IP100
Integrated 10/100 Ethernet MAC + PHY

1 Features

- Single chip 10/100BASE, half or full duplex Ethernet Media Access Controller
- IEEE 802.3 compliant 100BASE-TX PHY
- IEEE 802.3 compliant 10BASE-T PHY
- IEEE 802.3 full duplex flow control
- IEEE 802.3 compliant 100BASE-FX PCS and PMA
- PCI Bus master scatter/gather DMA on any byte boundary
- Full operation with PCI Clock from 12.5 MHz to 33 MHz
- PCI Revision 2.2 compliant
- On-chip transmit and receive FIFO buffers
- On-chip LED drivers
- Power management capabilities for ACPI 1.0 compliant systems
- WakeOnLAN support
- Management statistics gathering
- IP multicast receive and filter support using 64 bit hash table
- Transmit polling
- Auto pad insertion for short packets
- Programmable minimum Inter Packet Gap
- Programmable transmit and receive FIFO watermarks
- On-chip crystal oscillator
- On-chip voltage regulator
- 2.5/3.3V CMOS with 5V tolerant I/O
- 0.25µm technology
- 128-pin PQFP

2 General description

The IP100 is a single-chip, full duplex, 10/100Mbps Ethernet MAC + PHY incorporating a 32-bit PCI with bus master support. The IP100 is designed for use in a variety of applications including workstation NICs, PC motherboards, and other systems utilizing a PCI bus that require network connectivity to an Ethernet or Fast Ethernet LAN.

The IP100 includes a PCI bus interface unit, IEEE 802.3 compliant MAC, transmit and receive FIFO buffers, IEEE 802.3 compliant 100BASE-TX, 10BASE-T, and 100BASE-FX PHY, serial EEPROM interface, expansion ROM interface, and LED drivers.

The IP100 implements a rich set of control and status registers. Accessible via the PCI interface, these registers provide a host system visibility into the features and operating state of the IP100. Network management statistics are also recorded, and host access to registers of the PHY device are facilitated through the IP100's PCI interface.

The IP100 supports features for use in "Green PCs" or systems where control over system power consumption is desired. The IP100 supports several power down states, and the ability to issue a system "wake event" via reception of unique, user defined Ethernet frames. In addition, the IP100 can assert a wake event in response to changes in the Ethernet link status.

Preliminary, Specification subject to change without notice.

Revision #	Change Description
IP100-DS-R02	Initial release.
IP100-DS-R03	1. Delete CLKRUN. 2. Delete Card Bus descriptions. 3. Modified VREF definition.

BLOCK DIAGRAM

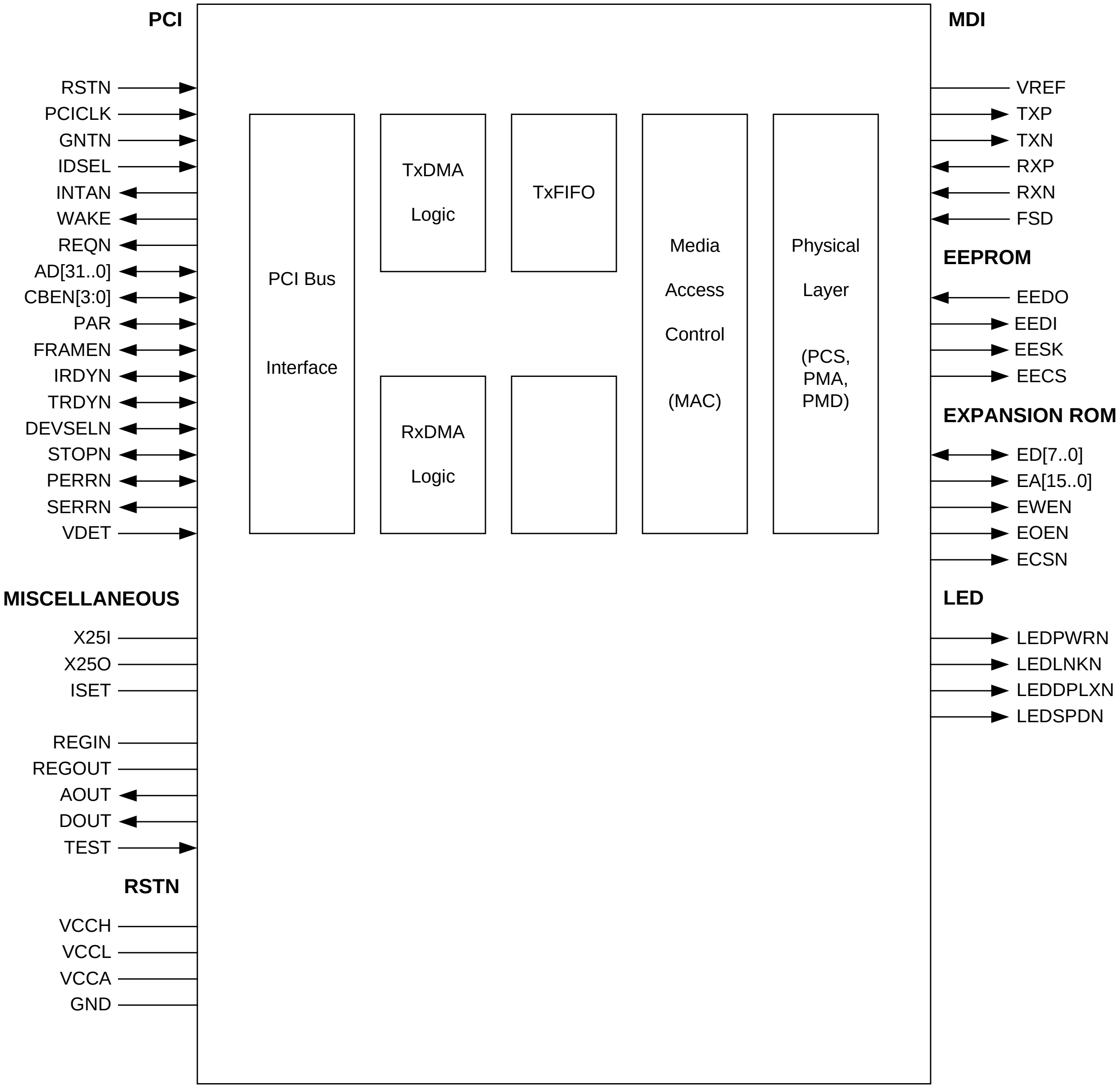


FIGURE 1: IP100 Block Diagram

Preliminary, Specification subject to change without notice.

PIN DESIGNATIONS

PIN NO.	PIN NAME	PIN NO.	PIN NAME	PIN NO.	PIN NAME	PIN NO.	PIN NAME
1	GND	33	VCCH	64	ED2	97	REGOUT
2	AD27	34	AD15	66	ED1 / LEDSPDN	98	REGIN
3	AD26	35	AD14	67	ED0 / LEDDPLXN	99	AGND
4	AD25	36	AD13	68	GND	100	VCCL
5	AD24	37	AD12	69	EA0	101	VREF
6	VCCH	38	GND	70	EA1	102	AGND
7	CBEN3	39	AD11	71	EA2	103	TXP
8	IDSEL	40	AD10	72	EA3	104	TXN
9	AD23	41	AD9	73	EA4	105	VCCA
10	GND	42	AD8	74	EA5	106	AGND
11	AD22	43	CBEN0	75	EA6	107	VCCA
12	AD21	44	AD7	76	VCCH	108	RXP
13	GND	45	VCCH	77	VCCL	109	RXN
14	AD20	46	AD6	78	EA7	110	AGND
15	AD19	47	GND	79	EA8	111	ISET
16	AD18	48	AD5	80	EA9	112	AOUT
17	AD17	49	AD4	81	EA10	113	DOUT
18	AD16	50	GND	82	EA11	114	TEST
19	GND	51	VCCL	83	EA12 / EEDO	115	VDET
20	VCCH	52	AD3	84	EA13 / EEDI	116	GND
21	CBEN2	53	AD2	85	GND	117	WAKE / PMEN
22	FRAMEN	54	AD1	86	EA14 / EESK	118	INTAN
23	IRDYN	55	AD0	87	EA15	119	RSTN
24	VCCL	56	EWEN/ LEDPWRN	88	GND	120	GND
25	TRDYN	57	GND	89	ECSN	121	PCICLK
26	DEVSELN	58	EOEN/ LEDLNKN	90	EECS	122	VCCH
27	STOPN	59	ED7	91	X25O	123	GNTN
28	PERRN	60	VCCH	92	X25I	124	REQN
29	GND	61	ED6	93	VCCP	125	AD31
30	SERRN	62	ED5	94	AGND	126	AD30
31	PAR	63	ED4	95	FSD	127	AD29
32	CBEN1	64	ED3	96	VCCA	128	AD28

TABLE 1 : IP100 Pin Designations

PIN DESCRIPTIONS

PIN NAME	PIN TYPE	PIN DESCRIPTION
		PCI INTERFACE
RSTN	INPUT	Reset, asserted LOW. RSTN will cause the IP100 to reset all of its functional blocks. RSTN must be asserted for a minimum duration of 10 PCICLK cycles.
PCICLK	INPUT	PCI Bus Clock. This clock is used to drive the PCI bus interfaces and the internal DMA logic. All bus signals are sampled on the rising edges of PCICLK. PCICLK can operate from 0MHz to 33MHz.
GNTN	INPUT	PCI Bus Grant, asserted LOW. GNTN signals access to the PCI bus has been granted to IP100.
IDSEL	INPUT	Initialization Device Select. The IDSEL is used to select the IP100 during configuration read and write transactions.
INTAN	OUTPUT	Interrupt Request, asserted LOW. The IP100 asserts INTAN to request an interrupt, when any one of the programmed interrupt event occurs.
WAKE	OUTPUT	Wake Event, assertion level is programmable (see the WakePolarity bit of the WakeEvent register). The IP100 asserts WAKE to signal the detection of a wake event.
REQN	OUTPUT	Request, asserted LOW. The IP100 asserts REQN to request PCI bus master operation.
AD [31..0]	IN/OUT	PCI Bus Address/Data. Address and data are multiplexed on the AD pins. The AD pins carry the physical address during the first clock cycle of a transaction, and carry data during the subsequent clock cycles.
CBEN [3..0]	IN/OUT	PCI Bus Command/Byte Enable, asserted LOW. Bus command and byte enables are multiplexed on the CBEN pins. CBEN specify the bus command during the address phase transaction, and carry byte enables during the data phase.
PAR	IN/OUT	Parity. PCI Bus parity is even across AD[31..0] and CBEN[3..0]. The IP100 generates PAR during address and write data phases as a bus master, and during read data phase as a target. It checks for correct PAR during read data phase as bus master, during every address phase as a bus slave, and during write data phases as a target.
FRAMEN	IN/OUT	PCI Bus Cycle Frame, asserted LOW. FRAMEN is an indication of a transaction. It is asserted at the beginning of the address phase of the bus transaction and de-asserted before the final transfer of the data phase of the transaction.
IRDYN	IN/OUT	Initiator Ready, asserted LOW. A bus master asserts IRDYN to indicate valid data phases on AD[31..0] during write data phases, indicates it is ready to accept data during read data phases. A target will monitor IRDYN.
TRDYN	IN/OUT	Target Ready, asserted LOW. A bus target asserts TRDYN to indicate valid read data phases, and to indicate it is ready to accept data during write data phases. A bus master will monitor TRDYN.
DEVSELN	IN/OUT	Device Select, asserted LOW. The IP100 asserts DEVSELN when it is selected as a target during a bus transaction. It monitors DEVSELN for any target to acknowledge a bus transaction initiated by the IP100.
STOPN	IN/OUT	Stop, asserted LOW. STOPN is driven by the slave target to inform the bus master to terminate the current transaction.

TABLE 2 : IP100 Pin Descriptions

Preliminary, Specification subject to change without notice.

PIN DESCRIPTIONS (continued)

PIN NAME	PIN TYPE	PIN DESCRIPTION
		PCI INTERFACE
PERRN	IN/OUT	Parity Error, asserted LOW. The IP100 asserts PERRN when it checks and detects a bus parity errors. When it is generating PAR output, the IP100 monitors for any reported parity error on PERRN.
SERRN	OUTPUT	System Error, asserted LOW.
VDET	INPUT	Power Detect. The IP100 detects PCI bus power supply loss when VDET is LOW.
		EEPROM INTERFACE
EECS	OUTPUT	EEPROM Chip Select. EECS is asserted by the IP100 to access the EEPROM. EECS is connected directly to the chip select input of the EEPROM device.
EESK	OUTPUT	EEPROM Serial Clock. EESK is an output connected directly to the clock input of the EEPROM device.
EEDI	OUTPUT	EEPROM Data Input. EEDI is an output connected directly to the data input of the EEPROM device.
EEDO	INPUT	EEPROM Data Output. EEDO is an input connected directly to the data output of the EEPROM device.
		EXPANSION ROM INTERFACE
EA [15..0]	OUTPUT	Expansion ROM Address Bus. EA is the expansion ROM address (see ExpRomAddr and ExpRomData registers).
ED [7..0]	IN/OUT	Expansion ROM Data Bus. ED is the expansion ROM data bus (see ExpRomAddr and ExpRomData registers).
EWEN	OUTPUT	Expansion ROM Write Enable. EWEN is the expansion ROM write enable signal during accesses to the expansion ROM (see ExpRomAddr and ExpRomData registers).
EOEN	OUTPUT	Expansion ROM Output Enable. EOEN is the expansion ROM output enable signal during accesses to the expansion ROM (see ExpRomAddr and ExpRomData registers).
ECSN	OUTPUT	Expansion ROM Chip Select. ECSN is asserted by the IP100 to access the Expansion ROM. ECSN is connected directly to the chip select input of the Expansion ROM device.
		LED DRIVERS
LEDSPDN	OUTPUT	Speed Status LED. LEDSPDN is the speed status LED driver. The speed status LED driver is LOW when the link speed is 100Mbps, and HIGH when the link speed is 10Mbps. Additional functionality of the speed status LED signal is based on the LEDMode bit of the AsicCtrl register.
LEDDPLXN	OUTPUT	Duplex Status LED. LEDDPLXN is the duplex status LED driver. The duplex status LED driver is LOW when the link is full duplex, and HIGH when the link is half duplex. Additional functionality of the speed status LED signal is based on the LEDMode bit of the AsicCtrl register.

TABLE 2 : IP100 Pin Descriptions

PIN DESCRIPTIONS (continued)

PIN NAME	PIN TYPE	PIN DESCRIPTION
LED DRIVERS		
LEDPWRN	OUTPUT	Power Status LED. LEDPWRN is the power status LED driver. The functionality of the power status LED signal is based on the LEDMode bit of the AsicCtrl register.
LEDLNKN	OUTPUT	Link Status LED. LEDLNKN is the link status LED driver. The functionality of the link status LED signal is based on the LEDMode bit of the AsicCtrl register.
MDI		
RXP	INPUT	Receive input. When in 100BASE-TX mode, this receives MLT3 data from the isolation transformer. When in 100BASE-FX mode, this is a PECL input.
RXN	INPUT	Receive input. When in 100BASE-TX mode, this receives MLT3 data from the isolation transformer. When in 100BASE-FX mode, this is a PECL input.
TXP	OUTPUT	Transmit output. When in 100BASE-TX mode, this is an MLT-3 driver. When in 100BASE-FX mode, this is a PECL driver.
TXN	OUTPUT	Transmit output. When in 100BASE-TX mode, this is an MLT-3 driver. When in 100BASE-FX mode, this is a PECL driver.
FSD	INPUT	Fiber optic signal detect. For 100BASE-FX applications, FSD is connected to the signal detect output pin of a fiber optic module at PECL level. Connecting this pin to GND will force the IP100 in the 100BASE-TX mode.
VREF	ANALOG	For FX usage, this pin can be left floating. For UTP application, its purpose is to provide a common mode reference voltage to the transmit transformer and it should connect this pin to the center tap of the transmit transformer. For EMI consideration, it is also suggested to place a bead (100Mhz) between the center tap of the transformer and the VREF pin. If the noise is still disturbing the signals, a 39pF capacitor should also be considered.
MISCELLANEOUS		
X25I	OSCIN	25MHz Crystal Oscillator Input. The external 25MHz crystal and capacitor is connected to the on-chip crystal oscillator circuit through X25I input. Alternately, X25I can be driven by an external clock source.
X25O	OSCOUT	25MHz Crystal Oscillator Output. The external crystal and capacitor is also connected to the output of the on-chip crystal oscillator circuit through X25O. When X25I is driven by an external clock source, X25O should be left unconnected.
ISET	ANALOG	Band Gap Resistor. Connect a 6.2kohm, 1% resister between ISET and GND.
REGIN	ANALOG	Voltage regulator input. The internal 2.5V voltage regulator requires a 2N2905 type, PNP transistor be connected between REGIN and REGOUT.
REGOUT	ANALOG	Voltage regulator output. The internal 2.5V voltage regulator requires a 2N2905 type, PNP transistor be connected between REGIN and REGOUT.
AOUT	IN/OUT	Analog Test. Do not connect for normal operation.
DOUT	IN/OUT	Digital Test. Do not connect for normal operation.
TEST	INPUT	Test. Enables the IP100 test modes.
POWER AND GROUND		
VCCH	POWER	+3.3 volt I/O power supply.
VCCL	POWER	+2.5 volt digital logic power supply.
VCCA	POWER	+2.5 volt analog power supply.
GND	GROUND	Power return.
AGND	GROUND	Power return.

TABLE 2 : IP100 Pin Descriptions

3 Acronyms and Glossary

LAN	Local Area Network
MAC	Media Access Control Layer, or a device implementing the functions of this layer (a Media Access Controller)
PCI	Peripheral Component Interface
NIC	Network Interface Cards
FIFO	First In First Out
EPROM	Erasable Programmable Read Only Memory
EEPROM	Electrically Erasable Programmable Read Only Memory
LED	Light Emitting Diode
PHY	Physical Layer, or device implementing functions of the Physical Layer
CSMA/CD	Carrier Sense Multiple Access with Collision Detect
FCS	Frame Check Sequence
SFD	Start of Frame Delimiter
CRC	Cyclic Redundancy Check
IP	Internet Protocol
TFD	Transmit Frame Descriptor
RFD	Receive Frame Descriptor
DMA	Direct Memory Access
ACPI	Advanced Configuration and Power Management

4 Standards Compliance

The IP100 implements functionality compliant with the following standards:

- IEEE 802.3 Fast Ethernet
- IEEE 802.3 Full Duplex Flow Control
- PCI Local Bus Revision 2.2
- PCI Bus Power Management Interface Revision 1.1
- ACPI Revision 1.0

5 Functional Description

The IP100 is composed of various functional blocks as shown in Figure 1 on page 2. An overview of the functions performed by each block are as follows:

5.1 Media Access Control

The MAC block implements the IEEE Ethernet 802.3 Media Access Control functions with 802.3 Full Duplex and Flow Control enhancements. In half duplex mode, the MAC implements the CSMA/CD algorithm. Full duplex mode by definition does not utilize CSMA/CD, allowing data to be transmitted on demand. An optional flow control mechanism in full duplex mode is provided via the MAC Control PAUSE function. Additionally, the MAC also performs the following functions in either half or full duplex mode:

- Optional transmit FCS generation
- Padding to the minimum legal frame size
- Preamble and SFD generation
- Preamble and SFD removal
- Receive frame FCS checking and optional FCS stripping
- Receive frame destination address matching
- Support for multicast and broadcast frame reception or rejection (via filtering)
- Selective InterFrame Gap to avoid capture effect
- MAC Loopback

The MAC is responsible for generation of hardware signals to update the internal statistics counters.

5.2 Physical Layer

The IP100 supports both IEEE 802.3 100BASE-TX and 100BASE-FX signaling. The 100BASE-X transmit logic performs 4B5B encoding/decoding, parallel to serial, and serial to parallel conversion, and NRZ-NRZI signaling. In the case of 100BASE-TX, scrambling and MLT-3 encoding are also done before the data is transmitted on to the media. The receive 100BASE-X circuitry, recovering data from either an MLT-3 signal (100BASE-TX) or a PECL input (100BASE-FX), generates four bit nibbles to send to the MAC.

The Media Dependent Interface selection is done by the FSD pin. If the FSD pin is connected directly to GND, the IP100 PHY layer is operating in 100BASE-TX mode. If FSD is connected to the output of an optical module, then the PHY layer is in 100BASE-FX mode.

The IP100 PHY also includes a full set of registers for controlling the PHY as outlined in the IEEE 802.3 specification.

5.3 On-Chip Voltage Regulator

The IP100 has an integrated voltage regulator for reduced system cost. The voltage regulator is used to provide the 2.5 V power to the PCB. When used with a 2N2905 PNP transistor based circuit as shown in Figure 2, the REGIN and REGOUT pins will regulate the current through the transistor, providing a stable 2.5 V reference.

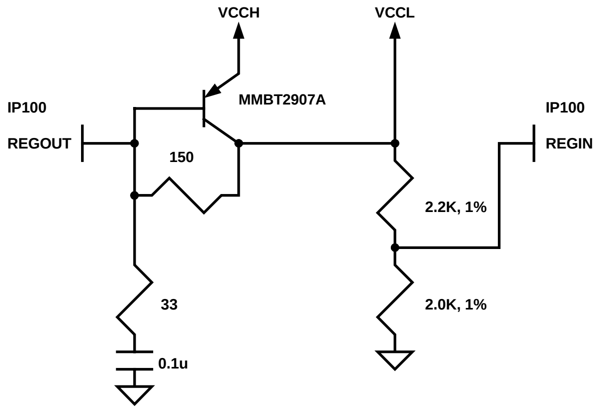


FIGURE 2: External PNP Transistor Based Regulator Circuit

5.4 PCI Bus Interface

The PCI Bus Interface implements the protocols and signals needed to operate the IP100 in a PCI bus. The IP100 can be either a PCI bus master or slave. The PCI Bus Interface is also responsible for managing the DMA interfaces and the host processors access to the IP100 registers. Arbitration logic within the PCI Bus Interface block accepts bus requests from the TxDMA Logic and RxDMA Logic.

The PBI also manages interrupt generation for a host processor.

5.5 TxDMA Logic

The IP100 supports a multi-frame, multi-fragment DMA gather process. Descriptors representing frames are built and linked in system memory by a host processor. The TxDMA Logic is responsible for transferring the multi-fragment frame data from the host memory into the TxFIFO.

The TxDMA Logic monitors the amount of free space in the TxFIFO, and uses this value to decide when to request a TxDMA. A TxDMABurstThresh register is used to delay the bus request until there is enough free space in the TxFIFO for a long burst.

To prevent a TxFIFO under run condition the TxDMA logic forwards an urgent request to the arbiter, regardless of the TxDMABurstThresh constraint, when the number of occupied bytes in the TxFIFO drops below the value in TxDMAUrgentThresh register.

5.6 TxFIFO

The IP100 uses 2K bytes of transmit data buffer between the TxDMA Logic and Transmit MAC. When the TxDMA logic determines there is enough space available in the TxFIFO, the TxDMA Logic will move any pending frame data into the TxFIFO. The TxReleaseThresh register value determines the amount of data which must be transmitted out of the TxFIFO before the FIFO memory space occupied by that data can be released for use by another frame.

A TxReleaseError occurs when a frame experiences a collision after the TxFIFO release threshold has been crossed. The IP100 will not be able to retransmit this frame from the TxFIFO and the complete frame must be transferred from the host system memory to the TxFIFO again by TxDMA Logic.

5.7 RxDMA Logic

The IP100 supports a multi-frame, multi-fragment DMA scatter process. Descriptors representing frames are built and linked in system memory by the host processor. The RxDMA Logic is responsible for transferring the frame data from the RxFIFO to the host memory.

The RxDMA Logic monitors the number of bytes in the RxFIFO. After a number of bytes have been received, the frame is "visible". A frame is visible if:

- The frame being received is determined not to be a runt, OR
- The entire frame has been received

After a frame becomes visible, the RxDMA Logic will issue a request to the arbiter when the number of bytes in the RxFIFO is greater than the value in the RxDMABurstThresh. To prevent receive overruns, a RxDMA Urgent Request is made when the amount of free space in the RxFIFO falls below the value in RxDMAUrgentThresh.

5.8 RxFIFO

The IP100 uses 2K bytes of receive data buffer between the Receive MAC and RxDMA Logic. The

Preliminary, Specification subject to change without notice.

values in RxDMABurstThresh determine how many bytes of a frame must be received into RxFIFO before RxDMA Logic is allowed to begin data transfer.

5.9 EEPROM Interface

The external serial EEPROM is used for non-volatile storage of such information as the node address, system ID, and default configuration settings. As part of initialization after system reset, the IP100 reads from the EEPROM and places the data into certain host-accessible registers.

5.10 Expansion ROM Interface

The IP100 provides support for an optional Expansion ROM. The Expansion ROM is configured through the PCI configuration register, which maps the ROM into the memory space of the host system. The ROM contents can be scanned, copied to system RAM, and executed at system initialization time.

The ROM is also byte-read and byte-write accessible to the host CPU using the ExpRomData and ExpRomAddr registers. This allows a diagnostic program to read or modify the ROM contents without having to write to configuration registers.

The Expansion ROM pins are shared with the Modem Interface pins, as use of an Expansion ROM is not permitted in multi-function applications.

6 Operation

6.1 Initialization

The IP100 provides several resets. The assertion of the hardware reset signal on the PCI bus causes a complete reset of the IP100. A similar reset is available via software using the GlobalReset bit of the AsicCtrl register. The AsicCtrl register also allows for selective reset of particular functional blocks of the IP100. See the Registers and Data Structures section for details on using the AsicCtrl register for resetting the IP100.

Shortly after reset, the IP100 will read the contents of an external EEPROM, placing the data read into the following registers:

- ConfigParm
- AsicCtrl (least significant 16 bits)
- SubsystemVendorId
- SubsystemId
- StationAddress
- Data

There are several other registers which must be configured by the host during initialization. These registers include the IP100 PCI configuration registers which are set during a Power On Self Test (POST) routine performed by the host system. Specifically, the registers set during this stage of initialization are:

- ConfigCommand enables adapter operation by allowing it to respond to and generate PCI bus cycles. ConfigCommand is also used to enable parity error generation.
- IoBaseAddress sets the I/O base address for the IP100 registers.
- MemBaseAddress sets the memory base address for the IP100 registers.
- ExpRomBaseAddress sets the base address and size for an installed expansion ROM, if any.
- CacheLineSize indicates the system’s cache line size. This value is used by the IP100 to optimize bus master data transfers.
- LatencyTimer sets the length of time the IP100 can hold the PCI bus as a bus master.
- InterruptLine maps IP100’s interrupt request to a specific interrupt line (level) on the system board.
- AsicCtrl is used to setup internal operations and parameters.

