MB86960

NETWORK INTERFACE CONTROLLER with ENCODER/DECODER (NICE)

DATA SHEET

APRIL1993

FEATURES

- High-performance packet buffer architecture pipelines data for highest throughput
- 20 Mbyte/second data transfer rate to/from the system bus
- on-chip buffer controller manages pointers, reduces software overhead
- Efficient, configurable two bank transmit buffer and ring receive buffer
- Bus-compatible with most popular microprocessors, including RISC
- Complies with international standards for Ethernet, ISO/ANSI/IEEE 8802-3
- High-speed burst and single transfer DMA
- 64-element hash table for multicast address filtering
- High-speed, low-power CMOS technology
- Power down mode reduces power dissipation for battery-powered equipment
- Available in 100-pin plastic quad flat package

GENERAL DESCRIPTION

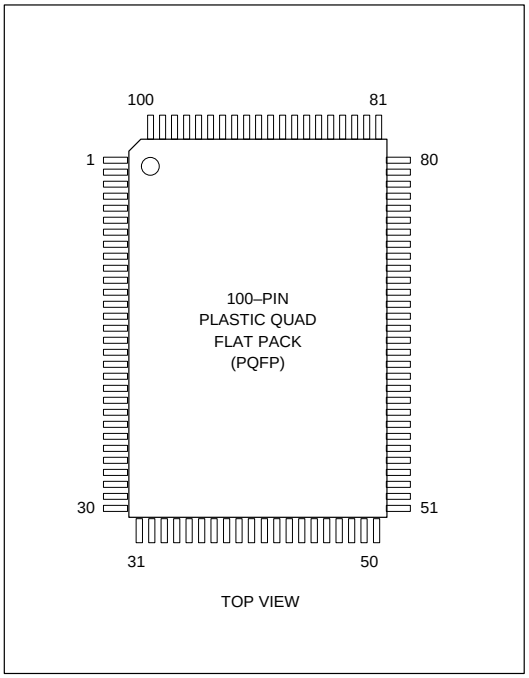
The MB86960 Network Interface Controller with Encoder/Decoder (NICE™) is a high-performance, highly integrated monolithic device which incorporates both network controller, complete with buffer management, and Manchester encoder/decoder. It allows implementation of a 7-chip solution for an Ethernet interface when used with either of Fujitsu's bus interface chips, the MB86953 for PC/XT/AT or the MB86954 for Micro Channel™, and either of Fujitsu's transceiver chips, the MBL8392A coaxial transceiver or MB86962 10BASE-T twisted-pair transceiver.

The unique buffer management architecture of the MB86960 allows packet data to access a buffer memory area from the host and from the network media simultaneously, with virtually no interaction. The network controller updates all receive and transmit pointers internally, thus reducing the software overhead required to control these operations, resulting in superior benchmark speed and application performance. The NICE device has a partitionable 2, 4, 8, or 16 kilobyte, two-bank, transmit buffer which allows multiple data packets to be "chained" together and transmitted to the network from a single transmit command, thus allowing greater design flexibility and throughput. Receive packets are captured in a ring buffer which can be configured in various sizes from 4 to 62 kilobytes, depending on memory equipped and amount used for the transmit buffer.

Possible configurations for the system bus interface include I/O mapping, memory mapping and DMA access, or a combination of these. With a 20 Mbyte/sec bandwidth, the NICE system bus interface allows you to use the full throughput capacity of its unique packet buffering architecture. The NICE controller's selectable bus modes provide both big- and little-endian byte ordering, permitting an efficient data interface with most microprocessors and higher-level protocols.

Implemented in Fujitsu's high-speed, low-power CMOS process, the MB86960 is supplied in a 100-pin plastic quad flat package for surface mounting.

PIN CONFIGURATION



PIN ASSIGNMENTS AND DESCRIPTIONS

Supplied in a 100-pin plastic quad flat pack, the NICE controller presents a small foot-print to the board design, and is surface-mountable with its gull-wing leads. See Pin Configuration and Pin Assignments for the pin numbering.

