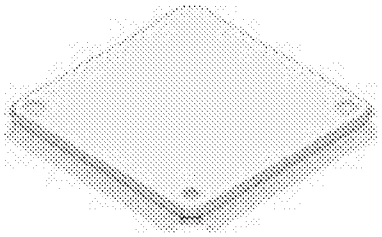
xDSL WAN Ethernet Bridge Controller

MB86976



Package

- 144-pin, plastic QFP
- FPT-144P-M03

Description

The Fujitsu MB86976 xDSL WAN Ethernet Bridge Controller is a CMOS VLSI device designed for LAN-WAN bridging applications. It offers a high level of integration that provides flexibility and seamless integration of 10 Mbps Ethernet over xDSL for Small Office Home Office (SOHO) access. The MB86976 is a multi-protocol data communications controller that supports two channels. It accepts LAN packets directly through its on-chip 10 Mbps Ethernet Media Access Controller (MAC) and it accepts a WAN serial bit stream (PPP, PPP-LEX or HDLC) with a bit rate of up to 8.2 Mbps. It provides LAN-to-WAN and WAN-to-LAN translation between LAN packets and the WAN serial bit stream. The MB86976 includes a hardware-based advanced MAC address filter and programmable features to accept or reject packets meeting user-defined data patterns, thus accelerating complex, bridging, routing, and firewall capabilities to wire speed. The MB86976 is ideally suited for building xDSL-ready Ethernet remote routers and modem equipment.

The highly integrated MB86976 device also supports system designs with standard interfaces. The processor interface supports 8- and 16-bit, multiplexed and non-multiplexed bus, microcontrollers from Intel, Philips, and Siemens. The MB86976 also interfaces to industry-standard 70 ns fast page mode 256Kx16 DRAMs, standard 10 Mbps Ethernet Transceivers (such as Fujitsu's MB86961A Universal Interface for 10Base-T or AUI for10Base-2 networks) and standard DMT (Discrete Multitone) or CAP (Carrierless Amplitude and Phase) xDSL modem chip sets.

To further ease integration of the MB86976 into a board-level product, designer kits are available.

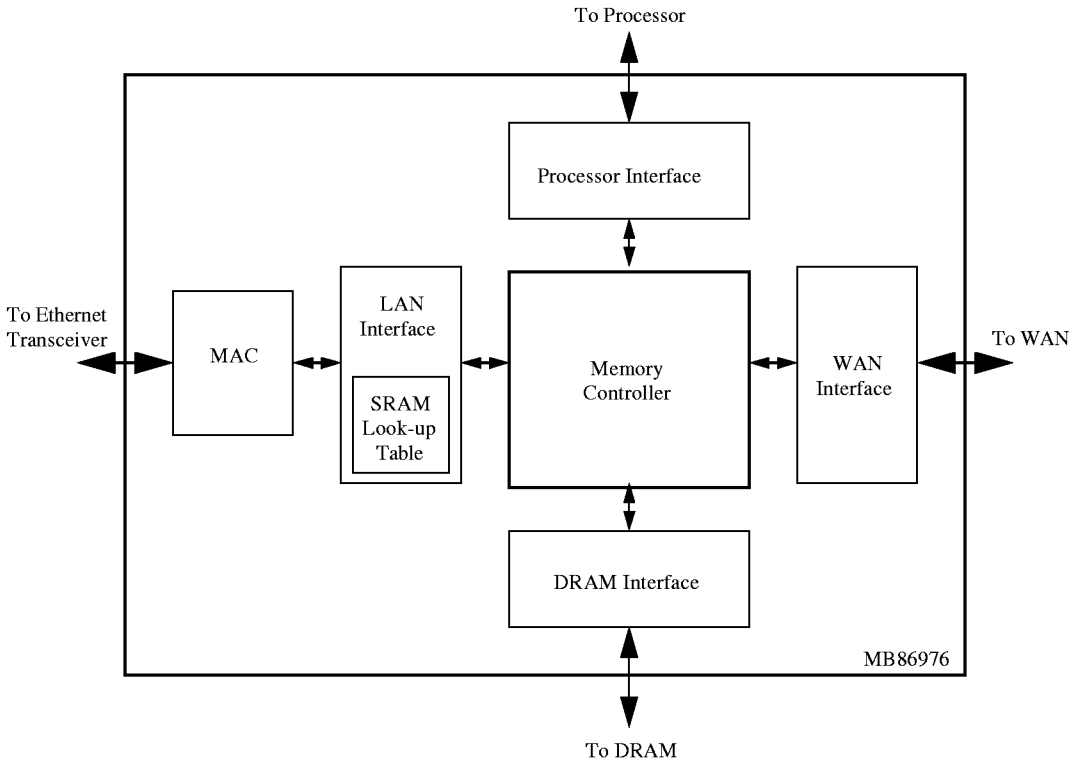
xDSL WAN Ethernet Bridge Controller

Features

- 10 Mbps full/half duplex MAC
- Store and forward packet architecture
- Dedicated DMA for memory to memory transfers between the microprocessor and DRAM
- Advanced address recognition & management
  - — SNAP encapsulated protocol filtering
  - — Hash filtering for up to 8,000 MAC addresses on both LAN and WAN ports
  - — Look-up Table Filtering for up to 8 filter strings on LAN port; can be used for protocol or special address filtering, or other user-defined filters
  - — Multicast & broadcast address recognition
  - — Bridge Protocol Data Unit (BPDU)
  - — Auto learning bridge: static and dynamic
  - — Hardware assist for Spanning Tree Algorithm IEEE802.1d, aging, and self-learning mechanism on LAN port
  - — Supports filter and network statistics
- Serial communication channel with full xDSL data bandwidth rate up to 8.2 Mbps upstream and downstream
- Multi-protocol WAN support: PPP, PPP-LEX, HDLC, and compressed HDLC
- Integrated 16-bit data DRAM controller, interfaces with standard 256K X 16, 70 ns (4Mb) fast page mode DRAMs with CAS-before-RAS refresh
- Interface for standard 8- and 16-bit, non-multiplexed and multiplexed AD bus, microcontrollers and microprocessors
- Direct interface to Fujitsu 10BASE-T and AUI MB86961A Transceiver
- Diagnostic and status LEDs for network activity indications
- JTAG boundary scan
- 40 MHz, 5 V, 0.35 micron CMOS design in a 144-pin PQFP package

Table of Contents

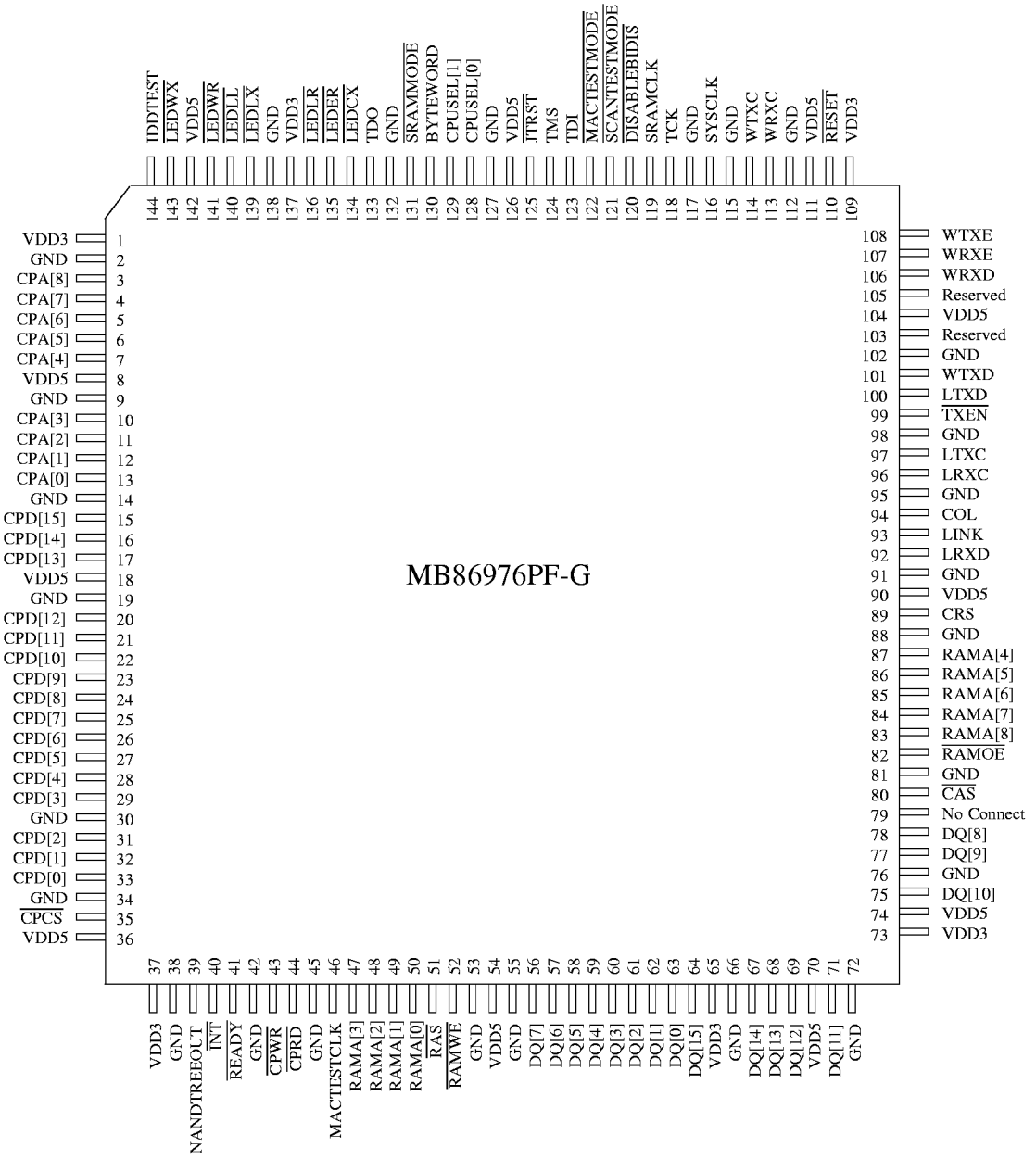
TOP LEVEL BLOCK DIAGRAM



PIN LAYOUT, ASSIGNMENTS AND DESCRIPTION

Supplied in a 144-pin plastic quad flat package (PQFP), the MB86976 presents a small footprint, reducing board space requirements, and is surface mountable with its gull-wing leads.

See Pin Assignment for the pin names and Pin Description for details. Pin names with overlines are active low signals. Pin types are Input (I), Output (O), and Bidirectional (B).



PIN ASSIGNMENT

Pin No.	Pin Name	Type	Pin No.	Pin Name	Type	Pin No.	Pin Name	Type
1	VDD3	-	41	$\overline{\text{READY}}$	O	81	GND	-
2	GND	-	42	GND	-	82	$\overline{\text{RAMOE}}$	O
3	CPA[8]	I(pu)	43	$\overline{\text{CPWR}}$	I	83	RAMA[8]	O
4	CPA[7]	I(pu)	44	$\overline{\text{CPRD}}$	I	84	RAMA[7]	O
5	CPA[6]	I(pu)	45	GND	-	85	RAMA[6]	O
6	CPA[5]	I(pu)	46	MACTESTCLK	I(pu)	86	RAMA[5]	O
7	CPA[4]	I(pu)	47	RAMA[3]	O	87	RAMA[4]	O
8	VDD5	-	48	RAMA[2]	O	88	GND	-
9	GND	-	49	RAMA[1]	O	89	CRS	I
10	CPA[3]	I(pu)	50	RAMA[0]	O	90	VDD5	-
11	CPA[2]	I(pu)	51	$\overline{\text{RAS}}$	O	91	GND	-
12	CPA[1]	I(pu)	52	$\overline{\text{RAMWE}}$	O	92	LRXD	I
13	CPA[0]	I(pu)	53	GND	-	93	LINK	I
14	GND	-	54	VDD5	-	94	COL	I
15	CPD[15]	B(pu)	55	GND	-	95	GND	-
16	CPD[14]	B(pu)	56	DQ[7]	B(pu)	96	LRXC	I
17	CPD[13]	B(pu)	57	DQ[6]	B(pu)	97	LTXC	I
18	VDD5	-	58	DQ[5]	B(pu)	98	GND	-
19	GND	-	59	DQ[4]	B(pu)	99	$\overline{\text{TXEN}}$	O
20	CPD[12]	B(pu)	60	DQ[3]	B(pu)	100	LTXD	O
21	CPD[11]	B(pu)	61	DQ[2]	B(pu)	101	WTXD	O
22	CPD[10]	B(pu)	62	DQ[1]	B(pu)	102	GND	O
23	CPD[9]	B(pu)	63	DQ[0]	B(pu)	103	Reserved	O
24	CPD[8]	B(pu)	64	DQ[15]	B(pu)	104	VDD5	-
25	CPD[7]	B(pu)	65	VDD3	-	105	Reserved	I(pu)
26	CPD[6]	B(pu)	66	GND	-	106	WRXD	I
27	CPD[5]	B(pu)	67	DQ[14]	B(pu)	107	WRXE	I
28	CPD[4]	B(pu)	68	DQ[13]	B(pu)	108	WTXE	I
29	CPD[3]	B(pu)	69	DQ[12]	B(pu)	109	VDD3	-
30	GND	-	70	VDD5	-	110	$\overline{\text{RESET}}$	I(pu)
31	CPD[2]	B(pu)	71	DQ[11]	B(pu)	111	VDD5	-
32	CPD[1]	B(pu)	72	GND	-	112	GND	-
33	CPD[0]	B(pu)	73	VDD3	-	113	WRXC	I
34	GND	-	74	VDD5	-	114	WTXC	I
35	$\overline{\text{CPCS}}$	I	75	DQ[10]	B(pu)	115	GND	-
36	VDD5	-	76	GND	-	116	SYSCLK	I
37	VDD3	-	77	DQ[9]	B(pu)	117	GND	-
38	GND	-	78	DQ[8]	B(pu)	118	TCK	I
39	NANDTREEOUT	O	79	No Connect		119	SRAMCLK	I(pu)
40	$\overline{\text{INT}}$	O	80	$\overline{\text{CAS}}$	O	120	$\overline{\text{DISABLEBIDIS}}$	I(pu)

Pin No.	Pin Name	Type	Pin No.	Pin Name	Type	Pin No.	Pin Name	Type
121	$\overline{\text{SCANTESTMODE}}$	I(pu)	129	CPUSEL[1]	I(pu)	137	VDD3	-
122	$\overline{\text{MACTESTMODE}}$	I(pu)	130	BYTEWORD	I(pu)	138	GND	-
123	TDI	I(pu)	131	$\overline{\text{SRAMMODE}}$	I(pu)	139	$\overline{\text{LEDLX}}$	O
124	TMS	I(pu)	132	GND	-	140	$\overline{\text{LEDLL}}$	O
125	$\overline{\text{JTRST}}$	I(pu)	133	TDO	O	141	$\overline{\text{LEDWR}}$	O
126	VDD5	-	134	$\overline{\text{LEDCX}}$	O	142	VDD5	-
127	GND	-	135	$\overline{\text{LEDER}}$	O	143	$\overline{\text{LEDWX}}$	O
128	CPUSEL[0]	I(pu)	136	$\overline{\text{LEDLR}}$	O	144	IDDTEST	I(pu)

Note: VDD3: Internal VDD, 3.3V
VDD5: I/O VDD, 5.0V
GND: GROUND

(PU): INTERNAL pull-up resistor, 50KΩ
Reserved: do NOT connect

PIN DESCRIPTIONS

LAN Interface

Pin Name	Pin No.	Type	Description
$\overline{\text{TXEN}}$	99	O	LAN Transceiver Enable: Active low signal. Enables data transmission.
LTXC	97	I	LAN Transmit Clock: 10 MHz transmit clock.
LTXD	100	O	LAN Transmit Data: Transmit data from MB86976 to the ethernet transceiver.
LRXD	92	I	LAN Receive Data: MB86976 receives data from the ethernet transceiver.
LRXC	96	I	LAN Receive Clock: 10 MHz receive clock, synchronous with the receive data on LRXD.
CRS	89	I	LAN Carrier Sense: Notifies MB86976 of activity on the network.
COL	94	I	Collision Detect: Notifies MB86976 of a collision detected on the network.
LINK	93	I	Link: Indicates link up status to MB86976. Polarity depends on transceiver.

