when it detects an active tx_fifoavail. Tx MAC then formats this data into an Ethernet packet and passes it to the G/MII module. Tx MAC is disabled while Tx_en is Low (Bit_3 of the Mode Register) and should only be enabled after the associated registers are properly initialized. Once enabled, Tx MAC continuously monitors FIFO interface for an indication that frame(s) are ready to be transmitted. In the 1G mode, Tx MAC and Tx FIFO interface operations are synchronous to txmac_clk derived from sys_clk. In the 10/100 mode, the Tx MAC is clocked by txmac_clk (derived from the tx_clk supplied from the PHY device). Tx FIFO interface signals are always synchronous to txmac_clk. In 10/100 mode, Tx MAC can be configured to operate in either half-duplex or full-duplex modes. This is done by writing to bit[5] of the Transmit and Receive Control Register. In full-duplex operation, receiver's buffer may quickly fill up. When this happens, the receiver sends flow control (Pause) frames to the transmitter, requesting for it to stop transmitting. Once the receiver is able to free up its buffers, transmitter finishes transmitting current frame and stops for the duration specified in the Pause frame.

2.1.3.1. Transmitting Frames

By default, Transmit MAC is configured to generate the FCS pattern for the frame to be transmitted. However, this can be prevented by setting bit[2] of the Transmit and Receive Control Register. This feature is useful if the frames being presented for transmission already contain the FCS field. When FCS field generation is disabled, it becomes user's responsibility to ensure that short frames are properly padded before the FCS is generated. If MAC receives a frame shorter than 64 bytes when FCS generation is disabled, the frame is sent as is and a Statistic Vector for the condition is generated. The DA, SA, L/T, and DATA fields are derived from higher applications through LMMI interface and then encapsulated into an Un-tagged Ethernet frame. This frame is not sent over the network until the network has been idle for a minimum of Inter-Packet Gap (IPG) time. The frame encapsulation consists of adding the Preamble bits, the Start of Frame Data (SFD) bits and the CRC check sum to the end of the frame (FCS). If padding is not disabled, all short frames are padded with hexadecimal 00. Tx MAC requires a continuous stream of data for the entire frame. There cannot be any bubbles of "no data transfer" within a frame. If the MAC was able to transmit a frame without any errors, the tx_done signal is asserted. Once the transmission has ended, data on the tx_statvec_o bus is presented to the LMMI, including all the statistical information collected in the process of transmitting the frame. Data on this bus is qualified by assertion of the tx_staten signal. After Transmit MAC is done transmitting a frame, it waits for more frames from the FIFO interface. During this time, it goes to an idle state; this can be detected by reading the Transmit and Receive Status Register. Since the Mode Register can be written at any time, Tx MAC can be disabled while it is actively transmitting a frame. In such cases, MAC completely transmits the current frame and then return to the Idle state. The control registers should be programmed only after MAC has returned to the Idle state.

2.1.3.2. Transmitting PAUSE Frame

Two different methods are used for transmitting a PAUSE frame. In the first method, the application layer forms a PAUSE frame and submits it for transmission via the FIFO. In the other method, the application layer signals the Tx MAC directly to transmit a PAUSE frame. This is accomplished by asserting tx_sndpausreq_i. In this case, the Tx MAC completes transmission of the current packet and then transmit a PAUSE frame with the PAUSE time value supplied through the tx_sndpaustim_i bus.

2.1.3.3. Retries on Collision

When operating in the half-duplex mode, the Transmit MAC has the capability to perform re-transmission of frames that have experienced in-window collision up to the specified maximum. This is possible because the MAC always buffers the first 64 bytes of the frame.

If the MAC has been disabled while it is backing off (soon after a collision), it only returns to the IDLE state after it has successfully transmitted the frame or has exceeded the retry limit.

In the 10/100 mode, the Tx MAC provides the following information:

- Whether the frame deferred before transmission
- The number of times the frame experiences collision before transmission

This information is sent as a part of the statistics vector. For a frame transmitted without any errors, the statistics vector, qualified by the enable signal, is asserted along with the tx_done_o signal. When the frame experiences excessive collision or late collision, the statistics bit for the appropriate condition is set and the tx_discfrm_o signal is asserted. This indicates an error condition.

2.2. Signal Description

Table 2.2. TSEMAC IP Core Signal Description

Port Name	I/O	Width	Description
Clock and Reset			
rxmac_clk_i	In	1	Receive MAC Application Interface Clock. This clock is used by the client application and MAC. All outputs driven by the Rx MAC on the client side are synchronous to this clock. This clock's frequency is 125 MHz,12.5 MHz, or 1.25 MHz depending on the mode 1G/100/10 respectively. *Note: this clock can be viewed as a "byte" clock, since all Rx MAC bytes are aligned with this clock. This clock is derived from the system G/MII rx_clk.* Always 125 MHz in Easy Connect mode.
txmac_clk_i	In	1	Transmit MAC Application Interface Clock. This clock is used by the client application and MAC. All inputs to the Tx MAC on the client side should be synchronous to this clock. This clock's frequency is 125,12.5 or 1.25 MHz depending on the mode 1G/100/10 respectively. *Note: this clock can be viewed as a "byte" clock, since all Tx MAC bytes should be aligned with this clock. This clock is derived from the system sys_clk or tx_clk. (1G or 10/100 respectively).* Always 125 MHz in Easy Connect mode.
mdc_i	In	1	Management Data Clock. This clock is used only when the Management Interface module is implemented.
clk_i	In	1	Host Interface (APB or AHB-Lite) Clock
reset_n_i	In	1	Reset. This is an active low asynchronous signal that resets the internal registers and internal logic. When activated, the I/O signals are driven to their inactive levels.
AXI4 Stream Master Interface			
tvalid_mstr_o	Output	High	Data Validation. Indicates that the Master is driving a valid transfer. A transfer takes place when both tvalid_mstr_o and tready_mstr_i are asserted.
tready_mstr_i	Input	High	Ready to Start Transaction. Indicates that the Slave can accept a transfer in the current cycle.
tdata_mstr_o[7:0]	Output	N/A	Write data. It is the primary payload that is used to provide the data that is passing across the interface.
tlast_mstr_o	Output	High	Indicates the boundary of a packet.
AXI4 Stream Slave Interface			
tdata_slv_i[7:0]	Input	N/A	Read data. It is the primary payload that is used to provide the data that is passing across the interface.
tvalid_slv_i	Input	High	Data Validation. Read transaction is complete and tdata_slv_i contains valid data.
tready_slv_o	Output	High	Ready to Start Transaction. Indicates that the Slave can accept a transfer in the current cycle.
tlast_slv_i	Input	High	Indicates the boundary of a packet.
Interrupt			
int_o	Out	1	Interrupt. Stays high as long as any enabled interrupt is pending.
Transmit MAC Control and Status Signals			
tx_sndpaustim_i[15:0]	In	16	PAUSE Frame Timer. This signal indicates the PAUSE time value that should be sent in the PAUSE frame.
tx_sndpausreq_i	In	1	PAUSE Frame Request. When asserted, the TSEMAC IP core transmits a PAUSE frame. This is also the qualifying signal for the tx_sndpausetim_i bus.
tx_fifoctrl_i	In	1	FIFO Control Frame. This signal indicates whether the current frame in the Transmit FIFO is a control frame or a data frame. The following values apply: • 1 = Control frame 0 = Normal frame

