LAN9210

Small Form Factor Single-Chip Ethernet Controller with HP Auto-MDIX Support

PRODUCT FEATURES

Datasheet

Highlights

- Optimized for standard performance applications
- Efficient architecture with low CPU overhead
- Easily interfaces to most 16-bit embedded CPU's
- Integrated PHY with HP Auto-MDIX support
- Integrated checksum offload engine helps reduce CPU load
- Low pin count and small body size package for small form factor system designs
- Supports audio & video streaming over Ethernet: multiple standard-definition (SD) MPEG2 streams

Target Applications

- Basic cable, satellite, and IP set-top boxes
- Digital video recorders
- Video-over IP solutions, IP PBX & video phones
- Wireless routers & access points
- Audio distribution systems
- Printers, kiosks, security systems
- General embedded applications

Key Benefits

- Non-PCI Ethernet controller for performance sensitive applications
  - 16-bit interface
  - Burst-mode read support
- Minimizes dropped packets
  - Internal buffer memory can store over 200 packets
  - Automatic PAUSE and back-pressure flow control
- Minimizes CPU overhead
  - Supports Slave-DMA
  - Interrupt Pin with Programmable Hold-off timer
- Reduces system cost and increases design flexibility
- SRAM-like interface easily interfaces to most embedded CPU's or SoC's
- Reduced Power Modes
  - Numerous power management modes
  - Wake on LAN
  - Magic packet wakeup
  - Wakeup indicator event signal
  - Link Status Change
- Single chip Ethernet controller
  - Fully compliant with IEEE 802.3/802.3u standards
  - Integrated Ethernet MAC and PHY
  - 10BASE-T and 100BASE-TX support
  - Full- and Half-duplex support
  - Full-duplex flow control
  - Backpressure for half-duplex flow control
  - Preamble generation and removal
  - Automatic 32-bit CRC generation and checking
  - Automatic payload padding and pad removal
  - Loop-back modes
- Flexible address filtering modes
  - One 48-bit perfect address
  - 64 hash-filtered multicast addresses
  - Pass all multicast
  - Promiscuous mode
  - Inverse filtering
  - Pass all incoming with status report
  - Disable reception of broadcast packets
- Integrated 10/100 Ethernet PHY
  - Supports HP Auto-MDIX
  - Auto-negotiation
  - Supports energy-detect power down
- Host bus interface
  - Simple, SRAM-like interface
  - 16-bit data bus
  - 16Kbyte FIFO with flexible TX/RX allocation
  - One configurable host interrupt
- Miscellaneous features
  - Small form factor, 56-pin QFN lead-free RoHS Compliant package
  - Integrated 1.8V regulator
  - Integrated checksum offload engine
  - Mixed endian support
  - General Purpose Timer
  - Optional EEPROM interface
  - Support for 3 status LEDs multiplexed with Programmable GPIO signals
- Single 3.3V Power Supply with 5V tolerant I/O
- 0°C to +70°C Commercial Temperature Support

Order Number(s):

LAN9210-ABZJ for 56-pin, QFN Lead-free RoHS Compliant package (0 to +70°C Temp Range)

This product meets the halogen maximum concentration values per IEC61249-2-21

For RoHS compliance and environmental information, please visit www.smsc.com/rohs

Copyright © 2012 SMSC or its subsidiaries. All rights reserved.

Circuit diagrams and other information relating to SMSC products are included as a means of illustrating typical applications. Consequently, complete information sufficient for construction purposes is not necessarily given. Although the information has been checked and is believed to be accurate, no responsibility is assumed for inaccuracies. SMSC reserves the right to make changes to specifications and product descriptions at any time without notice. Contact your local SMSC sales office to obtain the latest specifications before placing your product order. The provision of this information does not convey to the purchaser of the described semiconductor devices any licenses under any patent rights or other intellectual property rights of SMSC or others. All sales are expressly conditional on your agreement to the terms and conditions of the most recently dated version of SMSC's standard Terms of Sale Agreement dated before the date of your order (the "Terms of Sale Agreement"). The product may contain design defects or errors known as anomalies which may cause the product's functions to deviate from published specifications. Anomaly sheets are available upon request. SMSC products are not designed, intended, authorized or warranted for use in any life support or other application where product failure could cause or contribute to personal injury or severe property damage. Any and all such uses without prior written approval of an Officer of SMSC and further testing and/or modification will be fully at the risk of the customer. Copies of this document or other SMSC literature, as well as the Terms of Sale Agreement, may be obtained by visiting SMSC's website at http://www.smsc.com. SMSC is a registered trademark of Standard Microsystems Corporation ("SMSC"). Product names and company names are the trademarks of their respective holders.

SMSC DISCLAIMS AND EXCLUDES ANY AND ALL WARRANTIES, INCLUDING WITHOUT LIMITATION ANY AND ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND AGAINST INFRINGEMENT AND THE LIKE, AND ANY AND ALL WARRANTIES ARISING FROM ANY COURSE OF DEALING OR USAGE OF TRADE. IN NO EVENT SHALL SMSC BE LIABLE FOR ANY DIRECT, INCIDENTAL, INDIRECT, SPECIAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES; OR FOR LOST DATA, PROFITS, SAVINGS OR REVENUES OF ANY KIND; REGARDLESS OF THE FORM OF ACTION, WHETHER BASED ON CONTRACT; TORT; NEGLIGENCE OF SMSC OR OTHERS; STRICT LIABILITY; BREACH OF WARRANTY; OR OTHERWISE; WHETHER OR NOT ANY REMEDY OF BUYER IS HELD TO HAVE FAILED OF ITS ESSENTIAL PURPOSE, AND WHETHER OR NOT SMSC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

Chapter 1 General Description 9
1.1 Block Diagram 10
1.2 Internal Block Overview 11
1.3 10/100 Ethernet PHY 11
1.4 10/100 Ethernet MAC 11
1.5 Receive and Transmit FIFOs 12
1.6 Interrupt Controller 12
1.7 GPIO Interface 12
1.8 Serial EEPROM Interface 12
1.9 Power Management Controls 12
1.10 General Purpose Timer 12
1.11 Host Bus Interface (SRAM Interface) 13

Chapter 2 Pin Description and Configuration 14
2.1 Pin List 15
2.2 Buffer Types 19

Chapter 3 Functional Description 21
3.1 10/100 Ethernet MAC 21
3.2 Flow Control 22
3.2.1 Full-Duplex Flow Control 22
3.2.2 Half-Duplex Flow Control (Backpressure) 22
3.2.3 Virtual Local Area Network (VLAN) Support 22
3.3 Address Filtering Functional Description 23
3.4 Filtering Modes 24
3.4.1 Perfect Filtering 24
3.4.2 Hash Only Filtering 24
3.5 Wake-up Frame Detection 25
3.5.1 Magic Packet Detection 27
3.6 Checksum Offload Engines (COE) 28
3.6.1 Receive Checksum Offload Engine (RXCOE) 28
3.6.2 Transmit Checksum Offload Engine (TXCOE) 31
3.7 Host Bus Operations 32
3.7.1 Bus Writes 32
3.7.2 Bus Reads 33
3.7.3 Mixed Endian Support 33
3.7.4 Word Swap Function 33
3.8 General Purpose Timer (GP Timer) 36
3.9 EEPROM Interface 37
3.9.1 MAC Address Auto-Load 37
3.9.2 EEPROM Host Operations 37
3.10 Power Management 42
3.10.1 System Description 42
3.10.2 Functional Description 42
3.10.3 Internal PHY Power-Down Modes 45
3.11 Detailed Reset Description 46
3.11.1 Hardware Reset Input (nRESET) 47
3.11.2 Resume Reset Timing 47
3.11.3 Soft Reset (SRST) 47
3.11.4 PHY Reset Timing 47
3.12 TX Data Path Operation 47

3.12.1 TX Buffer Format . . . 49
3.12.2 TX Command Format . . . 50
3.12.3 TX Data Format . . . 52
3.12.4 TX Status Format . . . 53
3.12.5 Calculating Actual TX Data FIFO Usage . . . 54
3.12.6 Transmit Examples . . . 55
3.12.7 Transmitter Errors . . . 60
3.12.8 Stopping and Starting the Transmitter . . . 60
3.13 RX Data Path Operation . . . 60
3.13.1 RX Slave PIO Operation . . . 61
3.13.2 RX Packet Format . . . 63
3.13.3 RX Status Format . . . 64
3.13.4 Stopping and Starting the Receiver . . . 65
3.13.5 Receiver Errors . . . 65

Chapter 4 Internal Ethernet PHY . . . 66
4.1 Top Level Functional Description . . . 66
4.2 100Base-TX Transmit . . . 66
4.2.1 4B/5B Encoding . . . 66
4.2.2 Scrambling . . . 68
4.2.3 NRZI and MLT3 Encoding . . . 68
4.2.4 100M Transmit Driver . . . 68
4.2.5 100M Phase Lock Loop (PLL) . . . 68
4.3 100Base-TX Receive . . . 69
4.3.1 100M Receive Input . . . 69
4.3.2 Equalizer, Baseline Wander Correction and Clock and Data Recovery . . . 69
4.3.3 NRZI and MLT-3 Decoding . . . 69
4.3.4 Descrambling . . . 70
4.3.5 Alignment . . . 70
4.3.6 5B/4B Decoding . . . 70
4.4 10Base-T Transmit . . . 70
4.4.1 10M Transmit Data across the internal MII bus . . . 70
4.4.2 Manchester Encoding . . . 70
4.4.3 10M Transmit Drivers . . . 71
4.5 10Base-T Receive . . . 71
4.5.1 10M Receive Input and Squelch . . . 71
4.5.2 Manchester Decoding . . . 71
4.5.3 Jabber Detection . . . 71
4.6 Auto-negotiation . . . 71
4.7 Parallel Detection . . . 73
4.7.1 Re-starting Auto-negotiation . . . 73
4.7.2 Disabling Auto-negotiation . . . 73
4.7.3 Half vs. Full-Duplex . . . 73
4.8 HP Auto-MDIX . . . 74

Chapter 5 Register Description . . . 76
5.1 Register Nomenclature and Access Attributes . . . 77
5.2 RX and TX FIFO Ports . . . 77
5.2.1 RX FIFO Ports . . . 77
5.2.2 TX FIFO Ports . . . 77
5.3 System Control and Status Registers . . . 78
5.3.1 ID_REV—Chip ID and Revision . . . 79
5.3.2 IRQ_CFG—Interrupt Configuration Register . . . 79
5.3.3 INT_STS—Interrupt Status Register . . . 81

5.3.4 INT_EN—Interrupt Enable Register . . . 83
5.3.5 BYTE_TEST—Byte Order Test Register . . . 84
5.3.6 FIFO_INT—FIFO Level Interrupts . . . 84
5.3.7 RX_CFG—Receive Configuration Register . . . 85
5.3.8 TX_CFG—Transmit Configuration Register . . . 86
5.3.9 HW_CFG—Hardware Configuration Register . . . 87
5.3.10 RX_DP_CTRL—Receive Datapath Control Register . . . 90
5.3.11 RX_FIFO_INF—Receive FIFO Information Register . . . 91
5.3.12 TX_FIFO_INF—Transmit FIFO Information Register . . . 91
5.3.13 PMT_CTRL— Power Management Control Register . . . 92
5.3.14 GPIO_CFG—General Purpose IO Configuration Register . . . 94
5.3.15 GPT_CFG-General Purpose Timer Configuration Register . . . 95
5.3.16 GPT_CNT-General Purpose Timer Current Count Register . . . 96
5.3.17 WORD_SWAP—Word Swap Control . . . 96
5.3.18 FREE_RUN—Free-Run 25MHz Counter . . . 97
5.3.19 RX_DROP– Receiver Dropped Frames Counter . . . 97
5.3.20 MAC_CSR_CMD – MAC CSR Synchronizer Command Register . . . 98
5.3.21 MAC_CSR_DATA – MAC CSR Synchronizer Data Register . . . 98
5.3.22 AFC_CFG – Automatic Flow Control Configuration Register . . . 99
5.3.23 E2P_CMD – EEPROM Command Register . . . 101
5.3.24 E2P_DATA – EEPROM Data Register . . . 103
5.4 MAC Control and Status Registers . . . 104
5.4.1 MAC_CR—MAC Control Register . . . 105
5.4.2 ADDRH—MAC Address High Register . . . 108
5.4.3 ADDRL—MAC Address Low Register . . . 109
5.4.4 HASHH—Multicast Hash Table High Register . . . 110
5.4.5 HASHL—Multicast Hash Table Low Register . . . 110
5.4.6 MII_ACC—MII Access Register . . . 111
5.4.7 MII_DATA—MII Data Register . . . 111
5.4.8 FLOW—Flow Control Register . . . 112
5.4.9 VLAN1—VLAN1 Tag Register . . . 113
5.4.10 VLAN2—VLAN2 Tag Register . . . 113
5.4.11 WUFF—Wake-up Frame Filter . . . 114
5.4.12 WUCSR—Wake-up Control and Status Register . . . 114
5.4.13 COE_CR—Checksum Offload Engine Control Register . . . 115
5.5 PHY Registers . . . 116
5.5.1 Basic Control Register . . . 117
5.5.2 Basic Status Register . . . 118
5.5.3 PHY Identifier 1 . . . 118
5.5.4 PHY Identifier 2 . . . 119
5.5.5 Auto-negotiation Advertisement . . . 119
5.5.6 Auto-negotiation Link Partner Ability . . . 120
5.5.7 Auto-negotiation Expansion . . . 121
5.5.8 Mode Control/Status . . . 121
5.5.9 Special Modes . . . 122
5.5.10 Special Control/Status Indications . . . 123
5.5.11 Interrupt Source Flag . . . 124
5.5.12 Interrupt Mask . . . 124
5.5.13 PHY Special Control/Status . . . 125

Chapter 6 Timing Diagrams . . . 126
6.1 Host Interface Timing . . . 126
6.1.1 Special Restrictions on Back-to-Back Write/Read Cycles . . . 126
6.1.2 Special Restrictions on Back-to-Back Read Cycles . . . 128

6.2 PIO Reads . . . 129
6.3 PIO Burst Reads . . . 130
6.4 RX Data FIFO Direct PIO Reads . . . 131
6.5 RX Data FIFO Direct PIO Burst Reads . . . 132
6.6 PIO Writes . . . 133
6.7 TX Data FIFO Direct PIO Writes . . . 134
6.8 Reset Timing . . . 135
6.9 EEPROM Timing . . . 136

Chapter 7 Operational Characteristics . . . 137
7.1 Absolute Maximum Ratings* . . . 137
7.2 Operating Conditions** . . . 137
7.3 Power Consumption (Device Only) . . . 138
7.4 Power Consumption (Device and System Components) . . . 139
7.5 Worst Case Current Consumption . . . 140
7.6 DC Electrical Specifications . . . 141
7.7 Clock Circuit . . . 143

Chapter 8 Package Outline . . . 144
8.1 56-QFN Package . . . 144

Chapter 9 Datasheet Revision History . . . 146

List of Figures

Figure 1.1 System Block Diagram . . . 10
Figure 1.2 Internal Block Diagram . . . 11
Figure 2.1 Pin Configuration (Top View) . . . 14
Figure 3.1 VLAN Frame . . . 23
Figure 3.2 RXCOE Checksum Calculation . . . 28
Figure 3.3 Type II Ethernet Frame . . . 29
Figure 3.4 Ethernet Frame with VLAN Tag . . . 29
Figure 3.5 Ethernet Frame with Length Field and SNAP Header . . . 29
Figure 3.6 Ethernet Frame with VLAN Tag and SNAP Header . . . 30
Figure 3.7 Ethernet Frame with Multiple VLAN Tags and SNAP Header . . . 30
Figure 3.8 LAN9210 Host Data Path Diagram . . . 34
Figure 3.9 FIFO Access Byte Ordering . . . 35
Figure 3.10 EEPROM Access Flow Diagram . . . 38
Figure 3.11 EEPROM ERASE Cycle . . . 39
Figure 3.12 EEPROM ERAL Cycle . . . 39
Figure 3.13 EEPROM EWDS Cycle . . . 40
Figure 3.14 EEPROM EWEN Cycle . . . 40
Figure 3.15 EEPROM READ Cycle . . . 41
Figure 3.16 EEPROM WRITE Cycle . . . 41
Figure 3.17 EEPROM WRAL Cycle . . . 41
Figure 3.18 PME and PME_INT Signal Generation . . . 45
Figure 3.19 Simplified Host TX Flow Diagram . . . 49
Figure 3.20 TX Buffer Format . . . 50
Figure 3.21 TX Example 1 . . . 56
Figure 3.22 TX Example 2 . . . 57
Figure 3.23 TX Example 3 . . . 59
Figure 3.24 Host Receive Routine Using Interrupts . . . 61
Figure 3.25 Host Receive Routine with Polling . . . 61
Figure 3.26 RX Packet Format . . . 63
Figure 3.27 RX Packet Format with RX Checksum . . . 64
Figure 4.1 100Base-TX Data Path . . . 66
Figure 4.2 Receive Data Path . . . 69
Figure 4.3 Direct Cable Connection vs. Cross-over Cable Connection . . . 75
Figure 5.1 Memory Map . . . 76
Figure 5.2 Example ADDRL, ADDRH and EEPROM Setup . . . 109
Figure 6.1 PIO Read Cycle Timing . . . 129
Figure 6.2 PIO Burst Read Cycle Timing . . . 130
Figure 6.3 RX Data FIFO Direct PIO Read Cycle Timing . . . 131
Figure 6.4 RX Data FIFO Direct PIO Burst Read Cycle Timing . . . 132
Figure 6.5 PIO Write Cycle Timing . . . 133
Figure 6.6 TX Data FIFO Direct PIO Write Timing . . . 134
Figure 6.7 Reset Timing . . . 135
Figure 6.8 EEPROM Timing . . . 136
Figure 8.1 56 Pin QFN Package Definition . . . 144
Figure 8.2 56 Pin QFN Recommended PCB Land Pattern . . . 145

List of Tables

Table 2.1 Host Bus Interface Signals . . . 15
Table 2.2 LAN Interface Signals . . . 15
Table 2.3 Serial EEPROM Interface Signals . . . 16
Table 2.4 System and Power Signals . . . 17
Table 2.5 56-QFN Package Pin Assignments . . . 19
Table 2.6 Buffer Types . . . 19
Table 3.1 Address Filtering Modes . . . 23
Table 3.2 Wake-Up Frame Filter Register Structure . . . 26
Table 3.3 Filter i Byte Mask Bit Definitions . . . 26
Table 3.4 Filter i Command Bit Definitions . . . 26
Table 3.5 Filter i Offset Bit Definitions . . . 27
Table 3.6 Filter i CRC-16 Bit Definitions . . . 27
Table 3.7 TX Checksum Preamble . . . 32
Table 3.8 Endian Ordering Logic Operation . . . 36
Table 3.9 Required EECLK Cycles . . . 42
Table 3.10 Power Management States . . . 44
Table 3.11 Reset Sources and Affected Circuitry . . . 46
Table 3.12 TX Command 'A' Format . . . 51
Table 3.13 TX Command 'B' Format . . . 52
Table 3.14 TX DATA Start Offset . . . 52
Table 4.1 4B/5B Code Table . . . 67
Table 4.2 CRS Behavior . . . 74
Table 5.1 Direct Address Register Map . . . 78
Table 5.2 RX Alignment Bit Definitions . . . 85
Table 5.3 Valid TX/RX FIFO Allocations . . . 89
Table 5.4 EEPROM Enable Bit Definitions . . . 95
Table 5.5 Backpressure Duration Bit Mapping . . . 100
Table 5.6 MAC CSR Register Map . . . 104
Table 5.7 ADDRL, ADDRH and EEPROM Byte Ordering . . . 109
Table 5.8 LAN9210 PHY Control and Status Register . . . 116
Table 5.9 MODE Control . . . 122
Table 6.1 Read After Write Timing Rules . . . 127
Table 6.2 Read After Read Timing Rules . . . 128
Table 6.3 PIO Read Timing . . . 129
Table 6.4 PIO Burst Read Timing . . . 130
Table 6.5 RX Data FIFO Direct PIO Read Timing . . . 131
Table 6.6 RX Data FIFO Direct PIO Burst Read Cycle Timing . . . 132
Table 6.7 PIO Write Cycle Timing . . . 133
Table 6.8 TX Data FIFO Direct PIO Write Timing . . . 134
Table 6.9 Reset Timing . . . 135
Table 6.10 EEPROM Timing Values . . . 136
Table 7.1 Power Consumption Device Only . . . 138
Table 7.2 Power Consumption Device and System Components . . . 139
Table 7.3 Maximum Supply Current Characteristics . . . 140
Table 7.4 I/O Buffer Characteristics . . . 141
Table 7.5 100BASE-TX Transceiver Characteristics . . . 142
Table 7.6 10BASE-T Transceiver Characteristics . . . 142
Table 7.7 LAN9210 Crystal Specifications . . . 143
Table 8.1 56 Pin QFN Package Parameters . . . 144
Table 9.1 Customer Revision History . . . 146

Chapter 1 General Description

The LAN9210 is a full-featured, single-chip 10/100 Ethernet controller designed for embedded applications where performance, flexibility, ease of integration and system cost control are required. The LAN9210 is fully IEEE 802.3 10BASE-T and 802.3u 100BASE-TX compliant, and supports HP Auto-MDIX.

The LAN9210 includes an integrated Ethernet MAC and PHY with a high-performance SRAM-like slave interface. The simple, yet highly functional host bus interface provides a glue-less connection to most common 16-bit microprocessors and microcontrollers as well as 32-bit microprocessors with a 16-bit external bus. The integrated checksum offload engines enable the automatic generation of the 16-bit checksum for received and transmitted Ethernet frames, offloading the task from the CPU. The LAN9210 also includes large transmit and receive data FIFOs to accommodate high latency applications. In addition, the LAN9210 memory buffer architecture allows highly efficient use of memory resources by optimizing packet granularity.

Applications

The LAN9210 is well suited for many medium-performance embedded applications, including:

- Printers, kiosks, POS terminals and security systems
- Audio distribution systems
- General embedded systems
- Basic cable, satellite and IP set-top boxes
- Voice-over-IP solutions

The LAN9210 also supports features which reduce or eliminate packet loss. Its internal 16-KByte SRAM can hold over 200 received packets. If the receive FIFO gets too full, the LAN9210 can automatically generate flow control packets to the remote node, or assert back-pressure on the remote node by generating network collisions.

The LAN9210 supports numerous power management and wakeup features. The LAN9210 can be placed in a reduced power mode and can be programmed to issue an external wake signal via several methods, including “Magic Packet”, “Wake on LAN” and “Link Status Change”. This signal is ideal for triggering system power-up using remote Ethernet wakeup events. The device can be removed from the low power state via a host processor command.

1.1 Block Diagram

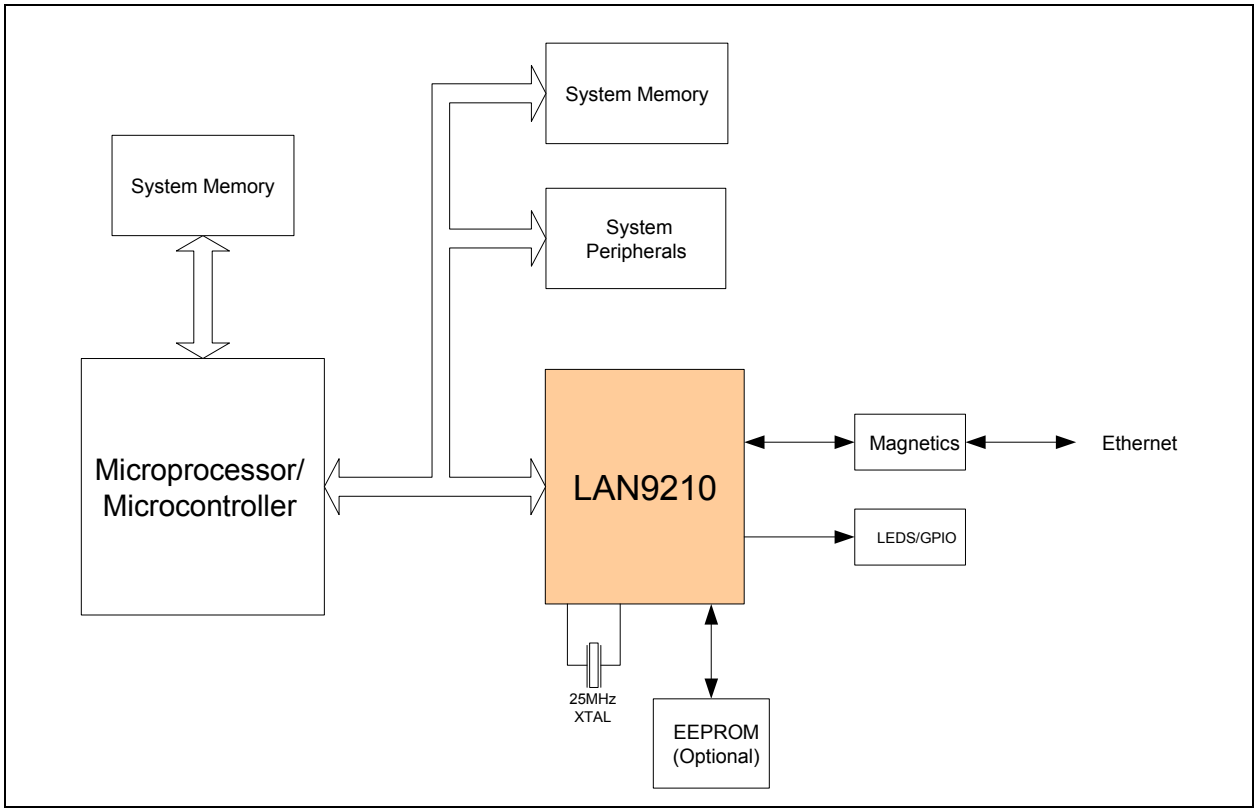


Figure 1.1 System Block Diagram

The SMSC LAN9210 integrated 10/100 MAC/PHY controller is a peripheral chip that performs the function of translating parallel data from a host controller into Ethernet packets. The LAN9210 Ethernet MAC/PHY controller is designed and optimized to function in an embedded environment. All communication is performed with programmed I/O transactions using the simple SRAM-like host interface bus.

The diagram shown above, describes a typical system configuration of the LAN9210 in a typical embedded environment.

The LAN9210 is a general purpose, platform independent, Ethernet controller. The LAN9210 consists of four major functional blocks. The four blocks are:

- 10/100 Ethernet PHY
- 10/100 Ethernet MAC
- RX/TX FIFOs
- Host Bus Interface (HBI)

1.2 Internal Block Overview

This section provides an overview of each of these functional blocks as shown in Figure 1.2, "Internal Block Diagram".

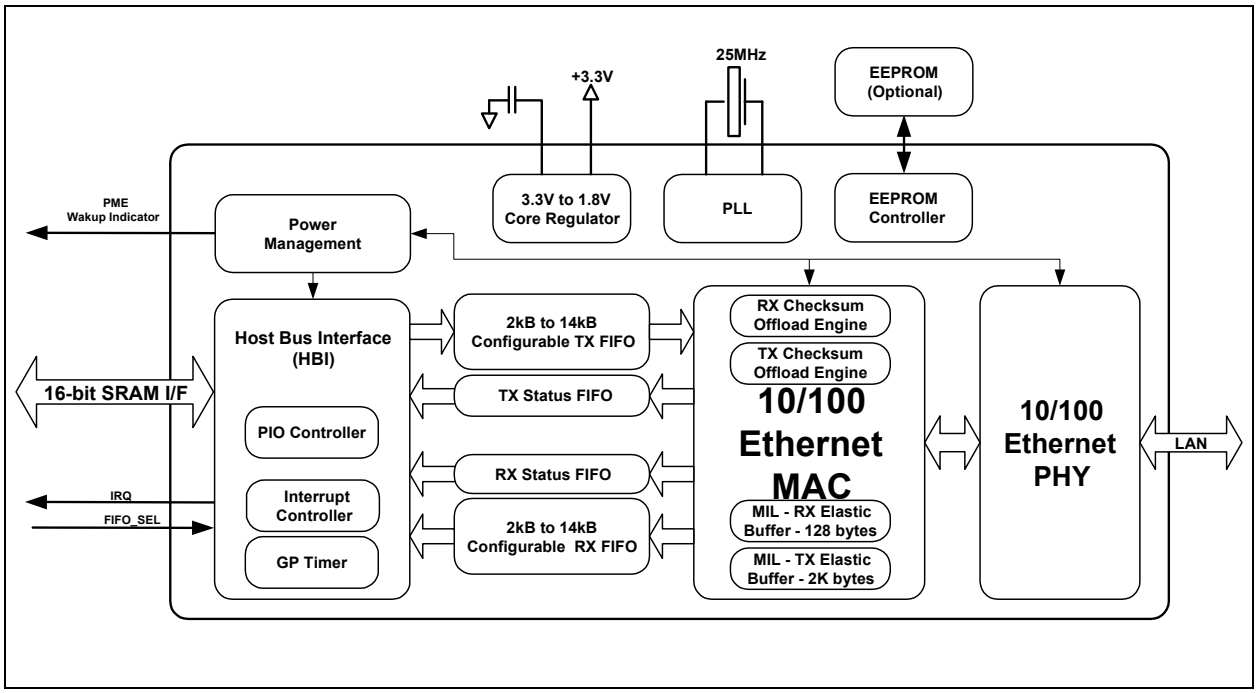


Figure 1.2 Internal Block Diagram

1.3 10/100 Ethernet PHY

The LAN9210 integrates an IEEE 802.3 physical layer for twisted pair Ethernet applications. The PHY can be configured for either 100 Mbps (100BASE-TX) or 10 Mbps (10BASE-T) Ethernet operation in either full or half duplex configurations. The PHY block supports HP Auto-MDIX and auto-negotiation.

Minimal external components are required for the utilization of the Integrated PHY.

1.4 10/100 Ethernet MAC

The transmit and receive data paths are separate within the MAC allowing the highest performance especially in full duplex mode. The data paths connect to the PIO interface Function via separate busses to increase performance. Payload data as well as transmit and receive status is passed on these busses.

A third internal bus is used to access the MAC's Control and Status Registers (CSR's). This bus is accessible from the host through the PIO interface function.

On the backend, the MAC interfaces with the internal 10/100 PHY through a MII (Media Independent Interface) port internal to the LAN9210. The MAC CSR's also provide a mechanism for accessing the PHY's internal registers through the internal SMI (Serial Management Interface) bus.

The MAC Interface Layer (MIL), within the MAC, contains a 2K Byte transmit and a 128 Byte receive FIFO which is separate from the TX and RX FIFOs. The FIFOs within the MAC are not directly accessible from the host interface. The differentiation between the TX/RX FIFO memory buffers and the MAC buffers is that when the transmit or receive packets are in the MAC buffers, the host no longer

can control or access the TX or RX data. The MAC buffers (both TX and RX) are in effect the working buffers of the Ethernet MAC logic. In the case of reception, the data must be moved first to the RX FIFOs for the host to access the data. For TX operations, the MIL operates in store-and-forward mode and will queue an entire frame before beginning transmission.

1.5 Receive and Transmit FIFOs

The Receive and Transmit FIFOs allow increased packet buffer storage to the MAC. The FIFOs are a conduit between the host interface and the MAC through which all transmitted and received data and status information is passed. Deep FIFOs allow a high degree of latency tolerance relative to the various transport and OS software stacks thus reducing or minimizing overrun conditions. Like the MAC, the FIFOs have separate receive and transmit data paths. In addition, the RX and TX FIFOs are configurable in size, allowing increased flexibility.

1.6 Interrupt Controller

The LAN9210 supports a single programmable interrupt. The programmable nature of this interrupt allows the user the ability to optimize performance dependent upon the application requirement. Both the polarity and buffer type of the interrupt pin are configurable for the external interrupt processing. The interrupt line can be configured as an open-drain output to facilitate the sharing of interrupts with other devices. In addition, a programmable interrupt de-assertion interval is provided.

1.7 GPIO Interface

A 3-bit GPIO and 2-bit GPO (Multiplexed on the EEPROM and LED Pins) interface is included in the LAN9210. It is accessible through the host bus interface via the CSRs. The GPIO signals can function as inputs, push-pull outputs and open drain outputs. The GPIO's (GPO's are not configurable) can also be configured to trigger interrupts with programmable polarity.

1.8 Serial EEPROM Interface

A serial EEPROM interface is included in the LAN9210. The serial EEPROM is optional and can be programmed with the LAN9210 MAC address. The LAN9210 can optionally load the MAC address automatically after hardware reset, or soft reset.

1.9 Power Management Controls

The LAN9210 supports comprehensive array of power management modes to allow use in power sensitive applications. Wake on LAN, Link Status Change and Magic Packet detection are supported by the LAN9210. An external PME (Power Management Event) interrupt is provided to indicate detection of a wakeup event.

1.10 General Purpose Timer

The general-purpose timer has no dedicated function within the LAN9210 and may be programmed to issue a timed interrupt.

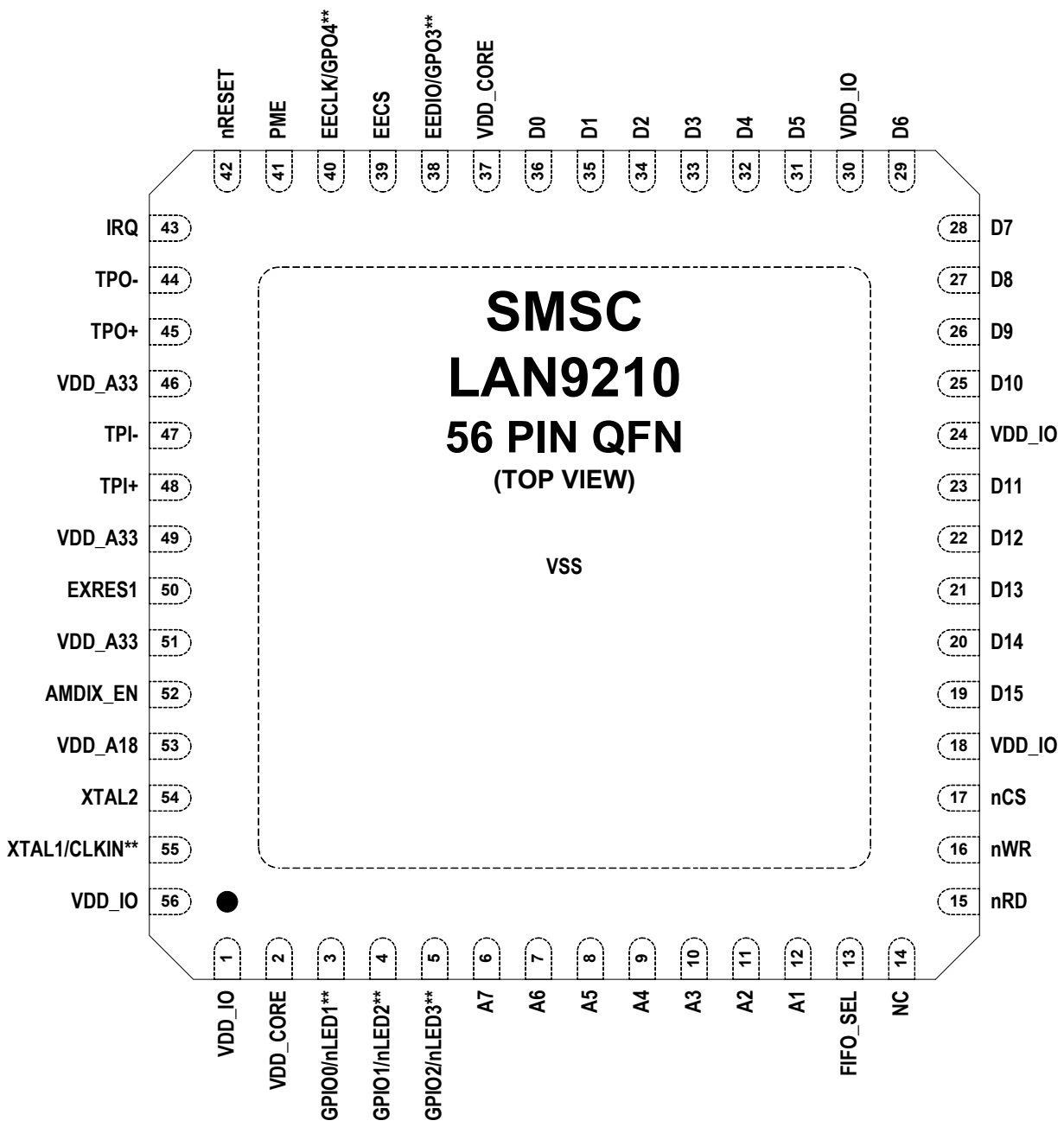
1.11 Host Bus Interface (SRAM Interface)

The host bus interface provides a FIFO interface for the transmit and receive data paths, as well as an interface for the LAN9210 Control and Status Registers (CSR's).

The host bus interface is the primary bus for connection to the embedded host system. This interface models an asynchronous SRAM. TX FIFO, RX FIFO, and CSR's are accessed through this interface. Programmed I/O transactions are supported.

The LAN9210 host bus interface supports 16-bit bus transfers. Internally, all data paths are 32-bits wide. The LAN9210 can be interfaced to either Big-Endian or Little-Endian processors and includes mixed endian support for FIFO accesses.

Chapter 2 Pin Description and Configuration



**DENOTES A MULTIFUNCTION PIN

NOTE: When HP Auto-MDIX is activated, the TPO+/- pins can function as TPI+/- and vice-versa

NOTE: Exposed pad (VSS) on bottom of package must be connected to ground

Figure 2.1 Pin Configuration (Top View)

2.1 Pin List

Table 2.1 Host Bus Interface Signals

NAME	SYMBOL	BUFFER TYPE	# PINS	DESCRIPTION
Host Data	D[15:0]	I/O8	16	Bi-directional data port.
Host Address	A[7:1]	IS	7	7-bit Address Port. Used to select Internal CSR's and TX and RX FIFOs.
Read Strobe	nRD	IS	1	Active low strobe to indicate a read cycle.
Write Strobe	nWR	IS	1	Active low strobe to indicate a write cycle. This signal, qualified with nCS, is also used to wakeup the LAN9210 when it is in a reduced power state.
Chip Select	nCS	IS	1	Active low signal used to qualify read and write operations. This signal qualified with nWR is also used to wakeup the LAN9210 when it is in a reduced power state.
Interrupt Request	IRQ	O8/OD8	1	Programmable Interrupt request. Programmable polarity, source and buffer types.
FIFO Select	FIFO_SEL	IS	1	When driven high all accesses to the LAN9210 are to the RX or TX Data FIFOs. In this mode, the A[7:3] upper address inputs are ignored.

Table 2.2 LAN Interface Signals

NAME	SYMBOL	BUFFER TYPE	NUM PINS	DESCRIPTION
TPO+	TPO+	AO	1	Transmit Positive Output (normal) Receive Positive Input (reversed)
TPO-	TPO-	AO	1	Transmit Negative Output (normal) Receive Negative Input (reversed)
TPI+	TPI+	AI	1	Receive Positive Input (normal) Transmit Positive Input (reversed)
TPI-	TPI-	AI	1	Receive Negative Input (normal) Transmit Negative Output (reversed)
PHY External Bias Resistor	EXRES1	AI	1	Must be connected to ground through a 12.4K ohm 1% resistor.

Note: The pin names for the twisted pair pins shown above apply to a normal connection. If HP Auto-MDIX is enabled and a reverse connection is detected, or a reverse connection is manually selected, the input pins become outputs, and vice-versa, as indicated in the descriptions.

Table 2.3 Serial EEPROM Interface Signals

NAME	SYMBOL	BUFFER TYPE	NUM PINS	DESCRIPTION
EEPROM Data, GPO3, TX_EN, TX_CLK	EEDIO/GPO3/ TX_EN/TX_CLK	I/O8	1	**EEPROM Data:** This bi-directional pin can be connected to a serial EEPROM DIO. This is optional. **General Purpose Output 3:** This pin can also function as a general purpose output, or it can be configured to monitor the TX_EN or TX_CLK signals on the internal MII port. When configured as a GPO signal, or as a TX_EN/TX_CLK monitor, the EECS pin is deasserted so as to never unintentionally access the serial EEPROM. This signal cannot function as a general-purpose input.
EEPROM Chip Select	EECS	O8	1	Serial EEPROM chip select.
EEPROM Clock, GPO4 RX_DV, RX_CLK	EECLK/GPO4/ RX_DV/RX_CLK	O8 (PU)	1	**EEPROM Clock:** Serial EEPROM Clock pin. **General Purpose Output 4:** This pin can also function as a general-purpose output, or it can be configured to monitor the RX_DV or RX_CLK signals on the internal MII port. When configured as a GPO signal, or as an RX_DV/RX_CLK monitor, the EECS pin is deasserted so as to never unintentionally access the serial EEPROM. This signal cannot function as a general-purpose input. **Note:** When the EEPROM interface is not used, the EECLK pin must be left unconnected. **Note:** This pin must not be pulled low by an external resistor or driven low externally under any conditions.