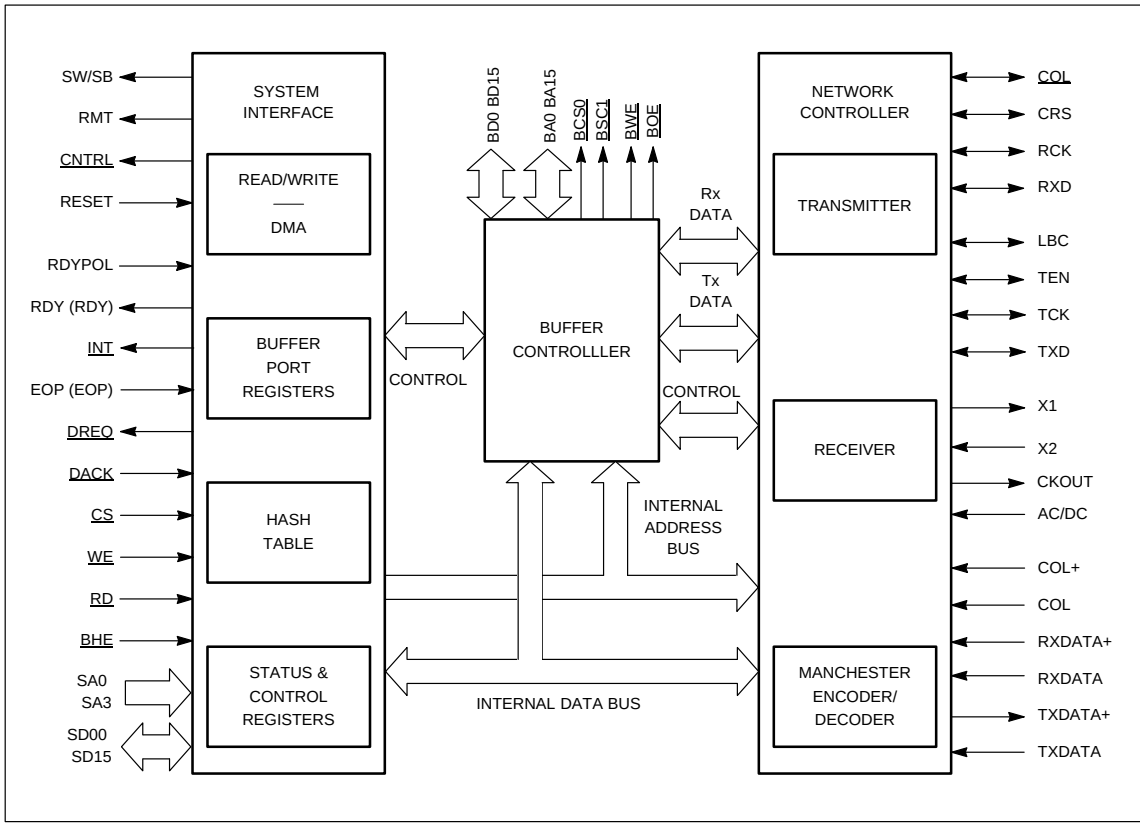
PIN ASSIGNMENTS

PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE	PIN NO.	PIN NAME	TYPE
1	SD11	B	26	DREQ	O	51	BCS0	O	76	X1	I
2	SD10	B	27	DACK	I	52	BCS1	O	77	X2	O
3	V_{CC}	—	28	V_{CC}	—	53	V_{CC}	—	78	V_{CC}	—
4	GND	—	29	RD	I	54	BA0	O	79	GND	—
5	SD9	B	30	WE	I	55	BA1	O	80	TCK	B
6	SD8	B	31	RESET	I	56	BA2	O	81	RXDATA–	I
7	CS	I	32	BD0	B	57	BA3	O	82	RXDATA+	I
8	BHE	I	33	BD1	B	58	BA4	O	83	COL–	I
9	SW/SB	O	34	BD2	B	59	BA5	O	84	COL+	I
10	SA0	I	35	BD3	B	60	BA6	O	85	V_{CC}	—
11	SA1	I	36	BD4	B	61	BA7	O	86	LBC	B
12	SA2	I	37	BD5	B	62	BA8	O	87	AC/DC	I
13	SA3	I	38	BD6	B	63	BA9	O	88	RCK	B
14	RDY(RDY)	O	39	BD7	B	64	BA10	O	89	CKOUT	O
15	GND	—	40	GND	—	65	GND	—	90	GND	—
16	SD0	B	41	BD8	B	66	BA11	O	91	RXD	B
17	SD1	B	42	BD9	B	67	BA12	O	92	COL	B
18	SD2	B	43	BD10	B	68	BA13	O	93	CRS	B
19	SD3	B	44	BD11	B	69	BA14	O	94	RDYPOL	I
20	SD4	B	45	BD12	B	70	BA15	O	95	CNTRL	O
21	SD5	B	46	BD13	B	71	TEN	B	96	RMT	O
22	SD6	B	47	BD14	B	72	TXD	B	97	SD15	B
23	SD7	B	48	BD15	B	73	GND	—	98	SD14	B
24	EOP(EOP)	I	49	BOE	O	74	TXDATA–	O	99	SD13	B
25	INT	O	50	BWE	O	75	TXDATA+	O	100	SD12	B

Note: B = Bidirectional I/O
I = Standard Input
O = Totem Pole Output

Dual function pins have two names with the second in parentheses ( ).

BLOCK DIAGRAM



PIN DESCRIPTIONS

System Bus Interface Pins

SYMBOL	TYPE	DESCRIPTION
RESET	I	**HARDWARE RESET:** Active high. A minimum pulse of 300 nanoseconds in duration is required. This pin resets NICE's internal pointers and registers to the appropriate state. Note: NICE must be reset after power start before using.
RDY (RDY)	O	**READY:** This output is asserted to indicate to the host that NICE is ready to complete the requested read of write operation. It will also be asserted if the device is unable to respond to the request for a read or write within 2.4 microsecond, In that case, NICE will also assert INT and the bus read error status bit, DLCR1<6>, or bus write error status bit DLCR<0>. RDY(RDY) may be an active low or active high signal as determined by RDYPOL, pin 94. If RDYPOL is a "1", RDY(RDY) will be active high. If RDYPOL is tied to a "0" RDY(RDY) will be an active low signal.
RDYPOL	I	**READY POLARITY SELECT:** Control input to select the polarity of RDY(RDY), pin 14. When this pin is tied high, RDY(RDY) will be active high. If RDYPOL is tied low, RDY(RDY) will be an active low signal.
WE	I	**WRITE:** The WE pin is an active low input that enables a write operation form the host system to the buffer memory port or to internal registers selected by system address inputs SA0-3.
RD	I	**READ:** Active low input specifies that the current transfer between NICE and the host system is a read from one of NICE's internal registers or its data port as selected by SA0-3.
CS	I	**CHIP SELECT:** This active low input signal is the chip select for NICE.
BHE	I	**BYTE HIGH ENABLE:** Active Low. This is the byte/word control line. It is used only when NICE is configured for a 16-bit data bus by the SB/SW bit of DLCR6. It allows word, upper byte only or lower byte only transfers. The address select pin SA0 is used with BHE for byte or word transfers. as follows.
INT	O	**INTERRUPT:** Active low. Indicates that NICE requires host system attention after successful transmission or reception of a packet, or if any error condition occur, if an EOP (end of process) signal from the host occurs after the completion of the DMA cycle. The Interrupt signal is maskable and can be disabled by writing a 0 to the appropriate mask bit.
EOP(EOP)	I	**END OF PROCESS:** Indicates to NICE that the DMA transfer is finished. When the host DMA controller asserts EOP(EOP), further assertion of NICE's bus request output. DREQ, will be discontinued.

SB/SW	BHE	SA0	FUNCTION
0	0	0	Word transfer
0	0	1	Byte transfer on upper half of data bus (SD15-8)
0	1	0	Byte transfer on lower half of data bus (SD7-0)
0	1	1	Reserved
1	X	X	Byte transfer (SD7-0)

Note: B = Bidirectional I/O
I = Standard Input
O = Totem Pole Output

Dual function pins have two names with the second in parentheses ( ).

PIN DESCRIPTIONS

System Bus Interface Pins (continued)

SYMBOL	TYPE	DESCRIPTION
CNTRL	O	**CONTROL:** This pin is the complement of the register bit CNTRL, DLCR4<2>. It is used to control external functions.
RMT	O	**REMOTE CONTROL PACKET:** When DLCR5<2> is set high, this pin follows the RMT 0900H bit (DCLR1<4>) which indicates that a complete special packet with type field= 0900H has been received. This is intended for use as a remotely-controlled hardware function from other nodes in the network.
DREQ	O	**DMA REQUEST:** Issued to the DMA controller to indicate that NICE has data available to be read in its receive buffer, of is ready to accept data into its transmit buffer.
DACK	I	**DMA ACKNOWLEDGE:** Active low, indicate that the DMA controller is ready to transfer data between the host system and NICE's buffer memory through BMPR8.
SA<3:0>	I	**SYSTEM ADDRESS LINES:** Specify which of the internal registers of ports of NICE is selected for read/write operations.
SD<15:0>	B	**SYSTEM DATA BUS:** All data, command and status transfers between the host system and NICE take place over the bidirectional, 3-state, bus. The direction of the transfer is controlled by RD and WE. The register or buffer port being accessed is selected by a combination of DACK (if active, selecting the Buffer Port), of the address pins SA3-0 and register bank select bits REG BANK 1 and REG BANK 0, DLCR7<3:2>. The portion of the data bus over which the transaction occurs is controlled by SB/SW, BHE, and SA0.
SW/SB	O	**SYSTEM WORD/SYSTEM BYTE CONFIGURATION:** This signal output reflects the inverse of DCLR6<5>, SB/SW. If SW/SB=1, the system interface is configured for word transfers, If SW/SB=0, the system interface is configured for byte-wide transfer on SD7-0, the lower byte.