Port Name	I/O	Width	Description
tx_staten_o	Out	1	Transmit Statistics Vector Enable. When asserted, the contents of the statistics vector bus tx_statvec_o are valid.
tx_macread_o	Out	1	Transmit FIFO Read. This is the TSEMAC IP core Transmit FIFO read request, asserted by the TSEMAC IP core when it intends to read the FIFO.
tx_statvec_o[30:0]	Out	31	Transmit Statistics Vector. This bus includes useful information about the frame that is just transmitted. The corresponding bit locations of this bus are defined as follows: • tx_statvec_o [0] - UNICAST frame • tx_statvec_o [1] - Multicast frame • tx_statvec_o [2] - BROACAST frame • tx_statvec_o [3] - Bad FCS frame • tx_statvec_o [4] - JUMBO frame • tx_statvec_o [5] - FIFO under-run • tx_statvec_o [6] - PAUSE frame • tx_statvec_o [7] - VLAN tagged frame • tx_statvec_o [21:8] - Number of bytes in the transmitted frame • tx_statvec_o [22] - Deferred transmission • tx_statvec_o [23] - Excessive deferred transmission • tx_statvec_o [24] - Late collision • tx_statvec_o [25] - Excessive collision • tx_statvec_o [29:26] - Number of early collisions • tx_statvec_o [30] - FCS generation is disabled and a short frame was transmitted
tx_done_o	Out	1	Transmit Done. This signal is asserted for one clock cycle after transmitting a frame if no errors were present in transmission.
tx_discfrm_o	Out	1	Discard Frame. This signal is asserted at the end of a frame transmit process if the TSEMAC IP core detected an error. The possible conditions are: • A FIFO under-run • Late collision (10/100 Mode only) • Excessive Collisions (10/100 Mode only) The user application normally moves the pointer to next frame in these conditions.
Receive MAC Control and Status Signals			
rx_stat_vector_o[31:0]	Out	32	Receive Statistics Vector. This bus indicates the events encountered during frame reception. This bus is qualified by the rx_stat_en_o signal. The definition of each signal is explained in the Receive MAC section of this User Guide. The corresponding bit locations of this bus are defined as follows: • rx_stat_vector_o [15:0] - Frame Byte Count • rx_stat_vector_o [16] - VLAN Tag Detected • rx_stat_vector_o [17] - Pause Frame • rx_stat_vector_o [18] - Control Frame • rx_stat_vector_o [19] - Unsupported Opcode • rx_stat_vector_o [20] - Dribble Nibble • rx_stat_vector_o [21] - Broadcast Address • rx_stat_vector_o [22] - Multicast Address • rx_stat_vector_o [23] - Receive OK • rx_stat_vector_o [24] - Length Check Error • rx_stat_vector_o [25] - CRC Error • rx_stat_vector_o [26] - Packet Ignored • rx_stat_vector_o [27] - Carrier Event Previously Seen • rx_stat_vector_o [28] – Unused • rx_stat_vector_o [29] - IPG Violation

Port Name	I/O	Width	Description
			• rx_stat_vector_o [30] - Short Frame • rx_stat_vector_o [31] - Long Frame
ignore_pkt_i	In	1	Ignore Next Packet. This signal is asserted by the host to prevent a Receive FIFO Full condition. The Receive MAC continues dropping packets as long as this signal is asserted. This is an asynchronous signal.
rx_error_o	Out	1	Receive Packet Error. When asserted, this signal indicates the packet contains error(s). This signal is qualified with the rx_eof_o signal. The rx_error_o signal is asserted for any of the following three conditions: • The rx_er* signal on the GMII is asserted by the PHY during frame reception • There are Rx FCS errors on received frames There is a length check error on the received frame
rx_fifo_error_o	Out	1	Receive FIFO Error. This signal is asserted when the Rx FIFO is full and the Rx FIFO is being written to by the Rx MAC. When this error signal is asserted the rx_write signal is de-asserted as long as the rx_fifo_error_o signal is asserted. The rx_fifo_error_o signal is de-asserted when the end of packet exits the receive FIFO.
Management Interface (Available when MIIM is selected from User Interface)			
mdc_i	In	1	Management Data Clock. This clock is used only when the Management Interface module is implemented.
mdi_i	In	1	Management Data Input. Used to transfer information from the PHY to the management module.
mdo_o	Out	1	Management Data Output. Used to transmit information from the management module to the PHY.
mdio_en_o	Out	1	Management Data Out Enable. Asserted whenever mdo_o is valid. This can be used to implement a bi-directional signal for mdi_i and mdo_o.
txd_pos_o[7:0][1] txd_neg_o[3:0][1] txd_o[7:0][2]	Out	8	txd_pos_o[7:4] - Transmit Data Sent to the PHY Chip - High nibble. In 1G mode, these bits are used as the GMII txd[7:4] bits after they are pipelined outside the core through some I/O flip-flops clocked at 125 MHz (txmac_clk_i). These bits are not used in the 10/100 mode. txd_pos_o[3:0], txd_neg_o[3:0] - Transmit Data Sent to the PHY Chip - Low nibble. In both 1G mode and 10/100 mode, both the txd_pos[3:0] and txd_neg[3:0] bits are used to generate the G/MII txd[3:0] bits after they are muxed outside the core through some I/O DDR cells. *Note: in the 1G mode, txd_pos_o[3:0] and txd_neg_o[3:0] always has the same value, whereas in the 10/100 mode, the txd_pos_o[3:0] has the high nibble of the byte transmitted and txd_neg_o[3:0] has the low nibble. In the 1G mode, the txmac_clk_i rate is 125 MHz and in the 10/100 mode, the clock rate is 1.25 MHz and 12.5 MHz respectively.* txd_o[7:0] - Transmit Data Sent to the PHY Interface. These GMII Tx data outputs go to the SGMII PCS IP core (SGMII Easy Connect option) or to the 1G GMII PHY interface (Gigabit interface option).
tx_en_o	Out	1	Transmit Enable. Asserted by the TSEMAC IP core to indicate the txd_o bus contains valid frame.
tx_er_o	Out	1	Transmit Error. Asserted when the TSEMAC IP core generates a coding error on the byte currently being transferred.
rx_dv_pos_i[1] rx_dv_neg_i[1] rx_dv_i[2]	In		Receive Data Valid. Indicates the data on the rxd_o bus is valid. rx_dv signalto is used in the SGMII Easy Connect MAC option and the Gigabit MAC option. rx_dv_pos_i and rx_dv_neg_i are used only in the Classic TSEMAC IP core interface option. This signal is pipelined before entering the core with the rxmac_clk_i. *Note: the "_pos" signals are sampled on the rising edge of the rxmac_clk_i and the "_neg" signals are sampled on the falling edge of the rxmac_clk_i, before being presented to the MAC core.*