Table 2.4 System and Power Signals

NAME	SYMBOL	BUFFER TYPE	NUM PINS	DESCRIPTION
Crystal 1, Clock In	XTAL1/CLKIN	ICLK	1	External 25MHz Crystal Input. This pin can also be connected to single-ended TTL oscillator (CLKIN). If this method is implemented, XTAL2 should be left unconnected.
Crystal 2	XTAL2	OCLK	1	External 25MHz Crystal output.
Reset	nRESET	IS (PU)	1	Active-low reset input. Resets all logic and registers within the LAN9210. This signal is pulled high with a weak internal pull-up resistor. **Note:** The LAN9210 must be reset on power-up via nRESET or following power-up via a soft reset (SRST). The LAN9210 must always be read at least once after reset, or upon return from a power-saving state or write operations will not function. See Section 3.11, "Detailed Reset Description," on page 46 for additional information
Wakeup Indicator	PME	O8/OD8	1	When programmed to do so, is asserted when the LAN9210 detects a wake event and is requesting the system to wake up from the associated sleep state. The polarity and buffer type of this signal is programmable. **Note:** Detection of a Power Management Event, and assertion of the PME signal will not wakeup the LAN9210. The LAN9210will only wake up when it detects a host write cycle (assertion of nCS and nWR). Although any write to the LAN9210, regardless of the data written, will wake-up the device when it is in a power-saving mode, it is required that the BYTE_TEST register be used for this purpose.
Auto-MDIX Enable	AMDIX_EN	I (PU)	1	Enables Auto-MDIX. Pull high or leave unconnected to enable Auto-MDIX, pull low to disable Auto-MDIX.
No Connect	NC		1	No Connect. This pin must be left open.

Table 2.4 System and Power Signals (continued)

NAME	SYMBOL	BUFFER TYPE	NUM PINS	DESCRIPTION
General Purpose I/O data, nLED1 (Speed Indicator), nLED2 (Link & Activity Indicator), nLED3 (Full-Duplex Indicator).	GPIO[2:0]/ nLED[3:1]	IS/O12/ OD12	3	**General Purpose I/O data:** These three general-purpose signals are fully programmable as either push-pull output, open-drain output or input by writing the GPIO_CFG configuration register in the CSR's. They are also multiplexed as GP LED connections. GPIO signals are Schmitt-triggered inputs. When configured as LED outputs these signals are open-drain. **nLED1 (Speed Indicator).** This signal is driven low when the operating speed is 100Mb. During auto-negotiation, when the cable is disconnected, and during 10Mbs operation, this signal is driven high. **nLED2 (Link & Activity Indicator).** This signal is driven low (LED on) when the LAN9210 detects a valid link. This signal is pulsed high (LED off) for 80mS whenever transmit or receive activity is detected. This signal is then driven low again for a minimum of 80mS, after which time it will repeat the process if TX or RX activity is detected. Effectively, LED2 is activated solid for a link. When transmit or receive activity is sensed LED2 will flash as an activity indicator. **nLED3 (Full-Duplex Indicator).** This signal is driven low when the link is operating in full-duplex mode.
+3.3V I/O Power	VDD_IO	P	5	+3.3V I/O logic power supply pins
Common Ground	VSS	P	1 pad	Common Ground
+3.3V Analog Power	VDD_A33	P	3	+3.3V analog power supply pins. See Note 2.1.
+1.8V Analog Power	VDD_A18	P	1	+1.8V analog power supply pin. This pin must be connected externally to VDD_CORE. See Note 2.1.
Core Voltage Decoupling	VDD_CORE	P	2	+1.8 V from internal core regulator. Both pins must be connected together externally. Each pin requires a 0.01uF decoupling capacitor. In addition, pin 2 requires a bulk 4.7uF capacitor (<2 Ohm ESR) in parallel. These pins must not be used to supply power to other external devices. See Note 2.1.

Note 2.1 Please refer to the SMSC application note AN16.6 - "Migrating from LAN9215 to the LAN9210/LAN9211" for additional details.

Table 2.5 56-QFN Package Pin Assignments

PIN NUM	PIN NAME	PIN NUM	PIN NAME	PIN NUM	PIN NAME	PIN NUM	PIN NAME
1	VDD_IO	15	nRD	29	D6	43	IRQ
2	VDD_CORE	16	nWR	30	VDD_IO	44	TPO-
3	GPIO0/nLED1	17	nCS	31	D5	45	TPO+
4	GPIO1/nLED2	18	VDD_IO	32	D4	46	VDD_A33
5	GPIO2/nLED3	19	D15	33	D3	47	TPI-
6	A7	20	D14	34	D2	48	TPI+
7	A6	21	D13	35	D1	49	VDD_A33
8	A5	22	D12	36	D0	50	EXRES1
9	A4	23	D11	37	VDD_CORE	51	VDD_A33
10	A3	24	VDD_IO	38	EEDIO/GPO3	52	AMDIX_EN
11	A2	25	D10	39	EECS	53	VDD_A18
12	A1	26	D9	40	EECLK/GPO4	54	XTAL2
13	FIFO_SEL	27	D8	41	PME	55	XTAL1/CLKIN
14	NC	28	D7	42	nRESET	56	VDD_IO
EXPOSED PAD MUST BE CONNECTED TO VSS							

2.2 Buffer Types

Table 2.6 Buffer Types

TYPE	DESCRIPTION
I	Input pin
IS	Schmitt triggered Input
O12	Output with 12mA sink and 12mA source
OD12	Open-drain output with 12mA sink
OD8	Open-drain output with 8mA sink
O8	Output 8mA symmetrical drive
PU	50uA (typical) internal pull-up

Table 2.6 Buffer Types

TYPE	DESCRIPTION
PD	50uA (typical) internal pull-down
AI	Analog input
AO	Analog output
AIO	Analog bi-directional
ICLK	Crystal oscillator input pin
OCLK	Crystal oscillator output pin

Note 2.2

Chapter 3 Functional Description

3.1 10/100 Ethernet MAC

The Ethernet Media Access controller (MAC) incorporates the essential protocol requirements for operating an Ethernet/IEEE 802.3-compliant node and provides an interface between the host subsystem and the internal Ethernet PHY. The MAC can operate in either 100-Mbps or 10-Mbps mode.

The MAC operates in both half-duplex and full-duplex modes. When operating in half-duplex mode, the MAC complies fully with Section 4 of ISO/IEC 8802-3 (ANSI/IEEE standard) and ANSI/IEEE 802.3 standards. When operating in full-duplex mode, the MAC complies with IEEE 802.3x full-duplex operation standard.

The MAC provides programmable enhanced features designed to minimize host supervision, bus utilization, and pre- or post-message processing. These features include the ability to disable retries after a collision, dynamic FCS (Frame Check Sequence) generation on a frame-by-frame basis, automatic pad field insertion and deletion to enforce minimum frame size attributes, layer 3 checksum calculation for transmit and receive operations, and automatic retransmission and detection of collision frames.

The MAC can sustain transmission or reception of minimally-sized back-to-back packets at full line speed with an interpacket gap (IPG) of 9.6 microseconds for 10 Mbps and 0.96 microseconds for 100 Mbps.

The primary attributes of the MAC Function are:

- Transmit and receive message data encapsulation
- Framing (frame boundary delimitation, frame synchronization)
- Error detection (physical medium transmission errors)
- Media access management
- Medium allocation (collision detection, except in full-duplex operation)
- Contention resolution (collision handling, except in full-duplex operation)
- Flow control during full-duplex mode
- Decoding of control frames (PAUSE command) and disabling the transmitter
- Generation of control frames
- Interface to the internal PHY.
- Checksum offload engine for calculation of layer 3 transmit and receive checksum.

The transmit and receive data paths are separate within the LAN9210 from the MAC to host interface allowing the highest performance, especially in full duplex mode. Payload data as well as transmit and receive status are passed on these busses.

A third internal bus is used to access the MAC's "Control and Status Registers" (CSR's). This bus is also accessible from the host.

On the backend, the MAC interfaces with the 10/100 PHY through an MII (Media Independent Interface) port which is internal to the LAN9210. The MAC CSR's also provide a mechanism for accessing the PHY's internal registers through the internal SMI (Serial Management Interface) bus.

The receive and transmit FIFOs allow increased packet buffer storage to the MAC. The FIFOs are a conduit between the host interface and the MAC through which all transmitted and received data and status information is passed. Deep FIFOs allow a high degree of latency tolerance relative to the various transport and OS software stacks reducing and minimizing overrun conditions. Like the MAC, the FIFOs have separate receive and transmit data paths.

The LAN9210 can store up to 250 Ethernet packets utilizing FIFOs, totaling 16K bytes, with a packet granularity of 4 bytes. This memory is shared by the RX and TX blocks and is configurable in terms of allocation. This depth of buffer storage minimizes or eliminates receive overruns.

3.2 Flow Control

The LAN9210 Ethernet MAC supports full-duplex flow control using the pause operation and control frame. It also supports half-duplex flow control using back pressure.

3.2.1 Full-Duplex Flow Control

The pause operation inhibits data transmission of data frames for a specified period of time. A Pause operation consists of a frame containing the globally assigned multicast address (01-80-C2-00-00-01), the PAUSE opcode, and a parameter indicating the quantum of slot time (512 bit times) to inhibit data transmissions. The PAUSE parameter may range from 0 to 65,535 slot times. The Ethernet MAC logic, on receiving a frame with the reserved multicast address and PAUSE opcode, inhibits data frame transmissions for the length of time indicated. If a Pause request is received while a transmission is in progress, then the pause will take effect after the transmission is complete. Control frames are received and processed by the MAC and are passed on.

The MAC also transmits control frames (pause command) via both hardware and software control. The software driver requests the MAC to transmit a control frame and gives the value of the PAUSE time to be used in the control frame. The MAC Function constructs a control frame with the appropriate values set in all the different fields (as defined in the 802.3x specification) and transmits the frame to the MII interface. The transmission of the control frame is not affected by the current state of the Pause timer value that is set because of a recently received control frame.

3.2.2 Half-Duplex Flow Control (Backpressure)

In half-duplex mode, back pressure is used for flow control. Whenever the receive buffer/FIFO becomes full or crosses a certain threshold level, the MAC starts sending a Jam signal. The MAC transmit logic enters a state at the end of current transmission (if any), where it waits for the beginning of a received frame. Once a new frame starts, the MAC starts sending the jam signal, which will result in a collision. After sensing the collision, the remote station will back off its transmission. The MAC continues sending the jam to make other stations defer transmission. The MAC only generates this collision-based back pressure when it receives a new frame, in order to avoid any late collisions.

3.2.3 Virtual Local Area Network (VLAN) Support

Virtual Local Area Networks or VLANs, as defined within the IEEE 802.3 standard, provide network administrators one means of grouping nodes within a larger network into broadcast domains. To implement a VLAN, four extra bytes are added to the basic Ethernet packet. As shown in Figure 3.1, "VLAN Frame", the four bytes are inserted after the Source Address Field and before the Type/Length field. The first two bytes of the VLAN tag identify the tag, and by convention are set to the value 0x8100. The last two bytes identify the specific VLAN associated with the packet; they also provide a priority field.

The LAN9210 supports VLAN-tagged packets. The LAN9210 provides two registers which are used to identify VLAN-tagged packets. One register should normally be set to the conventional VLAN ID of 0x8100. The other register provides a way of identifying VLAN frames tagged with a proprietary (not 0x8100) identifier. If a packet arrives bearing either of these tags in the two bytes succeeding the Source Address field, the controller will recognize the packet as a VLAN-tagged packet. In this case, the controller increases the maximum allowed packet size from 1518 to 1522 bytes (normally the controller filters packets larger than 1518 bytes). This allows the packet to be received, and then processed by host software, or to be transmitted on the network.

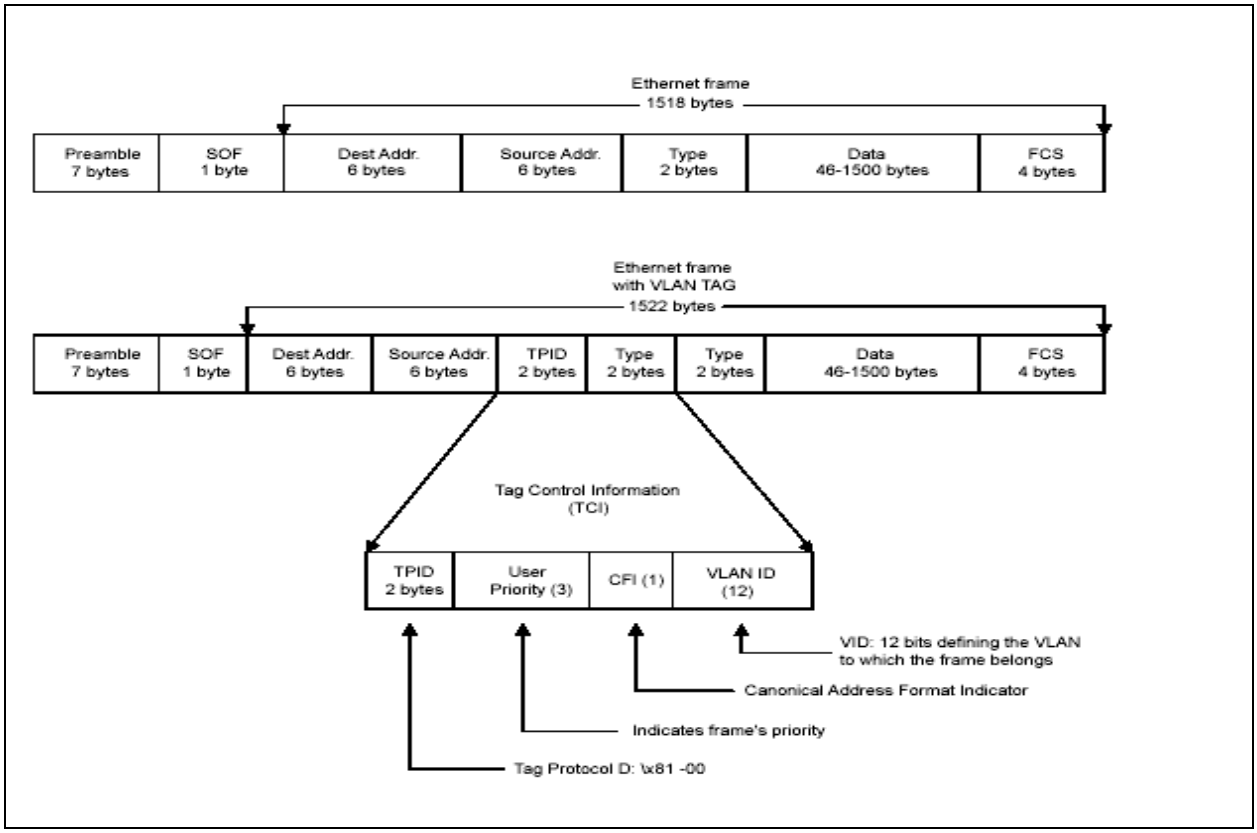


Figure 3.1 VLAN Frame

3.3 Address Filtering Functional Description

The Ethernet address fields of an Ethernet Packet, consists of two 6-byte fields: one for the destination address and one for the source address. The first bit of the destination address signifies whether it is a physical address or a multicast address.

The LAN9210 address check logic filters the frame based on the Ethernet receive filter mode that has been enabled. Filter modes are specified based on the state of the control bits in Table 3.1, "Address Filtering Modes", which shows the various filtering modes used by the Ethernet MAC Function. These bits are defined in more detail in the “MAC Control Register”. Please refer to Section 5.4.1, "MAC_CR—MAC Control Register" for more information on this register.

If the frame fails the filter, the Ethernet MAC function does not receive the packet. The host has the option of accepting or ignoring the packet.

Table 3.1 Address Filtering Modes

MCPAS	PRMS	INVFILT	HO	HPFILT	DESCRIPTION
0	0	0	0	0	MAC address perfect filtering only for all addresses.
0	0	0	0	1	MAC address perfect filtering for physical address and hash filtering for multicast addresses

Table 3.1 Address Filtering Modes (continued)

MCPAS	PRMS	INVFILT	HO	HPFILT	DESCRIPTION
0	0	0	1	1	Hash Filtering for physical and multicast addresses
0	0	1	0	0	Inverse Filtering
X	1	0	X	X	Promiscuous
1	0	0	0	X	Pass all multicast frames. Frames with physical addresses are perfect-filtered
1	0	0	1	1	Pass all multicast frames. Frames with physical addresses are hash-filtered

3.4 Filtering Modes

3.4.1 Perfect Filtering

This filtering mode passes only incoming frames whose destination address field exactly matches the value programmed into the MAC Address High register and the MAC address low register. The MAC address is formed by the concatenation of the above two registers in the MAC CSR Function.

3.4.2 Hash Only Filtering

This type of filtering checks for incoming Receive packets with either multicast or physical destination addresses, and executes an imperfect address filtering against the hash table.

During imperfect hash filtering, the destination address in the incoming frame is passed through the CRC logic and the upper six bits of the CRC register are used to index the contents of the hash table. The hash table is formed by merging the register's multicast hash table high and multicast hash table low in the MAC CSR Function to form a 64-bit hash table. The most significant bit determines the register to be used (High/Low), while the other five bits determine the bit within the register. A value of 00000 selects Bit 0 of the multicast hash table low register and a value of 11111 selects Bit 31 of the multicast hash table high register.

3.4.2.1 Hash Perfect Filtering

In hash perfect filtering, if the received frame is a physical address, the LAN9210 Packet Filter block perfect-filters the incoming frame's destination field with the value programmed into the MAC Address High register and the MAC Address Low register. If the incoming frame is a multicast frame, however, the LAN9210 packet filter function performs an imperfect address filtering against the hash table.

The imperfect filtering against the hash table is the same imperfect filtering process described in the "Hash Only Filtering" section above.

3.4.2.2 Inverse Filtering

In inverse filtering, the Packet Filter Block accepts incoming frames with a destination address not matching the perfect address (i.e., the value programmed into the MAC Address High register and the MAC Address Low register in the CRC block and rejects frames with destination addresses matching the perfect address).

For all filtering modes, when the MCPAS bit is set, all multicast frames are accepted. When the PRMS bit is set, all frames are accepted regardless of their destination address. This includes all broadcast frames as well.

3.5 Wake-up Frame Detection

Setting the Wake-Up Frame Enable bit (WUEN) in the “WUCSR—Wake-up Control and Status Register”, places the LAN9210 MAC in the wake-up frame detection mode. In this mode, normal data reception is disabled, and detection logic within the MAC examines receive data for the pre-programmed wake-up frame patterns. The LAN9210 can be programmed to notify the host of the wake-up frame detection with the assertion of the host interrupt (IRQ) or assertion of the power management event signal (PME). Upon detection, the Wake-Up Frame Received bit (WUFR) in the WUCSR is set. When the host clears the WUEN bit the LAN9210 will resume normal receive operation.

Before putting the MAC into the wake-up frame detection state, the host must provide the detection logic with a list of sample frames and their corresponding byte masks. This information is written into the Wake-up Frame Filter register (WUFF). Please refer to Section 5.4.11, "WUFF—Wake-up Frame Filter," on page 114 for additional information on this register.

The MAC supports four programmable filters that support many different receive packet patterns. If remote wake-up mode is enabled, the remote wake-up function receives all frames addressed to the MAC. It then checks each frame against the enabled filter and recognizes the frame as a remote wake-up frame if it passes the wakeup frame filter register’s address filtering and CRC value match.

In order to determine which bytes of the frames should be checked by the CRC module, the MAC uses a programmable byte mask and a programmable pattern offset for each of the four supported filters.

The pattern’s offset defines the location of the first byte that should be checked in the frame. Since the destination address is checked by the address filtering Function, the pattern offset is always greater than 12.

The byte mask is a 31-bit field that specifies whether or not each of the 31 contiguous bytes within the frame, beginning in the pattern offset, should be checked. If bit j in the byte mask is set, the detection logic checks byte offset +j in the frame. In order to load the Wake-up Frame Filter register, the host LAN driver software must perform eight writes to the Wake-up Frame Filter register (WUFF). The Diagram shown in Table 3.2, "Wake-Up Frame Filter Register Structure" below, shows the wake-up frame filter register’s structure.

Note 3.1 Wake-up frame detection can be performed when the LAN9210 is in the D0 or D1 power states. In the D0 state, wake-up frame detection is enabled when the WUEN bit is set.

Note 3.2 Wake-up frame detection, as well as Magic Packet detection, is always enabled and cannot be disabled when the device enters the D1 state.

Note 3.3 When wake-up frame detection is enabled via the WUEN bit of the WUCSR—Wake-up Control and Status Register, a broadcast wake-up frame will wake-up the device despite the state of the Disable Broadcast Frame (BCAST) bit in the MAC_CR—MAC Control Register.

Table 3.2 Wake-Up Frame Filter Register Structure

Filter 0 Byte Mask							
Filter 1 Byte Mask							
Filter 2 Byte Mask							
Filter 3 Byte Mask							
Reserved	Filter 3 Command	Reserved	Filter 2 Command	Reserved	Filter 1 Command	Reserved	Filter 0 Command
Filter 3 Offset		Filter 2 Offset		Filter 1Offset		Filter 0 Offset	
Filter 1 CRC-16				Filter 0 CRC-16			
Filter 3 CRC-16				Filter 2 CRC-16			

The Filter i Byte Mask defines which incoming frame bytes Filter i will examine to determine whether or not this is a wake-up frame. Table 3.3, describes the byte mask's bit fields.

Table 3.3 Filter i Byte Mask Bit Definitions

FILTER I BYTE MASK DESCRIPTION	
FIELD	**DESCRIPTION**
31	Must be zero (0)
30:0	**Byte Mask:** If bit j of the byte mask is set, the CRC machine processes byte number pattern - (offset + j) of the incoming frame. Otherwise, byte pattern - (offset + j) is ignored.

The Filter i command register controls Filter i operation. Table 3.4 shows the Filter I command register.

Table 3.4 Filter i Command Bit Definitions

FILTER i COMMANDS	
FIELD	**DESCRIPTION**
3	**Address Type:** Defines the destination address type of the pattern. When bit is set, the pattern applies only to multicast frames. When bit is cleared, the pattern applies only to unicast frames.
2:1	**RESERVED**
0	**Enable Filter:** When bit is set, Filter i is enabled, otherwise, Filter i is disabled.

The Filter i Offset register defines the offset in the frame's destination address field from which the frames are examined by Filter i. Table 3.5 describes the Filter i Offset bit fields.

Table 3.5 Filter i Offset Bit Definitions

FILTER I OFFSET DESCRIPTION	
FIELD	**DESCRIPTION**
7:0	**Pattern Offset:** The offset of the first byte in the frame on which CRC is checked for wake-up frame recognition. The minimum value of this field must be 12 since there should be no CRC check for the destination address and the source address fields. The MAC checks the first offset byte of the frame for CRC and checks to determine whether the frame is a wake-up frame. Offset 0 is the first byte of the incoming frame's destination address.

The Filter i CRC-16 register contains the CRC-16 result of the frame that should pass Filter i.

Table 3.6 describes the Filter i CRC-16 bit fields.

Table 3.6 Filter i CRC-16 Bit Definitions

FILTER I CRC-16 DESCRIPTION	
FIELD	**DESCRIPTION**
15:0	**Pattern CRC-16:** This field contains the 16-bit CRC value from the pattern and the byte mask programmed to the wake-up filter register Function. This value is compared against the CRC calculated on the incoming frame, and a match indicates the reception of a wakeup frame.

3.5.1 Magic Packet Detection

Setting the Magic Packet Enable bit (MPEN) in the “WUCSR—Wake-up Control and Status Register”, places the LAN9210 MAC in the “Magic Packet” detection mode. In this mode, normal data reception is disabled, and detection logic within the MAC examines receive data for a Magic Packet. The LAN9210 can be programmed to notify the host of the “Magic Packet” detection with the assertion of the host interrupt (IRQ) or assertion of the power management event signal (PME). Upon detection, the Magic Packet Received bit (MPR) in the WUCSR is set. When the host clears the MPEN bit the LAN9210 will resume normal receive operation. Please refer to Section 5.4.12, "WUCSR—Wake-up Control and Status Register," on page 114 for additional information on this register

In Magic Packet mode, the Power Management Logic constantly monitors each frame addressed to the node for a specific Magic Packet pattern. It checks only packets with the MAC’s address or a broadcast address to meet the Magic Packet requirement. The Power Management Logic checks each received frame for the pattern 48h FF_FF_FF_FF_FF_FF after the destination and source address field.

Then the Function looks in the frame for 16 repetitions of the MAC address without any breaks or interruptions. In case of a break in the 16 address repetitions, the PMT Function scans for the 48'hFF_FF_FF_FF_FF_FF pattern again in the incoming frame.

The 16 repetitions may be anywhere in the frame but must be preceded by the synchronization stream. The device will also accept a multicast frame, as long as it detects the 16 duplications of the MAC address. If the MAC address of a node is 00h 11h 22h 33h 44h 55h, then the MAC scans for the following data sequence in an Ethernet: Frame.

Destination Address Source Address ……FF FF FF FF FF FF

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55 00 11 22 33 44 55

…CRC

It should be noted that Magic Packet detection can be performed when LAN9210 is in the D0 or D1 power states. In the D0 state, “Magic Packet” detection is enabled when the MPEN bit is set. In the D1 state, Magic Packet detection, as well as wake-up frame detection, are automatically enabled when the device enters the D1 state.

3.6 Checksum Offload Engines (COE)

The LAN9210 contains two checksum offload engines, which offload the calculation of the 16-bit checksum for transmitted and received Ethernet frames. The functionality of the checksum offload engines is described in the following sections:

- Receive Checksum Offload Engine (RXCOE)
- Transmit Checksum Offload Engine (TXCOE)

3.6.1 Receive Checksum Offload Engine (RXCOE)

The receive checksum offload engine provides assistance to the CPU by calculating a 16-bit checksum for a received Ethernet frame. The RXCOE readily supports the following IEEE802.3 frame formats:

- Type II Ethernet frames
- SNAP encapsulated frames
- Support for up to 2, 802.1q VLAN tags

The resulting checksum value can also be modified by software to support other frame formats.

The RXCOE has two modes of operation. In mode 0, the RXCOE calculates the checksum between the first 14 bytes of the Ethernet frame and the FCS. This is illustrated in Figure 3.2.

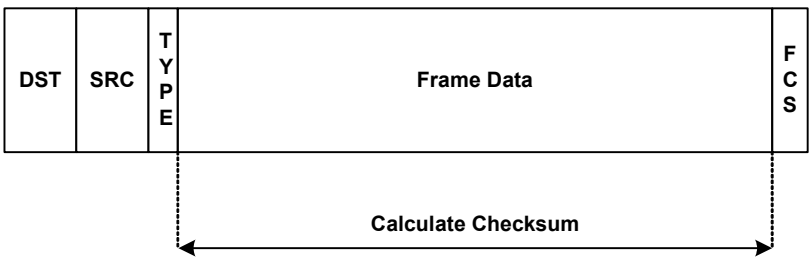


Figure 3.2 RXCOE Checksum Calculation

In mode 1, the RXCOE supports VLAN tags and a SNAP header. In this mode the RXCOE calculates the checksum at the start of L3 packet. The VLAN1 tag register is used by the RXCOE to indicate what protocol type is to be used to indicate the existence of a VLAN tag. This value is typically 8100h.

Example frame configurations:

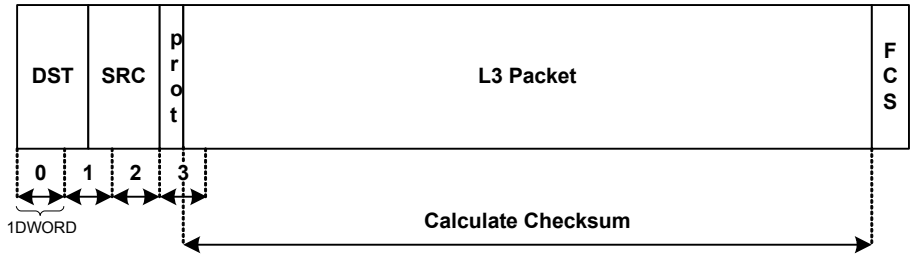


Figure 3.3 Type II Ethernet Frame

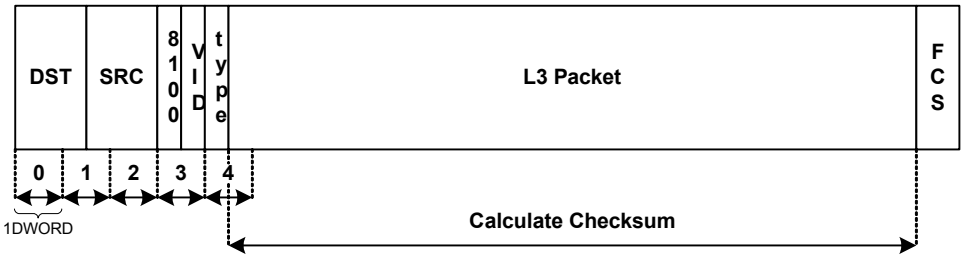


Figure 3.4 Ethernet Frame with VLAN Tag

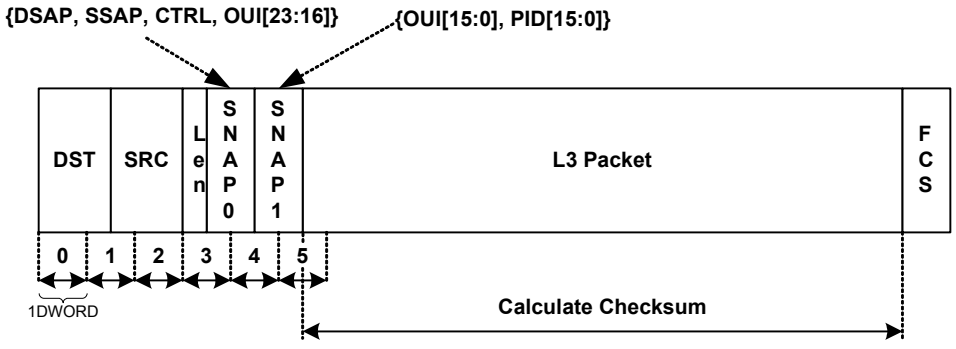


Figure 3.5 Ethernet Frame with Length Field and SNAP Header

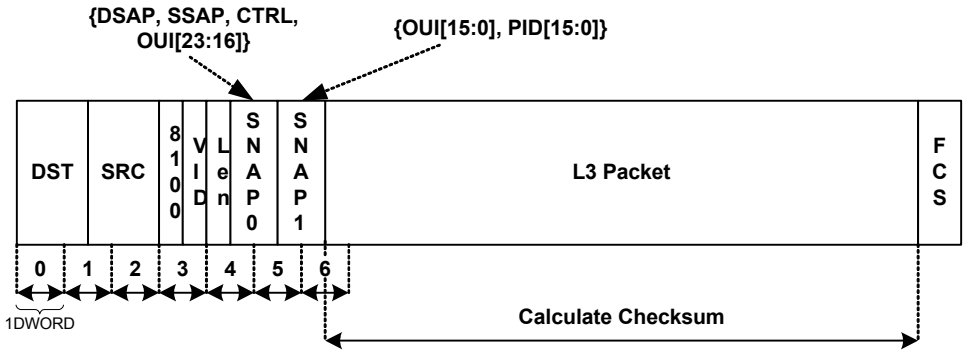


Figure 3.6 Ethernet Frame with VLAN Tag and SNAP Header

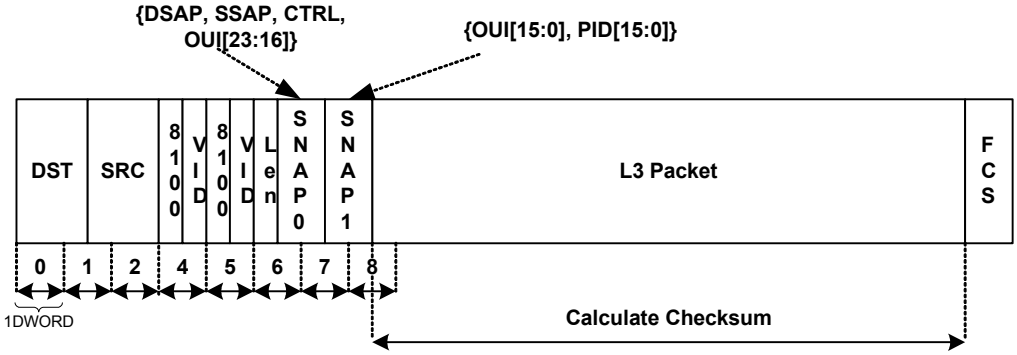


Figure 3.7 Ethernet Frame with Multiple VLAN Tags and SNAP Header

The RXCOE supports a maximum of two VLAN tags. If there are more than two VLAN tags, the VLAN protocol identifier for the third tag is treated as an Ethernet type field. The checksum calculation will begin immediately after the type field.

The RXCOE resides in the RX path within the MAC. As the RXCOE receives an Ethernet frame it calculates the 16-bit checksum. The RXCOE passes the Ethernet frame to the RX Data FIFO with the checksum appended to the end of the frame. The RXCOE inserts the checksum immediately after the last byte of the Ethernet frame. The packet length field in the RX status word (refer to Section 3.13.3) will indicate that the frame size has increased by two bytes to accommodate the checksum.

Setting the RXCOE_EN bit in the COE_CR—Checksum Offload Engine Control Register enables the RXCOE, while the RXCOE_MODE bit selects the operating mode. When the RXCOE is disabled, the the received data is simply passed through the RXCOE unmodified.

Note: Software applications must stop the receiver and flush the RX data path before changing the state of the RXCOE_EN or RXCOE_MODE bits.

Note: When the RXCOE is enabled, automatic pad stripping must be disabled (bit 8 (PADSTR) of the MAC_CR—MAC Control Register) and vice versa. These functions cannot be enabled simultaneously.

3.6.1.1 RX Checksum Calculation

The checksum is calculated 16 bits at a time. In the case of an odd sized frame, an extra byte of zero is used to pad up to 16 bits.

Consider the following packet: DA, SA, Type, B0, B1, B2 … BN, FCS

Let [A, B] = A*256 + B;

If the packet has an even number of octets then

checksum = [B1, B0] + C0 + [B3, B2] + C1 + … + [BN, BN-1] + CN-1

Where C0, C1, ... CN-1 are the carry out results of the intermediate sums.

If the packet has an odd number of octets then

checksum = [B1, B0] + C0 + [B3, B2] + C1 + … + [0, BN] + CN-1

3.6.2 Transmit Checksum Offload Engine (TXCOE)

The transmit checksum offload engine provides assistance to the CPU by calculating a 16-bit checksum, typically for TCP, for a transmit Ethernet frame. The TXCOE calculates the checksum and inserts the results back into the data stream as it is transferred to the MAC.

To activate the TXCOE and perform a checksum calculation, the host must first set the TX checksum offload engine enable bit (TXCOE_EN) in the COE_CR—Checksum Offload Engine Control Register. The host then pre-pends a 3 DWORD buffer to the data that will be transmitted. The pre-pended buffer includes a TX Command ‘A’, TX Command ‘B’, and a 32-bit TX checksum preamble. When bit 14 (CK) of the TX Command ‘B’ is set in conjunction with bit 13 (FS) of TX Command ‘A’ and bit 16 (TXCOE_EN) of the COE_CR register, the TXCOE will perform a checksum calculation on the associated packet. When these three bits are set, a 32-bit TX checksum preamble must be pre-pended to the beginning of the TX packet (refer to Table 3.7). The TX checksum preamble instructs the TXCOE on the handling of the associated packet. Bits 11:0 of the TX checksum preamble define the byte offset at which the data checksum calculation will begin (TXCSSP). The checksum calculation will begin at this offset and will continue until the end of the packet. The data checksum calculation must not begin in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet. When the calculation is complete, the checksum will be inserted into the packet at the byte offset defined by bits 27:16 of the TX checksum preamble (TXCSLOC). The TX checksum cannot be inserted in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet. If the CK bit is not set in the first TX Command ‘B’ of a packet, the packet is passed directly through the TXCOE without modification, regardless if the TXCOE_EN is set. An example of a TX packet with a pre-pended TX checksum preamble can be found in Section 3.12.6.3, "TX Example 3". In this example the host writes the packet data to the ethernet controller in four fragments, the first containing the TX Checksum Preamble. Figure 3.23 shows how these fragments are loaded into the TX Data FIFO. For more information on the TX Command ‘A’ and TX Command ‘B’, refer to Section 3.12.2, "TX Command Format".

If the TX packet already includes a partial checksum calculation (perhaps inserted by an upper layer protocol), this checksum can be included in the hardware checksum calculation by setting the TXCSSP field in the TX checksum preamble to include the partial checksum. The partial checksum can be replaced by the completed checksum calculation by setting the TXCSLOC pointer to point to the location of the partial checksum.

Table 3.7 TX Checksum Preamble

FIELD	DESCRIPTION
31:28	**RESERVED**
27:16	**TXCSLOC - TX Checksum Location** This field specifies the byte offset where the TX checksum will be inserted in the TX packet. The checksum will replace two bytes of data starting at this offset. **Note:** The TX checksum cannot be inserted in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet.
15:12	**RESERVED**
11:0	**TXCSSP - TX Checksum Start Pointer** This field indicates start offset, in bytes, where the checksum calculation will begin in the associated TX packet. **Note:** The data checksum calculation must not begin in the MAC header (first 14 bytes) or in the last 4 bytes of the TX packet.

Note: When the TXCOE is enabled, the third DWORD of the pre-pended packet is not transmitted. However, 4 bytes must be added to the packet length field in TX Command 'B'.

Note: The TX checksum preamble must be DWORD-aligned (i.e., the two least significant bits of the Data Start Offset fields in TX Command "A" must be zero). Any valid buffer end alignment setting can be used.

Note: Software applications must stop the transmitter and flush the TX data path before changing the state of the TXCOE_EN bit. However, the CK bit of TX Command 'B' can be set or cleared on a per-packet basis.

3.6.2.1 TX Checksum Calculation

The TX checksum calculation is performed using the same operation as the RX checksum shown in Section 3.6.1.1, with the exception that the calculation starts as indicated by the preamble, and the transmitted checksum is the one's-compliment of the final calculation.

Note: When the TX checksum offload feature is invoked, if the calculated checksum is 0000h, it is left unaltered. UDP checksums are optional under IPv4, and a zero checksum calculated by the TX checksum offload feature will erroneously indicate to the receiver that no checksum was calculated, however, the packet will typically not be rejected by the receiver. Under IPv6, however, according to RFC 2460, the UDP checksum is not optional. A calculated checksum that yields a result of zero must be changed to FFFFh for insertion into the UDP header. IPv6 receivers discard UDP packets containing a zero checksum. **Thus, this feature must not be used for UDP checksum calculation under IPv6.**

3.7 Host Bus Operations

3.7.1 Bus Writes

The host processor is required to perform two contiguous 16-bit writes to complete a single DWORD transfer. This DWORD must begin and end on a DWORD address boundary (A[2] and higher, cannot change during a sixteen bit write). No ordering requirements exist. The processor can access either the low or high word first, as long as the next write is performed to the other word. If a write to the same word is performed, the LAN9210 disregards the transfer.

3.7.2 Bus Reads

The host processor is required to perform two consecutive 16-bit reads to complete a single DWORD transfer. This DWORD must begin and end on a DWORD address boundary (A[2] and higher, cannot change during a sixteen bit read). No ordering requirements exist. The processor can access either the low or high word first, as long as the next read is performed from the other word. If a read to the same word is performed, the data read is invalid and should be re-read. This is not a fatal error. The LAN9210 will reset its read counters and restart a new cycle on the next read.

3.7.3 Mixed Endian Support

In order to allow flexibility with a range of designs, the LAN9210 supports mixed endian Data FIFO accesses. The LAN9210 provides the ability to select Data FIFO endianess separately for accesses through the Data FIFO ports (addresses 00h-3Ch) or using the FIFO_SEL input signal. This is accomplished via the FPORTEND and FSELEND bits of the HW_CFG—Hardware Configuration Register, respectively.

The FPORTEND bit determines the endianess of RX and TX Data FIFO host accesses made through the Data FIFO port addresses (00h-3Ch). When FPORTEND is cleared, Data FIFO port accesses utilize little endian byte ordering. When FPORTEND is set, Data FIFO port accesses utilize big endian byte ordering.

The FSELEND bit determines the endianess of RX and TX Data FIFO host accesses when using the FIFO_SEL signal. When FSELEND is cleared, FIFO_SEL accesses utilize little endian byte ordering. When FSELEND is set, FIFO_SEL accesses utilize big endian byte ordering.

In addition to mixed endian support, the LAN9210 provides a word swap function, as described in Section 3.7.4. The word swap function combined with the endianess select bits described above determines how the Data/Status FIFO's and CSR host access byte ordering is applied. Table 3.8 describes the various operation modes of the endianess and word swap ordering logic. Figure 3.9 illustrates the FIFO access byte ordering under various endianess and word swap settings. Refer to Section 3.7.4 for additional details.

Note: CSR and status FIFO accesses are not affected by the FPORTEND and FSELEND endianess select bits.

3.7.4 Word Swap Function

In addition to mixed endian functionality, the LAN9210 supports a Word Swap Function. This feature is controlled by the Word Swap Register, which is described in Section 5.3.17, "WORD_SWAP—Word Swap Control," on page 96. This register affects how words on the data bus are written to or read from the Control and Status Registers and the Transmit and Receive Data/Status FIFOs.

Both the word swap function and the mixed endian control bits contain the ability to change the byte ordering of host data path accesses. Figure 3.8 illustrates the order in which the word swap and endianess select logic is applied within the LAN9210. Logically, the endian control logic is applied after the word swap logic for write operations, and before the word swap logic for read operations.