The IP100 can be accessed across the PCI bus without setting the PCI registers or loading data from an external EEPROM. In this Forced Configuration mode (useful for embedded applications without an EEPROM), the IP100 is configured as follows:

- I/O base address 0x200
- I/O target cycles enabled
- Memory target cycles disabled
- Bus master cycles enabled
- Expansion ROM cycles disabled

6.2 Register Programming

After initialization, an additional set of registers specific to operation of the Ethernet network must be programmed.

The first setting relates to the Auto-Negotiation function. The IP100 PHY layer performs the Auto-Negotiation process, and the host system must communicate with the PHY to determine the link status. Once the result of Auto-Negotiation is determined, if a full duplex mode has been chosen, the host system must set the FullDuplexEnable bit in the MACCtrl0 register. Other modes chosen during Auto-Negotiation do not require any IP100 register settings.

The ReceiveMode register determines which types of frames, based on address matching mechanism, the IP100 will receive. The end station address is loaded from the EEPROM, or the host system can set the address directly. Then, by setting the ReceiveUnicast bit in the ReceiveMode register, the IP100 will receive unicast frames whose destination address matches the value in the StationAddress register.

The ReceiveMulticastHash bit in ReceiveMode enables a filtering mechanism for Ethernet multicast frames. This filtering mechanism uses a 64-bit hash table (HashTable register) for selective reception of Ethernet multicast frames.

Additionally, Ethernet frames containing IP multicast destination addresses can also be received by setting the ReceiveIPMulticast bit in the ReceiveMode register. IP multicast, or Host Extension for IP Multicasting, datagrams map to frames with Ethernet destination addresses of 0x01005e****** (where * represents any hexadecimal value).

The MACCtrl0 and MACCtrl1 registers are used to configure parameters including full duplex, flow control, and statistics gathering.

In half duplex mode, the IP100 implements the CSMA/CD algorithm. If multiple nodes on the same network attempt to transmit simultaneously, a collision will occur resulting in re-transmission. In full duplex mode, the IP100 can transmit and receive frames simultaneously without incurring collisions. To configure the IP100 for full duplex mode operation, the host system must detect a full duplex physical link via the PHY Status Register, and must set the FullDuplexEnable bit in the MACCtrl0 register.

The IEEE 802.3x Full Duplex standard defines a special frame known as the PAUSE MAC Control frame. The PAUSE frame is used to implement flow control in full duplex networks allowing stations on opposite ends of a full duplex link the ability to inhibit transmission of data frames for a specified period of time. The PAUSE frame format is defined as shown in Figure 3.

FIELD		LENGTH (BYTES)
DA	0x0180C2000001	6
SA		6
TYPE	0x8808	2
OPCODE	0x0001	2
PAUSE TIME		2
PAD		42

FIGURE 3: PAUSE Frame

Whenever the FlowControlEnable bit in the MACCtrl0 register is set, the IP100 looks for any incoming PAUSE frame. If found, the IP100 inhibits transmission of all data frames for the time specified in the two-byte pause_time field. The pause_time field is specified in slot times relative to the current data rate; one slot time is 51.2 us at 10 Mbps, and 5.12 us at 100 Mbps. The transmission of PAUSE frames is the responsibility of the host. The MAC Control frame must be constructed by the host and placed into the TxFIFO. For end station applications, host system should only accept PAUSE frames, and not generate them. Flow control is designed to originate from network devices such as switches.

6.3 TxDMA and Frame Transmission

The TxDMA Logic transfers frame data from the host system memory to the IP100 based on a linked list of frame descriptors called TFDs. The host system creates a list of TFDs in system memory, where each TFD contains the memory locations of one or more fragments of a frame as shown in Figure 4.

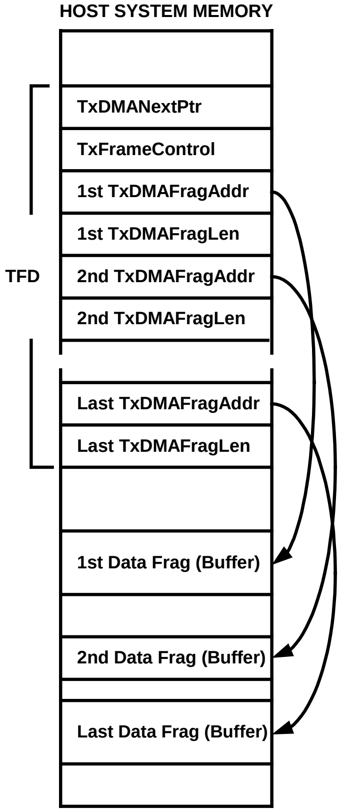


FIGURE 4: TxDMA Data Structure

The TFD format is covered in the Registers and Data Structures section.
The resulting linked list of TFDs is referred to as the TxDMAList, as shown in Figure 5.

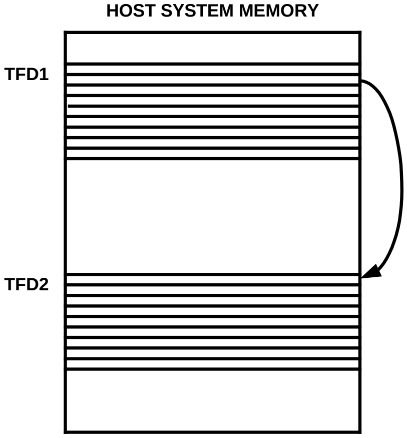


FIGURE 5: TxDMA List of Two TFDs

In the simple case of a single frame, the host system must create a TFD within the host system memory containing the addresses and lengths of the fragments of data to be transmitted. The host system must write zero into TxDMANextPtr since this is the only frame. The host starts the TxDMA Logic by writing the memory location (a non-zero address) of the TFD into TxDMAListPtr register. The TxDMA Logic begins transferring data into the IP100.

The IP100 first fetches the fragment addresses and fragment lengths from the TFD and writes them one at a time into registers, which are used to control the data transfer operations. If the TxDMA Logic transfers more data than can fit into the TxFIFO, an overrun will occur.

The TxDMAListPtr I/O register within the IP100 contains the physical address that points to the head of the TxDMAList. TxDMAListPtr must point to addresses which are on 8-byte boundaries. A value of zero in the TxDMAListPtr register implies there are no pending TFD's for the IP100 to process.

Generally, it is desirable for the host system to queue multiple frames. Multiple TFD's are linked together in a list by pointing the TxDMANextPtr of each TFD at the next TFD. The last TFD in the linked list should have a value of zero for it's TxDMANextPtr.

The TxDMA process returns to the idle state upon detection of a zero value for TxDMANextPtr. When a new frame is available to transfer, the host system must write the address of the new TFD into the TxDMANextPtr memory location of the last TFD, and either set the TxEnable bit, or utilize the IP100's automatic polling capability. Using automatic polling, the IP100 will monitor the TxDMANextPtr memory location until a non-zero value is found at that location in system memory. The TxDMAPollPeriod register controls this polling function, which is enabled when TxDMAPollPeriod contains a non-zero value. The value written to TxDMAPollPeriod determines the TxDMANextPtr polling interval.

In response to a TxDMAComplete interrupt, when data transfer by TxDMA is finished, the host acknowledges the interrupt and returns the frame data buffers to the system. In the case of a multi-frame TxDMAList, multiple frames may have been transferred by TxDMA when the host system enters its interrupt service routine. The host system can traverse the list of TFD's, examining the TxDMAComplete bit in each TFD to determine which frames have been transferred by TxDMA.

The IP100 fetches the TFC before frame data transfer, and again at the end of TxDMA operation to examine the TxDMAIndicate bit. This allows the host system to change TxDMAIndicate while data transfer of the frame is in progress. For instance, a frame's TFD might be at the end of the TxDMAList when it starts TxDMA, so the host system would probably set TxDMAIndicate to generate an interrupt. However, if during the TxDMA process of this frame, the host system added a new TFD to the end of the list, it might clear TxDMAIndicate in the currently active TFD so that the interrupt is delayed until the next TFD.

The IP100 has the ability to automatically round up the length of a transmit frame. This is useful in some NOS environments in which frame lengths need to be an even number of words. The frame length is rounded up to either a word or dword boundary, depending upon the value of WordAlign. Host systems may disable frame length word-alignment by setting the WordAlign bits in the TransmitFrameControl to x1.

The MAC will initiate frame transmission (if transmission is enabled) as soon as either the entire frame is resident in the TxFIFO register.

As a frame transmits out of the TxFIFO, it is desirable to be able to release the FIFO space so that it may be used for another frame. The value programmed into TxReleaseThresh determines how much of a frame must be transmitted before its FIFO space can be released.

6.4 Frame Reception and RxDMA

The frame RxDMA mechanism is similar to the TxDMA mechanism. RxDMA is structured around a linked list of frame descriptors, called RFDs. RFDs contain pointers to the fragment buffers into which the IP100 is to place receive data, as shown in Figure 6.

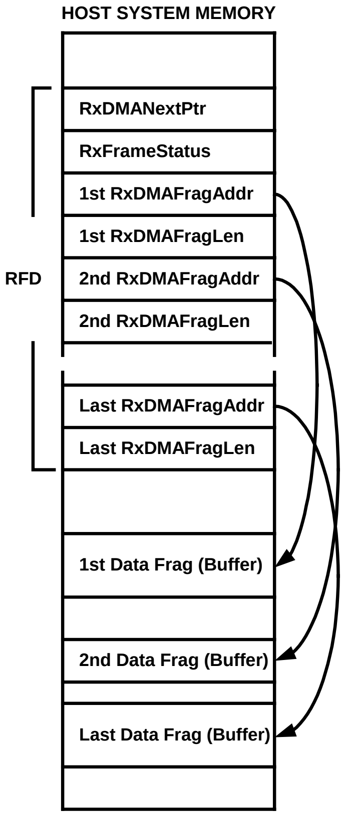


FIGURE 6: RxDMA Data Structure

The RFD format is covered in the Registers and Data Structures section.

Similar to TFDs, the resulting linked list of RFDs is referred to as the RxDMAList. One option available to RxDMA that differs from TxDMA is that the RxDMAList can be formed into a ring as shown in Figure 7. A host system can allocate a number of full size frame buffers, create a RFD for each one, and link the RFDs into a circular list. As frames are received and transferred by RxDMA, a RxDMAComplete interrupt will be generated for each frame.

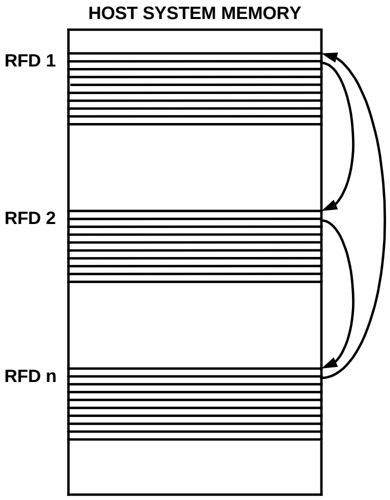


FIGURE 7: RxDMA List Shown in Ring

The host system must create a RxDMAList and the associated buffers prior to reception of a frame. One approach calls for the host system to allocate a block of full size (i.e. large enough to hold a maximum size Ethernet frame of 1518 bytes) frame buffers in system data space and create RFDs that point to them. Another approach is for the host system to request the buffers from the protocol ahead of time.

After reset, the IP100 receive function is disabled. Once the RxEnable bit is set, frames will be received according to the matching mode programmed in ReceiveMode register. Reception can be disabled by setting the RxDisable bit. If set while a frame is being received, RxDisable only takes effect after the active frame reception is finished.The receive function begins with the RxDMA Logic in the idle state. The RxDMA Logic will begin processing a RxDMAList as soon as a non-zero address is written into the RxDMAListPtr register. The host system creates a RFD with the addresses and lengths of the buffers to be used and programs the RxDMAListPtr register to point to the head of the list. The host system must program a zero into the RxDMANextPtr of the last RFD to indicate the end of the RxDMAList. When a frame is received in the RxFIFO, the IP100 fetches the fragment address and fragment length values one by one from the current RFD, and writes these values into internal registers which control data transfer operations. Similar to TxDMA, the RxDMA Logic will return to the idle state when the RxDMAListPtr register is zero.

For RxDMA lists configured as a ring, the host system should clear the RxDMAComplete bit within the ReceiveFrameStatus field of the RFD from which the host system has finished reading data. If RxDMAPollPeriod is zero the host system should also issue a RxDMAResume in case the IP100 has halted due to detection of a set RxDMAComplete bit within the ReceiveFrameStatus field of the next RFD in the ring. If the IP100 fetches a RxDMAListPtr for a RFD that has already been used (a RFD in which the RxDMAComplete bit is set in ReceiveFrameStatus), the RxDMA Logic will either assert an implicit RxDMAHalt or, if the RxDMAPollPeriod register is set to a non-zero value, the RxDMA Logic will automatically recheck RxDMAComplete periodically until it is cleared.

The IP100 can be configured to generate a RxDMAComplete interrupt when RxDMA completes a frame transfer. In response to a RxDMAComplete interrupt, the host system must examine the ReceiveFrameStatus field in the RFD of the received frame to determine the size of the frame and whether there were any errors. The host system must then copy the frame out of the receive buffers, if needed.

In general, when the host system enters its interrupt service routine, multiple frames may have been transferred by RxDMA. The host system can read RxDMAListPtr to determine which RFDs in the list have been used. The host system begins at the head of the RFD list, and traverses the list until it reaches the RFD whose address matches RxDMAListPtr. However, since I/O operations are costly, it is more efficient to use the RxDMAComplete bit in each RFD to determine which frames have been transferred by RxDMA.

In some host systems, it may be desirable to copy received frame data out of the scatter buffer to the protocol buffer while the frame is still being transferred by RxDMA. The RxDMAStatus register is provided for this purpose. If the host system sets the RxDMAHalt bit in the DMACtrl register, reads the RxDMAListPtr register and the RxDMAStatus register, then sets the RxDMAResume bit in the DMACtrl register, the host system can determine how much of the frame has been transferred by RxDMA. The RxDMAStatus register indicates the number of bytes transferred by RxDMA for the current RFD pointed to by the RxDMAListPtr register. The host system can then perform memory copies out of the RFD buffer concurrently with the RxDMA operation.

6.5 Interrupts

The term “interrupt” is used loosely to refer to interrupts and indications. An interrupt is the actual assertion of the hardware interrupt signal on the PCI bus. An indication, or a set bit in the IntStatus register, is the reporting of any event enabled by the host. The host system will configure the IP100 to generate an interrupt for any indication that is of interest to it. There are 11 different types of interrupt indications that can be generated by the IP100. The IntEnable register controls which of the 11 indication bits can assert a hardware interrupt. In order for an indication bit to be allowed to generate an interrupt, its corresponding bit-position in IntEnable must be set. When responding to an interrupt, the host reads the IntStatus register to determine the cause of the interrupt. The least significant bit of IntStatus, InterruptStatus, is always set whenever any of the interrupts are asserted. InterruptStatus must be explicitly acknowledged (cleared) by writing a 1 into the bit in order to prevent spurious interrupts on the host bus.

Interrupts are acknowledged by the host carrying out various actions specific to each interrupt.

7 Statistics

The IP100 implements 16 statistics counters of various widths. Each statistic implemented complies to the corresponding definition given in the IEEE 802.3 standard. Setting the StatisticsEnable bit in the MACCtrl1 register enables the gathering of statistics. Reading a statistics register will clear the read register. Statistic registers may be read without disabling statistics gathering. For diagnostics and testing purposes, the host system may write a value to a statistic register, in which case the value written is added to the current value of the register. Whenever one or more of the statistics registers reaches 75% of its maximum value, an UpdateStats interrupt is generated. Reading that statistics register will acknowledge the UpdateStats interrupt. A summary of the transmit and receive statistics follows. Detailed descriptions of the statistic registers related to data transmission and reception can be found in the Registers and Data Structures section.

7.1 Transmit Statistics

- FramesTransmittedOk: The number frames of all types transmitted without errors. Loss of carrier is not considered to be an error by this statistic.
- BroadcastFramesTransmittedOk: The number of frames with broadcast destination address that are transmitted without errors.
- MulticastFramesTransmittedOk: The number of frames with multicast destination address that are transmitted without errors.
- OctetsTransmittedOk: The number of total octets for all frames transmitted without error.
- FramesWithDeferredXmission: A count of frames whose transmission was delayed on it’s first attempt because network traffic.
- FramesWithExcessiveDeferral: If the transmission of a frame has been deferred for an excessive period of time due to network traffic, the event is recorded in this statistic.
- SingleCollisionFrames: Frames that are transmitted without errors after one and only one collision (including late collisions) are counted by this register.
- MultipleCollisionFrames: All frames transmitted without error after experiencing from 2 through 15 collisions (including late collisions) are counted here.
- LateCollisions: Every occurrence of a late collision (there could be more than one per frame transmitted) is counted by this statistic.
- FramesAbortedDueToXSColls: If the transmission of a frame had to be aborted due to excessive collisions, the event is recorded in this statistic.
- CarrierSenseErrors: Frames that were transmitted without error but experienced a loss of carrier are counted by this statistic.

7.2 Receive Statistics

- FramesReceivedOk: Frames of all types that are received without error are counted here.
- BroadcastFramesReceivedOk: Frames of broadcast destination address that are received without error are counted here.
- MulticastFramesReceivedOk: Frames of

Preliminary, Specification subject to change without notice.

multicast destination address that are received without error are counted here.

- OctetsReceivedOk: A total octet count for all frames received without error.
- FramesLostRxErrors: This is a count of frames that would otherwise be received by the IP100, but could not be accepted due to an overrun condition in the RxFifo.

8 PCI Bus Master Operation

The IP100 supports all of the PCI memory commands and decides on a burst-by-burst basis which command to use in order to maximize bus efficiency. The list of PCI memory commands is shown below. For all commands, "read" and "write" are with respect to the IP100 (i.e. read implies the IP100 obtains information from an off-chip location, write implies the IP100 sends information to an off-chip location).

- Memory Read (MR)
- Memory Read Multiple (MRM).
- Memory Write (MW)
- Memory Write Invalidate (MWI)

MR is used for all fetches of descriptor information. For reads of transmit frame data, MR, or MRM is used, depending upon the remaining number of bytes in the fragment, the amount of free space in the TxFIFO, and whether the RxDMA Logic is requesting a bus master operation.

MW is used for all descriptor writes. Writes of receive frame data use either MW or MWI, depending upon the remaining number of bytes in the fragment, the amount of frame data in the RxFIFO, and whether the TxDMA Logic is requesting a bus master operation.

The IP100 provides three configuration bits to control the use of advanced memory commands. The MWIEnable bit in the ConfigCommand configuration register allows the host to enable or disable the use of MWI. The MWIDisable bit in DMACtrl allows the host system the ability to disable the use of MWI. MWIDisable is cleared by default, enabling MWI.

The IP100 provides a set of registers that control the PCI burst behavior. These registers allow a trade-off to be made between PCI bus efficiency and under run/overrun frequency. Arbitration logic within the PCI Bus Interface block accepts bus requests from the TxDMA Logic and RxDMA Logic. The TxDMA Logic uses the TxDMABurstThresh register, as described in the TxDMA Logic section, to delay the bus request until there is enough free space in the TxFIFO for a long, efficient burst. The TxDMA Logic can also make an urgent bus request as described in the TxDMA Logic section, where burst efficiency is sacrificed in favor of avoiding a TxFIFO under run condition.

The RxDMA process is described in the RxDMA Logic section. Typically, RxDMA requests will be forwarded to the Arbiter, however RxDMA Urgent Requests are also possible in order to prevent RxFIFO overruns.

9 Power Management

The IP100 supports operating system directed power management according to the ACPI specification. Power management registers in the PCI configuration space, as defined by the PCI Bus Power Management Interface specification, Revision 1.0 are described in 10.0.

The IP100 supports several power management states. The PowerState field in the PowerMgmtCtrl register determines IP100's current power state. The power states are defined as follows:

- D0 Uninitialized (power state 0) is entered as a result of hardware reset, or after a transition from D3 Hot to D0. This state is the same as D0 Active except that the PCI configuration registers are uninitialized. In this state, the IP100 is unable to respond to PCI I/O, memory and configuration cycles and can not operate as a PCI master The IP100 cannot signal wake (PMEN) from the D0 state.
- D0 Active (power state 0) is the normal operational power state for the IP100. In this state, the PCI configuration registers have been initialized by the system, including the IoSpace, MemorySpace, and Bus-Master bits in ConfigCommand, so the IP100 is able to respond to PCI I/O, memory and configuration cycles and can operate as a PCI master. The IP100 cannot signal wake (PMEN) from the D0 state.
- D1 (power state 1) is a "light-sleep" state. The

IP100 optionally supports this state determined by the D1Support bit in the ConfigParm word in EEPROM. The D1 state allows transition back to D0 with no delay. In this state, the IP100 responds to PCI configuration accesses, to allow the system to change the power state. In D1 the IP100 does not respond to any PCI I/O or memory accesses. The IP100's function in the D1 state is to recognize wake events and link state events and pass them on to the system by asserting the PMEN signal on the PCI bus.

- D2 (power state 2) is a partial power-down state. The IP100 optionally supports this state determined by the D2Support bit in the ConfigParm word in EEPROM. D2 allows a faster transition back to D0 than is possible from the D3 state. In this state, the IP100 responds to PCI configuration accesses, to allow the system to change the power state. In D2 the IP100 does not respond to any PCI I/O or memory accesses. The IP100's function in the D2 state is to recognize wake events and link state events and pass them on to the system by asserting the PMEN signal on the PCI bus.
- D3 Hot (power state 3) is the full power-down state for the IP100. In D3 Hot, the IP100 loses all PCI configuration information except for the value in PowerState. In this state, the IP100 responds to PCI configuration accesses, to allow the system to change the power state back to D0 Uninitialized. In D3 hot, the IP100 does not respond to any PCI I/O or memory accesses. The IP100's main responsibility in the D3 Hot state is to recognize wake events and link state events and signal those to the system by asserting the PMEN signal on the PCI bus.
- D3 Cold (power state undefined) is the power-off state for the IP100. The IP100 does not function in this state. When power is restored, the system guarantees the assertion of hardware reset, which puts the IP100 into the D0 Uninitialized state.

The IP100 can generate wake events to the system as a result of Wake Packet reception, Magic Packet reception, or due to a change in the link status. The WakeEvent register gives the host system control over which of these events are passed to the system. Wake events are signaled over the PCI bus using the PMEN pin.

A Wake Packet event is controlled by the WakePktEnable bit in WakeEvent register. WakePktEnable has no effect when IP100 is in the D0 power state, as the wake process can only take place in states D1, D2, or D3. When the IP100 detects a Wake Packet, it signals a wake event on PMEN (if PMEN assertion is enabled), and sets the WakePktEvent bit in the WakeEvent register. The IP100 can signal that a wake event has occurred when it receives a pre-defined frame from another station. The host system transfers a set of frame data patterns into the transmit FIFO using the transmit DMA function before placing the IP100 in a power-down state. Once powered down, the IP100 compares receive frames with the frame patterns in the transmit FIFO. When a matching frame is received (and also passes the filtering mode set in the ReceiveMode register), a wake event is signaled.

The frame patterns in the transmit FIFO specify which bytes in received frames are to be examined. Each byte in the transmit FIFO specifies a four bit relative offset (from the start of the received frame) in the most significant nibble and a four bit length indicator in the least significant nibble. Relative offsets describe the number of bytes of the received frame to skip from the last relevant byte, beginning with byte 0x00. Relative offsets with a value of 0xF indicate the actual relative offset is larger than 15, and is specified by the next 8 bit value in the transmit FIFO. Length indicators with a value of 0xF indicate the actual length indicator is larger than 15, and is specified by the next 8 bit value in the transmit FIFO. If both the relative offset, and the length indicator are 0xF, the first byte following the relative offset/length indicator pair is the actual relative offset, and the second following byte is the actual length indicator. A byte value of 0x00 indicates the end of the pattern for that wake frame. Immediately following the end-of-pattern is a 4-byte CRC. The calculation used to for the CRC is the same polynomial as the Ethernet MAC FCS.

An example pseudo-packet (based on the ARP packet example from Appendix A of the "OnNow Network Device Class Power Management Specification") which would be loaded into the transmit FIFO of the IP100 is shown in Figure 8.

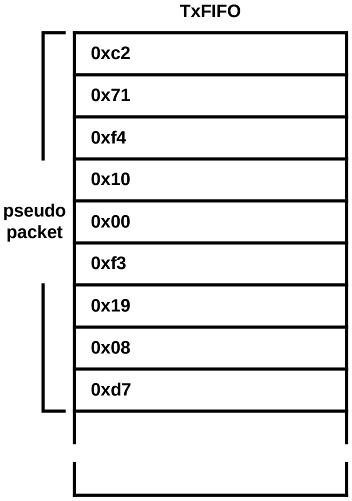


FIGURE 8: Example Psuedo Packet

Using the pseudo packet in Figure 8, the IP100 will assert a wake event if a packet of the form shown in Figure 9 is received whereby a 32-bit CRC over the indicated bytes of the received packet yields the value 0xf31908d7.

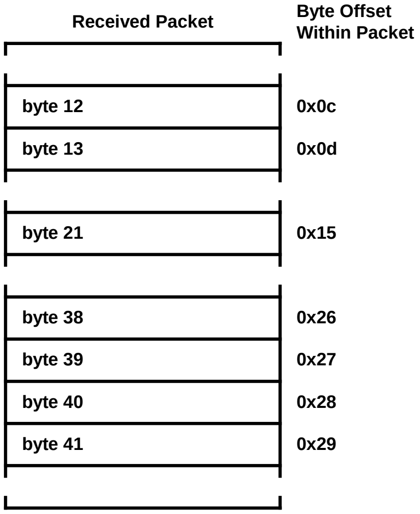


FIGURE 9: Example Wake Packet

The IP100 also supports Magic Packet™ technology developed by Advanced Micro Devices to allow remote wake-up of a sleeping station on a network via transmission of a special frame. Once the IP100 has been placed in Magic Packet mode and put to sleep, it scans all incoming frames addressed to it for a data sequence consisting of 16 consecutive repetitions of its own 48-bit Ethernet MAC StationAddress. This sequence can be located anywhere within the frame, but must be preceded by a synchronization stream.