WAN Interface

Pin Name	Pin No.	Type	Description
WRXD	106	I	WAN Receive Data: Incoming serial data is sampled on the rising edge of WRXC.
WRXC	113	I	WAN Receive Clock: This is a free running clock, not burst.
WRXE	107	I	WAN Receive Enable: Asserted when valid data is ready for reception.
WTXD	101	O	WAN Transmit Data: Data will be asserted on the negative edge of WTXC, and the transceiver should recover data on the positive edge of WTXC.
WTXC	114	I	WAN Transmit Clock: This is a free running clock.
WTXE	108	I	WAN Transmit Enable: The MB86976 latches the enable on a positive edge of WTXC, and transmits valid data on the following negative edge of WTXC for transmission.

Processor Interface

Pin Name	Pin No.	Type	Description
CPD[15:0]	15-17, 20-29, 31-33	B(pu)	Processor Data Bus: Data transfers take place over this bus.
BYTEWORD	130	I(pu)	Processor Byte Wide/Word Wide Interface Select: =0 Selects 8-bit, =1 selects 16-bit processor interface.
$\overline{\text{CPRD}}$	44	I	Processor Read: Active low signal for Read operations.
$\overline{\text{CPCS}}$	35	I	Processor Chip Select: Active low signal.
$\overline{\text{CPWR}}$	43	I	Processor Write: Active low signal for Write operations.
CPUSEL[1:0]	129,128	I(pu)	Processor Select: Selects the type of processor that is being used. 00: Selects an Intel 80c32 or Philips XA-G3 family of microcontrollers. 01: Selects a Siemens c163 family of microcontrollers. 10-11: Reserved
CPA[8:0]	3-7, 10-13	I(pu)	Processor Address Bus: MB86976 Register Address space.
$\overline{\text{INT}}$	40	O	Interrupt: Active low signal. Asserted by the MB86976 when it needs service by the microcontroller. NOT open-drain, cannot be wire-ANDed.
$\overline{\text{READY}}$	41	O	Ready: Active low signal. Acknowledge completion of bus transaction. Cannot be wire-ANDed.

DRAM Interface

Pin Name	Pin No.	Type	Description
RAMA[8:0]	83-87, 47-50	O	DRAM Address Bus: Multiplexed Row and Column Addresses.
$\overline{\text{RAS}}$	51	O	Row Address Strobe: Active low signal. Addresses the row on the DRAM.
$\overline{\text{CAS}}$	80	O	Column Address Strobe: Active low signal. Addresses the column on the DRAM.
$\overline{\text{RAMWE}}$	52	O	Write Enable: When low, enables writes to the DRAM. When high, enables read from the DRAM.
$\overline{\text{RAMOE}}$	82	O	Output Enable: Enables read from the DRAM.
DQ[15:0]	64, 67-69, 71, 75, 77, 78, 56-63	B(pu)	Data Input/Output: All DRAM data transfers take place over this bus, and are 16 bit wide. Upper and Lower Column Address Strobe lines on byte-capable memories need to be tied together.

LED Interface

Pin Name	Pin No.	Type	Description
$\overline{\text{LEDWR}}$	141	O	WAN Receive Active LED: Asserted when receiving data on the WAN port.
$\overline{\text{LEDWX}}$	143	O	WAN Transmit Active LED: Asserted when transmitting data on the WAN port.
$\overline{\text{LEDLL}}$	140	O	Link LED: Asserted when the 10BASE-T link is operational, from LINK pin.
$\overline{\text{LEDLR}}$	136	O	LAN Receive Active LED: Asserted when receiving on the LAN port.
$\overline{\text{LEDLX}}$	139	O	LAN Transmit Active LED: Asserted when transmitting on the LAN port.
$\overline{\text{LEDCX}}$	134	O	LAN Collision LED: Asserted when a collision is detected on the LAN.
$\overline{\text{LEDER}}$	135	O	Error LED: Active for any internal buffer or FIFO overflow.

System Pins

Pin Name	Pin No.	Type	Description
SYSCLK	116	I	System Clock: Main system clock, 40MHz.
$\overline{\text{RESET}}$	110	I(pu)	Reset Input: Active low. Complete system reset, including the MAC.

Test Pins (not used in normal operation)

Pin Name	Pin No.	Type	Description
TCK	118	I	JTAG Test Clock. Tie LOW if not in use.
TMS	124	I(pu)	JTAG Test Mode. Tie LOW in normal operation.
$\overline{\text{JTRST}}$	125	I(pu)	JTAG Asynchronous Reset: Active low. Tie HIGH if not in use.
TDI	123	I(pu)	JTAG Test Data In. Tie LOW if not in use.
TDO	133	O	JTAG Test Data Out
$\overline{\text{MACTESTMODE}}$	122	I(pu)	MAC Test Clock Select: Tie HIGH for normal operation.
SRAMCLK	119	I(pu)	RAMScan Mode Clock: Tie HIGH for normal operation.
$\overline{\text{SRAMMODE}}$	131	I(pu)	RAMScan Mode Enable: Tie HIGH for normal operation.
$\overline{\text{SCANTESTMODE}}$	121	I(pu)	Scan Mode Enable: Tie HIGH for normal operation.
NANDTREEOUT	39	O	Parametric NAND Tree Out: For Fujitsu Testing.
$\overline{\text{DISABLEBIDIS}}$	120	I(pu)	Bidirectional Disable: Tie HIGH for normal operation. Forces CPD and DQ Buses into Input Mode.
IDDTEST	144	I	Chip Testing Enable: For Fujitsu testing. Leave floating for normal operation.
MACTESTCLK	46	I(pu)	MAC Test Clock: MUST BE TIED HIGH for normal operation.

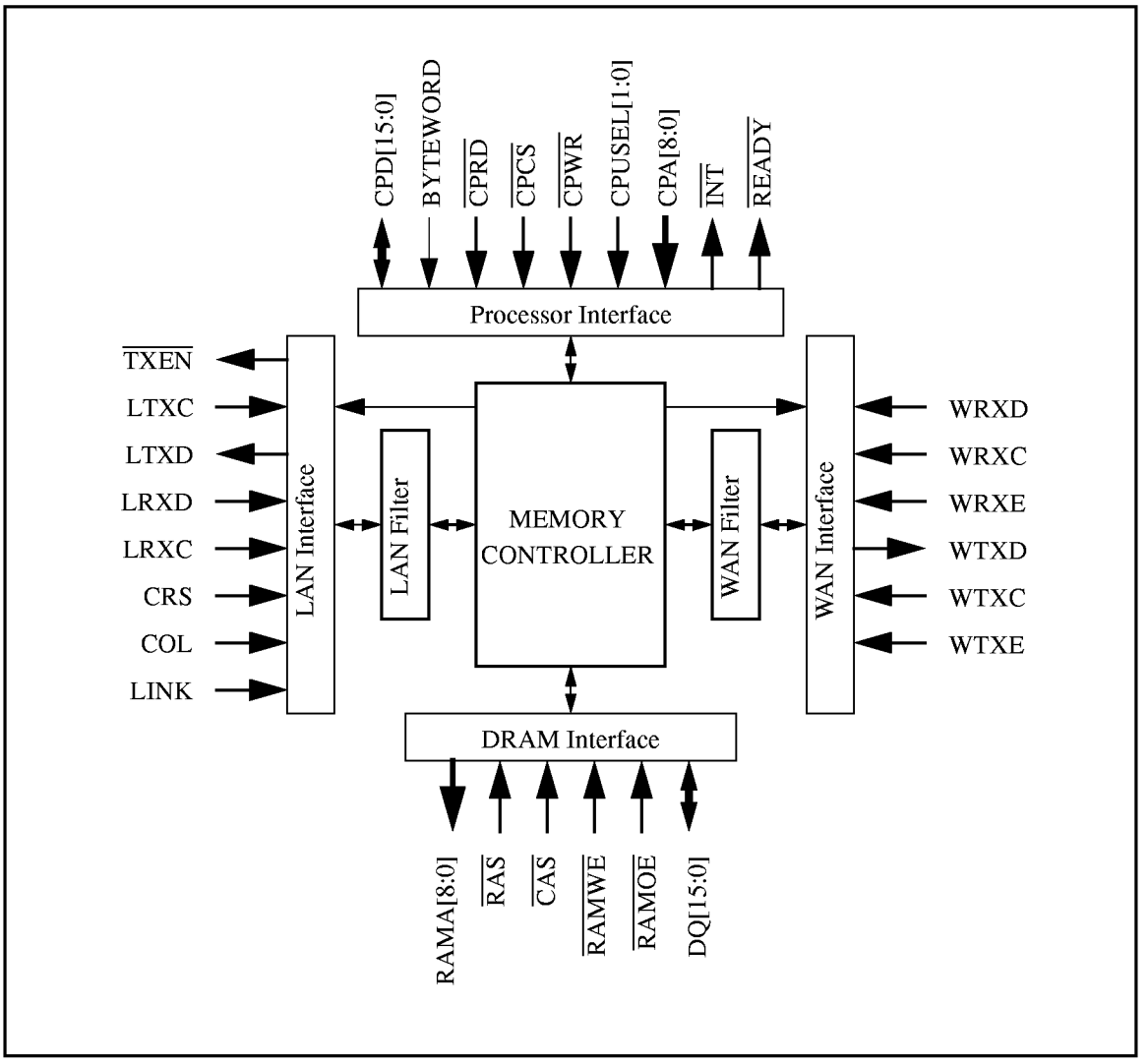
Functional Description

The MB86976 provides remote bridging of data between a 10 Mbps Ethernet LAN and a multi-protocol point-to-point WAN. The WAN port supports a serial datastream with a maximum bit rate of 8 Mbps. Advanced address filtering and user-defined content filtering enable high-performance bridging and routing functions.

The MB86976 requires only a low cost microcontroller, a 256Kx16 DRAM, an Ethernet transceiver, and an xDSL modem or T1 framer to create a complete system configuration.

The Functional Block Diagram shows the five major blocks of the MB86976: LAN Interface, WAN Interface, Processor Interface, DRAM Interface, and Memory Controller. A functional overview of each interface, the device's bridging functions (including LAN and WAN filtering), and its related system features are described here.

Functional Block Diagram



LAN Interface

The LAN Interface controls transmission and reception of frame data on the LAN link. The MB86976 incorporates a fully functional 10 Mbps Ethernet Media Access Controller (MAC) core. Outputs are provided for receive, transmit, collision, and link test LEDs.

The MAC interfaces with an external transceiver (such as the Fujitsu MB86961A Universal Interface for 10BASE-T) using a 10 Mbps serial interface. The MAC presents and expects the raw Ethernet frame as a bitstream to and from the transceiver. Manchester encoding/decoding and all additional signalling is performed by the transceiver.

MAC Register Map

The MAC contains registers to control its operation, report on status, and return statistics on device operation. They are accessed as a subset of the MB86976's overall register set, and from the processor's perspective, are no different from other registers. The address map and a basic description of the register set are listed under the section headed"Functional Control and Status Registers" on page 23.

Lan Data Transfer

All transfers of LAN frame data occur between the Memory Controller and the internal MAC. Received frames are transferred to a temporary buffer in the Memory Controller before being transferred to the frame buffer in the external DRAM. The filter engine accesses received frame data from this temporary buffer within the Memory Controller. Frames originating in the processor or the WAN must first be transferred to this frame buffer, where they are queued for transmission.

Operation

The Ethernet MAC operates under control of the external processor. Transmission and reception must be configured through writes to the LAN_TxControl (LAN Transmit Control), LAN_RxControl (LAN Receive Control), and MAC_CTL (MAC Control) registers. Once configured, normal MAC operation requires a minimum of intervention by the processor.

Information regarding transmission of data through the MAC core is given for informational value only. The operation here is transparent to the user.

Transmission

Once the MAC transmission function is enabled through a write to the transmit enable bit of the Transmit Control register, the MAC is capable of transmitting frames. The Memory Controller can begin writing data to the Transmit FIFO at any time. The MAC will wait for 64 bytes of data to be written before acquiring the network link, so that back-off with retransmission is possible without extra intervention.

As long as there is space, the Memory Controller can feed additional frame data to the MAC Transmit FIFO. To maximize usage of Fast Page Mode memory access, the Memory Controller will transfer frame data in 4-word blocks whenever that amount of data is available. When passing the final byte of frame data, the Memory Controller indicates an end of frame. The MAC then completes the transmission, and sets the transmit status registers. The MAC also implements the interframe gap without processor intervention.

Reception

The reception process begins once the CPU has configured the MB86976 MAC by writing to the Rx_EN (receive enable) bit of the LAN_RxControl register. Once in operation, the MAC monitors the data passed to it by the physical layer and checks for valid frames. If it identifies a frame preamble, it will begin reception. The MAC strips the preamble and Start Frame Delimiter (SFD) from the frame and begins calculation of the CRC. Once the MAC receives 8 Bytes in the 16-Byte receive FIFO it will signal that the Memory Controller must accept data from it at its earliest convenience. The MAC then exchanges data with the Memory Controller. This process is repeated until the frame is entirely received, when the MAC will set the receive status registers.

Interrupts

There are a number of transmit and receive interrupt options. They are configured by setting the appropriate bits of the MAC Receive and Transmit Control registers. They are masked by clearing these bits. Even when these interrupts are not set, the Receive and Control Status registers record for each frame whether a given interrupt would have been asserted, so that the processor may poll the status registers to determine interrupt status without suffering the penalty of a hardware interrupt.