Port Name	I/O	Width	Description
rxd_pos_i[7:0][1] rxd_neg_i[3:0][1] rxd_i[7:0][2]	In		Receive Data Bus. Data is driven by the PHY on these lines and is valid whenever rx_dv_pos/neg is asserted. These signals are pipelined before entering the core with the rxmac_clk_i. *Note: the "_pos" signals are sampled on the rising edge of the rxmac_clk_i and the "_neg" signals are sampled on the falling edge of the rxmac_clk_i, before being presented to the MAC core.* rxd_i[7:0] - Receive Data from the PHY Interface. These GMII Rx data inputs (valid whenever rx_dv_i is asserted) come from the SGMII_PCS IP core (SGMII Easy Connect option) or from the 1G GMII PHY interface (Gigabit MAC option).
rx_er_pos_i[1] rx_er_neg_i[1] rx_er_i[2]	In		Receive Data Error. This signal is asserted by the external PHY device when it detects an error during frame reception. This signal is pipelined before entering the core with the rxmac_clk_i. Note the "_pos" signals are sampled on the rising edge of the rxmac_clk_i and the "_neg" signals are sampled on the falling edge of the rxmac_clk_i, before being presented to the MAC core. rx_er - Receive Data Error from the PHY Interface.
col_i	In		Collision. This active-high signal indicates a collision occurred during transmission. This signal is valid for half-duplex operation in Fast Ethernet (10/100) for the Classic and SGMII Easy Connect options only. Otherwise, it is ignored.
crs_i	In		Carrier Sense. This signal, when logic high, indicates the network has activity. Otherwise, it indicates the network is idle. This signal is valid for half-duplex operation in Fast Ethernet (10/100) for the Classic and SGMII Easy Connect options only.
AHB-Lite Host Interface (Available when AHB-Lite is selected from User Interface)			
ahbl_hsel_i	In	1	AHB-Lite Select signal.
ahbl_hready_i	In	1	AHB-Lite Ready Input signal.
ahbl_haddr_i	In	8	AHB-Lite Address signal. Size: Interface Address Width
ahbl_hburst_i[2:0]	In	3	AHB-Lite Burst Type signal.
ahbl_hsize_i[2:0]	In	3	AHB-Lite Transfer Size signal.
ahbl_hmastlock_i	In	1	AHB-Lite Lock signal.
ahbl_hprot_i[3:0]	In	4	AHB-Lite Protection Control signal.
ahbl_htrans_i[1:0]	In	2	AHB-Lite Transfer Type signal.
ahbl_hwrite_i	In	1	AHB-Lite Direction signal. Write = High, Read = Low.
ahbl_hwdata_i	In	8	AHB-Lite Write Data signal. Size: Interface Data Width
ahbl_hreadyout_o	Out	1	AHB-Lite Ready Output signal.
ahbl_hrdata_o	Out	8	AHB-Lite Read Data signal. Size: Interface Data Width
ahbl_hresp_o	Out	1	AHB-Lite Transfer Response signal.
APB Host Interface (Available when APB is selected from User Interface)			
apb_paddr_i	In	8	APB Address signal. Size: Interface Address Width
apb_psel_i	In	1	APB Select signal.
apb_penable_i	In	1	APB Enable signal.
apb_pwrite_i	In	1	APB Direction signal.
apb_pwdata_i	In	8	APB Write Data signal. Size: Interface Data Width
apb_pready_o	Out	1	APB Ready signal.
apb_prdata_o	Out	8	APB Read Data signal. Size: Interface Data Width
apb_pslverr_o	Out	1	APB Slave Error signal.
Miscellaneous			
txmac_clk_en_i	In		Tx Clock Enable. This input signal is a clock enable used only in the SGMII Easy Connect option. The SGMII_PCS IP core drives this signal. The clock enable is always high for 1G operation. For 100 Mbps operation the clock enable is asserted high once every ten (125MHz) clocks, and for 10 Mbps operation the clock enable is asserted high once every hundred (125MHz) clocks.

Port Name	I/O	Width	Description
rxmac_clk_en_i	In		Rx Clock Enable. This input signal is a clock enable used only in the SGMII Easy Connect option. The SGMII_PCS IP core drives this signal. The clock enable is always high for 1G operation. For 100 Mbps operation the clock enable is asserted high once every ten (125MHz) clocks, and for 10 Mbps operation the clock enable is asserted high once every hundred (125MHz) clocks.
cpu_if_gbit_en_o	Out		CPU Interface 1G Mode Enabled Indication. This signal, when high, is an indication from the CPU interface that the 1G mode is enabled. This signal reflects the state of bit 0 of the MAC mode register.

Notes:
1. Classic TSEMAC IP core option
2. Gigabit MAC or SGMII Easy Connect MAC options

2.3. Attribute Summary

The configurable attributes of the TSEMAC IP Core are shown in Table 2.3 and are described in Table 2.4. The attributes can be configured through the IP Catalog Module/IP wizard in the Lattice Radiant software.

Table 2.3. Attributes Table

Attribute	Selectable Values	Default	Dependency on Other Attributes
General			
Interface	APB, AHBL	AHBL	—
Select MAC Operating Option	Classic TSEMAC, Gigabit MAC, SGMII Easy Connect	Classic TSEMAC	—
Include MIIM Module	Selected / Not Selected	Not Selected	—

Table 2.4. Attributes Descriptions

Attribute	Description
General	
Interface	This attribute selects the type of Host Interface, either AHB-Lite or APB.
Select MAC Operating Option	This attribute configures the usage type of the TSEMAC IP Core. See the Configuration Options section.
Include MIIM Module	When selected, the Management Interface (MDIO) is included in the TSEMAC IP Core.