The synchronization stream is defined as 6 bytes of 0xFF. For example, if the MAC address programmed into the StationAddress register is 0x11:22:33:44:55:66, then the IP100 would be scanning for the frame data shown in Figure 10.

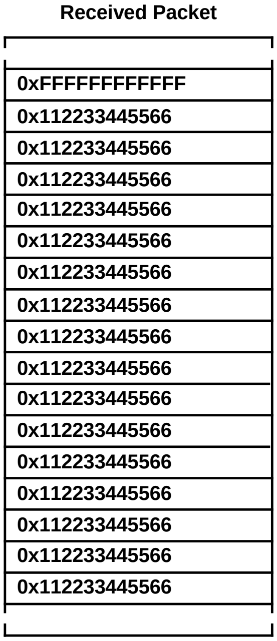


FIGURE 10: Example Magic Packet

Magic Packet wake up is controlled by the MagicPktEnable bit in the WakeEvent register. A wake event can only take place in the D1, D2, or D3 states, and MagicPktEnable has no effect when the IP100 is in the D0 power state. The Magic Packet must also pass the address matching criteria set in ReceiveMode. A Magic Packet may also be a broadcast frame. When the IP100 detects a Magic Packet, it signals a wake event on PMEN (if PMEN assertion is enabled), and sets the MagicPktEvent bit in WakeEvent.

The IP100 can also signal a wake event when it senses a change in the network link state, either from LINK_OK to LINK_FAIL, or vice versa. Link state wake is controlled by the LinkEventEnable bit in the WakeEvent register. At the time LinkEventEnable is set by the host system, the IP100 samples the current link state. It then waits for the link state to change. If the link state changes before the IP100 returns to state D0 or

LinkEventEnable is cleared, LinkEvent is set in WakeEvent, and (if it is enabled) the PMEN signal is asserted.

9.1 Wake Event

When a desired wake event occurs, the IP100 sets the appropriate event bit in the WakeEvent register, sets the PmeStatus bit in the PowerMgmtCtrl register, and asserts the WAKE signal.

The host system responds to PMEN by scanning the power management configuration registers of all devices, looking for the device which asserted PMEN. If the device with the IP100 signaled wake, the system will find PmeStatus set in IP100's PowerMgmtCtrl register. The operating system then clears the PmeEn bit in the PowerMgmtCtrl register causing PMEN to be de-asserted.

The operating system raises the power state (probably to D0) by writing to the PowerState bits in the PowerMgmtCtrl register. If the IP100 was previously in the D3 state, PCI configuration is lost and must be restored by the operating system.

The host system must set TxReset to clear any wake patterns out of the transmit FIFO (if this is not done, the patterns will be treated as frames and transmitted once the transmitter is enabled).

The host system reads the WakeEvent register to determine the wake event, and if requested, passes it back to the operating system. The host system restores any volatile state that was saved in the power down sequence. The host system re-enables interrupts by programming IntEnable. The host system restores the RxDMAList (and any other data structures required for operation). Any wake packets in the receive FIFO are transferred by receive DMA and passed to the operating system.

9.2 Power Down

D0u and D0i are mutually exclusive. The moment that the IoBaseAddress register or the MemBaseAddress is written by the host, then IP100 enters D0u. When D0u is active, the PHY is completely powered down and all clocks (AsicClk, TxClk, RxClk) are gated off except PciClk. The IP100 consumes less than 70 mA in D0u. Make sure that the LED drivers are off. When IP100 is in D0i, the PHY is powered up and ready to transmit and receive packets. In Forced Config mode (motherboard applications), IP100 is always in D0i mode. Another way to indicate D0u is to check the non-zero value in the programmable field of IoBaseAddress or MemBaseAddress. In WOL the PHY should be fully functional to receive Magic Packets. In WOL mode, D0u should be forced to low before passing to the PHY.

When in the D3 state, if PMEN is de-asserted on the PCI bus the IP100 will enter Power Down. When in Power Down mode, the PHY is completely powered down. All clocks (AsicClk, TxClk, RxClk) except PciClk are gated off. When the PCI bus is powered down, the IP100 consumes less than 5 mA.

10 Registers and Data Structures

10.1 PHY Registers

The IP100 includes a full set of PHY registers which can be accessed through the internal MDC/MDIO interface. The MAC and PHY, although integrated, act as if they are separate. The PHY registers must be accessed throught the PhyCtrl register.

10.1.1 Control Register

Class............................ PHY Registers, Control
Access Method .... Accessed through PhyCtrl register
Register Address 0x00
Default 0x3100
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	Reset	R/W	1 = Software Reset (self clearing). While the IP100 is resetting, Reset will remain a logic 1 and write attempts to any PHY Registers are not accepted. 0 = Normal operation.
14	Loopback	R/W	1 = PHY loopback mode. When Loopback is a logic 1, the IP100 will be isolated from the network media. All transmit data from the MAC will return to the MAC as receive data. Collision indications are disabled unless Collision Test is a logic 1. 0 = Normal operation.
13	Speed	R/W	1 = 100 Mb/s. 0 = 10 Mb/s.
12	Auto-Negotiation	R/W	1 = Auto-negotiation enabled. 0 = Auto-negotiation disabled.
11	Power Down	R/W	1 = PHY power down. When Power Down is a logic 1, the IP100 PHY will enter Power Down mode. While in Power Down mode, the IP100 PHY will not respond to transmit data, or received data but will respond to management transactions. 0 = Normal operation.
10	Isolate	R/W	1 = Isolate PHY. When Isolate is a logic 1, the IP100 PHY will be isolated from the MAC. When isolated, the IP100 will not respond to transmit or receive data, but will respond to management transactions. 0 = Normal operation.
9	Restart Auto-Negotiation	R/W	1 = Restart auto-negotiation (self clearing). When Restart Auto-Negotiation is a logic 1, the IP100 will restart Auto-Negotiation, depending on the Auto-Negotiation bit. If the Auto-Negotiation bit is a logic 0, then Restart Auto-Negotiation has no effect. 0 = Normal operation.
8	Duplex Mode	R/W	1 = Full duplex. 0 = Half duplex.

Preliminary, Specification subject to change without notice.

BIT	BIT NAME	R/W	BIT DESCRIPTION
7	Collision Test	R/W	1 = Enable COL test. If Collision Test is a logic 1 and transmit data is sent to the PHY, the IP100 PHY will assert the collision signal within 512 bit times (where 1 bit time = 100ns for 10Mbps operation and 1 bit time = 10ns for 100Mbps operation). When transmit data is removed, the IP100 PHY will deassert the collision signal within 4 bit times. 0 = Normal operation.
6..0	Reserved	N/A	Reserved for future use.

10.1.2 Status Register

Class............................ PHY Registers
Access Method............. Accessed through PhyCtrl register
Register Address.......... 0x01
Default 0x7849
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	100BASE-T4	R	1 = PHY 100BASE-T4 capable. 0 = PHY not 100BASE-T4 capable.
14	100BASE-X Full Duplex	R	1 = PHY 100BASE-X Half Duplex capable. 0 = PHY not 100BASE-X Half Duplex capable.
13	100BASE-X Half Duplex	R	1 = PHY 100BASE-X Half Duplex capable. 0 = PHY not 100BASE-X Half Duplex capable.
12	10BASE-T Full Duplex	R	1 = PHY 10BASE-T Full Duplex capable. 0 = PHY not 10BASE-T Full Duplex capable.
11	10BASE-T Half Duplex	R	1 = PHY 10BASE-T Half Duplex capable. 0 = PHY not 10BASE-T Half Duplex capable.
10..7	Reserved	N/A	Reserved for future use.
6	Preamble Suppression	R	1 = Management preamble may be suppressed. 0 = Management preamble required.
5	Auto-Negotiation Complete	R	1 = Auto-Negotiation complete. 0 = Auto-Negotiation not complete. If Auto-Negotiation Complete is a logic 1, the auto-negotiation process is complete and the contents of the Auto-Negotiation Link Partner Ability and Auto-Negotiation Expansion registers are valid. If Auto-Negotiation Complete is a logic 0, the Auto-Negotiation process is not complete, and the contents of the Auto-Negotiation Link Partner Ability and Auto-Negotiation Expansion registers are undefined. If the Auto-Negotiation bit of the Control Register is a logic 0, then Auto-Negotiation Complete is always a logic 0.
4	Remote Fault	R	1 = Remote Fault detected. 0 = Remote Fault not detected. If Remote Fault is a logic 1, the IP100 has detected a remote fault condition. Remote Fault will remain a logic 1 until the remote fault condition no longer exists, and Remote Fault is read by the host system.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3	Auto-Negotiation Ability	R	1 = Auto-Negotiation capable. 0 = Auto-Negotiation incapable. If Auto-Negotiation Ability is a logic 1, the IP100 is capable of performing Auto-Negotiation. Auto-Negotiation Ability depends on the external mode setting of the IP100.
2	Link Status	R	1 = Link is up. 0 = Link is down. When Link Status is a logic 0, Link Status will remain a logic 0 until the link state changes, and Link Status is read by the host system.
1	Jabber Detect	R	1 = Jabber detected. 0 = Jabber not detected. When Jabber Detect is a logic 1, Jabber Detect will remain a logic 0 until the jabber condition no longer exists, and Jabber Detect is read by the host system.
0	Extended Capability	R	1 = Extended capabilities register exists.

10.1.3 PHY Identifier 1

Class............................ PHY Registers
Access Method .... Accessed through PhyCtrl register
Register Address 0x02
Default 0x0243
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	PHY ID Number	R	Bits 3 through 18 of the OUI. OUI = 0090C3

10.1.4 PHY Identifier 2

Class............................ PHY Registers
Access Method .... Accessed through PhyCtrl register
Register Address 0x03
Default 0x0C40
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..10	PHY ID Number	R	Bits 19 through 24 of the OUI. OUI = 0090C3
9..4	Model Number	R	Manufacturer's model number.
3..0	Revision Number	R	Four bit manufacturer's revision number.

10.1.5 Auto-Negotiation Advertisement

Class............................ PHY Registers
Access Method .... Accessed through PhyCtrl register
Register Address 0x04
Default 0x01E1
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	Next Page capable	R/W	1 = Next Page capable. 0 = Next Page incapable.
14	Reserved	N/A	Reserved for future use.
13	Remote Fault	R/W	1 = Remote Fault supported. 0 = Remote Fault not supported.
12..11	Reserved	N/A	Reserved for future use.
10	Pause	R/W	1 = Pause function supported. 0 = Pause not supported.
9	100BASE-T4	R/W	1 = 100BASE-T4 capable. 0 = 100BASE-T4 incapable.
8	100BASE-TX Full Duplex	R/W	1 = PHY 100BASE-TX Full Duplex capable. 0 = PHY not 100BASE-TX Full Duplex capable.
7	100BASE-TX Half Duplex	R/W	1 = PHY 100BASE-TX Half Duplex capable. 0 = PHY not 100BASE-TX Half Duplex capable.
6	10BASE-T Full Duplex	R/W	1 = PHY 10BASE-T Full Duplex capable. 0 = PHY not 10BASE-T Full Duplex capable.
5	10BASE-T Half Duplex	R/W	1 = PHY 10BASE-T Half Duplex capable. 0 = PHY not 10BASE-T Half Duplex capable.
4..0	Selector Field	R/W	IEEE 802.3 selector field

10.1.6 Auto-Negotiation Link Partner Ability

Class............................ PHY Registers
Access Method............. Accessed through PhyCtrl register
Register Address.......... 0x05
Default 0x0000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	Next Page	R	1 = Next Page capable. 0 = Next Page incapable.
14	Acknowledge	R	1 = Link Code Word received. 0 = Link Code Word not received.
13	Remote Fault	R	1 = Remote Fault detected. 0 = Remote Fault not detected. If Remote Fault is a logic 1, then the Remote Fault bit of the Status Register register will be a logic 1.
12..10	Reserved	N/A	Reserved for future use.

Preliminary, Specification subject to change without notice.

BIT	BIT NAME	R/W	BIT DESCRIPTION
9	100BASE-T4	R	1 = 100BASE-T4 capable. 0 = 100BASE-T4 incapable.
8	100BASE-TX Full Duplex	R	1 = 100BASE-TX Full Duplex capable. 0 = Not 100BASE-TX Full Duplex capable.
7	100BASE-TX Half Duplex	R	1 = 100BASE-TX Half Duplex capable. 0 = Not 100BASE-TX Half Duplex capable.
6	10BASE-T Full Duplex	R	1 = 10BASE-T Full Duplex capable. 0 = Not 10BASE-T Full Duplex capable.
5	10BASE-T Half Duplex	R	1 = 10BASE-T Half Duplex capable. 0 = Not 10BASE-T Half Duplex capable.
4..0	Selector Field	R	Selector Field

10.1.7 Auto-Negotiation Expansion

Class............................ PHY Registers
Access Method............ Accessed through PhyCtrl register
Register Address.......... 0x06
Default 0x0000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..5	Reserved	N/A	Reserved for future use.
4	Parallel Detection	R	1= Parallel detection fault. 0 = No parallel detection fault. If Parallel Detection is a logic 1, Parallel Detection will remain a logic 1 until the parallel detection fault condition no longer exists, and Parallel Detection has been read by the host system.
3	Link partner Next Page Able	R	1 = Link partner Next Page capable. 0 = Link partner not Next Page capable.
2	Next Page Able	R	1 = Local device Next Page capable. 0 = Local device not Next Page capable.
1	Page Received	R	1 = New page received. 0 = New page not received. If Page Received is a logic 1, Page Received will remain a logic 1 Page Received has been read by the host system.
0	Link partner Auto-Negotiation Able	R	1 = Link partner Auto-Negotiation capable. 0 = Link partner not Auto-Negotiation capable.

10.1.8 10BASE-T Control

Class............................ PHY Registers
Access Method .... Accessed through PhyCtrl register
Register Address 0x12
Default 0x7000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	F_link_10	R/W	Force speed to 10M and can transmit data. 0 = Normal operation. 1 = Force 10M and can transmit data.
14	NLP enable	R/W	Tx NLP enabled. 0 = Disable normal link pulse function. 1 = Enable normal link pulse function.
13	HRTBTEN	R/W	Heartbeat enable. 0 = Disable heartbeat function. 1 = Enable heartbeat function.
12	JABEN	R/W	Jabber function enable. 0 = Disable Jabber function. 1 = Enable Jabber function.
11	reprt_en	R/W	Repeater Mode Enable. To configure the IP100 in repeater mode, reprt_en must be a logic 1. If reprt_en is a logic 0, the IP100 is in NIC mode.
10..1	Reserved	N/A	Reserved for future use.
0	r18_10_polrev	R	10BaseT Polarity 0 = 10BaseT polarity correct. 1 = 10BaseT polarity reversed.

10.1.9 100BASE-TX Control

Class............................ PHY Registers
Access Method .... Accessed through PhyCtrl register
Register Address 0x13
Default 0x4000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	r19_savepw	R/W	Save power enable.
14	r19_utpdet	R/W	Enable UTP link detection.
13	r19_load_dsp	R/W	Load load_dsp signal 1 = Fix load_dsp signal to 0. 0 = Output normal load_dsp signal.
12	r19_f_link_100	R/W	Force speed to 100M and can transmit data. 0 = Normal operation. 1 = Force 100M and can transmit data.
11..8	Reserved	N/A	Reserved for future use.
7	updated_r4_en	R/W	Update register_4 signal.

BIT	BIT NAME	R/W	BIT DESCRIPTION
6	savepw_a_en	R/W	Save power enable for analog.
5	r19_pd_ok	R	0 = Parallel detection OK. 1 = Parallel detection fail.
4	Reserved	N/A	Reserved for future use.
3	Repeater	R/O	0 = Repeater mode. 1 = DTE mode.
2..0	Reserved	N/A	Reserved for future use.

10.1.10 100BASE-TX Status

Class............................ PHY Registers
Access Method .... Accessed through PhyCtrl register
Register Address 0x14
Default 0x0010
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15	100TX_fdx	R	Nway select 100Base-TX full duplex.
14	100TX_hdx	R	Nway select 100Base-TX half duplex.
13	10BT_fdx	R	Nway select 10BaseT full duplex.
12	10BT_hdx	R	Nway select 10BaseT half duplex.
11..10	Reserved	N/A	Reserved for future use.
9	fef_on	R	Far end fault function enabled.
8..4	phyaddr[4:0]	R	PHY address bits 4 through 0.
3..2	Reserved	N/A	Reserved for future use.
1	r20_digital_lpbk	R/W	Loopback. If r20_digital_lpbk is a logic 1, the IP100 will not transmit data onto the network. All data received on the transmit interface, will be placed on the receive interface.
0	fiber_testing	R/W	Speed up fiber module link speed for testing.

10.2 DMA Data Structures

A TFD is used to move data destined for transmission onto an Ethernet network, from the host system memory to the transmit FIFO within the IP100. A TFD is 16 to 512 bytes in length, and it's location in host system memory is indicated by the value in the TxDMAListPtr register.

A RFD is used to move data obtained from an Ethernet network, from the receive FIFO within the IP100 to the host system memory. A RFD is 16 to 512 bytes in length, and it's location in host system memory is indicated by the value in the RxDMAListPtr register. There are two formats for an RFD, differentiated by the ImpliedBufferEnable bit of the RxFrameStatus field.

Figure 11 shows the two DMA data structures.

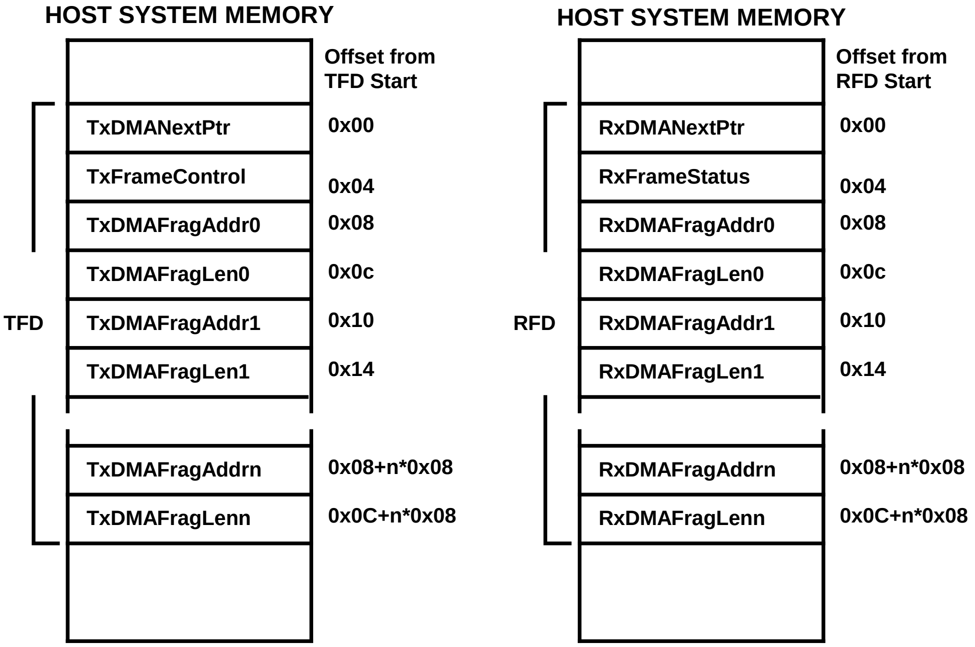


FIGURE 11: TFD and RFD DMA Data Structures

10.2.1 RxDMAFragAddr

Class............................ DMA Data Structures, RFD
Base Address .... Start of RFD
Address Offset .... 0x08+n*0x08 for nth fragment (where n=0,1,...63)
Access Mode .... Read/Write
Width .... 32 bits

BIT	BIT NAME	BIT DESCRIPTION
31..0	RxDMAFragAddr	Receive DMA Fragment Address. The RxDMAFragAddr contains the physical address of a contiguous block of system memory to which receive data is to be transferred by receive DMA. A fragment can start on any byte boundary.

10.2.2 RxDMAFragLen

Class............................ DMA Data Structures, RFD
Base Address .... Start of RFD
Address Offset .... 0x0C+n*0x08 for nth fragment (where n=0,1,...63)
Access Mode .... Read/Write
Width .... 32 bits

The RxDMAFragLen contains fragment length and control information for the block of data pointed to by the corresponding RxDMAFragAddr.

BIT	BIT NAME	BIT DESCRIPTION
12..0	FragLen	Fragment Length. The length of the contiguous block of data pointed to by the previous RxDMAFragAddr.
30..13	Reserved	Reserved for future use.
31	RxDMALastFrag	Receive DMA Last Fragment. RxDMALastFrag is set to a logic 1 by the IP100 to indicate the last fragment of the receive frame.

10.2.3 RxDMANextPtr

Class............................ DMA Data Structures, RFD
Base Address .... Start of RFD
Address Offset 0x00
Access Mode .... Read/Write
Width .... 32 bits

BIT	BIT NAME	BIT DESCRIPTION
31..0	RxDMANextPtr	Receive DMA Next Pointer. RxDMANextPtr contains the physical address of the next RFD in the receive DMA list. For the last RFD in the receive DMA list, RxDMANextPtr must be 0x00000000. RFDs must be aligned on 8-byte physical address boundaries.

10.2.4 RxFrameStatus

Class............................ DMA Data Structures, RFD
Base Address .... Start of RFD
Address Offset 0x04
Access Mode .... Read/Write
Width .... 32 bits

At the end of a receive DMA transfer, the IP100 writes the value of the RxDMAStatus register to RxFrameStatus.

BIT	BIT NAME	BIT DESCRIPTION
12..0	RxDMAFrameLen	Receive DMA Frame Length. RxDMAFrameLen indicates the true frame length, except in the case where the frame is larger than the total number of bytes specified in all of the FragLen subfields of the RxDMAFragLen RFD field in which case, the RxDMAOverflow bit will be set.
13	Reserved	Reserved for future use.

BIT	BIT NAME	BIT DESCRIPTION
14	RxFrameError	Receive Frame Error. RxFrameError indicates that an error occurred during receipt of the frame. The host system should examine RxFIFOOverrun, RxRuntFrame, RxAlignmentError, RxFCSError, and RxOversizedFrame to determine the type of error(s). RxFrameError is undefined until RxDMAComplete is a logic 1.
15	RxDMAComplete	Receive DMA Complete. RxDMAComplete indicates the frame transfer from the IP100 to the host system is complete.
16	RxFIFOOverrun	Receive FIFO Overrun. RxFIFOOverrun indicates the data was not removed from the receive FIFO fast enough to keep up with the rate data was entering the receive FIFO from the media. Bytes may be missing from the frame at one or more unpredictable locations within the frame. RxFIFOOverrun is undefined until RxDMAComplete is a logic 1.
17	RxRuntFrame	Received Runt Frame. RxRuntFrame indicates the received frame was a runt (less than 60 bytes in length, measured from the DA field to the end of the Data field). RxRuntFrame is undefined until RxDMAComplete is a logic 1.
18	RxAlignmentError	Receive Alignment Error. RxRuntFrame indicates that the received frame had an alignment error. RxAlignmentError is undefined until RxDMAComplete is a logic 1.
19	RxFCSError	Receive Frame Check Sequence Error. RxFCSError indicates a FCS checksum error on the frame data. RxFCSError is undefined until RxDMAComplete is a logic 1
20	RxOversizedFrame	Receive Oversized Frame. RxOversizedFrame indicates the frame size was equal to or greater than the value set in the MaxFrameSize register. RxOversizedFrame is undefined until RxDMAComplete is a logic 1.
22..21	Reserved	Reserved for future use.
23	DribbleBits	Dribble Bits. DribbleBits indicates that the frame had accompanying dribble bits. DribbleBits is informational only, and does not indicate a frame error.
24	RxDMAOverflow	Receive DMA Overflow. RxDMAOverflow indicates that the RFD did not have sufficient buffer space to hold all of the frame data. For this condition, the IP100 transfers as much data as possible and discards the remainder of the frame.
27..25	Reserved	Reserved for future use.
28	ImpliedBufferEnable	Implied Buffer Enable. ImpliedBufferEnable enables a special receive DMA mode. If ImpliedBufferEnable is a logic 1 when the IP100 reads the RFD, the IP100 will assume there is one receive buffer of length 1528 bytes, starting immediately after ReceiveFrameStatus at (RFD address + 0x08). The host system sets ImpliedBufferEnable when it prepares the RFD. The IP100 tests ImpliedBufferEnable before receive DMA for a frame begins at the same time it tests the RxDMAComplete bit. When the IP100 updates RxFrameStatus field at the end of the receive DMA operation (in order to set RxDMAComplete) the value written to ImpliedBufferEnable is undefined. The host system cannot assume a certain value is left in ImpliedBufferEnable after the RFD is used. Therefore, the host system must write the desired value to ImpliedBufferEnable every time after releasing a RFD to the IP100.
31..29	Reserved	Reserved for future use.

10.2.5 TxDMAFragAddr

Class............................ DMA Data Structures, TFD
Base Address .... Start of TFD
Address Offset.............. 0x08+n*0x08 for nth fragment (where n=0,1,...63)
Access Mode................ Read/Write
Width .... 32 bits

BIT	BIT NAME	BIT DESCRIPTION
31..0	TxDMAFragAddr	Transmit Fragment Address. TxDMAFragAddr contains the physical address of a contiguous block of data to be transferred from the host system to the IP100. A fragment can start on any byte boundary.

10.2.6 TxDMAFragLen

Class............................ DMA Data Structures, TFD
Base Address .... Start of TFD
Address Offset .... 0x0C+n*0x08 for nth fragment (where n=0,1,...63)
Access Mode .... Read/Write
Width .... 32 bits

Transmit Fragment Length (TxDMAFragLen) contains fragment length and control information for the block of data pointed to by the corresponding TxDMAFragAddr.