WAN Interface

The WAN Interface controls the transmission and reception of frame data onto the WAN link. It supports four WAN protocols: Compressed High-level Data Link Communication (HDLC), HDLC, Point-to-Point (PPP) and PPP with LAN Extension (PPP-LEX). The MB86976's default configuration is for Compressed HDLC communication. The WAN Interface is capable of operating in full-duplex mode at speeds of up to 8.2 Mbps upstream and downstream.

The WAN Interface is divided logically into receive and transmit modules. The receive module performs four major functions. It:

1. Recovers the physical bitstream from the WAN link, including removal of zero insertions;
2. Examines the packet to decode its destination (LAN, processor, or both);
3. Removes the WAN encapsulation in accordance with the selected WAN protocol type; and
4. Assembles the resulting frame for transfer to the frame buffer.

The transmit module performs analogous, but opposite functions. It:

1. Receives frames from the Memory Controller and buffers them in a FIFO;
2. Constructs the proper WAN encapsulation via configuration registers and a Frame Check Sequence (FCS) generator;
3. Performs zero insertion; and
4. Transmits the bit stream synchronous to the external transmission clock.

Configuration Registers

There are ten configuration registers which configure operation of the WAN link. These are listed under the section headed "Functional Control and Status Registers" on page 23.

The WAN_TxContol and WAN_RxControl registers control the choice of protocol on the transmit and receive links, respectively. Depending upon the protocol chosen, additional control registers are used to perform proper data encapsulation. All registers are 8 bits in size, unless otherwise noted.

WAN Frame Formats

WAN protocol	Header Format	Data Value
C-HDLC	< No WAN Header >	
HDLC	HDLC_address	FF
	HDLC_control	03
PPP	HDLC_address	FF
	HDLC_control	03
	PPP_Protocol2	00
	PPP_Protocol1	41
PPP-LEX	HDLC_address	FF
	HDLC_control	03
	PPP_Protocol2	00
	PPP_Protocol1	41
	LEX_FIZ0	00
	LEX_MacType	01

This table shows the Header fields in the order transmitted for each of the possible WAN protocols. Following each field is a typical value (in hex) that may be transmitted in this field in order for the WAN receive port to accept a frame in that protocol. Note that the values programmed into the PPP_Protocol1 & PPP_Protocol2 register in this table are not the default values. The default values for these two fields must not be used. After the header fields, the WAN frame contains (encapsulates) everything in the original LAN frame after (but not including) the Ethernet SFD field.

Finally, after this encapsulated LAN frame, two bytes of WAN checksum (the WAN FCS) are always appended for each protocol.

Operation

Transmission

The WAN Interface informs the Memory Controller module when there are 4 free words in the WAN Interface transmit FIFO for a new transmission operation. Once the Memory Controller has identified a packet bound for WAN transmit and this signal has been asserted, the Memory Controller begins loading the transmit FIFO with the packet data. Transfer of additional words from the Memory Controller to the WAN Interface transmit FIFO will occur at the discretion of the WAN Interface, with no assumptions on throughput or acceptable idle windows.

After the FIFO is loaded, the WAN Interface transmit logic will transmit the appropriate header information as required by the chosen protocol. Note that packets originating in the CPU will

already contain PPP protocol fields, while LAN packets will not. The header fields that must be transmitted will have their values read in from the appropriate MB86976 registers: HDLC_address, HDLC_control, PPP_Protocol, LEX_FIZ0 and LEX_MacType. The values in these registers are transmitted least significant bit first.

WAN frames sourced from the CPU (instead of the LAN port) will not have the PPP specific fields automatically added by the hardware, and so firmware must explicitly prepend the appropriate data to the frame it has constructed. The Memory Controller informs the WAN Interface of the originator of the packet so that it may choose the appropriate encapsulation.

If the WTXE pin is asserted, the MB86976 places bits onto WTXD synchronous with every falling edge of WTXC. If a frame is ready and the WAN_Tx_ENable bit in WAN_TxControl is hi, the WAN Interface then begins transmitting the data loaded in the transmit FIFO, otherwise HDLC flags (01111110) are transmitted. A zero bit will be inserted after each occurrence of five consecutive one bits, as per the HDLC protocol for synchronous transmission. When WTXE is deasserted, the MB86976 halts transmission.

In parallel, the FCS checksum generator computes the FCS for the current frame prior to zero insertion. After the Memory Controller reaches the end of the frame and the final data words from the FIFO have been transmitted, the generated FCS followed by an HDLC flag is transmitted to complete the WAN packet.

While the FCS is being transmitted, the WAN Interface indicates to the Memory Controller that another transmission may be initiated, since free space will exist in the transmit FIFO. If a second frame is currently queued in the frame buffer, it can be loaded into the FIFO, and back-to-back transmission can be achieved. The WAN Interface transmits the next frame after the previous frame has completed and at least one flag sequence has been transmitted. If the transmit FIFO is not full at the completion of the flag sequence, the flag sequence will be repeated until a full FIFO is detected.

Reception

The WAN Interface receive module will operate with a receive clock up to a maximum frequency of 8.2 MHz. On each positive transition of WRXC that is accompanied by an active WRXE, the receive logic samples WRXD and stores the sampled bit in the open position of an 8-bit serial-to-parallel register. Logic attached to this register continually senses for an 8-bit sequence that matches the HDLC flag (01111110). Once this flag is identified, subsequent bits are considered to be the input data stream unless and until another flag sequence is encountered. Zero stripping is also completed in this module following the receipt of five contiguous one bits, consistent with the HDLC transmission protocol.

Each 8-bit byte recovered from WRXD will be processed for header information, according to the protocol choice selected in WAN_RxControl. If PPP or PPP-LEX are selected, the MSB of the protocol2 field is checked to determine whether the frame is a control packet or a data packet. Frames with the MSB of the protocol field set are control packets and are forwarded to the processor. Data packets continue on to the next stage of filtering. Note that header encapsulation is preserved for packets destined for the processor, and removed for packets destined for the LAN. After determining the packet destination, the MB86976 loads received data bytes into the receive FIFO. The receive FIFO consists of eight 16-bit words, with control outputs to the Memory Controller to indicate start/end of frame, frame destination (processor, LAN, or both), and the discard of a frame already partially or wholly transferred.

If PPP or PPP-LEX mode is selected, one bit of both of the PPP_Protocol bytes is checked on receive. The LSbit of the received PPP_Protocol2 field must be a zero, and the LSbit of the PPP_Protocol1 field must be a one. If either condition does not occur, the frame is discarded and WAN_RxErr_Cnt increments. WAN_Reject_Frames does not increment.

The HDLC_address header field, if present, must equal FF, and the HDLC_control field must then have a value of 03, or the WAN frame will be rejected as a protocol error.

If present in the selected WAN mode, the LEX_FIZ0 and LEX_MacType header fields of the WAN frame must have values that match the corresponding MB86976 register settings, or that frame will be rejected as a protocol error.

In parallel with frame reception, each incoming byte is used to compute the FCS. Once the terminating flag byte is detected, the FCS is computed by the polynomial [$x^{16} + x^{12} + x^5 + 1$] and checked against the constant 1D0Fh, as specified in ISO 3309, sec. 4.6.2. If the FCS comparison fails, the WAN Interface directs the Memory Controller to discard the frame.

Also in parallel with frame reception, each incoming frame's length, in number of bytes, is counted. If the frame contains

more bytes than the maximum allowed size of an Ethernet frame with maximum WAN overhead (1526 = 1518+8), the WAN Interface signals the Memory Controller to discard the frame.

The encapsulated MAC Destination Address in frames received on the WAN link will also be passed through the SADB filtering engine and searched against the source address database. This allows the WAN frames to be filtered for the MB86976 MAC address as well as any static entries the processor may wish to add to the database. WAN Frames sent to the CPU as a result of such filtering will not have any WAN header information associated with them. WAN Frames sent to the CPU will have a WAN header only if they were directed to the CPU as a result of the MSBit set in the PPP_Protocol2 field of a received WAN frame (when the receive mode is set to one of the two PPP modes).

Statistics

Six registers provide statistical information regarding frame transmission and reception. The statistics counters are read-only and reset only at system reset. They are updated as each logged event occurred, e.g., the byte counter is updated as each byte is received, the frame counter is updated as each frame is received, etc. It is the responsibility of software to maintain larger counters to deal with overflows. These registers are described at the end of "WAN Interface Registers (Group WAN)" on page 34.

Processor Interface

Three families of 8- and 16-bit microcontrollers are supported: Intel 80c32, Philips P51XA-G3, and Seimens c163. To support the variety of interfaces offered by these processors, the Processor Interface manages processor access to the MB86976 memory system. Commands are issued to the Processor Interface which in turn accesses the MB86976 register space and DRAM memory space.

The choice of processor interface is controlled by the hardwiring of CPUSEL[1:0], and determines which of the available signals are used and how they are interpreted by the MB86976. See table "CPU to MB86976 Pin Mapping" on page 14 for the mapping from each processor's external pins to the external pins of the MB86976.

CPU to MB86976 Pin Mapping

CPU Type	CPU Pin	MB86976 Pin
Intel 80c32 and similar (MB86976 CPUSEL = 00)	Port 0(AD7:AD0)	CPD[7:0]
	Latched Port 0	CPA[7:0]
	Port 2 (A8)	CPA[8]
	$\overline{RD}$	$\overline{CPRD}$
	$\overline{WR}$	$\overline{CPWR}$
	$\overline{INT0}$ or $\overline{INT1}$	$\overline{INT}$
	GND	BYTEWORD
Philips P51XA-G3 (MB86976 CPUSEL = 00)	A11D7-A4D0	CPD[7:0]
	A8D4 - A4D0	CPA[8:4] (latched)
	A3-A0	CPA[3:0]
	$\overline{WR}$	$\overline{CPWR}$
	$\overline{RD}$	$\overline{CPRD}$
	WAIT	$\overline{READY}$
	$\overline{INT0}$ or $\overline{INT1}$	$\overline{INT}$
	GND	BYTEWORD
Siemens c163 (MB86976 CPUSEL = 01)	P0[15:0]	CPD[15:0]
	P1[8:0]	CPA[8:0] (DEMUX mode)
	$\overline{WR/WRL}$	$\overline{CPWR}$
	$\overline{RD}$	$\overline{CPRD}$
	EX0IN-EX7IN	$\overline{INT}$
	$\overline{READY}$	$\overline{READY}$
	VDD	BYTEWORD

MB86976 Communication with processor

The processor is capable of accessing the MB86976 register space and (indirectly) the external DRAM memory space through the Processor Interface. No DRAM memory accesses are performed directly by the processor; rather, it will write data and addresses into temporary registers in the Processor Interfacein the MB86976 and internal logic will subsequently perform the memory access. This buffering allows isolation between the distinct timing regimes of the processor and the DRAM. By mediating memory accesses with the Processor Interface, the MB86976 centralizes control of all memory transactions for each of the particular processors.

The supported processors offer different size data buses. To accommodate this difference the BYTEWORD input pin to the MB86976 will select either byte-wide (8 bit) or word-wide (16 bit) data. The address space on the MB86976 has been chosen to make access to the memory simple and consistent in these two modes. All addresses refer to byte quantities, but every 8, 16 and 32 bit

register is aligned on an even-numbered address. The memory-mapped register layout is little endian. That is, the least significant byte of any register always starts at an even address.

When byte-wide data is selected by a '0' on the BYTEWORD pin, the 8-bit registers are addressed directly, and the 16-bit registers as two successively addressed bytes, with the least significant byte at the lower address. The 32-bit registers are interpreted as four successively addressed bytes, with the least significant byte at the lowest address.

When word-wide databus operation is selected by a '1' on the BYTEWORD pin, and an 8-bit register needs to be addressed alone, it will be found aligned on an even-numbered address and the next odd-numbered (empty or padding) byte is concatenated to form 16 bits. That is, when word-wide data is selected by BYTEWORD, the addressed byte and the next higher byte are chosen by a read/write transaction. While in word mode, odd-number addresses are not allowed (i.e., the LSbit of the address of all word-wide transactions is considered to be zero). For 8-bit registers, the high-order byte must be ignored by software. The 16-bit registers are aligned on an even-numbered addresses and addressed directly in word mode. The 32 bit registers can be interpreted as two successively addressed words, with the least significant word at the lower address.

The registers FrameBuffer and DMA_FIFO are special exceptions to the above pattern in that their width depends on the state of BYTEWORD. They are word-wide when the MB86976 is in word mode, and byte-wide when it is in byte mode.

The supported processors also utilize different mechanisms for interfacing with memory. The Intel 80c32 uses a semi-synchronous protocol which assumes the memory device can respond to the read or write strobe in time before the end of the cycle. The Siemens c163 and Philips XA-G3 on the other hand, use an asynchronous interface that waits for the addressed device, in this case the MB86976, to assert a $\overline{\text{READY}}$ signal after each read or write request. The Processor Interface contains separate read/write hardware to support these two types of interfaces.

Semi-Synchronous Mode

The Intel 80c32 interfaces with external memory and peripherals without explicit acknowledgment of bus transactions. This restriction is not an issue for memory accesses within the MB86976 register space, as the device can respond to these transactions within the available bus window. General accesses to the external DRAM, however, could exceed this window during MB86976 operation if they were allowed. To address this latency mismatch between the 80c32 and the DRAM channel on the MB86976, a (pseudo) DMA mode is supplied for all direct processor accesses to DRAM.

For reads and writes to the register space, the processor executes each read/write operation individually. The Processor Interface latches the memory transaction and decodes the address. If the transaction is within the register space, the Processor Interface will pass the request to the register set. Conflicts between processor access to the register set and MB86976 updates of those registers are settled in favor of the processor, delaying the update. Because the register set is readily accessible, the device is able to respond to a read transaction within the read window of the Intel 80c32.

Note that address latching must occur externally for the multiplexed address/data bus on the 80c32 and the P51XA-G3 microcontrollers, using their ALE pin. Address latching is not required for the C163 in demultiplexed mode. The address bits to be latched in each case differ. The $\overline{\text{READY}}$ output will be active during 80c32 mode transactions to allow for connection to the P51XA-G3 WAIT pin.

Processor-DRAM pseudo-DMA operations

On receipt of a frame bound for the processor, the MB86976 loads an 8-word buffer in the Processor Interface with the next eight words of the frame, including the header word. Once at least four words are resident in the buffer, the ProcessorPacket flag is set. After polling or clearing the interrupt, the processor reads four words from location FrameBuffer. When this FIFO becomes half empty, the processor loads it with the next four words of the frame and then sets the FrB_Read_GO register. The processor polls this register until it is set, at which point the processor can retrieve the next set of words from the local buffer. At the end of the frame, the processor uses its knowledge of the frame length, taken from the header word, to determine the correct total number of reads from FrameBuffer.