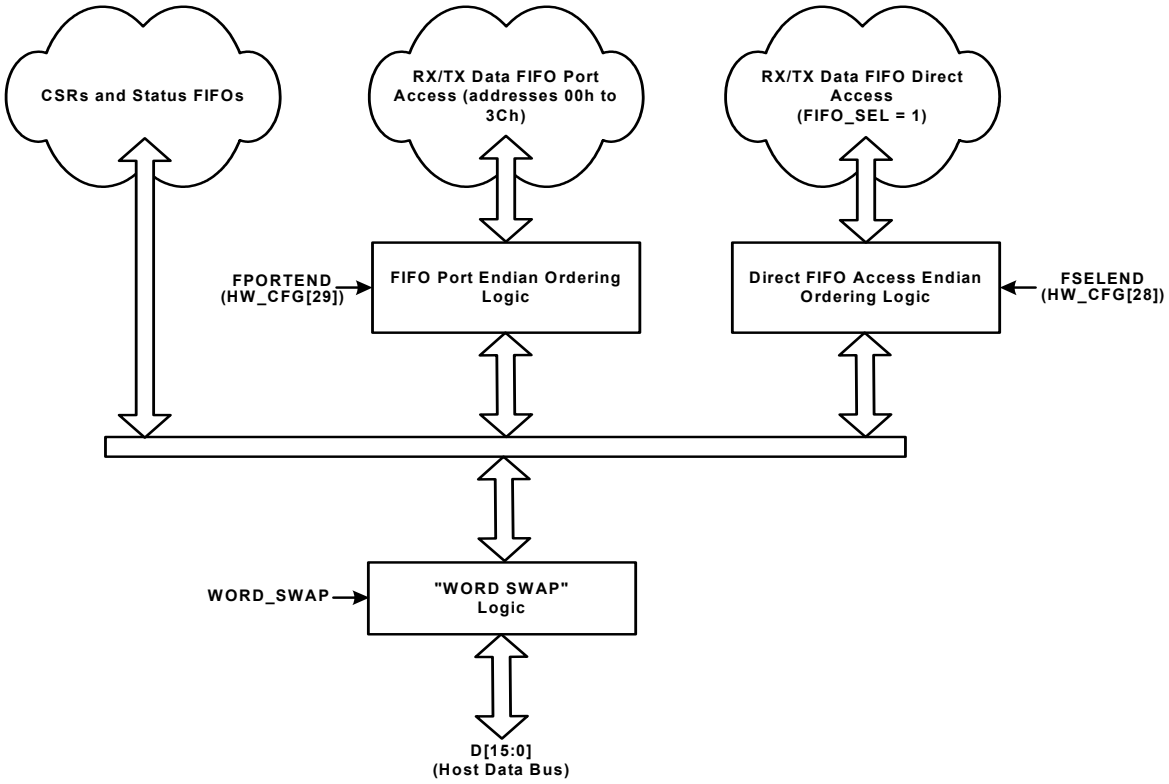


Figure 3.8 LAN9210 Host Data Path Diagram

Data path operations for the various supported endianess and word swap configurations are illustrated in Figure 3.9. Table 3.8, "Endian Ordering Logic Operation" illustrates the byte ordering applied by the endian logic for each type of host access. This figure and table assume an internal byte ordering of 3-2-1-0, where ‘3’ is the most significant byte (data[31:24]) and ‘0’ is the least significant byte (data[7:0]).

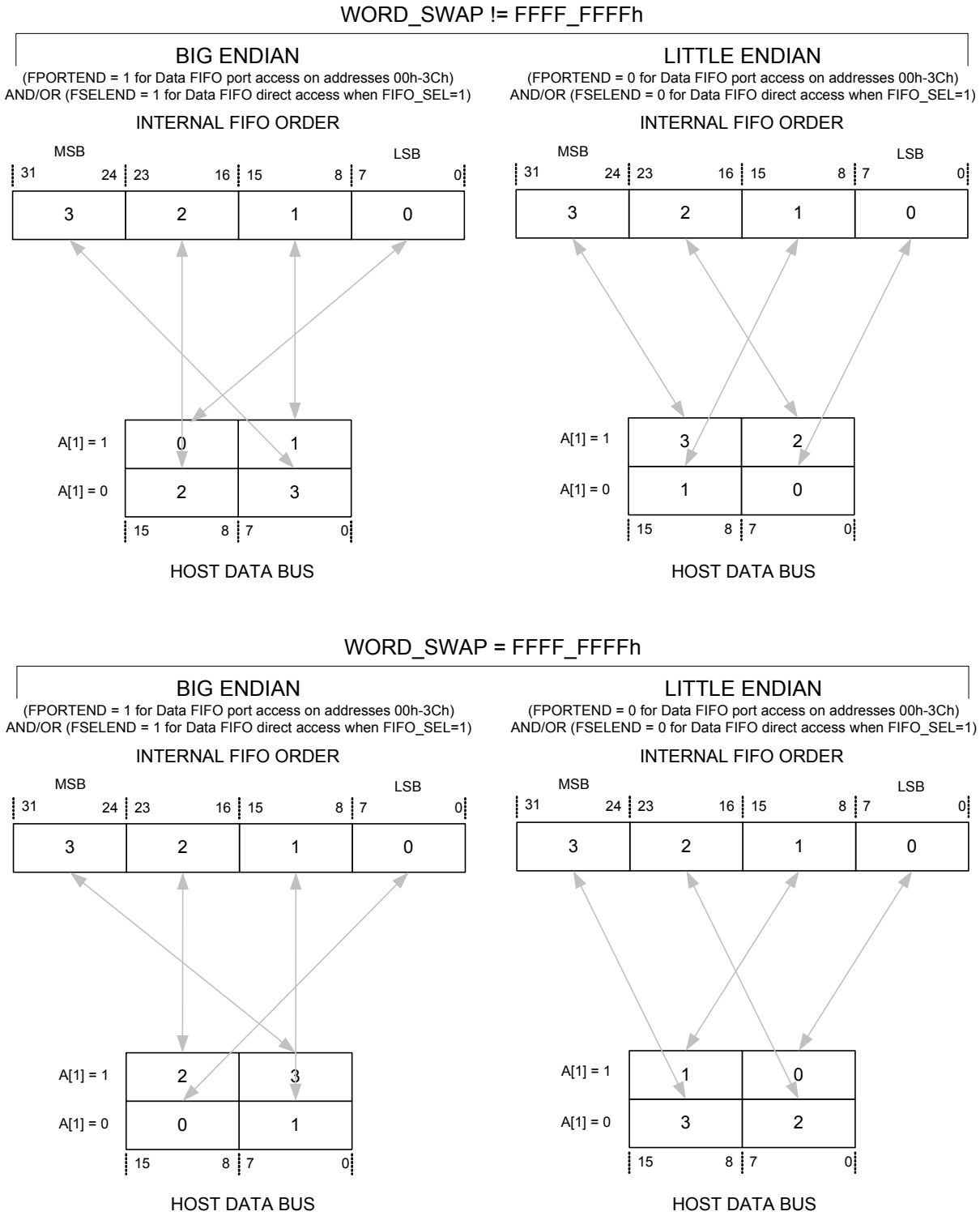


Figure 3.9 FIFO Access Byte Ordering

Table 3.8 Endian Ordering Logic Operation

			FIFO Access via Data FIFO Port (00h-3Ch)		Direct FIFO Access via FIFO_SEL		CSR Access	
			Host Data Bus		Host Data Bus		Host Data Bus	
			D[15:8]	D[7:0]	D[15:8]	D[7:0]	D[15:8]	D[7:0]
WORD_SWAP != FFFF_FFFFh	FPORTEND=0 FSELEND=0	A1=1	3	2	3	2	3	2
		A1=0	1	0	1	0	1	0
	FPORTEND=1 FSELEND=0	A1=1	0	1	3	2	3	2
		A1=0	2	3	1	0	1	0
	FPORTEND=0 FSELEND=1	A1=1	3	2	0	1	3	2
		A1=0	1	0	2	3	1	0
	FPORTEND=1 FSELEND=1	A1=1	0	1	0	1	3	2
		A1=0	2	3	2	3	1	0
WORD_SWAP = FFFF_FFFFh	FPORTEND=0 FSELEND=0	A1=1	1	0	1	0	1	0
		A1=0	3	2	3	2	3	2
	FPORTEND=1 FSELEND=0	A1=1	2	3	1	0	1	0
		A1=0	0	1	3	2	3	2
	PORTEND=0 FSELEND=1	A1=1	1	0	2	3	1	0
		A1=0	3	2	0	1	3	2
	PORTEND=1 FSELEND=1	A1=1	2	3	2	3	1	0
		A1=0	0	1	0	1	3	2

3.8 General Purpose Timer (GP Timer)

The General Purpose Timer is a programmable block that can be used to generate periodic host interrupts. The resolution of this timer is 100uS.

The GP Timer loads the GPT_CNT Register with the value in the GPT_LOAD field and begins counting down when the TIMER_EN bit is set to a '1.' On a reset, or when the TIMER_EN bit changes from set '1' to cleared '0,' the GPT_LOAD field is initialized to FFFFh. The GPT_CNT register is also initialized to FFFFh on a reset. Software can write the pre-load value into the GPT_LOAD field at any time; e.g., before or after the TIMER_EN bit is asserted. The GPT Enable bit TIMER_EN is located in the GPT_CFG register.

Once enabled, the GPT counts down either until it reaches 0000h or until a new pre-load value is written to the GPT_LOAD field. At 0000h, the counter wraps around to FFFFh, asserts the GPT interrupt status bit and the IRQ signal if the GPT_INT_EN bit is set, and continues counting. The GPT interrupt status bit is in the INT_STS Register. The GPT_INT hardware interrupt can only be set if the GPT_INT_EN bit is set. GPT_INT is a sticky bit (R/WC); i.e., once the GPT_INT bit is set, it can only be cleared by writing a '1' to the bit.

3.9 EEPROM Interface

The LAN9210 can optionally load its MAC address from an external serial EEPROM. If a properly configured EEPROM is detected by the LAN9210 at power-up, hard reset or soft reset, the ADDRH and ADDRL registers will be loaded with the contents of the EEPROM. If a properly configured EEPROM is not detected, it is the responsibility of the host LAN Driver to set the IEEE addresses.

The LAN9210 EEPROM controller also allows the host system to read, write and erase the contents of the Serial EEPROM. The EEPROM controller supports most “93C46” type EEPROMs configured for 128 x 8-bit operation.

3.9.1 MAC Address Auto-Load

On power-up, hard reset or soft reset, the EEPROM controller attempts to read the first byte of data from the EEPROM (address 00h). If the value A5h is read from the first address, then the EEPROM controller will assume that an external Serial EEPROM is present. The EEPROM controller will then access the next EEPROM byte and send it to the MAC Address register byte 0 (ADDRL[7:0]). This process will be repeated for the next five bytes of the MAC Address, thus fully programming the 48-bit MAC address. Once all six bytes have been programmed, the “MAC Address Loaded” bit is set in the E2P_CMD register. A detailed explanation of the EEPROM byte ordering with respect to the MAC address is given in Section 5.4.3, "ADDRL—MAC Address Low Register," on page 109.

If an 0xA5h is not read from the first address, the EEPROM controller will end initialization. It is then the responsibility of the host LAN driver software to set the IEEE address by writing to the MAC’s ADDRH and ADDRL registers.

The host can initiate a reload of the MAC address from the EEPROM by issuing the RELOAD command via the E2P command (E2P_CMD) register. If the first byte read from the EEPROM is not A5h, it is assumed that the EEPROM is not present, or not programmed, and the MAC address reload will fail. The “MAC Address Loaded” bit indicates a successful reload of the MAC address.

3.9.2 EEPROM Host Operations

After the EEPROM controller has finished reading (or attempting to read) the MAC after power-on, hard reset or soft reset, the host is free to perform other EEPROM operations. EEPROM operations are performed using the E2P_CMD and E2P data (E2P_DATA) registers. Section 5.3.23, "E2P_CMD – EEPROM Command Register," on page 101 provides an explanation of the supported EEPROM operations.

If the EEPROM operation is the “write location” (WRITE) or “write all” (WRAL) commands, the host must first write the desired data into the E2P_DATA register. The host must then issue the WRITE or WRAL command using the E2P_CMD register by setting the EPC_CMD field appropriately. If the operation is a WRITE, the EPC_ADDR field in E2P_CMD must also be set to the desired location. The command is executed when the host sets the EPC_BSY bit high. The completion of the operation is indicated when the EPC_BSY bit is cleared.

If the EEPROM operation is the “read location” (READ) operation, the host must issue the READ command using the E2P_CMD with the EPC_ADDR set to the desired location. The command is executed when the host sets the EPC_BSY bit high. The completion of the operation is indicated when the EPC_BSY bit is cleared, at which time the data from the EEPROM may be read from the E2P_DATA register.

Other EEPROM operations are performed by writing the appropriate command to the EPC_CMD register. The command is executed when the host sets the EPC_BSY bit high. The completion of the operation is indicated when the EPC_BSY bit is cleared. In all cases the host must wait for EPC_BSY to clear before modifying the E2P_CMD register.

Note: The EEPROM device powers-up in the erase/write disabled state. To modify the contents of the EEPROM the host must first issue the EWEN command.

If an operation is attempted, and an EEPROM device does not respond within 30mS, the LAN9210 will timeout, and the EPC timeout bit (EPC_TO) in the E2P_CMD register will be set.

Figure 3.10, "EEPROM Access Flow Diagram" illustrates the host accesses required to perform an EEPROM Read or Write operation.

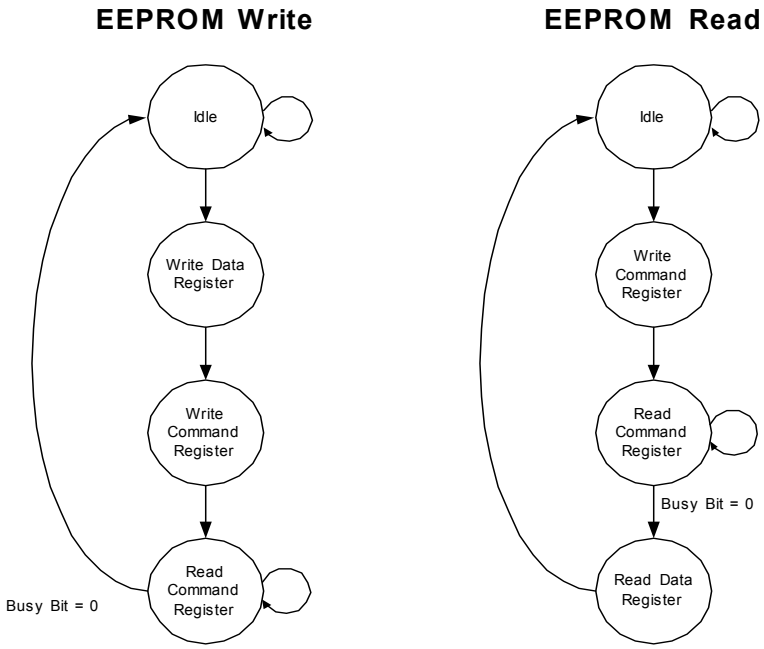


Figure 3.10 EEPROM Access Flow Diagram

The host can disable the EEPROM interface through the GPIO_CFG register. When the interface is disabled, the EEDIO and ECLK signals can be used as general-purpose outputs, or they may be used to monitor internal MII signals.

3.9.2.1 Supported EEPROM Operations

The EEPROM controller supports the following EEPROM operations under host control via the E2P_CMD register. The operations are commonly supported by “93C46” EEPROM devices. A description and functional timing diagram is provided below for each operation. Please refer to the E2P_CMD register description in Section 5.3.23, "E2P_CMD – EEPROM Command Register," on page 101 for E2P_CMD field settings for each command.

ERASE (Erase Location): If erase/write operations are enabled in the EEPROM, this command will erase the location selected by the EPC Address field (EPC_ADDR). The EPC_TO bit is set if the EEPROM does not respond within 30ms.

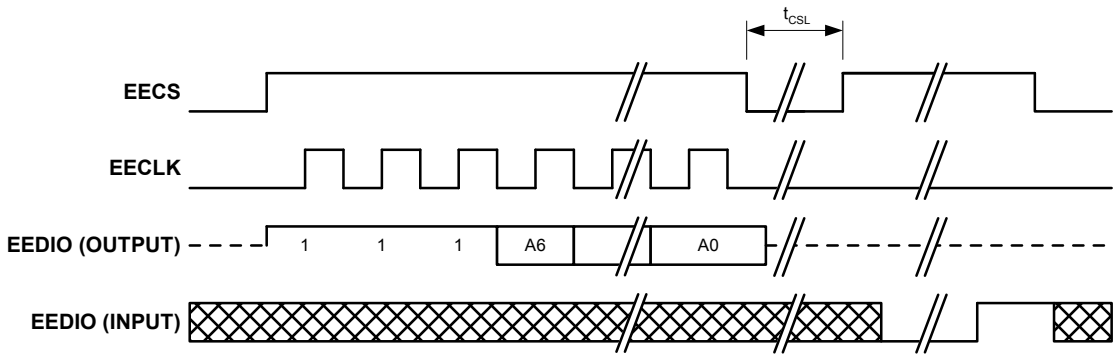


Figure 3.11 EEPROM ERASE Cycle

ERAL (Erase All): If erase/write operations are enabled in the EEPROM, this command will initiate a bulk erase of the entire EEPROM.The EPC_TO bit is set if the EEPROM does not respond within 30ms.

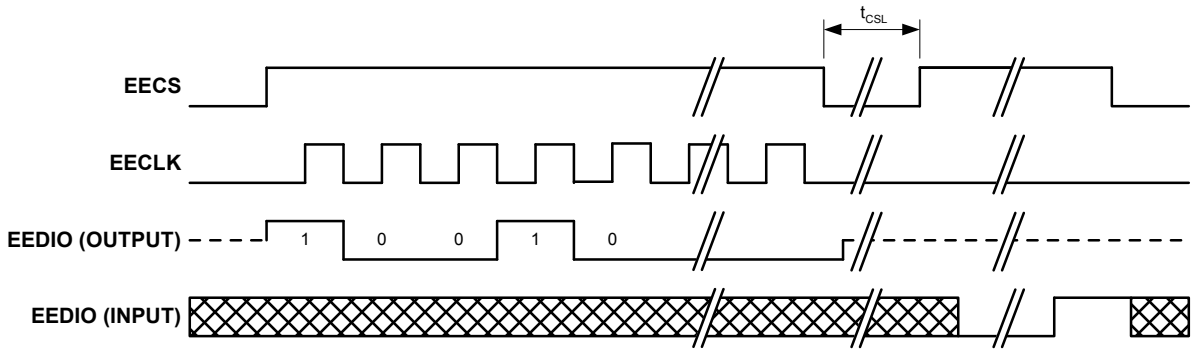


Figure 3.12 EEPROM ERAL Cycle

EWDS (Erase/Write Disable): After issued, the EEPROM will ignore erase and write commands. To re-enable erase/write operations issue the EWEN command.

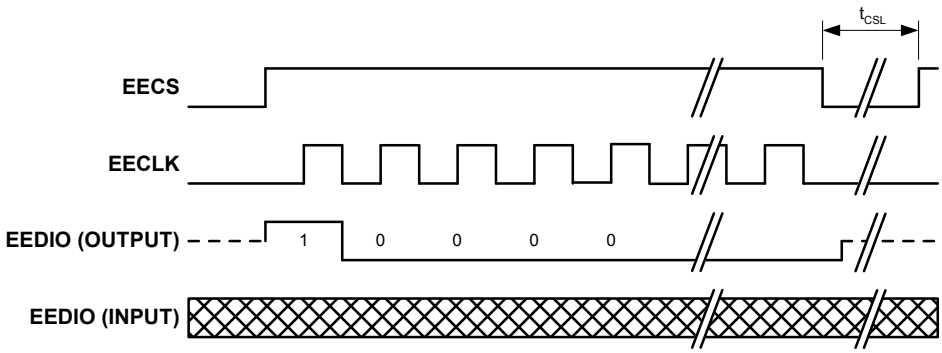


Figure 3.13 EEPROM EWDS Cycle

EWEN (Erase/Write Enable): Enables the EEPROM for erase and write operations. The EEPROM will allow erase and write operations until the “Erase/Write Disable” command is sent, or until power is cycled.

Note: The EEPROM device will power-up in the erase/write-disabled state. Any erase or write operations will fail until an Erase/Write Enable command is issued.

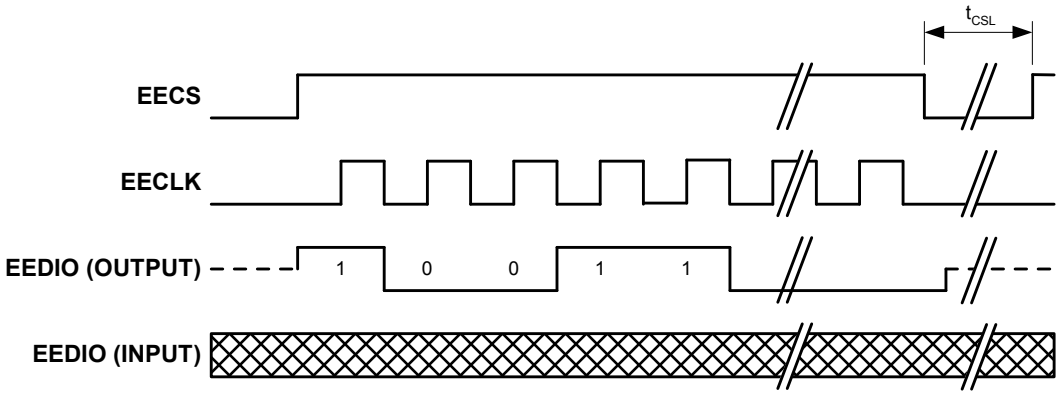


Figure 3.14 EEPROM EWEN Cycle

READ (Read Location): This command will cause a read of the EEPROM location pointed to by EPC Address (EPC_ADDR). The result of the read is available in the E2P_DATA register.

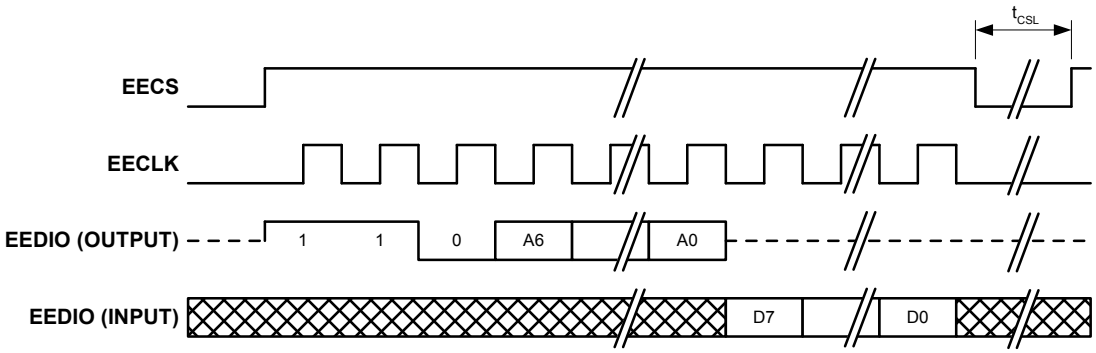


Figure 3.15 EEPROM READ Cycle

WRITE (Write Location): If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to the EEPROM location selected by the EPC Address field (EPC_ADDR). The EPC_TO bit is set if the EEPROM does not respond within 30ms.

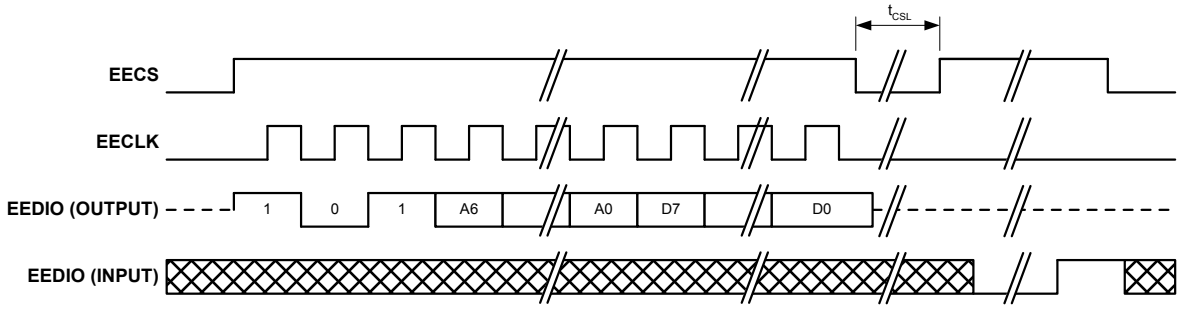


Figure 3.16 EEPROM WRITE Cycle

WRAL (Write All): If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to every EEPROM memory location. The EPC_TO bit is set if the EEPROM does not respond within 30ms.

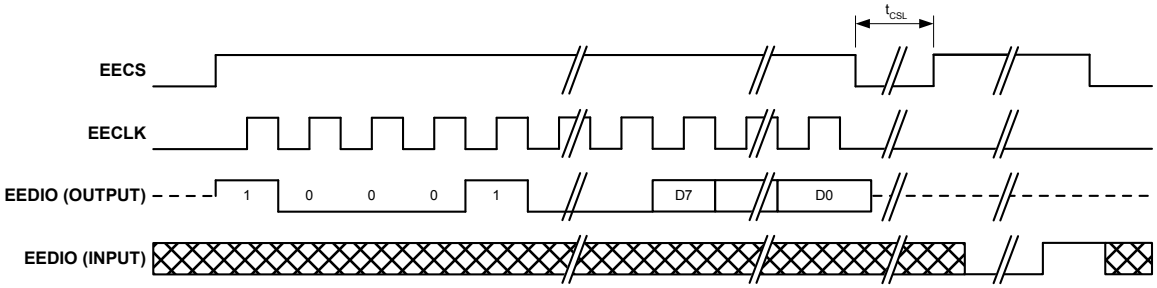


Figure 3.17 EEPROM WRAL Cycle

Table 3.9, "Required EECLK Cycles", shown below, shows the number of EECLK cycles required for each EEPROM operation.

Table 3.9 Required EECLK Cycles

OPERATION	REQUIRED EECLK CYCLES
ERASE	10
ERAL	10
EWDS	10
EWEN	10
READ	18
WRITE	18
WRAL	18

3.9.2.2 MAC Address Reload

The MAC address can be reloaded from the EEPROM via a host command to the E2P_CMD register. If a value of 0xA5h is not found in the first address of the EEPROM, the EEPROM is assumed to be un-programmed and MAC Address Reload operation will fail. The “MAC Address Loaded” bit indicates a successful load of the MAC address. The EPC_LOAD bit is set after a successful reload of the MAC address.

3.9.2.3 EEPROM Command and Data Registers

Refer to Section 5.3.23, "E2P_CMD – EEPROM Command Register," on page 101 and Section 5.3.24, "E2P_DATA – EEPROM Data Register," on page 103 for a detailed description of these registers. Supported EEPROM operations are described in these sections.

3.9.2.4 EEPROM Timing

Refer to Section 6.9, "EEPROM Timing," on page 136 for detailed EEPROM timing specifications.

3.10 Power Management

The LAN9210 supports power-down modes to allow applications to minimize power consumption. The following sections describe these modes.

3.10.1 System Description

Power is reduced to various modules by disabling the clocks as outlined in Table 3.10, “Power Management States,” on page 44. All configuration data is saved when in either of the two low power states. Register contents are not affected unless specifically indicated in the register description.

3.10.2 Functional Description

There is one normal operating power state, D0 and there are two power saving states: D1, and D2. Upon entry into either of the two power saving states, only the PMT_CTRL register is accessible for read operations. In either of the power saving states the READY bit in the PMT_CTRL register will be cleared. Reads of any other addresses are forbidden until the READY bit is set. All writes, with the exception of the wakeup write to BYTE_TEST, are also forbidden until the READY bit is set. Only when in the D0 (Normal) state, when the READY bit is set, can the rest of the device be accessed.

Note 3.4 The LAN9210 must always be read at least once after power-up, reset, or upon return from a power-saving state, otherwise write operations will not function.

In system configurations where the PME signal is shared amongst multiple devices, the WUPS field within the PMT_CTRL register can be read to determine which LAN9210 device is driving the PME signal.

When the LAN9210 is in a power saving state (D1 or D2), a write cycle to the BYTE_TEST register will return the LAN9210 to the D0 state. Table 7.2, “Power Consumption Device and System Components,” on page 139 and Table 7.2, “Power Consumption Device and System Components,” on page 139, shows the power consumption values for each power state.

Note 3.5 When the LAN9210 is in a power saving state, a write of any data to the BYTE_TEST register will wake-up the device. DO NOT PERFORM WRITES TO OTHER ADDRRESSES while the READY bit in the PMT_CTRL register is cleared.

3.10.2.1 D1 Sleep

Power consumption is reduced in this state by disabling clocks to portions of the internal logic as shown in Table 3.10. In this mode the clock to the internal PHY and portions of the MAC are still operational. This state is entered when the host writes a '01' to the PM_MODE bits in the Power Management (PMT_CTRL) register. The READY bit in PMT_CTRL is cleared when entering the D1 state.

Wake-up frame and Magic Packet detection are automatically enabled in the D1 state. If properly enabled via the WOL_EN and PME_EN bits, the LAN9210 will assert the PME hardware signal upon the detection of the wake-up frame or magic packet. The LAN9210 can also assert the host interrupt (IRQ) on detection of a wake-up frame or magic packet. Upon detection, the WUPS field in PMT_CTRL will be set to a 10b.

Note 3.6 The PME interrupt status bit (PME_INT) in the INT_STS register is set regardless of the setting of PME_EN.

Note 3.7 Wake-up frame and Magic Packet detection is automatically enabled when entering the D1 state. For wake-up frame detection, the wake-up frame filter must be programmed before entering the D1 state (see Section 3.5, "Wake-up Frame Detection," on page 25). If used, the host interrupt and PME signal must be enabled prior to entering the D1 state.

A write to the BYTE_TEST register, regardless of whether a wake-up frame or Magic Packet was detected, will return LAN9210 to the D0 state and will reset the PM_MODE field to the D0 state. As noted above, the host is required to check the READY bit and verify that it is set before attempting any other reads or writes of the device.

Note 3.8 The host must only perform read accesses prior to the ready bit being set.

Once the READY bit is set, the LAN9210 is ready to resume normal operation. At this time the WUPS field can be cleared.

3.10.2.2 D2 Sleep

In this state, as shown in Table 3.10, all clocks to the MAC and host bus are disabled and the PHY is placed in a reduced power state. To enter this state, the EDPWRDOWN bit in register 17 of the PHY (Mode Control/Status register) must be set. This places the PHY in the Energy Detect mode. The PM_MODE bits in the PMT_CTRL register must then be set to 10b. Upon setting the PM_MODE bits, the LAN9210 will enter the D2 sleep state. The READY bit in PMT_CTRL is cleared when entering the D2 state.

Note 3.9 If carrier is present when this state is entered detection will occur immediately.

If properly enabled via the ED_EN and PME_EN bits, the LAN9210 will assert the PME hardware signal upon detection of a valid carrier. Upon detection, the WUPS field in PMT_CTRL will be set to a 01b.

Note 3.10 The PME interrupt status bit on the INT_STS register (PME_INT) is set regardless of the setting of PME_EN.

A write to the BYTE_TEST register, regardless of whether a carrier was detected, will return the LAN9210 to the D0 state and will reset the PM_MODE field to the D0 state. As noted above, the host is required to check the READY bit and verify that it is set before attempting any other reads or writes of the device. Before the LAN9210 is fully awake from this state the EDPWRDOWN bit in register 17 of the PHY must be cleared in order to wake the PHY. Do not attempt to clear the EDPWRDOWN bit until the READY bit is set. After clearing the EDPWRDOWN bit the LAN9210 is ready to resume normal operation. At this time the WUPS field can be cleared.

Table 3.10 Power Management States

Device BLOCK	D0 (NORMAL OPERATION)	D1 (WOL)	D2 (ENERGY DETECT)
PHY	Full ON	Full ON	Energy Detect Power-Down
MAC Power Management	Full ON	RX Power Mgmt. Block On	OFF
MAC and Host Interface	Full ON	OFF	OFF
Internal Clock	Full ON	Full ON	OFF

KEY
CLOCK ON
BLOCK DISABLED – CLOCK ON
FULL OFF

3.10.2.3 Power Management Event Indicators

Figure 3.18 is a simplified block diagram of the logic that controls the external PME, and internal pme_interrupt signals. The pme_interrupt signal is used to set the PME_INT status bit in the INT_STS register, which, if enabled, will generate a host interrupt upon detection of a power management event. The PME_INT status bit in INT_STS will remain set until the internal pme_interrupt signal is cleared by clearing the WUPS bits, or by clearing the corresponding WOL_EN or ED_EN bit. After clearing the internal pme_interrupt signal, the PME_INT status bit may be cleared by writing a '1' to this bit in the INT_STS register. It should be noted that the LAN9210 can generate a host interrupt regardless of the state of the PME_EN bit, or the external PME signal.

The external PME signal can be setup for pulsed, or static operation. When the PME_IND bit in the PMT_CTRL register is set to a '1', the external PME signal will be driven active for 50ms upon detection of a wake-up event. When the PME_IND bit is cleared, the PME signal will be driven continuously upon detection of a wake-up event. The PME signal is deactivated by clearing the WUPS bits, or by clearing the corresponding WOL_EN or ED_EN bit. The PME signal can also be deactivated by clearing the PME_EN bit.

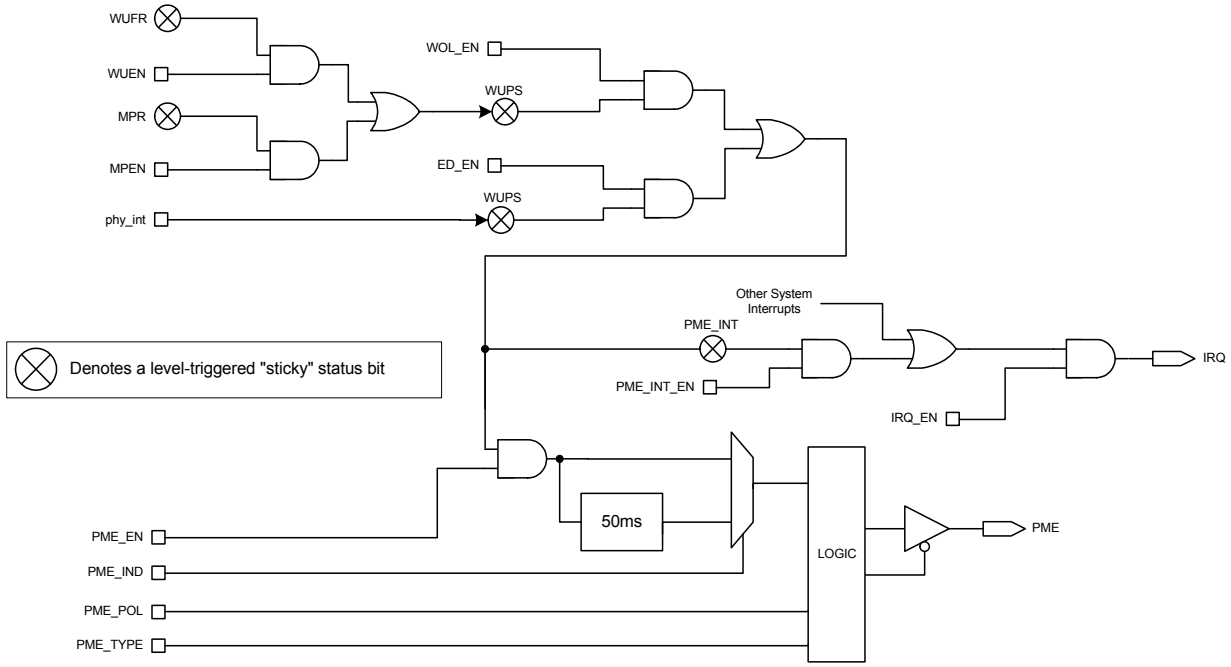


Figure 3.18 PME and PME_INT Signal Generation

3.10.3 Internal PHY Power-Down Modes

There are 2 power-down modes for the internal PHY:

3.10.3.1 General Power-Down

This power-down is controlled by register 0, bit 11. In this mode the internal PHY, except the management interface, is powered-down and stays in that condition as long as Phy register bit 0.11 is HIGH. When bit 0.11 is cleared, the PHY powers up and is automatically reset. Please refer to Section 5.5.1, "Basic Control Register," on page 117 for additional information on this register.

3.10.3.2 Energy Detect Power-Down

This power-down mode is activated by setting the Phy register bit 17.13 to 1. Please refer to Section 5.5.8, "Mode Control/Status," on page 121 for additional information on this register. In this mode when no energy is present on the line, the PHY is powered down, with the exception of the management interface, the SQUELCH circuit and the ENERGYON logic. The ENERGYON logic is used to detect the presence of valid energy from 100Base-TX, 10Base-T, or Auto-negotiation signals

In this mode, when the ENERGYON signal is low, the PHY is powered-down, and nothing is transmitted. When energy is received - link pulses or packets - the ENERGYON signal goes high, and the PHY powers-up. It automatically resets itself into the state it had prior to power-down, and asserts the INT7.1 bit of the register defined in Section 5.5.11, "Interrupt Source Flag," on page 124. If the ENERGYON interrupt is enabled, this event will cause an interrupt to the host. The first and possibly the second packet to activate ENERGYON may be lost. When 17.13 is low, energy detect power-down is disabled.

3.11 Detailed Reset Description

The LAN9210 has four reset sources:

- Hardware Reset Input Pin (nRESET)
- Soft Reset (SRST)
- PHY Soft Reset via PMT_CTRL bit 10 (PHY_RST)
- PHY Soft Reset via PHY Basic Control Register (PHY REG 0.15)

Table 3.11 shows the effect of the various reset sources on the LAN9210's circuitry.

Note: For proper operation, the LAN9210 must be reset on power-up via the hardware reset input (nRESET) or soft reset (SRST). To accomplish this, nRESET should be asserted for the minimum period of 30ms at power-up. Alternatively, a soft reset may be performed following power-up by setting the SRST bit of the HW_CFG register once the READY bit in the PMT_CTRL register has been set. Refer to Section 3.11.1, "Hardware Reset Input (nRESET)" and Section 3.11.3, "Soft Reset (SRST)" for additional information.

Table 3.11 Reset Sources and Affected Circuitry

RESET SOURCE	PLL	HBI Note 3.13	NASR REGISTERS Note 3.13	MIL	MAC	PHY Note 3.11	EEPROM MAC ADDR. RELOAD Note 3.12	CONFIG. STRAPS LATCHED
nRESET	X	X	X	X	X	X	X	X
SRST		X		X	X		X	
PHY_RST						X		
PHY REG 0.15						X		

Note 3.11 After any PHY reset, the application must wait until the “Link Status” bit in the PHY’s “Basic Status Register” (PHY Reg. 1.2) is set before attempting to transmit or receive data.

Note 3.12 After a power-up, nRESET or SRST, the LAN9210 will automatically check for the presence of an external EEPROM. After any of these resets the application must verify that the EPC Busy Bit (E2P_CMD, bit 31) is cleared before attempting to access the EEPROM, or change the function of the GPO/GPIO signals, or before modifying the ADDRH or ADDRL registers in the MAC.

Note 3.13 HBI - “Host Bus Interface”, NASR - Not affected by software reset.

APPLICATION NOTE: Under normal conditions, the READY bit in PMT_CTRL will be set (high -"1") after an internal reset (22ms). If the software driver polls this bit and it is not set within 100ms, then an error condition occurred.

3.11.1 Hardware Reset Input (nRESET)

A hardware reset will occur when the nRESET input signal is driven low. The READY bit in the PMT_CTRL register can be read from the host interface, and will read back a '0' until the hardware reset is complete. Upon completion of the hardware reset, the READY bit in PMT_CTRL is set high.

After the "READY" bit is set, the LAN9210 can be configured via its control registers. The nRESET signal is pulled-high internally by the LAN9210 and can be left unconnected if unused. If used, nRESET must be driven low for a minimum period as defined in Section 6.8, "Reset Timing," on page 135. If nRESET is unused, the device must be reset following power-up via a soft reset (SRST).

APPLICATION NOTE: Under normal conditions, the READY bit in PMT_CTRL will be set (high -"1") immediately. If the software driver polls this bit and it is not set within 100ms, then an error condition occurred.

3.11.2 Resume Reset Timing

After issuing a write to the BYTE_TEST register to wake the LAN9210 from a power-down state, the READY bit in PMT_CTRL will assert (set High) within 2ms.

APPLICATION NOTE: Under normal conditions, the READY bit in PMT_CTRL will be set (high -"1") within 2 ms. If the software driver polls this bit and it is not set within 100ms, then an error condition occurred.

3.11.3 Soft Reset (SRST)

Soft reset is initiated by writing a '1' to bit 0 of the HW_CFG register (SRST). This self-clearing bit will return to '0' after approximately 2 μs, at which time the Soft Reset is complete. Soft reset does not clear control register bits marked as NASR. Following power-on, a soft reset must not be performed until the READY bit in the PMT_CTRL register has been set.

APPLICATION NOTE: Under normal conditions, the READY bit in PMT_CTRL will be set (high -"1") immediately, (within 2μs). If the software driver polls this bit and it is not set within 100ms, then an error condition occurred.

3.11.4 PHY Reset Timing

The following sections specify the operation and time required for the internal PHY to become operational after various resets or when returning from the reduced power state.

3.11.4.1 PHY Soft Reset via PMT_CTRL bit 10 (PHY_RST)

The PHY soft reset is initiated by writing a '1' to bit 10 of the PMT_CTRL register (PHY_RST). This self-clearing bit will return to '0' after approximately 100 μs, at which time the PHY reset is complete.

3.11.4.2 PHY Soft Reset via PHY Basic Control Register (PHY Reg. 0.15)

The PHY Reg. 0.15 Soft Reset is initiated by writing a '1' to bit 15 of the PHY's Basic Control Register. This self-clearing bit will return to '0' at which time the PHY reset is complete.

3.12 TX Data Path Operation

Data is queued for transmission by writing it into the TX data FIFO. Each packet to be transmitted may be divided among multiple buffers. Each buffer starts with a two DWORD TX command (TX command

'A' and TX command 'B'). The TX command instructs the LAN9210 on the handling of the associated buffer. Packet boundaries are delineated using control bits within the TX command.