BIT	BIT NAME	BIT DESCRIPTION
12..0	FragLen	Fragment Length. FragLen is the length of the contiguous block of data pointed to by TxDMAFragAddr. The maximum fragment length is 8192 bytes.
30..13	Reserved	Reserved for future use.
31	TxDMAFragLast	Transmit DMA Last Fragment. TxDMAFragLast is set by the host system to indicate the last fragment of the transmit frame and that the IP100 should proceed to the next TFD.

10.2.7 TxDMANextPtr

Class............................ DMA Data Structures, TFD
Base Address .... Start of TFD
Address Offset 0x00
Access Mode .... Read/Write
Width .... 32 bits

BIT	BIT NAME	BIT DESCRIPTION
31..0	TxDMANextPtr	Transmit DMA Next Pointer. TxDMANextPtr contains the physical address of the next TFD in the transmit DMA list. A value of zero for TxDMANextPtr accompanies the last frame of the list and it indicates there are no more TFD's in the transmit DMA list. All TFD's must be aligned on a 8-byte physical address boundary.

10.2.8 TxFrameControl

Class............................ DMA Data Structures, TFD
Base Address .... Start of TFD
Address Offset 0x04
Access Mode .... Read/Write
Width .... 32 bits
TxFrameControl contains frame control information for the transmit DMA function and the transmit function.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>BIT DESCRIPTION</th></tr>
<tr><td>1..0</td><td>WordAlign</td><td>Word Alignment. WordAlign determine the boundary to which transmit frame lengths are rounded up in the transmit FIFO, and transmitted onto the network medium.
<table>
<tr><th>BIT 1</th><th>BIT 0</th><th>ALIGNMENT</th></tr>
<tr><td>0</td><td>0</td><td>Align to Double Word</td></tr>
<tr><td>1</td><td>0</td><td>Align t o Word</td></tr>
<tr><td>x</td><td>1</td><td>Alignment Disabled</td></tr>
</table>
When using word alignment, it is the responsibility of the host system to recognize that any added bytes necessary to achieve the desired alignment may affect byte oriented functions and fields (i.e. if the Ethernet Length/Type field holds a frame length, this value is not updated to reflect any bytes added via word alignment).</td></tr>
<tr><td>9..2</td><td>FrameId</td><td>Frame Identification. FrameId can be used as a frame ID or sequence number and can be used by the host system (via the TxStatus register) to determine frames which experienced errors.</td></tr>
<tr><td>12..10</td><td>Reserved</td><td>Reserved for future use.</td></tr>
<tr><td>13</td><td>FcsAppendDisable</td><td>FCS Append Disable. If FcsAppendDisable is a logic 1, the IP100 will not append the 4-byte FCS to the end of each transmit frame. In this case, the host system must supply the frame’s FCS as part of the data transferred via transmit DMA. An exception exists when a transmit under run occurs; in this case a guaranteed-bad FCS will be appended to the frame by the IP100. When FcsAppendDisable is a logic 0, the IP100 will compute and append FCS for each transmit frame.</td></tr>
<tr><td>14</td><td>Reserved</td><td>Reserved for future use.</td></tr>
<tr><td>15</td><td>TxIndicate</td><td>Transmit Indicate. If TxIndicate is a logic 1, the IP100 will issue a TxComplete interrupt when transmission of the frame completes.</td></tr>
<tr><td>16</td><td>TxDMAComplete</td><td>Transmit DMA Complete. When TxDMAComplete is a logic 1, the frame transfer by transmit DMA is complete. The IP100 sets TxDMAComplete to a logic 1 after completing transfer via transmit DMA, all fragments specified in the TFD.</td></tr>
<tr><td>30..17</td><td>Reserved</td><td>Reserved for future use.</td></tr>
<tr><td>31</td><td>TxDMAIndicate</td><td>Transmit DMA Indicate. If TxDMAIndicate is a logic 1, the IP100 will issue a TxDMAComplete interrupt upon completion of transmit DMA for this frame. The TFC is read twice by the IP100; the first time to write the TFC to the transmit FIFO before frame data transfer, and again after the transmit DMA operation is complete to test TxDMAIndicate in order to determine whether to generate an interrupt. This dual read process allows the host system time to change TxDMAIndicate while the transmit DMA transfer is in progress.</td></tr>
</table>

10.3 Wake Event Data Structures

The first Wake Event Data Structure is the Pseudo Packet. A Pseudo Packet is a set of patterns loaded into the IP100 TxFIFO which specify bytes to be examined within received frames. A CRC is calculated over these bytes and compared with a CRC value supplied in the Pseudo Packet. If a match is found, the IP100 issues a Wake Event. The matching technique may result in false wake events being reported to the host system. Each Pseudo Packet consists of one or more byte-offset/byte-count pairs (or Pseudo Patterns), a terminator symbol, and a 4-byte CRC value. The byte offsets within the Pseudo Patterns indicate the number of received frame bytes to be skipped in order to reach the next group of bytes to be included in the CRC calculation. The byte-counts within the Pseudo Patterns indicate the number of bytes in the next group to be included in the CRC calculation. The terminator indicates the end of the Pseudo Patterns for the Pseudo Packet. Immediately following the terminator is a 4-byte CRC. If there is another Pseudo Packet, it will immediately follow the CRC value.

The second Wake Event Data Structure is the Magic Packet. Magic Packets are uniquely formatted frames, which upon reception invoke a Wake Event by the IP100. Once the IP100 has been placed in Magic Packet mode and put to sleep, it scans all incoming frames addressed to it for a data sequence consisting of a synchronization stream followed immediately by 16 consecutive repetitions of the station's own 48-bit Ethernet MAC station address. The sequence can be located anywhere within the received frame.

The pseudo packet and Magic Packet data structures are shown in Figure 12

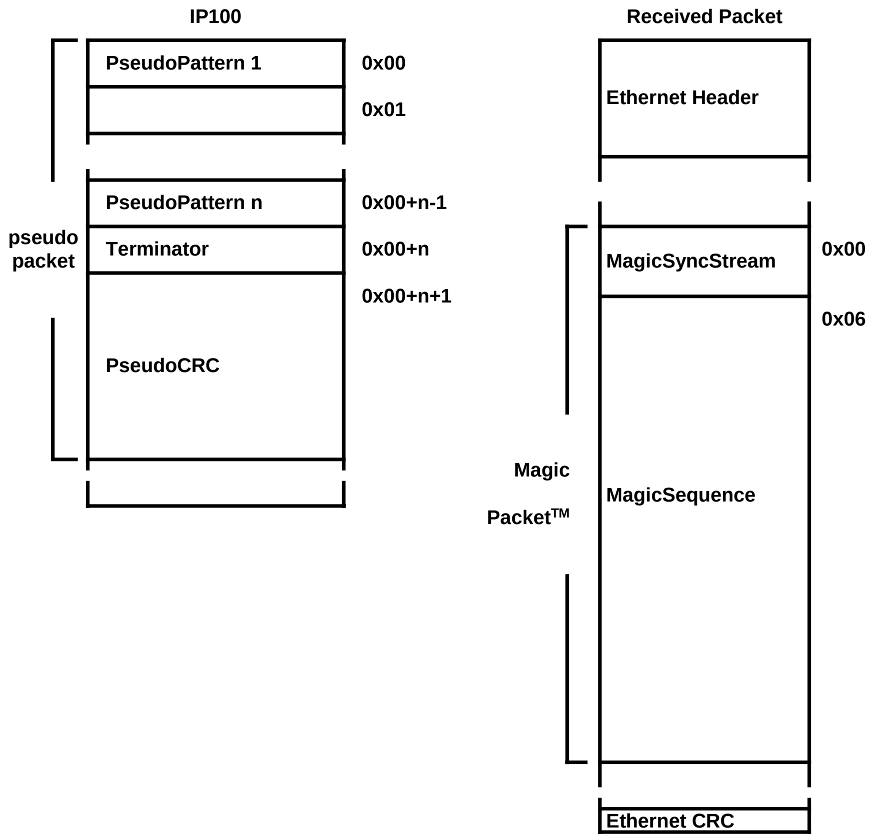


FIGURE 12 : Wake Event Data Structures, Pseudo Packet and Magic Packet

Preliminary, Specification subject to change without notice.

10.3.1 MagicSequence

Class............................ Wake Event Data Structures, Magic Packet
Base Address .... Start of Magic Packet
Address Offset 0x06
Access Mode .... Read only
Width .... 768 bits

BIT	BIT NAME	BIT DESCRIPTION
767..0	MagicSequence	Magic Sequence. A sequence of 96 bytes, consisting of 16 consecutive, identical 6 bytes sequences, where each 6 byte sequence equals the station address of the station receiving the Magic Packet.

10.3.2 MagicSyncStream

Class............................ Wake Event Data Structures, Magic Packet
Base Address .... Start of Magic Packet
Address Offset 0x00
Access Mode .... Read only
Width .... 48 bits

BIT	BIT NAME	BIT DESCRIPTION
47..0	MagicSyncStream	Magic Packet Sync Stream. A stream of 6 bytes with the value 0xff indicates the start of the MagicSequence.

10.3.3 PseudoCRC

Class............................ Wake Event Data Structures, Pseudo Packet
Base Address .... Start of Pseudo Packet
Address Offset .... 0x00+n+1 for n PseudoPatterns
Access Mode .... Write only
Width .... 32 bits
The 32-bit CRC as defined in the IEEE 802.3 Ethernet standard for the FCS, taken over the bytes (indicated by the PseudoPattern values) of a received frame.

BIT	BIT NAME	BIT DESCRIPTION
7..0	PsuedoCRCbyte0	The least significant byte of the PseudoCRC.
15..8	PsuedoCRCbyte1	The second lest significant byte of the PseudoCRC.
23..16	PsuedoCRCbyte2	The second most significant byte of the PseudoCRC.
31..24	PsuedoCRCbyte3	The most significant byte of the PseudoCRC.

10.3.4 PseudoPattern

Class............................. Wake Event Data Structures, Pseudo Packet
Base Address .... Start of Pseudo Packet
Address Offset .... 0x00 thru 0x00+n-1 for nth PseudoPattern
Access Mode .... Write only
Width .... 8 bits

BIT	BIT NAME	BIT DESCRIPTION
3..0	ByteCount	ByteCount can take on a value of 0x0 to 0xe. A value of 0xf indicates an extended value. The extended value will occupy 8 bits and is contained in the next PseudoPattern. If both the ByteOffset and the ByteCount values are 0xf, the next PseudoPattern will be the extended ByteOffset, and the PseudoPattern after that will be the extended ByteCount.
7..4	ByteOffset	ByteOffset can take on a value of 0x0 to 0xe. A value of 0xf indicates an extended value. The extended value will occupy 8 bits and is contained in the next PseudoPattern. If both the ByteOffset and the ByteCount values are 0xf, the next PseudoPattern will be the extended ByteOffset, and the PseudoPattern after that will be the extended ByteCount.

10.3.5 Terminator

Class............................. Wake Event Data Structures, Pseudo Packet
Base Address .... Start of Pseudo Packet
Address Offset .... 0x00+n for n PseudoPattern
Access Mode .... Write only
Width .... 8 bits

BIT	BIT NAME	BIT DESCRIPTION
7..0	Terminator	A value of 0x00 indicates the end of the PseudoPattern.

10.4 LAN I/O Registers

The host interacts with the IP100 mainly through slave registers, which occupy 128 bytes in the host system's I/O space, memory space, or both. Generally, registers are referred to as "I/O registers", implying that the registers may in fact be mapped and accessed by the host system in memory space. I/O registers must be accessed with instructions that are no larger than the bit-width of that register.

The IP100 LAN I/0 register layout is show in Figure 13.

<table>
<tr><th>BYTE 3</th><th>BYTE 2</th><th>BYTE 1</th><th>BYTE 0</th><th>ADDR OFFSET</th></tr>
<tr><td colspan="4">HashTable[63..32]</td><td>0x64</td></tr>
<tr><td colspan="4">HashTable[31..0]</td><td>0x60</td></tr>
<tr><td></td><td>PhyCtrl</td><td>TxReleaseThresh</td><td>ReceiveMode</td><td>0x5C</td></tr>
<tr><td colspan="2">MaxFrameSize</td><td colspan="2">StationAddress[47..32]</td><td>0x58</td></tr>
<tr><td colspan="4">StationAddress[31..0]</td><td>0x54</td></tr>
<tr><td colspan="2">MACCtrl1</td><td colspan="2">MACCtrl0</td><td>0x50</td></tr>
<tr><td colspan="2">IntStatus</td><td colspan="2">IntEnable</td><td>0x4C</td></tr>
<tr><td colspan="2">IntStatusAck</td><td></td><td></td><td>0x48</td></tr>
<tr><td colspan="2">TxStatus</td><td>WakeEvent</td><td>ExpRomData</td><td>0x44</td></tr>
<tr><td colspan="4">ExpRomAddr</td><td>0x40</td></tr>
<tr><td colspan="2">FIFOCtrl</td><td></td><td></td><td>0x38</td></tr>
<tr><td colspan="2">EepromCtrl</td><td colspan="2">EepromData</td><td>0x34</td></tr>
<tr><td colspan="4">AsicCtrl</td><td>0x30</td></tr>
<tr><td colspan="4">ForceEvent</td><td>0x2C</td></tr>
<tr><td colspan="4">FunctionEventMask</td><td>0x28</td></tr>
<tr><td colspan="4">FunctionEvent</td><td>0x24</td></tr>
<tr><td colspan="4">FunctionPresentState</td><td>0x20</td></tr>
<tr><td></td><td></td><td></td><td>DebugCtrl</td><td>0x1C</td></tr>
<tr><td></td><td></td><td colspan="2">Countdown</td><td>0x18</td></tr>
<tr><td></td><td>RxDMAPollPeriod</td><td>RxDMAUrgentThresh</td><td>RxDMABurstThresh</td><td>0x14</td></tr>
<tr><td></td><td colspan="2">RxDMAStatus</td><td></td><td>0x0C</td></tr>
<tr><td></td><td>TxDMAPollPeriod</td><td>TxDMAUrgentThresh</td><td>TxDMABurstThresh</td><td>0x08</td></tr>
<tr><td colspan="4">TxDMAListPtr</td><td>0x04</td></tr>
<tr><td colspan="4">DMACtrl</td><td>0x00</td></tr>
</table>

FIGURE 13 : IP100 I/O Register Map

Preliminary, Specification subject to change without notice.

10.4.1 AsicCtrl

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x30
Default .... 0x00004000 (default values for ExpRomSize, PhySpeed10, PhySpeed100, PhyMedia are dependent on EEPROM settings, and ForcedConfig[1..0], ForcedCon-fig[2] are dependent on ED signal pin states)
Width .... 32 bits

AsicCtrl provides chip-specific, non-host-related settings. The contents of the least significant byte of AsicCtrl are read from EEPROM at reset.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>0</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>1</td><td>ExpRomSize</td><td>R</td><td>Expansion ROM Size. ExpRomSize specifies the size of the Expansion ROM interfaced to the IP100. If the expansion ROM is 32kB, set ExpRomSize to a logic 0. If the expansion ROM is 64kB, set ExpRomSize to a logic 1.</td></tr>
<tr><td>2</td><td>TxLargeEnable</td><td>R/W</td><td>Transmit Large Frames Enable. If TxLargeEnable is a logic 1, the IP100 may transmit frames which are larger in size than the IP100 transmit FIFO.</td></tr>
<tr><td>3</td><td>RxLargeEnable</td><td>R/W</td><td>Receive Large Frames Enable. If RxLargeEnable is a logic 1, the IP100 may receive frames which are larger than the IP100 receive FIFO.</td></tr>
<tr><td>4</td><td>ExpRomDisable</td><td>R/W</td><td>Expansion ROM Disable. If ExpRomDisable is a logic 1, accesses to the on-adapter Expansion ROM are disabled and read to the Expansion ROM return 0x00000000 while writes to the Expansion ROM are ignored.</td></tr>
<tr><td>5</td><td>PhySpeed10</td><td>R</td><td>Physical Device 10Mbps Capable. If PhySpeed10 is a logic 1, the IP100 PHY is capable of operating at 10Mbps.</td></tr>
<tr><td>6</td><td>PhySpeed100</td><td>R</td><td>Physical Device 100Mbps Capable. If PhySpeed100 is a logic 1, the IP100 PHY is capable of operating at 100Mbps.</td></tr>
<tr><td>7</td><td>PhyMedia</td><td>R</td><td>Physical Device Media Type. If PhyMedia is a logic 1, copper media is in use. If PhyMedia is a logic 0, fiber media is in use. The combination of PhyMedia, PhySpeed100, and PhySpeed10 defines the capabilities of the PHY.
<table>
<tr><th>PHYMEDIA</th><th>PHYSPEED100</th><th>PHYSPEED10</th><th>PHY CAPABILITY</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Undefined</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>10BASE-T</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>100BASE-T</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>100BASE-T or 10BASE-T</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Undefined</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>10BASE-F</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>100BASE-F</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>100BASE-F or 10BASE-F</td></tr>
</table></td></tr>
</table>

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>9..8</td><td>ForcedConfig[1..0]</td><td>R/W</td><td>Forced Configuration. ForcedConfig[1..0] is used to enable and select a Forced Configuration mode for the IP100. Forced Configuration mode is targeted toward embedded applications which do not utilize an EEPROM. In Forced Configuration mode, the IP100 is accessed via a PCI bus without first performing PCI configuration or loading parameters from an EEPROM.
The ForcedConfig[1..0] bits 9 through 8 can also be set on reset using ED[4:3].
<table>
<tr><th>BIT 9</th><th>BIT 8</th><th>FORCED CONFIGURATION MODE</th></tr>
<tr><td>0</td><td>0</td><td>None</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>X</td><td>Reserved</td></tr>
</table>
In Forced Configuration mode 1, the IP100 is configured as follows:
I/O base address = 0x200
I/O target cycles = enabled
Memory target cycles = disabled
Bus master cycles = enabled
Expansion ROM cycles = disabled.</td></tr>
<tr><td>10</td><td>ForcedConfig[2]</td><td>R/W</td><td>Forced Configuration Bit 2. ForcedConfig[2] is used to select an alternate Vendor and Device for the IP100.
The ForcedConfig[2] can also be set on reset using ED[2].
<table>
<tr><th>FORCEDCONFIG[2]</th><th>VENDOR ID</th><th>DEVICE ID</th></tr>
<tr><td>0</td><td>0x13F0</td><td>0x0201</td></tr>
</table></td></tr>
<tr><td>12..11</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>13</td><td>SpeedupMode</td><td>R/W</td><td>Speed Up Mode. SpeedupMode is used for simulation purposes only. When SpeedupMode is a logic 1 IP100 operation is modified to decrease simulation time. SpeedupMode can also be set on reset using signal pin ED[5].</td></tr>
<tr><td>14</td><td>LEDMode</td><td>R/W</td><td>Light Emitting Diode Mode. LEDMode is used to control the LED signal pin (EOEN, LEDDPLXN, LEDPWRN) functionality. When LEDMode is a logic 0 the LED signal pins operate in LED mode 0. When LEDMode is a logic 1 the LED signal pins operate in LED mode 1.
Note, when LED signals alternate between logic 1/0, they alternate over a 41.89ms period where the logic 0 (or LED ON) state persists for 5.24ms and the logic 1 (or LED OFF) state persists for 36.65ms.</td></tr>
</table>

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td></td><td></td><td></td><td><table>
<tr><td>LED SIGNAL PIN</td><td>MODE 0</td><td>MODE 1</td></tr>
<tr><td>LEDPWRN</td><td>Continuous logic 0 when power is applied. Alternating logic 1/0 when frame transmission in progress.</td><td>Continuous logic 0 when power is applied.</td></tr>
<tr><td>EOEN</td><td>Continuous logic 0 when Ethernet link is valid. Alternating logic 1/0 when frame reception in progress.</td><td>Continuous logic 0 when Ethernet link is valid. Alternating logic 1/0 when frame reception or transmission in progress.</td></tr>
<tr><td>LEDDPLXN</td><td>Continuous logic 0 when the IP100 is configured for full duplex operation. Alternating logic 1/0 when the IP100 detects a collision.</td><td>Continuous logic 0 when the IP100 is configured for full duplex operation. Continuous logic 1 when the IP100 is configured for half duplex operation.</td></tr>
<tr><td>LEDSPDN</td><td colspan="2">Continuous logic 0 when Ethernet link speed is 100Mbps, and continuous logic 1 when Ethernet link speed is 10Mbps.</td></tr>
</table></td></tr>
<tr><td>15</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>16</td><td>GlobalReset</td><td>W</td><td>Global Reset. When GlobalReset is a logic 1, the IP100 resets the logic functions and registers specified by the DMA, FIFO, Network, Host, and AutoInit bits (related to both the transmit and receive processes as applicable). The LAN PCI Configuration Registers are not affected by GlobalReset. GlobalReset is self-clearing.</td></tr>
<tr><td>17</td><td>RxReset</td><td>W</td><td>Receive Reset. When RxReset is a logic 1 the IP100 resets all of the receive logic functions and registers specified by the DMA, FIFO, and Network bits. RxReset is self-clearing, and should not be used after initialization except to recover from receive errors such as a receive FIFO over run.</td></tr>
<tr><td>18</td><td>TxReset</td><td>W</td><td>Transmit Reset. When TxReset is a logic 1 the IP100 resets all of the transmit logic functions and registers specified by the DMA, FIFO, and Network bits. TxReset is self clearing, and is required to be used after a transmit underrun error.</td></tr>
<tr><td>19</td><td>DMA</td><td>W</td><td>DMA Reset. DMA selects (when a logic 1) or excludes (when a logic 0) the IP100 DMA functions and registers (see below) for/from reset based on the value of the GlobalReset, RxReset, and TxReset bits. The DMA bit is self-clearing.</td></tr>
<tr><td>20</td><td>FIFO</td><td>W</td><td>FIFO Reset. FIFO selects (when a logic 1) or excludes (when a logic 0) the IP100 FIFO functions and registers for/from reset based on the value of the GlobalReset, RxReset, and TxReset bits. The FIFO bit is self-clearing.</td></tr>
<tr><td>21</td><td>Network</td><td>W</td><td>Network Reset. Network selects (when a logic 1) or excludes (when a logic 0) the IP100 network functions and registers for/from reset based on the value of the GlobalReset, RxReset, and TxReset bits. The Network bit is self-clearing.</td></tr>
</table>

BIT	BIT NAME	R/W	BIT DESCRIPTION
22	Host	W	Host Reset. Host selects (when a logic 1) or excludes (when a logic 0) the IP100 host bus interface logic functions and registers for/from reset based on the value of the GlobalReset bit. The Host bit is self-clearing.
23	AutoInit	W	Automatic Initialization Reset. AutoInit selects (when a logic 1) or excludes (when a logic 0) the IP100 auto-initialization logic function for/from re-loading IP100 parameters from an EEPROM based on the value of the GlobalReset bit. The AutoInit bit is self-clearing.
24	Reserved	N/A	Reserved for future use.
25	InterruptRequest	W	Interrupt Request. When InterruptRequest is a logic 1, the IntRequested bit of the IntStatus register is set to a logic 1. InterruptRequest is self-clearing.
26	ResetBusy	R/W	Reset Busy. When ResetBusy is a logic 1 a reset process is in progress. After asserting a reset using the GlobalReset, RxReset, or TxReset bits, the ResetBusy bit must be polled (or periodically read) until it is a logic 0 indicating the reset operation is complete.
29..27	Reserved	N/A	Reserved for future use.
30	Reserved/CardBus	N/A R	Reserved for future use. Card Bus Select. If Reserved/CardBus is a logic 1, the host interface of the IP100 is CardBus. If Reserved/CardBus is a logic 0, the host interface of the IP100 is PCI (including miniPCI). Reserved/CardBus is set via EA[0] (with a logic inversion) at the end of a reset via the RSTN signal or a IP100 power cycle. When Reserved/CardBus is a logic 0, the CardBus functionality needs to be disabled.
31	Reserved	N/A	Reserved for future use.

10.4.2 Countdown

Class............................ LAN I/O Registers, Interrupt
Base Address .... IoBaseAddress register value
Address Offset 0x18
Default 0x00000000
Width .... 16 bits

Countdown is a programmable down-counter that will generate an interrupt upon its expiration. If the CountdownIntEnable bit in DMACtrl is set, the IntRequested interrupt will be generated when Countdown counts through zero. Countdown has two modes of operation that is selected by the CountdownMode bit in DMACtrl. When CountdownMode is cleared, Countdown is loaded by the host software with an initial countdown value, then decrements at a rate determined by the CountdownSpeed bit in DMACtrl. When Countdown reaches zero, it continues to count down, wrapping to 0xFFFF. When CountdownMode is set, Countdown begins counting only when TxDMAComplete in IntStatus becomes set.

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	Countdown	R	Value of current state of Countdown timer.

10.4.3 DebugCtrl

Class............................ LAN I/O Registers, Interrupt
Base Address .... IoBaseAddress register value
Address Offset 0x1c
Default 0x00
Width .... 8 bits
The DebugCtrl register configures the IP100 to enable or disable logic to handle known failure modes.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	RetryLockEnable	R/W	Retry Lock Enable. RetryLockEnable configures the IP100 DMA operation if a retry deadlock condition is encountered. A retry deadlock may occur if a PCI transaction between the IP100 and the host system is terminated with a Retry PCI command. In this situation, the IP100 DMA arbitration logic may switch between the up and down operations which may result in a second PCI transaction request onto the PCI bus. This second PCI transaction request must complete before the original, retried transaction is placed back onto the PCI bus. The scenario described above situation is a violation of PCI protocol and may create a deadlock situation if IP100 is interacting with a PCI master device (i.e. system controller or bridge) which requires transactions to complete in order. If RetryLockEnable is a logic 0, the IP100 will operate as described above, and may deadlock if a PCI transaction is terminated with a retry. If RetryLockEnable is a logic 1, and a transaction is terminated with a retry, the IP100 will remain operating in its current state (up or down).
7..1	Reserved	N/A	Reserved for future use.