Note: B = Bidirectional I/O
I = Standard Input
O = Totem Pole Output

Dual function pins have two names with the second in parentheses ( ).

PIN DESCRIPTIONS

Buffer Memory Interface Pins

SYMBOL	TYPE	DESCRIPTION
BCS0 BSC1	O O	**BUFFER CHIP SELECT:** BCS1 and BSC0 are the chip select lines, most significant byte and least significant byte respectively, of the dedicated buffer SRAMs, Active low.
BOE	O	**BUFFER OUTPUT ENABLE:** This active low signal is the output enable for the Buffer SRAM, and is asserted by NICE during buffer memory read cycles.
BWE	O	**BUFFER WRITE ENABLE:** Active low. Used as a write strobe to the buffer SRAM memory during write operations.
BD<15:0>	B B	**BUFFER DATA:** Data lines between the SRAM buffer memory and NICE. This SRAM data bus is configurable for an 8-bit or 16-bit data size by BUFFER BYTE/BUFFER WORD, BB/BW, in DLCR6<4>. The transfer byte order within a word, most significant or least-significant byte first, is determined by DATA_ORDER, DLCR7<1>.
BA<15:0>	O O	**BUFFER ADDRESS:** These lines address up to 64 kilobytes of SRAM buffer memory.

Network Interface Pins

SYMBOL	TYPE	DESCRIPTION
TXDATA+ TXDATA–	O O	**TRANSMIT INTERFACE PAIR:** These are the differential outputs to the transceiver for transmitting.
RXDATA+ RXDATA–	I I	**RECEIVED DATA:** These are the Manchester differential inputs from the transceiver to the receiver.
COL+ COL–	I I	**COLLISION:** These differential inputs are driven with a 10 MHz signal when the transceiver detects a collision on the media.
AC/DC	I	**AC/DC COUPLING SELECT:** AC/DC = 1 selects AC coupling; 0 selects DC coupling for the TXDATA ± outputs. When AC coupling is selected, both TXDATA+ and TXDATA– are driven to the same output voltage level during the transmit idle period to prevent saturation of the isolation transformer. With DC coupling, these outputs remain at a 1 level during idle periods.

System Clock Pins

SYMBOL	TYPE	DESCRIPTION
X1	I	**CRYSTAL INPUT:** Connection for one side of the 20 MHz crystal, or input for an external 20 MHz clock source.
X2	O	**CRYSTAL OUTPUT:** Connection for the other side of the 20 MHz crystal. Leave unconnected if an external clock is used.
CKOUT	O	**CLOCK OUTPUT:** 20 MHz free-running clock output provided by the crystal oscillator circuit.

PIN DESCRIPTIONS

The following eight pins are provided for optional connection to an external encoder/decoder. They can also be used for test purposes, but are not used in a typical network interface application. Special encoder/decoder modes are selected by setting DLCR7<7:6>. Refer to the **MEDIUM CONNECTION** section under **SYSTEM CONFIGURATION** for a complete description of these modes.

Controller — Encoder/Decoder Interface Pins

SYMBOL	TYPE	DESCRIPTION
TXD	B	**TRANSMIT DATA:** NRZ transmit serial data. Normally not used. In "Encoder/Decoder Bypass" mode and "NICE + Monitor" mode, an output which can be fed to an external encoder. In "Encoder/Decoder Test" mode, an input to the on-chip encoder.
TCK	B	**TRANSMIT DATA CLOCK:** Clock synchronous with TXD serial data. Normally not used. In "Encoder/Decoder Bypass" mode, a 10 MHz data clock input used by the controller to synchronize the TXD data signal. In "NICE + Monitor" mode and "Encoder/Decoder Test" mode, an output from the on-chip encoder.
TEN	B	**TRANSMIT ENABLE:** Normally not used. In "NICE + Monitor" mode and "Encoder/Decoder Bypass" mode, this pin is an output which can be used to control an external encoder. When asserted high, TEN enables the encoding of the transmitted data. In "Encoder/Decoder Test" mode, an input to the on-chip encoder.
$\overline{\text{COL}}$	B	**COLLISION PRESENCE:** Normally not used. In "Encoder/Decoder Bypass" mode, an active low input which indicates that the collision inputs COL are active, signifying that the transceiver has detected a collision on the media. In "NICE + Monitor" mode and "Encoder/Decoder Test" mode, this is an output from the on-chip encoder/decoder section. $\overline{\text{COL}}$ is a normally high input which changes to a 10 MHz pulse stream during collision detection.
LBC	B	**LOOPBACK CONTROL:** Normally not used. In "Encoder/Decoder Bypass" mode and "NICE + Monitor" mode, this output should be connected to the loopback control pin of the external encoder/decoder. When this output is asserted high, the external encoder/decoder is placed in loopback mode. TEN enables the encoding of the transmitted data. In "Encoder/Decoder Test" mode, this is an input with the same function.
RXD	B	**RECEIVE SERIAL DATA:** Normally not used. In "Encoder/Decoder Bypass" mode, "NICE + Monitor" mode and "Encoder/Decoder Test" mode, an NRZ serial bit stream from a decoder or demodulator to the data link controller.
RCK	B	**RECEIVE DATA CLOCK:** Normally not used. In "Encoder/Decoder Bypass" mode, "NICE + Monitor" mode and "Encoder/Decoder Test" mode, this pin is an input for the serial data clock as recovered by the external decoder or demodulator.
CRS	B	**CARRIER SENSE:** Normally not used. In "Encoder/Decoder Bypass" mode, an input indicating the presence of incoming data on the network. In "NICE + Monitor" mode and "Encoder/Decoder Test" mode, an output from the on-chip encoder/decoder. Assertion of this active high input indicates that a carrier has been sensed at the RXDATA inputs.