To write a frame into the frame buffer for transmission on the LAN or WAN links, the processor writes four words to location FrameBuffer. It then writes to location FrB_Write_GO, which allows the processor to pull the four words from the internal FIFO and store them into the DRAM. The first word of the frame must be the header, which specifies the size and destination of the frame. Documentation of the header structure appears in "Frame Buffer

Database Format" on page 16. Once the internal FIFO is emptied, the processor will signal the ProcessorComp interrupt. After clearing the interrupt, the processor writes the next four words of the frame, and the cycle is repeated until the entire frame has been written. At the end of the frame, the last group of four or fewer words is written into the memory. Instead of FrB_Write_GO, the processor writes to location End_of_Frame, which signals the processor to empty the FIFO, even though it may contain fewer than four words. The processor will signal a final ProcessorComp interrupt, at which point it is ready to begin accepting another frame from the processor.

Reads and writes directly to the DRAM are accomplished through a DMA submodule of the Processor Interface. For reads, the processor will supply a start address in DMA_StartAddress, a length value in DMA_Length, and write the DMA_Read_GO register. The Processor Interface will fetch the required data from the DRAM and store it in a temporary buffer, and then assert the DMA_Done bit of the interrupt register. By polling this register, or through interrupt, the processor can determine whether the data is available, when it performs repeated reads to this buffer until the data is transferred. Writes are performed similarly, by writing the start address, length, and data words to buffers in the Processor Interface and then writing the DMA_Write_GO bit. The processor may then poll or be interrupted by the interrupt register in a manner identical to that for reads.

Asynchronous Mode

The asynchronous mode is controlled by the use of a feedback signal from the MB86976 to confirm each read/write transaction with the processor. The Siemens c163 uses its $\overline{\text{READY}}$ input to confirm transactions and the Philips XA-G3 uses its WAIT input. The MB86976 will drive these signals from its $\overline{\text{READY}}$ output. For purposes of this discussion, each of the processorsí acknowledge signals will be referred to as $\overline{\text{READY}}$. All transaction mechanisms for the asynchronous processors are otherwise identical to those of the Intel 80c32. Although the asynchronous processors do not suffer from the latency mismatch constraint of the 80c32, the pseudo-DMA mode for direct accesses to the DRAM is preserved for these processors as well.

Interrupts

The MB86976 will communicate with the processor through the assertion of its interrupt pin, $\overline{\text{INT}}$. All interrupt sources will be maskable so that the processor may control the communications behavior. The 16-bit interrupt register is readable by the processor, and the INTerruptMASK register is both readable and writable. These are listed under the section "Interrupt and Interrupt Mask Register" on page 32.

At power-up, all the interrupts are masked. The processor must write to the INTerruptMASK register to enable desired interrupts. Every interrupt source is cleared by reading the interrupt register. Masked interrupts are asserted in the interrupt register, but not at the interrupt pin, allowing the processor to determine the interrupt status by polling without being interrupted.

Processor Interface Control Register Set

The Processor Interface contains 12 registers for controlling processor interaction with the MB86976, and for counting statistics. These are listed under the section "Interrupt and Interrupt Mask Register" on page 32.

DRAM Interface

The 256k x 16 page mode external DRAM is used for the Frame Buffer (transient frame storage) in addition to the Source Address Database. The Frame Buffer Database occupies the lower 224K (0 to (224K-1)) of the DRAM in the address space 00000h-37FFFh. The Source Address Database occupies the upper 32K (224K to (256K-1)) of the DRAM in the address space 38000h-3FFFFh.

The Frame Buffer

The Frame Buffer Database stores a maximum of 224 frames, each containing no more than 1023 16-bit words, or 2046 Bytes. Bits are stored msb to lsb.

Each table entry is comprised of a minimum of 2 and a maximum of 1023 16-bit words, as shown in table below.

Frame Buffer Database Format

bit[15:5]	bit[4:2]	bit[1:0]
length	destination	source
	data value(0)	
	data value(1)	
	...	
	data value(n)	

where:

Bit Field	Description
length	Number of bytes in the frame. E.g., a maximum length Ethernet frame would have 1518 bytes, including CRC; therefore, the length field would be set to 5EEh.
destination	000 = reject 001 = forward to processor 010 = forward to LAN port 100 = forward to WAN port 011 = forward to processor and LAN port 101 = forward to processor and WAN port 110 = forward to LAN port and WAN port 111 = in use
source	00 = processor 01 = LAN receive port 10 = WAN receive port 11 = unused
data value	Data field. In the event that the length of the frame is an odd number of bytes, the last eight bits of the frame are located at bit[7:0]

There are several types of FIFO within the MB86976 Some are "virtual" FIFOs existing *within* the Frame Buffer Database and are not *separate* physical entities, while others are physical register buffers between the internal LAN and WAN ports and the memory interface module. All these FIFOs are essentially transparent to the user, apart from some initialization of the DRAM-based FIFOs.

Source Address Database

The Source Address Database stores a maximum of 8K entries. Each DRAM-based 4 word entry is addressed using the 18 bit address format shown in the table below. The hash value is calculated using the algorithm in "Hashing Algorithm" on page 20.

bit[17:15]	bit[14:5]:	bit[4:2]	bit[1:0]
base address	hash value	entry select	word select

where:

Field	Description
base address	Fixed at 7h (DRAM address 224k).
hash value	Selects one of 1024 "buckets". Generated from the hash of the Destination Address.
entry select	Selects one of 8 SADB entries in the given "bucket".
word select	Selects one of four 16-bit words associated with the given SADB entry.

The format of Source Address Database entries is shown in the table below. Each entry is comprised of four 16-bit words: one header and three data words.

Source Address Database Entry Format

bit[15:6]	bit[5:3]	bit[2]	bit[1]	bit[0]
spare	destination	dynamic	valid	time-stamp
source address[40:47], source address[32:39]				
source address[24:31], source address[16:23]				
source address[8:15], source address[0:7]				

where:

Field	Description
spare	Unused at this time
destination	Frame destination value (see below)
dynamic	0 = this Source Address is a static entry 1 = this Source Address was auto-learned
valid	0 = entry is invalid 1 = entry is valid
timestamp	1-bit value indicating the relative age of the entry.

The action of the destination field value depends on the source of the frame, and other filtering criteria discussed in "Filtering" on page 18, but is summarized below.

destin value	Frame from LAN	Frame from WAN
none	WAN & autolearn	LAN
000	reject (default for auto-learned entries)	LAN
001	CPU & WAN	CPU
100	WAN	reject
101	CPU&WAN	CPU
111	(invalid)	(invalid)

Memory Controller

Arbitration

There are 14 memory access requestors performing a total of 14 memory related operations. In addition, there are two requestors, internal to the memory module, that request memory access autonomously for housekeeping functions.

Processor Interface

The processor receives and transmits frames to/from the LAN or WAN ports. In addition, the processor reads and/or writes to any location within the DRAM address space. The Processor Interface therefore contains four requestors: receive port, transmit port, DMA write, and DMA read.

LAN Interface

The LAN Interface reads and writes frames into and out of the Frame Buffer Database. The LAN Interface contains two requestors therefore: receive port and transmit port.

WAN Interface

The WAN Interface reads and writes frames into and out of the Frame Buffer Database. The WAN Interface contains two requestors therefore: receive port and transmit port.

Filtering State Machine

The Filtering State Machine reads and writes Source Address entries into and out of the Source Address Database. It contains three requestors: LAN Destination Address filter, LAN Source Address maintenance, and WAN Destination Address filter state machines.

Housekeeping Functions

Housekeeping functions are performed periodically to maintain valid memory structures. Unlike the other memory requestors, the housekeeping functions operate autonomously and require no intervention from external modules. The three housekeeping requestors include the DRAM initialization, DRAM refresh, and Source Address Database aging.

Priority

The memory arbiter prioritizes memory requests in the following order using a round-robin scheme:

1. LAN port frame reads/writes
2. WAN port frame reads/writes
3. LAN port Destination Address filtering
4. Source Address Database maintenance
5. WAN port Destination Address filtering
6. Processor frame reads/writes (Processor requests can be stalled indefinitely.)
7. Processor DMA
8. DRAM refresh
9. Source Address Database aging (Processing may occur only during the LAN interframe gap period.)

Filtering

A key function of the MB86976 is to minimize superfluous traffic between any of the LAN, WAN, and Processor ports. The device accomplishes this by filtering received frames from both the LAN and WAN ports, determining their intended destination, and rejecting or passing them accordingly. For example, a frame received from the LAN port whose intended destination is also on the LAN side of the bridge, is rejected. A frame received from the LAN port whose intended destination is on the processor or WAN side of the bridge is passed to the corresponding block.

Various levels of frame filtering are performed on all frames received from both the LAN and WAN receive ports unless configured otherwise by the corresponding control register bit(s). In every case, frame validity is checked by the appropriate interface first, and invalid frames (e.g. CRC failure, short frames, etc.) are rejected as needed.

LAN Filtering

Frames received from the LAN receive port are filtered via two mechanisms: the source address database (SADB) in the external DRAM and the user-defined look-up table (UDLT).

Source Address Database Filtering

The source address database filtering process accomplishes three objectives. The first is the implementation of an Ethernet bridge as specified in the 802.1d specification. The second is to provide a method to selectively forward or reject frames containing user-defined destination address(es). The third is to provide a conduit for the passage of frames from the LAN receive port to the processor. Only one filtering operation occurs per frame. Note that

the MB86976 deviates from the 802.1d specification in regards to bridging operations in that frames are not aged. There exists no mechanism that will timestamp incoming frames to ensure that the frames are not "obsolete" prior to transmission.

The following steps are performed for each new frame as it is received from the LAN receive port:

1. A10-bit hash value (see "Hashing Algorithm" on page 20) is calculated from the 48-bit Destination Address and is used as the pointer to one of 1024 database "buckets".
2. The corresponding "bucket" is linearly searched until a valid entry containing an identical 48-bit address is found or the end of the "bucket" is reached. Each bucket may contain up to 8 entries.
3. If a matching entry is found, the Destination field of the matching entry is captured.

   If no matching entry is found, the Destination field defaults to the WAN transmit port.

The result of this filtering, along with the results of the (UDLT) look-up table filtering process, are OR'd together in eliminating possible destinations of the frame. In the event that either the found Destination field or the look-up table filtering result is "000," the frame is rejected.

Auto-Learning

As new Source Addresses are encountered, they are automatically added or "learned" by the Source Address Database, if space permits. As entries age, they are periodically removed to make room for new entries.

Since auto-learned entries in the Source Address Database are marked with the Destination field set to "000", the next frame on the LAN that has this as a destination address will be rejected from crossing the WAN link. This serves to keep the frame local to the LAN network.

Static Address Entries

One of the consequences of the auto-learning of Source Addresses is that entries will periodically expire and need to be re-learned. If a user wishes to avoid this periodic purging and learning cycle, the Source Address Database can be configured by the firmware with static entries.

For example, if the user knows that a given device will always be on the LAN port of the bridge, its address may be loaded into the Source Address Database by the firmware. This static filter entry is identical to that created by the auto-learning process with one exception - the dynamic field is set to "0" to indicate a static entry. A static entry is exempt from the periodic purging of the database and persists until it is removed by the firmware.

Use the following steps to configure the Source Address Database with a static entry:

1. Generate a 10-bit hash value using the 48-bit Destination Address that is to be learned.
2. Use the hash value as the pointer to one of 1024 database "buckets" and write into the first free location that does not contain a static entry. The dynamic field must be set to "0," the valid field to "1," and the Destination field to the desired value. In the event that multiple static entries hash to the same value, the entire "bucket" may need to be read prior to finding an available free location

The filtering process using static entries is identical to the filtering process using dynamic entries. If an ethernet frame comes in with the same MAC source address as an existing static entry, the static entry is preserved and the incoming frame is not learned.

Static entries can contain any values for the fields in the header word, which allows configuring the Source Address Database to prohibit the transmission of frames to a given remote Destination Address. Similarily, to allow frames intended for the MB86976 to reach the processor, the MB86976's own MAC address (as assigned by the system developer) should be loaded by firmware into the source address database with the destination field set to 001 to forward the frame to the processor in addition to the WAN transmit port. An appropriate filtering entry in the UDLT is required to forward frames to the processor alone.

Additonal uses for static entries include allowing frames containing a multicast address or Bridge Protocol Data Unit (BPDU) frames to reach the processor. The desired multicast address or the BPDU address range 0180_C200_0000 to 0180_C200_000F should be loaded into the source address database with the destination field set to 001.

Source Address Database Maintenance

The Source Address Database requires periodic maintenance in the form of learning new entries, updating existing entries, and purging older entries.

Learning New Entries & Updating Existing Entries

As each new frame is received from the LAN port, the following

steps are performed automatically:

1. A 10-bit hash value is calculated from the 48-bit Source Address and is used as the pointer to one of 1024 database "buckets".
2. The corresponding "bucket" is linearly searched until an entry containing an identical, valid 48-bit address is found or the end of the "bucket" is reached.
3. If a valid matching entry is found, the timestamp field in the header of the matching entry is set to 1.

   If no matching entry is found, and at least one empty location exists, the Source Address is written into the database, setting both the timestamp and valid fields to 1. If an empty location does not exist, then exit.

Note that ethernet frames with errors (CRC, illegally large or small) do not update the SADB.

Purging Outdated Entries

To allow the MB86976 to operate in the "auto-learning" mode, a purging routine must be periodically executed to maintain a Source Address Database of the most recent entries. Not purging the Source Address Database may cause it to fill up and prevent learning of new addresses.

The default aging period is set (upon power-up) to 300 seconds with a resolution of 1 second in accordance with the 802.1d specification. This value may be overwritten by the processor at any time with any 9-bit unsigned value in SADB_Aging representing the aging period in seconds.

Every aging period (default 300 seconds), the following steps are performed:

1. Read the header word of the first entry in the Source Address Database.
2. If the dynamic bit is 0, proceed to the next entry.
3. If the value contained in the timestamp field is non-zero, then reset the timestamp bit to 0.
   If the value contained in the timestamp field is 0 and the valid bit is non-zero, then reset the valid bit to 0.
   If neither case exists, do nothing.
4. Repeat steps 2 and 3 for all header words in the Source Address Database.

It should be noted that due to the implementation of the packet aging function, there will exist situations where source address database entries may remain in the database for up to twice the specified aging period before being purged.

Hashing Algorithm

An exclusive-or polynomial function generates the 10-bit hash value used to index entries in the source address database. The hash value is as follows where n is for 0 through 9 for each one of the 10 bits:

$$hash(n) = a(n) \oplus a(n+10) \oplus a(n+20) \oplus a(n+30) \oplus a(n+40)$$

Bits 8 and 9 of the hash value are a function of 4 bits of the source address. All other bits are a function of 5 bits.

User-Defined Look-up Table Filtering

The user-defined look-up table filtering process enables generic filtering capabilities. By configuring the 24 16-bit user-defined table entries, a user is able to search for any sequence of bits within the first 32 bytes of a frame.

Through judicious partitioning of the 24 available table entries, up to 8 multiple filters can be constructed.