10.4.4 DMACtrl

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x00
Default 0x00000000
Width .... 32 bits
DMACtrl controls some of the bus master functions in the receive DMA and transmit DMA logic, and contains status bits. DMACtrl is cleared by a reset.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	RxDMAHalted	R	Receive DMA Halted. RxDMAHalted is a logic 1 whenever receive DMA is halted by setting RxDMAHalt or an implicit halt due to fetching a RFD with RxDMAComplete in RxFrameStatus already a logic 1. RxDMAHalted is cleared by setting RxDMAResume to a logic 1.
1	TxDMACmplReq	R	Transmit DMA Complete Request. TxDMACmplReq is equivalent to the TxDMAIndicate field in the TxFrameControl of the current TFD.
2	TxDMAHalted	R	Transmit DMA Halted. TxDMAHalted is a logic 1whenever transmit DMA is halted by setting TxDMAHalt. TxDMAHalted is cleared by setting TxDMAResume to a logic 1.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3	RxDMAComplete	R	Receive DMA Complete. RxDMAComplete is equivalent to the RxDMAComplete bit in the IntStatus register. RxDMAComplete is different from the RxDMAComplete bit in RxDMAStatus. RxDMAComplete is latched once a frame receive DMA transfer has completed. RxDMAComplete is cleared by acknowledging the RxDMAComplete bit in the IntStatus register.
4	TxDMAComplete	R	TxDMAComplete. TxDMAComplete is the same as the TxDMAComplete bit in IntStatus register. TxDMAComplete is cleared by acknowledging the TxDMAComplete bit in the IntStatus register.
7..5	Reserved	N/A	Reserved for future use.
8	RxDMAHalt	W	Receive DMA Halt. If RxDMAHalt is a logic 1, the receive DMA is halted. RxDMAHalt is self-clearing and writing a 0 is ignored. See RxDMAHalted to determine the running state of receive DMA.
9	RxDMAResume	W	Receive DMA Resume. If RxDMAResume is a logic 1, the receive DMA is resumed. RxDMAResume is self-clearing and writing a 0 is ignored. See RxDMAHalted to determine the running state of receive DMA.
10	TxDMAHalt	W	Transmit DMA Halt. If TxDMAHalt is a logic 1, the transmit DMA is halted. TxDMAHalt is self-clearing and writing a 0 is ignored. See TxDMAHalted to determine the running state of transmit DMA.
11	TxDMAResume	R/W	Transmit DMA Resume. If TxDMAResume is a logic 1, the transmit DMA is resumed. TxDMAResume is self-clearing and writing a 0 is ignored. See TxDMAHalted to determine the running state of transmit DMA.
13..12	Reserved	N/A	Reserved for future use.
14	TxDMAInProg	R	Transmit DMA in Progress. If TxDMAInProg is a logic 1, a transmit DMA operation is in progress. TxDMAInProg is primarily used by the host system in an under run recovery routine. The host system waits for TxDMAInProg to be a logic 0 before setting the TxReset bit of the AsicCtrl register to clear the under run condition. Before checking TxDMAInProg, issue TxDMAHalt.
15	DMAHaltBusy	R	DMA Halt Busy. DMAHaltBusy indicates that a DMA Halt operation (TxDMAHalt or RxDMAHalt) is in progress and the host system should wait for DMAHaltBusy to be cleared before performing other actions.
16	Reserved	N/A	Reserved for future use.
17	Reserved	R/W	Reserved for future use.
18	CountdownSpeed	R/W	Countdown Speed. CountdownSpeed sets the speed at which the Countdown register decrements. When CountdownSpeed is a logic 1, the decrement rate of Countdown is once every 3.2 us (i.e. 4 byte times at 10 Mbps). When CountdownSpeed is a logic 1, the decrement rate of Countdown is once every 320 ns (i.e. 4 byte times at 100 Mbps). By setting appropriate CountdownSpeed for the wire speed, conversions can be made between byte times and counter values using simple shift operations.

BIT	BIT NAME	R/W	BIT DESCRIPTION
19	CountdownMode	R/W	Countdown Mode. CountdownMode controls the operating mode of the Countdown register. With CountdownMode is a logic 0, Countdown begins decrementing as soon as a nonzero value is written to Countdown. With CountdownMode is a logic 1, Countdown will not begin decrementing until the TxDMAComplete bit in the IntStatus register is a logic 1. See the Countdown register definition for more information on the Countdown modes.
20	MWIDisable	R/W	PCI MWI Command Disable. If Setting MWIDisable is a logic 1, the IP100 will not use the Memory Write Invalidate (MWI) PCI command.
21	Reserved	N/A	Reserved for future use.
22	RxDMAOverrun-Frame	R/W	Receive DMA Overrun Frame. If RxDMAOverrunFrame is a logic 0, receive DMA will discard receive overrun frames without transferring them to the host system. WhenRxDMAOverrunFrame is a logic 1, receive DMA will transfer overrun frames to the host system. Overrun frames are any frame which is received while the receive FIFO is full.
23	CountdownIntEnable	R	Countdown Interrupt Enable. CountdownIntEnable specifies whether expiration of the Countdown register will generate an interrupt. If CountdownIntEnable is a logic 0, Countdown expiration will not set the IntRequested bit of the IntStatus register. IfCountdownIntEnable is a logic 1, expiration of Countdown will set the IntRequested bit of the IntStatus register. The state of CountdownIntEnable is set by the IP100. CountdownIntEnable is cleared automatically by if the IntRequested bit in the IntStatus register is a logic 1, or when a zero value is written to Countdown. CountdownIntEnable is set implicitly when a non-zero value is written into Countdown. This allows the host to write a non-zero value to Countdown and an interrupt will be generated in a corresponding amount of time. By writing a zero value to Countdown the host can suppress interrupts.
29..24	Reserved	N/A	Reserved for future use.
30	TargetAbort	R	Bus Target Abort. TargetAbort is a logic 1 when the IP100 experiences a target abort sequence when operating as a bus master. TargetAbort indicates a fatal error, and must be cleared before further transmit DMA or receive DMA operation can proceed. TargetAbortt is cleared via the GlobalReset, and DMA bits of the AsicCtrl register.
31	MasterAbort	R	Bus Master Abort. MasterAbort is a logic 1 when the IP100 experiences a master abort sequence when operating as a bus master. MasterAbort indicates a fatal error, and must be cleared before further transmit DMA or receive DMA operation can proceed. MasterAbortt is cleared via the GlobalReset, and DMA bits of the AsicCtrl register.

10.4.5 EepromCtrl

Class............................ LAN I/O Registers, External Interface Control
Base Address .... IoBaseAddress register value
Address Offset 0x36
Default 0x0000
Width .... 16 bits

EepromCtrl provides the host with a method for issuing commands to the IP100's serial EEPROM controller. Individual 16-bit word locations within the EEPROM may be written, read or erased. Also, the EEPROM's WriteEnable, WriteDisable, EraseAll and WriteAll commands can be issued. Two-bit opcodes and 8-bit addresses are written to this register to cause the IP100 to carry out the desired EEPROM command. If data is to be written to the EEPROM, the 16-bit data word must be written to EepromData by the host prior to issuing the associated write command. Similarly, if data is to be read from the EEPROM, the read data will be available via EepromData register. The EEPROM is a slow device, and it is important that the host system use the EepromBusy bit to determine when a new command may be issued to the EEPROM.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>7..0</td><td>EepromAddress</td><td>R/W</td><td>EEPROM Address. EepromAddress identify one of the 256, sixteen-bit words to be the target for the ReadRegister, WriteRegister or EraseRegister commands.
Bits 7 and 6 are further defined to identify a sub-command based on the value of EepromOpcode. The definition of bits 7 and 6 are valid when the EepromOpcode in bits 9 and 8 equals 00.
<table>
<tr><th>BIT 7</th><th>BIT 6</th><th>SUB-COMMAND</th></tr>
<tr><td>0</td><td>0</td><td>WriteDisable</td></tr>
<tr><td>0</td><td>1</td><td>WriteAll</td></tr>
<tr><td>1</td><td>0</td><td>EraseAll</td></tr>
<tr><td>1</td><td>1</td><td>WriteEnable</td></tr>
</table></td></tr>
<tr><td>9..8</td><td>EepromOpcode</td><td>R/W</td><td>EEPROM Operation Code. EepromOpcode specifies one of three individual commands and a single group of four sub-commands.
<table>
<tr><th>BIT 9</th><th>BIT 8</th><th>OPCODE COMMAND</th></tr>
<tr><td>0</td><td>0</td><td>Write Enable/Disable & Write/
Erase All sub-commands</td></tr>
<tr><td>0</td><td>1</td><td>WriteRegister</td></tr>
<tr><td>1</td><td>0</td><td>ReadRegister</td></tr>
<tr><td>1</td><td>1</td><td>EraseRegister</td></tr>
</table>
Note, after every WriteRegister, EraseRegister opcode, or WriteAll or EraseAll subcommand, the IP100 will automatically issue a WriteDisable command to the EEPROM. Therefore, a WriteEnable command must be issued to the EEPROM prior to every WriteRegister, EraseRegister opcode, or WriteAll or EraseAll subcommand.</td></tr>
<tr><td>14..10</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>15</td><td>EepromBusy</td><td>R</td><td>EEPROM Busy. EepromBusy is a logic 1 during the execution of EEPROM commands. Further commands should not be issued to EepromCtrl nor should data be read from Eeprom-Data while EepromBusy is a logic 1.</td></tr>
</table>

10.4.6 EepromData

Class............................ LAN I/O Registers, External Interface Control
Base Address .... IoBaseAddress register value
Address Offset 0x34
Default 0x0000
Width .... 16 bits
EepromData is a 16-bit data register for use with the adapter's serial EEPROM. Data from the EEPROM should be read by the host system from EepromData register based on the state of the EepromBusy bit of the EepromCtrl register. Data to be written to the EEPROM is written to EepromData prior to issuing the write command to EepromCtrl.

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	EepromData	R/W	EEPROM Data. Data read from, or to be written to, the external EEPROM.

10.4.7 ExpRomAddr

Class............................ LAN I/O Registers, External Interface Control
Base Address .... IoBaseAddress register value
Address Offset 0x40
Default 0x00000000
Width .... 32 bits
ExpRomAddr holds the address to be used for direct I/O accesses of the Expansion ROM. To access a byte in the Expansion ROM, write the address of the byte to be accessed into ExpRomAddr. Then issue either a read or a write to ExpRomData. For reads, the ROM value will be returned by the read instruction. For writes, the new value will be programmed into the ROM upon completion of the write instruction.

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	ExpRomAddr	R/W	Expansion ROM Address. Address used for accessing expansion ROM.
31..16	Reserved	N/A	Reserved for future use.

10.4.8 ExpRomData

Class............................ LAN I/O Registers, External Interface Control
Base Address .... IoBaseAddress register value
Address Offset 0x44
Default 0x00
Width .... 8 bits
ExpRomData is the data port for performing direct I/O byte-wide accesses of the Expansion ROM. A read of ExpRomData returns the ROM byte value from the location specified by ExpRomAddr. A write to ExpRomData causes the write data to be programmed into the ROM location specified by ExpRomAddr.
Note: The Atmel EPROM devices supported by IP100 must be programmed in 64-byte pages. Refer to the Atmel Flash Memory Device data book for information on programming EPROMs.

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	ExpRomData	R/W	Expansion ROM Data. Data read from, or to be written to, expansion ROM.

10.4.9 FIFOCtrl

Class............................ LAN I/O Registers, FIFO Control
Base Address .... IoBaseAddress register value
Address Offset 0x3a
Default 0x0000
Width .... 16 bits
FIFOCtrl provides various control and indications for the transmit FIFO and the receive FIFO diagnostic.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	RAMTest Mode	R/ W	RAM Tes t Mode. I f RAMTestMode is a logic 1, the FIFO RAM is in the test mode.
8..1	Reserved	N/A	Reserved for future use.
9	RxOverrunFrame	R/ W	Receive Overrun Frame. RxOverrunFrame determines how the IP100 handles receive overrun frames. If RxOverrunFrame is a logic 0, the IP100 discards all overrun frames. If RxOverrunFrame is a logic 1, the IP100 will retain all overrun frames, so that they may be inspected by the host for diagnostic purposes.
10	Reserved	N/A	Reserved for future use.
11	RxFIFOFull	R	Receive FIFO Full. If RxFIFOFull is a logic 1, the receive FIFO is full. RxFIFOFull does not in itself indicate an overrun condition. However, if more data is received while RxFIFOFull is a logic 1, an overrun will occur. RxFIFOFull is cleared as soon as the receive FIFO is no longer full.
13..12	Reserved	N/A	Reserved for future use.
14	Transmitting	R	Transmit Indicator. Transmitting is set to a logic 1 whenever a frame is being transmitted or during a transmit deferral).
15	Receiving	R	Receive Indicator. Receiving is set to a logic 1 whenever a frame is being received. No action is expected on the part of the host based on the state of Receiving.

10.4.10 ForceEvent

Class............................ LAN I/O Registers, CardBus Status Change
Base Address .... IoBaseAddress register value
Address Offset 0x2c
Default Value 0x00000000
Width .... 32 bits
ForceEvent is used to force values in the FunctionEvent register. Writing a logic 1 to any bit in ForceEvent sets the corresponding bit in the FunctionEvent register, simulating a status change event. If other functions are active, they may alter the values previously forced into the FunctionEvent register.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3..0	Reserved	N/A	Reserved for future use.
4	GWAKE	R/W	GWAKE. Writing a logic 1 to GWAKE sets the GWAKE bit of the FunctionEvent register, but not the GWAKE bit of the FunctionPresentState register.
14..5	Reserved	N/A	Reserved for future use.
15	INTR	R/W	INTR. Writing a logic 1 to INTR sets the INTR bit of the FunctionEvent register, but not the INTR bit of the FunctionPresentState register.

Preliminary, Specification subject to change without notice.

BIT	BIT NAME	R/W	BIT DESCRIPTION
31..16	Reserved	N/A	Reserved for future use.

10.4.11 FunctionEvent

Class............................ LAN I/O Registers, CardBus Status Change
Base Address .... IoBaseAddress register value
Address Offset 0x24
Default Value 0x00000000
Width .... 32 bits
FunctionEvent contains the bits which can be used to generate status change interrupts, depending upon the setting of the FunctionEventMask register. This register is disabled when the CardBus bit of AsicCtrl is low.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3..0	Reserved	N/A	Reserved for future use.
4	GWAKE	R/W	GWAKE. GWAKE is a logic 1 and remains a logic 1 when the GWAKE bit of the FunctionPresentState register is a logic 1. GWAKE is cleared by writing a logic 1 to GWAKE. Writing a logic 1 to GWAKE also clears the PmeStatus bit of the PowerMgmtCtrl register.
14..5	Reserved	N/A	Reserved for future use.
15	INTR	R/W	INTR. INTR is a logic 1 whenever an interrupt is pending regardless of the INTR bit in the FunctionEventMask register. INTR is cleared by writing a logic 1 to INTR.
31..16	Reserved	N/A	Reserved for future use.

10.4.12 FunctionEventMask

Class............................ LAN I/O Registers, CardBus Status Change
Base Address .... IoBaseAddress register value
Address Offset 0x28
Default Value 0x00000000
Width .... 32 bits
FunctionEventMask masks the FunctionEvent register. This register is disabled when the CardBus bit of AsicCtrl is low.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3..0	Reserved	N/A	Reserved for future use.
4	GWAKE	R/W	GWAKE. If GWAKE is a logic 1, and WKUP is a logic 1 the GWAKE bit of the FunctionEvent register may assert the CSTSCHGN signal.
13..5	Reserved	N/A	Reserved for future use.
14	WKUP	R/W	WKUP. If WKUP is a logic 1, all events (WriteProtect, Ready/Busy, BatterryVoltageDetect, and GWAKE) may assert the CSTSCHGN signal.
15	INTR	R/W	INTR. If INTR is a logic 1, setting the INTR bit of the FunctionEvent register will assert the INTAN.
31..16	Reserved	N/A	Reserved for future use.

10.4.13 FunctionPresentState

Class............................ LAN I/O Registers, CardBus Status Change
Base Address .... IoBaseAddress register value
Address Offset 0x20
Default Value 0x00000000
Width .... 32 bits
FunctionPresentState reflects the current state of each condition that can cause a status change event.
This register is disabled when the CardBus bit of AsicCtrl is low.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3..0	Reserved	N/A	Reserved for future use.
4	GWAKE	R	GWAKE. GWAKE reflects the current state of the wakeup events not represented by the PCMCIA WP (Write Protect), RDY (Ready), or BVD (Battery Voltage Detect) events. See the PmeStatus bit of the PowerMgmtCtrl register.
14..5	Reserved	N/A	Reserved for future use.
15	INTR	R	INTR. INTR is the logical OR of all the interrupt causing bits after having been filtered through the IntEnable register.
31..16	Reserved	N/A	Reserved for future use.

10.4.14 HashTable

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x66, 0x64, 0x62, 0x60
Default 0x0000000000000000
Width .... 64 bits (accessible as 4, 16 bit words)
The host system stores a 64-bit hash table in this register for selectively receiving multicast frames. Setting the ReceiveMulticastHash bit in ReceiveMode register enables the filtering mechanism.

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	HashTableWord0	R/W	The least significant word of the hash table, corresponding to address 0x60.
31..16	HashTableWord1	R/W	The second least significant word of the hash table, corresponding to address 0x62.
47..32	HashTableWord2	R/W	The second most significant word of the hash table, corresponding to address 0x64.
63..48	HashTableWord3	R/W	The most significant word of the hash table, corresponding to address 0x66.

The IP100 applies a cyclic-redundancy-check (the same CRC used to calculate the frame data FCS) to the destination address of all incoming multicast frames (with multicast bit set). The low-order 6 bits of the CRC result are used as an addressing index into the hash table. The MSB of HashTable[3] is the most significant bit, and the LSB of HashTable[0] is the least significant bit, addressed by the 6-bit index. If the HashTable bit addressed by the index is a logic 1, the frame is accepted by the IP100 and transferred to higher layers. If the addressed hash table bit is a logic 0, the frame is discarded.

10.4.15 IntEnable

Class............................ LAN I/O Registers, Interrupt
Base Address .... IoBaseAddress register value
Address Offset.............. 0x4c
Default 0x0000
Width .... 16 bits

Enables individual interrupts as specified in the IntStatus register. Setting a bit in IntEnable will allow the specific source to generate an interrupt on the PCI bus. IntEnable is cleared by a read of IntStatusAck.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	Reserved	N/A	Reserved for future use.
1	EnHostError	R/W	Enable Host Error Interrupt. Enables the HostError interrupt.
2	EnTxComplete	R/W	Enable Transmit Complete Interrupt. Enables the TxComplete interrupt.
3	EnMACControlFrame	R/W	Enable MAC Control Frame Interrupt. Enables the MACControlFrame interrupt.
4	EnRxComplete	R/W	Enable Receive Complete Interrupt. Enables the RxComplete interrupt.
5	Reserved	R/W	Reserved for future use.
6	EnIntRequested	R/W	Enable Interrupt Requested Interrupt. Enables the IntRequested interrupt.
7	EnUpdateStats	R/W	Enable Update Stats Interrupt. Enables the UpdateStats interrupt.
8	EnLinkEvent	R/W	Enable Link Event Interrupt. Enables the LinkEvent interrupt.
9	EnTxDMAComplete	R/W	Enable Transmit DMA Complete Interrupt. Enables the TxDMAComplete interrupt.
10	EnRxDMAComplete	R/W	Enable Receive DMA Complete Interrupt. Enables the RxD-MAComplete interrupt.
11	Reserved/ EnModemInt	N/A, R/W	Reserved for future use. Enable Modem Interrupt. Enables the Reserved/ModemInt interrupt.
15..12	Reserved	N/A	Reserved for future use.

10.4.16 IntStatus

Class…......................... LAN I/O Registers, Interrupt
Base Address …............ IoBaseAddress register value
Address Offset …........... 0x4e
Default ….…................... 0x0000
Width …......................... 16 bits

IntStatus indicates the source of interrupts and indications on the IP100. All bits except InterruptStatus are the interrupt causing sources for the IP100. Each interrupt source can be individually disabled using the IntEnable register. The host system may acknowledge most interrupts by writing a logic 1 into the corresponding IntStatus bit, which will cause the IP100 to clear the interrupt indication.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	InterruptStatus	R/W	Interrupt Status. InterruptStatus is a logic 1 when the IP100 is driving the bus interrupt signal (INTAN). InterruptStatus is a logical OR of all the interrupt causing sources after they have been filtered through the IntEnable register.

BIT	BIT NAME	R/W	BIT DESCRIPTION
1	HostError	R/W	Host Error Interrupt. HostError is a logic 1 when a catastrophic error related to the bus interface occurs. Catastrophic bus interface errors include PCI target abort and PCI master abort. A HostError interrupt requires a setting the GlobalReset and DMA bits of the AsicCtrl register.
2	TxComplete	R/W	Transmit Complete. TxComplete is a logic 1 when a frame whose TxIndicate bit in the TFD's TxFrameControl field is a logic 1, has been successfully transmitted or for any frame that experiences a transmission error. A TxComplete interrupt requires writing to TxStatus register to advance the status queue.
3	MACControlFrame	R/W	MAC Control Frame Received Interrupt. MACControlFrame is a logic 1 when a MAC Control frame has been received by the IP100.
4	RxComplete	R/W	Receive Complete Interrupt. RxComplete is a logic 1 when one or more entire frames have been received into the receive FIFO. RxComplete is automatically acknowledged by the receive DMA logic as it transfers frames.
5	Reserved	R/W	Reserved for future use.
6	IntRequested	R/W	Interrupt Requested Interrupt. IntRequested is a logic 1 after the host system requests an interrupt by setting InterruptRequest bit of the AsicCtrl register or via the expiration of the Countdown register.
7	UpdateStats	R/W	Update Statistics Interrupt. UpdateStats is a logic 1 to indicate that one or more of the statistics registers is nearing an overflow condition (typically half of its maximum value). The host system should respond to an UpdateStats interrupt by reading all of the statistic registers, thereby acknowledging and clearing UpdateStats bit.
8	LinkEvent	R/W	Link Event Interrupt. LinkEvent is a logic 1 to indicate a change in the state of the Ethernet link.
9	TxDMAComplete	R/W	Transmit DMA Complete Interrupt. TxDMAComplete is a logic 1 to indicate that a transmit DMA operation has completed, and the frame's corresponding TFD had the TxDMAComplete bit in the TxFrameControl field set to a logic 1.
10	RxDMAComplete	R/W	Receive DMA Complete Interrupt. RxDMAComplete is a logic 1 to indicate that a frame receive DMA operation has completed.
11	Reserved/ ModemInt	N/A R/W	Reserved for future use. Modem Interrupt. Reserved/ModemInt is a logic 1 to indicate that the MINT signal has been asserted.
15..12	Reserved	N/A	Reserved for future use.

10.4.17 IntStatusAck

Class............................ LAN I/O Registers, Interrupt
Base Address .... IoBaseAddress register value
Address Offset 0x4a
Default 0x0000
Width .... 16 bits

IntStatusAck is another version of the IntStatus register, having the same bit definition as IntStatus, but providing additional functionality to reduce the number of I/O operations required to perform common tasks related to interrupt handling. In addition to returning the IntStatus value, when read IntStatusAck also acknowledges the TxDMAComplete, RxDMAComplete, IntRequested, MACControlFrame, and LinkEvent interrupt bits within the IntStatus register (if they are set), and clears the IntEnable register (preventing subsequent events from generating an interrupt).

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	InterruptStatus	R	See InterruptStatus in the IntStatus register.
1	HostError	R	See HostError in the IntStatus register.
2	TxComplete	R	See TxComplete in the IntStatus register.
3	MACControlFrame	R	See MACControlFrame in the IntStatus register.
4	RxComplete	R	See RxComplete in the IntStatus register.
5	Reserved	R	Reserved for future use.
6	IntRequested	R	See IntRequested in the IntStatus register.
7	UpdateStats	R	See UpdateStats in the IntStatus register.
8	LinkEvent	R	See LinkEvent in the IntStatus register.
9	TxDMAComplete	R	See TxDMAComplete in the IntStatus register.
10	RxDMAComplete	R	See RxDMAComplete in the IntStatus register.
11	Reserved/ ModemInt	N/A R/W	Reserved for future use. See Reserved/ModemInt in the IntStatus register.
15..12	Reserved	N/A	Reserved for future use.

10.4.18 MACCtrl0

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x50
Default 0x0000
Width .... 16 bits

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>1..0</td><td>IFSSelect</td><td>R/W</td><td>Inter-Frame Spacing Select. IFSSelect indicates the minimum number of bit times between the end of one Ethernet frame, and the beginning of another when the IP100 is deferring after a collision with the IP100 the last device to successfully acquire the network (in half duplex mode). By selecting a large value for IFSSelect, the IP100 will become less “aggressive” on the network and may defer more often (preventing the IP100 from “capturing” the network). The performance of the IP100 may decrease as the IFSSelect value is increased from the standard value.
<table>
<tr><td>BIT1</td><td>BIT0</td><td>INTER-FRAME SPACING IN BIT TIMES</td></tr>
<tr><td>0</td><td>0</td><td>96 (802.3 standard value, and default)</td></tr>
<tr><td>0</td><td>1</td><td>128</td></tr>
<tr><td>1</td><td>0</td><td>224</td></tr>
<tr><td>1</td><td>1</td><td>544</td></tr>
</table></td></tr>
<tr><td>4..2</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
</table>

BIT	BIT NAME	R/W	BIT DESCRIPTION
5	FullDuplexEnable	R/W	Full Duplex Enable. Setting FullDuplexEnable to a logic 1 configures the IP100 to function in a full duplex manner. When operating in full duplex, the IP100 disables transmitter deference to receive traffic, allowing simultaneous receive and transmit traffic. Operation in full duplex has the side-effect of disabling CarrierSenseErrors statistics collection, since full duplex operation requires carrier sense to be masked to the transmitter. TxReset and RxReset bits in AsicCtrl must be set for changes ofFullDuplexEnable to take effect.
6	RcvLargeFrames	R/W	Receive Large Frames. RcvLargeFrames determines the frame size at which the RxOversizedFrame bit of the RxFrameStatus field is set for receive frames. When RcvLargeFrames is a logic 0, minimum OversizedFrame size is 1514 bytes. When RcvLargeFrames is set, minimum OversizedFrame size is 4491 bytes. (This value was the maximum FDDI frame size of 4500 bytes, subtracting bytes for fields that have no Ethernet equivalent.) The frame size at which an OversizedFrame error will be flagged includes the destination and source addresses, the type/length field, and the FCS field.
7	Reserved	N/A	Reserved for future use.
8	FlowControlEnable	R/W	Flow Control Enable. If FlowControlEnable is a logic 0, the IP100 treats all incoming frames as data frames. If FlowControlEnable is a logic 1, flow control is enabled and the IP100 will act upon incoming flow control PAUSE frames. If FlowControlEnable is a logic 1, FullDuplexEnable should also be set to a logic 1.
9	RcvFCS	R/W	Receive FCS. If RcvFCS is a logic 1, the IP100 will include the receive frame's FCS along with the frame data transferred to the host system. If RcvFCS is a logic 0, the IP100 will remove the FCS from the frame before transferring the frame to the host system. The state of RcvFCS does not affect the IP100's checking of the frame's FCS and its posting of FCS errors. RcvFCS should only be changed when the receiver is disabled (via the RxDisable bit of the MACCtrl1 register) and after resetting the receive FIFO (via the FIFO bit of the AsicCtrl register).
10	FIFOLoopback	R/W	FIFO Loopback. If FIFOLoopback is a logic 1, the IP100 will enter FIFO Loopback Mode and force data to loopback from the transmit FIFO directly into the receive FIFO. When using FIFO Loopback Mode, it is the host system's responsibility to ensure that proper interframe spacing is ensured. To accommodate proper interframe spacing, the host system must not load more than one transmit frame into the transmit FIFO at a time while in FIFO Loopback Mode. The TxReset and RxReset bits of the AsicCtrl register must be set after changing the value of FIFOLoopback.
11	MACLoopback	R/W	MAC Loopback. If MACLoopback is a logic 1, the IP100 will enter MAC Loopback Mode and force data to loopback from the MAC transmit interface to the MAC receive interface. The TxReset and RxReset bits in AsicCtrl register must be set after changing the value of MACLoopback.
15..12	Reserved	N/A	Reserved for future use.