PIN DESCRIPTIONS

Mode Configuration and Encoder/Decoder Pin Input/Output Table

CNF1	CNF0	MODE	TXD	TCK	TEN	LBC	RXD	RCK	CRS	COL
0	0	NORMAL NICE	Internal E/D is used with NICE controller, no signal appears on pins							
			none	none	none	none	none	none	none	none
0	1	NICE + MONITOR	Internal E/D is used with NICE controller, signal appears on output							
			O	O	O	O	O	O	O	O
1	0	E/D BYPASS	External E/D is used with NICE controller, internal E/D is shutdown							
			O	I	O	O	I	I	I	I
1	1	E/D TEST	Internal E/D is used only, NICE controller section is shutdown							
			I	O	I	I	O	O	O	O

Device Power Pins

SYMBOL	DESCRIPTION
V_{CC}	**POWER SUPPLY:** A +5 V_{DC} ±5% supply is required.
GND	SYSTEM GROUND:

Ordering Code

PACKAGE STYLE	PACKAGE CODE	ORDERING CODE
100-Pin Plastic Quad Flat Pack	FPT-100P–M01	MB86960APF–G

SYSTEM CONFIGURATION

A highly integrated system configuration can be achieved with the NICE controller. Figure 1 illustrates a low chip count LAN controller with NICE, a bus interface chip such as Fujitsu's MB86953 or MB86954, and either a coaxial transceiver such as Fujitsu's MBL8392A or a 10BASE-T twisted pair transceiver such as Fujitsu's MB86962. Because of its high integration and unique, innovative architecture, which handles all aspects of packet management and storage, a local microprocessor is not required.

The NICE controller connects to the host system bus to provide command and status interfaces as well as packet data access. Command and status registers can be directly accessed by the host processor when mapped into the I/O or memory space of the host. Through a port on the device, data packets to be transmitted to the media are transferred first from host memory to a dedicated buffer memory for temporary storage until transmitted. Received data packets are first stored in the buffer memory, then later transferred to the host memory.

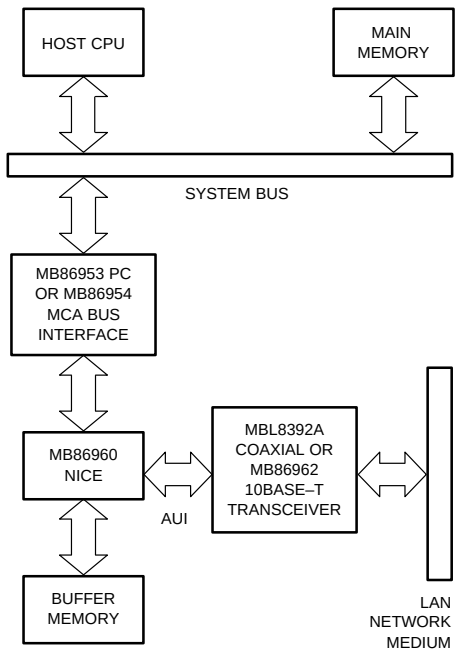


Figure 1.Typical System Configuration

Medium Connection

Connection to the LAN medium can be accomplished with any of the popular connection methods: 1) on-board connection to unshielded twisted pair through a 10BASE-T transceiver, 2) on-board connection to a thin 50-Ohm coaxial cable through a 10BASE2 transceiver or 3) off-board connection to any other type of medium, such as standard Ethernet coaxial cable (10BASE5), through an Attachment Unit Interface (AUI) connector.

NICE has an encoder/decoder (E/D) on chip. An external encoder/decoder can be used by making the NICE chip act like a controller alone (depending on customer's needs). This option can be changed by using bits 7 and 6 of DLCR7.

Eight pins related to the on-chip E/D can be configured by DLCR7<7:6> (register DLCR7, bits 7 through 6) to operate in one of four modes. These pins are TXD, TCK, TEN, LBC, RXD, RCK, CRS, and COL. In the "Normal NICE" mode, an internal E/D is used. In this mode, the pins are all electrically isolated and no signals appear on the pins. In "NICE + Monitor" mode, all the pins listed above are outputs whose specific signals appear on the pins and can be monitored externally. In "Encoder/Decoder Bypass" mode, an external encoder/decoder is used with the NICE controller, its own internal E/D is shut down. In this mode, the pins are either outputs or inputs as needed to control the external encoder/decoder. In the "Encoder/Decoder Test" mode, only the E/D on NICE is active and accessible, the NICE controller section is shut down. In this mode, the pins are outputs or inputs for an encoder/decoder, with the opposite control direction of the outputs or inputs in the "Encoder/Decoder Bypass" mode. The various possibilities are shown in Figure 2 and the table below.

DLCR7 Bit 7	DLCR7 Bit 6	Function
0	0	Normal NICE
0	1	NICE plus Monitor
1	0	Encoder/Decoder Bypass
1	1	Encoder/Decoder Test