Note that unlike the Source Address Database filtering process, multiple filters will require multiple filtering operations per frame.

Look-up Table Format

Each table entry is comprised of three 16-bit words-header, data, and mask—as shown in table below. The header controls the word offset of the frame word to be matched, the id value of the filter string that this match relates to, the logic operation of the match, and the destination that the frame should receive if total matching occurs, as well as flags for validity, the beginning and end of the string.

Examples of UDLT lookup table entries are given in "UDLT Programming Examples" on page 40.

Table Entry Format

bit[15]	bit[14:10]	bit[9:8]	bit[7]	bit[6]	bit[5:3]	bit[2]	bit[1:0]
enable	word offset	destination	start	stop	filter id	unused	operation
data value							
mask value							

where:

Bit Field	Description
enable	0 = filter is disabled, 1 = filter is enabled
word offset	Number of 16-bit words, referenced from the start of the Destination Address, where the filter begins.
destination	00 = reject 01 = forward to CPU only 10 = forward to WAN transmit port only 11 = forward to both the CPU and WAN transmit port
start	1 = indicates the start of a new filter string, otherwise 0
stop	1 = indicates the end of a filter string, otherwise 0
filter id	Indicates to which filter string this filter word belongs.
operations	00 = "=", if "(comparand && mask) = data" comparison returns true 01 = "<", if "(comparand && mask) < data" comparison returns true 10 = ">", if "(comparand && mask) > data" comparison returns true 11 = invalid operation
data value	"Raw" data field. Bits which are to be masked are set to 0.
mask value	Mask which is applied to the comparand prior to the comparison. 0 = ignore this bit. 1 = use this bit in the comparison E.g., a mask value of FFFFh directs the comparator to use all 16 bits in the comparison operation.

Notes: 1. The look-up table must be loaded in a sequential manner, so that as one traverses it, the word offset field never decreases. Multiple filter strings must be interleaved to ensure this

2. The user must ensure that conflicting table entries are not present, e.g., two filter sequences that are both true for a given frame yet have conflicting Destination fields.

Look-up Table Processing

As each new frame is received from the LAN port, the following steps are performed:

1. Upon reception of the first word of a frame from the MAC core, the word offset counter is reset to 0. In addition, the 8 compare result registers are set to 0.
2. As each additional word is received, the word offset counter is incremented accordingly.
3. When the current table entry's word offset value equals the value of the word offset counter, the comparison is performed using the data, mask, and operation values associated with this table entry.
4. The result of the comparison is stored into one of 8 compare result registers as indexed by the value of the filter id field. The result of the comparison (1 = true, 0 = false) is logically ANDed with the current value of the compare result register and stored in its place.

   If the start field is 1, indicating the start of a filter sequence, only the result of the comparison is stored in the compare result register.
5. The table entry index value is then incremented to the next valid table entry and steps 3 and 4 repeated.
6. When the end of any filter sequence is reached (the stop field equals 1), the content of the corresponding compare result register is examined.

   If the entire filter string was found within the frame, the value is 1 and the content of the Destination field is used along with the results of the Destination Address filtering process to determine the destination of the frame.

   If the entire filter string was not found within the frame, the value is 0 and the Destination field is ignored.

WAN Filtering

Frames received from the WAN receive port are filtered with respect to the contents of the protocol field. This is in addition to and does not replace the frame validity checks performed by the WAN Interface module, i.e. CRC verification.

Protocol Field

When the WAN is in modes HDLC, PPP or PPP-LEX, the protocol field of frames received from the WAN receive port is used to determine the destination of the frame. Based on the value, the frame may be forwarded to the processor, LAN transmit port, or both. Protocol values come in three ranges and indicate the type of message being received:

Encapsulated data messages have a protocol field value between 0000h and 3FFFh, Network Control Protocol messages have a protocol field value between 8000h and BFFFh, and Link Control Protocol messages have a protocol field value between C000h and FFFFh.

Due to the unique range of values, only the most significant bit of the protocol field requires checking. If the protocol field MSB value is 0, the frame contains an encapsulated data message and is forwarded to the next stage of processing. If it is 1, the frame contains either a Network Control Protocol (NCP) or Link Control Protocol (LCP) message, and is written to the processor frame buffer as well as the LAN.

Encapsulated Data Messages

When new frames received from the WAN receive port are identified as encapsulated data messages, the following steps are performed:

1. A 10-bit hash value is calculated from the 48-bit Destination Address and is used as the pointer to one of 1024 Source Address database "buckets".
2. The corresponding "bucket" is linearly searched until a valid entry containing an identical 48-bit address is found or the end of the "bucket" is reached.
3. If a valid matching entry is found and the Destination field is set to "000", then the frame is forwarded to the LAN port.

   If a valid matching entry is found and the Destination field is set to "001", then the frame is forwarded to the processor.

   If a valid matching entry is found and the Destination field is set to "101", then the frame is forwarded to both the LAN transmit port and the processor.

   If no valid matching entry is found, the frame is forwarded to the LAN transmit port by default.

Network Control Protocol (NCP) and Link Control Protocol (LCP) Messages

All frames received from the WAN receive port identified as either a NCP or LCP message are unconditionally forwarded to the processor.

Processor Filtering Statistics

Filtering statistics are available for maintenance and performance monitoring. These are listed under the section headed Functional Control and Status Register.

System Features

In addition to the signals associated with the varous interfaces, clock and reset signals are provided, as well as LED outputs for diagnostics.

Clocks

The MB86976 is operated by a single system clock running at 40 MHz. This is sufficient to perform all processing functions for memory and filtering operations. It can recover off-chip input signals up to a 20 MHz frequency.

The only other clock domain on the MB86976 is wholly contained within the LAN MAC, which uses the transmit and receive clocks from the external transceiver for part of its operation. These clock domains are entirely isolated from the remainder of the MB86976, and all system functions that access the MAC do so in the 40 MHz system clock domain.

LED Outputs

The MB86976 provides seven status outputs intended to drive system diagnostic LEDs.

The MB86976 imports the Carrier Sense and Collision Detect signals from outputs generated by the external Ethernet transceiver, e.g., the Fujitsu MB86961A. The other status LEDs are generated by on-chip monitoring logic. Each LED output is pulse stretched such that the minimum assertion time for any of the LEDs is 25 ms, and will sink up to 12 mA.

Reset

The MB86976 has a single reset input, $\overline{\text{RESET}}$, and synchronizes this input and distributes it throughout the chip as a synchronous reset signal. The MB86976 makes the assumption that the system clock will be available during system reset. The mimimum period for $\overline{\text{RESET}}$ assertion is 2 SYSCLK periods.

Following reset, all on-board state machines are set to begin operation only after configuration by the processor. The Ethernet MAC will undergo both hardware and software reset operations when driven by the $\overline{\text{RESET}}$ pin.

The MB86976 will not begin operation until signalled by processor writes to the appropriate transmit/receive control registers.

JTAG Support

The MB86976 supports both scan testing and IEEE 1149.1 Boundary Scan. The Boundary Scan Controller is accessible through the
standard JTAG Interface (via TCK, TMS, TDI, TDO, and $\overline{\text{JTRST}}$ signals). Boundary Scan cells are present on all chip I/O, including monitor-only cells on the system clock pins.

The MB86976 Scan testing is independent of the JTAG interface and has a total of seven scan chains. They are accessible through an independent pin (TMS) which toggles between scan mode and system mode. The scan data in and out are multiplexed with existing pins and selected by $\overline{\text{SCANTESTMODE}}$ for this purpose. For details contact the factory.

Functional Control and Status Registers

The control and status registers on the MB86976 are accessed through direct register addresses x000H through x100H. Some of the registers listed in the Internal Register Address Map are not required/supported for normal operation. They are provided for test purposes at the factory and will not be discussed.

Type Description (TYPE)

The following legend of descriptions applies to the type column of the register bit description tables:

R: Readable bit.
W: Writable bit.
C: Clears associated status bit and/or interrupt when 1 is written: no effect when 0 is written.
N: Not used: reserved; write only 0 when written.
0/1: Initial state after hardware reset.

Internal Register Address Map

Listed by numerical address offset. Groups refer to the bit map tables that follow, and are interleaved in this map. .

System Addr. Offset (Hex)	Registers				Initial Value
	Symbol	Group	Reg. Size (Bits)	Description	
000	MAC_Control	LAN	16	MAC basic operating mode Control.	0000
004	CAM_Control	LAN	8	CAM Control, types of MAC addresses to accept.	00
008	LAN_TxControl	LAN	16	LAN Transmit option Control.	0000
00C	MAC_TxStatus	LAN	16	LAN Transmit Status, collision count, interrupt, carrier sense.	0000
010	LAN_RxControl	LAN	16	LAN Receive option Control, long/short, no CRC, interrupt...	0000
014	MAC_RxStatus	LAN	16	LAN Receive Status, long/short, CRC, alignment, etc.	0000
034	ALIGN_CNT	LAN	16	Counts LAN packets received with alignment error.	0000
038	CRC_CNT	LAN	16	Counts LAN packets received with CRC error.	0000
03C	MISS_CNT	LAN	16	Counts LAN packets received with system problem error.	0000
040	INTERRUPT	CPU	16	Read the interrupt sources.	0000
042	INTerruptMASK	CPU	16	Mask interrupt sources.	0000
048	FrameBuffer	CPU	8 or 16	Read/Write FIFOs for frames CPU <-> MB86976.	00/00
04A	End_of_Frame	CPU	1	Asserted for last transfer of sent frame.	0
04C	FrB_Write_GO	CPU	1	Transfer current send frame words to MB86976 FrameBuffer.	0
04E	FrB_Read_GO	CPU	1	Transfer current receive frame words from Frame Buffer.	0
050	DMA_StartAddress	CPU	24	Register to hold the start address of DMA transfer (18 bits).	00 0000
054	DMA_Length	CPU	4	Register to hold length, in words, of DMA transfer.	0
056	DMA_Read_GO	CPU	1	Asserted to begin DMA read operation.	0
05A	DMA_Write_GO	CPU	1	Asserted to begin DMA write operation.	0
05E	DMA_FIFO	CPU	8 or 16	Top of Data FIFO for DMA transactions.	00/00
068	HDLC_Address	WAN	8	HDLC Address Field, common to all protocols.	FF
06A	HDLC_Control	WAN	8	HDLC Control Field, common to all protocols.	03
06C	PPP_Protocol	WAN	16	First & Second byte of PPP protocol field.	4100
06E	LEX_FIZO	WAN	8	PPP-LEX FIZO Field.	00
070	LEX_MacType	WAN	8	PPP-LEX MACTYPE Field.	01
072	WAN_TxControl	WAN	8	Control register configures the WAN transmit operation.	00
074	WAN_RxControl	WAN	8	Control register to configure the WAN receive operation.	00
076	WAN_TxStatus	WAN	8	WAN Transmit Status register to report on the last result.	00
078	WAN_RxStatus	WAN	8	WAN Receive Status register to report on the last result.	00
07A	WAN_RxByte_Cnt	WAN	32	WAN Receive Byte Count, every byte received from WAN.	0000 0000
07E	WAN_RxFrame_Cnt	WAN	16	WAN Receive Frame Count, every frame received.	0000
080	WAN_RxErr_Cnt	WAN	16	WAN Receive Frame Error Count, every frame with an error.	0000
082	WAN_RxOverLength	WAN	16	Counts every frame exceeding the WAN Max, 1526 bytes.	0000
084	WAN_TxByte_Cnt	WAN	32	WAN Transmit Byte Count, every transmitted byte.	0000 0000
088	WAN_TxFrame_Cnt	WAN	16	WAN Transmit Frame Count, every frame transmitted.	0000
090	SADB_Aging	FBFLT	9	SADB Aging Period in Seconds (initial 300 decimal).	12C
092	FILTer_Control	FBFLT	8	Enable SADB, UDLT filtering.	07

System Addr. Offset (Hex)	Registers				Initial Value
	Symbol	Group	Reg. Size (Bits)	Description	
094	CPU_Reserve	FBFLT	8	Number of Frame Buffer spaces to reserve for the CPU port.	08
096	LAN_Reserve	FBFLT	8	Number of Frame Buffer spaces to reserve for the LAN port.	08
098	WAN_Reserve	FBFLT	8	Number of Frame Buffer spaces to reserve for the WAN port.	08
09A	Learned_SA	FBFLT	16	Counts new SA received and added to database.	0000
09C	NOT_Learned_SA	FBFLT	16	New SA recvd, not added to SADB, no free space in bucket.	0000
09E	UpDated_SA	FBFLT	16	Previously learned SA received - time stamp updated.	0000
0A2	CPU_Drop_Frames	FBFLT	16	CPU frames discarded due to Frame Buffer overflow.	0000
0A4	LAN_Drop_Frames	FBFLT	16	LAN frames discarded due to Frame Buffer overflow.	0000
0A6	WAN_Drop_Frames	FBFLT	16	WAN frames discarded due to Frame Buffer overflow.	0000
0A8	LAN_Reject_Frames	FBFLT	16	LAN Frames Rejected due to Filtering results.	0000
0AA	WAN_Reject_Frames	FBFLT	16	WAN Frames Rejected due to Filtering results.	0000
0AC	CPU_Rx_Frames	FBFLT	16	Total number of frames received by the CPU.	0000
0AE	WAN_Rx_Frames	FBFLT	16	Total number of frames received by the WAN.	0000
0B0	CPU_Tx_Frames	FBFLT	16	Total number of frames transmitted by the CPU.	0000
0B2	WAN_Tx_Frames	FBFLT	16	Total number of frames transmitted by the WAN.	0000
0B4	CPU_HiWtr_Level	FBFLT	16	CPU Tx FIFO high watermark (cleared on read).	00
0B6	LAN_HiWtr_Level	FBFLT	16	LAN Tx FIFO high watermark (cleared on read).	00
0B8	WAN_HiWtr_Level	FBFLT	16	WAN Tx FIFO high watermark (cleared on read).	00
0BA	MASTER_HiWtr_Level	FBFLT	16	Sum of the LAN, WAN and CPU HiWtr_Levels.	00
0BC	DRAM_Init_Done	CPU	1	Zeroed when DRAM data initialized after power-up.	00
0C0	JTAG_ID	CPU	32	JTAG ID register for part version determination (Read Only).	2F00 2009
0DC	LAN_TxStatus	LAN	16	LAN Transmit Status. Loaded from MAC_TxStatus.	0000
0DE	LAN_Opt_Set	LAN	3	Write-only, options for the MAC. Must be programmed to 03.	0
0E0	LAN_RxByte_Cnt	LAN	32	LAN Receive Bytes Count, including bad frames.	0000 0000
0E4	LAN_Rx_Frames	LAN	16	LAN Receive Frames Count, including bad frames.	0000
0E6	LAN_Rx_Broadcast	LAN	16	LAN Receive Multicast frames (DA LSb = 1) Count.	0000
0E8	LAN_Rx_Error	LAN	16	LAN Receive Error Frames (CRC, align, size) Count.	0000
0EA	LAN_Rx_RUNTS	LAN	16	LAN Receive Runt Frames & Collison Fragments Count.	0000
0EC	LAN_Rx_CRC	LAN	16	LAN Receive bad CRC Count.	0000
0EE	LAN_RxOverLength	LAN	16	LAN Receive Over Length Frames Count (over 1518 bytes).	0000
0F2	LAN_RxStatus	LAN	16	LAN Receive Status bits.	0000
0F4	LAN_TxByte_Cnt	LAN	32	LAN Transmit Byte Count (includes aborts).	0000 0000
0F8	LAN_Tx_Frames	LAN	16	LAN Transmit Frames Count (includes aborts).	0000
0FA	LAN_Tx_Error	LAN	16	LAN Transmit Error Count (aborted due to error).	0000
0FC	LAN_TxExCollision	LAN	16	LAN Transmit Excessive (>16) Collision Aborted Count.	0000
0FE	LAN_Tx_Broadcast	LAN	16	LAN Transmit Multicast frames (DA LSb = 1) Count.	0000
100	Lookup_Table	FBFLT	72*16	User-programmable lookup table (UDLT) filter.	