The host provides a 16-bit Packet Tag field in the TX command. The Packet Tag value is appended to the corresponding TX status DWORD. All Packet Tag fields must have the same value for all buffers in a given packet. If tags differ between buffers in the same packet the TXE error will be asserted. Any value may be chosen for a Packet Tag as long as all tags in the same Packet are identical. Packet Tags also provide a method of synchronization between transmitted packets and their associated status. Software can use unique Packet Tags to assist with validating matching status completions.

Note 3.14 The use of packet tags is not required by the hardware. This is a software LAN driver only application example for use of this field.

A Packet Length field in the TX command specifies the number of bytes in the associated packet. All Packet Length fields must have the same value for all buffers in a given packet. Hardware compares the Packet Length field and the actual amount of data received by the Ethernet controller. If the actual packet length count does not match the Packet Length field as defined in the TX command, the Transmitter Error (TXE) flag is asserted.

The LAN9210 can be programmed to start payload transmission of a buffer on a byte boundary by setting the "Data Start Offset" field in the TX command. The "Data Start Offset" field points to the actual start of the payload data within the first 8 DWORDs of the buffer. Data before the "Data Start Offset" pointer will be ignored. When a packet is split into multiple buffers, each successive buffer may begin on any arbitrary byte.

The LAN9210 can be programmed to strip padding from the end of a transmit packet in the event that the end of the packet does not align with the host burst boundary. This feature is necessary when the LAN9210 is operating in a system that always performs multi-word bursts. In such cases the LAN9210 must guarantee that it can accept data in multiples of the Burst length regardless of the actual packet length. When configured to do so, the LAN9210 will accept extra data at the end of the packet and will remove the extra padding before transmitting the packet. The LAN9210 automatically removes data up to the boundary specified in the Buffer End Alignment field specified in each TX command.

The host can instruct the LAN9210 to issue an interrupt when the buffer has been fully loaded into the TX FIFO contained in the LAN9210 and transmitted. This feature is enabled through the TX command 'Interrupt on Completion' field.

Upon completion of transmission, irrespective of success or failure, the status of the transmission is written to the TX status FIFO. TX status is available to the host and may be read using PIO operations. An interrupt can be optionally enabled by the host to indicate the availability of a programmable number TX status DWORDS.

Before writing the TX command and payload data to the TX FIFO, the host must check the available TX FIFO space by performing a PIO read of the TX_FIFO_INF register. The host must ensure that it does not overfill the TX FIFO or the TX Error (TXE) flag will be asserted.

The host proceeds to write the TX command by first writing TX command 'A', then TX command 'B'. After writing the command, the host can then move the payload data into the TX FIFO. TX status DWORD's are stored in the TX status FIFO to be read by the host at a later time upon completion of the data transmission onto the wire.

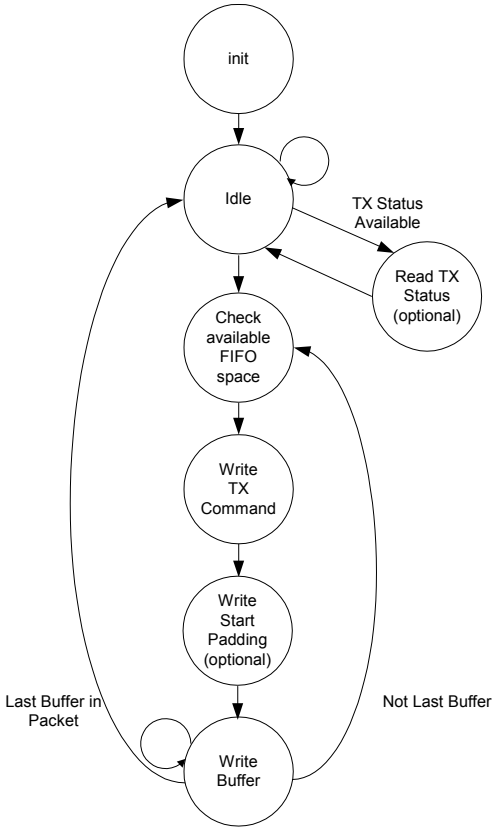


Figure 3.19 Simplified Host TX Flow Diagram

3.12.1 TX Buffer Format

TX buffers exist in the host's memory in a given format. The host writes a TX command word into the TX data buffer before moving the Ethernet packet data. The TX command A and command B are 32-bit values that are used by the LAN9210 in the handling and processing of the associated Ethernet packet data buffer. Buffer alignment, segmentation and other packet processing parameters are included in the command structure. The following diagram illustrates the buffer format.

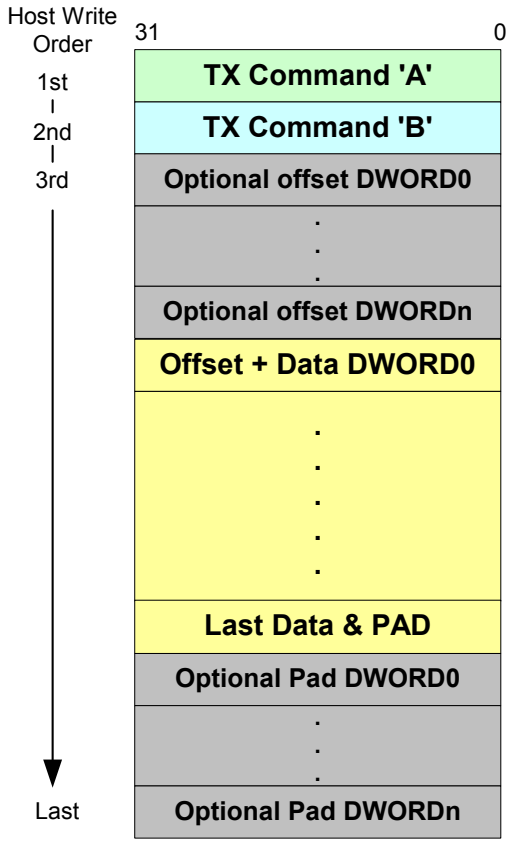


Figure 3.20 TX Buffer Format

Figure 3.20, "TX Buffer Format", shows the TX Buffer as it is written into the LAN9210. It should be noted that not all of the data shown in this diagram is actually stored in the TX data FIFO. This must be taken into account when calculating the actual TX data FIFO usage. Please refer to Section 3.12.5, "Calculating Actual TX Data FIFO Usage," on page 54 for a detailed explanation on calculating the actual TX data FIFO usage.

3.12.2 TX Command Format

The TX command instructs the TX FIFO controller on handling the subsequent buffer. The command precedes the data to be transmitted. The TX command is divided into two, 32-bit words; TX command ‘A’ and TX command ‘B’.

There is a 16-bit packet tag in the TX command ‘B’ command word. Packet tags may, if host software desires, be unique for each packet (i.e., an incrementing count). The value of the tag will be returned in the RX status word for the associated packet. The Packet tag can be used by host software to uniquely identify each status word as it is returned to the host.

Both TX command ‘A’ and TX command ‘B’ are required for each buffer in a given packet. TX command ‘B’ must be identical for every buffer in a given packet. If the TX command ‘B’ words do not match, the Ethernet controller will assert the Transmitter Error (TXE) flag.

TX COMMAND ‘A’

Table 3.12 TX Command 'A' Format

BITS	DESCRIPTION
31	**Interrupt on Completion.** When set, the TXDONE flag will be asserted when the current buffer has been fully loaded into the TX FIFO. This flag may be optionally mapped to a host interrupt.
30:26	**Reserved.** These bits are reserved. Always write zeros to this field to guarantee future compatibility.
25:24	**Buffer End Alignment.** This field specifies the alignment that must be maintained on the last data transfer of a buffer. The host will add extra DWORDs of data up to the alignment specified in the table below. The LAN9210 will remove the extra DWORDs. This mechanism can be used to maintain cache line alignment on host processors.
23:21	**Reserved.** These bits are reserved. Always write zeros to this field to guarantee future compatibility
20:16	**Data Start Offset (bytes).** This field specifies the offset of the first byte of TX data. The offset value can be anywhere from 0 bytes to 31 a Byte offset.
15:14	**Reserved.** These bits are reserved. Always write zeros to this field to guarantee future compatibility
13	**First Segment (FS).** When set, this bit indicates that the associated buffer is the first segment of the packet.
12	**Last Segment.** When set, this bit indicates that the associated buffer is the last segment of the packet
11	**Reserved.** These bits are reserved. Always write zeros to this field to guarantee future compatibility.
10:0	**Buffer Size (bytes).** This field indicates the number of bytes contained in the buffer following this command. This value, along with the Buffer End Alignment field, is read and checked by the LAN9210 and used to determine how many extra DWORD’s were added to the end of the Buffer. A running count is also maintained in the LAN9210 of the cumulative buffer sizes for a given packet. This cumulative value is compared against the Packet Length field in the TX command ‘B’ word and if they do not correlate, the TXE flag is set. **Note:** The buffer size specified does not include the buffer end alignment padding or data start offset added to a buffer.

[25]	[24]	End Alignment
0	0	4-byte alignment
0	1	16-byte alignment
1	0	32-byte alignment
1	1	Reserved

TX COMMAND 'B'

Table 3.13 TX Command 'B' Format

BITS	DESCRIPTION
31:16	**Packet Tag.** The host should write a unique packet identifier to this field. This identifier is added to the corresponding TX status word and can be used by the host to correlate TX status words with their corresponding packets. **Note:** The use of packet tags is not required by the hardware. This field can be used by the LAN software driver for any application. Packet Tags is one application example.
15	**Reserved.** This bit is reserved. Always write zeros to this bit to guarantee future compatibility.
14	**TX Checksum Enable (CK).** When this bit is set in conjunction with the first segment (FS) bit in TX Command 'A' and the TX checksum offload engine enable bit (TXCOE_EN) in the COE_CR—Checksum Offload Engine Control Register, the TX checksum offload engine (TXCOE) will calculate a L3 checksum for the associated frame.
13	**Add CRC Disable.** When set, the automatic addition of the CRC is disabled.
12	**Disable Ethernet Frame Padding.** When set, this bit prevents the automatic addition of padding to an Ethernet frame of less than 64 bytes. The CRC field is also added despite the state of the Add CRC Disable field.
11	**Reserved.** These bits are reserved. Always write zeros to this field to guarantee future compatibility.
10:0	**Packet Length (bytes).** This field indicates the total number of bytes in the current packet. This length does not include the offset or padding. If the Packet Length field does not match the actual number of bytes in the packet the Transmitter Error (TXE) flag will be set.

3.12.3 TX Data Format

The TX data section begins at the third DWORD in the TX buffer (after TX command 'A' and TX command 'B'). The location of the first byte of valid buffer data to be transmitted is specified in the "Data Start Offset" field of the TX command 'A' word. Table 3.14, "TX DATA Start Offset", shows the correlation between the setting of the LSB's in the "Data Start Offset" field and the byte location of the first valid data byte. Additionally, transmit buffer data can be offset by up to 7 additional DWORDS as indicated by the upper three MSB's (5:2) in the "Data Start Offset" field.

Table 3.14 TX DATA Start Offset

Data Start Offset [1:0]:	11	10	01	00
First TX Data Byte:	D[31:24]	D[23:16]	D[15:8]	D[7:0]

TX data is contiguous until the end of the buffer. The buffer may end on a byte boundary. Unused bytes at the end of the packet will not be sent to the MIL for transmission.

The Buffer End Alignment field in TX command 'A' specifies the alignment that must be maintained for the associated buffer. End alignment may be specified as 4-, 16-, or 32-byte. The host processor is responsible for adding the additional data to the end of the buffer. The hardware will automatically remove this extra data.

3.12.3.1 TX Buffer Fragmentation Rules

Transmit buffers must adhere to the following rules:

- Each buffer can start and end on any arbitrary byte alignment
- The first buffer of any transmit packet can be any length

- Middle buffers (i.e., those with First Segment = Last Segment = 0) must be greater than, or equal to 4 bytes in length
- The final buffer of any transmit packet can be any length

The MIL operates in store-and-forward mode and has specific rules with respect to fragmented packets. The total space consumed in the TX FIFO (MIL) must be limited to no more than 2KB - 3 DWORDs (2,036 bytes total). Any transmit packet that is so highly fragmented that it takes more space than this must be un-fragmented (by copying to a driver-supplied buffer) before the transmit packet can be sent to the LAN9210.

One approach to determine whether a packet is too fragmented is to calculate the actual amount of space that it will consume, and check it against 2,036 bytes. Another approach is to check the number of buffers against a worst-case limit of 86 (see explanation below).

3.12.3.2 Calculating Worst-Case TX FIFO (MIL) Usage

The actual space consumed by a buffer in the MIL TX FIFO consists only of any partial DWORD offsets in the first/last DWORD of the buffer, plus all of the whole DWORDs in between. Any whole DWORD offsets and/or alignments are stripped off before the buffer is loaded into the TX Data FIFO, and TX command words are stripped off before the buffer is written to the MIL TX FIFO, so none of those DWORDs count as space consumed. The worst-case overhead for a TX buffer is 6 bytes, which assumes that it started on the high byte of a DWORD and ended on the low byte of a DWORD. A TX packet consisting of 86 such fragments would have an overhead of 516 bytes (6 * 86) which, when added to a 1514-byte max-size transmit packet (1516 bytes, rounded up to the next whole DWORD), would give a total space consumption of 2,032 bytes, leaving 4 bytes to spare; this is the basis for the "86 fragment" rule mentioned above.

3.12.4 TX Status Format

TX status is passed to the host CPU through a separate FIFO mechanism. A status word is returned for each packet transmitted. Data transmission is suspended if the TX status FIFO becomes full. Data transmission will resume when the host reads the TX status and there is room in the FIFO for more “TX Status” data.

The host can optionally choose to not read the TX status. The host can optionally ignore the TX status by setting the “TX Status Discard Allow Overrun Enable” (TXSAO) bit in the TX Configuration Register (TX_CFG). If this option is chosen TX status will not be written to the FIFO. Setting this bit high allows the transmitter to continue operation with a full TX status FIFO. In this mode the status information is still available in the TX status FIFO, and TX status interrupts still function. In the case of an overrun, the TXSUSED counter will stay at zero and no further TX status will be written to the TX status FIFO until the host frees space by reading TX status. If TXSAO is enabled, a TXE error will not be generated if the TX status FIFO overruns. In this mode the host is responsible for re-synchronizing TX status in the case of an overrun.

BITS	DESCRIPTION
31:16	**Packet TAG.** Unique identifier written by the host into the Packet Tag field of the TX command ‘B’ word. This field can be used by the host to correlate TX status words with the associated TX packets.
15	**Error Status (ES).** When set, this bit indicates that the Ethernet controller has reported an error. This bit is the logical OR of bits 11, 10, 9, 8, 2, 1 in this status word.
14:12	**Reserved.** These bits are reserved. Always write zeros to this field to guarantee future compatibility.
11	**Loss of Carrier.** When set, this bit indicates the loss of carrier during transmission.

BITS	DESCRIPTION
10	**No Carrier.** When set, this bit indicates that the carrier signal from the transceiver was not present during transmission. **Note:** During 10/100 Mbps full-duplex transmission, the value of this bit is invalid and should be ignored.
9	**Late Collision.** When set, indicates that the packet transmission was aborted after the collision window of 64 bytes.
8	**Excessive Collisions.** When set, this bit indicates that the transmission was aborted after 16 collisions while attempting to transmit the current packet.
7	**Reserved.** This bit is reserved. Always write zeros to this field to guarantee future compatibility.
6:3	**Collision Count.** This counter indicates the number of collisions that occurred before the packet was transmitted. It is not valid when excessive collisions (bit 8) is also set.
2	**Excessive Deferral.** If the deferred bit is set in the control register, the setting of the excessive deferral bit indicates that the transmission has ended because of a deferral of over 24288 bit times during transmission.
1	**Reserved.** This bit is reserved. Always write zero to this bit to guarantee future compatibility.
0	**Deferred.** When set, this bit indicates that the current packet transmission was deferred.

3.12.5 Calculating Actual TX Data FIFO Usage

The following rules are used to calculate the actual TX data FIFO space consumed by a TX Packet:

- TX command 'A' is stored in the TX data FIFO for every TX buffer.
- TX command 'B' is written into the TX data FIFO when the First Segment (FS) bit is set in TX command 'A.'
- When TX checksum is enabled, the 4-byte TX checksum preamble is written into TX Data FIFO.
- Any DWORD-long data added as part of the “Data Start Offset” is removed from each buffer before the data is written to the TX data FIFO. Any data that is less than 1 DWORD is passed to the TX data FIFO.
- Payload from each buffer within a Packet is written into the TX data FIFO.
- Any DWORD-long data added as part of the End Padding is removed from each buffer before the data is written to the TX data FIFO. Any end padding that is less than 1 DWORD is passed to the TX data FIFO.

3.12.6 Transmit Examples

3.12.6.1 TX Example 1

In this example a single, 111-Byte Ethernet packet will be transmitted. This packet is divided into three buffers. The three buffers are as follows:

Buffer 0:

- 7-Byte “Data Start Offset”
- 79-Bytes of payload data
- 16-Byte “Buffer End Alignment”

Buffer 1:

- 0-Byte “Data Start Offset”
- 15-Bytes of payload data
- 16-Byte “Buffer End Alignment”

Buffer 2:

- 10-Byte “Data Start Offset”
- 17-Bytes of payload data
- 16-Byte “Buffer End Alignment”

Figure 3.21, "TX Example 1" illustrates the TX command structure for this example, and also shows how data is passed to the TX data FIFO.

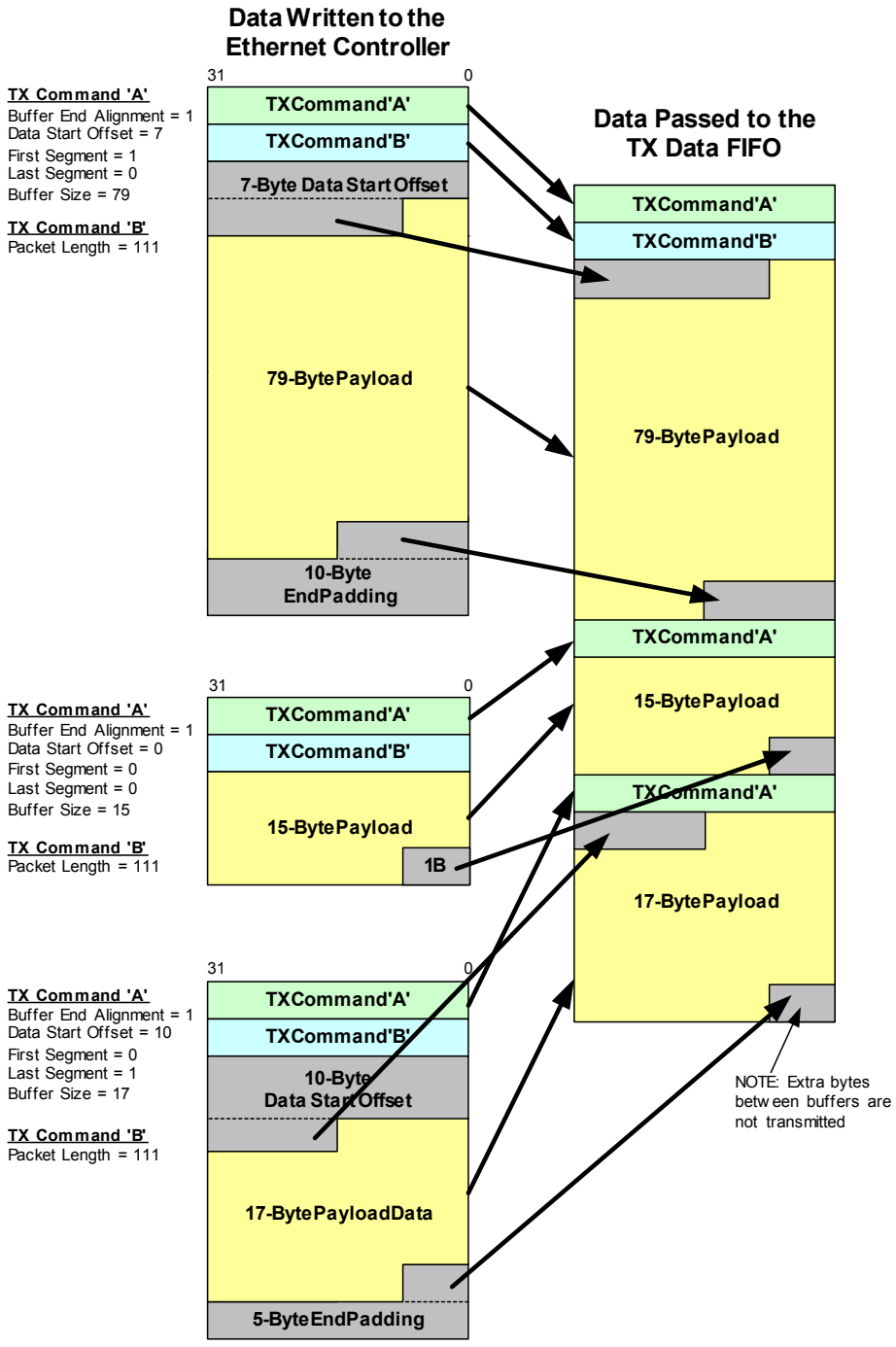


Figure 3.21 TX Example 1

3.12.6.2 TX Example 2

In this example, a single 183-Byte Ethernet packet will be transmitted. This packet is in a single buffer as follows:

- 2-Byte “Data Start Offset”
- 183-Bytes of payload data
- 4-Byte “Buffer End Alignment”

Figure 3.22, "TX Example 2" illustrates the TX command structure for this example, and also shows how data is passed to the TX data FIFO. Note that the packet resides in a single TX Buffer, therefore both the FS and LS bits are set in TX command ‘A’.

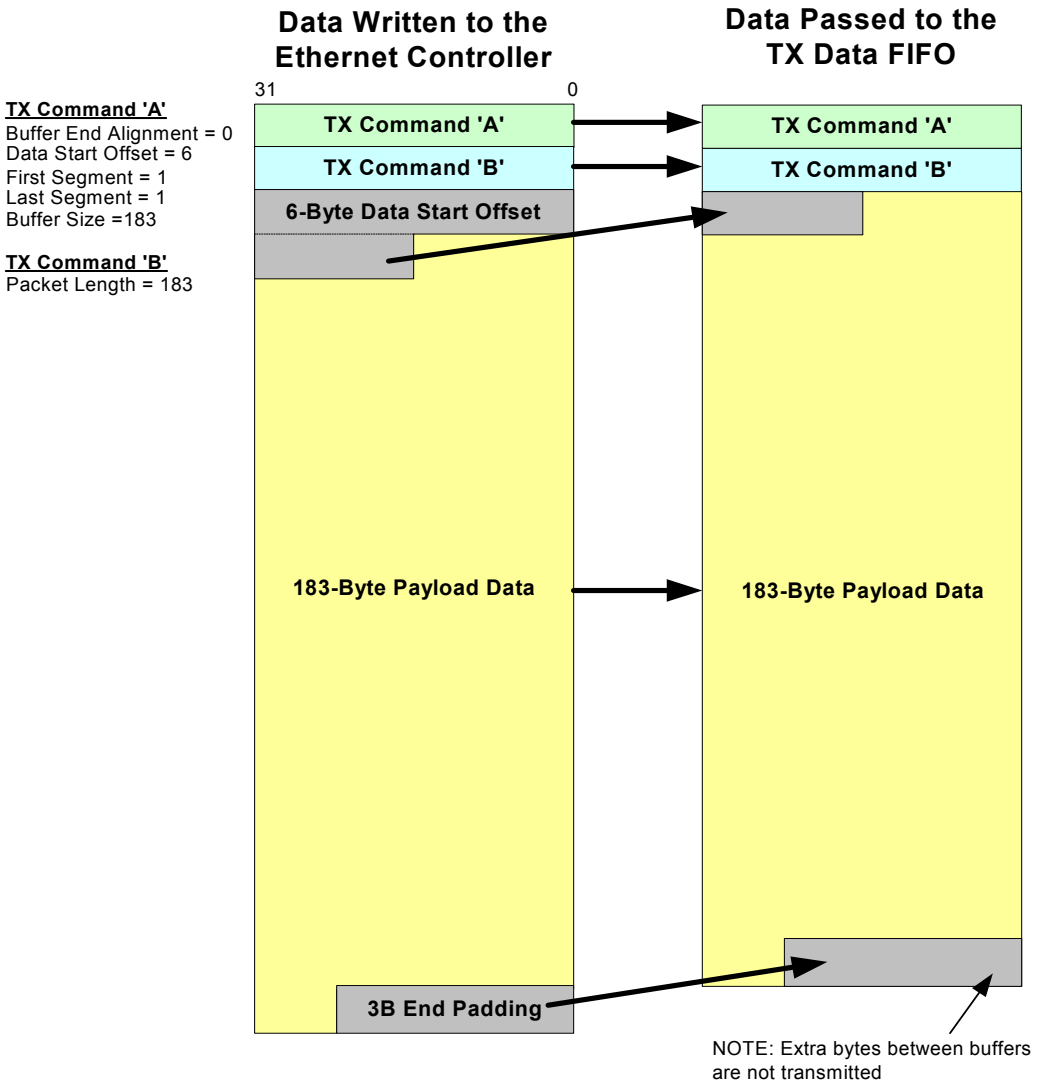


Figure 3.22 TX Example 2

3.12.6.3 TX Example 3

In this example a single, 111-Byte Ethernet packet will be transmitted with a TX checksum. This packet is divided into four buffers. The four buffers are as follows:

Buffer 0:

- 4-Byte “Data Start Offset”
- 4-Byte Checksum Preamble
- 16-Byte “Buffer End Alignment”

Buffer 1:

- 7-Byte “Data Start Offset”
- 79-Bytes of payload data
- 16-Byte “Buffer End Alignment”

Buffer 2:

- 0-Byte “Data Start Offset”
- 15-Bytes of payload data
- 16-Byte “Buffer End Alignment”

Buffer 3:

- 10-Byte “Data Start Offset”
- 17-Bytes of payload data
- 16-Byte “Buffer End Alignment”

Figure 3.21, "TX Example 1" illustrates the TX command structure for this example, and also shows how data is passed to the TX data FIFO.

Note: In order to perform a TX checksum calculation on the associated packet, bit 14 (CK) of the TX Command ‘B’ must be set in conjunction with bit 13 (FS) of TX Command ‘A’ and bit 16 (TXCOE_EN) of the COE_CR register. For more information, refer to Section 3.6.2, "Transmit Checksum Offload Engine (TXCOE)".

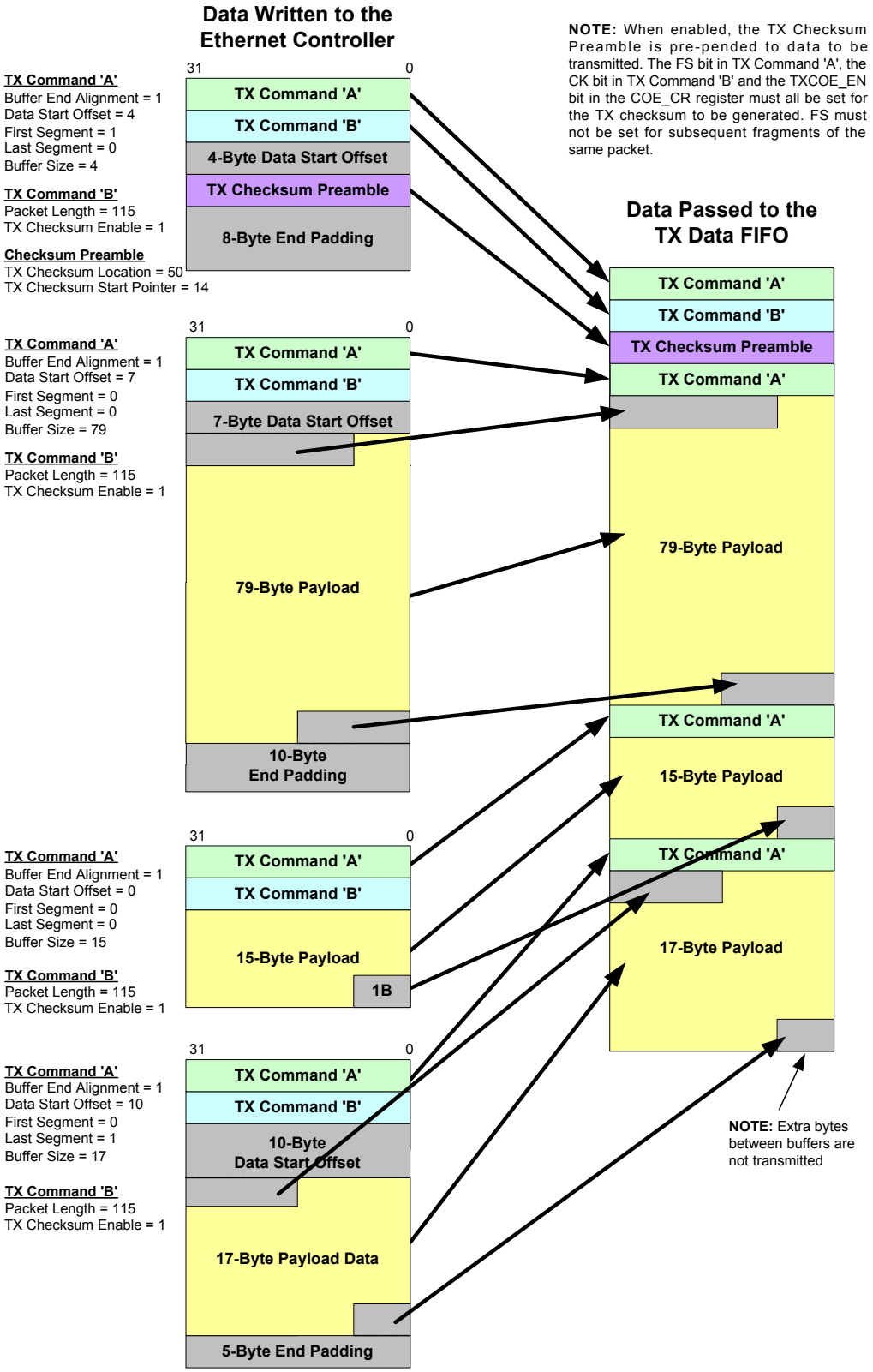


Figure 3.23 TX Example 3

3.12.7 Transmitter Errors

If the Transmitter Error (TXE) flag is asserted for any reason, the transmitter will continue operation. TX Error (TXE) will be asserted under the following conditions:

- If the actual packet length count does not match the Packet Length field as defined in the TX command.
- Both TX command 'A' and TX command 'B' are required for each buffer in a given packet. TX command 'B' must be identical for every buffer in a given packet. If the TX command 'B' words do not match, the Ethernet controller will assert the Transmitter Error (TXE) flag.
- Host overrun of the TX data FIFO.
- Overrun of the TX status FIFO (unless TXSAO is enabled)

3.12.8 Stopping and Starting the Transmitter

To halt the transmitter, the host must set the TX_STOP bit in the TX_CFG register. The transmitter will finish sending the current frame (if there is a frame transmission in progress). When the transmitter has received the TX status for this frame, it will clear the TX_STOP and TX_ON bits, and will pulse the TXSTOP_INT.

Once stopped, the host can optionally clear the TX status and TX data FIFOs. The host must re-enable the transmitter by setting the TX_ON bit. If the there are frames pending in the TX data FIFO (i.e., TX data FIFO was not purged), the transmission will resume with this data.

3.13 RX Data Path Operation

When an Ethernet Packet is received, the MIL first begins to transfer the RX data. This data is loaded into the RX data FIFO. The RX data FIFO pointers are updated as data is written into the FIFO.

The last transfer from the MIL is the RX status word. The LAN9210 implements a separate FIFO for the RX status words. The total available RX data and status queued in the RX FIFO can be read from the RX_FIFO_INF register. The host may read any number of available RX status words before reading the RX data FIFO.

The host must use caution when reading the RX data and status. The host must never read more data than what is available in the FIFOs. If this is attempted an underrun condition will occur. If this error occurs, the Ethernet controller will assert the Receiver Error (RXE) interrupt. If an underrun condition occurs, a soft reset is required to regain host synchronization.

A configurable beginning offset is supported in the LAN9210. The RX data Offset field in the RX_CFG register controls the number of bytes that the beginning of the RX data buffer is shifted. The host can set an offset from 0-31 bytes. The offset may be changed in between RX packets, but it must not be changed during an RX packet read.

The LAN9210 can be programmed to add padding at the end of a receive packet in the event that the end of the packet does not align with the host burst boundary. This feature is necessary when the LAN9210 is operating in a system that always performs multi-DWORD bursts. In such cases the LAN9210 must guarantee that it can transfer data in multiples of the Burst length regardless of the actual packet length. When configured to do so, the LAN9210 will add extra data at the end of the packet to allow the host to perform the necessary number of reads so that the Burst length is not cut short. Once a packet has been padded by the H/W, it is the responsibility of the host to interrogate the Packet length field in the RX status and determine how much padding to discard at the end of the Packet.

It is possible to read multiple packets out of the RX data FIFO in one continuous stream. It should be noted that the programmed Offset and Padding will be added to each individual packet in the stream, since packet boundaries are maintained.

3.13.1 RX Slave PIO Operation

Using PIO mode, the host can either implement a polling or interrupt scheme to empty the received packet out of the RX data FIFO. The host will remain in the idle state until it receives an indication (interrupt or polling) that data is available in the RX data FIFO. The host will then read the RX status FIFO to get the packet status, which will contain the packet length and any other status information. The host should perform the proper number of reads, as indicated by the packet length *plus* the start offset *and* the amount of optional padding added to the end of the frame, from the RX data FIFO.

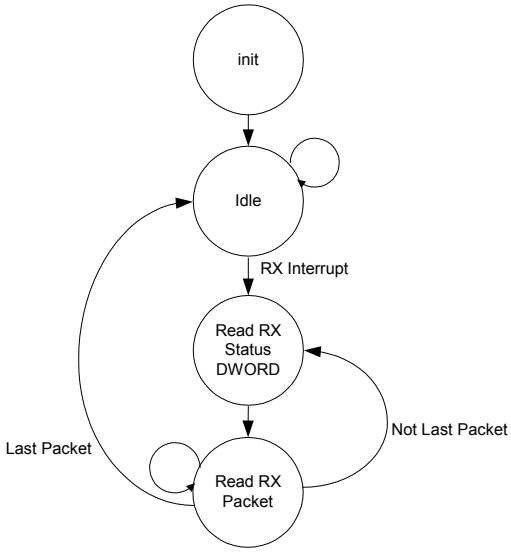


Figure 3.24 Host Receive Routine Using Interrupts

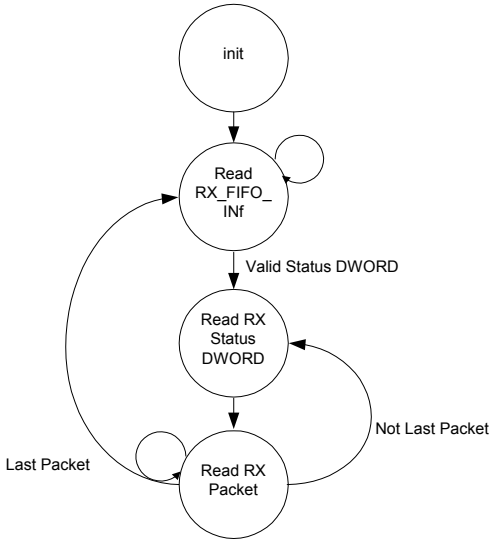


Figure 3.25 Host Receive Routine with Polling

3.13.1.1 Receive Data FIFO Fast Forward

The RX data path implements an automatic data discard function. Using the RX data FIFO Fast Forward bit (RX_FFWD) in the RX_DP_CTRL register, the host can instruct the LAN9210 to skip the packet at the head of the RX data FIFO. The RX data FIFO pointers are automatically incremented to the beginning of the next RX packet.

When performing a fast-forward, there must be at least 4 DWORDs of data in the RX data FIFO for the packet being discarded. For less than 4 DWORDs do not use RX_FFWD. In this case data must be read from the RX data FIFO and discarded using standard PIO read operations.

After initiating a fast-forward operation, do not perform any reads of the RX data FIFO, RX status FIFO, or the TX status FIFO until the RX_FFWD bit is cleared. Other resources can be accessed during this time (i.e., any registers and/or the TX data FIFO). After the fast-forward operation has completed and the RX_FFWD bit has been cleared, a wait time restriction must be observed before reading the TX or RX status FIFO's, as specified in Section 6.1.2, "Special Restrictions on Back-to-Back Read Cycles," on page 128. Also note that the RX_FFWD will only fast-forward the RX data FIFO, not the RX status FIFO.

The receiver does not have to be stopped to perform a fast-forward operation.

3.13.1.2 Force Receiver Discard (Receiver Dump)

In addition to the Receive data Fast Forward feature, LAN9210 also implements a receiver "dump" feature. This feature allows the host processor to flush the entire contents of the RX data and RX status FIFOs. When activated, the read and write pointers for the RX data and status FIFOs will be returned to their reset state. To perform a receiver dump, the LAN9210 receiver must be halted. Once the receiver stop completion is confirmed, the RX_DUMP bit can be set in the RX_CFG register. The RX_DUMP bit is cleared when the dump is complete. For more information on stopping the receiver, please refer to Section 3.13.4, "Stopping and Starting the Receiver," on page 65. For more information on the RX_DUMP bit, please refer to Section 5.3.7, "RX_CFG—Receive Configuration Register," on page 85.

3.13.2 RX Packet Format

The RX status words can be read from the RX status FIFO port, while the RX data packets can be read from the RX data FIFO. RX data packets are formatted in a specific manner before the host can read them as shown in Figure 3.26. It is assumed that the host has previously read the associated status word from the RX status FIFO, to ascertain the data size and any error conditions.

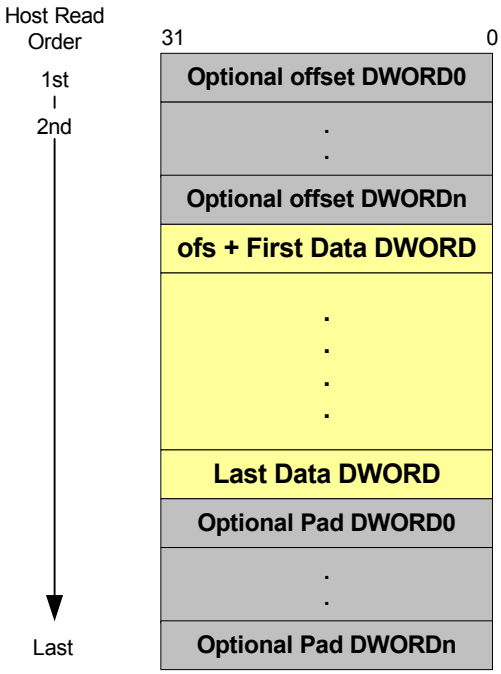


Figure 3.26 RX Packet Format

Figure 3.27 shows the RX packet format when the RX checksum is enabled. The RX checksum data appended to the data payload is treated just as an additional 4-bytes within the RX Data FIFO. The RX checksum is enabled by setting the RXCOE_EN bit in the COE_CR—Checksum Offload Engine Control Register. For more information on the RX checksum, refer to Section 3.6.1, "Receive Checksum Offload Engine (RXCOE)".

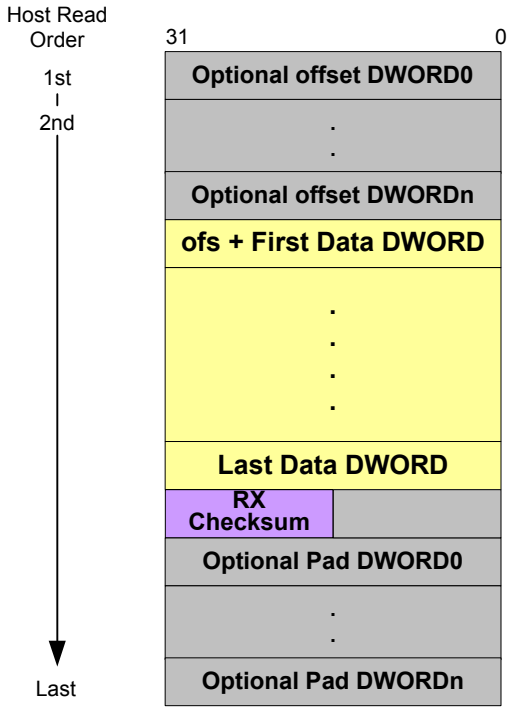


Figure 3.27 RX Packet Format with RX Checksum

3.13.3 RX Status Format

BITS	DESCRIPTION
31	**Reserved.** This bit is reserved. Reads 0.
30	**Filtering Fail.** When set, this bit indicates that the associated frame failed the address recognizing filtering.
29:16	**Packet Length.** The size, in bytes, of the corresponding received frame.
15	**Error Status (ES).** When set this bit indicates that the MIL has reported an error. This bit is the Internal logical “or” of bits 11,7,6 and 1.
14	**Reserved.** These bits are reserved. Reads 0.
13	**Broadcast Frame.** When set, this bit indicates that the received frame has a Broadcast address.
12	**Length Error (LE).** When set, this bit indicates that the actual length does not match with the length/type field of the received frame.
11	**Runt Frame.** When set, this bit indicates that frame was prematurely terminated before the collision window (64 bytes). Runt frames are passed on to the host only if the Pass Bad Frames bit MAC_CR Bit [16] is set.
10	**Multicast Frame.** When set, this bit indicates that the received frame has a Multicast address.
9:8	**Reserved.** These bits are reserved. Reads 0.