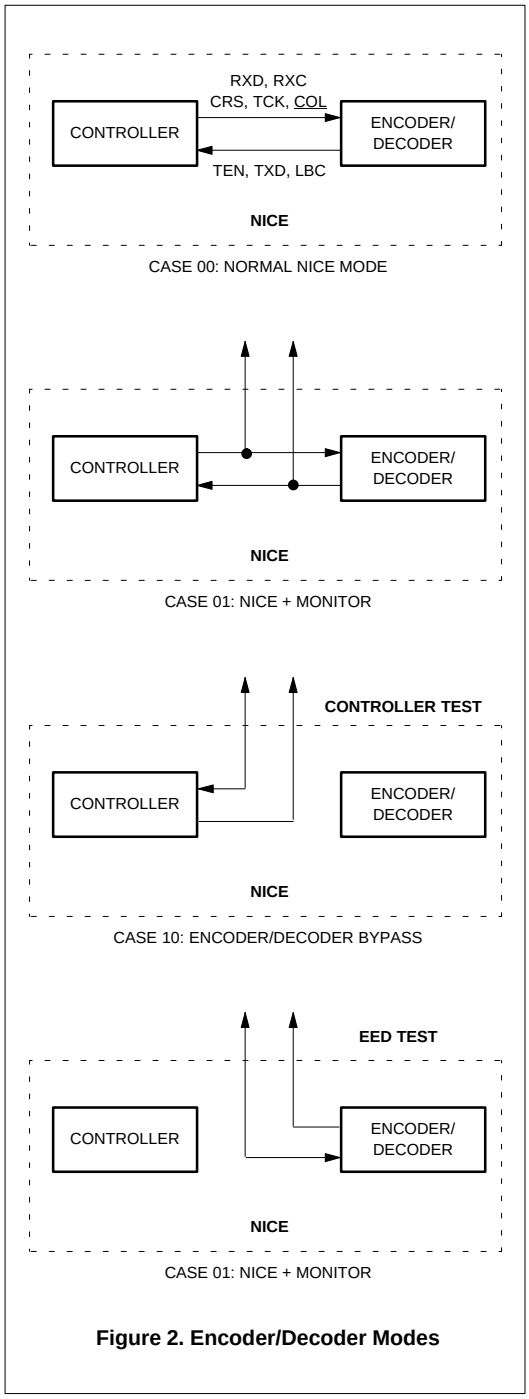


Figure 2. Encoder/Decoder Modes

CRYSTAL OSCILLATOR

The ISO/ANSI/IEEE 8802–3 international LAN standard specifies a bit clock rate of 10 Mbit/sec. This is obtained from a 20 MHz clock generated by the on-chip oscillator, which operates from an external crystal connected between pins X1 and X2 on the NICE chip. Crystal capacitance as specified by the manufacturer should be connected from X1 and X2 to ground, considering any stray capacitance which can vary the crystal's frequency. See Figure 3 for typical values. A crystal with the following specifications is recommended: quartz (AT-cut; 20 MHz; frequency/accuracy of ±50 ppm at 25°C to 70°C; parallel resonant with 20 pF load in a fundamental mode. Possible vendors include: Ecliptek Corp. (Costa Mesa, CA), p/n ECSM200-20.000; and M-tron Industries, Inc. (Yankton, SD), p/n MP-1 & MP-2, with 20MHz, 50ppm over 0°C to 70°C, and 18 pF fundamental load.

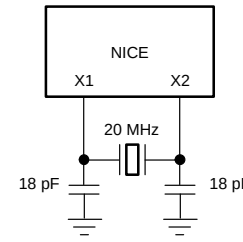


Figure 3. Crystal Connection

The clock also serves as a reference for an internal phase locked loop which is used for clock recovery in the decoder section. Internal clocks are shut down when DLCR7 bit 5, PWRDN, is invoked for Power Down Mode.

SRAM CONFIGURATIONS

Eight different configurations of SRAM for the packet buffer are possible as illustrated in Figure 4. First, the width of the SRAM data path can be 8 or 16 bits, selected by programming the Buffer Byte/Buffer Word (BB/BW) bit, bit 4 of DLCR6. If this bit is set to 1, byte-wide data is selected; if set to 0, the width is word-wide (16 bits). The SB/SW Bit, DLCR6 bit 5, selects the system bus width, while the BB/BW Bit, DLCR6 bit 4 selects the SRAM buffer width. Secondly, the depth of the SRAM is programmed by setting Buffer Size 1 and 0 (BS1 and BS0), bits 1 and 0 in DLCR6. Depth selections are 8, 16, 32 or 64 kilobytes.

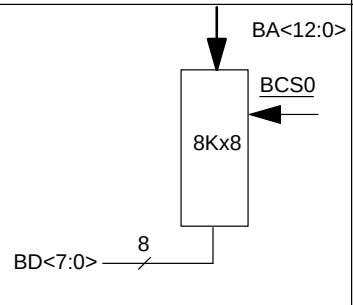
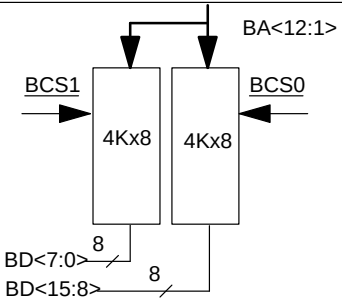
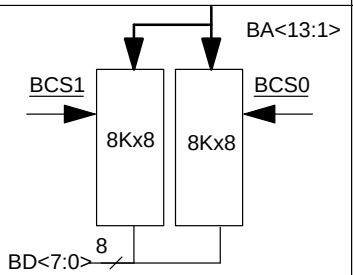
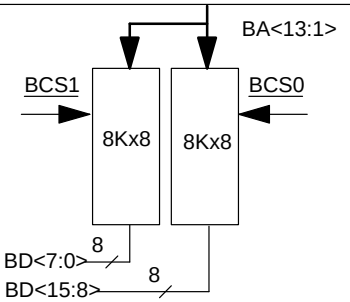
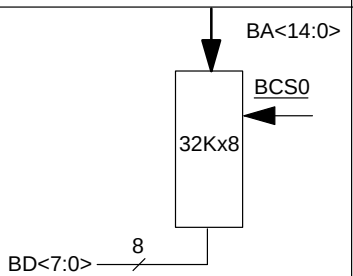
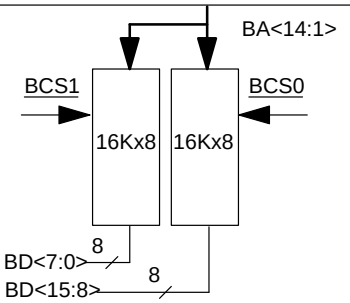
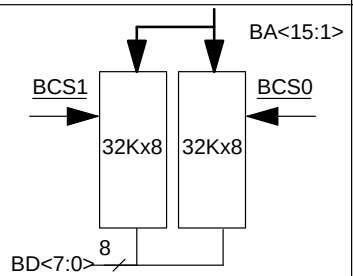
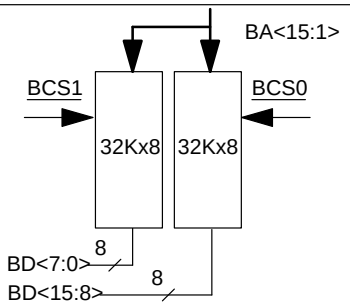
BS1 DLCR6<1>	BS0 DLCR<0>	8–BIT BUFFER BUS	16–BIT BUFFER BUS
0	0		
0	1		
1	0		
1	1		

Figure 4. SRAM Configuration

SB/SW	BB/BW	SYSTEM	BUFFER
0	0	word	word
0	1	word	byte
1	0	Do not use	
1	1	byte	byte

FUNCTIONAL DESCRIPTION

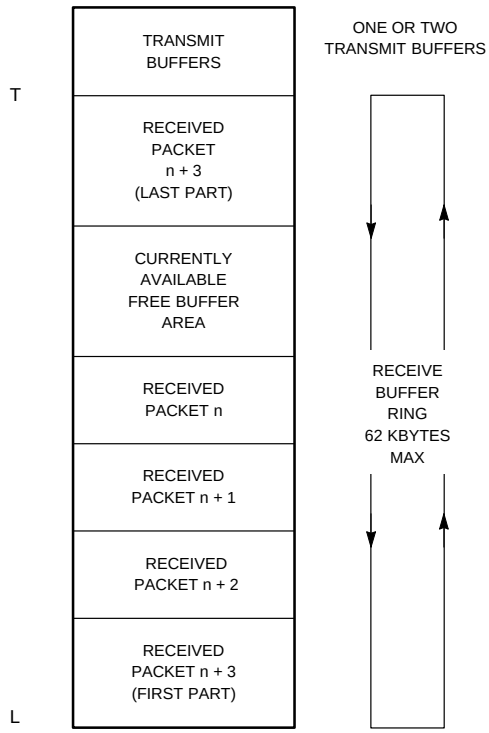
The MB86960 combines the functions of an Ethernet network controller with packet buffer management, and a 10 Mbit/s Manchester encoder/decoder. It consists of four major functional blocks: buffer controller, system interface, transmit controller with Manchester encoder, and receive controller with Manchester decoder.