2.4. Register Description

This section provides detailed descriptions of TSEMAC IP Core registers. Note that registers that are not available to users are highlighted in gray.

The register address map, shown in Table 2.5, specifies the available IP Core registers.

Table 2.5. Register Address Map

Offset	Register Name	Description
00H - 01H	Mode register	Enables/Disables IP Core functions.
02H - 03H	Transmit and Receive Control register	This register can be overwritten only when Rx MAC and Tx MAC are disabled. This register controls various features of MAC.
04H - 05H	Maximum Packet Size register	This register can be overwritten only when MAC is disabled. All frames longer than the value (number of bytes) in this register is tagged as long frames.
08H - 09H	Inter-Packet Gap register	Time between packet transmission.
0AH - 0BH	TSEMAC IP Core Address register 0	Contains the Ethernet address of the port.
0CH - 0DH	TSEMAC IP Core Address register 1	
0EH - 0FH	TSEMAC IP Core Address register 2	
12H - 13H	Transmit and Receive Status register	This register reports events that have occurred while receiving or transmitting a packet.
14H - 15H	GMII Management Interface Control register	The GMII Management Access register controls the Management Interface Module. This register can be overwritten only when the interface is not busy. A write operation is ignored when the interface is busy.
16H - 17H	GMII Management Data register	The contents of this register is transmitted when a Write operation is to be performed. When a Read operation is performed, this register contains the value that was read from a PHY register.
32H - 33H	VLAN Tag Length/type register	The VLAN tag register has the VLAN TAG field of the most recent tagged frame that was received. This is a read only register.
22H - 23H	Multicast_table_0 register	Multicast Table. Eight tables that make a 64-bit hash.
24H - 25H	Multicast_table_1 register	
26H - 27H	Multicast_table_2 register	
28H - 29H	Multicast_table_3 register	
2AH - 2BH	Multicast_table_4 register	
2CH - 2DH	Multicast_table_5 register	
2EH - 2FH	Multicast_table_6 register	
30H - 31H	Multicast_table_7 register	
34H - 35H	Pause Opcode register	PAUSE Opcode.
4EH	Interrupt Enable register	The interrupt enable register controls whether interrupts in the int_status_r register causes the int_o signal to assert or not. It does not affect the contents of the int_status_r register.
4EF	Interrupt Status register	Reading int_status_r register returns a set of bits representing all interrupts currently pending in the IP Block. The status bits are independent of the enable bits; in other words, status bits may indicate pending interrupts, even though those interrupts are disabled in the int_enable_r register. The logic, which handles interrupts, must mask the contents of int_status_r and int_enable_r registers before determining which interrupts to service. int_o signal is asserted whenever both an interrupt status bit and the corresponding interrupt enable bit are set. The signal is generated by bitwise AND-ing int_status_r and int_enable_r and then does the reduction OR of the result. Writing int_status_r register clears pending interrupts for each bit set to '1', so the "write 1 to clear" is applied to clear pending interrupts.

The behavior of registers to write and read access is defined by its access type, which is defined in Table 2.6.

Table 2.6. Access Type Definition

Access Type	Behavior on Read Access	Behavior on Write Access
RO	Returns register value	Ignores write access
RW	Returns register value	Updates register value
RW1C	Returns register value	Clear the register on write of 1, write value 0 is ignored
RSVD	Returns 0	Ignores write access

2.4.1. Mode Register

Table 2.7. Mode Register

Bit Field	Name	Access	Width	Description
15:4	—	RSVD	12	—
3	Tx_en	RW	1	Transmit Enable. When this bit is set, the Tx MAC is enabled to transmit frames. When reset, the Tx MAC completes transmission of the packet currently being processed, then stops.
2	Rx_en	RW	1	Receive Enable. When this bit is set, the Rx MAC is enabled to receive frames. When reset, the Rx MAC completes reception of the packet currently being processed, then stops.
1	FC_en	RW	1	Flow-control Enable. When set, this bit enables the flow control functionality of the Tx MAC. This bit should be set to enable the Tx MAC to transmit a PAUSE frame via the tx_sndpausreq and tx_sndpaustim[15:0] MAC input ports.
0	Gbit_en	RW	1	Gigabit Enable. For the Classic Tri-Speed MAC option, in order to operate in GbE mode, this bit must be set high. For 10/100 mode, this bit must be set low. For the SGMII Easy Connect MAC option, this bit does not control anything (note the MAC operation speed is determined by the clock enables provided by the SGMII IP core). This bit echoes back what is written to it. For the Gigabit MAC option, this bit is a read-only bit and always reads back as logic high. This bit does not control anything in the core. Note, the state of this bit is useful for system use, since the cpu_if_gbit_en output signal, from the core, always reflects the state of this register bit.

2.4.2. Transmit and Receive Control Register

Table 2.8. Transmit and Receive Control Register

Bit Field	Name	Access	Width	Description
15:9	—	RSVD	7	—
8	Receive_short	RW	1	Receive Short Frames. When high, enables the Rx MAC to receive frames shorter than 64 bytes.
7	Receive_brdcst	RW	1	Receive Broadcast. When high, enables the Rx MAC to receive broadcast frames
6	Drop_control	RW	1	Drop control. When high, received pause control frames are dropped internal to the MAC and not transferred to the external Rx client FIFO.
5	Hden	RW	1	Half-duplex Enable (10/100 mode only). When high, configures the Tx MAC to operate in half-duplex mode.
4	Receive_mltcst	RW	1	Receive Multicast. When high, the multicast frames are received per the filtering rules for such frames. When low, no Multicast (except PAUSE) frames is received.
3	Receive_pause	RW	1	Receive PAUSE. When set, the Rx MAC indicates the Rx PAUSE frame reception to the Tx MAC and thereby cause the Tx MAC to pause sending data frames for the period specified within the Rx PAUSE frame. Note this indication is independent of the Drop_control bit setting.
2	Tx_dis_fcs	RW	1	Transmit Disable FCS. When set, the FCS field generation is disabled in the Tx MAC.
1	Discard_fcs	RW	1	Rx Discard FCS and Pad. When set, the FCS and any of the padding bytes of an IEEE 802.3 frame are stripped off the frame before it is transferred to the Rx FIFO. When low, the entire frame is transferred into the Rx FIFO. Note: Discarding padding bytes is only applicable to pure IEEE 802.3 frames (such as in backplane applications) and does not function on Ethernet frames (IP, UDP, ICMP, etc.) where the length field is now interpreted as a protocol type field.
0	Prms	RW	1	Promiscuous Mode. When asserted, all filtering schemes are abandoned and the Rx MAC receives frames with any address.