BITS	DESCRIPTION
7	**Frame Too Long.** When set, this bit indicates that the frame length exceeds the maximum Ethernet specification of 1518 bytes. This is only a frame too long indication and will not cause the frame reception to be truncated.
6	**Collision Seen.** When set, this bit indicates that the frame has seen a collision after the collision window. This indicates that a late collision has occurred.
5	**Frame Type.** When set, this bit indicates that the frame is an Ethernet-type frame (Length/Type field in the frame is greater than 1500). When reset, it indicates the incoming frame was an 802.3 type frame. This bit is not set for Runt frames less than 14 bytes.
4	**Receive Watchdog time-out.** When set, this bit indicates that the incoming frame is greater than 2048 bytes through 2560 bytes, therefore expiring the Receive Watchdog Timer.
3	**MII Error.** When set, this bit indicates that a receive error (RX_ER asserted) was detected during frame reception.
2	**Dribbling Bit.** When set, this bit indicates that the frame contained a non-integer multiple of 8 bits. This error is reported only if the number of dribbling bits in the last byte is 4 in the MII operating mode, or at least 3 in the 10 Mbps operating mode. This bit will not be set when the collision seen bit[6] is set. If set and the CRC error bit[1] is cleared, then the packet is considered to be valid.
1	**CRC Error.** When set, this bit indicates that a CRC error was detected. This bit is also set when the RX_ER pin is asserted during the reception of a frame even though the CRC may be correct. This bit is not valid if the received frame is a Runt frame, or a late collision was detected or when the Watchdog Time-out occurs.
0	**Reserved.** These bits are reserved. Reads 0

3.13.4 Stopping and Starting the Receiver

To stop the receiver, the host must clear the RXEN bit in the MAC Control Register. When the receiver is halted, the RXSTOP_INT will be pulsed. Once stopped, the host can optionally clear the RX status and RX data FIFOs. The host must re-enable the receiver by setting the RXEN bit.

3.13.5 Receiver Errors

If the Receiver Error (RXE) flag is asserted for any reason, the receiver will continue operation. RX Error (RXE) will be asserted under the following conditions:

- A host underrun of RX data FIFO
- A host underrun of the RX status FIFO
- An overrun of the RX status FIFO

It is the duty of the host to identify and resolve any error conditions.

Chapter 4 Internal Ethernet PHY

4.1 Top Level Functional Description

Functionally, the internal PHY can be divided into the following sections:

- 100Base-TX transmit and receive
- 10Base-T transmit and receive
- Internal MII interface to the Ethernet Media Access Controller
- Auto-negotiation to automatically determine the best speed and duplex possible
- Management Control to read status registers and write control registers

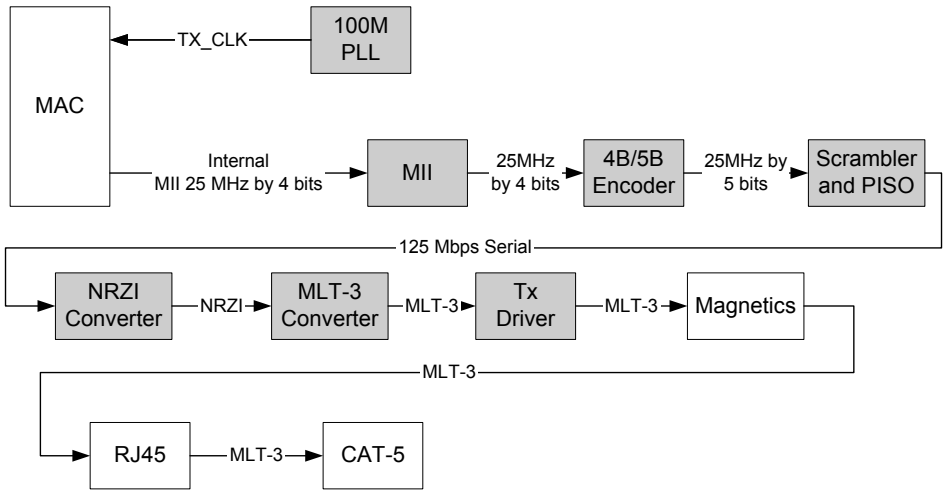


Figure 4.1 100Base-TX Data Path

4.2 100Base-TX Transmit

The data path of the 100Base-TX is shown in Figure 4.1. Each major block is explained below.

4.2.1 4B/5B Encoding

The transmit data passes from the MII block to the 4B/5B encoder. This block encodes the data from 4-bit nibbles to 5-bit symbols (known as “code-groups”) according to Table 4.1. Each 4-bit data-nibble is mapped to 16 of the 32 possible code-groups. The remaining 16 code-groups are either used for control information or are not valid.

The first 16 code-groups are referred to by the hexadecimal values of their corresponding data nibbles, 0 through F. The remaining code-groups are given letter designations with slashes on either side. For example, an IDLE code-group is /I/, a transmit error code-group is /H/, etc.

The encoding process may be bypassed by clearing bit 6 of register 31. When the encoding is bypassed the 5th transmit data bit is equivalent to TX_ER.

Table 4.1 4B/5B Code Table

CODE GROUP	SYM	RECEIVER INTERPRETATION			TRANSMITTER INTERPRETATION		
11110	0	0	0000	DATA	0	0000	DATA
01001	1	1	0001		1	0001	
10100	2	2	0010		2	0010	
10101	3	3	0011		3	0011	
01010	4	4	0100		4	0100	
01011	5	5	0101		5	0101	
01110	6	6	0110		6	0110	
01111	7	7	0111		7	0111	
10010	8	8	1000		8	1000	
10011	9	9	1001		9	1001	
10110	A	A	1010		A	1010	
10111	B	B	1011		B	1011	
11010	C	C	1100		C	1100	
11011	D	D	1101		D	1101	
11100	E	E	1110		E	1110	
11101	F	F	1111		F	1111	
11111	I	IDLE			Sent after /T/R until TX_EN		
11000	J	First nibble of SSD, translated to “0101” following IDLE, else RX_ER			Sent for rising TX_EN		
10001	K	Second nibble of SSD, translated to “0101” following J, else RX_ER			Sent for rising TX_EN		
01101	T	First nibble of ESD, causes de-assertion of CRS if followed by /R/, else assertion of RX_ER			Sent for falling TX_EN		
00111	R	Second nibble of ESD, causes deassertion of CRS if following /T/, else assertion of RX_ER			Sent for falling TX_EN		
00100	H	Transmit Error Symbol			Sent for rising TX_ER		
00110	V	INVALID, RX_ER if during RX_DV			INVALID		
11001	V	INVALID, RX_ER if during RX_DV			INVALID		
00000	V	INVALID, RX_ER if during RX_DV			INVALID		
00001	V	INVALID, RX_ER if during RX_DV			INVALID		
00010	V	INVALID, RX_ER if during RX_DV			INVALID		
00011	V	INVALID, RX_ER if during RX_DV			INVALID		

Table 4.1 4B/5B Code Table (continued)

CODE GROUP	SYM	RECEIVER INTERPRETATION	TRANSMITTER INTERPRETATION
00101	V	INVALID, RX_ER if during RX_DV	INVALID
01000	V	INVALID, RX_ER if during RX_DV	INVALID
01100	V	INVALID, RX_ER if during RX_DV	INVALID
10000	V	INVALID, RX_ER if during RX_DV	INVALID

4.2.2 Scrambling

Repeated data patterns (especially the IDLE code-group) can have power spectral densities with large narrow-band peaks. Scrambling the data helps eliminate these peaks and spread the signal power more uniformly over the entire channel bandwidth. This uniform spectral density is required by FCC regulations to prevent excessive EMI from being radiated by the physical wiring.

The scrambler also performs the Parallel In Serial Out conversion (PISO) of the data.

4.2.3 NRZI and MLT3 Encoding

The scrambler block passes the 5-bit wide parallel data to the NRZI converter where it becomes a serial 125MHz NRZI data stream. The NRZI is encoded to MLT-3. MLT3 is a tri-level code where a change in the logic level represents a code bit “1” and the logic output remaining at the same level represents a code bit “0”.

4.2.4 100M Transmit Driver

The MLT3 data is then passed to the analog transmitter, which launches the differential MLT-3 signal, on outputs TXP and TXN, to the twisted pair media via a 1:1 ratio isolation transformer. The 10Base-T and 100Base-TX signals pass through the same transformer so that common “magnetics” can be used for both. The transmitter drives into the 100Ω impedance of the CAT-5 cable. Cable termination and impedance matching require external components.

4.2.5 100M Phase Lock Loop (PLL)

The 100M PLL locks onto reference clock and generates the 125MHz clock used to drive the 125 MHz logic and the 100Base-Tx Transmitter.

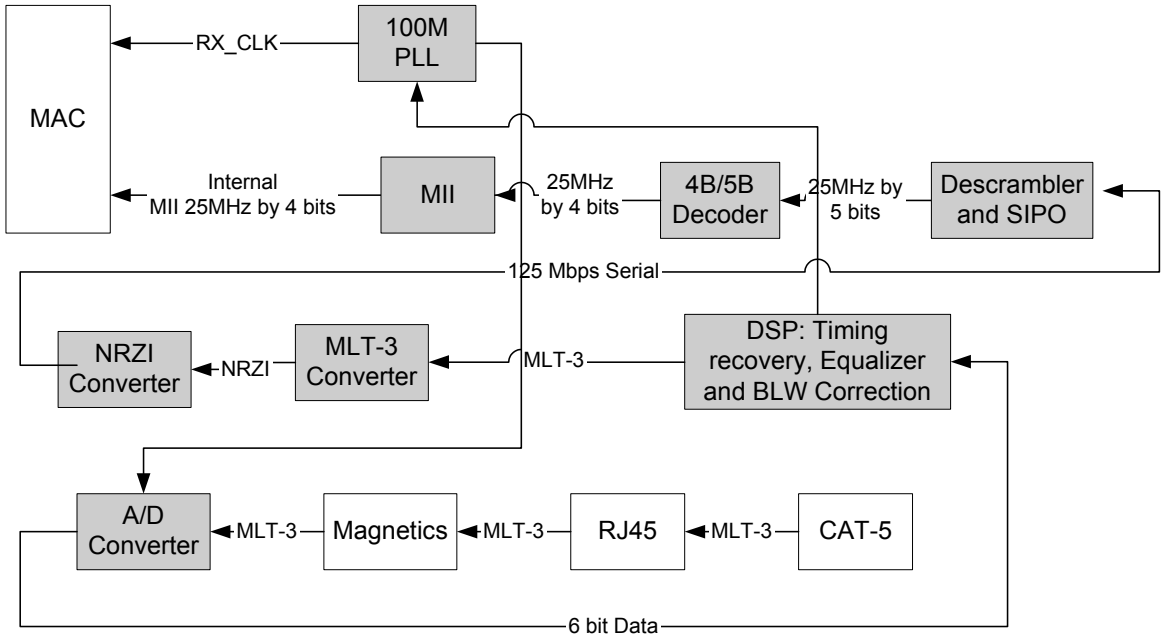


Figure 4.2 Receive Data Path

4.3 100Base-TX Receive

The receive data path is shown in Figure 4.2. Detailed descriptions are given below.

4.3.1 100M Receive Input

The MLT-3 from the cable is fed into the PHY (on inputs RXP and RXN) via a 1:1 ratio transformer. The ADC samples the incoming differential signal at a rate of 125M samples per second. Using a 64-level quanitizer it generates 6 digital bits to represent each sample. The DSP adjusts the gain of the ADC according to the observed signal levels such that the full dynamic range of the ADC can be used.

4.3.2 Equalizer, Baseline Wander Correction and Clock and Data Recovery

The 6 bits from the ADC are fed into the DSP block. The equalizer in the DSP section compensates for phase and amplitude distortion caused by the physical channel consisting of magnetics, connectors, and CAT- 5 cable. The equalizer can restore the signal for any good-quality CAT-5 cable between 1m and 150m.

If the DC content of the signal is such that the low-frequency components fall below the low frequency pole of the isolation transformer, then the droop characteristics of the transformer will become significant and Baseline Wander (BLW) on the received signal will result. To prevent corruption of the received data, the PHY corrects for BLW and can receive the ANSI X3.263-1995 FDDI TP-PMD defined “killer packet” with no bit errors.

The 100M PLL generates multiple phases of the 125MHz clock. A multiplexer, controlled by the timing unit of the DSP, selects the optimum phase for sampling the data. This is used as the received recovered clock. This clock is used to extract the serial data from the received signal.

4.3.3 NRZI and MLT-3 Decoding

The DSP generates the MLT-3 recovered levels that are fed to the MLT-3 converter. The MLT-3 is then converted to an NRZI data stream.

4.3.4 Descrambling

The descrambler performs an inverse function to the scrambler in the transmitter and also performs the Serial In Parallel Out (SIPO) conversion of the data.

During reception of IDLE (/I/) symbols. the descrambler synchronizes its descrambler key to the incoming stream. Once synchronization is achieved, the descrambler locks on this key and is able to descramble incoming data.

Special logic in the descrambler ensures synchronization with the remote PHY by searching for IDLE symbols within a window of 4000 bytes (40us). This window ensures that a maximum packet size of 1514 bytes, allowed by the IEEE 802.3 standard, can be received with no interference. If no IDLE-symbols are detected within this time-period, receive operation is aborted and the descrambler re-starts the synchronization process.

The descrambler can be bypassed by setting bit 0 of register 31.

4.3.5 Alignment

The de-scrambled signal is then aligned into 5-bit code-groups by recognizing the /J/K/ Start-of-Stream Delimiter (SSD) pair at the start of a packet. Once the code-word alignment is determined, it is stored and utilized until the next start of frame.

4.3.6 5B/4B Decoding

The 5-bit code-groups are translated into 4-bit data nibbles according to the 4B/5B table. The SSD, /J/K/, is translated to “0101 0101” as the first 2 nibbles of the MAC preamble. Reception of the SSD causes the PHY to assert the internal RX_DV signal, indicating that valid data is available on the Internal RXD bus. Successive valid code-groups are translated to data nibbles. Reception of either the End of Stream Delimiter (ESD) consisting of the /T/R/ symbols, or at least two /I/ symbols causes the PHY to de-assert the internal carrier sense and RX_DV.

These symbols are not translated into data.

4.4 10Base-T Transmit

Data to be transmitted comes from the MAC layer controller. The 10Base-T transmitter receives 4-bit nibbles from the MII at a rate of 2.5MHz and converts them to a 10Mbps serial data stream. The data stream is then Manchester-encoded and sent to the analog transmitter, which drives a signal onto the twisted pair via the external magnetics.

The 10M transmitter uses the following blocks:

- MII (digital)
- TX 10M (digital)
- 10M Transmitter (analog)
- 10M PLL (analog)

4.4.1 10M Transmit Data across the internal MII bus

The MAC controller drives the transmit data onto the internal TXD BUS. When the controller has driven TX_EN high to indicate valid data, the data is latched by the MII block on the rising edge of TX_CLK. The data is in the form of 4-bit wide 2.5MHz data.

4.4.2 Manchester Encoding

The 4-bit wide data is sent to the TX10M block. The nibbles are converted to a 10Mbps serial NRZI data stream. The 10M PLL locks onto the external clock or internal oscillator and produces a 20MHz

clock. This is used to Manchester encode the NRZ data stream. When no data is being transmitted (TX_EN is low), the TX10M block outputs Normal Link Pulses (NLPs) to maintain communications with the remote link partner.

4.4.3 10M Transmit Drivers

The Manchester encoded data is sent to the analog transmitter where it is shaped and filtered before being driven out as a differential signal across the TXP and TXN outputs.

4.5 10Base-T Receive

The 10Base-T receiver gets the Manchester- encoded analog signal from the cable via the magnetics. It recovers the receive clock from the signal and uses this clock to recover the NRZI data stream. This 10M serial data is converted to 4-bit data nibbles which are passed to the controller across the MII at a rate of 2.5MHz.

This 10M receiver uses the following blocks:

- Filter and SQUELCH (analog)
- 10M PLL (analog)
- RX 10M (digital)
- MII (digital)

4.5.1 10M Receive Input and Squelch

The Manchester signal from the cable is fed into the PHY (on inputs RXP and RXN) via 1:1 ratio magnetics. It is first filtered to reduce any out-of-band noise. It then passes through a SQUELCH circuit. The SQUELCH is a set of amplitude and timing comparators that normally reject differential voltage levels below 300mV and detect and recognize differential voltages above 585mV.

4.5.2 Manchester Decoding

The output of the SQUELCH goes to the RX10M block where it is validated as Manchester encoded data. The polarity of the signal is also checked. If the polarity is reversed (local RXP is connected to RXN of the remote partner and vice versa), then this is identified and corrected. The reversed condition is indicated by the flag “XPOL“, bit 4 in register 27. The 10M PLL is locked onto the received Manchester signal and from this, generates the received 20MHz clock. Using this clock, the Manchester encoded data is extracted and converted to a 10MHz NRZI data stream. It is then converted from serial to 4-bit wide parallel data.

The RX10M block also detects valid 10Base-T IDLE signals - Normal Link Pulses (NLPs) - to maintain the link.

4.5.3 Jabber Detection

Jabber is a condition in which a station transmits for a period of time longer than the maximum permissible packet length, usually due to a fault condition, that results in holding the TX_EN input for a long period. Special logic is used to detect the jabber state and abort the transmission to the line, within 45ms. Once TX_EN is deasserted, the logic resets the jabber condition.

4.6 Auto-negotiation

The purpose of the Auto-negotiation function is to automatically configure the PHY to the optimum link parameters based on the capabilities of its link partner. Auto-negotiation is a mechanism for exchanging configuration information between two link-partners and automatically selecting the highest

performance mode of operation supported by both sides. Auto-negotiation is fully defined in clause 28 of the IEEE 802.3 specification.

Once auto-negotiation has completed, information about the resolved link can be passed back to the controller via the internal Serial Management Interface (SMI). The results of the negotiation process are reflected in the Speed Indication bits in register 31, as well as the Link Partner Ability Register (Register 5).

The auto-negotiation protocol is a purely physical layer activity and proceeds independently of the MAC controller.

The advertised capabilities of the PHY are stored in register 4 of the SMI registers. The default advertised by the PHY is determined by user-defined on-chip signal options.

The following blocks are activated during an Auto-negotiation session:

- Auto-negotiation (digital)
- 100M ADC (analog)
- 100M PLL (analog)
- 100M equalizer/BLW/clock recovery (DSP)
- 10M SQUELCH (analog)
- 10M PLL (analog)
- 10M Transmitter (analog)

When enabled, auto-negotiation is started by the occurrence of one of the following events:

- Hardware reset
- Software reset
- Power-down reset
- Link status down
- Setting register 0, bit 9 high (auto-negotiation restart)

On detection of one of these events, the PHY begins auto-negotiation by transmitting bursts of Fast Link Pulses (FLP). These are bursts of link pulses from the 10M transmitter. They are shaped as Normal Link Pulses and can pass uncorrupted down CAT-3 or CAT-5 cable. A Fast Link Pulse Burst consists of up to 33 pulses. The 17 odd-numbered pulses, which are always present, frame the FLP burst. The 16 even-numbered pulses, which may be present or absent, contain the data word being transmitted. Presence of a data pulse represents a “1”, while absence represents a “0”.

The data transmitted by an FLP burst is known as a “Link Code Word.” These are defined fully in IEEE 802.3 clause 28. In summary, the PHY advertises 802.3 compliance in its selector field (the first 5 bits of the Link Code Word). It advertises its technology ability according to the bits set in register 4 of the SMI registers.

There are 4 possible matches of the technology abilities. In the order of priority these are:

- 100M full-duplex (Highest priority)
- 100M half-duplex
- 10M full-duplex
- 10M half-duplex

If the full capabilities of the PHY are advertised (100M, full-duplex), and if the link partner is capable of 10M and 100M, then auto-negotiation selects 100M as the highest performance mode. If the link partner is capable of half and full-duplex modes, then auto-negotiation selects full-duplex as the highest performance operation.

Once a capability match has been determined, the link code words are repeated with the acknowledge bit set. Any difference in the main content of the link code words at this time will cause auto-negotiation to re-start. Auto-negotiation will also re-start if not all of the required FLP bursts are received.

Writing register 4 bits [8:5] allows software control of the capabilities advertised by the PHY. Writing register 4 does not automatically re-start auto-negotiation. Register 0, bit 9 must be set before the new abilities will be advertised. Auto-negotiation can also be disabled via software by clearing register 0, bit 12.

The LAN9210 does not support “Next Page" capability.

4.7 Parallel Detection

If the LAN9210 is connected to a device lacking the ability to auto-negotiate (i.e. no FLPs are detected), it is able to determine the speed of the link based on either 100M MLT-3 symbols or 10M Normal Link Pulses. In this case the link is presumed to be half-duplex per the IEEE standard. This ability is known as “Parallel Detection. This feature ensures inter operability with legacy link partners. If a link is formed via parallel detection, then bit 0 in register 6 is cleared to indicate that the Link Partner is not capable of auto-negotiation. The Ethernet MAC has access to this information via the management interface. If a fault occurs during parallel detection, bit 4 of register 6 is set.

Register 5 is used to store the Link Partner Ability information, which is coded in the received FLPs. If the Link Partner is not auto-negotiation capable, then register 5 is updated after completion of parallel detection to reflect the speed capability of the Link Partner.

4.7.1 Re-starting Auto-negotiation

Auto-negotiation can be re-started at any time by setting register 0, bit 9. Auto-negotiation will also re-start if the link is broken at any time. A broken link is caused by signal loss. This may occur because of a cable break, or because of an interruption in the signal transmitted by the Link Partner. Auto-negotiation resumes in an attempt to determine the new link configuration.

If the management entity re-starts Auto-negotiation by writing to bit 9 of the control register, the LAN9210 will respond by stopping all transmission/receiving operations. Once the break_link_timer is done, in the Auto-negotiation state-machine (approximately 1200ms) the auto-negotiation will re-start. The Link Partner will have also dropped the link due to lack of a received signal, so it too will resume auto-negotiation.

4.7.2 Disabling Auto-negotiation

Auto-negotiation can be disabled by setting register 0, bit 12 to zero. The device will then force its speed of operation to reflect the information in register 0, bit 13 (speed) and register 0, bit 8 (duplex). The speed and duplex bits in register 0 should be ignored when auto-negotiation is enabled.

4.7.3 Half vs. Full-Duplex

Half-duplex operation relies on the CSMA/CD (Carrier Sense Multiple Access / Collision Detect) protocol to handle network traffic and collisions. In this mode, the internal carrier sense signal, CRS, responds to both transmit and receive activity. In this mode, If data is received while the PHY is transmitting, a collision results.

In full-duplex mode, the PHY is able to transmit and receive data simultaneously. In this mode, the internal CRS responds only to receive activity. The CSMA/CD protocol does not apply and collision detection is disabled.

Table 4.2 describes the behavior of the internal CRS bit under all receive/transmit conditions. The internal CRS signal is used to trigger bit 10 (No Carrier) of the TX Status word (See Section 3.12.4, "TX Status Format"). The CRS value, and subsequently the No Carrier value, are invalid during any

full-duplex transmission. Therefore, these signals cannot be used as a verification method of transmitted packets when transmitting in 10/100 Mbps full-duplex modes.

Table 4.2 CRS Behavior

MODE	SPEED	DUPLEX	ACTIVITY	CRS BEHAVIOR (Note 4.1)
Manual	10 Mbps	Half-Duplex	Transmitting	Active
Manual	10 Mbps	Half-Duplex	Receiving	Active
Manual	10 Mbps	Full-Duplex	Transmitting	Low
Manual	10 Mbps	Full-Duplex	Receiving	Active
Manual	100 Mbps	Half-Duplex	Transmitting	Active
Manual	100 Mbps	Half-Duplex	Receiving	Active
Manual	100 Mbps	Full-Duplex	Transmitting	Low
Manual	100 Mbps	Full-Duplex	Receiving	Active
Auto-Negotiation	10 Mbps	Half-Duplex	Transmitting	Active
Auto-Negotiation	10 Mbps	Half-Duplex	Receiving	Active
Auto-Negotiation	10 Mbps	Full-Duplex	Transmitting	Low
Auto-Negotiation	10 Mbps	Full-Duplex	Receiving	Active
Auto-Negotiation	100 Mbps	Half-Duplex	Transmitting	Active
Auto-Negotiation	100 Mbps	Half-Duplex	Receiving	Active
Auto-Negotiation	100 Mbps	Full-Duplex	Transmitting	Low
Auto-Negotiation	100 Mbps	Full-Duplex	Receiving	Active

Note 4.1 The LAN9210 10/100 PHY internal CRS signal operates in two modes: Active and Low. When in Active mode, the internal CRS will transition high and low upon line activity, where a high value indicates a carrier has been detected. In Low mode, the internal CRS stays low and does not indicate carrier detection. The internal CRS signal and No Carrier (bit 10 of the TX Status word) cannot be used as a verification method of transmitted packets when transmitting in 10/100 Mbps full-duplex mode.

4.8 HP Auto-MDIX

HP Auto-MDIX facilitates the use of CAT-3 (10 Base-T) or CAT-5 (100 Base-T) media UTP interconnect cable without consideration of interface wiring scheme. If a user plugs in either a direct connect LAN cable, or a cross-over patch cable, as shown in Figure 4.3, the SMSC LAN9210 Auto-MDIX PHY is capable of configuring the TPO and TPI twisted pair pins for correct transceiver operation.

The internal logic of the device detects the TX and RX pins of the connecting device. Since the RX and TX line pairs are interchangeable, special PCB design considerations are needed to accommodate the symmetrical magnetics and termination of an Auto-MDIX design.

The Auto-MDIX function can be disabled through an internal register 27.15, or the external control pins AMDIX_EN. When disabled the TX and RX pins can be configured with the Channel Select (CH_SELECT) pin as desired.

The figure below shows the signal names at the RJ-45 connector, The mapping of these signals to the pins on the LAN9210 is as follows:
TXP = TPO+
TXN = TPO-
RXP = TPI+
RXN = TPI-

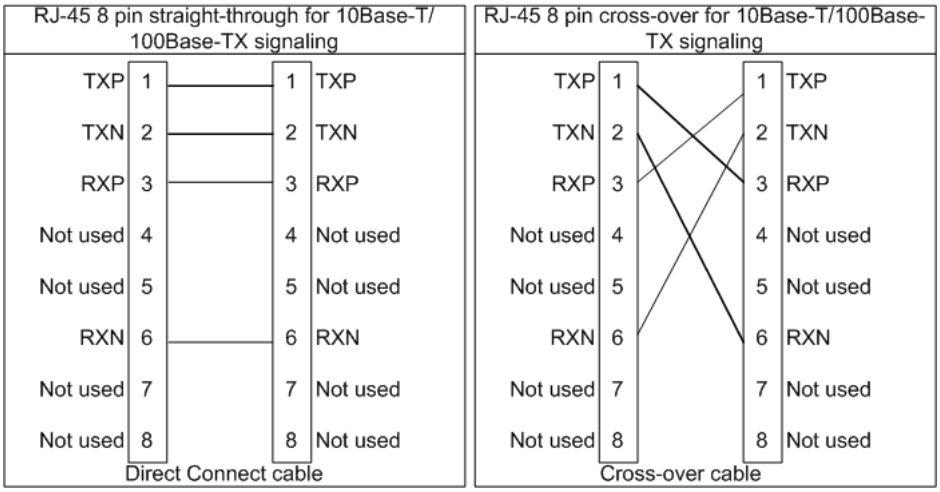


Figure 4.3 Direct Cable Connection vs. Cross-over Cable Connection

Chapter 5 Register Description

The following section describes all LAN9210 registers and data ports.

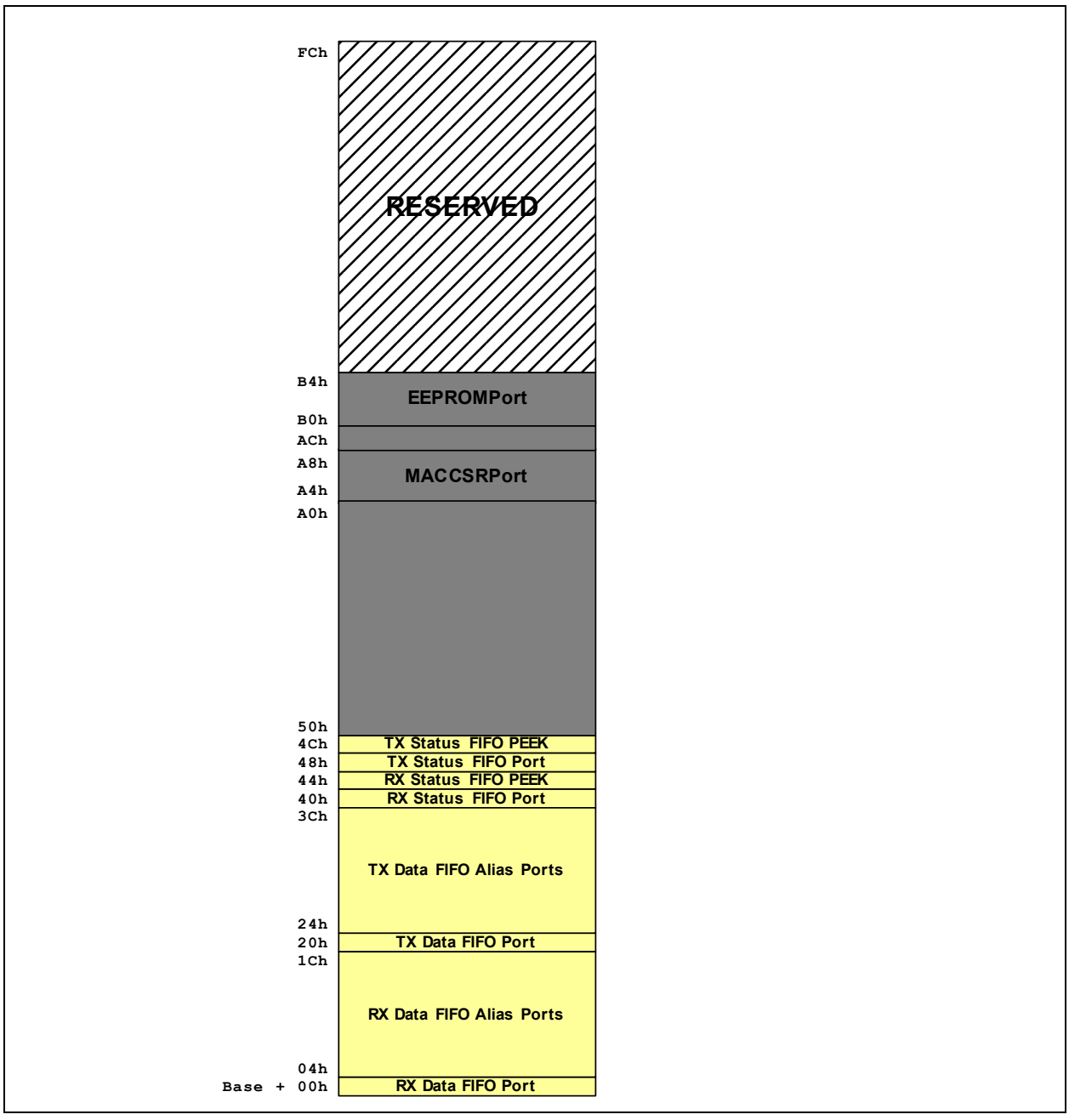


Figure 5.1 Memory Map

5.1 Register Nomenclature and Access Attributes

SYMBOL	DESCRIPTION
RO	**Read Only:** If a register is read only, writes to this register have no effect.
WO	**Write Only:** If a register is write only, reads always return 0.
R/W	**Read/Write:** A register with this attribute can be read and written
R/WC	**Read/Write Clear:** A register bit with this attribute can be read and written. However, a write of a 1 clears (sets to 0) the corresponding bit and a write of a 0 has no effect.
RC	**Read to Clear:** A register bit with this attribute is cleared when read.
LL	**Latch Low:** Clear on read of register
LH	**Latch High:** Clear on read of register
SC	**Self-Clearing**
NASR	**Not Affected by Software Reset**
Reserved Bits	Certain bits within registers are listed as “Reserved”. Unless stated otherwise, these bits must be written with zeros for future compatibility. The values of these bits are not guaranteed when read.
Reserved Registers	Certain configuration registers within the LAN9210 are listed as “Reserved”. These registers are not guaranteed to return any particular value when read. These registers must not be written to or modified by system failure; doing so could result in failure of the device and system.
Default States	At Reset - System reset or Software Reset - internal registers are set to their default states. The default states provide a minimum level of functionality needed to successfully bring up a system, but do not necessarily provide desired or optimal configuration of the device. It is the responsibility of the system initialization software to properly determine the operating parameters and optional system features that are applicable, and to program the LAN9210 registers accordingly.

5.2 RX and TX FIFO Ports

The LAN9210 contains four host-accessible FIFOs: RX Status, RX Data, TX Status, and TX Data FIFOs. The sizes Data FIFOs and the RX Status FIFO are configurable through the CSRs.

5.2.1 RX FIFO Ports

The RX Data Path contains two Read-Only FIFOs: RX Status and RX Data. The RX Status FIFO has two ports at different address locations. The RX Status FIFO Port causes the top of the RX Status FIFO to be “popped”, and is destructive. The RX Status FIFO PEEK Port allows the top of the RX Status FIFO to be read without “popping” it.

The RX Data FIFO has a single port; reading data from this port always causes the top of the RX Data FIFO to be “popped”. This port is aliased to 16 WORD locations. The host may access the top of the RX Data FIFO through any of these locations.

5.2.2 TX FIFO Ports

The TX Data Path consists of two FIFOs, TX Status and RX Data. The TX Status FIFO also has two ports at different locations. When the TX Status FIFO Port is read, the top of the TX Status FIFO is popped. When the TX Status FIFO PEEK Port is read, the top of the TX Status FIFO is not popped.

The TX data FIFO is Write Only. It is aliased to 16 WORD locations . The host may access the top of the TX Data FIFO through any of these locations.

5.3 System Control and Status Registers

Table 5.1, "Direct Address Register Map", lists the registers that are directly addressable by the host bus.

Table 5.1 Direct Address Register Map

CONTROL AND STATUS REGISTERS			
BASE ADDRESS + OFFSET	**SYMBOL**	**REGISTER NAME**	**DEFAULT**
50h	ID_REV	Chip ID and Revision.	See Page 79.
54h	IRQ_CFG	Main Interrupt Configuration	00000000h
58h	INT_STS	Interrupt Status	00000000h
5Ch	INT_EN	Interrupt Enable Register	00000000h
60h	RESERVED	Reserved for future use	-
64h	BYTE_TEST	Read-only byte order testing register	87654321h
68h	FIFO_INT	FIFO Level Interrupts	48000000h
6Ch	RX_CFG	Receive Configuration	00000000h
70h	TX_CFG	Transmit Configuration	00000000h
74h	HW_CFG	Hardware Configuration	00050000h
78h	RX_DP_CTL	RX Datapath Control	00000000h
7Ch	RX_FIFO_INF	Receive FIFO Information	00000000h
80h	TX_FIFO_INF	Transmit FIFO Information	00001200h
84h	PMT_CTRL	Power Management Control	00000000h
88h	GPIO_CFG	General Purpose IO Configuration	00000000h
8Ch	GPT_CFG	General Purpose Timer Configuration	0000FFFFh
90h	GPT_CNT	General Purpose Timer Count	0000FFFFh
94h	RESERVED	Reserved for future use	-
98h	WORD_SWAP	WORD SWAP Register	00000000h
9Ch	FREE_RUN	Free Run Counter	-
A0h	RX_DROP	RX Dropped Frames Counter	00000000h
A4h	MAC_CSR_CMD	MAC CSR Synchronizer Command (MAC CSR's are indexed through this register)	00000000h
A8h	MAC_CSR_DATA	MAC CSR Synchronizer Data	00000000h
ACh	AFC_CFG	Automatic Flow Control Configuration	00000000h
B0h	E2P_CMD	EEPROM Command	00000000h
B4h	E2P_DATA	EEPROM Data	00000000h
B8h - FCh	RESERVED	Reserved for future use	-

5.3.1 ID_REV—Chip ID and Revision

Offset: 50h Size: 32 bits

This register contains the ID and Revision fields for this design.

BITS	DESCRIPTION	TYPE	DEFAULT
31-16	**Chip ID.** This read-only field identifies this design	RO	9210h
15-0	**Chip Revision**	RO	0000h

5.3.2 IRQ_CFG—Interrupt Configuration Register

Offset: 54h Size: 32 bits

This register configures and indicates the state of the IRQ signal.

BITS	DESCRIPTION	TYPE	DEFAULT
31:24	**Interrupt Deassertion Interval (INT_DEAS).** This field determines the Interrupt Request Deassertion Interval in multiples of 10 microseconds. Setting this field to zero causes the device to disable the INT_DEAS Interval, reset the interval counter, and issue any pending interrupts. If a new, non-zero value is written to this field, any subsequent interrupts will obey the new setting. **Note:** This field does not apply to the PME interrupt.	R/W	0
23-15	**Reserved**	RO	-
14	**Interrupt Deassertion Interval Clear (INT_DEAS_CLR**). Writing a one to this register clears the de-assertion counter in the IRQ Controller, thus causing a new de-assertion interval to begin (regardless of whether or not the IRQ Controller is currently in an active de-assertion interval).	SC	0
13	**Interrupt Deassertion Status (INT_DEAS_STS).** When set, this bit indicates that interrupts are currently in a deassertion interval, and will not be delivered to the IRQ pin. When this bit is clear, interrupts are not currently in a deassertion interval, and will be delivered to the IRQ pin.	SC	0
12	**Master Interrupt (IRQ_INT).** This read-only bit indicates the state of the internal IRQ line, regardless of the setting of the IRQ_EN bit, or the state of the interrupt de-assertion function. When this bit is high, one of the enabled interrupts is currently active.	RO	0
11-9	**Reserved**	RO	-
8	**IRQ Enable (IRQ_EN) –** This bit controls the final interrupt output to the IRQ pin. When clear, the IRQ output is disabled and permanently deasserted. This bit has no effect on any internal interrupt status bits.	R/W	0
7-5	**Reserved**	RO	-

BITS	DESCRIPTION	TYPE	DEFAULT
4	**IRQ Polarity (IRQ_POL) –** When cleared, enables the IRQ line to function as an active low output. When set, the IRQ output is active high. When IRQ is configured as an open-drain output this field is ignored, and the interrupt output is always active low.	R/W NASR	0
3-1	**Reserved**	RO	-
0	**IRQ Buffer Type (IRQ_TYPE) –** When cleared, enables IRQ to function as an open-drain buffer for use in a Wired-Or Interrupt configuration. When set, the IRQ output is a Push-Pull driver. When configured as an open-drain output the IRQ_POL field is ignored, and the interrupt output is always active low.	R/W NASR	0

5.3.3 INT_STS—Interrupt Status Register

Offset: 58h Size: 32 bits

This register contains the current status of the generated interrupts. Writing a 1 to the corresponding bits acknowledges and clears the interrupt.

BITS	DESCRIPTION	TYPE	DEFAULT
31	**Software Interrupt (SW_INT).** This interrupt is generated when the SW_INT_EN bit is set high. Writing a one clears this interrupt.	R/WC	0
30-26	**Reserved**	RO	-
25	**TX Stopped (TXSTOP_INT).** This interrupt is issued when STOP_TX bit in TX_CFG is set, and the transmitter is halted.	R/WC	0
24	**RX Stopped (RXSTOP_INT).** This interrupt is issued when the receiver is halted.	R/WC	0
23	**RX Dropped Frame Counter Halfway (RXDFH_INT).** This interrupt is issued when the RX Dropped Frames Counter counts past its halfway point (7FFFFFFFh to 80000000h).	R/WC	0
22	**Reserved**	RO	0
21	**TX IOC Interrupt (TX_IOC).** When a buffer with the IOC flag set has finished being loaded into the TX FIFO, this interrupt is generated.	R/WC	0
20	**RX DMA Interrupt (RXD_INT).** This interrupt is issued when the amount of data programmed in the RX DMA Count (RX_DMA_CNT) field of the RX_CFG register has been transferred out of the RX FIFO.	R/WC	0
19	**GP Timer (GPT_INT).** This interrupt is issued when the General Purpose timer wraps past zero to FFFFh.	R/WC	0
18	**PHY (PHY_INT).** Indicates a PHY Interrupt event.	RO	0
17	**Power Management Event Interrupt (PME_INT).** This interrupt is issued when a Power Management Event is detected as configured in the PMT_CTRL register. This interrupt functions independent of the PME signal, and will still function if the PME signal is disabled. Writing a '1' clears this bit regardless of the state of the PME hardware signal. **Notes:** ▪ Detection of a Power Management Event, and assertion of the PME signal will not wakeup the LAN9210. The LAN9210 will only wake up when it detects a host write cycle of any data to the BYTE_TEST register. ▪ The Interrupt Deassertion interval does not apply to the PME interrupt.	R/WC	0
16	**TX Status FIFO Overflow (TXSO).** Generated when the TX Status FIFO overflows.	R/WC	0
15	**Receive Watchdog Time-out (RWT).** Interrupt is generated when a packet larger than 2048 bytes has been received.	R/WC	0
14	**Receiver Error (RXE).** Indicates that the receiver has encountered an error. Please refer to Section 3.13.5, "Receiver Errors," on page 65 for a description of the conditions that will cause an RXE.	R/WC	0
13	**Transmitter Error (TXE).** When generated, indicates that the transmitter has encountered an error. Please refer to Section 3.12.7, "Transmitter Errors," on page 60, for a description of the conditions that will cause a TXE.	R/WC	0

BITS	DESCRIPTION	TYPE	DEFAULT
12:11	**Reserved**	RO	-
10	**TX Data FIFO Overrun Interrupt (TDFO).** Generated when the TX data FIFO is full, and another write is attempted.	R/WC	0
9	**TX Data FIFO Available Interrupt (TDFA).** Generated when the TX data FIFO available space is greater than the programmed level.	R/WC	0
8	**TX Status FIFO Full Interrupt (TSFF).** Generated when the TX Status FIFO is full.	R/WC	0
7	**TX Status FIFO Level Interrupt (TSFL).** Generated when the TX Status FIFO reaches the programmed level.	R/WC	0
6	**RX Dropped Frame Interrupt (RXDF_INT).** This interrupt is issued whenever a receive frame is dropped.	R/WC	0
5	**Reserved**	RO	-
4	**RX Status FIFO Full Interrupt (RSFF).** Generated when the RX Status FIFO is full.	R/WC	0
3	**RX Status FIFO Level Interrupt (RSFL).** Generated when the RX Status FIFO reaches the programmed level.	R/WC	0
2-0	**GPIO [2:0] (GPIOx_INT).** Interrupts are generated from the GPIO's. These interrupts are configured through the GPIO_CFG register.	R/WC	000

5.3.4 INT_EN—Interrupt Enable Register

Offset: 5Ch Size: 32 bits

This register contains the interrupt masks for IRQ. Writing 1 to any of the bits enables the corresponding interrupt as a source for IRQ. Bits in the INT_STS register will still reflect the status of the interrupt source regardless of whether the source is enabled as an interrupt in this register.