LAN Interface Register Bit Maps (Group LAN)

MAC_Control (Register 00)

Bit	Symbol	Type	Description
0	HALT_REQ	RW0	Halt Request: Stops transmission and reception after completion of any current packets.
1	HALT_IMM	RW0	Halt Immediate: Stop transmission and reception of LAN traffic immediately.
2	RESET	RW0	Software Reset: Reset all Ethernet controller state machines and FIFOs within the MAC core of the MB86976, (wait 2 sysclk cycles before performing any other operation).
3	FULL_DUP	RW0	Full Duplex: Allows transmission to begin while reception is occurring.
4	MAC_LOOP	RW0	MAC Loopback: Cause transmission signals to be presented as input to the receive circuit without leaving the controller.
6:5	CONN	RW01	Connection Mode: 00= Automatic, 01=force 10-Mbit/s endec, 1X = Invalid.
7	0	N	Reserved: Unassigned.
8	AlignRoll	R0	Alignment Roll: Alignment error counter rolled over.
9	CRCRoll	R0	CRC Roll: CRC error counter rolled over.
10	MissRoll	R0	Missed error counter rolled over. (Read only)
11	EnAlignRoll	RW0	Enable Alignment Roll: Interrupt when alignment error counter rolls over.
12	EnCRCRoll	RW0	Enable CRC Roll: Interrupt when CRC error counter rolls over.
13	EnMissRoll	RW0	Enable Missed Rolled: Interrupt when missed error counter rolls over.
14	0	N	Reserved: Unassigned.
15	Link	R	Buffered signal on the LINK pin. (Read only)

The MAC control register provides global control and status information for the MAC embedded within the MB86976. The MissRoll, CRCRoll and AlignRoll bits are status bits. All the others are control bits.

The register affects both transmission and reception. Transmit and receive can also be controlled individually. This register may be written after power up to select customized operating features. It does not need to be written or read during normal operation.

After reset is complete, the controller clears the reset bit. Some PHYs (ethernet transceivers) may not support full duplex. Mac Loopback overrides the full duplex bit. Some 10-Mb/s PHYs may signal LINK to indicate a different status condition. In automatic connect mode, it will not send until there is receive activity on the 10 Mb/s interface which will then select the 10 Mb/s endec.

For MB86976 operation: The Connection mode field in this register, bits 6 and 5, must be set to 01 - Force 10Mbit/s.

CAM_Control (CAM Control. Register 04)

Bit	Symbol	Type	Description
0	StationAcc	RW	Accept any packet with a unicast station address
1	GroupAcc	RW	Accept any packet with a multicast-group address.
2	BroadAcc	RW	Accept any packet with a broadcast address.

For MB86976 operation: Set the CAM Control register to 07 hex - Accept unicast, multicast and broadcast addresses - i.e. promiscuous mode. The CAM referred to by this register is for an internal test condition only.

LAN_TxControl (LAN Transmit Control, Register 08)

Bit	Symbol	Type	Description
0	Tx_EN	RW0	Transmit Enable: If zero, stop transmission immediately.
1	Tx_Halt	RW0	Transmit Halt Request: Halt transmission after completing any current packet.
2	No_Pad	RW0	Suppress Padding: Do not generate pad bytes for packets with less than 64 bytes.
3	No_CRC	RW0	Suppress CRC: Do not add the CRC at the end of a packet.
4	0	N	Reserved: Write 0
5	No_Def	RW0	No Defer: Disable the defer counter and excessive deferral checking.
7:6	0	N	Reserved: Unassigned.
8	En_Under	RW0	Enable Underrun: Interrupt if the MAC transmit FIFO becomes empty during transmission.
9	En_Ex_Defer	RW0	Enable Excessive Deferral: Interrupt if the MAC defers for MAX_DEFERRAL time = 3.2768 ms for 10 Mbits/s
10	En_No_Carr	RW0	Enable No Carrier: Interrupt if carrier sense is not detected during the transmission of a packet.
11	En_Ex_Coll	RW0	Enable Excessive Collision: Interrupt if 16 collisions occur in the same packet.
12	En_Late_Coll	RW0	Enable Late Collision: Interrupt if a collision occurs after 512 bit times (64 byte times).
13	En_Tx_Par	RW0	Enable Transmit Parity: Interrupt if the MAC transmit FIFO has an internal parity error.
14	En_Compl	RW0	Enable Completion: Interrupt when the MAC transmits or discards one packet.
15	0	N	Reserved: Unassigned.

This register affects the way in which Ethernet traffic is being transmitted on the LAN interface.To receive an interrupt after each packet, set the enable completion and all the MAC error enable bits and clear bit 12, MacTxInterrupt, of the INTerruptMASK, reg 42h. Interrupts may also be enabled only for specific conditions.

MAC_TxStatus (Register 0C)

Bit	Symbol	Type	Description
3:0	TxColl	R0	Count of the collisions in transmitting a single packet. If 16 collisions occur, TxColl will be zero, and ExColl is set.
4	ExColl	R0	Set if 16 collisions occur in the same packet. Transmission skipped.
6:5		N	Reserved
7	IntTx	R0	Set if transmission of packet caused any interrupt condition (includes En_Compl).
8	Underrun	R0	MAC transmit FIFO becomes empty during transmission.
9	Ex_Defer	R0	MAC defers for MAX_DEFERRAL: = 3.27680 ms for 10 Mbits/s
10	NO_CarrS	R0	Carrier sense is not detected during the transmission of a packet (from SFD to CRC).
11	Tx10Stat	R0	= 1 if packet was transmitted via the 10-Mb/s interface, = 0 invalid internal mode.
12	LateColl	R0	A collision occurs after 512 bit times (64 byte times).
13	TxPar	R0	MAC transmit FIFO has detected a parity error (internal error, may clear Tx_En).
14	Tx_Compl	R0	Completion: MAC transmits or discards one packet.
15	TxHalted	R0	Transmission was halted by clearing Tx_En or setting TxHalt.

The transmission status flags are set whenever the corresponding event occurs. In addition, an interrupt is generated if the corresponding enable bit in the transmit control register is set and bit 12, MacTxInterrupt, of the INTerruptMASK, reg 42h, is cleared.

The low order five bits can be read and masked as a single collision count, i.e when ExColl is 1, TxColl is 0. If TxColl is non-zero, then ExColl is 0.

LAN_RxControl (LAN Receive Control Register 10)

Bit	Symbol	Type	Description
0	Rx_EN	RW0	Receive Enable: If zero, stop reception immediately.
1	Rx_Halt	RW0	Receive Halt Request: Halt reception after completing any current packet.
2	Long_EN	RW0	Long Enable: Allow reception of frames longer than 1518 bytes
3	Short_EN	RW0	Short Enable: Allow reception of frames shorter than 64 bytes.
4	Strip_CRC	RW0	Strip CRC Value: Check the CRC, but strip it from the message.
5	0	N	Reserved: Unassigned.
6	Ignore_CRC	RW0	Ignore CRC Value: Do not check the CRC.
7	0	N	Reserved: Unassigned.
8	En_Align	RW0	Enable Alignment: Interrupt upon receipt of a packet whose length in bits is not a multiple of eight, and whose CRC is invalid.
9	En_CRC_Err	RW0	Enable CRC Error: Interrupt upon receipt of a packet whose CRC is invalid.
10	En_OverFlow	RW0	Enable Overflow: Interrupt upon receipt of a packet when the MAC receive FIFO is full.
11	En_Long_Error	RW0	Enable Long Error: Interrupt upon receipt of a frame longer than 1518 bytes, unless the long enable bit is set.
12	0	N	Reserved: Unassigned.
13	En_Rx_Par	RW0	Enable Receive Parity: Interrupt if the MAC receive FIFO has an internal parity error.
14	En_Good	RW0	Enable Good: Interrupt upon receipt of a packet with no errors.
15	0	N	Reserved: Unassigned.

This register affects the way in which Ethernet traffic is being received on the LAN interface. To receive an interrupt after each packet, set the good enable and all the error enable bits, and clear bit 13, MacRxInterrupt, of the INTerruptMASK, reg 42h. Interrupts may also be enabled only for specific conditions.

It is very strongly recommended that neither LongEn nor ShortEn be asserted (set to one), as correct operation of the MB86976 in all cases then cannot be guaranteed.

* The frame lengths above do not include preamble and Start Frame Delimiter (SFD).

MAC_RxStatus (MAC Receive Status Register 14).

Bit	Symbol	Type	Description
5:0		N	Reserved
6	IntRx	RW0	Set if reception of packet caused any interrupt condition (includes En_Good).
7	Rx10Stat	RW0	= 1 if packet was received via the 10-Mb/s interface, = 0 invalid internal mode.
8	AlignErr	RW0	Frame length in bits was not a multiple of eight and the CRC was invalid.
9	CRCErr	RW0	CRC at end of packet did not match computed value.
10	Overflow	RW0	The MAC receive FIFO was full when it needed to store a received byte.
11	LongErr	RW0	Received a frame longer than 1518 bytes.* Not set if the En_Long_Error bit in the LAN_RxControl register, register 10, is set.
12		N	Reserved
13	RxPar	RW0	MAC receive FIFO has detected an (internal) parity error.
14	Good	R0	Successfully received a packet with no errors. If En_Good = 1 (LAN_RxControl, Reg. 10), an interrupt is generated on each packet received successfully
15	RxHalted	RW0	Reception interrupted by clearing RxEn or setting RxHalt.

The receive status flags are set whenever the corresponding event occurs. Once set, a flag stays set until another packet arrives. In addition, an interrupt is generated if the corresponding enable bit in the LAN_RxControl register is set, and bit 13, MacRxInterrupt, of the INTerruptMASK, reg 42h, is clear.

A MAC receive parity error sets RxPar, if interrupt is enabled. Note that this is an internal error, and will never assert under normal conditions. Software is responsible for separating alignment, CRC, and frame too long errors, and reporting them correctly as management information. The MAC will tolerate up to a maximum of 2 bits of dribble received after an otherwise good frame. Any frame with more dribble is an alignment error.

ALIGN_CNT (Alignment Count. Register 34)

Bit	Symbol	Type	Description
15:0	ALIGN_CNT	RC0	The number of packets received with Alignment error. This counter increments at the end of reception if the receive block detects an Alignment error.

CRC_CNT (CRC Count. Register 38)

Bit	Symbol	Type	Description
15:0	CRC_CNT	RC0	The number of packets received with CRC errors. It increments if the receive block detects a CRC error at the end of reception. If the received packet contains other errors, such as Alignment, this counter does not increment.

MISS_CNT (Missed Error Count. Register 3C)

Bit	Symbol	Type	Description
15:0	MISS_CNT	RC0	Counts the number of valid packets which are rejected by the MAC unit because the MAC receive FIFO overflows, a parity error occurs, or the Receive Enable bit (Rx_EN) is cleared. This count excludes packets the SADB filter rejects.

These system error count registers provide a count of packets discarded due to various types of errors. Together with status information for packets transmitted and received, these counters provide the information needed for station management. It is the responsibility of software to maintain a larger global count.

The missed error count register rolling over from 7FFF h to 8000 h sets the Missed Roll bit in the MAC control register. It also generates an interrupt if the Enable Missed Roll bit is set.

If station management software wants more frequent interrupts, the missed error count register can be set to a value closer to the roll over value of 7FFF. For example, setting the register to 7F00 would provide for an interrupt after counting 256 occurrences.

LAN_TxStatus (LAN Transmit Status. Register DC)

Bit	Symbol	Type	Description
15:0	LAN_TxStatus	R0	LAN Transmit Status: Loaded from MAC_TxStatus.

This register is loaded on completion of a LAN frame transmission from MAC_TxStatus (Register 0C). The bits are exactly the same as in that register, but are retained stable until the completion of the next transmitted frame.

LAN_Opt_Set (LAN Option Set. Register DE)

Bit	Symbol	Type	Description
15:0	LAN_Opt_Set	R0	LAN Option Set: Used as a control signal to the MAC core within the MB86976. (Program to 0003h for normal operation)

LAN_RxByte_Cnt (LAN Receive Byte Count. Register E0)

Bit	Symbol	Type	Description
32:0	LAN_RxByte_Cnt	R0	LAN Receive Bytes: Number of bytes received from the LAN port after the preamble and SFD. If the CRC bytes are stripped, those bytes are not counted by LAN_RxByte_Cnt.

LAN_Rx_Frames (LAN Receive Frames. Register E4)

Bit	Symbol	Type	Description
15:0	LAN_Rx_Frames	R0	LAN Receive Frames: Total number of frames received on the LAN receive interface.

LAN_Rx_Broadcast (LAN Receive Broadcast. Register E6)

Bit	Symbol	Type	Description
7:0	LAN_Rx_Broadcast	R0	LAN Receive Broadcast: Increments when the LSB of the destination address of a frame that is received on the LAN receive interface is one. Includes bad frames.

LAN_Rx_Error (LAN Receive Errors. Register E8)

Bit	Symbol	Type	Description
15:0	LAN_Rx_ERR	R0	LAN Receive Errors: Counts when a packet is received with a CRC or alignment error, or is too large.

LAN_Rx_RUNTS (LAN Receive Runts. Register EA)

Bit	Symbol	Type	Description
15:0	LAN_Rx_RUNTS	R0	LAN Receive Runts: Counts Rx runt frames/collision fragments (frames less than 64 bytes, or less than 14 bytes if ShortEn in reg 10 is set).

LAN_Rx_CRC (LAN Receive CRC. Register EC)

Bit	Symbol	Type	Description
15:0	LAN_Rx_CRC	R0	LAN Receive CRC: Counts bad CRC frames received on the LAN.