2.4.3. Maximum Packet Size Register

Table 2.9. Maximum Packet Size Register

Bit Field	Name	Access	Width	Description
15:0	Max_frame	RW	16	Maximum size of the packet than can be handled by the core.

2.4.4. IPG (Inter-Packet Gap) Register

Table 2.10. Inter-Packet Gap Register

Bit Field	Name	Access	Width	Description
15:5	—	RSVD	12	—
4:0	IPG	RW	5	Inter-packet gap value in units of byte time.

2.4.5. MAC Address Register {0,1,2}

Table 2.11. MAC Address Register

Bit Field	Name	Access	Width	Description
15:0	Mac_addr	RW	16	Ethernet address assigned to the port supported by the TSEMAC IP core. MAC Address Registers 0-2 contain the Ethernet address of the port. MAC Address Register [0] has the two bytes that are transmitted first and MAC Address Register [2] has the two bytes that are transmitted last. Bit[8] through Bit[15] are transmitted first while bit[0] through bit[7] are transmitted last. Note that the MAC address is stored in the registers in Hexadecimal form. For example, setting the MAC Address to AC-DE-48-00-00-80 would require writing 0xAC (octet 0) to address 0x0B (high byte of Mac_addr[15:0]), 0xDE (octet 1) to address 0x0A (Low byte of Mac_addr[15:0]), 0x48 (octet 2) to address 0x0D (high byte of Mac_addr[15:0]), 0x00 (octet 3) to address 0x0C (Low byte of Mac_addr[15:0]), 0x00 (octet 4) to address 0x0F (high byte of Mac_addr[15:0]), and 0x80 (octet 5) to address 0x0E (Low byte of Mac_addr[15:0]).

2.4.6. Transmit and Receive Status Register

Table 2.12. Transmit and Receive Status Register

Bit Field	Name	Access	Width	Description
15:11	—	RSVD	2	—
10	Rx_idle	RW	1	Receive MAC Idle. Receive MAC in idle condition used to reset configurations by CPU interface.
9	Tagged_frame	RW	1	Tagged Frame. Tagged frame received.
8	Brdcst_frame	RW	1	Broadcast Frame. Indicates that a Broadcast packet was received.
7	Multcst_frame	RW	1	Multicast Frame. Indicates that a Multicast packet was received.
6	IPG_shrink	RW	1	IPG Shrink. Received frame with shrunk IPG (IPG < 96 bit time).
5	Short_frame	RW	1	Short Packet. Indicates that a packet shorter than 64 bytes has been received.
4	Long_frame	RW	1	Too Long Packet. Indicates receipt of a packet longer than the maximum allowable packet size specified in the MAX_PKT_SIZE Register.
3	Error frame	RW	1	rx_er Asserted. Indicates the frame was received with the rx_er signal asserted.
2	CRC	RW	1	CRC Error. Indicates a packet was received with a CRC error.
1	Pause_frame	RW	1	PAUSE Frame. Indicates a PAUSE frame was received.
0	Tx_idle	RW	1	Transmit MAC Idle. Transmit MAC in idle condition, used to reset configurations by CPU interface.

2.4.7. VLAN Tag Register

Table 2.13. VLAN Tag Register

Bit Field	Name	Access	Width	Description
15:0	VLAN	RO	16	This field defines length/type of field of the VLAN tag when inserted into transmitted frames.

2.4.8. GMII Management Register Access Control Register

Table 2.14. GMII Management Register Access Control Register

Bit Field	Name	Access	Width	Description
15	—	RSVD	1	—
14	Cmd_fin	RW	1	Command Finished. When high, it means the interface has completed the intended operation. This bit is set to 0 when the interface is busy.
13	RW_phyreg	RW	1	Read/Write PHY Registers When '1' -> write operation When '0' -> read operation
12:8	Phy_add	RW	5	GMII PHY Address. The address of the accessed PHY Bit 12 is the most significant bit, and it is the first PHY address bit to be transmitted and received.
7:5	—	RSVD	3	—
4:0	Reg_add	RW	5	GMII Register Address. The address of the register accessed. Bit 4 is the most significant bit and is the first register address bit to be transmitted or received.

2.4.9. GMII Management Access Data Register

Table 2.15. GMII Management Access Data Register

Bit Field	Name	Access	Width	Description
15:0	GMII_dat	RW	16	GMII Data. Bit 15 is the most significant bit corresponding to bit 15 of the accessed register.

2.4.10. Multicast Table Registers (0-7)

Table 2.16. Multicast Table Register

Bit Field	Name	Access	Width	Description
15:8	—	RSVD	8	—
7:0	Multicast_table	RW	8	When the core is programmed to receive multicast frames, a filtering scheme is used to decide whether the frame should be received or not. The six middle bits of the most significant byte of the CRC value, calculated for the destination address, are used as a key to the 64-bit hash table. The three most significant bits select one of the eight tables, and the three least significant bits select a bit. The frame is received only if this bit is set.

2.4.11. Pause OpCode Register

Table 2.17. Pause OpCode Register

Bit Field	Name	Access	Width	Description
15:0	Pause_OpCode	RW	16	This register contains the PAUSE Opcode, which is compared against the Opcode in the received PAUSE frame. This value is also be included in any PAUSE frame transmitted by TSEMAC IP. Bit 15 is transmitted first and bit 0 is transmitted last.

2.4.12. Interrupt Status Register

Table 2.18. Interrupt Status Register

Bit Field	Name	Access	Width	Description
15:8	—	RSVD	8	—
7	tx_overflow_int	RW1C	1	tx_overflow_int indicates that a write request (tx_wren) during the prior clock cycle was rejected, because the FIFO is full. Overflowing the FIFO is not destructive to the contents of the FIFO. 0 – No interrupt 1 – Interrupt pending Write 1 to clear.
6	tx_full_int	RW1C	1	tx_full_int, indicates that the Tx FIFO is full. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.
5	tx_afull_int	RW1C	1	tx_afull_int, indicates that the number of words in the Tx FIFO is greater than or equal to the threshold. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.
4	tx_aempty_int	RW1C	1	tx_aempty_int, indicates that the number of words in the FIFO is less than or equal to the threshold. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.
3	rx_underflow_int	RW1C	1	rx_underflow_int, indicates that a read request (rx_rden) during the previous clock cycle was rejected because the FIFO is empty. Underflowing the FIFO is not destructive to the FIFO. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.
2	rx_empty_int	RW1C	1	rx_empty_int, indicates that the FIFO is empty. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.
1	rx_aempty_int	RW1C	1	rx_aempty_int, indicates that the number of words in the FIFO is less than or equal to the threshold. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.
0	rx_afull_int	RW1C	1	rx_afull_int, indicates that the number of words in the Tx FIFO is greater than or equal to the threshold. 0 – No interrupt. 1 – Interrupt pending. Write 1 to clear.