BITS	DESCRIPTION	TYPE	DEFAULT
31	Software Interrupt (SW_INT_EN)	R/W	0
30:26	Reserved	RO	-
25	TX Stopped Interrupt Enable (TXSTOP_INT_EN)	R/W	0
24	RX Stopped Interrupt Enable (RXSTOP_INT_EN)	R/W	0
23	RX Dropped Frame Counter Halfway Interrupt Enable (RXDFH_INT_EN).	R/W	0
22	Reserved	RO	0
21	TX IOC Interrupt Enable (TIOC_INT_EN)	R/W	0
20	RX DMA Interrupt (RXD_INT).	R/W	0
19	GP Timer (GPT_INT_EN)	R/W	0
18	PHY (PHY_INT_EN)	R/W	0
17	Power Management Event Interrupt Enable (PME_INT_EN)	R/W	0
16	TX Status FIFO Overflow (TXSO_EN)	R/W	0
15	Receive Watchdog Time-out Interrupt (RWT_INT_EN)	R/W	0
14	Receiver Error Interrupt (RXE_INT_EN)	R/W	0
13	Transmitter Error Interrupt (TXE_INT_EN)	R/W	0
12:11	Reserved	RO	-
10	TX Data FIFO Overrun Interrupt (TDFO_INT_EN)	R/W	0
9	TX Data FIFO Available Interrupt (TDFA_INT_EN)	R/W	0
8	TX Status FIFO Full Interrupt (TSFF_INT_EN)	R/W	0
7	TX Status FIFO Level Interrupt (TSFL_INT_EN)	R/W	0
6	RX Dropped Frame Interrupt Enable (RXDF_INT_EN)	R/W	0
5	Reserved	RO	-
4	RX Status FIFO Full Interrupt (RSFF_INT_EN)	R/W	0
3	RX Status FIFO Level Interrupt (RSFL_INT_EN)	R/W	0
2-0	GPIO [2:0] (GPIOx_INT_EN).	R/W	000

5.3.5 BYTE_TEST—Byte Order Test Register

Offset: 64h Size: 32 bits

This register can be used to determine the byte ordering of the current configuration

BITS	DESCRIPTION	TYPE	DEFAULT
31:0	**Byte Test**	RO	87654321h

5.3.6 FIFO_INT—FIFO Level Interrupts

Offset: 68h Size: 32 bits

This register configures the limits where the FIFO Controllers will generate system interrupts.

BITS	DESCRIPTION	TYPE	DEFAULT
31-24	**TX Data Available Level.** The value in this field sets the level, in number of 64 Byte blocks, at which the TX FIFO Available interrupt (TFDA) will be generated. When the TX data FIFO free space is greater than this value a TX FIFO Available interrupt (TDFA) will be generated.	R/W	48h
23-16	**TX Status Level.** The value in this field sets the level, in number of DWORDs, at which the TX Status FIFO Level interrupt (TSFL) will be generated. When the TX Status FIFO used space is greater than this value an TX Status FIFO Level interrupt (TSFL) will be generated.	R/W	00h
15-8	**Reserved**	RO	-
7-0	**RX Status Level.** The value in this field sets the level, in number of DWORDs, at which the RX Status FIFO Level interrupt (RSFL) will be generated. When the RX Status FIFO used space is greater than this value an RX Status FIFO Level interrupt (RSFL) will be generated.	R/W	00h

5.3.7 RX_CFG—Receive Configuration Register

Offset: 6Ch Size: 32 bits

This register controls the LAN9210 receive engine.

BITS	DESCRIPTION	TYPE	DEFAULT
31:30	**RX End Alignment.** This field specifies the alignment that must be maintained on the last data transfer of a buffer. The LAN9210 will add extra DWORDs of data up to the alignment specified in the table below. The host is responsible for removing these extra DWORDs. This mechanism can be used to maintain cache line alignment on host processors. Please refer to Table 5.2 for bit definitions **Note:** The desired RX End Alignment must be set before reading a packet. The RX end alignment can be changed between reading receive packets, but must not be changed if the packet is partially read.	R/W	00b
29-28	**Reserved**	RO	-
27-16	**RX DMA Count (RX_DMA_CNT).** This 12-bit field indicates the amount of data, in DWORDS, to be transferred out of the RX data FIFO before asserting the RXD_INT. After being set, this field is decremented for each DWORD of data that is read from the RX data FIFO. This field can be overwritten with a new value before it reaches zero.	R/W	000h
15	**Force RX Discard (RX_DUMP).** This self-clearing bit clears the RX data and status FIFOs of all pending data. When a '1' is written, the RX data and status pointers are cleared to zero. **Note:** Please refer to section "Force Receiver Discard (Receiver Dump)" on page 62 for a detailed description regarding the use of RX_DUMP.	SC	0
14-13	**Reserved**	RO	-
12-8	**RX Data Offset (RXDOFF).** This field controls the offset value, in bytes, that is added to the beginning of an RX data packet. The start of the valid data will be shifted by the number of bytes specified in this field. An offset of 0-31 bytes is a valid number of offset bytes. **Note:** The two LSBs of this field (D[9:8]) must not be modified while the RX is running. The receiver must be halted, and all data purged before these two bits can be modified. The upper three bits (DWORD offset) may be modified while the receiver is running. Modifications to the upper bits will take affect on the next DWORD read.	R/W	00000
7-0	**Reserved**	RO	-

Table 5.2 RX Alignment Bit Definitions

[31]	[30]	End Alignment
0	0	4-byte alignment
0	1	16-byte alignment
1	0	32-byte alignment
1	1	Reserved

5.3.8 TX_CFG—Transmit Configuration Register

Offset: 70h Size: 32 bits

This register controls the transmit functions on the LAN9210 Ethernet Controller.

BITS	DESCRIPTION	TYPE	DEFAULT
31-16	**Reserved.**	RO	-
15	**Force TX Status Discard (TXS_DUMP).** This self-clearing bit clears the TX status FIFO of all pending status DWORD's. When a '1' is written, the TX status pointers are cleared to zero.	SC	0
14	**Force TX Data Discard (TXD_DUMP).** This self-clearing bit clears the TX data FIFO of all pending data. When a '1' is written, the TX data pointers are cleared to zero.	SC	0
13-3	**Reserved**	RO	-
2	**TX Status Allow Overrun (TXSAO).** When this bit is cleared, data transmission is suspended if the TX Status FIFO becomes full. Setting this bit high allows the transmitter to continue operation with a full TX Status FIFO. **Note:** This bit does not affect the operation of the TX Status FIFO Full interrupt.	R/W	0
1	**Transmitter Enable (TX_ON).** When this bit is set (1), the transmitter is enabled. Any data in the TX FIFO will be sent. This bit is cleared automatically when STOP_TX is set and the transmitter is halted.	R/W	0
0	**Stop Transmitter (STOP_TX).** When this bit is set (1), the transmitter will finish the current frame, and will then stop transmitting. When the transmitter has stopped this bit will clear. All writes to this bit are ignored while this bit is high.	SC	0

5.3.9 HW_CFG—Hardware Configuration Register

Offset: 74h Size: 32 bits

Note: The transmitter and receiver must be stopped before writing to this register. Refer to Section 3.12.8, "Stopping and Starting the Transmitter," on page 60 and Section 3.13.4, "Stopping and Starting the Receiver," on page 65 for details on stopping the transmitter and receiver.

BITS	DESCRIPTION	TYPE	DEFAULT
31	**Reserved**	RO	-
30	**Reserved**	RO	-
29	**FIFO Port Endian Ordering (FPORTEND).** This control bit determines the endianess of RX and TX data FIFO host accesses when accessed through the RX/TX Data FIFO ports, including the alias addresses (any access from 00h to 3Ch). When this bit is cleared, data FIFO port accesses utilize little endian byte ordering. When this bit is set, data FIFO port accesses utilize big endian byte ordering. Please refer to section Section 3.7.3, "Mixed Endian Support," on page 33 for more information on this feature.	R/W NASR	0
28	**Direct FIFO Access Endian Ordering (FSELEND).** This control bit determines the endianess of RX and TX data FIFO host accesses when accessed using the FIFO_SEL signal. When this bit is cleared, FIFO_SEL accesses utilize little endian byte ordering. When this bit is set, FIFO_SEL accesses utilize big endian byte ordering. Please refer to section Section 3.7.3, "Mixed Endian Support," on page 33 for more information on this feature.	R/W NASR	0
27-25	**Reserved**	RO	-
24	**AMDIX_EN Strap State**. This read-only bit reflects the state of the AMDIX_EN strap pin (pin 73). This pin can be overridden by PHY Registers 27.15 and 27.13	RO	AMDIX Strap Pin
23-21	**Reserved**	RO	
20	**Must Be One (MBO).** This bit must be set to “1” for normal device operation.	R/W	0
16-19	**TX FIFO Size (TX_FIF_SZ).** Sets the size of the TX FIFOs in 1KB values to a maximum of 14KB. The TX Status FIFO consumes 512 bytes of the space allocated by TX_FIF_SIZ, and the TX data FIFO consumes the remaining space specified by TX_FIF_SZ. The minimum size of the TX FIFOs is 2KB (TX data and status combined). The TX data FIFO is used for both TX data and TX commands. The RX Status and data FIFOs consume the remaining space, which is equal to 16KB – TX_FIF_SIZ. See Section 5.3.9.1, "Allowable settings for Configurable FIFO Memory Allocation," on page 89 for more information.	R/W	5h
15-2	**Reserved**	RO	-
1	**Soft Reset Timeout (SRST_TO)**. If a software reset is attempted when the PHY is not in the operational state (RX_CLK and TX_CLK running), the reset will not complete and the soft reset operation will timeout and this bit will be set to a ‘1’. The host processor must correct the problem and issue another soft reset.	RO	0

BITS	DESCRIPTION	TYPE	DEFAULT
0	**Soft Reset (SRST)**. Writing 1 generates a software initiated reset. This reset generates a full reset of the MAC CSR's. The SCSR's (system command and status registers) are reset except for any NASR bits. Soft reset also clears any TX or RX errors (TXE/RXE). This bit is self-clearing. **Notes:** ■ Do not attempt a soft reset unless the PHY is fully awake and operational. After a PHY reset, or when returning from a reduced power state, the PHY must given adequate time to return to the operational state before a soft reset can be issued. ■ The LAN9210 must always be read at least once after power-up, reset, or upon return from a power-saving state or write operations will not function.	SC	0

5.3.9.1 Allowable settings for Configurable FIFO Memory Allocation

TX and RX FIFO space is configurable through the CSR - HW_CFG register defined above. The user must select the FIFO allocation by setting the TX FIFO Size (TX_FIF_SZ) field in the hardware configuration (HW_CFG) register. The TX_FIF_SZ field selects the total allocation for the TX data path, including the TX Status FIFO size. The TX Status FIFO size is fixed at 512 Bytes (128 TX Status DWORDs). The TX Status FIFO length is subtracted from the total TX FIFO size with the remainder being the TX data FIFO Size. Note that TX data FIFO space includes both commands and payload data.

RX FIFO Size is the remainder of the unallocated FIFO space (16384 bytes – TX FIFO Size). The RX Status FIFO size is always equal to 1/16 of the RX FIFO Size. The RX Status FIFO length is subtracted from the total RX FIFO size with the remainder being the RX data FIFO Size.

For example, if TX_FIF_SZ = 6 then:

Total TX FIFO Size = 6144 Bytes (6KB)

TX Status FIFO Size = 512 Bytes (Fixed)

TX Data FIFO Size = 6144 – 512 = 5632 Bytes

RX FIFO Size = 16384 – 6144 = 10240 Bytes (10KB)

RX Status FIFO Size = 10240 / 16 = 640 Bytes (160 RX Status DWORDs)

RX Data FIFO Size = 10240 – 640 = 9600 Bytes

Table 5.3 shows every valid setting for the TX_FIF_SZ field. Note that settings not shown in this table are reserved and should not be used.

Note: The RX data FIFO is considered full 4 DWORDs before the length that is specified in the HW_CFG register.

Table 5.3 Valid TX/RX FIFO Allocations

TX_FIF_SZ	TX DATA FIFO SIZE (BYTES)	TX STATUS FIFO SIZE (BYTES)	RX DATA FIFO SIZE (BYTES)	RX STATUS FIFO SIZE (BYTES)
2	1536	512	13440	896
3	2560	512	12480	832
4	3584	512	11520	768
5	4608	512	10560	704
6	5632	512	9600	640
7	6656	512	8640	576
8	7680	512	7680	512
9	8704	512	6720	448
10	9728	512	5760	384
11	10752	512	4800	320
12	11776	512	3840	256
13	12800	512	2880	192
14	13824	512	1920	128

In addition to the host-accessible FIFOs, the MAC Interface Layer (MIL) contains an additional 2K bytes of TX, and 128 bytes of RX FIFO buffering. These sizes are fixed, and cannot be adjusted by the host.

As space in the TX MIL (Mac Interface Layer) FIFO frees, data is moved into it from the TX data FIFO. Depending on the size of the frames to be transmitted, the MIL can hold up to two Ethernet frames. This is in addition to any TX data that may be queued in the TX data FIFO.

Conversely, as data is received by the LAN9210, it is moved from the MAC to the RX MIL FIFO, and then into the RX data FIFO. When the RX data FIFO fills up, data will continue to collect in the RX MIL FIFO. If the RX MIL FIFO fills up and overruns, subsequent RX frames will be lost until room is made in the RX data FIFO. For each frame of data that is lost, the RX Dropped Frames Counter (RX_DROP) is incremented.

RX and TX MIL FIFO levels are not visible to the host processor. RX and TX MIL FIFOs operate independent of the TX data and RX data and status FIFOs. FIFO levels set for the RX and TX data and Status FIFOs do not take into consideration the MIL FIFOs.

5.3.10 RX_DP_CTRL—Receive Datapath Control Register

Offset: 78h Size: 32 bits

This register is used to discard unwanted receive frames.

BITS	DESCRIPTION	TYPE	DEFAULT
31	**RX Data FIFO Fast Forward (RX_FFWD):** Writing a ‘1’ to this bit causes the RX data FIFO to fast-forward to the start of the next frame. This bit will remain high until the RX data FIFO fast-forward operation has completed. No reads should be issued to the RX data FIFO while this bit is high. **Note:** Please refer to section “Receive Data FIFO Fast Forward” on page 62 for detailed information regarding the use of RX_FFWD.	R/W SC	0b
30-0	**Reserved**	RO	-

5.3.11 RX_FIFO_INF—Receive FIFO Information Register

Offset: 7Ch Size: 32 bits

This register contains the used space in the receive FIFOs of the LAN9210 Ethernet Controller.

BITS	DESCRIPTION	TYPE	DEFAULT
31-24	**Reserved**	RO	-
23-16	**RX Status FIFO Used Space (RXSUSED).** Indicates the amount of space in DWORDs, used in the RX Status FIFO.	RO	00h
15-0	**RX Data FIFO Used Space (RXDUSED).**). Reads the amount of space in bytes, used in the RX data FIFO. For each receive frame, this field is incremented by the length of the receive data rounded up to the nearest DWORD (if the payload does not end on a DWORD boundary).	RO	0000h

5.3.12 TX_FIFO_INF—Transmit FIFO Information Register

Offset: 80h Size: 32 bits

This register contains the free space in the transmit data FIFO and the used space in the transmit status FIFO in the LAN9210.

BITS	DESCRIPTION	TYPE	DEFAULT
31-24	**Reserved**	RO	-
23-16	**TX Status FIFO Used Space (TXSUSED).** Indicates the amount of space in DWORDS used in the TX Status FIFO.	RO	00h
15-0	**TX Data FIFO Free Space (TDFREE).** Reads the amount of space in bytes, available in the TX data FIFO. The application should never write more data than is available, as indicated by this value.	RO	1200h

5.3.13 PMT_CTRL— Power Management Control Register

Offset: 84h Size: 32 bits

This register controls the Power Management features. This register can be read while the LAN9210 is in a power saving mode.

Note: The LAN9210 must always be read at least once after power-up, reset, or upon return from a power-saving state or write operations will not function.

BITS	DESCRIPTION	TYPE	DEFAULT
31:14	**RESERVED**	RO	-
13-12	**Power Management Mode (PM_MODE)** – These bits set the LAN9210 into the appropriate Power Management mode. Special care must be taken when modifying these bits. **Encoding**: 00b – D0 (normal operation) 01b – D1 (wake-up frame and magic packet detection are enabled) 10b – D2 (can perform energy detect) 11b – RESERVED - Do not set in this mode **Note:** When the LAN9210 is in any of the reduced power modes, a write of any data to the BYTE_TEST register will wake-up the device. DO NOT PERFORM WRITES TO OTHER ADDRRESSES while the READY bit in this register is cleared.	SC	00b
11	**RESERVED**	RO	-
10	**PHY Reset (PHY_RST) –** Writing a '1' to this bit resets the PHY. The internal logic automatically holds the PHY reset for a minimum of 100us. When the PHY is released from reset, this bit is automatically cleared. All writes to this bit are ignored while this bit is high.	SC	0b
9	**Wake-On-Lan Enable (WOL_EN)** – When set, the PME signal (if enabled with PME_EN) will be asserted in accordance with the PME_IND bit upon a WOL event. When set, the PME_INT will also be asserted upon a WOL event, regardless of the setting of the PME_EN bit.	R/W	0b
8	**Energy-Detect Enable (ED_EN)** - When set, the PME signal (if enabled with PME_EN) will be asserted in accordance with the PME_IND bit upon an Energy-Detect event. When set, the PME_INT will also be asserted upon an Energy Detect event, regardless of the setting of the PME_EN bit.	R/W	0b
7	**RESERVED**	RO	-
6	**PME Buffer Type (PME_TYPE) –** When cleared, enables PME to function as an open-drain buffer for use in a Wired-Or configuration. When set, the PME output is a Push-Pull driver. When configured as an open-drain output the PME_POL field is ignored, and the output is always active low.	R/W NASR	0b

BITS	DESCRIPTION	TYPE	DEFAULT
5-4	**WAKE-UP Status (WUPS) –** This field indicates the cause of a wake-up event detection as follows 00b -- No wake-up event detected 01b -- Energy detected 10b -- Wake-up frame or magic packet detected 11b -- Indicates multiple events occurred WUPS bits are cleared by writing a '1' to the appropriate bit. The device must return to the D0 state (READY bit set) before these bits can be cleared. **Note:** In order to clear this bit, it is required that all event sources be cleared as well. The event sources are described in Figure 3.18 PME and PME_INT Signal Generationon page 45.	R/WC	00
3	**PME indication (PME_IND).** The PME signal can be configured as a pulsed output or a static signal, which is asserted upon detection of a wake-up event. When set, the PME signal will pulse active for 50mS upon detection of a wake-up event. When clear, the PME signal is driven continuously upon detection of a wake-up event. The PME signal can be deactivated by clearing the WUPS bits, or by clearing the appropriate enable (refer to Section 3.10.2.3, "Power Management Event Indicators," on page 45).	R/W	0b
2	**PME Polarity (PME_POL).** This bit controls the polarity of the PME signal. When set, the PME output is an active high signal. When reset, it is active low. When PME is configured as an open-drain output this field is ignored, and the output is always active low.	R/W NASR	0b
1	**PME Enable (PME_EN).** When set, this bit enables the external PME signal. This bit does not affect the PME interrupt (PME_INT).	R/W	0b
0	**Device Ready (READY).** When set, this bit indicates that LAN9210 is ready to be accessed. This register can be read when LAN9210 is in any power management mode. Upon waking from any power management mode, including power-up, the host processor can interrogate this field as an indication when LAN9210 has stabilized and is fully alive. Reads and writes of any other address are invalid until this bit is set. **Note:** With the exception of HW_CFG and PMT_CTRL, read access to any internal resources is forbidden while the READY bit is cleared. **Note:** On power-up, this bit can be polled to indicate when a valid soft reset (SRST) can be performed.	RO	-

5.3.14 GPIO_CFG—General Purpose IO Configuration Register

Offset: 88h Size: 32 bits

This register configures the GPIO and LED functions.

BITS	DESCRIPTION	TYPE	DEFAULT
31	Reserved	RO	-
30:28	**LED[3:1] enable (LEDx_EN).** A '1' sets the associated pin as an LED output. When cleared low, the pin functions as a GPIO signal. LED1/GPIO0 – bit 28 LED2/GPIO1 – bit 29 LED3/GPIO2 – bit 30	R/W	000
27	**Reserved**	RO	-
26:24	**GPIO Interrupt Polarity 0-2 (GPIO_INT_POL).** When set high, a high logic level on the corresponding GPIO pin will set the corresponding INT_STS register bit. When cleared low, a low logic level on the corresponding GPIO pin will set the corresponding INT_STS register bit. GPIO Interrupts must also be enabled in GPIOx_INT_EN in the INT_EN register. GPIO0 – bit 24 GPIO1 – bit 25 GPIO2 – bit 26 **Note:** GPIO inputs must be active for greater than 40nS to be recognized as interrupt inputs.	R/W	000
23	**Reserved**	RO	-
22:20	**EEPROM Enable (EEPR_EN).** The value of this field determines the function of the external EEDIO and EECLK: Please refer to Table 5.4 for the EEPROM Enable bit function definitions. **Note:** The host must not change the function of the EEDIO and EECLK pins when an EEPROM read or write cycle is in progress. Do not use reserved settings.	R/W	000
19	**Reserved**	RO	-
18:16	**GPIO Buffer Type 0-2 (GPIOBUFn).** When set, the output buffer for the corresponding GPIO signal is configured as a push/pull driver. When cleared, the corresponding GPIO set configured as an open-drain driver. GPIO0 – bit 16 GPIO1 – bit 17 GPIO2 – bit 18	R/W	000
15:11	**Reserved**	RO	-
10:8	**GPIO Direction 0-2 (GPDIRn).** When set, enables the corresponding GPIO as output. When cleared the GPIO is enabled as an input. GPIO0 – bit 8 GPIO1 – bit 9 GPIO2 – bit 10	R/W	0000
7:5	**Reserved**	RO	-

BITS	DESCRIPTION	TYPE	DEFAULT
4:3	**GPO Data 3-4 (GPODn).** The value written is reflected on GPOn. GPO3 – bit 3 GPO4 – bit 4	R/W	00
2:0	**GPIO Data 0-2 (GPIODn)**. When enabled as an output, the value written is reflected on GPIOn. When read, GPIOn reflects the current state of the corresponding GPIO pin. GPIO0 – bit 0 GPIO1 – bit 1 GPIO2 – bit 2	R/W	000

Table 5.4 EEPROM Enable Bit Definitions

[22]	[21]	[20]	EEDIO FUNCTION	EECLK FUNCTION
0	0	0	EEDIO	EECLK
0	0	1	GPO3	GPO4
0	1	0	Reserved	
0	1	1	GPO3	RX_DV
1	0	0	Reserved	
1	0	1	TX_EN	GPO4
1	1	0	TX_EN	RX_DV
1	1	1	TX_CLK	RX_CLK

5.3.15 GPT_CFG-General Purpose Timer Configuration Register

Offset: 8Ch Size: 32 bits

This register configures the General Purpose timer. The GP Timer can be configured to generate host interrupts at intervals defined in this register.

BITS	DESCRIPTION	TYPE	DEFAULT
31-30	**Reserved**	RO	-
29	**GP Timer Enable (TIMER_EN).** When a one is written to this bit the GP Timer is put into the run state. When cleared, the GP Timer is halted. On the 1 to 0 transition of this bit the GPT_LOAD field will be preset to FFFFh.	R/W	0
28-16	**Reserved**	RO	-
15-0	**General Purpose Timer Pre-Load (GPT_LOAD).** This value is pre-loaded into the GP-Timer.	R/W	FFFFh

5.3.16 GPT_CNT-General Purpose Timer Current Count Register

Offset: 90h Size: 32 bits

This register reflects the current value of the GP Timer.

BITS	DESCRIPTION	TYPE	DEFAULT
31-16	**Reserved**	RO	-
15-0	**General Purpose Timer Current Count (GPT_CNT).** This 16-bit field reflects the current value of the GP Timer.	RO	FFFFh

5.3.17 WORD_SWAP—Word Swap Control

Offset: 98h Size: 32 bits

This register controls how words from the host data bus are mapped to the CSRs and Data FIFOs inside the LAN9210. The LAN9210 always sends data from the Transmit Data FIFO to the network so that the low order word is sent first, and always receives data from the network to the Receive Data FIFO so that the low order word is received first.

BITS	DESCRIPTION	TYPE	DEFAULT
31:0	**Word Swap.** If this field is set to 00000000h, or anything except 0xFFFFFFFFh, the LAN9210 maps words with address bit A[1]=1 to the high order words of the CSRs and Data FIFOs, and words with address bit A[1]=0 to the low order words of the CSRs and Data FIFOs. If this field is set to 0xFFFFFFFFh, the LAN9210 maps words with address bit A[1]=1 to the low order words of the CSRs and Data FIFOs, and words with address bit A[1]=0 to the high order words of the CSRs and Data FIFOs. **Note:** Word swap is used in conjunction with the mixed endian functionality to determine the final byte ordering. Refer to Section 3.7.3, "Mixed Endian Support" for more information.	R/W NASR	00000000h

5.3.18 FREE_RUN—Free-Run 25MHz Counter

Offset: 9Ch Size: 32 bits

This register reflects the value of the free-running 25MHz counter.

BITS	DESCRIPTION	TYPE	DEFAULT
31:0	**Free Running SCLK Counter (FR_CNT):** **Note:** This field reflects the value of a free-running 32-bit counter. At reset the counter starts at zero and is incremented for every 25MHz cycle. When the maximum count has been reached the counter will rollover. Since the bus interface is 16-bits wide, and this is a 32-bit counter, the count value is latched on the first read. The FREE_RUN counter can take up to 160nS to clear after a reset event. **Note:** This counter will run regardless of the power management states D0, D1 or D2.	RO	-

5.3.19 RX_DROP– Receiver Dropped Frames Counter

Offset: A0h Size: 32 bits

This register indicates the number of receive frames that have been dropped.

BITS	DESCRIPTION	TYPE	DEFAULT
31-0	**RX Dropped Frame Counter (RX_DFC).** This counter is incremented every time a receive frame is dropped. RX_DFC is cleared on any read of this register. An interrupt can be issued when this counter passes through its halfway point (7FFFFFFFh to 80000000h).	RC	00000000h

5.3.20 MAC_CSR_CMD – MAC CSR Synchronizer Command Register

Offset: A4h Size: 32 bits

This register is used to control the read and write operations with the MAC CSR's

BITS	DESCRIPTION	TYPE	DEFAULT
31	**CSR Busy.** When a 1 is written into this bit, the read or write operation is performed to the specified MAC CSR. This bit will remain set until the operation is complete. In the case of a read this means that the host can read valid data from the data register. The MAC_CSR_CMD and MAC_CSR_DATA registers should not be modified until this bit is cleared.	SC	0
30	**R/nW.** When set, this bit indicates that the host is requesting a read operation. When clear, the host is performing a write.	R/W	0
29-8	Reserved.	RO	-
7-0	**CSR Address.** The 8-bit value in this field selects which MAC CSR will be accessed with the read or write operation.	R/W	00h

5.3.21 MAC_CSR_DATA – MAC CSR Synchronizer Data Register

Offset: A8h Size: 32 bits

This register is used in conjunction with the MAC_CSR_CMD register to perform read and write operations with the MAC CSR's

BITS	DESCRIPTION	TYPE	DEFAULT
31-0	**MAC CSR Data.** Value read from or written to the MAC CSR's.	R/W	00000000h

5.3.22 AFC_CFG – Automatic Flow Control Configuration Register

Offset: ACh Size: 32 bits

This register configures the mechanism that controls both the automatic, and software-initiated transmission of pause frames and back pressure.

Note: The LAN9210 will not transmit pause frames or assert back pressure if the transmitter is disabled.

BITS	DESCRIPTION	TYPE	DEFAULT
31:24	**Reserved**	RO	-
23:16	**Automatic Flow Control High Level (AFC_HI).** Specifies, in multiples of 64 bytes, the level at which flow control will trigger. When this limit is reached the chip will apply back pressure or will transmit a pause frame as programmed in bits [3:0] of this register. During full-duplex operation only a single pause frame is transmitted when this level is reached. The pause time transmitted in this frame is programmed in the FCPT field of the FLOW register in the MAC CSR space. During half-duplex operation each incoming frame that matches the criteria in bits [3:0] of this register will be jammed for the period set in the BACK_DUR field.	R/W	00h
15:8	**Automatic Flow Control Low Level (AFC_LO).** Specifies, in multiples of 64 bytes, the level at which a pause frame is transmitted with a pause time setting of zero. When the amount of data in the RX data FIFO falls below this level the pause frame is transmitted. A pause time value of zero instructs the other transmitting device to immediately resume transmission. The zero time pause frame will only be transmitted if the RX data FIFO had reached the AFC_HI level and a pause frame was sent. A zero pause time frame is sent whenever automatic flow control in enabled in bits [3:0] of this register. **Note:** When automatic flow control is enabled the AFC_LO setting must always be less than the AFC_HI setting.	R/W	00h
7:4	**Backpressure Duration (BACK_DUR).** When the LAN9210 automatically asserts back pressure, it will be asserted for this period of time. This field has no function and is not used in full-duplex mode. Please refer to Table 5.5, describing Backpressure Duration bit mapping for more information.	R/W	0h
3	**Flow Control on Multicast Frame (FCMULT).** When this bit is set, the LAN9210 will assert back pressure when the AFC level is reached and a multicast frame is received. This field has no function in full-duplex mode.	R/W	0
2	**Flow Control on Broadcast Frame (FCBRD).** When this bit is set, the LAN9210 will assert back pressure when the AFC level is reached and a broadcast frame is received. This field has no function in full-duplex mode.	R/W	0
1	**Flow Control on Address Decode (FCADD).** When this bit is set, the LAN9210 will assert back pressure when the AFC level is reached and a frame addressed to the LAN9210 is received. This field has no function in full-duplex mode.	R/W	0

BITS	DESCRIPTION	TYPE	DEFAULT
0	**Flow Control on Any Frame (FCANY).** When this bit is set, the LAN9210 will assert back pressure, or transmit a pause frame when the AFC level is reached and any frame is received. Setting this bit enables full-duplex flow control when the LAN9210 is operating in full-duplex mode. When this mode is enabled during half-duplex operation, the Flow Controller does not decode the MAC address and will send a pause frame upon receipt of a valid preamble (i.e., immediately at the beginning of the next frame after the RX data FIFO level is reached). When this mode is enabled during full-duplex operation, the Flow Controller will immediately instruct the MAC to send a pause frame when the RX data FIFO level is reached. The MAC will queue the pause frame transmission for the next available window. Setting this bit overrides bits [3:1] of this register.	R/W	0

Table 5.5 Backpressure Duration Bit Mapping

	BACKPRESSURE DURATION	
[19:16]	100Mbs Mode	10Mbs Mode
0h	5uS	7.2uS
1h	10uS	12.2uS
2h	15uS	17.2uS
3h	25uS	27.2uS
4h	50uS	52.2uS
5h	100uS	102.2uS
6h	150uS	152.2uS
7h	200uS	202.2uS
8h	250uS	252.2uS
9h	300uS	302.2uS
Ah	350uS	352.2uS
Bh	400uS	402.2uS
Ch	450uS	452.2uS
Dh	500uS	502.2uS
Eh	550uS	552.2uS
Fh	600uS	602.2uS

5.3.23 E2P_CMD – EEPROM Command Register

Offset: B0h Size: 32 bits

This register is used to control the read and write operations with the Serial EEPROM.

BITS	DESCRIPTION	TYPE	DEFAULT
31	**EPC Busy:** When a 1 is written into this bit, the operation specified in the EPC command field is performed at the specified EEPROM address. This bit will remain set until the operation is complete. In the case of a read this means that the host can read valid data from the E2P data register. The E2P_CMD and E2P_DATA registers should not be modified until this bit is cleared. In the case where a write is attempted and an EEPROM is not present, the EPC Busy remains busy until the EPC Time-out occurs. At that time the busy bit is cleared. **Note:** EPC busy will be high immediately following power-up or reset. After the EEPROM controller has finished reading (or attempting to read) the MAC address from the EEPROM the EPC Busy bit is cleared.	SC	0

<table>
<tr><th>BITS</th><th>DESCRIPTION</th><th>TYPE</th><th>DEFAULT</th></tr>
<tr><td>30-28</td><td>
EPC command. This field is used to issue commands to the EEPROM controller. The EPC will execute commands when the EPC Busy bit is set. A new command must not be issued until the previous command completes. This field is encoded as follows:
<table>
<tr><th>[30]</th><th>[29]</th><th>[28]</th><th>OPERATION</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>READ</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>EWDS</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>EWEN</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>WRITE</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>WRAL</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>ERASE</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>ERAL</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Reload</td></tr>
</table>
READ (Read Location): This command will cause a read of the EEPROM location pointed to by EPC Address. The result of the read is available in the E2P_DATA register.

EWDS (Erase/Write Disable): After issued, the EEPROM will ignore erase and write commands. To re-enable erase/write operations issue the EWEN command.

EWEN (Erase/Write Enable): Enables the EEPROM for erase and write operations. The EEPROM will allow erase and write operations until the Erase/Write Disable command is sent, or until power is cycled.
Note: The EEPROM device will power-up in the erase/write-disabled state. Any erase or write operations will fail until an Erase/Write Enable command is issued.

WRITE (Write Location): If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to the EEPROM location selected by the EPC Address field.

WRAL (Write All): If erase/write operations are enabled in the EEPROM, this command will cause the contents of the E2P_DATA register to be written to every EEPROM memory location.

ERASE (Erase Location): If erase/write operations are enabled in the EEPROM, this command will erase the location selected by the EPC Address field.

ERAL (Erase All): If erase/write operations are enabled in the EEPROM, this command will initiate a bulk erase of the entire EEPROM.

RELOAD (MAC Address Reload): Instructs the EEPROM controller to reload the MAC address from the EEPROM. If a value of 0xA5 is not found in the first address of the EEPROM, the EEPROM is assumed to be un-programmed and MAC Address Reload operation will fail. The “MAC Address Loaded” bit indicates a successful load of the MAC address.
</td><td>R/W</td><td>0</td></tr>
<tr><td>27-10</td><td>Reserved.</td><td>RO</td><td>-</td></tr>
</table>

BITS	DESCRIPTION	TYPE	DEFAULT
9	**EPC Time-out.** If an EEPROM operation is performed, and there is no response from the EEPROM within 30mS, the EEPROM controller will time-out and return to its idle state. This bit is set when a time-out occurs indicating that the last operation was unsuccessful. **Note:** If the EEDIO signal pin is externally pulled-high, EPC commands will not time out if the EEPROM device is missing. In this case the EPC Busy bit will be cleared as soon as the command sequence is complete. It should also be noted that the ERASE, ERAL, WRITE and WRAL commands are the only EPC commands that will time-out if an EEPROM device is not present -*and*- the EEDIO signal is pulled low	R/WC	0
8	**MAC Address Loaded.** When set, this bit indicates that a valid EEPROM was found, and that the MAC address programming has completed normally. This bit is set after a successful load of the MAC address after power-up, or after a RELOAD command has completed	R/WC	-
7-0	**EPC Address.** The 8-bit value in this field is used by the EEPROM Controller to address the specific memory location in the Serial EEPROM. This is a Byte aligned address.	R/W	00h

5.3.24 E2P_DATA – EEPROM Data Register

Offset: B4h Size: 32 bits

This register is used in conjunction with the E2P_CMD register to perform read and write operations with the Serial EEPROM.

BITS	DESCRIPTION	TYPE	DEFAULT
31-8	**Reserved**	RO	-
7:0	**EEPROM Data.** Value read from or written to the EEPROM.	R/W	00h

5.4 MAC Control and Status Registers

These registers are located in the MAC module and are accessed indirectly through the MAC-CSR synchronizer port. Table 5.6, "MAC CSR Register Map", shown below, lists the MAC registers that are accessible through the indexing method using the MAC_CSR_CMD and MAC_CSR_DATA registers (see sections MAC_CSR_CMD – MAC CSR Synchronizer Command Register and MAC_CSR_DATA – MAC CSR Synchronizer Data Register).

Table 5.6 MAC CSR Register Map

MAC CONTROL AND STATUS REGISTERS			
INDEX	**SYMBOL**	**REGISTER NAME**	**DEFAULT**
1	MAC_CR	MAC Control Register	00040000h
2	ADDRH	MAC Address High	0000FFFFh
3	ADDRL	MAC Address Low	FFFFFFFFh
4	HASHH	Multicast Hash Table High	00000000h
5	HASHL	Multicast Hash Table Low	00000000h
6	MII_ACC	MII Access	00000000h
7	MII_DATA	MII Data	00000000h
8	FLOW	Flow Control	00000000h
9	VLAN1	VLAN1 Tag	00000000h
A	VLAN2	VLAN2 Tag	00000000h
B	WUFF	Wake-up Frame Filter	00000000h
C	WUCSR	Wake-up Control and Status	00000000h
D	COE_CR	Checksum Offload Engine Control	00000000h

5.4.1 MAC_CR—MAC Control Register

Offset:	1	Attribute:	R/W
Default Value:	00040000h	Size:	32 bits

This register establishes the RX and TX operation modes and controls for address filtering and packet filtering.

BITS	DESCRIPTION
31	**Receive All Mode (RXALL).** When set, all incoming packets will be received and passed on to the address filtering Function for processing of the selected filtering mode on the received frame. Address filtering then occurs and is reported in Receive Status. When reset, only frames that pass Destination Address filtering will be sent to the Application.
30-24	**Reserved**
23	**Disable Receive Own (RCVOWN).** When set, the MAC disables the reception of frames when the MII TX_EN signal is asserted. The MAC blocks the transmitted frame on the receive path. When reset, the MAC receives all packets the PHY gives, including those transmitted by the MAC.This bit should be reset when the Full Duplex Mode bit is set.
22	**Reserved**
21	**Loopback operation Mode (LOOPBK).** Selects the loop back operation modes for the MAC. This is only for full duplex mode 1'b0: Normal: No feedback 1'b1: Internal: Through MII In internal loopback mode, the TX frame is received by the Internal MII interface, and sent back to the MAC without being sent to the PHY. **Note:** When enabling or disabling the loopback mode it can take up to 10μs for the mode change to occur. The transmitter and receiver must be stopped and disabled when modifying the LOOPBK bit. The transmitter or receiver should not be enabled within10μs of modifying the LOOPBK bit.
20	**Full Duplex Mode (FDPX).** When set, the MAC operates in Full-Duplex mode, in which it can transmit and receive simultaneously. In Full-Duplex mode, the heartbeat check is disabled and the heartbeat fail status should thus be ignored.
19	**Pass All Multicast (MCPAS).** When set, indicates that all incoming frames with a Multicast destination address (first bit in the destination address field is 1) are received. Incoming frames with physical address (Individual Address/Unicast) destinations are filtered and received only if the address matches the MAC Address.
18	**Promiscuous Mode (PRMS).** When set, indicates that any incoming frame is received regardless of its destination address.
17	**Inverse filtering (INVFILT).** When set, the address check Function operates in Inverse filtering mode. This is valid only during Perfect filtering mode.
16	**Pass Bad Frames (PASSBAD).** When set, all incoming frames that passed address filtering are received, including runt frames and collided frames.
15	**Hash Only Filtering mode (HO).** When set, the address check Function operates in the Imperfect Address Filtering mode both for physical and multicast addresses
14	**Reserved**

BITS	DESCRIPTION
13	**Hash/Perfect Filtering Mode (HPFILT).** When reset (0), the LAN9210 will implement a perfect address filter on incoming frames according the address specified in the MAC address register. When set (1), the address check Function does imperfect address filtering of multicast incoming frames according to the hash table specified in the multicast hash table register. If the Hash Only Filtering mode (HO) bit is set (1), then the physical (IA) are imperfect filtered too. If the Hash Only Filtering mode (HO) bit is reset (0), then the IA addresses are perfect address filtered according to the MAC Address register
12	**Late Collision Control (LCOLL).** When set, enables retransmission of the collided frame even after the collision period (late collision). When reset, the MAC disables frame transmission on a late collision. In any case, the Late Collision status is appropriately updated in the Transmit Packet status.
11	**Disable Broadcast Frames (BCAST).** When set, disables the reception of broadcast frames. When reset, forwards all broadcast frames to the application. **Note:** When wake-up frame detection is enabled via the WUEN bit of the WUCSR—Wake-up Control and Status Register, a broadcast wake-up frame will wake-up the device despite the state of this bit.
10	**Disable Retry (DISRTY).** When set, the MAC attempts only one transmission. When a collision is seen on the bus, the MAC ignores the current frame and goes to the next frame and a retry error is reported in the Transmit status. When reset, the MAC attempts 16 transmissions before signaling a retry error.
9	**Reserved**
8	**Automatic Pad Stripping (PADSTR).** When set, the MAC strips the pad field on all incoming frames, if the length field is less than 46 bytes. The FCS field is also stripped, since it is computed at the transmitting station based on the data and pad field characters, and is invalid for a received frame that has had the pad characters stripped. Receive frames with a 46-byte or greater length field are passed to the Application unmodified (FCS is not stripped). When reset, the MAC passes all incoming frames to the host unmodified. **Note:** When PADSTR is enabled, the RX Checksum Offload Engine must be disabled (bit 0 (RXCOE_EN) of the COE_CR—Checksum Offload Engine Control Register) and vice versa. These functions cannot be enabled simultaneously.