Note: External loopback is controlled by the PHY. To utilize external loopback, the host system must enable a loopback mode within the PHY using the MII Management Interface. For the true “on-the-wire” loopback mode, use a loopback plug (connector), clear the FIFOLoopback, and MACLoopback and any PHY loopback bits to zero, set the FullDuplexEnable bit to a logic 1, and enable the full duplex mode within the PHY.

10.4.19 MACCtrl1

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x52
Default 0x0000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	CollisionDetect	R	Collision Detect. CollisionDetect provides a real-time indication of the state of the COL signal within the IP100.
1	CarrierSense	R	Carrier Sense. CarrierSense provides a real-time indication of the state of the CRS signal within IP100.
2	TxInProg	R	Transmit In Progress. TxInProg provides a real-time indication that a frame is being transmitted. If TxInProg is a logic 1, a frame transmission is in progress. TxInProg is used by the host system during under run recovery to delay before setting the issuing a TxReset bit in the AsicCtrl register.
3	TxError	R	Transmit Error. If a transmit under run occurs (indicated via the TxUnderrun bit of the TxStatus register), TxError is a logic 1, indicating that the transmitter needs to be reset via the TxReset bit in the AsicCtrl register.
4	Reserved	N/A	Reserved for future use.
5	StatisticsEnable	W	Statistics Enable. Writing a logic 1 to StatisticsEnable will enable the IP100’s statistic registers. The state (enabled/disabled) of the IP100’s statistic registers is shown via StatisticsEnabled.
6	StatisticsDisable	W	Statistics Disable. Writing a logic 1 to StatisticsDisable will disable the IP100’s statistic registers. The state (enabled/disabled) of the IP100’s statistic registers is shown via StatisticsEnabled.
7	StatisticsEnabled	R	Statistics Enabled. If StatisticsEnabled is a logic 1, the IP100’s statistic registers are enabled.
8	TxEnable	W	Transmit Enable. Writing a logic 1 to TxEnable will enable the IP100 to transmit frames. The state (enabled/disabled) of the IP100’s transmitter is shown via TxEnabled.
9	TxDisable	W	Transmit Disable. Writing a logic 1 to TxDisable will disable the IP100 from transmitting frames. The state (enabled/disabled) of the IP100’s transmitter is shown via TxEnabled.
10	TxEnabled	R	Transmit Enabled. If TxEnabled is a logic 1, the IP100’s transmitter is enabled.
11	RxEnable	W	Receive Enable. Writing a logic 1 to RxEnable will enable the IP100 to receive frames. The state (enabled/disabled) of the IP100’s receiver is shown via RxEnabled.

BIT	BIT NAME	R/W	BIT DESCRIPTION
12	RxDisable	W	Receive Disable. Writing a logic 1 to RxDisable will disable the IP100 from receiving frames. The state (enabled/disabled) of the IP100's receiver is shown via RxEnabled.
13	RxEnabled	R	Receive Enabled. If RxEnabled is a logic 1, the IP100's receiver is enabled.
14	Paused	R	Paused. If Paused is a logic 1, the IP100 has received a PAUSE MAC Control Frame and the IP100 has halted the transmitter for the duration indicated in the PAUSE Frame's pause_time field. Paused will become a logic 0 when the IP100's transmitter ends is pause operation.
15	Reserved	N/A	Reserved for future use.

10.4.20 MaxFrameSize

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset.............. 0x5a
Default .... 0x1514 or 0x4491 based on the RcvLargeFrames bit in the MACCtrl0 register
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	MaxFrameSize	R/W	Maximum Frame Size. Received frames with sizes equal to or larger than MaxFrameSize will be flagged as oversize via the RxOversizedFrame bit in RxDMAStatus field of the frame's RFD.

10.4.21 PhyCtrl

Class............................ LAN I/O Registers, External Interface Control
Base Address .... IoBaseAddress register value
Address Offset 0x5e
Default 0x00
Width .... 8 bits

PhyCtrl contains control bits for the internal MII Management Interface. The MII Management Interface is used to access registers in the IP100 PHY. The host system accesses the MII Management Interface by writing and reading bit patterns to the PhyCtrl register.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	MgmtClk	R/W	Management Clock. MgmtClk drives the management clock (MDC) signal to the PHY.
1	MgmtData	R/W	Management Data bit. MgmtData is directly connected to the MDIO signal connected to the PHY. Data can be written to or read from the PHY by writing or read MgmtData based on the value of MgmtDir.
2	MgmtDir	R/W	Management Data Direction. If the MgmtDir is a logic 1, the value written to MgmtData is driven onto the MDIO signal connected to the PHY. When MgmtDir is a logic 0, data driven onto MDIO by the PHY can be read from MgmtData.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3	Reserved	N/A	Reserved for future use.
4	PhyDuplexPolarity	R/W	PHY Duplex Polarity. If PhyDuplexPolarity is a logic 0, the duplex status signal between the PHY and the MAC is active low.
5	PhyDuplexStatus	R	PHY Duplex Status. If PhyDuplexStatus is a logic 1, the PHY is operating in full duplex mode. If PhyDuplexStatus is a logic 0, the PHY is operating in half duplex mode.
6	PhySpeedStatus	R	PHY Speed Status. PhySpeedStatus provides a real-time indication of the IP100's PHY speed. If PhySpeedStatus is a logic 1, the IP100's PHY is operating at 100Mbps operation. If PhySpeedStatus is a logic 0, the IP100's PHY is operating at 10Mbps operation.
7	PhyLinkStatus	R	PHY Link Status. PhyLinkStatus provides a real-time indication of the IP100's PHY link state. If PhyLinkStatus is a logic 1, the IP100's PHY link is up. If PhyLinkStatus is a logic 0, the IP100's PHY link is down.

10.4.22 ReceiveMode

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x5c
Default 0x00
Width .... 8 bits
ReceiveMode sets the receive filter of the IP100.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	ReceiveUnicast	R/W	Receive Unicast. If ReceiveUnicast is a logic 1, the IP100 receives unicast frames (frames where the DA field matches the 48-bit value in the StationAddress register.
1	ReceiveMulticast	R/W	Receive Multicast. If ReceiveMulticast is a logic 1, the IP100 receives all multicast frames, including broadcast frames.
2	ReceiveBroadcast	R/W	Receive Broadcast. If ReceiveBroadcast is a logic 1, the IP100 receives all broadcast frames.
3	ReceiveAllFrames	R/W	Receive All Frames. If ReceiveAllFrames is a logic 1, the IP100 receives all frames.
4	ReceiveMulticast-Hash	R/W	Receive Multicast Hash. If ReceiveMulticastHash is a logic 1, the IP100 receives all which pass the hash filtering mechanism defined in the HashTable register..
5	ReceiveIPMulticast	R/W	Receive IP Multicast. If ReceiveIPMulticast is a logic 1, the IP100 receives all multicast IP datagrams, which are mapped into Ethernet multicast frames with destination address of 01:00:5e:xx:xx:xx as defined in RFC 1112 and RFC 1700. The first 3 bytes require exact match, and the last 3 bytes are ignored.
7..6	Reserved	N/A	Reserved for future use.

10.4.23 RxDMABurstThresh

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x14
Default 0x08
Width .... 8 bits

RxDMABurstThresh register sets the threshold for receive DMA bus master requests by the IP100 based upon the number of used bytes in the receive FIFO, in units of 32 bytes. When the used space exceeds the threshold, the IP100 may make a receive DMA request on the PCI bus. However, if the used space exceeds the RxDMAFrameLen field in the current RFD, the IP100 will make receive DMA bus request regardless of whether the used space exceeds the RxDMABurstThresh or not. RxDMABurstThresh may be overridden by the urgent request mechanism. See the PCI Bus Master Operation section for information about the relationship between RxDMABurstThresh and RxDMAUrgentThresh. Any value less than 0x08 is invalid and is interpreted as 0x08.

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	RxDMABurstThresh	R/W	Receive DMA Burst Threshold. The RxDMABurstThresh is the of 32 byte words which must be present in the receive FIFO prior to the assertion of a receive DMA bus master request.

10.4.24 RxDMAListPtr

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x10
Default 0x00000000
Width .... 32 bits

RxDMAListPtr is the physical address of the current receive DMA Frame Descriptor in the Receive DMA List. A value of 0x00000000 for RxDMAListPtr indicates that no more RFDs are available to accept receive frames. RxDMAListPtr only points to addresses on 8-byte boundaries, so RFDs must be aligned on 8-byte physical address boundaries. RxDMAListPtr may be written directly by the host system to point the IP100 to the head of a newly created Receive DMA List. RxDMAListPtr is also updated by the IP100 as it processes RFDs in the Receive DMA List. As the IP100 finishes processing a RFD, it loads RxDMAListPtr with the value from the RxDMANextPtr field of the current RFD in order to move on to the next RFD in the Receive DMA List. If the IP100 loads a value of 0x00000000 from the current RFD, the receive DMA logic enters the idle state, waiting for a non-zero value to be written to RxDMAListPtr. To avoid access conflicts between the IP100 and the host system, the host system must set the RxDMAHalt bit in the DMACtrl register before writing to RxDMAListPtr.

BIT	BIT NAME	R/W	BIT DESCRIPTION
31..0	RxDMAListPtr	R/W	Receive DMA List Pointer. RxDMAListPtr is the physical address, on a 8-byte boundary, of the current RFD in the Receive DMA List.

10.4.25 RxDMAPollPeriod

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x16
Default 0x00
Width .... 8 bits

RxDMAPollPeriod determines the rate at which the current RFD's RxDMAComplete bit of the RxFrameStatus field in the RFD is polled for a logic 0. Polling is disabled when RxDMAPollPeriod is 0x00. RxDMAPollPeriod is specified in increments of 320 ns. The maximum value is 127 (or 40.64 us).

BIT	BIT NAME	R/W	BIT DESCRIPTION
6..0	RxDMAPollPeriod	R/W	Receive DMA Poll Period. RxDMAPollPeriod is the number of 320ns intervals between polls of the RxDMAComplete bit in the RxFrameStatus field of the current RFD.
7	Reserved	N/A	Reserved for future use.

10.4.26 RxDMAStatus

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x0c
Default 0x00000000
Width .... 32 bits

RxDMAStatus shows the status of various operations in the receive DMA logic. Host systems should read RxDMAStatus only if the RxDMAHalted bit in the DMACtrl register is a logic 1. Otherwise the IP100 may change RFDs between accesses to RxDMAStatus. Many bits of RxDMAStatus are identical to corresponding bits of the RxFrameStatus field.\

BIT	BIT NAME	R/W	BIT DESCRIPTION
12..0	RxDMAFrameLen	R	Receive DMA Frame Length.RxDMAFrameLen is similar to the RxDMAFrameLen bit of the RxFrameStatus field, except that RxDMAFrameLen provides a real-time indication of the number of bytes transferred during a receive DMA operation.
13	Reserved	N/A	Reserved for future use.
14	RxFrameError	R	See the RxFrameError bit of the RxFrameStatus field.
15	RxDMAComplete	R	See the RxDMAComplete bit of the RxFrameStatus field.
16	RxFIFOOverrun	R	See the RxFIFOOverrun bit of the RxFrameStatus field.
17	RxRuntFrame	R	See the RxRuntFrame bit of the RxFrameStatus field.
18	RxAlignmentError	R	See the RxAlignmentError bit of the RxFrameStatus field.
19	RxFCSError	R	See the RxFCSError bit of the RxFrameStatus field.
20	RxOversizedFrame	R	See the RxOversizedFrame bit of the RxFrameStatus field.
22..21	Reserved	N/A	Reserved for future use.
23	DribbleBits	R	See the DribbleBits bit of the RxFrameStatus field.
24	RxDMAOverflow	R	See the RxDMAOverflow bit of the RxFrameStatus field.
31..25	Reserved	N/A	Reserved for future use.

10.4.27 RxDMAUrgentThresh

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x15
Default 0x04
Width .... 8 bits
RxDMAUrgentThresh sets a threshold at which the receive DMA logic will make a urgent bus master request. A urgent receive DMA request will have priority over all other requests on the IP100. The urgent bus request is made when the free space in the receive FIFO falls below the value in RxDMAUrgentThresh. A receive DMA urgent request is not subject to the RxDMABurstThresh constraint. When the receive FIFO is close to overrun, burst efficiency is sacrificed in favor of requesting the bus as quickly as possible. The value in RxDMAUrgentThresh represents free space in the receive FIFO in terms of 32-byte portions.

BIT	BIT NAME	R/W	BIT DESCRIPTION
4..0	RxDMAUrgentThresh	R/W	Receive DMA Urgent Threshold. RxDMAUrgentThresh is the minimum number of 32-byte words which must be available in the receive FIFO to avoid a receive DMA urgent request.
7..5	Reserved	N/A	Reserved for future use.

10.4.28 StationAddress

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x54, 0x56, 0x58
Default 0x000000000000
Width .... 48 bits (accessible as 3, 16 bit words)
StationAddress is used to define the individual destination address that the IP100 will respond to when receiving frames. Network addresses are generally specified in the form of 01:23:45:67:89:ab, where the bytes are received left to right, and the bits within each byte are received right to left (lsb to msb). The actual transmitted and received bits are in the order of 10000000 11000100 10100010 11100110 10010001 11010101.

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	StationAddressWord0	R/W	The least significant word of the station address, corresponding to address 0x54.
31..16	StationAddressWord1	R/W	The second least significant word of the station address, corresponding to address 0x56.
47..32	StationAddressWord2	R/W	The most significant word of the station address, corresponding to address 0x58.

The address comparison logic will compare the first 16 received destination address bits against StationAddressWord0, the second 16 received destination address bits against StationAddressWord1, and the third 16 received destination address bits against StationAddressWord2. The value set in StationAddress is not inserted into the source address field of frames transmitted by the IP100. The source address field for every frame must be specified by the host system as part of the frame data contents.

10.4.29 TxDMABurstThresh

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x08
Default 0x08
Width .... 8 bits

TxDMABurstThresh determines the threshold for when the IP100 makes transmit DMA bus master requests, based upon the available space in the transmit FIFO. TxDMABurstThresh represents free space in the transmit FIFO in multiples of 32 bytes. When the free space exceeds the threshold, the IP100 may make a transmit DMA request. However, if the free space exceeds the current FragLen subfield of the TxFrameControl field within the current TFD, the IP100 will make transmit DMA bus request regardless of whether the free space exceeds the TxDMABurstThresh or not. TxDMABurstThresh may be overridden by the TxDMAUrgentThresh mechanism. See the PCI Bus Master Operation section for information about the relationship between TxDMABurstThresh and TxDMAUrgentThresh. Any value less than 0x08 is invalid and is interpreted as 0x08.

BIT	BIT NAME	R/W	BIT DESCRIPTION
4..0	TxDMABurstThresh	R/W	Transmit DMA Burst Threshold. The number of 32-byte words which must be available in the transmit FIFO prior to assertion of a transmit DMA Burst Request.
7..5	Reserved	N/A	Reserved for future use.

10.4.30 TxDMAListPtr

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x04
Default 0x00000000
Width .... 32 bits

TxDMAListPtr holds the physical address of the current transmit DMA Frame Descriptor in the transmit DMA list. A value of zero in TxDMAListPtr is interpreted by the IP100 to mean that no more frames remain to be transferred by transmit DMA. TxDMAListPtr can only point to addresses on 8-byte boundaries, so TFD's must be aligned on 8-byte boundaries. TxDMAListPtr may be written directly by the host system to point the IP100 at the head of a newly created transmit DMA list. Writes to TxDMAListPtr are ignored while the current value in TxDMAListPtr is non-zero. To avoid access conflicts between the IP100 and the host system, the host system must set the TxDMAHalt bit of the DMACtrl register to a logic 1 before writing to TxDMAListPtr (unless the host system has specific knowledge that TxDMAListPtr contains zero).

BIT	BIT NAME	R/W	BIT DESCRIPTION
31..0	TxDMAListPtr	R/W	Transmit DMA List Pointer. TxDMAListPtr holds the physical address, on a 8-byte boundary, of the current TFD in the transmit DMA list. The host may examine the TxDMAListPtr to determine which frame(s) have been transferred by transmit DMA. Those frames in the transmit DMA list before the current TxDMAListPtr have already been transferred by transmit DMA. If the TxDMAListPtr is zero, then all the frames have been transmitted.

10.4.31 TxDMAPollPeriod

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x0a
Default 0x00
Width .... 8 bits

TxDMAPollPeriod determines the interval at which the current TFD is polled. If the current TFDs, TxDMANextPtr field is 0x00000000, the TxDMANextPtr field is polled to determine when a new TFD is ready to be processed. Polling is disabled when TxDMAPollPeriod is 0x00. TxDMAPollPeriod represents a multiple of 320 ns time intervals. The maximum value is 127 (or 40.64 us).

BIT	BIT NAME	R/W	BIT DESCRIPTION
6..0	TxDMAPollPeriod	R/W	Transmit DMA Poll Period. The number of 320ns intervals between polls of the current TFD's TxDMANextPtr field.
7	Reserved	N/A	Reserved for future use.

10.4.32 TxDMAUrgentThresh

Class............................ LAN I/O Registers, DMA
Base Address .... IoBaseAddress register value
Address Offset 0x09
Default 0x04
Width .... 8 bits

When the number of used bytes in the transmit FIFO falls below the value in the TxDMAUrgentThresh, the transmit DMA logic will make an urgent bus master request. An urgent transmit DMA request will have priority over the receive DMA, unless it is also making an urgent request. A transmit DMA urgent request is not subject to the TxDMABurstThresh constraint. The relaxation of the TxDMABurstThresh constraint for this condition is because the transmit FIFO is close to under run, and burst efficiency is sacrificed to avoid FIFO under run. TxDMAUrgentThresh represents data in the transmit FIFO in multiples of 32 bytes.

BIT	BIT NAME	R/W	BIT DESCRIPTION
5..0	TxDMAUrgent-Thresh	R/W	Transmit DMA Urgent Threshold. The minimum number of 32-byte words which must be occupied in the transmit FIFO to avoid assertion of a transmit DMA Urgent Request.
7..6	Reserved	N/A	Reserved for future use.

10.4.33 TxReleaseThresh

Class............................ LAN I/O Registers, FIFO Control
Base Address .... IoBaseAddress register value
Address Offset 0x5d
Default 0x08
Width .... 8 bits

TxReleaseThresh determines how much data of a frame must be transmitted before the transmit FIFO space can be released for use by another frame. Once the number of bytes equal to the value in TxReleaseThresh have been transmitted, that number of bytes are discarded from the transmit FIFO. Thereafter, bytes are discarded as they are transmitted to the network. A value of 0xff in TxReleaseThresh disables the release mechanism and transmit FIFO frame space is not released until the entire frame is transmitted. The TxReleaseError bit in the TxStatus register indicates when a frame experiences a collision after its release threshold has been crossed, preventing MAC from retry. When a release error occurs, the transmitter is disabled, and the frame's ID or sequence number is visible in TxStatus.

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	TxReleaseThresh	R/W	Transmit Release Threshold. The number of 16 byte words which must be transmitted before the space in the transmit FIFO occupied by the transmitted data can be released. To avoid excessive release errors due to in-window collisions, TxReleaseThresh should be greater than or equal to 0x04.

10.4.34 TxStatus

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x46
Default 0x0000
Width .... 16 bits

TxStatus returns the status of frame transmission or transmission attempts. TxStatus actually implements a queue of up to 31 transmit status bytes. A write of an arbitrary value to TxStatus will advance the queue to the next transmit status byte.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	Reserved	N/A	Reserved for future use.
1	TxReleaseError	R	Transmit Release Error. If TxReleaseError is a logic 1, a transmit release error occurred, meaning that the frame transmission experienced a collision after the front of the frame had already been released to the transmit FIFO free space. See TxReleaseThresh register.
2	TxStatusOverflow	R	Transmit Status Overflow. If TxReleaseError is a logic 1, the TxStatus stack is full and as a result the transmitter has been disabled. Writing an arbitrary value to TxStatus clears TxReleaseError but the transmitter must be re-enabled via the TxEnable bit of the MACCtrl1 register before transmissions may resume.
3	MaxCollisions	R	Maximum Collisions. If MaxCollisions is a logic 1, a frame was not successfully transmitted due to encountering 16 collisions. The TxEnable bit of the MACCtrl1 register is used to recover from this condition. The frame is discarded from the transmit FIFO.
4	TxUnderrun	R	Transmit Underrun. If TxUnderrun is a logic 1 the frame experienced an under run during the transmit process because the host system was unable to supply the frame data fast enough to keep up with the network data rate. An under run will halt the transmitter and the transmit FIFO. The TxReset bit of the AsicCtrl register is used to recover from an under run condition.
5	Reserved	N/A	Reserved for future use.
6	TxIndicateReqd	R	Transmit Indicate Requested. If TxIndicateReqd is a logic 1, the TxIndicate bit of the TxFrameControl field for the corresponding TFD was set.
7	TxComplete	R	Transmit Complete. If TxComplete is a logic 0, then the TxReleaseError, TxStatusOverflow, MaxCollisions, TxUnderrun, and TxIndicateReqd bits are undefined. If the host chooses to poll TxStatus while waiting for a frame transmission to complete, then TxComplete is used to determine that a frame transmission attempt has either experienced an error, or has completed successfully with the TxIndicate bit set in the TxFrameControl field of the corresponding TFD.
15..8	TxFrameId	R/W	Transmit Frame Identification. TxFrameId contains the value from the FrameId subfield within the TxFrameControl field of TFD corresponding to the currently transmitting or most recently transmitted frame. Host systems can use TxFrameId during transmit error recovery by scanning through the TFD's in the transmit DMA list, searching for a match between the TxFrameId value and a TxFrameControl's, FrameId value in the TFD.

10.4.35 WakeEvent

Class............................ LAN I/O Registers, Control and Status
Base Address .... IoBaseAddress register value
Address Offset 0x45
Default 0x00
Width .... 8 bits

WakeEvent contains enable bits to control which types of events can generate a wake event to the host system. WakeEvent also contains status bits indicating the specific wake events which have occurred.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	WakePktEnable	R/W	Wake Packet Enable. If WakePktEnable is a logic 1, the IP100 may generate wake events via a PCI interrupt due to Wake Packet reception. The PmeEn bit in the PowerMgmtCtrl register must be set in order for WakePktEnable to be recognized. WakePktEnable has no effect in power mode D0.
1	MagicPktEnable	R/W	Magic Packet Enable. If MagicPktEnable is a logic 1, the IP100 may generate wake events via a PCI interrupt due to Magic Packet reception. The PmeEn bit in the PowerMgmtCtrl register must be set in order for MagicPktEnable to be recognized. MagicPktEnable has no effect in power mode D0.
2	LinkEventEnable	R/W	Link Event Enable. If LinkEventEnable is a logic 1, the IP100 may generate wake events via a PCI interrupt due to a change in link status (cable connect or disconnect). The PmeEn bit in the PowerMgmtCtrl register must be set in order for LinkEventEnable to be recognized. LinkEventEnable has no effect in power mode D0.
3	WakePolarity	R/W	Wake Polarity. If WakePolarity is a logic 1, the WAKE signal will be asserted HIGH. If WakePolarity is a logic 0, the WAKE signal will be asserted LOW.
4	WakePktEvent	R	Wake Packet Event. If WakePktEvent is a logic 1, a wake packet (which meets the reception criteria set by the host system) has been received. WakePktEnable must be a logic 1 for WakePktEvent to operate. WakePktEvent is cleared following a read of the WakeEvent register.
5	MagicPktEvent	R	Magic Packet Event. If MagicPktEvent is a logic 1, a Magic Packet packet has been received. MagicPktEnable must be a logic 1 for MagicPktEvent to operate. MagicPktEvent is cleared following a read of the WakeEvent register.
6	LinkEvent	R	Link Event. If LinkEvent is a logic 1, a link status event has occurred. LinkEventEnable must be a logic 1 for LinkEvent to operate. LinkEvent is cleared following a read of the WakeEvent register.
7	WakeOnLanEnable	R/W	Wake On LAN Enable. If WakeOnLanEnable is a logic 1, the IP100 is in WakeOnLan mode regardless of the power management register settings in the configuration space. WakeOnLanEnable is loaded from the WakeOnLanEnable bit of AsicCtrl field within the EEPROM.

Preliminary, Specification subject to change without notice.

10.5 Statistic Registers

The Statistic registers implement several counters defined in the IEEE 802.3 standard. Note reading a statistic register will also clear that register. The statistics gathering must be enabled by setting the StatisticsEnable bit in MACCtrl1 for the statistics registers to count events.