The receive and transmit sections of the chip fully implement the ISO/ANSI/IEEE 8802-3 CSMA/CD specification for 10 Mbit/sec Ethernet. The transmitter assembles data packets for transmission and the receiver disassembles received data packets. Automatic generation and stripping of the 64-bit preamble, and generation and checking of the 32-bit CRC are provided on-chip. Other network functions provided on-chip include collision resolution by binary exponential backoff and re-transmission, several modes of address recognition, error detection and reporting, and serial/parallel and parallel/serial conversions.

BUFFER CONTROLLER

The MB86960 uses a dedicated buffer memory as shown in for intermediate storage of data packets to be transmitted, and of data packets received from the network. The buffer memory is connected directly to the controller rather than to a separate local microprocessor bus, thus eliminating the need for a local microprocessor. The buffer controller keeps track of buffer memory partitioning, allocation and updating of all receive and transmit pointers automatically, thus eliminating this task from software overhead. As a result of this automation and its high-performance packet buffering, the NICE controller can typically win benchmark performance tests over competing controllers.

Access to the buffer memory is managed by NICE's on-chip buffer controller. As required, it updates internal address pointers for the tasks of transmit, retransmit, receive, rejection of packets with errors and data transfers to and from the host. Thus the host is relieved of buffer management functions, making NICE easy to operate and substantially reducing software requirements. Packets with errors, such as CRC errors, are auto–matically rejected by NICE unless the host asserts the "accept bad packets" bit. When this bit is asserted, any packets received with alignment, CRC or short length errors are passed on to the host processor, and the appropriate error status bits are set to inform the host of the error. Similarly, by setting the "accept short packets" bit, reception of short packets down to 6 bytes in length is allowed. (Normal operation requires an IEEE minimum length packet of 60 bytes, excluding preamble and CRC.)



NOTES
1. Length (L) = 8, 16, 32, or 64 Kbytes.
2. Transmit Size (T) = 2 (one bank), 4 (two banks), 8 (two banks), or 16 (two banks) Kbytes.
3. Receive packets are aligned on an eight byte boundary
.
4. Each received packet is preceded by a four byte packet header as follows:

   Byte 0 – Packet Status

   Byte 1 – Not Used

   Byte 2 – Packet Size LSB's

   Byte 3 – Packet Size MSB's

Figure 5. Buffer Memory Organization

As shown in the buffer memory is divided into transmit and receive buffer areas. The partitioning of the memory is programmable, allowing the system to be configured with different proportions of space available for the transmitter and receiver functions. By programming the proportions, an optimum usage of the memory can be selected, according to the demands of a particular application.

The section of the memory used by the transmitter can be configured as a single buffer 2 kilobytes long, or as a pair of banks, each either 2, 4 or 8 kilobytes long. Within each buffer or bank, one or more packets can be written by the system until the available space is too small for another packet. Once started, the transmitter will transmit all packets in the buffer automatically before finishing with a status update, and an interrupt if so enabled. With the two-bank configurations, one bank can be transmitted while the other is being loaded. Using dual buffers and loading multiple packets for "packet chaining" gives the highest rate of transmission. This will boost performance for systems that require high throughput transmission.

NICE can be configured to operate with 8, 16, 32 or 64 kilobytes of total buffer memory size, including both transmit and receive spaces. This memory partitioning into transmit and receive sections is allocated by the system software. The total size of the transmit buffer space can be either 2, 4, 8 or 16 kilobytes. Immediately following the transmit buffer space is the receive buffer space, using the balance of the available memory.