2.4.13. Interrupt Enable Register

Table 2.19. Interrupt Enable Register

Bit Field	Name	Access	Width	Description
15:8	—	RSVD	8	—
7	tx_overflow_en	RW	1	tx_overflow_en, defines the interrupt enabled bit corresponding to Transmit Buffer Overflow Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
6	tx_full_en	RW	1	tx_full_en, defines the interrupt enabled bit corresponding to Transmit Buffer Full Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
5	tx_afull_en	RW	1	tx_afull_en, defines the interrupt enabled bit corresponding to Transmit Buffer Almost Full Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
4	tx_aempty_en	RW	1	tx_aempty_en, defines the Interrupt enabled of bit corresponding to Transmit Buffer Almost Empty Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
3	rx_underflow_en	RW	1	rx_underflow_en, defines the interrupt enabled bit corresponding to Receive Buffer Underflow Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
2	rx_empty_en	RW	1	rx_empty_en, defines the interrupt enabled bit corresponding to Receive Buffer Empty Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
1	rx_aempty_en	RW	1	rx_aempty_en, defines the interrupt enabled bit corresponding to Receive Buffer Almost Empty Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.
0	rx_afull_en	RW	1	rx_afull_en, defines the Interrupt enabled of bit corresponding to Receive Buffer Almost Full Interrupt source. 0 – Interrupt disabled. 1 – Interrupt enabled.

3. IP Generation and Evaluation

This section provides information on how to generate the TSEMAC IP Core using the Lattice Radiant Software and how to run synthesis and simulation. For more details on the Lattice Radiant Software, refer to the please refer to the Lattice Radiant Software 2.1 User Guide.

3.1. Licensing the IP

An IP core-specific license string is required to enable full use of the TSEMAC IP Core in a complete, top-level design.

The IP Core can be fully evaluated through functional simulation and implementation (synthesis, map, place and route) without an IP license string. This IP core supports Lattice's IP hardware evaluation capability, which makes it possible to create versions of the IP core, which operate in hardware for a limited time (approximately four hours) without requiring an IP license string. See the Hardware Evaluation section for further details. However, a license string is required to enable timing simulation and to generate bitstream file that does not include the hardware evaluation timeout limitation.

3.2. Generation and Synthesis

The Lattice Radiant Software allows you to customize and generate modules and IPs and integrate them into the device's architecture. The procedure for generating the TSEMAC IP Core in Lattice Radiant Software is described below.

To generate the TSEMAC IP Core:

1. Create a new Lattice Radiant Software project or open an existing project.
2. In the **IP Catalog** tab, double-click on **TSE_MAC** under **IP, Connectivity** category. The **Module/IP Block Wizard** opens as shown in Figure 3-1. Enter values in the **Instance name** and the **Create in** fields and click **Next**.

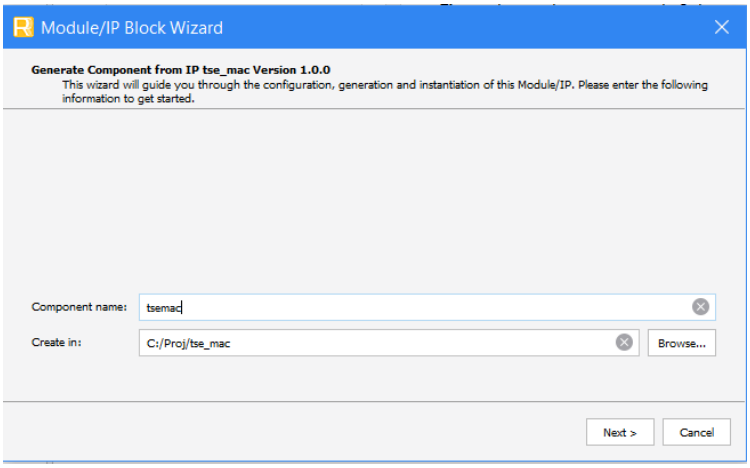

Figure 3-1. Module/IP Block Wizard

3. In the module's dialog box of the **Module/IP Block Wizard** window, customize the selected TSEMAC IP Core. As a sample configuration, see Figure 3-2. For configuration options, see the Attribute Summary section.