10.5.1 BroadcastFramesReceivedOk

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x7d
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	BroadcastFrames-ReceivedOk	R/W	Broadcast Frames Received OK is the count of the number of frames that are successfully received with destination address equal to the broadcast address (0xFFFFFFFFFFFF). BroadcastFramesReceivedOk does not include frames received with frames too long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error (i.e.overrun). BroadcastFramesReceivedOk will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.22. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when BroadcastFramesReceivedOk reaches a value of 0xC0. BroadcastFramesReceivedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of BroadcastFramesReceivedOk also clears the register.

10.5.2 BroadcastFramesTransmittedOk

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x7c
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	Broadcast-FramesTransmitted-Ok	R/W	Broadcast Frames Transmitted is the count of the number of frames that are successfully transmitted to the broadcast address (0xFFFFFFFFFFFF). Frames transmitted to other multicast addresses are excluded from this statistic. Broad-castFramesTransmittedOk will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.19. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when BroadcastFramesTransmittedOk reaches a value of 0xC0. BroadcastFramesTransmittedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of BroadcastFramesTransmittedOk also clears the register.

Preliminary, Specification subject to change without notice.

10.5.3 CarrierSenseErrors

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x74
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
3..0	CarrierSenseErrors	R/W	Carrier Sense Errors counts the number of times that the carrier sense signal (CRS) was de-asserted (a logic 0) during the transmission of a frame without collision. The carrier sense signal is not monitored for the purpose of this statistic until after the preamble and start-of-frame delimiter fields of the Ethernet frame have been transmitted. CarrierSenseErrors will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.13. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when CarrierSenseErrors reaches a value of 0xC0. CarrierSenseErrors is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of CarrierSenseErrors also clears the register.
7..4	Reserved	R/W	Reserved for future use.

10.5.4 FramesAbortedDueToXSColls

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x7b
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	FramesAborted-DueToXSColls	R/W	Frames Aborted Due to Excess Collisions counts the number of frames which are not transmitted successfully due to excessive collisions. FramesAbortedDueToXSColls will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.11. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when FramesAbortedDueToXSColls reaches a value of 0xC0. FramesAbortedDueToXSColls is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of FramesAbortedDueToXSColls also clears the register.

10.5.5 FramesLostRxErrors

Class............................. LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x79
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	FramesLostRxErrors	R/W	Frames Lost Due to Receive Errors is a count of the number of frames that should have been received (the destination address matched the filter criteria) but experienced a receive FIFO overrun error (the receive FIFO does not have enough free space to store the received data). FramesLostRxErrors only includes overruns that become apparent to the host system, and does not include frames that are completely ignored due to a completely full receive FIFO at the beginning of frame reception. FramesLostRxErrors will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.15. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when FramesLostRxErrors reaches a value of 0xC0. FramesLostRxErrors is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of FramesLostRxErrors also clears the register.

10.5.6 FramesReceivedOk

Class............................. LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x72
Default 0x0000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	FramesReceivedOk	R/W	Frames Received OK is the count of the number of frames that are successfully received. FramesReceivedOk does not include frames received with frames too long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error (i.e. overrun). FramesReceivedOk will wrap around to zero after reaching 0xFFFF. See IEEE 802.3 Clause 30.3.1.1.5. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when FramesReceivedOk reaches a value of 0xC0. FramesReceivedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of FramesReceivedOk also clears the register.

10.5.7 FramesTransmittedOk

Class............................. LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset.............. 0x70
Default 0x0000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	FramesTransmitted-OK	R/W	Frames Transmitted OK is a count of the number of frames that are successfully transmitted. FramesTransmittedOk will wrap around to zero after reaching 0xFFFF. See IEEE 802.3 Clause 30.3.1.1.2. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when FramesTransmittedOk reaches a value of 0xC0. FramesTransmittedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of FramesTransmittedOk also clears the register.

10.5.8 FramesWithDeferredXmission

Class............................. LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset.............. 0x78
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	FramesWith-DeferredXmission	R/W	Frames with Deferred Transmit is a count of the number of frames that must delay their first attempt of transmission because the medium was busy. Frames involved in any collisions are not counted by FramesWithDeferredXmission. FramesWithDeferredXmission wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.9. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when FramesWithDeferredXmission reaches a value of 0xC0. FramesWithDeferredXmission is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of FramesWithDeferredXmission also clears the register.

Preliminary, Specification subject to change without notice.

10.5.9 FramesWithExcessiveDeferal

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x7a
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	FramesWithExcessiveDeferal	R/W	Frames with Excessive Deferrals counts the number of frames that deferred for an excessive period of time (exceeding the defer limit). FramesWithExcessiveDeferal is only incremented once per LLC frame. FramesWithExcessiveDeferal will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.20. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when FramesWithExcessiveDeferal reaches a value of 0xC0. FramesWithExcessiveDeferal is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of FramesWithExcessiveDeferal also clears the register.

10.5.10 LateCollisions

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x75
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	LateCollisions	R/W	Late Collisions is a count of the number of times that a collision has been detected later than 1 slot time into the transmitted frame. LateCollisions will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.10. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when LateCollisions reaches a value of 0xC0. LateCollisions is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of LateCollisions also clears the register.

10.5.11 MulticastFramesReceivedOk

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x7f
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	MulticastFrames-ReceivedOk	R/W	Multicast Frames Received OK is the count of the number of frames that are successfully received to a group destination address other than the broadcast address (0xFFFFFFFFFFFF). MulticastFramesReceivedOk does not include frames received with frames too long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error (i.e. overrun). MulticastFramesReceivedOk will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.21. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when MulticastFramesReceivedOk reaches a value of 0xC0. MulticastFramesReceivedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of MulticastFramesReceivedOk also clears the register.

10.5.12 MulticastFramesTransmittedOk

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x7e
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	Multicast-FramesTransmitted-Ok	R/W	Multicast Frames Transmitted OK is a count of the number of frames that are successfully transmitted to a group destination address other than the broadcast address (0xFFFFFFFFFFFF). MulticastFramesTransmittedOk will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.18. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when MulticastFramesTransmittedOk reaches a value of 0xC0. MulticastFramesTransmittedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of MulticastFramesTransmittedOk also clears the register.

10.5.13 MultipleCollisionFrames

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x76
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	MultipleCollision-Frames	R/W	Multiple Collision Frames is a count of the number of frames that are involved in more than one collision and are subsequently transmitted successfully. MultipleCollisionFrames will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.4. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when MultipleCollisionFrames reaches a value of 0xC0. MultipleCollisionFrames is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of MultipleCollisionFrames also clears the register.

10.5.14 OctetsReceivedOk

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset 0x68
Default 0x00000000
Width .... 32 bits (accessible as 2, 16 bit words)

BIT	BIT NAME	R/W	BIT DESCRIPTION
19..0	OctetsReceivedOk	R/W	Octets Received OK is the count of the number of data and padding octets in frames that are successfully received. OctetsReceivedOk does not include frames received with frames too long, FCS, length or alignment errors, or frames lost due to internal MAC sublayer error (i.e. overrun). OctetsReceivedOk will wrap around to zero after reaching 0xFFFFFFFF. See IEEE 802.3 Clause 30.3.1.1.14. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when OctetsReceivedOk reaches a value of 0xC0. OctetsReceivedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of OctetsReceivedOk also clears the register.
31..20	Reserved	N/A	Reserved for future use.

10.5.15 OctetsTransmittedOk

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset.............. 0x6c
Default 0x00000000
Width .... 32 bits (accessible as 2, 16 bit words)

BIT	BIT NAME	R/W	BIT DESCRIPTION
19..0	OctetsTransmitted-Ok	R/W	Octets Transmitted OK is a count of data and padding octets of frames successfully transmitted. OctetsTransmittedOk will wrap around to zero after reaching 0xFFFFFFFF. See IEEE 802.3 Clause 30.3.1.1.8. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when OctetsTransmittedOk reaches a value of 0xC0. OctetsTransmittedOk is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of OctetsTransmittedOk also clears the register.
31..20	Reserved	N/A	Reserved for future use.

10.5.16 SingleCollisionFrames

Class............................ LAN I/O Registers, Statistics
Base Address .... IoBaseAddress register value
Address Offset.............. 0x77
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	Single Collision Frames	R/W	Single Collision Frames is a count of the number of frames that are involved in a single collision, and are subsequently transmitted successfully. SingleCollisionFrames will wrap around to zero after reaching 0xFF. See IEEE 802.3 Clause 30.3.1.1.3. An UpdateStats interrupt (UpdateStats bit within the IntStatus register) will occur when SingleCollisionFrames reaches a value of 0xC0. SingleCollisionFrames is enabled by writing a logic 1 to the StatisticsEnable bit in the MACCtrl1 register. A read of SingleCollisionFrames also clears the register.

10.6 LAN PCI Configuration Registers

PCI based systems use a slot-specific block of configuration registers to perform configuration of devices on the PCI bus. The configuration registers are accessed with PCI Configuration Cycles. The PCI bus supports two types of Configuration Cycles. Type 0 cycles are used to configure devices on the local PCI bus. Type 1 cycles are used to pass a configuration request to a PCI bus at a different hierarchical level. PCI Configuration Cycles are directed at one out of eight possible PCI logical functions within a single physical PCI device. A IP100 based PCI bus master device responds only to Type 0 Configuration Cycles, directed at function 0, or Type 1 cycles if an external modem is attached. Type 1 cycles (without a modem), and Type 0 cycles directed at functions other than 0, are ignored by the IP100.

Each PCI bus device is required to decode 256 bytes of configuration registers. Of these, the first 64 bytes are pre-defined by the PCI Specification. The remaining registers may be used as needed for PCI device-specific configuration registers. In PCI Configuration Cycles, the host system provides a slot-specific decode signal (IDSEL) which informs the PCI device that a configuration cycle is in progress. The PCI device responds by asserting DEVSELN, and decoding the specific configuration register from the address bus and the byte enable signals. See the PCI Expansion ROM specification for information on generating configuration cycles from driver software.

Figure 14 shows the PCI configuration registers implemented by IP100. All locations within the 256-byte configuration space that are not shown in the table, are not implemented and return zero when read.

BYTE 3	BYTE 2	BYTE 1	BYTE 0	ADDR OFFSET
Data		PowerMgmtCtrl		0x54
PowerMgmtCap		NextItemPtr	CapId	0x50
				0x4c
				0x48
				0x44
				0x40
MaxLat	MinGnt	InterruptPin	InterruptLine	0x3C
				0x38
			CapPtr	0x34
ExpRomBaseAddress				0x30
SubsystemId		SubsystemVendorId		0x2C
CISPointer				0x28
				0x24
				0x20
				0x1C
MemBaseAddress				0x18
IoBaseAddress				0x14
	HeaderType	LatencyTimer	CacheLineSize	0x0C
ClassCode			RevisionId	0x08
ConfigStatus		ConfigCommand		0x04
DeviceId		VendorId		0x00

FIGURE 14 : IP100 PCI Register Layout

10.6.1 CacheLineSize

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x0c
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	CacheLineSize	R/W	Cache Line Size. The system BIOS writes the system's cache line size into CacheLineSize. The host system uses CacheLineSize to optimize PCI bus master operation (choosing the best memory command, etc.). The value in CacheLineSize represents the number of double words in a cache. CacheLineSize values must be a power of two, from 0x04 to 0x80 (giving a range of 16 to 512 bytes). CacheLineSize values which are not a power of two, between 4 and 128 are interpreted as 0x00.

10.6.2 CapId

Class............................. LAN PCI Configuration Registers, Power Management
Base Address .... PCI device configuration header start
Address Offset 0x50
Default 0x01
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	CapId	R	Capabilities ID. CapId indicates the type of the capability data structure for the IP100. CapId is set to the value 0x01 to indicate a PCI Power Management structure.

10.6.3 CapPtr

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x34
Default 0x50
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	CapPtr	R	Capabilities Pointer. CapPtr indicates the beginning of a chain of registers which describe enhanced functions. CapPtr register returns 0x50, which is the address of the first in a series of power management registers.

10.6.4 CISPointer

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset.............. 0x28
Default Value 0x00000802
Width .... 32 bits

CISPointer identifies the location of the Card Information Structure (CIS). CISPointer contains the offset of CIS in memory space. CISPointer value is hardwired and can not be changed. Although the CIS is in memory, it is physically located in the EEPROM. Because the EEPROM access is so slow, the PCI target will issue a Retry for each new access.

BIT	BIT NAME	R/W	BIT DESCRIPTION
2..0	AddressSpace Indicator	R	Address Space Indicator. AddressSpaceIndicator specifies the base address within the space indicated. The AddressSpaceOffset is added to AddressSpaceIndicator to identify the start of the CIS. The AddressSpaceIndicator value is 0x2, indicating the IP100 only supports CIS access in the memory pointed to by the PCI Base Address Register 1.
27..3	AddressSpaceOffset	R	Address Space Offset. AddressSpaceOffset is the offset from the base address specified by the PCI Base Address Register 1. The IP100 supports CIS in memory space.
31..28	ROMImageNumber	R	ROM Image Number. ROMImageNumber is 0x0 indicating the IP100 does not support CIS access in the expansion ROM.

10.6.5 ClassCode

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset.............. 0x09
Default 0x020000
Width .... 24 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
23..0	ClassCode	R	Class Code. ClassCode identifies the general function of the PCI device. A value of 0x020000 indicates an Ethernet network controller.

10.6.6 ConfigCommand

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x04
Default 0x0000
Width .... 16 bits
ConfigCommand provides control over the adapter's ability to generate and respond to PCI cycles. When a zero is written to ConfigCommand, the IP100 is logically disconnected from the PCI bus, except for configuration cycles.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	IoSpace	R/W	I/O Space. When IoSpace is a logic the IP100 can respond to I/O space accesses (if the IP100 is in the D0 power state).
1	MemorySpace	R/W	Memory Space. When MemorySpace, and the AddressDecodeEnable bit in the ExpRomBaseAddress register are both a logic 1, and if the IP100 is in the D0 power state, the IP100 is able to decode accesses to an Expansion ROM (if present).
2	BusMaster	R/W	Bus Master. When BusMaster is a logic 1 the IP100 is able to initiate bus master cycles (if the adapter is in the D0 power state). Note: If the IP100 is initialized to PCI-X mode, BusMaster is ignored when initiating Split Completions.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3	Reserved	N/A	Reserved for future use.
4	MWIEnable	R/W	Memory Write and Invalidate Enable. When MWIEnable is a logic 1 the IP100 is permitted to use the MWI command.
5	Reserved	N/A	Reserved for future use.
6	ParityErrorResponse	R/W	Parity Error Response. When ParityErrorResponse is a logic 1 the IP100 responds to parity errors as defined within the PCI specification. When ParityErrorResponse is a logic 0, the IP100 ignores parity errors.
7	Reserved	N/A	Reserved for future use.
8	SERREnable	R/W	System Error Enable. When SERREnable is a logic 1, the SERRN signal is allowed to transition as appropriate. When SERREnable is a logic 0, the SERRN signal is a continuous logic 0.
15..9	Reserved	N/A	Reserved for future use.

10.6.7 ConfigStatus

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset.............. 0x06
Default 0x0210
Width .... 16 bits

ConfigStatus is used to record status information for PCI bus events. Read/write bits inConfigStatus can only be reset, not set, by writing to this register. Bits are reset by writing a one to the corresponding bit.

BIT	BIT NAME	R/W	BIT DESCRIPTION
3..0	Reserved	N/A	Reserved for future use.
4	Capabilities	R	Capabilities. Capabilities is a logic 1 to indicate a set of extended capabilities registers exists for the IP100. The CapPtr register indicates the first address location of the extended capabilities register set.
6..5	Reserved	N/A	Reserved for future use.
7	FastBackToBack	R	Fast Back to Back. When FastBackToBack is a logic 1 the IP100 when operating as a Target, supports fast back-to-back transactions as defined by the criteria in the section 3.4.2 of the PCI specification.
8	DataParityReported	R/W	Master Data Parity Error. When DataParityReported is a logic 1, the IP100 when operating as a Master, has detected the PERRN signal asserted, and the ParityErrorResponse bit in the ConfigCommand register as a logic 1.
10..9	DevselTiming	R	Device Select Timing. DevselTiming is used to encode the slowest time with which the IP100 asserts the DEVSELN signal. A value of 0x1 for DevselTiming indicates support for “medium” speed DEVSELN assertion.
11	SignaledTargetAbort	R/W	Signaled Target Abort. The IP100 sets SignaledTargetAbort to a logic 1 when the IP100 terminates a bus transaction with target-abort.
12	ReceivedTarget-Abort	R/W	Received Target Abort. The IP100 sets ReceivedTargetAbort to a logic 1 when, operating as a bus master, a IP100 bus transaction is terminated with target-abort.

Preliminary, Specification subject to change without notice.

BIT	BIT NAME	R/W	BIT DESCRIPTION
13	ReceivedMaster-Abort	R/W	Received Master Abort. The IP100 sets ReceivedMasterAbort to a logic 1 when, operating as a bus master, a IP100 bus transaction is terminated with master-abort.
14	SignaledSystem-Error	R/W	Signaled System Error. When SignaledSystemError is a logic 1, the IP100 asserts the SERRN signal.
15	DetectedParity-Error	R/W	Detected Parity Error. When DetectedParityError is a logic 1 the IP100 has detected a parity error, regardless of whether parity error handling is enabled.

10.6.8 Data

Class............................ LAN PCI Configuration Registers, Power Management
Base Address .... PCI device configuration header start
Address Offset 0x57
Default Value 0x00
Width .... 8 bits

Data reports the power consumption of the IP100. The values of DataSelect and DataScale in the PowerMgmtCtrl register are used to interpret the value of Data.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>7..0</td><td>Data</td><td>R</td><td>Data. Data reports power consumption and dissipation of the IP100 at worst case conditions. To properly interpret the value read from Data, it must be scaled by the factor indicated in the DataScale field of the PowerMgmtCtrl register. The value of Data depends on the value of the DataSelect field of the PowerMgmtCtrl register.
Data is loaded from the Data field in the EEPROM during a AutoInit reset.
<table>
<tr><th>DATASELECT</th><th>DATA</th></tr>
<tr><td>0x0</td><td>1 * DataScale Watts
D0 Power Consumption</td></tr>
<tr><td>0x1</td><td>1 * DataScale Watts
D1 Power Consumption</td></tr>
<tr><td>0x2</td><td>1 * DataScale Watts
D2 Power Consumption</td></tr>
<tr><td>0x3</td><td>1 * DataScale Watts
D3 Power Consumption</td></tr>
<tr><td>0x4</td><td>1 * DataScale Watts
D4 Power Dissipated</td></tr>
<tr><td>0x5</td><td>1 * DataScale Watts
D5 Power Dissipated</td></tr>
<tr><td>0x6</td><td>1 * DataScale Watts
D6 Power Dissipated</td></tr>
<tr><td>0x7</td><td>1 * DataScale Watts
D7 Power Dissipated</td></tr>
<tr><td>0x8 through 0xF</td><td>0x00
Reserved.</td></tr>
</table>
</td></tr>
</table>

10.6.9 DeviceId

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x02
Default Value 0x0201
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	DeviceId	R	DeviceId contains the 16-bit device ID for the IP100.

10.6.10 ExpRomBaseAddress

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x30
Default 0x00000000
Width .... 32 bits
ExpRomBaseAddress allows the system to define the base address for the adapter's Expansion ROM. ExpRomBaseAddress is disabled (read only with a value of 0x00000000) when the IP100 is in Multi-Function Mode (see the Reserved/MultiFunction bit in the AsicCtrl register).

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	AddressDecode-Enable	R/W	Address Decode Enable. If AddressDecodeEnable is a logic 0, the IP100's Expansion ROM is disabled. If AddressDecodeEnable is a logic 1, and the MemorySpace bit in the ConfigCommand register is a logic 1, the IP100 will respond to accesses to the expansion ROM space.
14..1	Reserved	N/A	Reserved for future use.
31..15	RomBaseAddress	R/W	ROM Base Address. RomBaseAddress contains the expansion ROM base address, or the upper 16 bits (or 15 bits, depending on the state of the ExpRomSize bit in the AsicCtrl register) of the Expansion ROM address range. If the ExpRomSize bit in the AsicCtrl register is a logic 0, all 16 bits of RomBaseAddress are valid. If the ExpRomSize bit in the AsicCtrl register is a logic 1, bits 31 through 16 of RomBaseAddress are valid, with bit 15 ignored (set to a logic 0) during write operations.

10.6.11 HeaderType

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x0e
Default .... 0x00 (single function)/0x80 (multi-function)
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	HeaderType	R	Header Type. If the Reserved/MultiFunction bit in the AsicCtrl register is a logic 0, HeaderType is set to 0x00 identifying the IP100 as a single-function PCI device and specifying the configuration register layout. If the Reserved/MultiFunction bit in the AsicCtrl register is a logic 1, HeaderType is set to 0x80 identifying the IP100 as a multi-function PCI device and specifying the configuration register layout.

10.6.12 InterruptLine

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x3c
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	InterruptLine	R/W	Interrupt Line. InterruptLine specifies the interrupt level used by the IP100. By setting InterruptLine the host system may configure the appropriate interrupt vector for its Interrupt Service Routine. For 80x86 processor based host systems, InterruptLine corresponds to the IRQ number (0x00 through 0x0F), with the value 0xFF corresponding to disabled interrupts.

10.6.13 InterruptPin

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x3d
Default 0x01
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	InterruptPin	R	Interrupt Pin. InterruptPin indicates which PCI interrupt signal the IP100 will utilize. The IP100 always utilizes the INTAN interrupt signal, corresponding to an InterruptPin value of 0x01.

10.6.14 IoBaseAddress

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x10
Default 0x00000001
Width .... 32 bits

IoBaseAddress is used to define the I/O base address for the IP100. PCI systems requires that the I/O base address be set as if the system used 32-bit I/O addressing. The upper 25 bits of IoBaseAddress are read/write accessible, indicating that the IP100 requires 128 bytes of I/O space in the system I/O map.

BIT	BIT NAME	R/W	BIT DESCRIPTION
0	IoBaseAddrInd	R	I/O Base Address Indicator. When IoBaseAddrInd is a logic 1, IoBaseAddress contains the valid I/O base address for the IP100.
6..1	Reserved	N/A	Reserved for future use.
31..7	IoBaseAddress	R/W	I/O Base Address. IoBaseAddress contains the 25 bit I/O base address value. The IP100 uses 128 bytes of I/O address space.

10.6.15 LatencyTimer

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x0d
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
2..0	Reserved	N/A	Reserved for future use.
7..3	LatencyTimer	R/W	Latency Timer. LatencyTimer indicates, in increments of 8 bus clocks, the length of time which the IP100 may hold the PCI bus in the presence of other bus requestors. Whenever the IP100 asserts the FRAMEN signal, the latency timer is started. When the latency timer count expires, the IP100 must relinquish the bus as soon as its GNTN signal has been deasserted.

10.6.16 MaxLat

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x3f
Default 0x0A
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	MaxLat	R	Maximum Latency. MaxLat specifies, in 250 ns increments, how often the IP100 requires bus access while operating as a bus master. Bits 5 through 1 of the MaxLat value are loaded from the ConfigParm field within an EEPROM during auto initialization of the IP100.

10.6.17 MemBaseAddress

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x14
Default 0x00000000
Width .... 32 bits
MemBaseAddress can be disabled via loading of the ConfigParm field from an EEPROM during auto-ini-tialization of the IP100.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>0</td><td>MemBaseAddrInd</td><td>R</td><td>Memory Base Address Indicator. When MemBaseAddrInd is a logic 1, MemBaseAddrInd contains the valid memory base address.</td></tr>
<tr><td>2..1</td><td>MemMapType</td><td>R</td><td>Memory Map Type. MemMapType defines how the host system maps the IP100's registers within the host system memory space. Bit 2 of MemMapType is always a logic 0, while bit 1 is loaded from the Lower1Meg bit of the ConfigParm field within an EEPROM during auto initialization of the IP100.
<table>
<tr><td>BIT 2</td><td>BIT 1</td><td>REGISTER MAPPING</td></tr>
<tr><td>0</td><td>0</td><td>Anywhere within a 32 bit address space</td></tr>
<tr><td>0</td><td>1</td><td>Lower 1 megabyte of 32 bit address space</td></tr>
<tr><td>1</td><td>x</td><td>Undefined</td></tr>
</table></td></tr>
<tr><td>6..3</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>31..7</td><td>MemBaseAddress</td><td>R/W</td><td>Memory Base Address. MemBaseAddress contains the 25 bit memory base address value. The IP100 uses 128 bytes of I/O space.</td></tr>
</table>

10.6.18 MinGnt

Class............................ LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x3e
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	MinGnt	R	Minimum Grant Time. MinGnt specifies, in 250 ns increments, how long a burst period the IP100 requires when operating as a bus master. Bits 7 through 4 of the MinGnt value are loaded from the ConfigParm field within an EEPROM during auto initialization of the IP100.

10.6.19 NextItemPtr

Class............................ LAN PCI Configuration Registers, Power Management
Base Address .... PCI device configuration header start
Address Offset 0x51
Default 0x00
Width .... 8 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
7..0	NextItemPtr	R	Next Item Pointer. NextItemPtr indicates the next capability data structure in the capabilities list. When NextItemPtr is set to the value 0x00, there are no further data structures.