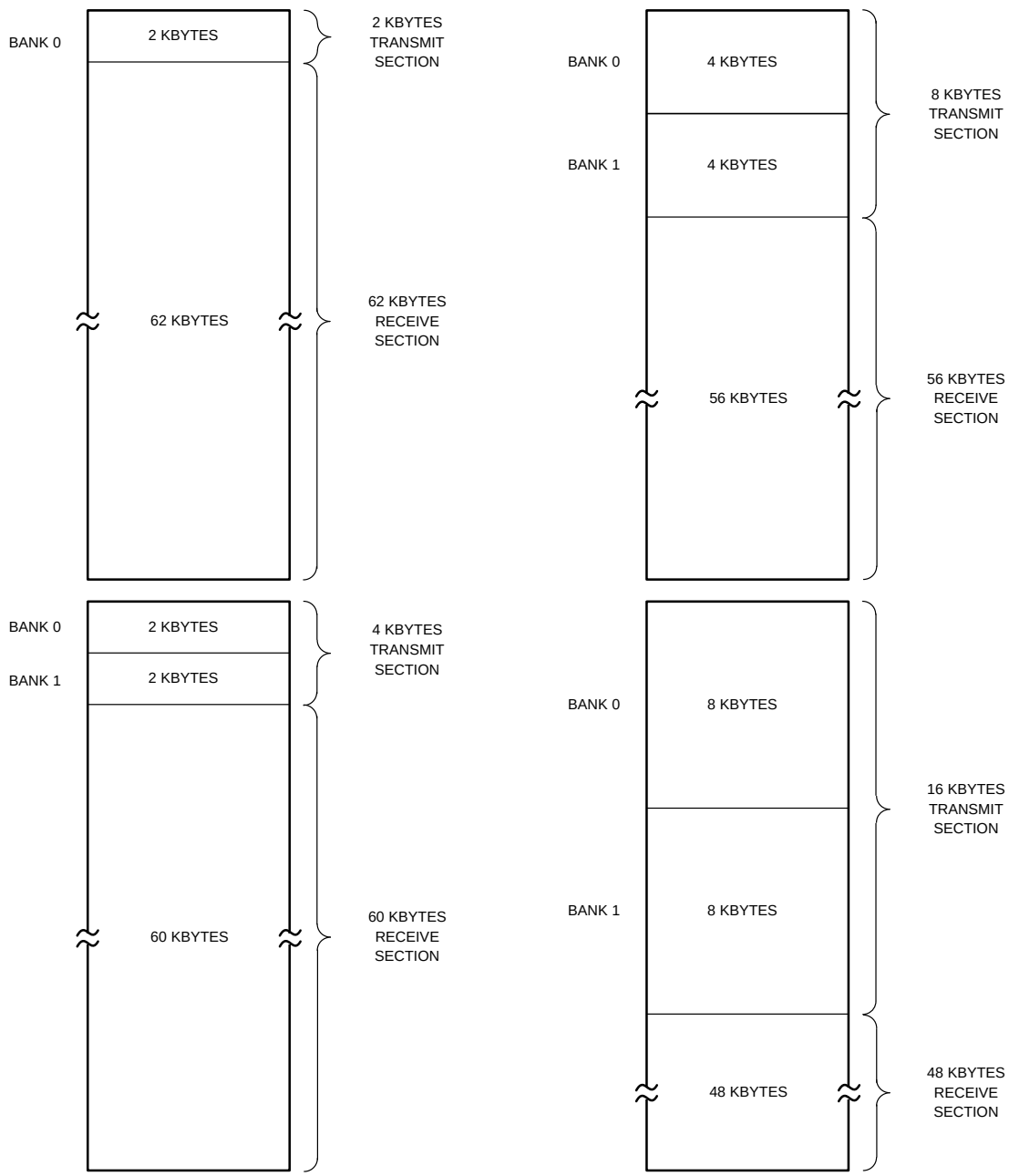


Figure 6. Transmit Buffer

As shown in Figure 6, the remaining memory not used for the transmitter is used for the receiver, and is automatically configured as a "ring buffer" by the chip. Packets are stored head-to-toe in the receive buffer, as they are in the transmit buffer. As packets are being stored in the receive buffer, as the end of the linear addressing space is reached, the chip's receive-write pointer automatically wraps around to the top of the receive addressing range to make a seamless ring. The receive-read pointer does the same as the packets are read out to the system. The MB86960 provides all the necessary buffer pointer management functions automatically, relieving the host system and its drivers of this time-consuming task. Since these tasks can be done faster in hardware than in software, this not only off-loads the host system, but it also speeds up the communication processes giving higher throughput.

The buffer controller automatically prioritizes and services requests for access to the memory from the transmitter, the receiver, and the host system. It updates all buffer memory pointers, allocates memory space for incoming data packets, and controls pertinent bits within the status registers providing complete packet management functions.

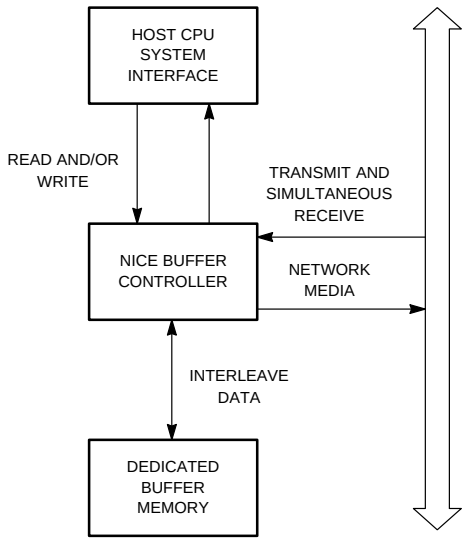


Figure 7. Simultaneous Operations and Buffer Control Capability of NICE

The NICE chip's arbitration mechanism provides packet management by interleaving packet data accesses to the buffer memory so that the operation appears to be "simultaneous": data can be written to or read from the buffer memory by the host via Buffer Memory Port Register 8 (BMPR8), while data packets are read out for transmission and/or written in for storage by the receiver. Each interface, whether host system or network access, appears to be served independently by the controller. Each interface has an associated FIFO to provide time for the buffer interleaving. Thus, packet data is "pipelined" through the system for highest performance and throughput, and the buffer controller supports all the cases of "simultaneous" access to the buffer memory, as illustrated in Figure 7 and as follows:

1. Data from the network is stored in the receive buffer.
2. The host retrieves packets from the receive buffer.
3. The host loads packet data into the transmit buffer.
4. The transmitter obtains data for transmission from the transmit buffer.
5. Any combination of the above can occur concurrently, including all four at once.

System Access to Buffer

NICE supports both programmed I/O, single cycle DMA and burst mode DMA transfers between the buffer memory and the host system. The host accesses the buffer memory by reading from or writing to NICE's Buffer Memory Port Register 8 (BMPR8). Data being read or written by the system passes through on-chip FIFO's to eliminate the effects of real-time interaction between the system, the transmitter and the receiver as they all access the buffer memory. All read and write operations to the external SRAM memory are controlled automatically by the NICE chip.

Transmitter Access to Buffer

The size of each of the transmit banks can be changed by programming the Buffer Size control bits, BS0 and BS1. The transmit buffer size is thus allocatable by the software to be a single 2-kilobyte transmit buffer, two 2-kilobyte transmit buffers, two 4-kilobyte transmit buffers, or two 8 kilobyte transmit buffers. In all configurations, a single packet or multiple packets can be loaded into the buffer at one time for transmission. When a single transmit buffer is used, the system and the transmitter time-share the use of the buffer. When two buffers are used, the system can load packets into one of the buffers while the contents of the other are being transmitted.

At reset, the pointers are initialized to point to the beginning of one of the transmit buffers. Each time the host writes data to the buffer via the Buffer Memory Port Register, an internal pointer is advanced to the next memory location within the transmit buffer. Once a data byte/word is written, it cannot be read and the internal pointer cannot be reversed.

Internal pointers managed by NICE control which of the two banks is selected for access by the host and which byte/word of the bank is accessed. NICE will switch banks as soon as the transmit start bit (TX START) of BMPR10 is written high by the host system. When this occurs, NICE will start transmitting at the earliest opportunity. Another automatically-managed pointer, the transmit-read pointer sequences through the bank being transmitted to read the packet data into the transmitter through its FIFO. If a collision occurs, the packet will be automatically retransmitted after a pseudo-random waiting interval called the backoff interval. If there are multiple packets in the bank, NICE will continue down the list, automatically transmitting them all. The details of this operation are described in the section on packet transmission.

NICE has the capability to transmit multiple back-to-back packets of varying legal Ethernet sizes to the LAN network. These packets may vary in length from 60 bytes to 1,514 bytes, excluding preamble and CRC fields. As shown in Figure 8, multiple packets can reside within one transmit bank, separated by a non-transmitted, two-byte header which provides the length of the packet in bytes. Upon reaching the end of the "list" or "chain" of packets, the transmitter will stop, update it's status bits and, if enabled, generate an interrupt.

Transmit Packet Data Formats

The packets to be transmitted, less preamble and CRC fields, are first loaded into the transmit buffer along with a two-byte header indicating the packet length in bytes. See Figure 8 for details. Multiple packets can be loaded at the same time provided there is room in the buffer. After the packets are loaded, the host initiates transmission by writing the number of packets just loaded into a register for this purpose on the chip. If the two-bank buffer configuration has been selected, the second bank can now be loaded with additional packets.