<table>
<tr><th>BITS</th><th>DESCRIPTION</th></tr>
<tr><td>7-6</td><td>BackOff Limit (BOLMT). The BOLMT bits allow the user to set its back-off limit in a relaxed or aggressive mode. According to IEEE 802.3, the MAC has to wait for a random number [r] of slot-times** after it detects a collision, where:
(eq.1)0 < r < $_2$K
The exponent K is dependent on how many times the current frame to be transmitted has been retried, as follows:
(eq.2)K = min (n, 10) where n is the current number of retries.
If a frame has been retried three times, then K = 3 and r= 8 slot-times maximum. If it has been retried 12 times, then K = 10, and r = 1024 slot-times maximum.
An LFSR (linear feedback shift register) 20-bit counter emulates a 20bit random number generator, from which r is obtained. Once a collision is detected, the number of the current retry of the current frame is used to obtain K (eq.2). This value of K translates into the number of bits to use from the LFSR counter. If the value of K is 3, the MAC takes the value in the first three bits of the LFSR counter and uses it to count down to zero on every slot-time. This effectively causes the MAC to wait eight slot-times. To give the user more flexibility, the BOLMT value forces the number of bits to be used from the LFSR counter to a predetermined value as in the table below.

<table>
<tr><th>BOLMT Value</th><th># Bits Used from LFSR Counter</th></tr>
<tr><td>2’b00</td><td>10</td></tr>
<tr><td>2’b01</td><td>8</td></tr>
<tr><td>2’b10</td><td>4</td></tr>
<tr><td>2’b11</td><td>1</td></tr>
</table>

Thus, if the value of K = 10, the MAC will look at the BOLMT if it is 00, then use the lower ten bits of the LFSR counter for the wait countdown. If the BOLMT is 10, then it will only use the value in the first four bits for the wait countdown, etc.
**Slot-time = 512 bit times. (See IEEE 802.3 Spec., Secs. 4.2.3.25 and 4.4.2.1)</td></tr>
<tr><td>5</td><td>Deferral Check (DFCHK). When set, enables the deferral check in the MAC. The MAC will abort the transmission attempt if it has deferred for more than 24,288 bit times. Deferral starts when the transmitter is ready to transmit, but is prevented from doing so because the CRS is active. Defer time is not cumulative. If the transmitter defers for 10,000 bit times, then transmits, collides, backs off, and then has to defer again after completion of back-off, the deferral timer resets to 0 and restarts. When reset, the deferral check is disabled in the MAC and the MAC defers indefinitely.</td></tr>
<tr><td>4</td><td>Reserved</td></tr>
<tr><td>3</td><td>Transmitter enable (TXEN). When set, the MAC’s transmitter is enabled and it will transmit frames from the buffer onto the cable.
When reset, the MAC’s transmitter is disabled and will not transmit any frames.</td></tr>
<tr><td>2</td><td>Receiver Enable (RXEN). When set (1), the MAC’s receiver is enabled and will receive frames from the internal PHY.
When reset, the MAC’s receiver is disabled and will not receive any frames from the internal PHY.</td></tr>
<tr><td>1-0</td><td>Reserved</td></tr>
</table>

5.4.2 ADDRH—MAC Address High Register

Offset:	2	Attribute:	R/W
Default Value:	0000FFFFh	Size:	32 bits

The MAC Address High register contains the upper 16-bits of the physical address of the MAC. The contents of this register are optionally loaded from the EEPROM at power-on through the EEPROM Controller if a programmed EEPROM is detected. The least significant byte of this register (bits [7:0]) is loaded from address 0x05 of the EEPROM. The second byte (bits [15:8]) is loaded from address 0x06 of the EEPROM. Please refer to Section 4.6 for more information on the EEPROM. Section 5.4.3 details the byte ordering of the ADDRL and ADDRH registers with respect to the reception of the Ethernet physical address.

BITS	DESCRIPTION
31-16	**Reserved**
15-0	**Physical Address [47:32].** This field contains the upper 16-bits (47:32) of the Physical Address of the LAN9210 device. The content of this field is undefined until loaded from the EEPROM at power-on. The host can update the contents of this field after the initialization process has completed.

5.4.3 ADDRL—MAC Address Low Register

Offset:	3	Attribute:	R/W
Default Value:	FFFFFFFFh	Size:	32 bits

The MAC Address Low register contains the lower 32 bits of the physical address of the MAC. The contents of this register are optionally loaded from the EEPROM at power-on through the EEPROM Controller if a programmed EEPROM is detected. The least significant byte of this register (bits [7:0]) is loaded from address 0x01 of the EEPROM. The most significant byte of this register is loaded from address 0x04 of the EEPROM. Please refer to Section 4.6 for more information on the EEPROM.

BITS	DESCRIPTION
31-0	**Physical Address [31:0].** This field contains the lower 32 bits (31:0) of the Physical Address of the LAN9210 device. The content of this field is undefined until loaded from the EEPROM at power-on. The host can update the contents of this field after the initialization process has completed.

Table 5.7 below illustrates the byte ordering of the ADDRL and ADDRH registers with respect to the reception of the Ethernet physical address. Also shown is the correlation between the EEPROM addresses and ADDRL and ADDRH registers.

Table 5.7 ADDRL, ADDRH and EEPROM Byte Ordering

EEPROM ADDRESS	ADDRN	ORDER OF RECEPTION ON ETHERNET
0x01	ADDRL[7:0]	1st
0x02	ADDRL[15:8]	2nd
0x03	ADDRL[23:16]	3rd
0x04	ADDRL[31:24]	4th
0x05	ADDRH[7:0]	5th
0x06	ADDRH[15:8]	6th

As an example, if the desired Ethernet physical address is 12-34-56-78-9A-BC, the ADDRL and ADDRH registers would be programmed as shown in Figure 5.2. The values required to automatically load this configuration from the EEPROM are also shown.

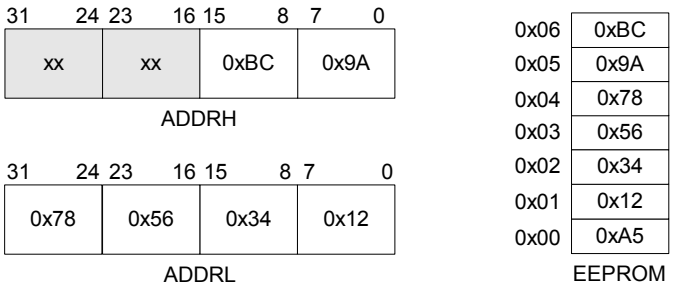


Figure 5.2 Example ADDRL, ADDRH and EEPROM Setup

Note: By convention, the left most byte of the Ethernet address (in this example 0x12) is the most significant byte and is transmitted/received first.

5.4.4 HASHH—Multicast Hash Table High Register

Offset:	4	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

The 64-bit Multicast table is used for group address filtering. For hash filtering, the contents of the destination address in the incoming frame is used to index the contents of the Hash table. The most significant bit determines the register to be used (Hi/Low), while the other five bits determine the bit within the register. A value of 00000 selects Bit 0 of the Multicast Hash Table Lo register and a value of 11111 selects the Bit 31 of the Multicast Hash Table Hi register.

If the corresponding bit is 1, then the multicast frame is accepted. Otherwise, it is rejected. If the “Pass All Multicast” (MCPAS) bit is set (1), then all multicast frames are accepted regardless of the multicast hash values.

The Multicast Hash Table Hi register contains the higher 32 bits of the hash table and the Multicast Hash Table Low register contains the lower 32 bits of the hash table.

BITS	DESCRIPTION
31-0	**Upper 32 bits of the 64-bit Hash Table**

5.4.5 HASHL—Multicast Hash Table Low Register

Offset:	5	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register defines the lower 32-bits of the Multicast Hash Table. Please refer to Table 5.4.4, "HASHH—Multicast Hash Table High Register" for further details.

BITS	DESCRIPTION
31-0	**Lower 32 bits of the 64-bit Hash Table**

5.4.6 MII_ACC—MII Access Register

Offset:	6	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register is used to control the Management cycles to the PHY.

BITS	DESCRIPTION
31-16	**Reserved**
15-11	**PHY Address:** For every access to this register, this field must be set to 00001b.
10-6	**MII Register Index (MIIRINDA):** These bits select the desired MII register in the PHY.
5-2	**Reserved**
1	**MII Write (MIIWnR):** Setting this bit tells the PHY that this will be a write operation using the MII data register. If this bit is not set, this will be a read operation, packing the data in the MII data register.
0	**MII Busy (MIIBZY):** This bit must be polled to determine when the MII register access is complete. This bit must read a logical 0 before writing to this register and MII data register. The LAN driver software must set (1) this bit in order for the LAN9210 to read or write any of the MII PHY registers. During a MII register access, this bit will be set, signifying a read or write access is in progress. The MII data register must be kept valid until the MAC clears this bit during a PHY write operation. The MII data register is invalid until the MAC has cleared this bit during a PHY read operation.

5.4.7 MII_DATA—MII Data Register

Offset:	7	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register contains either the data to be written to the PHY register specified in the MII Access Register, or the read data from the PHY register whose index is specified in the MII Access Register.

BITS	DESCRIPTION
31-16	**Reserved**
15-0	**MII Data.** This contains the 16-bit value read from the PHY read operation or the 16-bit data value to be written to the PHY before an MII write operation.

5.4.8 FLOW—Flow Control Register

Offset:	8	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register controls the generation and reception of the Control (Pause command) frames by the MAC's flow control block. The control frame fields are selected as specified in the 802.3x Specification and the Pause-Time value from this register is used in the "Pause Time" field of the control frame. In full-duplex mode the FCBSY bit is set until the control frame is transferred onto the cable. In half-duplex mode FCBSY is set while back pressure is being asserted. The host has to make sure that the Busy bit is cleared before writing the register. The Pass Control Frame bit (FCPASS) does not affect the sending of the frames, including Control Frames, to the Application Interface. The Flow Control Enable (FCEN) bit enables the receive portion of the Flow Control block.

This register is used in conjunction with the AFC_CFG register in the Slave CSRs to configure flow control. Software flow control is initiated using the AFC_CFG register.

Note: The LAN9210 will not transmit pause frames or assert back pressure if the transmitter is disabled.

BITS	DESCRIPTION
31-16	**Pause Time (FCPT).** This field indicates the value to be used in the PAUSE TIME field in the control frame. This field must be initialized before full-duplex automatic flow control is enabled.
15-3	**Reserved**
2	**Pass Control Frames (FCPASS).** When set, the MAC will pass the pause frame to the host. The Application must accept or discard a received frame based on the Packet Filter control bit. The MAC receives, decodes and performs the Pause function when a valid Pause frame is received in Full-Duplex mode and when flow control is enabled (FCE bit set). When reset, the MAC resets the Packet Filter bit in the Receive packet status. The MAC always passes the data of all frames it receives (including Flow Control frames) to the Application. Frames that do not pass Address filtering, as well as frames with errors, are passed to the Application. The Application must discard or retain the received frame's data based on the received frame's STATUS field. Filtering modes (Promiscuous mode, for example) take precedence over the FCPASS bit.
1	**Flow Control Enable (FCEN).** When set, enables the MAC Flow Control function. The MAC decodes all incoming frames for control frames; if it receives a valid control frame (PAUSE command), it disables the transmitter for a specified time (Decoded pause time x slot time). When reset, the MAC flow control function is disabled; the MAC does not decode frames for control frames. **Note:** Flow Control is applicable when the MAC is set in Full Duplex Mode. In Half-Duplex mode, this bit enables the Backpressure function to control the flow of received frames to the MAC.
0	**Flow Control Busy (FCBSY).** This bit is set high whenever a pause frame or back pressure is being transmitted. This bit should read logical 0 before writing to the Flow Control (FLOW) register. During a transfer of Control Frame, this bit continues to be set, signifying that a frame transmission is in progress. After the PAUSE control frame's transmission is complete, the MAC resets to 0. **Notes:** ▪ When writing this register the FCBSY bit must always be zero. ▪ Applications must always write a zero to this bit

5.4.9 VLAN1—VLAN1 Tag Register

Offset:	9	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register contains the VLAN tag field to identify VLAN1 frames. For VLAN frames the legal frame length is increased from 1518 bytes to 1522 bytes.

BITS	DESCRIPTION
31-16	**Reserved**
15-0	**VLAN1 Tag Identifier (VTI1).** This contains the VLAN Tag field to identify the VLAN1 frames. This field is compared with the 13th and 14th bytes of the incoming frames for VLAN1 frame detection. If used, this register must be set to 0x8100.

5.4.10 VLAN2—VLAN2 Tag Register

Offset:	A	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register contains the VLAN tag field to identify VLAN2 frames. For VLAN frames the legal frame length is increased from 1518 bytes to 1522 bytes.

BITS	DESCRIPTION
31-16	**Reserved**
15-0	**VLAN2 Tag Identifier (VTI2).** This contains the VLAN Tag field to identify the VLAN2 frames. This field is compared with the 13th and 14th bytes of the incoming frames for VLAN2 frame detection.If used, this register must be set to 0x8100.

5.4.11 WUFF—Wake-up Frame Filter

Offset:	B	Attribute:	WO
Default Value:	00000000h	Size:	32 bits

This register is used to configure the wake up frame filter.

BITS	DESCRIPTION
31-0	**Wake-Up Frame Filter (WFF).** Wake-Up Frame Filter (WFF). The Wake-up frame filter is configured through this register using an indexing mechanism. After hardware reset, or soft reset, the MAC loads the first value written to this location to the first DWORD in the Wake-up frame filter (filter 0 byte mask). The second value written to this location is loaded to the second DWORD in the wake-up frame filter (filter 1 byte mask) and so on. Once all eight DWORDs have been written, the internal pointer will once again point to the first entry and the filter entries can be modified in the same manner. **Note:** This is a write-only register.

5.4.12 WUCSR—Wake-up Control and Status Register

Offset:	C	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register contains data pertaining to the MAC's remote wake-up status and capabilities.

BITS	DESCRIPTION
31-10	**Reserved**
9	**Global Unicast Enable (GUE).** When set, the MAC wakes up from power-saving mode on receipt of a global unicast frame. A global unicast frame has the MAC Address [0] bit set to 0.
8-7	**Reserved**
6	**Remote Wake-Up Frame Received (WUFR).** The MAC, upon receiving a valid Remote Wake-up frame, sets this bit.
5	**Magic Packet Received (MPR).** The MAC, upon receiving a valid Magic Packet, sets this bit.
4-3	**Reserved**
2	**Wake-Up Frame enabled (WUEN).** When set, Remote Wake-Up mode is enabled and the MAC is capable of detecting wake-up frames as programmed in the wake-up frame filter.
1	**Magic Packet Enable (MPEN).** When set, Magic Packet Wake-up mode is enabled.
0	**Reserved**

5.4.13 COE_CR—Checksum Offload Engine Control Register

Offset:	D	Attribute:	R/W
Default Value:	00000000h	Size:	32 bits

This register controls the transmit and receive checksum offload engines.

BITS	DESCRIPTION
31-17	**Reserved**
16	**TX Checksum Offload Engine Enable (TXCOE_EN).** This bit enables/disables the Transmit COE. This bit may only be changed if the TX data path is disabled. 0: The TXCOE is bypassed 1: The TXCOE is enabled
15-2	**Reserved**
1	**RX Checksum Offload Engine Mode (RXCOE_MODE)** This register indicates whether the RXCOE will check for VLAN tags or a SNAP header prior to beginning its checksum calculation. In its default mode, the calculation will always begin 14 bytes into the frame. The RXCOE_MODE may only be changed if the ESS RX path is disabled. 0: Begin checksum calculation after first 14 bytes of Ethernet Frame 1: Begin checksum calculation at start of L3 packet by adjusting for VLAN tags and/or SNAP header.
0	**RX Checksum Offload Engine Enable (RXCOE_EN).** This bit enables/disables the Receive COE. This bit may only be changed if the RX data path is disabled. 0: The RXCOE is bypassed 1: The RXCOE is enabled **Note:** When the RXCOE is enabled, automatic pad stripping must be disabled (bit 8 (PADSTR) of the MAC_CR—MAC Control Register) and vice versa. These functions cannot be enabled simultaneously.

5.5 PHY Registers

The PHY registers are not memory mapped. These registers are accessed indirectly through the MAC via the MII_ACC and MII_DATA registers. An index must be used to access individual PHY registers. PHY Register Indexes are shown in Table 5.8, "LAN9210 PHY Control and Status Register".

Note: The NASR (**Not Affected by Software Reset)** designation is only applicable when bit 15 of the PHY Basic Control Register (Reset) is set.

Table 5.8 LAN9210 PHY Control and Status Register

PHY CONTROL AND STATUS REGISTERS	
INDEX (IN DECIMAL)	**REGISTER NAME**
0	Basic Control Register
1	Basic Status Register
2	PHY Identifier 1
3	PHY Identifier 2
4	Auto-Negotiation Advertisement Register
5	Auto-Negotiation Link Partner Ability Register
6	Auto-Negotiation Expansion Register
17	Mode Control/Status Register
18	Special Modes Register
27	Special Control/Status Indications
29	Interrupt Source Register
30	Interrupt Mask Register
31	PHY Special Control/Status Register

5.5.1 Basic Control Register

Index (In Decimal): 0 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15	**Reset.** 1 = software reset. Bit is self-clearing. For best results, when setting this bit do not set other bits in this register.	RW/SC	0
14	**Loopback.** 1 = loopback mode, 0 = normal operation	RW	0
13	**Speed Select.** 1 = 100Mbps, 0 = 10Mbps. Ignored if Auto Negotiation is enabled (0.12 = 1).	RW	See Note 5.1
12	**Auto-Negotiation Enable.** 1 = enable auto-negotiate process (overrides 0.13 and 0.8) 0 = disable auto-negotiate process.	RW	1
11	**Power Down.** 1 = General power down-mode, 0 = normal operation. **Note:** After this bit is cleared, the PHY may auto-negotiate with it's partner station. This process may take a few seconds to complete. Once auto-negotiation is complete, bit 5 of the PHY's Basic Status Register will be set.	RW	0
10	**Reserved**	RO	0
9	**Restart Auto-Negotiate.** 1 = restart auto-negotiate process 0 = normal operation. Bit is self-clearing.	RW/SC	0
8	**Duplex Mode.** 1 = full duplex, 0 = half duplex. Ignored if Auto Negotiation is enabled (0.12 = 1).	RW	See Note 5.1
7	**Collision Test.** 1 = enable COL test, 0 = disable COL test	RW	0
6-0	**Reserved**	RO	0

Note 5.1 The default value of this bit is determined by the auto-negotiation process.

5.5.2 Basic Status Register

Index (In Decimal): 1 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15	**100Base-T4.** 1 = T4 able, 0 = no T4 ability	RO	0
14	**100Base-TX Full Duplex.** 1 = TX with full duplex, 0 = no TX full duplex ability.	RO	1
13	**100Base-TX Half Duplex.** 1 = TX with half duplex, 0 = no TX half duplex ability.	RO	1
12	**10Base-T Full Duplex.** 1 = 10Mbps with full duplex 0 = no 10Mbps with full duplex ability	RO	1
11	**10Base-T Half Duplex.** 1 = 10Mbps with half duplex 0 = no 10Mbps with half duplex ability	RO	1
10-6	**Reserved**	RO	0
5	**Auto-Negotiate Complete.** 1 = auto-negotiate process completed 0 = auto-negotiate process not completed	RO	0
4	**Remote Fault.** 1 = remote fault condition detected 0 = no remote fault	RO/LH	0
3	**Auto-Negotiate Ability.** 1 = able to perform auto-negotiation function 0 = unable to perform auto-negotiation function	RO	1
2	**Link Status.** 1 = link is up, 0 = link is down	RO/LL	0
1	**Jabber Detect.** 1 = jabber condition detected 0 = no jabber condition detected	RO/LH	0
0	**Extended Capabilities.** 1 = supports extended capabilities registers 0 = does not support extended capabilities registers.	RO	1

5.5.3 PHY Identifier 1

Index (In Decimal): 2 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15-0	**PHY ID Number.** Assigned to the 3rd through 18th bits of the Organizationally Unique Identifier (OUI), respectively.	RO	0x0007h

5.5.4 PHY Identifier 2

Index (In Decimal): 3 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15-10	**PHY ID Number** Assigned to the 19th through 24th bits of the OUI.	RO	0xC0C3h
9 - 4	**Model Number.** Six-bit manufacturer's model number.	RO	
3 - 0	**Revision Number.** Four-bit manufacturer's revision number.	RO	

5.5.5 Auto-negotiation Advertisement

Index (In Decimal): 4 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:14	**Reserved**	RO	00
13	**Remote Fault.** 1 = remote fault detected, 0 = no remote fault	R/W	0
12	**Reserved**	R/W	0
11:10	**Pause Operation.** (See Note 5.2) 00 No PAUSE 01 Symmetric PAUSE 10 Asymmetric PAUSE 11 Advertise support for both Symmetric PAUSE and Asymmetric PAUSE	R/W	00
9	**Reserved**	RO	0
8	**100Base-TX Full Duplex.** 1 = TX with full duplex, 0 = no TX full duplex ability	R/W	1
7	**100Base-TX.** 1 = TX able, 0 = no TX ability	R/W	1
6	**10Base-T Full Duplex.** 1 = 10Mbps with full duplex 0 = no 10Mbps with full duplex ability	R/W	1
5	**10Base-T.** 1 = 10Mbps able, 0 = no 10Mbps ability	R/W	1
4:0	**Selector Field.** [00001] = IEEE 802.3	R/W	00001

Note 5.2 When both symmetric PAUSE and asymmetric PAUSE support are advertised (value of 11), the device will only be configured to, at most, one of the two settings upon auto-negotiation completion.

5.5.6 Auto-negotiation Link Partner Ability

Index (In Decimal): 5 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15	**Next Page.** 1 = next page capable, 0 = no next page ability. This device does not support next page ability.	RO	0
14	**Acknowledge.** 1 = link code word received from partner 0 = link code word not yet received **Note:** This bit will always read 0	RO	0
13	**Remote Fault.** 1 = remote fault detected, 0 = no remote fault	RO	0
12	**Reserved**	RO	0
11-10	**Pause Operation**. 00 No PAUSE supported by partner station 01 Symmetric PAUSE supported by partner station 10 Asymmetric PAUSE supported by partner station 11 Both Symmetric PAUSE and Asymmetric PAUSE supported by partner station	RO	00
9	**100Base-T4.** 1 = T4 able, 0 = no T4 ability	RO	0
8	**100Base-TX Full Duplex.** 1 = TX with full duplex, 0 = no TX full duplex ability	RO	0
7	**100Base-TX.** 1 = TX able, 0 = no TX ability	RO	0
6	**10Base-T Full Duplex.** 1 = 10Mbps with full duplex 0 = no 10Mbps with full duplex ability	RO	0
5	**10Base-T.** 1 = 10Mbps able, 0 = no 10Mbps ability	RO	0
4:0	**Selector Field.** [00001] = IEEE 802.3	RO	00001

5.5.7 Auto-negotiation Expansion

Index (In Decimal): 6 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15:5	**Reserved**	RO	0
4	**Parallel Detection Fault.** 1 = fault detected by parallel detection logic 0 = no fault detected by parallel detection logic	RO/LH	0
3	**Link Partner Next Page Able.** 1 = link partner has next page ability 0 = link partner does not have next page ability	RO	0
2	**Next Page Able.** 1 = local device has next page ability 0 = local device does not have next page ability	RO	0
1	**Page Received.** 1 = new page received 0 = new page not yet received	RO/LH	0
0	**Link Partner Auto-Negotiation Able.** 1 = link partner has auto-negotiation ability 0 = link partner does not have auto-negotiation ability	RO	0

5.5.8 Mode Control/Status

Index (In Decimal): 17 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15-14	**Reserved.** Write as 0; ignore on read.	RW	0
13	**EDPWRDOWN.** Enable the Energy Detect Power-Down mode: 0=Energy Detect Power-Down is disabled 1=Energy Detect Power-Down is enabled	RW	0
12-2	**Reserved.** Write as 0, ignore on read	RW	0
1	**ENERGYON.** Indicates whether energy is detected. This bit goes to a “0” if no valid energy is detected within 256ms. Reset to “1” by hardware reset, unaffected by SW reset.	RO	See Note 5.3
0	**Reserved.** Write as “0”. Ignore on read.	RW	0

Note 5.3 The default value of this bit will vary dependant on the current link state of the line.

5.5.9 Special Modes

Index (In Decimal): 18 Size: 16-bits

ADDRESS	DESCRIPTION	TYPE	DEFAULT
15-8	**Reserved**	RW, NASR	
7:5	**MODE:** PHY Mode of operation. Refer to Table 5.9 for more details.	RW, NASR	111
4:0	**PHYAD: PHY Address:** The PHY Address is used for the SMI address.	RW, NASR	00001b

Table 5.9 MODE Control

MODE	MODE DEFINITIONS	DEFAULT REGISTER BIT VALUES	
		REGISTER 0	REGISTER 4
		[13,12,10,8]	[8,7,6,5]
000	10Base-T Half Duplex. Auto-negotiation disabled.	0000	N/A
001	10Base-T Full Duplex. Auto-negotiation disabled.	0001	N/A
010	100Base-TX Half Duplex. Auto-negotiation disabled. CRS is active during Transmit & Receive.	1000	N/A
011	100Base-TX Full Duplex. Auto-negotiation disabled. CRS is active during Receive.	1001	N/A
100	100ase-TX Half Duplex is advertised. Auto-negotiation enabled. CRS is active during Transmit & Receive.	1100	0100
101	Repeater mode. Auto-negotiation enabled. 100Base-TX Half Duplex is advertised. CRS is active during Receive.	1100	0100
110	Reserved - Do not set the LAN9210 in this mode.	N/A	N/A
111	All capable. Auto-negotiation enabled.	X10X Note 5.4	1111

Note 5.4 When MODE=111, the register 0 bits 13 and 8 are variable dependant on the auto-negotiated speed and duplex.

5.5.10 Special Control/Status Indications

Index (In Decimal): 27 Size: 16-bits

ADDRESS	DESCRIPTION	MODE	DEFAULT
15	**Override AMDIX Strap** 0 - AMDIX_EN (pin 52) enables or disables HP Auto MDIX 1 - Override pin 52. PHY Register 27.14 and 27.13 determine MDIX function	RW	0
14	**Auto-MDIX Enable:** Only effective when 27.15=1, otherwise ignored. 0 = Disable Auto-MDIX. 27.13 determines normal or reversed connection. 1 = Enable Auto-MDIX. 27.13 must be set to 0.	RW	0
13	**Auto-MDIX State.** Only effective when 27.15=1, otherwise ignored. When 27.14 = 0 (manually set MDIX state): 0 = no crossover (TPO = output, TPI = input) 1 = crossover (TPO = input, TPI = output) When 27.14 = 1 (automatic MDIX) this bit must be set to 0. Do not use the combination 27.15=1, 27.14=1, 27.13=1.	RW	0
12:11	**Reserved:** Write as 0. Ignore on read.	RW	0
10	**VCOOFF_LP:** Forces the Receive PLL 10M to lock on the reference clock at all times: 0 - Receive PLL 10M can lock on reference or line as needed (normal operation) 1 - Receive PLL 10M is locked on the reference clock. In this mode 10M data packets cannot be received.	RW, NASR	0
9-5	**Reserved:** Write as 0. Ignore on read.	RW	0
4	**XPOL:** Polarity state of the 10Base-T: 0 - Normal polarity 1 - Reversed polarity	RO	0
3:0	**Reserved:** Read only - Writing to these bits have no effect.	RO	XXXXb

5.5.11 Interrupt Source Flag

Index (In Decimal): 29 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15-8	**Reserved.** Ignore on read.	RO/LH	0
7	**INT7.** 1= ENERGYON generated, 0= not source of interrupt	RO/LH	0
6	**INT6.** 1= Auto-Negotiation complete, 0= not source of interrupt	RO/LH	0
5	**INT5.** 1= Remote Fault Detected, 0= not source of interrupt	RO/LH	0
4	**INT4.** 1= Link Down (link status negated), 0= not source of interrupt	RO/LH	See Note 5.5
3	**INT3.** 1= Auto-Negotiation LP Acknowledge, 0= not source of interrupt	RO/LH	0
2	**INT2.** 1= Parallel Detection Fault, 0= not source of interrupt	RO/LH	0
1	**INT1.** 1= Auto-Negotiation Page Received, 0= not source of interrupt	RO/LH	0
0	**Reserved.**	RO/LH	0

Note 5.5 The default value of this bit will vary dependant on the current link state of the line.

5.5.12 Interrupt Mask

Index (In Decimal): 30 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15-8	**Reserved.** Write as 0; ignore on read.	RO	0
7-0	**Mask Bits.** 1 = interrupt source is enabled 0 = interrupt source is masked	RW	0

5.5.13 PHY Special Control/Status

Index (In Decimal): 31 Size: 16-bits

BITS	DESCRIPTION	TYPE	DEFAULT
15 - 13	**Reserved**	RO	000b
12	**Autodone.** Auto-negotiation done indication: 0 = Auto-negotiation is not done or disabled (or not active) 1 = Auto-negotiation is done	RO	0b
11-5	**Reserved.** Write as 0000010b, ignore on Read.	RW	0000010b
4-2	**Speed Indication.** HCDSPEED value: [001]=10Mbps half-duplex [101]=10Mbps full-duplex [010]=100Base-TX half-duplex [110]=100Base-TX full-duplex	RO	See Note 5.6
1-0	**Reserved.** Write as 0; ignore on Read	RO	00b

Note 5.6 The default value of this bit is determined by the auto-negotiation process.

Chapter 6 Timing Diagrams

6.1 Host Interface Timing

The LAN9210 supports the following host cycles:

Read Cycles:

- PIO Reads (nCS or nRD controlled)
- PIO Burst Reads (nCS or nRD controlled)
- RX Data FIFO Direct PIO Reads (nCS or nRD controlled)
- RX Data FIFO Direct PIO Burst Reads (nCS or nRD controlled)

Write Cycles:

- PIO writes (nCS and nWR controlled)
- TX Data FIFO direct PIO writes (nCS or nWR controlled)

All timing measurements were verified under the following conditions:

Temperature: .....0°C to +70°C

Device VDD:.....+3.30 V +/- 10%

Load Capacitance: 25pF

6.1.1 Special Restrictions on Back-to-Back Write/Read Cycles

It is important to note that there are specific restrictions on the timing of back-to-back write-read operations. These restrictions concern reading the control registers after any write cycle to the LAN9210 device. In many cases there is a required minimum delay between writing to the LAN9210, and the subsequent side effect (change in the control register value). For example, when writing to the TX Data FIFO, it takes up to 165ns for the level indication to change in the TX_FIFO_INF register.

In order to prevent the host from reading stale data after a write operation, minimum wait periods must be enforced. These periods are specified in Table 6.1, "Read After Write Timing Rules". The host processor is required to wait the specified period of time after any write to the LAN9210 before reading the resource specified in the table. These wait periods are for read operations that immediately follow any write cycle. Note that the required wait period is dependant upon the register being read after the write.

Performing "dummy" reads of the BYTE_TEST register is a convenient way to guarantee that the minimum write-to-read timing restriction is met. Table 6.1 also shows the number of dummy reads that are required before reading the register indicated. The number of BYTE_TEST reads in this table is based on the minimum timing for Tcycle (165ns). For microprocessors with slower busses the number of reads may be reduced as long as the total time is equal to, or greater than the time specified in the table. Note that dummy reads of the BYTE_TEST register are not required as long as the minimum time period is met.

Table 6.1 Read After Write Timing Rules

REGISTER NAME	MINIMUM WAIT TIME FOR READ FOLLOWING ANY WRITE CYCLE (IN NS)	NUMBER OF BYTE_TEST READS (ASSUMING T_{CYCLE} OF 165NS)
ID_REV	0	0
IRQ_CFG	165	1
INT_STS	165	1
INT_EN	165	1
BYTE_TEST	0	0
FIFO_INT	165	1
RX_CFG	165	1
TX_CFG	165	1
HW_CFG	165	1
RX_DP_CTRL	165	1
RX_FIFO_INF	0	0
TX_FIFO_INF	165	1
PMT_CTRL	330	2
GPIO_CFG	165	1
GPT_CFG	165	1
GPT_CNT	165	1
WORD_SWAP	165	1
FREE_RUN	330	2
RX_DROP	0	0
MAC_CSR_CMD	165	1
MAC_CSR_DATA	165	1
AFC_CFG	165	1
E2P_CMD	165	1
E2P_DATA	165	1

6.1.2 Special Restrictions on Back-to-Back Read Cycles

There are also restrictions on specific back-to-back read operations. These restrictions concern reading specific registers after reading resources that have side effects. In many cases there is a delay between reading the LAN9210, and the subsequent indication of the expected change in the control register values.

In order to prevent the host from reading stale data on back-to-back reads, minimum wait periods have been established. These periods are specified in Table 6.2, "Read After Read Timing Rules". The host processor is required to wait the specified period of time between read operations of specific combinations of resources. The wait period is dependant upon the combination of registers being read.

Performing "dummy" reads of the BYTE_TEST register is a convenient way to guarantee that the minimum wait time restriction is met. Table 6.2 also shows the number of dummy reads that are required for back-to-back read operations. The number of BYTE_TEST reads in this table is based on the minimum timing for Tcycle (165ns). For microprocessors with slower busses the number of reads may be reduced as long as the total time is equal to, or greater than the time specified in the table. Dummy reads of the BYTE_TEST register are not required as long as the minimum time period is met.

Table 6.2 Read After Read Timing Rules

AFTER READING...	WAIT FOR THIS MANY NS...	OR PERFORM THIS MANY READS OF BYTE_TEST... (ASSUMING Tcycle OF 165NS)	BEFORE READING...
RX Data FIFO	165	1	RX_FIFO_INF
RX Status FIFO	165	1	RX_FIFO_INF
TX Status FIFO	165	1	TX_FIFO_INF
RX_DROP	330	2	RX_DROP
RX_DP_CTRL	330	2	TX Status FIFO RX Status FIFO Note 6.1

Note 6.1 This restriction is only applicable after a fast-forward operation has been completed and the RX_FFWD bit has been cleared. Refer to Section 3.13.1.1, "Receive Data FIFO Fast Forward," on page 62 for more information.

6.2 PIO Reads

PIO reads can be used to access CSRs or RX Data and RX/TX status FIFOs. In this mode, counters in the CSRs are latched at the beginning of the read cycle. Read data is valid as indicated in the timing diagram. PIO reads can be performed using Chip Select (nCS) or Read Enable (nRD). Either or both of these control signals must go high between cycles for the period specified.

Note: Some registers have restrictions on the timing of back-to-back, write-read and read-read cycles.

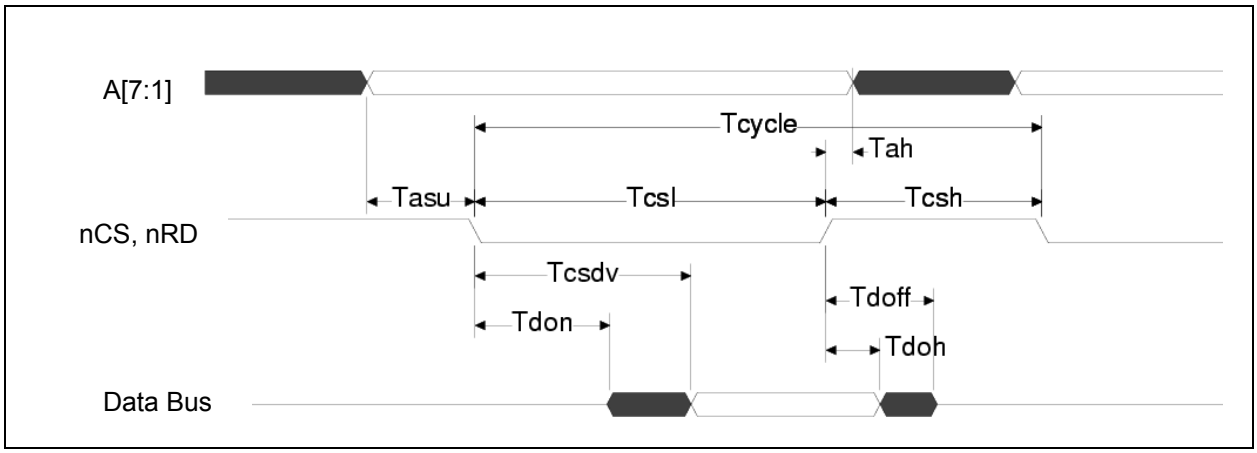


Figure 6.1 PIO Read Cycle Timing

Note: The “Data Bus” width is 16 bits

Table 6.3 PIO Read Timing

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{cycle}	Read Cycle Time	165			ns
t_{csl}	nCS, nRD Assertion Time	32			ns
t_{csh}	nCS, nRD Deassertion Time (see Note below)	13	133		ns
t_{csdv}	nCS, nRD Valid to Data Valid			30	ns
t_{asu}	Address Setup to nCS, nRD Valid	0			ns
t_{ah}	Address Hold Time	0			ns
t_{don}	Data Buffer Turn On Time	0			ns
t_{doff}	Data Buffer Turn Off Time			7	ns
t_{doh}	Data Output Hold Time	0			ns

Note: A PIO Read cycle begins when both nCS and nRD are asserted. The cycle ends when either or both nCS and nRD are deasserted. They may be asserted and deasserted in any order. Parameters t_{csh} and t_{csl} must be extended using wait states to meet the t_{cycle} minimum.

6.3 PIO Burst Reads

In this mode, performance is improved by allowing up to 16 WORD read cycles back-to-back. PIO Burst Reads can be performed using Chip Select (nCS) or Read Enable (nRD). Either or both of these control signals must go high between bursts for the period specified.

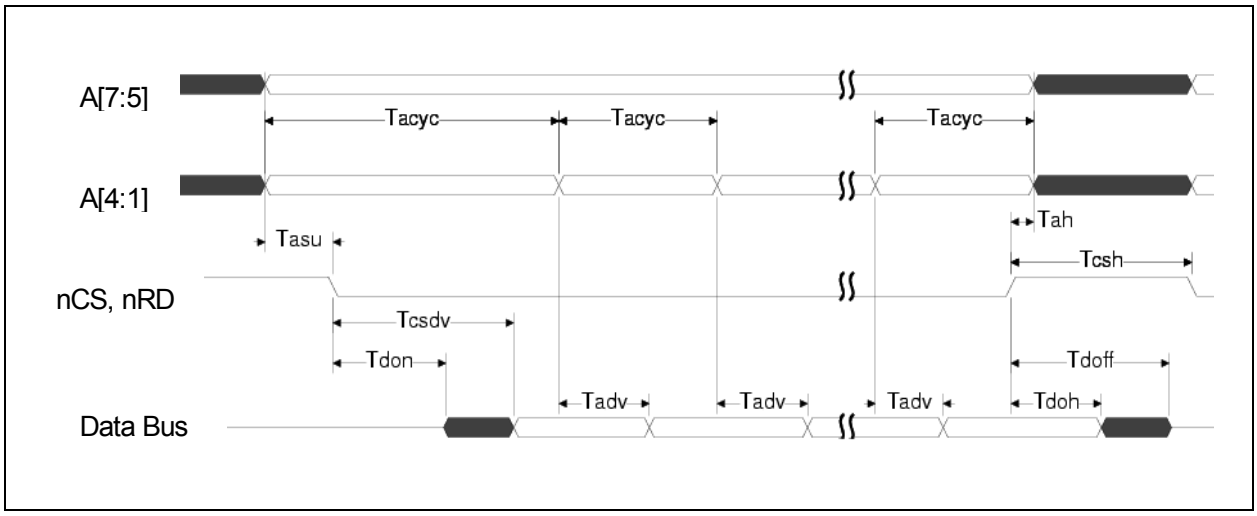


Figure 6.2 PIO Burst Read Cycle Timing

Note: The “Data Bus” width is 16 bits

Table 6.4 PIO Burst Read Timing

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{csh}	nCS, nRD Deassertion Time	13			ns
t_{csdv}	nCS, nRD Valid to Data Valid			30	ns
t_{acyc}	Address Cycle Time	165			
t_{asu}	Address Setup to nCS, nRD valid	0			ns
t_{adv}	Address Stable to Data Valid			40	
t_{ah}	Address Hold Time	0			ns
t_{don}	Data Buffer Turn On Time	0			ns
t_{doff}	Data Buffer Turn Off Time			7	ns
t_{doh}	Data Output Hold Time	0			ns

Note: A PIO Burst Read cycle begins when both nCS and nRD are asserted. The cycle ends when either or both nCS and nRD are deasserted. They may be asserted and deasserted in any order.

6.4 RX Data FIFO Direct PIO Reads

In this mode the upper address inputs are not decoded, and any read of the LAN9210 will read the RX Data FIFO. This mode is enabled when FIFO_SEL is driven high during a read access. This is normally accomplished by connecting the FIFO_SEL signal to high-order address line. This mode is useful when the host processor must increment its address when accessing the LAN9210. Timing is identical to a PIO read, and the FIFO_SEL signal has the same timing characteristics as the address lines.

Note that address lines A[2:1] are still used, and address bits A[7:3] are ignored.