10.6.20 PowerMgmtCap

Class............................ LAN PCI Configuration Registers, Power Management
Base Address .... PCI device configuration header start
Address Offset 0x52
Default Value 0x7602
Width .... 16 bits

PowerMgmtCap provides information about the IP100's power management capabilities. Several bits are loaded from the EEPROM during auto-initialization.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>2..0</td><td>Versi on</td><td>R</td><td>Version. Version is set to 0x2, indicating PCI Bus Power Management Specification Revision 1.1.</td></tr>
<tr><td>3</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>4</td><td>Vaux</td><td>R</td><td>Auxiliary Voltage. If Vaux is a logic 1, auxiliary power via the PCI Bus is required from the system to support PME in the D3cold state. If Vaux is a logic 0, auxiliary power is supplied from elsewhere (i.e. not from the PCI Bus) to support PME in the D3cold state.</td></tr>
<tr><td>8..5</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>9</td><td>D1Support</td><td>R</td><td>D1 Power State Support. When D1Support is a logic 1, the IP100 supports the D1 power state. D1Support is loaded from the ConfigParm field of an EEPROM during auto initialization of the IP100.</td></tr>
<tr><td>10</td><td>D2Support</td><td>R</td><td>D2 Power State Support. When D2Support is a logic 1, the IP100 supports the D2 power state. D2Support is loaded from the ConfigParm field of an EEPROM during auto initialization of the IP100.</td></tr>
<tr><td>15..11</td><td>PmeSupport</td><td>R</td><td>Power Management Event Support. PmeSupport indicates the power states from which the IP100 is able to generate a power management event by asserting the WAKE signal. Each bit corresponds to a power state. A logic 1 in a particular bit position indicates that events can be generated from the indicated power state.
<table>
<tr><th>BIT 15</th><th>BIT 14</th><th>BIT 13</th><th>BIT 12</th><th>BIT 11</th><th>POWER MANAGEMENT EVENT MAY BE GENERATED FROM STATE</th></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>D0</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>D1</td></tr>
<tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>D2</td></tr>
<tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>D3Hot</td></tr>
<tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>D3Cold</td></tr>
</table>
The IP100 hard-wires bit 11 to zero and bit 14 to one. The values of bits 12,13, and 15 are determined by bits 4, 5 and 3 respectively from the EEPROM ConfigParm.</td></tr>
</table>

Preliminary, Specification subject to change without notice.

10.6.21 PowerMgmtCtrl

Class............................ LAN PCI Configuration Registers, Power Management
Base Address .... PCI device configuration header start
Address Offset 0x54
Default Value 0x4000
Width .... 16 bits
PowerMgmtCtrl allows control over the power state and the power management interrupts.

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>1..0</td><td>PowerState</td><td>R/W</td><td>Power State. PowerState indicates the current power state of the IP100. If PowerState is set to a value other than 0x0, the IP100 will not respond to PCI I/O or memory cycles, nor will the IP100 be able to generate PCI bus master cycles.
<table>
<tr><th>BIT 1</th><th>BIT 0</th><th>POWER STATE</th></tr>
<tr><td>0</td><td>0</td><td>D0</td></tr>
<tr><td>0</td><td>1</td><td>D1</td></tr>
<tr><td>1</td><td>0</td><td>D2</td></tr>
<tr><td>1</td><td>1</td><td>D3</td></tr>
</table></td></tr>
<tr><td>7..2</td><td>Reserved</td><td>N/A</td><td>Reserved for future use.</td></tr>
<tr><td>8</td><td>PmeEn</td><td>R/W</td><td>Power Management Event Enable. When PmeEn is a logic 1, the IP100 is allowed to report wake events on the WAKE signal. The criteria for generating wake events is defined by the WakeEvent register. PmeEn is loaded from the ConfigParm field of an EEPROM during auto initialization of the IP100.</td></tr>
<tr><td>12..9</td><td>DataSelect</td><td>R/W</td><td>Data Select. DataSelect is used to select which data is to be reported through the Data register and DataScale field.</td></tr>
<tr><td>14..13</td><td>DataScale</td><td>R</td><td>Data Scale. DataScale indicates the scaling factor to be used when interpreting the value of the Data register. The interpretation of the scale values is defined as follows:
<table>
<tr><th>DATASCALE</th><th>SCALE FACTOR</th></tr>
<tr><td>0x0</td><td>Unknown</td></tr>
<tr><td>0x1</td><td>0.1</td></tr>
<tr><td>0x2</td><td>0.01</td></tr>
<tr><td>0x3</td><td>0.001</td></tr>
</table></td></tr>
<tr><td>15</td><td>PmeStatus</td><td>R/W</td><td>Power Management Event Status. When PmeStatus is a logic 1 a wake event has occurred. PmeStatus may be a logic 1 regardless of the value of PmeEn. Writing a logic 1 to PmeStatus will set PmeStatus to a logic 0. Writing a logic 0 to PmeStatus has no effect.</td></tr>
</table>

10.6.22 RevisionId

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x08
Default .... Depends on revision of actual device. See description below.
Width .... 8 bits

<table>
<tr><th>BIT</th><th>BIT NAME</th><th>R/W</th><th>BIT DESCRIPTION</th></tr>
<tr><td>7..0</td><td>RevisionId</td><td>R</td><td>Revision ID. RevisionId contains the revision code for the IP100.<table><tr><td>REVISIONID</td><td>SILICON REVISION</td></tr><tr><td>0x4</td><td>B0</td></tr></table></td></tr>
</table>

10.6.23 SubsystemId

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x2e
Default 0x0000
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	SubsystemId	R	Subsystem ID. SubsystemId contains the value loaded from the ConfigParm field within an EEPROM during auto initialization of the IP100.

10.6.24 SubsystemVendorId

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x2c
Default .... EEPROM Value
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	SubsystemVendorId	R	Subsystem Vendor ID. SubsystemVendorId contains the value loaded from the ConfigParm field within an EEPROM during auto initialization of the IP100.

10.6.25 VendorId

Class............................. LAN PCI Configuration Registers, Configuration
Base Address .... PCI device configuration header start
Address Offset 0x00
Default 0x13f0
Width .... 16 bits

BIT	BIT NAME	R/W	BIT DESCRIPTION
15..0	VendorId	R	Vendor ID. VendorId contains the unique 16-bit manufacturer’s ID as allocated by the PCI Special Interest Group. The manufacturer ID is 0x13F0.

10.7 EEPROM Data Format

Figure 15 summarizes the layout of the EEPROM.

16 BIT WORD	ADDR OFFSET
StationAddress	0x12
StationAddress	0x11
StationAddress	0x10
	0x0F
	0x0E
	0x0D
	0x0C
	0x0B
	0x0A
	0x09
	0x08
	0x07
	0x06
Data	0x05
	0x04
SubsystemId	0x03
SubsystemVendorId	0x02
AsicCtrl	0x01
ConfigParm	0x00

FIGURE 15 : IP100 EEPROM Field Layout

10.7.1 AsicCtrl

Class............................ EEPROM Data Format
Base Address .... 0x00, address written to EepromCtrl register
Address Offset 0x01
Access Mode .... Read Only
Width .... 16 bits
AsicCtrl supplies the value for the least significant byte of the AsicCtrl register and the WakeEvent register.

BIT	BIT NAME	BIT DESCRIPTION
0	Reserved	Reserved for future use.
1	ExpRomSize	Expansion ROM Size. ExpRomSize corresponds to the ExpRomSize bit in the AsicCtrl register.
2	TxLargeEnable	Transmit Large Frames Enable. TxLargeEnable corresponds to the TxLargeEnable bit in the AsicCtrl register.
3	RxLargeEnable	Receive Large Frames Enable. RxLargeEnable corresponds to the RxLargeEnable bit in the AsicCtrl register.
4	ExpRomDisable	Expansion ROM Disable. ExpRomDisable corresponds to the ExpRomDisable bit in the AsicCtrl register.
5	PhySpeed10	Physical Layer Device Speed 10Mbps. PhySpeed10 corresponds to the PhySpeed10 bit in the AsicCtrl register.
6	PhySpeed100	Physical Layer Device Speed 100Mbps. PhySpeed100 corresponds to the PhySpeed100 bit in the AsicCtrl register.
7	PhyMedia	Physical Layer Device Media. PhyMedia corresponds to the PhyMedia bit in the AsicCtrl register.
11..8	Reserved	Reserved for future use.
12	Vaux	Vaux. Vaux corresponds to the Vaux bit in the PowerMgmtCap register.
13	MultiFunction	Multi-Function Device. MultiFunction corresponds to the Reserved/MultiFunction bit in the AsicCtrl register.
14	Reserved	Reserved for future use.
15	WakeOnLanEnable	Wake On LAN Enable. WakeOnLanEnable corresponds to the WakeOnLanEnable bit in the WakeEvent register.

10.7.2 ConfigParm

Class............................ EEPROM Data Format
Base Address .... 0x00, address written to EepromCtrl register
Address Offset 0x00
Access Mode .... Read Only
Width .... 16 bits

BIT	BIT NAME	BIT DESCRIPTION
0	FastBackToBack	Fast Back to Back. FastBackToBack corresponds to the FastBackToBack bit in the ConfigStatus register.
1	Lower1Meg	Lower 1 Megabyte. Lower1Meg corresponds to bit 1 of the MemMapType bit in the MemBaseAddress register.

Preliminary, Specification subject to change without notice.

BIT	BIT NAME	BIT DESCRIPTION
2	DisableMemBase	Disable Memory Base Address Register. DisableMemBase does not correspond directly to any register accessible by the host system. If DisableMemBase is a logic 1 during auto initialization of the IP100, the MemBaseAddress register will be disabled. When disabled, the value returned when the MemBaseAddress register is read is undefined.
3	D3ColdPme	D3 Cold Power Management Event. D3ColdPme corresponds to the PmeSupport bit in the PowerMgmtCap register.
4	D1Support	D1 Power State Support. D1Support corresponds to the D1Support bit in the PowerMgmtCap register.
5	D2Support	D2 Power State Support. D2Support corresponds to the D2Support bit in the PowerMgmtCap register.
6	PmeEn	Power Management Event Enable. PmeEn corresponds to the PmeEn bit in the PowerMgmtCtrl register.
10..7	MinGnt	Minimum Grant. MinGnt corresponds to bits 7 through 4 of the MinGnt register.
15..11	MaxLat	Maximum Latency. MaxLat corresponds to bits 5 through 1 of the MaxLat register.

10.7.3 Data

Class............................. EEPROM Data Format
Base Address .... 0x00, address written to EepromCtrl register
Address Offset 0x05
Access Mode .... Read Only
Width .... 16 bits

BIT	BIT NAME	BIT DESCRIPTION
15..0	Data	Data. Data corresponds to the Data register.

10.7.4 StationAddress

Class............................. EEPROM Data Format
Base Address .... 0x00, address written to EepromCtrl register
Address Offset 0x10, 0x11, 0x12
Access Mode .... Read Only
Width .... 48 bits

BIT	BIT NAME	BIT DESCRIPTION
15..0	StationAddress0	Station Address 0. StationAddress0 (offset 0x10) corresponds to the StationAddressWord0 field of the StationAddress register.
31..16	StationAddress1	Station Address 1. StationAddress1 (offset 0x11) corresponds to the StationAddressWord1 field of the StationAddress register.
47..32	StationAddress2	Station Address 2. StationAddress2 (offset 0x12) corresponds to the StationAddressWord2 field of the StationAddress register.

10.7.5 SubsystemId

Class............................. EEPROM Data Format
Base Address .... 0x00, address written to EepromCtrl register
Address Offset 0x03
Access Mode .... Read Only
Width .... 16 bits

BIT	BIT NAME	BIT DESCRIPTION
15..0	SubsystemId	Subsystem ID. SubsystemId corresponds to the SubsystemId register.

10.7.6 SubsystemVendorId

Class............................. EEPROM Data Format
Base Address .... 0x00, address written to EepromCtrl register
Address Offset 0x02
Access Mode .... Read Only
Width .... 16 bits

BIT	BIT NAME	BIT DESCRIPTION
15..0	SubsystemVendorId	Subsystem Vendor ID. SubsystemVendorId corresponds to the SubsystemVendorId register.

Preliminary, Specification subject to change without notice.

11 Signal Requirements

Note, all signal requirements are guaranteed by design only.

11.1 Absolute Maximum Ratings

Storage Temperature...................-55ºC to +125ºC
Supply Voltage................................-0.3V to +3.6V
Environmental stresses above those listed in Absolute Maximum Ratings may cause permanent damage resulting in device failure. Functionality at or above the limits listed below is not guaranteed. Exposure to the environmental stress at the levels listed below for extended periods may adversely affect device reliability.

11.2 Operating Ranges

Commercial Devices
Ambient Temperature (T_A) ....0ºC to +70ºC
Supply Voltages (V_{CC} 3.3V) .... +3.3V ±5%
Supply Voltages (V_{CC} 2.5V) .... +2.5V ±5%
Input voltages .... +3.3V or +5V ±5%
Operating ranges define the limits of guaranteed device functionality.

11.3 DC Characteristics

DC characteristics are defined over commercial operating ranges unless specified otherwise.

PARAMETER SYMBOL	PARAMETER DESCRIPTION	TEST CONDITIONS	MIN		MAX	UNIT
V_{cc} 3.3V	3.3V power voltage		3.13		3.47	V
I_{cc} 3.3V	3.3V power current				TBD	mA
V_{cc} 2.5V	2.5V power voltage		2.37		2.63	V
I_{cc} 2.5V	2.5V power current				TBD	mA
TTL I/O						
V_{IH}	Input high voltage				2	V
V_{IL}	Input low voltage				0.8	V
I_{IN}	Input leakage current	$V_{IN} = V_{CC}$ /GND	-10		10	µA
V_{OH}	Output high voltage		2.4			V
V_{OL}	Output low voltage				0.4	V
I_{OTS}	Output tri-state leakage				10	µA
100BASE-TX RECEIVE						
V_B	RXP/RXN Input Bias Voltage	I_{ol} = 4 mA		2.7		V
R_{IN}	RXP/RXN Differential Input Resistance			6		kΩ
S_{ON}	Signal Detect Turn-On Threshold	5 MHz square wave input	300	365	430	mV
100BASE-TX TRANSMIT (MEASURED DIFFERENTIALLY AFTER 1:1 TRANSFORMER)						
V_{PDO}	Peak Differential Output	50Ωfrom each output to VCC	0.95		1.05	V
V_{OI}	Output Voltage Imbalance	50Ωfrom each output to VCC			0.4	V
	Overshoot				5	%
V_{REF}	Reference Voltage of ISET			1.25		V
10BASE-T RECEIVE						
V_{B10}	RXP/RXM Input Bias Voltage			1.4		V
R_{IN10}	RXP/RXN Differential Input Resistance			10		kΩ
V_{SQ}	Squelch Threshold	5 MHz square wave		250		mV
10BASE-T TRANSMIT (MEASURED DIFFERENTIALLY AFTER THE 1:1 TRANSFORMER)						
V_P	Peak Differential Output	50Ωfrom each output to VCC	2.2		2.8	V

TABLE 3 : DC Characteristics

11.4 AC Characteristics

PARAMETER SYMBOL	PARAMETER DESCRIPTION	TEST CONDITIONS	MIN		MAX	UNIT
PCI INTERFACE						
T_{rc}	RSTN cycle		300			ns
T_{rr}	Rising edge or RSTN to chip recovery		1			µs
T_{cc}	PCICLK cycle		30			ns
T_{ch}	PCICLK high		11			ns
T_{cl}	PCICLK low		11			ns
T_{rv}	PCICLK rise to bused signal valid		2		11	ns
T_{rvp}	PCICLK rise to REQN, GNTN valid		2		12	ns
T_{rzo}	PCICLK rise to signal on		2			ns
T_{roz}	PCICLK rise to signal off				28	ns
T_{su}	bused signal setup wrt PCICLK rise		7			ns
T_{sup1}	GNTN setup wrt PCICLK rise		10			ns
T_{sup2}	REQN setup wrt PCICLK rise		12			ns
T_{hd}	signal hold wrt PCICLK rise		0			ns
T_{rstoff}	RSTN low to output signal float				40	ns
XPANSION ROM INTERFACE - READ						
T_{adv}	ED valid from EA stable		0		150	ns
T_{odv}	ED valid from EOEN low		0		70	ns
T_{dvz}	ED tri-stated from EOEN high		0		40	ns
EXPANSION ROM INTERFACE - LOAD						
T_{as} /T_{os}	EA, EOEN setup wrt EWEN fall		0			ns
T_{ah}	EA hold wrt EWEN fall		50			ns
T_{ds}	ED setup wrt EWEN rise		35			ns
T_{dh} /T_{oh}	ED, EOEN hold wrt EWEN fall		0			ns
T_{wh}	EWEN write cycle high		100		-	ns
T_{wl}	EWEN write cycle low		90		-	ns

TABLE 4 : Switching Characteristics

Preliminary, Specification subject to change without notice.

PARAMETER SYMBOL	PARAMETER DESCRIPTION	TEST CONDITIONS	MIN		MAX	UNIT
EEPROM INTERFACE						
T_{skc}	EESK cycle		1us		-	ns
T_{skh}	EESK high		250		-	ns
T_{skl}	EESK low		250		-	ns
T_{cs}	EECS low		250		-	ns
T_{pd}	EEDI valid wrt EESK rise		100		-	ns
T_{csk}	EECS setup wrt EESK rise		50		-	ns
T_{csh}	EECS hold wrt EESK fall		0		-	ns
T_{dos}	EEDO setup wrt EESK rise		70		500	ns
T_{doh}	EEDO hold wrt EESK rise		-		500	ns
100BASE-TX MDI						
T_R /T_F	Rise/Fall Time		3		5	ns
T_{IMB}	Rise/Fall Time Imbalance		0		500	ps
T_{DCD}	Duty Cycle Distortion				0.5	ns
T_J	Maximum output jitter (peak to peak)			0.7	1.4	ns
10BASE-T MDI						
T_{R10} /T_{F10}	Rise/Fall Time			25		ns
T_{J10}	Transmit Jitter				3.5	ns
MISC INTERFACE						
T_{cc}	CLK25 cycle		40		40	ns
T_{ch}	CLK25 high		16		24	ns
T_{cl}	CLK25 low		16		24	ns

TABLE 4 : Switching Characteristics

Preliminary, Specification subject to change without notice.

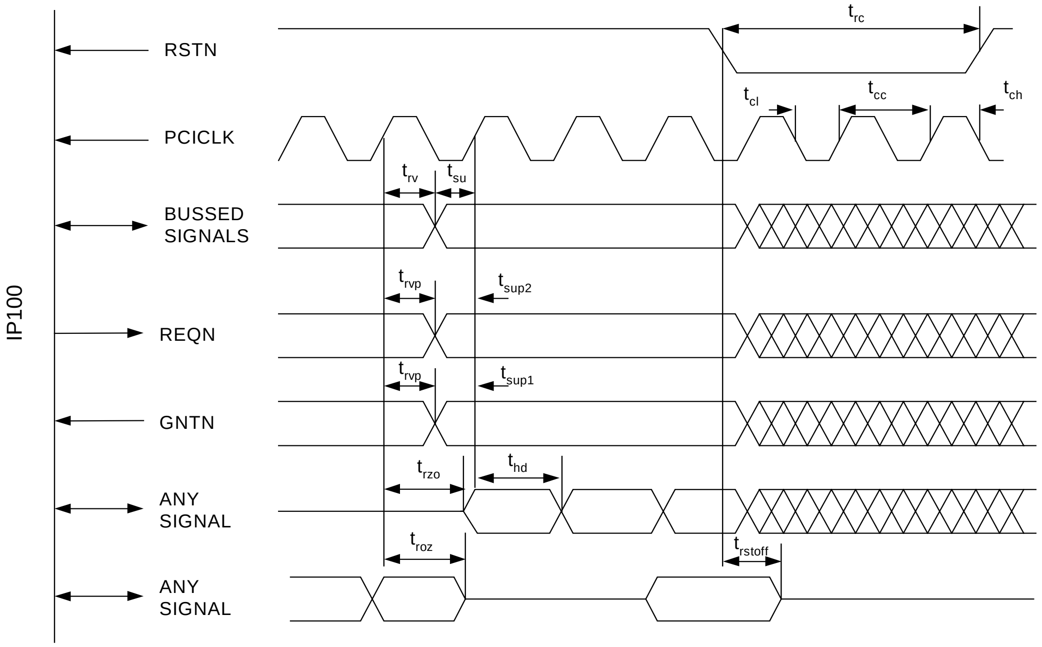


FIGURE 16 : PCI Switching Characteristics

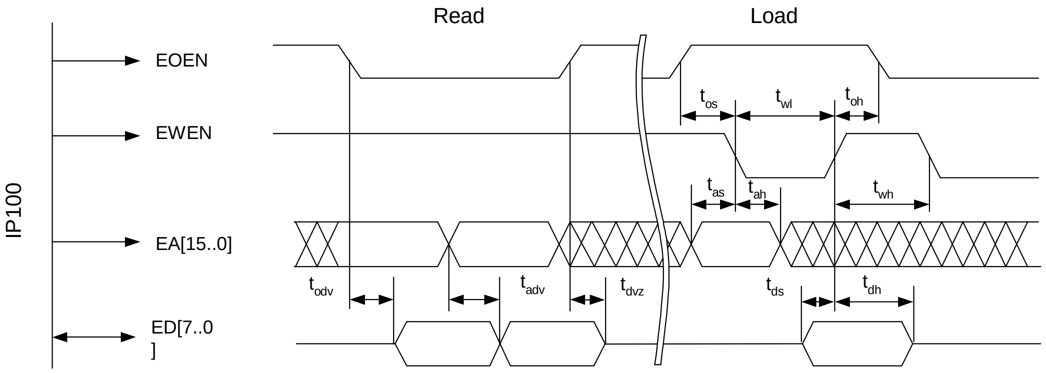


FIGURE 17 : Expansion ROM Switching Characteristics

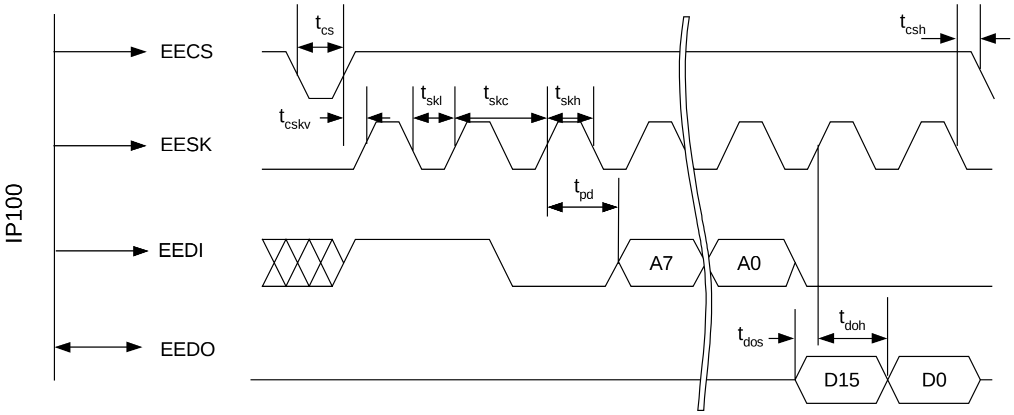
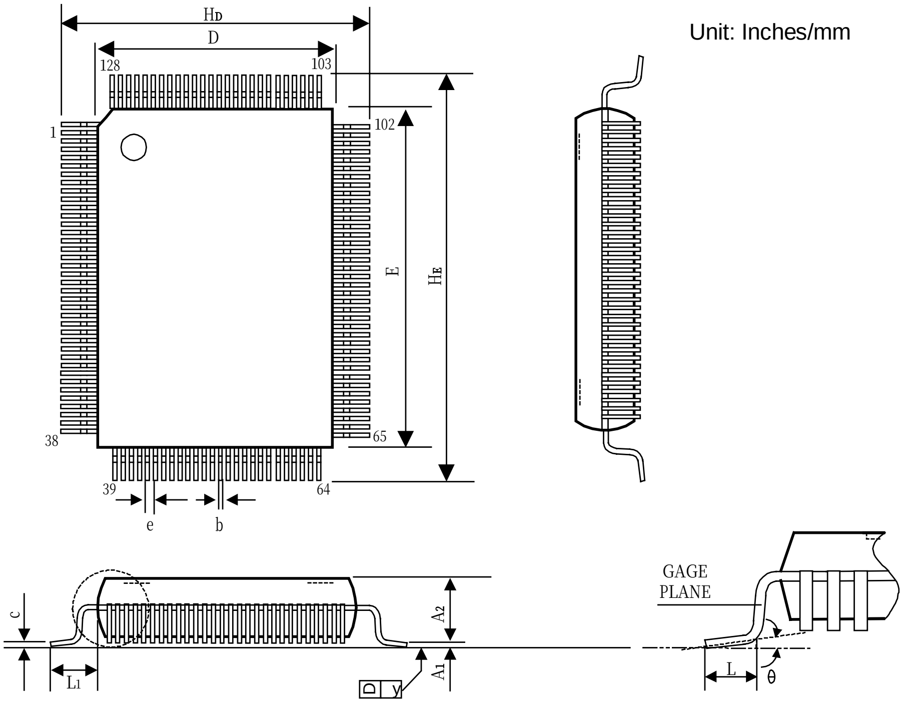


FIGURE 18 : EEPROM Switching Characteristics

Preliminary, Specification subject to change without notice.

12 Package Detail

128 PQFP Outline Dimensions



Unit: Inches/mm

Symbol	Dimensions In Inches			Dimensions In mm		
	Min.	Nom.	Max.	Min.	Nom.	Max.
A_1	0.010	0.014	0.018	0.25	0.35	0.45
A_2	0.107	0.112	0.117	2.73	2.85	2.97
b	0.007	0.009	0.011	0.17	0.22	0.27
c	0.004	0.006	0.008	0.09	0.15	0.20
H_D	0.669	0.677	0.685	17.00	17.20	17.40
D	0.547	0.551	0.555	13.90	14.00	14.10
H_E	0.906	0.913	0.921	23.00	23.20	23.40
E	0.783	0.787	0.791	19.90	20.00	20.10
e	-	0.020	-	-	0.50	-
L	0.025	0.035	0.041	0.65	0.88	1.03
L_1	-	0.063	-	-	1.60	-
y	-	-	0.004	-	-	0.10
θ	0°	-	12°	0°	-	12°

Note:

1. Dimension D & E do not include mold protrusion.
2. Dimension B does not include dambar protrusion.
   Total in excess of the B dimension at maximum material condition.
   Dambar cannot be located on the lower radius of the foot.

IC Plus Corp.

Headquarter
10F, No.47, Lane 2, Kwang-Fu Road, Sec. 2,
Hsin-Chu City, Taiwan 300, R.O.C.
TEL：886-3-575-0275 FAX：886-3-575-0475
Website：www.icplus.com.tw

Sales Office
4F, No. 106, Hsin-Tai-Wu Road, Sec.1,
Hsi-Chih, Taipei Hsien, Taiwan 221, R.O.C.
TEL：886-2-2696-1669 FAX：886-2-2696-2220