LAN_RxOverLength (LAN Receive Over Length. Register EE)

Bit	Symbol	Type	Description
15:0	LAN_RxOverLength	R0	LAN Receive OverLength: Number of frames received on the LAN receive interface greater than 1518 bytes.

LAN_RxStatus (LAN Receive Status. Register F2)

Bit	Symbol	Type	Description
5:0		N	Reserved: Unassigned.
6	IntRx	R0	Set if reception of packet caused any interrupt condition (includes En_Good).
7		N	Reserved: Unassigned.
8	AlignErr	R0	Alignment Error: Frame length in bits was not a multiple of eight and the CRC was invalid.
9	CRCErr	R0	CRC at end of packet did not match computed value.
10	Overflow	R0	The MAC receive FIFO was full when it needed to store a received byte.
11	LongErr	R0	Long Error: Received a frame longer than 1518 bytes, excluding preamble and SFD.
12		N	Reserved: Unassigned.
13	RxPar	R0	MAC receive FIFO has detected an (internal) parity error.
14	Good	R0	Good Received: Successfully received a packet with no errors.
15	RxHalted	R0	Reception Halted: Reception interrupted by user clearing RXEN or setting RXHALT.

(loaded from MAC_RxStatus, Register 14)

LAN_TxByte_Cnt (LAN Transmit Byte Count. Register F4)

Bit	Symbol	Type	Description
31:0	LAN_TxByte_Cnt	R0	LAN Transmit Bytes: Total number of bytes transmitted on the LAN transmit interface.

LAN_Tx_Frames (LAN Transmit Frames. Register F8)

Bit	Symbol	Type	Description
15:0	LAN_Tx_Frames	R0	LAN Transmit Frames: Total number of frames transmitted on the LAN transmit interface.

LAN_Tx_Error (LAN Transmit Errors. Register FA)

Bit	Symbol	Type	Description
15:0	LAN_Tx_Error	R0	LAN Transmit Errors: Counts when a packet being transmitted on the LAN has excessive collision, excessive deferral, late collision or (internal) FIFO parity error or under-run.

LAN_TxExCollision (LAN Transmit Excessive Collisions. Register FC)

Bit	Symbol	Type	Description
15:0	LAN_TxExCollision	R0	LAN Transmit Exc Collision: Transmit aborts due to excessive (more than 15) collisions.

LAN_Tx_Broadcast (LAN Transmit Broadcast. Register FE)

Bit	Symbol	Type	Description
15:0	LAN_Tx_Broadcast	R0	LAN Transmit Broadcast: Counts Multicast packets transmitted on the LAN

Note LAN_RxByte_Cnt, LAN_RxFrames and LAN_Rx_Broadcast include counts for appropriate undersized/fragment frames and oversized frames, LAN_TxByte_Cnt and LAN_Tx_Broadcast include collisions and aborted frames.

Processor Interface Registers (Group PROC)

Interrupt and Interrupt Mask Register

At power-up, all the interrupts are masked. The processor must write to the INTerruptMASK register to enable desired interrupts. Every interrupt source is cleared by reading the interrupt register. Masked interrupts are asserted in the interrupt register, but not at the interrupt pin, allowing the processor to determine the interrupt status by polling without being interrupted.

INTERRUPT (Register 40)

Bit	Symbol	Type	Description
3:0	0	N	Reserved: Unassigned.
4	Buffer_Full	RC0	Packet was discarded due to lack of space in the Frame Buffer.
5	0	N	Reserved: Unassigned.
6	DMA_Done	RC0	DMA read/write complete
7	0	N	Reserved: Unassigned.
8	Proc_Bnd_Pkt	RC0	Processor-Bound Packet: Informs the processor that at least one packet marked for the processor is in the Frame Buffer.
9	Proc_Buf_Full	RC0	Processor Buffer Full: Informs the processor that the buffer it wishes to write to is full.
10	Proc_Wr_Compl	RC0	Processor Write Complete: Informs the processor that the current outstanding write action has completed, and that it may initiate another write to the Frame Buffer if necessary.
11	0	N	Reserved: Unassigned.
12	MAC_Tx_INT	R0	Mac Transmit Interrupt: MAC transmit interrupt asserted
13	MAC_Rx_INT	R0	MacTransmit Interrupt: MAC receive interrupt asserted
14	LAN_LINK_Chng	R0	LAN Link change: Asserted when the LINK pin changes states.
15	0	N	Reserved: Unassigned.

INTerruptMASK (Interrupt Mask. Register 42)

Bit	Symbol	Type	Description
15:0	INTerruptMASK	RW0	Values written into this register will mask the corresponding values written into the Interrupt Register (Register 40).

FrameBuffer Frame Buffer (Register 48)

The Frame Buffer is a 16 bit by 4 words deep FIFO, between the processor interface and the DRAM interface. Frames from the LAN or the WAN, or both ports destined for the processor will transition through this Frame Buffer, and likewise, frames destined for either the LAN or WAN, or both, ports from the processor will transition through this Frame Buffer.

Bit	Symbol	Type	Description
7:0	FrameBuffer	RW0	Frame Buffer : This register is the top of a 8/16 bit by 4 word deep FIFO between the CPU and DRAM.
15:8	FrameBuffer	RW0	If BYTEWORD pin is Hi, upper byte of FrameBuffer word.

End_of_Frame (Register 4A)

Bit	Symbol	Type	Description
0	End_of_Frame	RW0	End of Frame: Writing to this bit will indicate to the processor that no more writing of data to the FrameBuffer, register 48, will take place.
15:1		N	Reserved: Unassigned.

FrB_Write_GO (FRAME BUFFER WRITE GO) (Register 4C)

Bit	Symbol	Type	Description
0	FrB_Write_GO	RW0	Frame Buffer Write Go: Writing to this bit will move the data from the FrameBuffer, residing at location 48, into the DRAM.
15:1		N	Reserved: Unassigned.

FrB_Read_GO (FRAME BUFFER READ GO) (Register 4E)

Bit	Symbol	Type	Description
0	FrB_Read_GO	RW0	Frame Buffer Read Go: When asserted, indicates that a new set of words of the current receive frame, that is residing in the Frame Buffer, are available for reading by the Processor.
15:1		N	Reserved: Unassigned.

DMA_StartAddress (Register 50)

Bit	Symbol	Type	Description
23:0	DMA_StartAddress	RW0	Start address of DMA transfer.
31:24		N	Reserved: Unassigned.

DMA_Length (Register 54)

Bit	Symbol	Type	Description
3:0	DMA_Length	RW0	Length of DMA transfers in words
16:4	0	N	Reserved: Unassigned.

DMA_Read_GO (Register 56)

Bit	Symbol	Type	Description
0	DMA_Read_GO	RW0	DMA Read Go: When asserted, will begin the DMA read operation.
15:1	0	N	Reserved: Unassigned.

DMA_Write_GO (Register 5A)

Bit	Symbol	Type	Description
0	DMA_Write_GO	RW0	Write Go: When asserted, DMA write operations will begin.
15:1	0	N	Reserved: Unassigned.

DMA_FIFO (Register 5E)

Bit	Symbol	Type	Description
7:0	DMA_FIFO	RW0	This register is the top of a 8/16 bit Data FIFO between the CPU and DRAM for DMA transactions.
15:8	0	N	If BYTEWORD pin is Hi, upper byte of DMA_FIFO word.

WAN Interface Registers (Group WAN)

HDLC_Address (Register 68)

Bit	Symbol	Type	Value	Description
7:0	HDLC_Address	RW	FFh	Stores the data link control address field common to HDLC, PPP and PPP-LEX .

HDLC_Control (Register 6A)

Bit	Symbol	Type	Value	Description
7:0	CONTROL	RW	03h	Stores the data link control address field common to HDLC, PPP and PPP-LEX .

PPP_Protocol (Register 6C)

Bit	Symbol	Type	Value	Description
7:0	PPP_Protocol1	RW	00h	First byte of PPP and PPP-LEX protocol field.
15:8	PPP_Protocol2	RW	41h	Second byte of PPP and PPP-LEX protocol field.

LEX_FIZO (Register 6E)

Bit	Symbol	Type	Description
3:0	PAD	RW0	PAD:
4	0	RW0	0: Reserved - Write 0.
5	Z	RW0	Z: IEEE 802.3. Pad must be zero filled to the minimum size.
6	I	RW0	I: Set this bit if the LAN ID is present. Else set, to 0 for LAN ID not present.
7	F	RW0	F: Set this bit if the LAN FCS is present. Else, set to 0 for LAN FCS not present.

LEX_MacType (Register 70)

Bit	Symbol	Type	Description
7:0	MAC_TYPE	RW0	MAC Type: PPP = LEX MAC type field. Default = 01h (IEEE 802.3 Ethernet with canonical addresses.)

WAN_TxControl (WAN Transmit Control. Register 72)

Bit	Symbol	Type	Description
0	Dis_WAN_Tx_FCS	RW0	Disable the WAN Frame Check Sequence (FCS): When asserted, disables the WAN Frame Check Sequence.
1	Dis_Zero_Bit_Ins	RW0	Disable Zero Bit Insertion: When asserted, disables the insertion of a zero bit after 5 consecutive one's have been transmitted on the WAN transmit interface.
3:2	WAN_Tx_Mode	RW00	Transmit mode . 00 - Compressed HDLC, 01 - HDLC, 10 - PPP, 11 - PPP-LEX.
4	WAN_Tx_ENable	RW0	WAN Transmit Enable: When asserted, enables transmission on the WAN interface.
7:5	0	N	Reserved: Unassigned.

The way in which a frame is transmitted on the WAN transmit interface, is determined by the bits in this register.

When the WAN is disabled due to bit 4 deasserted above, the WAN port TXD will emit a continuous sequence of logic 'ones' (i.e. the idle pattern).

WAN_RxControl (WAN Receive Control. Register 74)

Bit	Symbol	Type	Description
0	Dis_WAN_Rx_FCS	RW0	Disable the WAN Frame Check Sequence (FCS) verification
1	Dis_Zero_Bit_Del	RW0	Disables the removal of a zero bit after 5 consecutive one's have been received on the WAN receive port.
3:2	WAN_Rx_Mode	RW00	Reception mode . 00 - Compressed HDLC, 01 - HDLC, 10 - PPP, 11 - PPP-LEX.
4	WAN_Rx_ENable	RW0	WAN Receive Enable: When asserted, enables reception on the WAN interface.
7:5	0	N	Reserved: Unassigned.

The way in which a frame is received on the WAN receive interface, is determined by the bits in this register.

WAN_TxStatus (WAN Transmit Status. Register 76)

Bit	Symbol	Type	Description
1:0	Tx_Mode	R00	Packet type (0 - compressed HDLC, 1 - HDCL, 2 - PPP, 3 - PPP-LEX).
7:2	0	N	Reserved: Unassigned.

WAN_TxStatus is a read-only register. It will retain its value until the completion of the next transmission event, at which point it will be updated with the new status information

WAN_RxStatus (WAN Receive Status. Register 78).

Bit	Symbol	Type	Description
1:0	Rx_Mode	R00	Packet type (0 - compressed HDLC, 1 - HDCL, 2 - PPP, 3 - PPP-LEX).
2	Rx_EOF	R0	Receive End of Frame: When asserted, this bit indicates the end of a frame has been received on the WAN receive interface.
3	Rx_Error	R0	Indicates a reception error occurred on a frame on the WAN receive interface. The frame will be discarded.
4	Rx_Short	R0	Indicates that a short packet was received on the WAN receive interface.
5	Rx_Seven_ONEs	R0	Abort detected: Indicates that more than 7 consecutive ones were received on the WAN receive interface.
7:6		N	Reserved: Unassigned.

WAN_RxStatus is a read only register. It will retain its value until the completion of the next reception event, at which point it will be updated with the new status information.

WAN_RxByte_Cnt (WAN Receive Byte Count. Register 7A)

Bit	Symbol	Type	Description
31:0	WAN_RxByte_Cnt	R0	Count of the total number of bytes received on the WAN receive interface.

WAN_RxFrame_Cnt (WAN Receive Frame Count. Register 7E)

Bit	Symbol	Type	Description
15:0	WAN_RxFrame_Cnt	R0	Total number of frames, including good and bad, received on the WAN interface.

WAN_RxErr_Cnt (WAN Receive Frame Error Count. Register 80)

Bit	Symbol	Type	Description
15:0	WAN_RxErr_Cnt	R0	Count of the total number of frames received with errors on the WAN interface.

WAN_RxOverLength (WAN Receive Over Flow Count. Register 82)

Bit	Symbol	Type	Description
15::0	WAN_RxOverLength	R0	Count of the number of frames received on the WAN interface larger than 1526 bytes.

WAN_TxByte_Cnt (WAN Transmit Byte Count. Register 84)

Bit	Symbol	Type	Description
31:0	WAN_TxByte_Cnt	R0	Count of the total number of bytes transmitted on the WAN transmit interface.

WAN_TxFrame_Cnt (WAN Transmit Frame Count. Register 88)

Bit	Symbol	Type	Description
15:0	WAN_TxFrame_Cnt	R0	Total number of frames, good and bad, transmitted on the WAN interface.

FrameBuffer and Filter Registers (Group FBFLT)

SADB_Aging (SADB Aging. Register 90)

Bit	Symbol	Type	Description
8:0	SADB_Aging	R0	SADB Aging: The value written into this register represents the aging period of dynamic source address database entries.
15:9	0	N	Reserved: Unassigned.

FILTer_Control (SADB, UDLT Filter Control. Register 92)

Bit	Symbol	Type	Description
0	En_SADB_Filt	RW1	Enable SADB Filtering: 1 = Enables destination address filtering, 0 = disables destination address filtering, frames are by default forwarded to the WAN.
1	En_UDLT_Filt	RW1	Enable UDLT Filtering: Enables look-up table filtering.
2	UDLT_Default	RW0	Look-up table default: 1 = frames not matching filter criteria are forwarded to WAN 0 = frames not matching filter criteria are rejected.
7:3	0	N	Reserved: Unassigned.

The CPU, LAN and WAN reserve registers reserve a dedicated amount of FIFO space for each transmit port. This prevents one 'stalled' port from totally filling the Frame Buffer, and preventing the other ports from receiving data. The default size for each port is 8 frames. Note that under some conditions, the MB86976 may write into reserved Frame Buffer space to try to control congestion.

CPU_Reserve (Register 94)

Bit	Symbol	Type	Value	Description
7:0	CPU_Reserve	RW	08h	CPU Reserve: A minimum number of frame spaces reserved in the Frame Buffer for the CPU port (default and recommended minimum value is 08h).

LAN_Reserve (Register 96)

Bit	Symbol	Type	Value	Description
7:0	LAN_Reserve	RW	08h	LAN Reserve: A minimum number of frame spaces reserved in the Frame Buffer for the LAN port (default and recommended minimum value is 08h).

WAN_Reserve (Register 98)

Bit	Symbol	Type	Value	Description
7:0	WAN_Reserve	RW	08h	WAN Reserve: A minimum number of frame spaces reserved in the Frame Buffer for the WAN port (default and recommended minimum value is 08h).