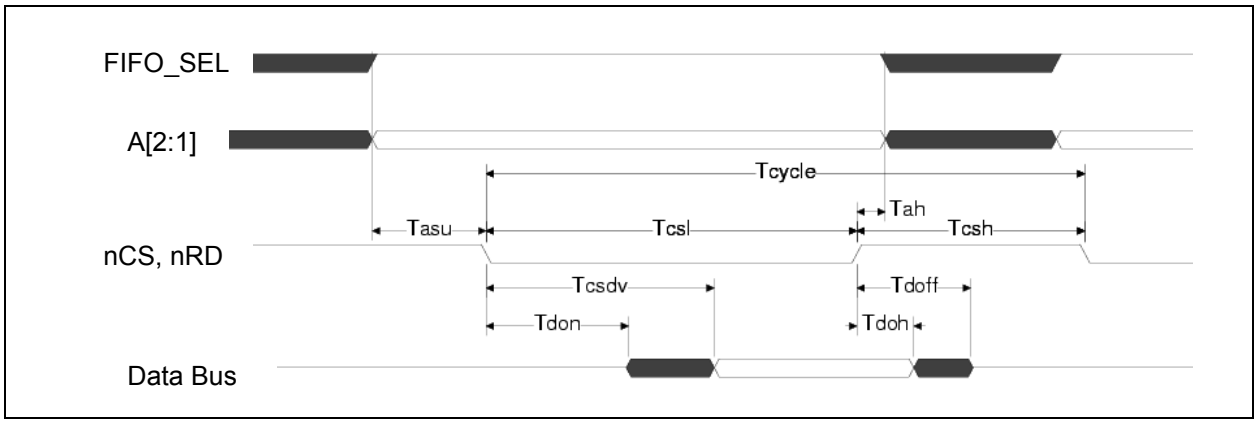


Figure 6.3 RX Data FIFO Direct PIO Read Cycle Timing

Note: The “Data Bus” width is 16 bits.

Table 6.5 RX Data FIFO Direct PIO Read Timing

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{cycle}	Read Cycle Time	165			ns
t_{csl}	nCS, nRD Assertion Time	32			ns
t_{csh}	nCS, nRD Deassertion Time (see Note below)	13	133		ns
t_{csdv}	nCS, nRD Valid to Data Valid			30	ns
t_{asu}	Address, FIFO_SEL Setup to nCS, nRD Valid	0			ns
t_{ah}	Address, FIFO_SEL Hold Time	0			ns
t_{don}	Data Buffer Turn On Time	0			ns
t_{doff}	Data Buffer Turn Off Time			7	ns
t_{doh}	Data Output Hold Time	0			ns

Note: An RX Data FIFO Direct PIO Read cycle begins when both nCS and nRD are asserted. The cycle ends when either or both nCS and nRD are de-asserted. They may be asserted and de-asserted in any order. Parameters t_{csh} and t_{csl} must be extended using wait states to meet the t_{cycle} minimum.

6.5 RX Data FIFO Direct PIO Burst Reads

In this mode the upper address inputs are not decoded, and any burst read of the LAN9210 will read the RX Data FIFO. This mode is enabled when FIFO_SEL is driven high during a read access. This is normally accomplished by connecting the FIFO_SEL signal to a high-order address line. This mode is useful when the host processor must increment its address when accessing the LAN9210. Timing is identical to a PIO Burst Read, and the FIFO_SEL signal has the same timing characteristics as the address lines.

In this mode, performance is improved by allowing an unlimited number of back-to-back read cycles. RX Data FIFO Direct PIO Burst Reads can be performed using Chip Select (nCS) or Read Enable (nRD). When either or both of these control signals go high, they must remain high for the period specified.

Note that address lines A[2:1] are still used, and address bits A[7:3] are ignored.

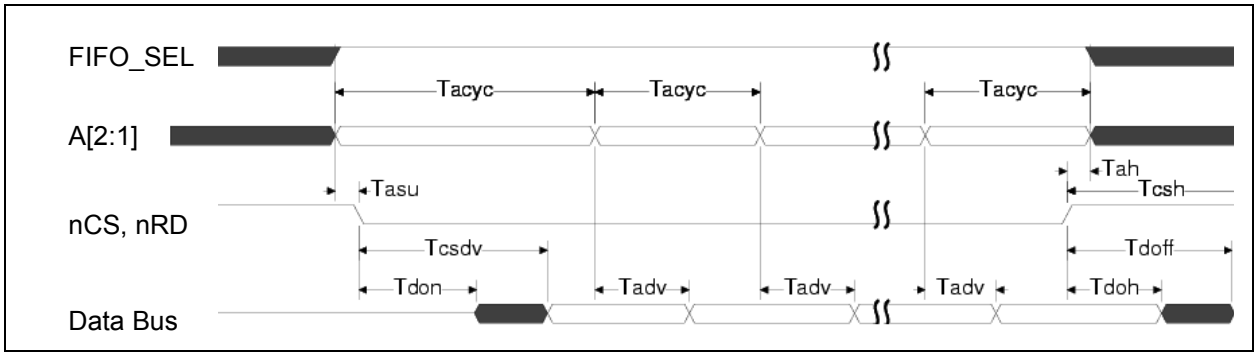


Figure 6.4 RX Data FIFO Direct PIO Burst Read Cycle Timing

Note: The “Data Bus” width is 16 bits.

Table 6.6 RX Data FIFO Direct PIO Burst Read Cycle Timing

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{csh}	nCS, nRD Deassertion Time	13			ns
t_{csdv}	nCS, nRD Valid to Data Valid			30	ns
t_{acyc}	Address Cycle Time	165			
t_{asu}	Address, FIFO_SEL Setup to nCS, nRD Valid	0			ns
t_{adv}	Address Stable to Data Valid			40	
t_{ah}	Address, FIFO_SEL Hold Time	0			ns
t_{don}	Data Buffer Turn On Time	0			ns
t_{doff}	Data Buffer Turn Off Time			7	ns
t_{doh}	Data Output Hold Time	0			ns

Note: An RX Data FIFO Direct PIO Burst Read cycle begins when both nCS and nRD are asserted. The cycle ends when either or both nCS and nRD are deasserted. They may be asserted and deasserted in any order.

6.6 PIO Writes

PIO writes are used for all LAN9210 write cycles. PIO writes can be performed using Chip Select (nCS) or Write Enable (nWR). Either or both of these control signals must go high between cycles for the period specified.

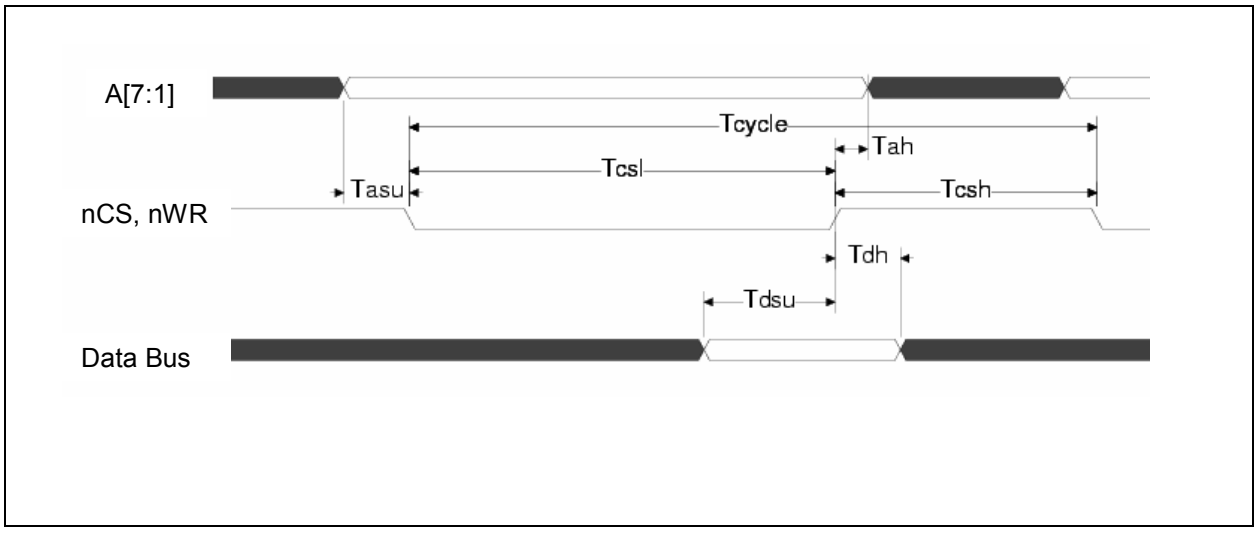


Figure 6.5 PIO Write Cycle Timing

Note: The “Data Bus” width is 16 bits.

Table 6.7 PIO Write Cycle Timing

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{cycle}	Write Cycle Time	165			ns
t_{csl}	nCS, nWR Assertion Time	32			ns
t_{csh}	nCS, nWR Deassertion Time (see Note below)	13	133		ns
t_{asu}	Address Setup to nCS, nWR Assertion	0			ns
t_{ah}	Address Hold Time	0			ns
t_{dsu}	Data Setup to nCS, nWR Deassertion	7			ns
t_{dh}	Data Hold Time	0			ns

Note: A PIO Write cycle begins when both nCS and nWR are asserted. The cycle ends when either or both nCS and nWR are deasserted. They may be asserted and deasserted in any order. Parameters t_{csh} and t_{csl} must be extended using wait states to meet the t_{cycle} minimum.

6.7 TX Data FIFO Direct PIO Writes

In this mode the upper address inputs are not decoded, and any write to the LAN9210 will write the TX Data FIFO. This mode is enabled when FIFO_SEL is driven high during a write access. This is normally accomplished by connecting the FIFO_SEL signal to a high-order address line. This mode is useful when the host processor must increment its address when accessing the LAN9210. Timing is identical to a PIO write, and the FIFO_SEL signal has the same timing characteristics as the address lines.

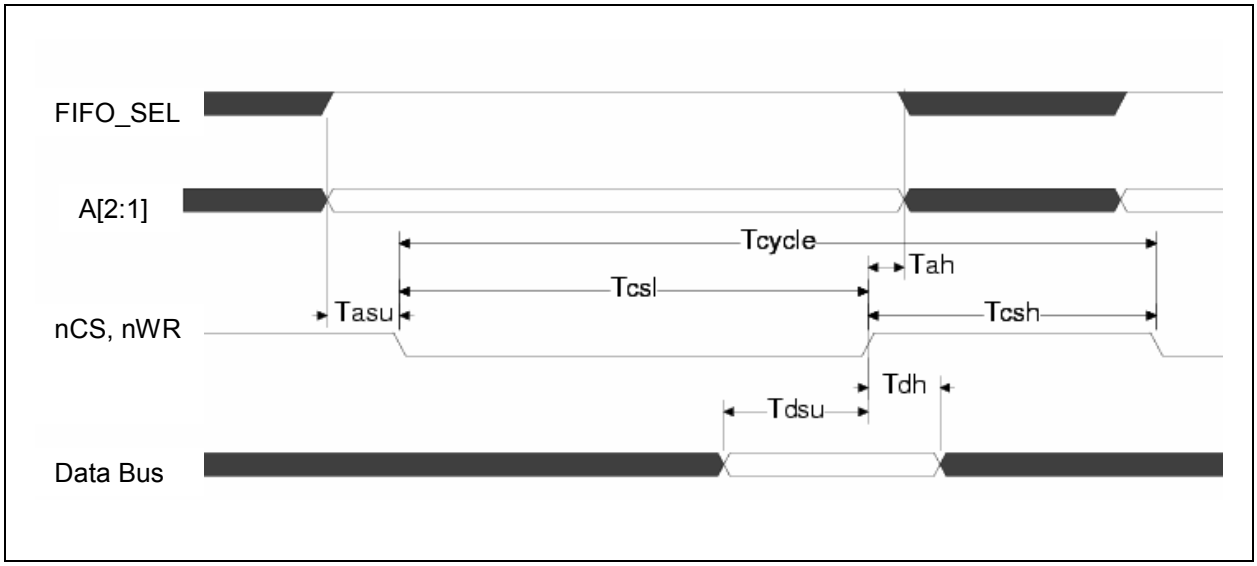


Figure 6.6 TX Data FIFO Direct PIO Write Timing

Note: The “Data Bus” width is 16 bits.

Table 6.8 TX Data FIFO Direct PIO Write Timing

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{cycle}	Write Cycle Time	165			ns
t_{csl}	nCS, nWR Assertion Time	32			ns
t_{csh}	nCS, nWR Deassertion Time (see Note below)	13	133		ns
t_{asu}	Address, FIFO_SEL Setup to nCS, nWR Assertion	0			ns
t_{ah}	Address, FIFO_SEL Hold Time	0			ns
t_{dsu}	Data Setup to nCS, nWR Deassertion	7			ns
t_{dh}	Data Hold Time	0			ns

Note: A TX Data FIFO Direct PIO Write cycle begins when both nCS and nWR are asserted. The cycle ends when either or both nCS and nWR are deasserted. They may be asserted and deasserted in any order. Parameters t_{csh} and t_{csl} must be extended using wait states to meet the t_{cycle} minimum.

6.8 Reset Timing

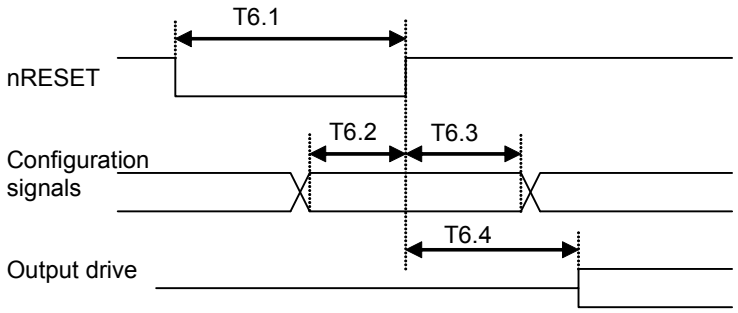


Figure 6.7 Reset Timing

Table 6.9 Reset Timing

PARAMETER	DESCRIPTION	MIN	TYP	MAX	UNITS	NOTES
T6.1	Reset Pulse Width	30			ms	
T6.2	Configuration input setup to nRESET rising	200			ns	
T6.3	Configuration input hold after nRESET rising	10			ns	
T6.4	Output Drive after nRESET rising			16	ns	

6.9 EEPROM Timing

The following specifies the EEPROM timing requirements for the LAN9210:

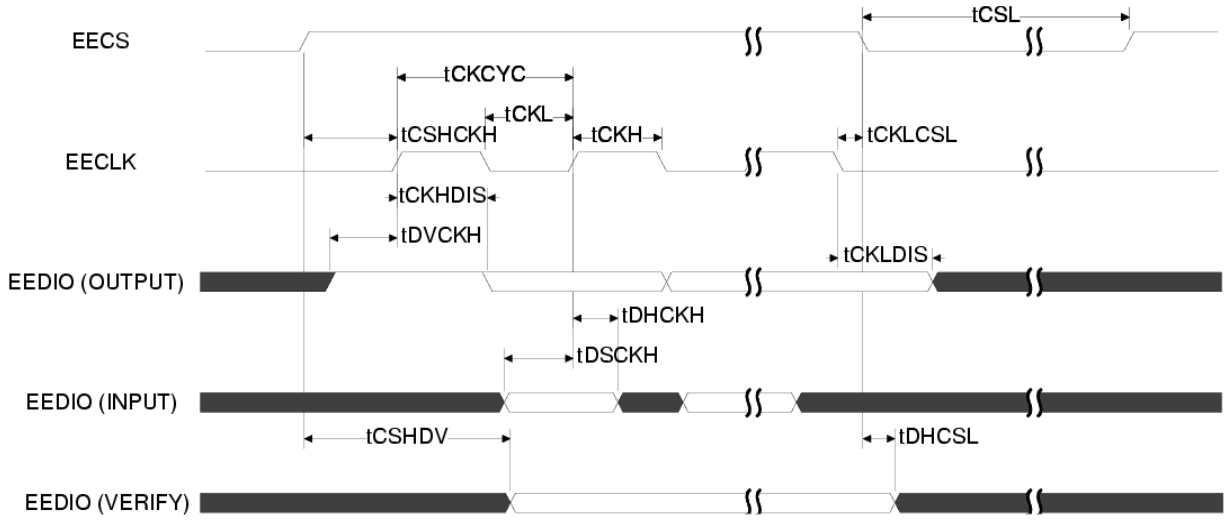


Figure 6.8 EEPROM Timing

Table 6.10 EEPROM Timing Values

SYMBOL	DESCRIPTION	MIN	TYP	MAX	UNITS
t_{CKCYC}	EECLK Cycle time	1110		1130	ns
t_{CKH}	EECLK High time	550		570	ns
t_{CKL}	EECLK Low time	550		570	ns
t_{CSHCKH}	EECS high before rising edge of EECLK	1070			ns
t_{CKLCSL}	EECLK falling edge to EECS low	30			ns
t_{DVCKH}	EEDIO valid before rising edge of EECLK (OUTPUT)	550			ns
t_{CKHDIS}	EEDIO disable after rising edge EECLK (OUTPUT)	550			ns
t_{DSCKH}	EEDIO setup to rising edge of EECLK (INPUT)	90			ns
t_{DHCKH}	EEDIO hold after rising edge of EECLK (INPUT)	0			ns
t_{CKLDIS}	EECLK low to data disable (OUTPUT)	580			ns
t_{CSHDV}	EEDIO valid after EECS high (VERIFY)			600	ns
t_{DHCSL}	EEDIO hold after EECS low (VERIFY)	0			ns
t_{CSL}	EECS low	1070			ns

Chapter 7 Operational Characteristics

7.1 Absolute Maximum Ratings*

Supply Voltage (VDD_A33, VDD_IO) (Note 7.1) . . . . . 0V to +3.3V+10%

Positive voltage on signal pins, with respect to ground (Note 7.2) . . . . . +6V

Negative voltage on signal pins, with respect to ground (Note 7.3) . . . . . -0.5V

Positive voltage on XTAL1/CLKIN, with respect to ground . . . . . +4.6V

Positive voltage on XTAL2, with respect to ground . . . . . +2.5V

Ambient Operating Temperature in Still Air (T_A). . . . . 0°C to +70°C

Storage Temperature. . . . . -65°C to +150°C

Lead Temperature Range . . . . . Refer to JEDEC Spec. J-STD-020

HBM ESD Performance . . . . . +/- 5kV

Note 7.1 When powering this device from laboratory or system power supplies, it is important that the absolute maximum ratings not be exceeded or device failure can result. Some power supplies exhibit voltage spikes on their outputs when AC power is switched on or off. In addition, voltage transients on the AC power line may appear on the DC output. If this possibility exists, it is suggested that a clamp circuit be used.

Note 7.2 This rating does not apply to the following pins: XTAL1/CLKIN, XTAL2, EXRES1.

Note 7.3 This rating does not apply to the following pins: EXRES1.

*Stresses exceeding those listed in this section could cause permanent damage to the device. This is a stress rating only. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. Functional operation of the device at any condition exceeding those indicated in Section 7.2, "Operating Conditions**", Section 7.6, "DC Electrical Specifications", or any other applicable section of this specification is not implied.

7.2 Operating Conditions**

Supply Voltage (VDD_A33, VDD_IO). . . . . +3.3V +/- 10%

Ambient Operating Temperature in Still Air (T_A). . . . . 0°C to +70°C

Note: Do not drive input signals without power supplied to the device.

Note: Apply and remove power to all power supply pins simultaneously, including the Ethernet magnetics. Do not apply power to individual supply pins without the others.

**Proper operation of the LAN9210 is guaranteed only within the ranges specified in this section.

7.3 Power Consumption (Device Only)

This section provides typical power consumption values for the LAN9210 in various modes of operation. These measurements were taken under the following conditions:

Temperature: ..... +25°C

Device VDD: ..... +3.30V

Note: Power dissipation is determined by operating frequency, temperature, and supply voltage, as well as external source/sink requirements.

Table 7.1 Power Consumption Device Only

MODE	Total Power - Typical (mW)
10BASE-T Operation	
D0, 10BASE-T /w traffic	232
D0, Idle	226
D1, Idle	165
D2, Energy Detect Power Down	60
D2, General Power Down	10
100BASE-TX Operation	
D0, 100BASE-TX /w traffic	358
D0, Idle	345
D1, Idle	253
D2, Energy Detect Power Down (Cable disconnected)	60
D2, General Power Down	10

Note 7.4 D0 = Normal Operation, D1 = WOL (Wake On LAN mode), D2= Low Power Energy Detect.

7.4 Power Consumption (Device and System Components)

This section provides typical power consumption values for a complete Ethernet interface based on the LAN9210, including the power dissipated by the magnetics and other passive components.

Note: The power measurements list below were taken under the following conditions:

Temperature: ..... +25°C

Device VDD:..... +3.30V

Note: Power dissipation is determined by operating frequency, temperature, and supply voltage, as well as external source/sink requirements.

Table 7.2 Power Consumption Device and System Components

MODE	Total Power - Typical (mW)
10BASE-T Operation	
D0, 10BASE-T /w traffic	573
D0, Idle	567
D1, Idle	449
D2, Energy Detect Power Down	60
D2, General Power Down	10
100BASE-TX Operation	
D0, 100BASE-TX /w traffic	495
D0, Idle	482
D1, Idle	391
D2, Energy Detect Power Down	60
D2, General Power Down	10

7.5 Worst Case Current Consumption

This section details the worst case current consumption for each device power supply. These values are provided to assist system designers with proper power supply design. These values cannot be used to determine typical power consumption of the device.

Note: The current measurements listed below were taken under the following conditions:

Temperature: .....0°C to +70°C

Device VDD:..... +3.30 V +/- 10%

Note: Current consumption is determined by operating frequency, temperature, and supply voltage, as well as external source/sink requirements.

Table 7.3 Maximum Supply Current Characteristics

PARAMETER	SUPPLY NAME	MAX	UNITS	NOTES
+3.3V I/O Supply Current	VDD_IO	86	mA	
+3.3V Analog Supply Current	VDD_A33	46	mA	

Note: Above values do not include the supply current for the magnetics. Based on the recommended implementation, the maximum supply current needed for the magnetics is 108mA.

7.6 DC Electrical Specifications

This section details the DC electrical specifications of the LAN9210 I/O buffers. The electrical specifications in this section are valid over the voltage range and the temperature range specified in Section 7.2, "Operating Conditions**".

Table 7.4 I/O Buffer Characteristics

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
I Type Input Buffer						
Low Input Level	V_{ILI}	-0.3		0.8	V	
High Input Level	V_{IHI}	2.0		5.5	V	
Input Leakage (V_{IN} = GND_IO or VDD_IO)	I_{IH}	-10		10	uA	Note 7.5
Input Leakage (V_{IN} = 5.5V)	I_{IH}			107	uA	Note 7.5, Note 7.6
Input Capacitance	C_{IN}			2.5	pF	
IS Type Input Buffer						
Low Input Level	V_{ILI}	-0.3			V	
High Input Level	V_{IHI}			5.5	V	
Negative-Going Threshold	V_{ILT}	1.01	1.18	1.35	V	Schmitt Trigger
Positive-Going Threshold	V_{IHT}	1.39	1.6	1.8	V	Schmitt Trigger
Schmitt Trigger Hysteresis (V_{IHT} - V_{ILT})	V_{HYS}	345	420	485	mV	
Input Leakage (V_{IN} = GND_IO or VDD_IO)	I_{IH}	-10		10	uA	Note 7.5
Input Leakage (V_{IN} = 5.5V)	I_{IH}			107	uA	Note 7.5, Note 7.6
Input Capacitance	C_{IN}			2.5	pF	
O12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12mA
High Output Level	V_{OH}	VDD - 0.4			V	I_{OH} = -12mA
OD12 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 12mA
OD8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8mA
O8 Type Buffer						
Low Output Level	V_{OL}			0.4	V	I_{OL} = 8mA

Table 7.4 I/O Buffer Characteristics (continued)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
High Output Level	V_{OH}	VDD - 0.4			V	I_{OH} = -8mA
ICLK Input Buffer						
Low Input Level	V_{ILCK}	-0.3		0.5	V	
High Input Level	V_{IHCK}	1.4		3.6	V	

Note 7.5 This specification applies to all inputs and tri-stated bi-directional pins. Internal pull-down and pull-up resistors add +/- 50uA per-pin (typical).

Note 7.6 This is the total V_{IN} input leakage for the entire device. This value should be divided by the number of pins driven to V_{IN} MAX to calculate per-pin leakage. For example, if 10 pins are driven to the maximum operational limit for V_{IN}, the per-pin input leakage is the maximum input leakage current divided by 10.

Table 7.5 100BASE-TX Transceiver Characteristics

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Peak Differential Output Voltage High	V_{PPH}	950	-	1050	mVpk	Note 7.7
Peak Differential Output Voltage Low	V_{PPL}	-950	-	-1050	mVpk	Note 7.7
Signal Amplitude Symmetry	V_{SS}	98	-	102	%	Note 7.7
Signal Rise & Fall Time	T_{RF}	3.0	-	5.0	nS	Note 7.7
Rise & Fall Time Symmetry	T_{RFS}	-	-	0.5	nS	Note 7.7
Duty Cycle Distortion	D_{CD}	35	50	65	%	Note 7.8
Overshoot & Undershoot	V_{OS}	-	-	5	%	
Jitter				1.4	nS	Note 7.9

Note 7.7 Measured at the line side of the transformer, line replaced by 100Ω (+/- 1%) resistor.

Note 7.8 Offset from16 nS pulse width at 50% of pulse peak

Note 7.9 Measured differentially.

Table 7.6 10BASE-T Transceiver Characteristics

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Transmitter Peak Differential Output Voltage	V_{OUT}	2.2	2.5	2.8	V	Note 7.10
Receiver Differential Squelch Threshold	V_{DS}	300	420	585	mV	

Note 7.10 Min/Max voltages guaranteed as measured with 100Ω resistive load.

7.7 Clock Circuit

The LAN9210 can accept either a 25MHz crystal (preferred) or a 25 MHz single-ended clock oscillator (±50 PPM) input. The LAN9210 shares the 25MHz clock oscillator input (CLKIN) with the crystal input XTAL1/CLKIN. If the single-ended clock oscillator method is implemented, XTAL2 should be left unconnected and CLKIN should be driven with a nominal 0-3.3V clock signal. The input clock duty cycle is 40% minimum, 50% typical and 60% maximum.

It is recommended that a crystal utilizing matching parallel load capacitors be used for the LAN9210 crystal input/output signals (XTAL1, XTAL2). See Table 7.7, "LAN9210 Crystal Specifications" for crystal specifications. Refer to application note AN10.7 - “Parallel Crystal Circuit Input Voltage Control” for additional information.

Table 7.7 LAN9210 Crystal Specifications

PARAMETER	SYMBOL	MIN	NOM	MAX	UNITS	NOTES
Crystal Cut	AT, typ					
Crystal Oscillation Mode	Fundamental Mode					
Crystal Calibration Mode	Parallel Resonant Mode					
Frequency	F_{fund}	-	25.000	-	MHz	
Frequency Tolerance @ 25°C	F_{tol}	-	-	+/-50	PPM	Note 7.11
Frequency Stability Over Temp	F_{temp}	-	-	+/-50	PPM	Note 7.11
Frequency Deviation Over Time	F_{age}	-	+/-3 to 5	-	PPM	Note 7.12
Total Allowable PPM Budget		-	-	+/-50	PPM	Note 7.13
Shunt Capacitance	C_O	-	7 typ	-	pF	
Load Capacitance	C_L	-	20 typ	-	pF	
Drive Level	P_W	300	-	-	uW	
Equivalent Series Resistance	R_1	-	-	50	Ohm	
Operating Temperature Range		0	-	+70	°C	
LAN9210 XTAL1/CLKIN Pin Capacitance		-	3 typ	-	pF	Note 7.14
LAN9210 XTAL2 Pin Capacitance		-	3 typ	-	pF	Note 7.14

Note 7.11 The maximum allowable values for Frequency Tolerance and Frequency Stability are application dependant. Since any particular application must meet the IEEE +/-50 PPM Total PPM Budget, the combination of these two values must be approximately +/-45 PPM (allowing for aging).

Note 7.12 Frequency Deviation Over Time is also referred to as Aging.

Note 7.13 The total deviation for the Transmitter Clock Frequency is specified by IEEE 802.3u as +/- 50 PPM.

Note 7.14 This number includes the pad, the bond wire and the lead frame. PCB capacitance is not included in this value. The XTAL1/CLKIN and XTAL2 pin and PCB capacitance values are required to accurately calculate the value of the two external load capacitors. These two external load capacitors determine the accuracy of the 25.000 MHz frequency.

Chapter 8 Package Outline

8.1 56-QFN Package

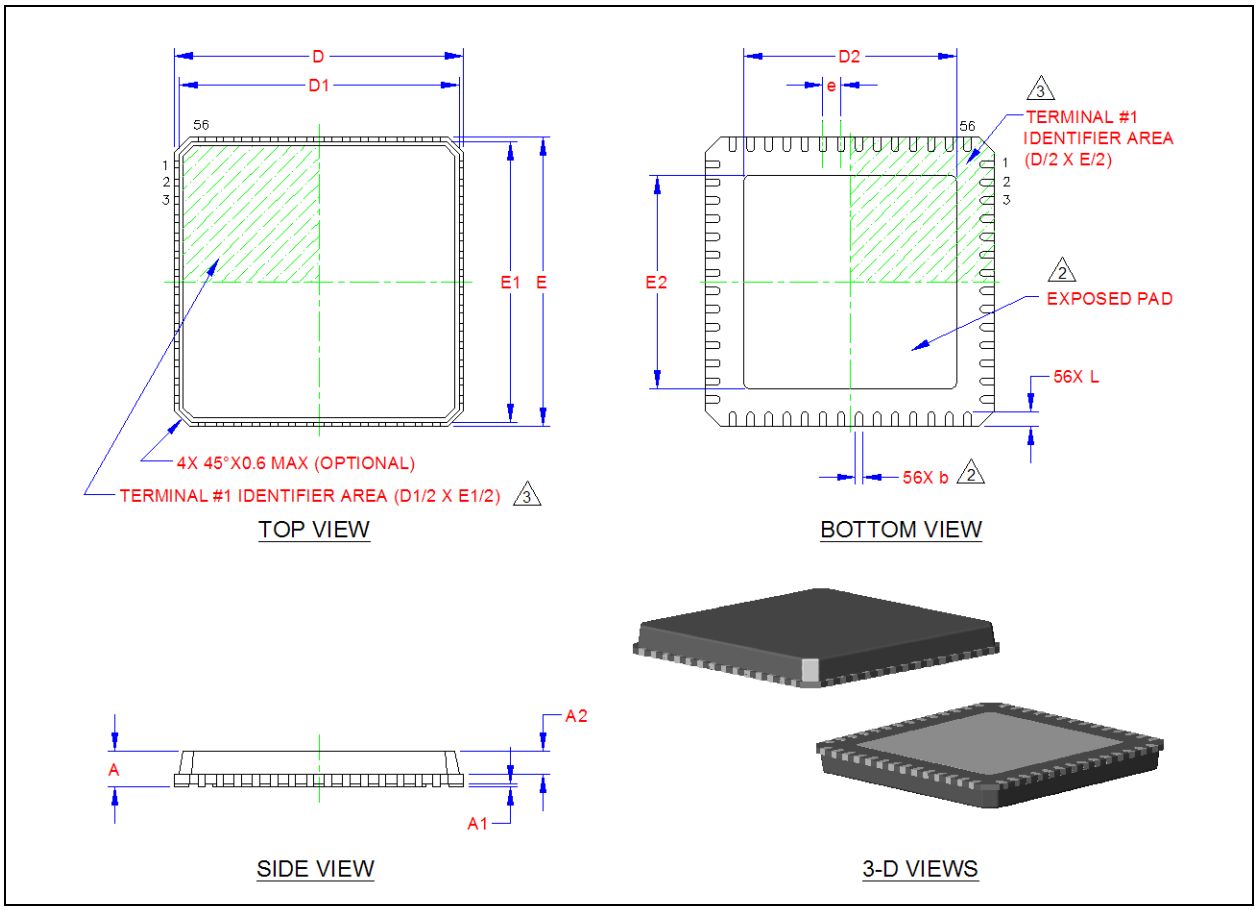


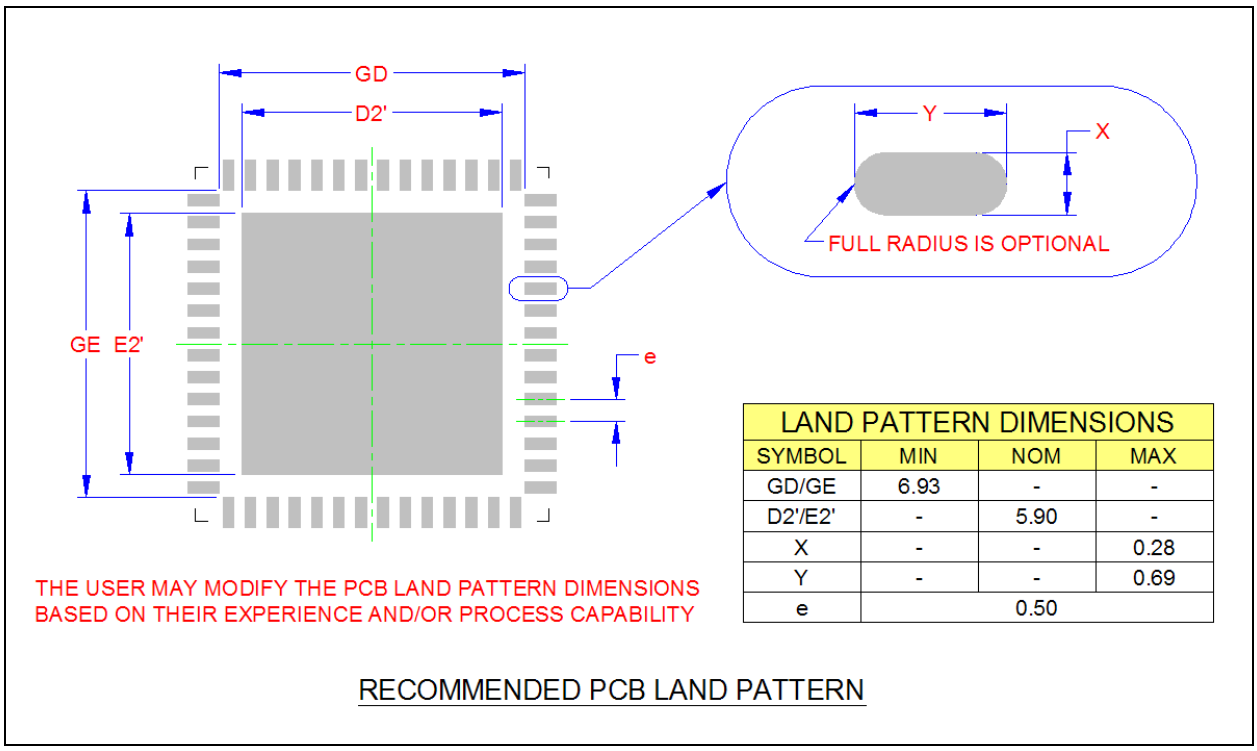
Figure 8.1 56 Pin QFN Package Definition

Table 8.1 56 Pin QFN Package Parameters

	MIN	NOMINAL	MAX	REMARKS
A	0.70	~	1.00	Overall Package Height
A1	0.00	0.02	0.05	Standoff
A2	~	~	0.90	Mold Cap Thickness
D/E	7.85	8.00	8.15	X/Y Body Size
D1/E1	7.55	~	7.95	X/Y Mold Cap Size
D2/E2	5.75	5.90	6.05	X/Y Exposed Pad Size
L	0.30	~	0.50	Terminal Length
b	0.18	0.25	0.30	Terminal Width
e		0.50 Basic		Terminal Pitch

Notes:

1. All dimensions are in millimeters.
2. Position tolerance of each terminal and exposed pad is ± 0.05 mm at maximum material condition. Dimension “b” applies to plated terminals and is measured between 0.15 and 0.30 mm from the terminal tip.
3. The pin 1 identifier may vary, but is always located within the zone indicated.



LAND PATTERN DIMENSIONS			
SYMBOL	MIN	NOM	MAX
GD/GE	6.93	-	-
D2'/E2'	-	5.90	-
X	-	-	0.28
Y	-	-	0.69
e		0.50	

Figure 8.2 56 Pin QFN Recommended PCB Land Pattern

Chapter 9 Datasheet Revision History

Table 9.1 Customer Revision History

REVISION LEVEL & DATE	SECTION/FIGURE/ENTRY	CORRECTION
Rev. 2.9 (03-01-12)	Section 5.4.8, "FLOW—Flow Control Register," on page 112	Updated Pass Control Frames (bit 2) description to “When set, the MAC will pass the pause frame to the host...”
Rev. 2.8 (07-14-11)	Table 2.4, “System and Power Signals,” on page 17	nLED1 description modified to indicate that the signal is driven low when operating speed is 100Mbs and is driven high during autonegotiation, when the cable is disconnected, and during 10 Mbs operation.
	Section 5.2.1, "RX FIFO Ports," on page 77	RX Data FIFO port is aliased to 16 WORD locations, not 16 DWORD locations.
	Section 5.2.2, "TX FIFO Ports," on page 77	TX Data FIFO port is aliased to 16 WORD locations, not 16 DWORD locations.
	Section 3.6.2.1, "TX Checksum Calculation," on page 32	Added note stating TX Checksum calculation should not be used for UDP packets under IPv6.
Rev. 2.7 (03-15-10)	Chapter 2, "Pin Description and Configuration," on page 14	Added pin 1 designator to pin diagram
	Section 7.2, "Operating Conditions**," on page 137	Added note: “Do not drive input signals without power supplied to the device.”
	Section 7.2, "Operating Conditions**," on page 137	Added note: “Apply and remove power to all power supply pins simultaneously, including the Ethernet magnetics. Do not apply power to individual supply pins without the others.”
Rev. 2.5 (11-13-08)	All	Fixed various typos
Rev. 2.4 (10-24-08)	All	Fixed various typos
	Table 7.1 on page 138, Table 7.2 on page 139, and Table 7.3 on page 140	Updated power consumption and supply current characteristics tables.
	Table 7.4, “I/O Buffer Characteristics,” on page 141	Added input capacitance values.
	Section 3.8, "General Purpose Timer (GP Timer)," on page 36	Changed incorrect “GPT_CNT” reference to “GPT_LOAD”: “On a reset, or when the TIMER_EN bit changes from set ‘1’ to cleared ‘0,’ the GPT_LOAD field is initialized to FFFFh.”
	Section 5.3.23, "E2P_CMD – EEPROM Command Register," on page 101	Corrected MAC Address Loaded (bit 8) type from “RO” to “R/WC”
	Table 7.7 on page 143	Updated crystal specifications: Drive Level: 300uW ESR: 50 Ohms.

Table 9.1 Customer Revision History (continued)

REVISION LEVEL & DATE	SECTION/FIGURE/ENTRY	CORRECTION
Rev. 2.3 (08-18-08)	Note 7.6 on page 142	Note following I/O Buffer Characteristics table modified: Changed from: ".....the per-pin input leakage is 10 divided by the maximum input leakage current." to: ".....the per-pin input leakage is the maximum input leakage current divided by 10."
Rev. 2.2 (06-19-08)	Figure 1.2, "Internal Block Diagram"	Diagram redone. The word “Core” was added to the regulator block title.
	Table 2.4, “System and Power Signals,” on page 17	Changed VDD_CORE/VDD18CORE bulk capacitor value from 10uF to 4.7uF.
Rev. 2.2 (06-10-08)	Auto-negotiation Advertisement on page 119	Bits 9 and 15 relabeled as Reserved, Read-Only (RO), with a default of 0.
	Auto-negotiation Advertisement on page 119	Fixed definition of bits 11:10 when equal to “11” by adding “advertise support for..” to beginning of definition. Also added note stating “When both symmetric PAUSE and asymmetric PAUSE support are advertised, the device will only be configured to, at most, one of the two settings upon auto-negotiation completion.”
	Section 3.5, "Wake-up Frame Detection," on page 25 and Section 5.4.1, "MAC_CR—MAC Control Register," on page 105	Added note: “When wake-up frame detection is enabled via the WUEN bit of the WUCSR—Wake-up Control and Status Register, a broadcast wake-up frame will wake-up the device despite the state of the Disable Broadcast Frame (BCAST) bit in the MAC_CR—MAC Control Register.”
	Section 5.4.12, "WUCSR—Wake-up Control and Status Register," on page 114	Fixed typo in bit 9: “... Mac Address [1:0] bit set to 0.” was changed to “...Mac Address [0] bit set to 0.”
	Section 3.6.1.1, "RX Checksum Calculation," on page 31	“checksum = [B0, B1] + C0 + [B2, B3] + C1 + … + [0, BN] + CN-1” changed to “checksum = [B1, B0] + C0 + [B3, B2] + C1 + … + [0, BN] + CN-1”
	Table 7.3 on page 140	Added 1.8V Analog Supply Current (VDD_A18) into the VDD_IO supply current and removed the 1.8V row.
Rev. 2.1 (05-13-08)	Section 1.1, "Block Diagram"	Removed the system memory block and arrow above the microprocessor/ microcontroller
Rev. 2.0 (04-11-08)	Section 7.6, "DC Electrical Specifications," on page 141	Input leakage current values added
Rev. 1.92 (10-22-07)	Chapter 2 Pin Description and Configurationon page 14	Pin assignment information re-organized into separate table.
	Transmit Checksum Offload Engine (TXCOE) section of Chapter 3, "Functional Description," on page 21.	Note added indicating the proper usage of the TX checksum preamble (DWORD alignment).
	EECLK pin description in Chapter 2 Pin Description and Configurationon page 14	Note added to EECLK pin description to indicate proper usage.