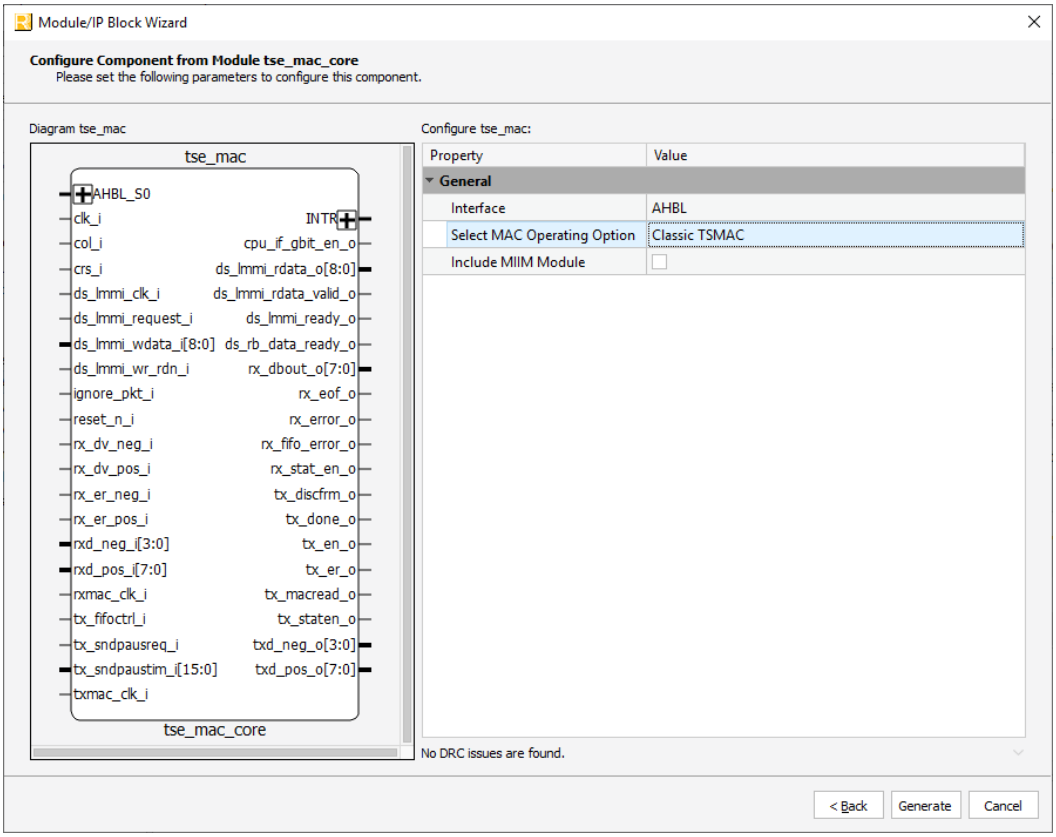

Figure 3-2. Configure User Interface of TSEMAC IP Core

4. Click **Generate**. The **Check Generating Result** dialog box opens, showing design block messages and results as shown in Figure 3-3.

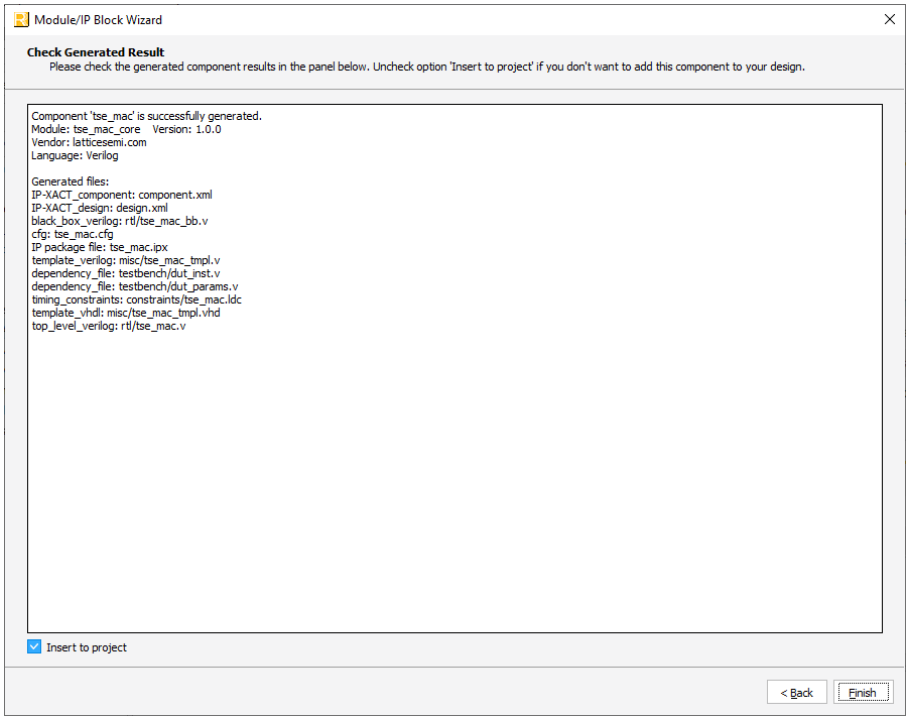

Figure 3-3. Check Generating Result

5. Click the **Finish** button. All the generated files are placed under the directory paths in the **Create in** and the **Instance name** fields shown in Figure 3-1.

The generated TSEMAC IP Core package includes the black box (<Instance Name>_bb.v) and instance templates (<Instance Name>_tmpl.v/vhd) that can be used to instantiate the core in a top-level design. An example RTL top-level reference source file (<Instance Name>.v) that can be used as an instantiation template for the IP core is also provided. You may also use this top-level reference as the starting template for the top-level for their complete design. The generated files are listed in Table 3.1.

Table 3.1. Generated File List

Attribute	Description
<Instance Name>.ipx	This file contains the information on the files associated to the generated IP.
<Instance Name>.cfg	This file contains the parameter values used in IP configuration.
component.xml	Contains the ipxact:component information of the IP.
design.xml	Documents the configuration parameters of the IP in IP-XACT 2014 format.
rtl/<Instance Name>.v	This file provides an example RTL top file that instantiates the IP core.
rtl/<Instance Name>_bb.v	This file provides the synthesis black box.
misc/<Instance Name>_tmpl.v misc /<Instance Name>_tmpl.vhd	These files provide instance templates for the IP core.

3.3. Running Functional Simulation

Running functional simulation can be performed after the IP is generated.

To run Verilog simulation:

1. Click the  button located on the **Toolbar** to initiate the **Simulation Wizard** shown in Figure 3-4.

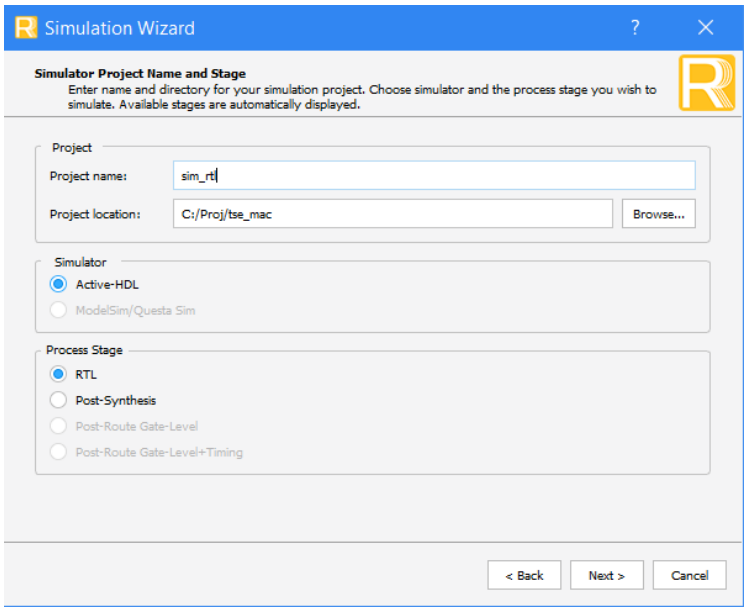

Figure 3-4. Simulation Wizard

2. Click **Next** to open the **Add and Reorder Source** window as shown in Figure 3-5.

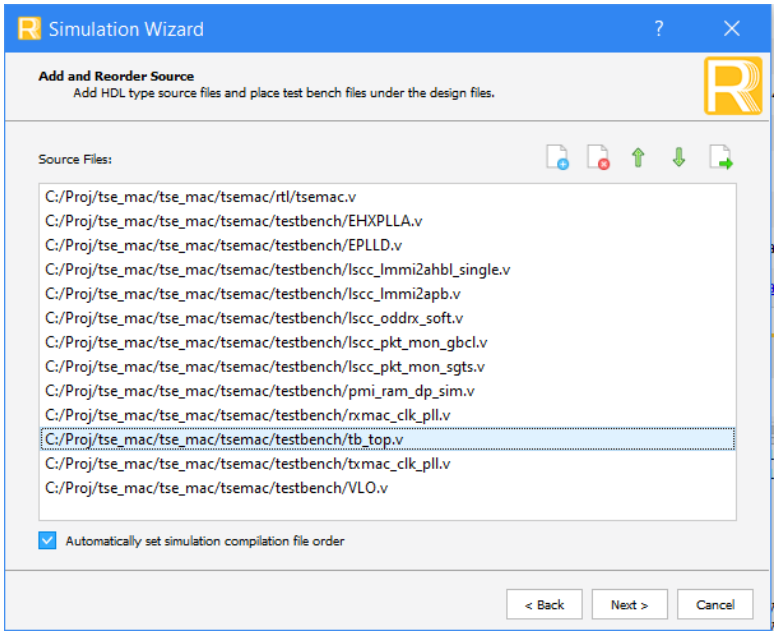

Figure 3-5. Adding and Reordering Source

3. Click **Next**. The **Summary** window is shown. Click **Finish** to run the simulation.

Note: It is necessary to follow the procedure above until it is fully automated in the Lattice Radiant Software Suite.

3.4. Hardware Evaluation

The TSEMAC IP Core supports Lattice's IP hardware evaluation capability when used with LIFCL devices. This makes it possible to create versions of the IP core that operate in hardware for a limited period of time (approximately four hours) without requiring the purchase of an IP license. It may also be used to evaluate the core in hardware in user-defined designs. The hardware evaluation capability may be enabled/disabled in the Strategy dialog box. It is enabled by default. To change this setting, go to Project > Active Strategy > LSE/Synplify Pro Settings.

4. Ordering Part Number

The Ordering Part Number (OPN) for this IP Core are the following:

- TS-MAC-CNX-U – Tri-Speed Ethernet MAC for CrossLink-NX - Single Design License
- TS-MAC-CNX-UT – Tri-Speed Ethernet MAC for CrossLink-NX - Site License
- TS-MAC-CTNX-U – Tri-Speed Ethernet MAC for Certus-NX - Single Design License
- TS-MAC-CTNX-UT – Tri-Speed Ethernet MAC for Certus-NX - Site License

References

For complete information on Lattice Radiant Project-Based Environment, Design Flow, Implementation Flow and Tasks, as well as on the Simulation Flow, see the Lattice Radiant Software 2.1 User Guide.

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Appendix A. Resource Utilization

Table A.1 shows the resource utilization for the LIFCL-40-9BG400I using Lattice Radiant Software.

For more information on Lattice Radiant Software, visit the Lattice web site at www.latticesemi.com/Products/DesignSoftwareAndIP.

Table A.1. Resource Utilization

Configuration	Slice Registers	LUTs	EBRs
Default	1576	2834	4

Revision History

Revision 1.2, June 2020

Section	Change Summary
Introduction	Updated Table 1.1: • Added Certus-NX support. • Added LFD2NX-40 as targeted device. • Updated Supported User Interfaces. • Updated Synopsis Synplify Pro version. • Updated Lattice Implementation to Lattice Radiant 2.1.
Functional Description	• Removed the block diagram of the TSEMAC IP Core. • Added Figure 2-1. Un-Tagged Ethernet Frame Format and Figure 2-2. VLAN-Tagged Ethernet Frame Format. • Updated Receive MAC (Rx MAC) information. • Updated Table 2.2. TSEMAC IP Core Signal Description.
Ordering Part Number	Added this section.
Appendix A. Resource Utilization	• Updated device to LIFCL-40-9BG400I. • Updated Table A.1. Resource Utilization.
All	Updated references to Lattice Radiant Software 2.1 User Guide.

Revision 1.1, February 2020

Section	Change Summary
Introduction	Updated Table 1.1 to add LIFCL-17 as targeted device.

Revision 1.0, November 2019

Section	Change Summary
All	Initial release.

www.latticesemi.com

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Development Software ***category:***

Click to view products by Lattice ***manufacturer:***

Other Similar products are found below :

RAPPID-560XBSW RAPPID-567XFSW DG-ACC-NET-CD SRP004001-01 SW006021-1NH SW163052 SYSWINEV21 Core429-SA SW500006-HPA CWP-BASIC-FL W128E13 CWP-PRO-FL AD-CCES-NODE-1 NT-ZJCAT1-EV4 CWA-BASIC-FL RAPPID-567XKSW CWA-STANDARD-R SW89CN0-ZCC CWA-LS-DVLPR-NL CWA-STANDARD-FL VDSP-21XX-PCFLOAT RAPPID-563XMSW IPS-EMBEDDED SWR-DRD-L-01 SDAWIR-4532-01 MPROG-PRO535E AFLCF-08-LX-CE060-R21 WS02-CFSC1-EV3-UP SYSMAC-STUDIO-EIPCPLR LIB-PL-PC-N-1YR-DISKID LS1043A-SWSP-PRM SW006026-COV 1120270005 1120270006 MIKROBASIC PRO FOR FT90X (USB DONGLE) MIKROC PRO FOR AVR (USB DONGLE LICENSE) MIKROC PRO FOR FT90X (USB DONGLE) MIKROBASIC PRO FOR AVR (USB DONGLE LICEN MIKROBASIC PRO FOR FT90X MIKROC PRO FOR DSPIC30/33 (USB DONGLE LI MIKROC PRO FOR FT90X MIKROC PRO FOR PIC32 (USB DONGLE LICENSE 52202-588 MIKROPASCAL PRO FOR ARM (USB DONGLE LICE MIKROPASCAL PRO FOR FT90X MIKROPASCAL PRO FOR FT90X (USB DONGLE) MIKROPASCAL PRO FOR PIC32 (USB DONGLE LI SW006021-2H SW006023-3 CWP-STANDARD-FL