Learned_SA (Learned Source Address. Register 9A)

Bit	Symbol	Type	Description
15:0	Learned_SA	R0	Learned Source Address: Counts new SA received and added to database.

NOT_Learned_SA (Not Learned Source Address. Register 9C)

Bit	Symbol	Type	Description
15:0	NOT_Learned_SA	R0	Not Learned Source Address: New SA received but not added to database as there is no free space in that line ('bucket') of the SADB database.

UpDated_SA (Updated Source Address. Register 9E)

Bit	Symbol	Type	Description
15:0	UpDated_SA	R0	Updated Source Address : Increments when a frame with a previously learned SA is received - time stamp updated. Also counts 'hits' on a Static SA.

CPU_Drop_Frames (CPU Dropped Frames. Register A2)

Bit	Symbol	Type	Description
15:0	CPU_Drop_Frames	R0	CPU Dropped: Number of frames discarded due to a Frame Buffer overflow on the CPU receive interface.

LAN_Drop_Frames (LAN Dropped Frames. Register A4)

Bit	Symbol	Type	Description
15:0	LAN_Drop_Frames	R0	LAN Dropped: Number of frames discarded due to a Frame Buffer overflow on the LAN receive interface.

WAN_Drop_Frames (WAN Dropped Frames. Register A6)

Bit	Symbol	Type	Description
15:0	WAN_Drop_Frames	R0	WAN Dropped: Number of frames discarded due to a Frame Buffer overflow on the WAN receive interface.

LAN_Reject_Frames (LAN Reject Frames. Register A8)

Bit	Symbol	Type	Description
15:0	LAN_Reject_Frms	R0	LAN Reject Frame: Number of frames discarded due to filtering results (i.e., via the source address database or the user defined look-up table).

WAN_Reject_Frames (WAN Reject Frames. Register AA)

Bit	Symbol	Type	Description
15:0	WAN_Reject_Frms	R0	WAN Reject Frame: Number of frames discarded due to filtering results (i.e., via the source address database or the user defined look-up table).

CPU_Rx_Frames (CPU Received Frames. Register AC)

Bit	Symbol	Type	Description
15:0	CPU_Rx_Frames	R0	CPU Receive Frames: Total number of frames received on the CPU receive interface.

WAN_Rx_Frames (WAN Received Frames. Register AE)

Bit	Symbol	Type	Description
15:0	WAN_Rx_Frames	R0	WAN_Receive Frames: Total number of frames received on the WAN receive port.

CPU_Tx_Frames (CPU Transmit Frames. Register B0)

Bit	Symbol	Type	Description
15:0	CPU_Tx_Frames	R0	CPU Transmit Frames: Total number of frames transmitted by the CPU.

WAN_Tx_Frames (WAN Transmit Frames. Register B2)

Bit	Symbol	Type	Description
15:0	WAN_Tx_Frames	R0	WAN Transmit Frames : Total number of frames transmitted by the WAN.

CPU_HiWtr_Level (CPU High Level. Register B4)

Bit	Symbol	Type	Description
15:0	CPU_HiWtr_Level	RO	CPU High Level: Indicates the maximum number of frames in the CPU Tx FIFO.

LAN_HiWtr_Level (Lan High Level. Register B6)

Bit	Symbol	Type	Description
15:0	LAN_HiWtr_Level	R0	LAN High Level: Indicates the maximum number of frames in the LAN Tx FIFO.

WAN_HiWtr_Level (WAN High Level. Register B8)

Bit	Symbol	Type	Description
15:0	WAN_HiWtr_Level	R0	WAN High Level: Indicates the maximum number of frames in the WAN Tx FIFO..

MASTER_HiWtr_Level (Master High Level. Register BA)

Bit	Symbol	Type	Description
15:0	MSTR_HiWtr_Level	R0	Master High Level: Is the sum of the LAN, WAN and CPU HiWtr_Levels.

Note: The "HighWaterLevel" counters indicate the worst depth of frame "pile-up" that occurred in the FrameBuffer for that type of destination port since the last time that register was read.

DRAM_Init_Done (DRAM Initialization Done. Register BC)

Bit	Symbol	Type	Description
0	DRAM_Init_Done	R0	DRAM Status: Asserted when the initialization of the DRAM has taken place just after power-up.
7:1	0	N	Reserved: Unassigned.

FULL_JTAG_ID (JTAG Indentification. Register C0)

Bit	Symbol	Type	Value	Description
32:0	JTAG_ID	R	2F00_2009h	JTAG_ID: This register contains the revision, manufacturing and production information.

Lookup_Table (Lookup Table Registers 100-18Fh)

Bit	Symbol	Type	Description
15:0	Lookup_Table	RW	Look-up Table: locations of the internal SRAM look-up table.

Programming the MB86976

To program the MB86976 to transmit and receive frames from the LAN interface or the WAN interface, set the following register bits.

Register	Address offset	Value (hex)	Comments
MAC_Control	00	28h	Full-duplex mode enabled & Connection mode: 10Mbps.
CAM_Control	04	07h	Accept all packets.
LAN_TxControl	08	01h	Transmit enable.
LAN_RxControl	09	11h	Check the CRC and then remove it from the frame.
LAN_Opt_Set	DE	03h	LAN internal Option Set
WAN_TxControl	76	10	Compressed HDLC mode of operation, WAN transmit enabled.
WAN_RxControl	77	11h	Disable WAN FCS, Compresses HDLC mode of operation. WAN receive enabled.
Lookup_Table	100-18Fh	00	All addresses in the lookup table, from 100h to 18Fh, should be initialized to zero. On power up, they are indeterminate.

Note: Setting the register bits to the above configuration is the bare minimum required to transmit and receive frames to and from the LAN and WAN interfaces. Interrupts are not enabled. Compressed HDLC is the chosen WAN protocol. Only the control words in the Lookup Table MUST be zeroed.

UDLT Programming Examples

Source Address Filtering Example

To selectively forward frames to the WAN transmit port based on the source address field, use the following in the UDLT::

bit[15]	bit[14:10]	bit[9:8]	bit[7]	bit[6]	bit[5:3]	bit[2]	bit[1:0]	mask word	data word
enable	word offset	destination	start	stop	filter id	unused	operation	mask value	data value
1	00011	10	1	0	filter #	0	00	FFFFh	source address[47:32]
1	00100	10	0	0	filter #	0	00	FFFFh	source address[31:16]
1	00101	10	0	1	filter #	0	00	FFFFh	source address[15:0]

The 48 bit address takes three of the 3-word table entries, each of which must be perfectly matched.

Ethernet 2.0 Protocol Filtering Example

To forward frames to the WAN transmit port based on the Ethernet v2.0 protocol field for AppleTalk over Ethernet (809B hex), use

bit[15]	bit[14:10]	bit[9:8]	bit[7]	bit[6]	bit[5:3]	bit[2]	bit[1:0]	mask word	data word
enable	word offset	destination	start	stop	filter id	unused	operation	mask value	data value
1	00110	10	1	1	filter #	0	00	FFFFh	809Bh

SNAP Encapsulated Filtering Example

to selectively forward frames to the processor based on SNAP encapsulated protocol field denoting AppleTalk over Ethernet (809B), the following table entries may be used:

The filter string that is searched for is: 05 DD AA AA 03 08 00 07 80 9B h::

bit[15]	bit[14:10]	bit[9:8]	bit[7]	bit[6]	bit[5:3]	bit[2]	bit[1:0]	mask word	data word
enable	word offset	destination	start	stop	filter id	unused	operation	mask value	data value
1	00110	10	1	0	filter #	0	01 *	FFFFh	05DCh *
1	00111	10	0	0	filter #	0	00	FFFFh	AAAAh
1	01000	10	0	0	filter #	0	00	FFFFh	0308h
1	01001	10	0	0	filter #	0	00	FFFFh	0007h
1	01010	10	0	1	filter #	0	00	FFFFh	809Bh

Note* : the incoming value 05DDh is matched using the '<' operator on data value 05DCh, to allow for a range of such values from 05DDh to FFFFh in the protocol field.

Multiple Filtering Examples

Through judicious partitioning of the 24 available table entries, up to 8 multiple filters can be constructed. For example, the look-up table could be configured to search for both 6 source addresses and 2 Ethernet protocols: note that the addresses and the protocols must

Type	# of bits	# of table entries	# of instances	total entries used
source address	48	3	6	18
Ethernet protocol	16	1	2	2

be interleaved in offset order. This uses a total of 20 table entries, and all 8 filters.

Alternatively, 4 different SNAP protocols, 1 source address, and one Ethernet protocol could be filtered using 6 filters and 24 table entries:

Type	# of bits	# of table entries	# of instances	total entries used
SNAP protocol	80	5	4	20
source address	48	3	1	3
Ethernet protocol	16	1	1	1

Software DMA and FrameBuffer Procedures

Procedure for completing a DMA write to the MB86976:

1. Write the physical memory address into register DMA_StartAddress, register 50. This will require 3 byte-wide writes to the 18-bit register, with the top 6 bits of the high-order byte ignored.
2. Write the number of 16-bit words to be written into register DMA_Length, register 54. The maximum number is eight.
3. Write the words to be written, starting with the low-order byte of the first word, followed by the high-order byte of the first word, and then repeat as necessary. The register (FIFO) for these writes is DMA_FIFO, register 5E. An uneven number of bytes is OK, but the data written into the MSB of the incomplete word will be unknown.
4. Write a 1 to the LSb of register DMA_Write_GO, register 5A.
5. Wait for interrupt DMA_Done, register 40, bit 6.
6. Repeat as necessary.

Procedure for completing a DMA read to the RCB:

1. Write the physical memory address into register DMA_StartAddress, register 50. This will require 3 byte-wide writes to the 18-bit register, with the top 6 bits of the high-order byte ignored.
2. Write the number of 16-bit words to be written into register DMA_Length, register 54. The maximum number is eight.
3. Write a 1 to the LSb of register DMA_Read_GO, register 56.
4. Wait for interrupt DMA_Done, register 40, bit 6.
5. Read the correct number of words from register (FIFO) DMA_FIFO, register 5E. In byte-wide data mode, the entire last word must be read, even if only the low-order byte is of interest.
6. Repeat as necessary.

Procedure for receiving a frame from the MB86976:

1. Wait for interrupt Proc_Bnd_Pkt, register 40, bit 8.
2. Poll the LSb of register FrB_Read_GO, register 4E, until it is set.
3. Read 4 16-bit words from register (FIFO) FrameBuffer, register 48. In byte-wide data mode, the first byte read is the LSB of the first word.
4. Once again, poll the LSb of register FrB_Read_GO, register 4E, until it is set.
5. Repeat steps 3 and 4. The first read word will be the header, which is used to determine the correct number of words to read. The frame should be read in 4-word blocks until the end, and then the last partial block should be read. Reads using byte-wide data should complete a whole-word, even if there are an odd number of bytes in the frame.

Procedure for sending a frame to the MB86976:

1. Write the header word and first 3 data (frame) words to register (FIFO) FrameBuffer, register 48.
2. Write a 1 to the LSb of register FrB_Write_GO, register 4C.
3. Wait for interrupt Proc_Wr_Compl, register 40, bit 10.
4. Repeat writes in groups of 4 words.
5. At the end of the frame, write the partial block of 4 words to the FrameBuffer, register 48.
6. Instead of writing to FrB_Write_GO, register 4C, write a 1 to the LSb of register End_of_Frame, register 4A.
7. Wait for interrupt Proc_Wr_Compl, register 40, bit 10, or Proc_Buf_Full, register 40, bit 9. If Proc_Buf_Full is asserted, the buffer was full and the frame was not stored for transmission. Repeat the tranmission in its entirety. If Proc_Wr_Compl is asserted, the frame was stored and will be transmitted. Note that Proc_Buf_Full is evaluated at the same time as Proc_Wr_Compl, so there is no danger that a successful transmission could be incorrectly signalled.

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings

Symbol	Parameter	Min	Max	Units
V_{DDE}	Supply Voltage		3.6	V
V_{DDI}	Supply Voltage		5.5	V
T_{ST}	Storage Temperature	-65	150	°C
P_D	Power Dissipation		900	mW
T_L	Lead Temperature (Soldering, 10 seconds)		260	°C

Note: Permanent device damage may occur if absolute maximum ratings are exceeded. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Recommended Operating Conditions

Symbol	Parameter	Conditions	Min	Type	Max	Units
V_{DDE}	Supply Voltage		4.5	5.0	5.5	V
V_{DDI}	Supply Voltage		3.0	3.3	3.6	V
V_{IH}	High-Level Input Voltage	5.0 V I/O	V_{DDI} x 0.65		V_{DDE}	V
V_{IL}	Low-Level Input Voltage	V_{SS} = 0 V	V_{SS}		V_{DDI} x 0.25	V
T_j	Junction Temperature		0		100	°C
T_A	Operating Temperature					°C
	ESD Tolerance	C_{ZAP} = 100 pF, R_{ZAP} = 1.5 kΩ			1.5	kV

DC Characteristics (V_{DDE} = 5.0 V ± 0.5 V, V_{DDI} = 3.3 V ± 0.3 V, Tj = 0-100°C)

Symbol	Parameter	Conditions		Min.	Typ	Max.	Units
I_{DDS}	Supply Current	$V_{IH} = V_{DD}$ and $V_{IL} = V_{SS}$, memory in standby mode			200		mA
V_{IH}	High-Level Input Voltage			2.5			V
V_{IL}	Low-Level Input Voltage					0.8	V
V_{OH2}	High-Level Output Voltage	LED driver Pins	I_{OL} = -12 mA,	V_{DDE} - 0.5		V_{DDE}	V
		All Others	I_{OL} = -4 mA,	V_{DDE} - 0.5		V_{DDE}	V
V_{OL}	Low-Level Output Voltage	LED driver Pins	I_{OL} = 12 mA,	V_{SS}		0.4	V
		All Others	I_{OL} = 4 mA,	V_{SS}		0.4	V
R_P	Input Pull-up Resistor	Pins with Pull-up	V_{IL} = 0 V	25	50	100	kΩ
I_L	Input Leakage Current	Pins without Pull-up	V_O = 0 V or V_{DDE}			± 5	µA
I_{OS2}	Output Short-Circuit Current	LED driver Pins	V_O = 0 V or V_{DDE}			± 360	mA
		All Others				± 90	mA

Pin Capacitance

Symbol	Parameter	Conditions	Min.	Max.	Units
C_{IN}	Input Pin Capacitance			16	pF
C_{OUT}	Output Pin Capacitance			16	pF
$C_{I/O}$	Input/Output Pin Capacitance			16	pF

Note: T_j = 25 °C, $V_{DD} = V_I$ = 0 V, f = 1 MHz

ORDERING INFORMATION

Part Number	Package Style	Package Code
MB86976PF-G	144-pin Plastic QFP	FPT-144P-M03

PACKAGE DIMENSIONS