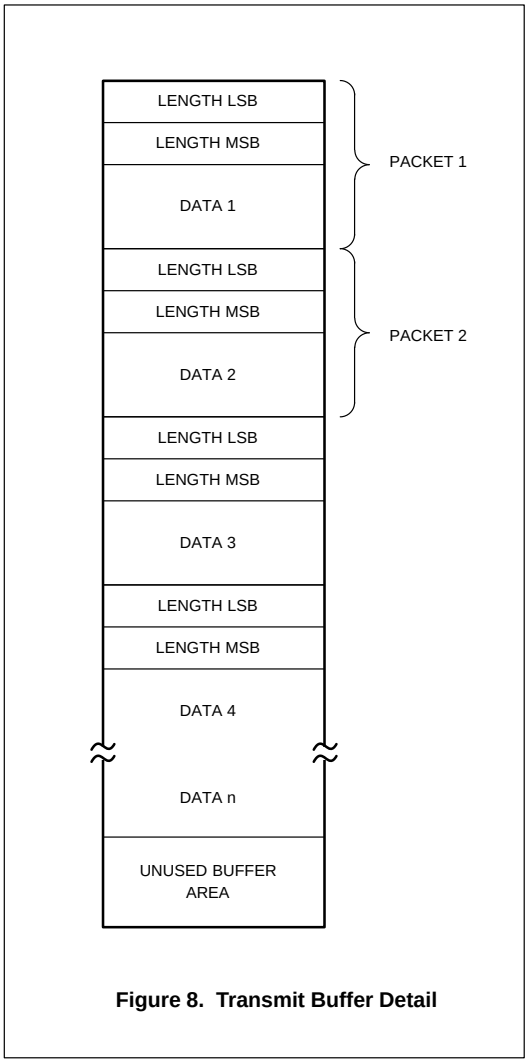


Figure 8. Transmit Buffer Detail

Receiver Access to Buffer

Once initialized and enabled, the receiver will automatically load any error-free incoming packets which pass the address filter into the receive buffer through an on-chip FIFO, appending a four-byte header to the front end which provides packet length and status. An interrupt can be provided to alert the host processor that a packet is available in the buffer. The host processor can read out receive packets as they become available. Continuous reception can continue as long as the receive buffer does not become full. If the host processor reads the receive packets from the buffer promptly, the buffer will not fill up. But if overflow does occur, an interrupt will be generated to indicate the problem. If this occurs, data should be read from the buffer to free space. As soon as space becomes available in the receive buffer, the receiver will automatically continue reception.

The receive buffer size can vary between a maximum of 62 kilobytes when 2 kilobytes is allocated for the transmit section and maximum memory size of 64 kilobytes is used, to a minimum of 4 kilobytes if 4 kilobytes is allocated for the transmit section and minimum memory size of 8 kilobytes is used. The receive section dynamically allocates space for each individual incoming data packet along an eight-byte "page" boundary. Each received packet is preceded by a four byte header which provides packet status and the length of that data packet. The data packets are linked or "chained" by internal pointers which use the length value in the packet header to calculate the length of the packet, and the starting address of the next packet. This buffer format is shown in Figure 9. Since NICE controls its dedicated buffer memory, FIFO size and depth are unimportant in this architecture, and need not be considered in system timing considerations.

A status bit in one of NICE's internal registers (RX BUF EMPTY) informs the host when one or more packets are resident in the receive buffer and available to be read. The host retrieves these packets from the buffer memory by successive reads of BMPR8. Once a data byte/word is read from the buffer memory, internal pointers are advanced to the next byte/word. As data is thus read by the system, that memory becomes available for reception of new packets. NICE automatically rejects an incoming packet if there is not enough buffer space to fully receive that packet. Therefore, there is no chance for packets already received to be "overrun" by incoming packets.

When DLCR5<5>, the ACPT BAD PKTS ("accept bad packets") bit is set to a "0" (disabled), a bad incoming packet causes NICE to release buffer space in which that packet is contained and reset its internal pointers so as to use that space for the next incoming packet. If this bit is set to a "1", the packet with a short length, alignment or CRC error will be accepted and the appropriate error bits in the status field of its header will be set. The same applies to DLCR5<3>, ACPT SHORT PKTS, which when high allows retention of packets below 60 bytes in length, excluding preamble and CRC (shorter than IEEE 802.3 minimum size).

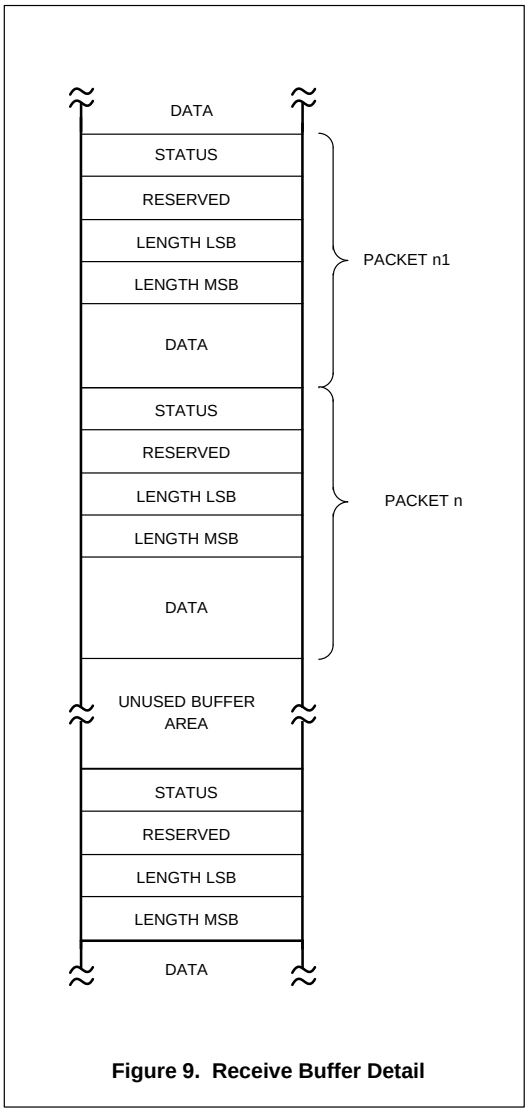


Figure 9. Receive Buffer Detail

Receive Packet Data Formats

Receive packets, less preamble and CRC fields, are stored in the buffer along with a four-byte header. The first byte gives status information, indicating errors, if any, that occurred during reception of the packet. Normally packets with errors are automatically discarded and eliminated from memory by the chip, but with a mode selection, the chip can allow reception of bad packets, with indication of their errors in the status byte of the header. The second byte of the header is reserved for possible future use. The last two bytes of the header give the byte count of the packet, less preamble and CRC. Refer to Figure 9.

SYSTEM INTERFACE

The system interface block provides the connection between NICE and the host CPU. NICE supports both 8-bit and 16-bit bus widths and byte or word transfers as determined by DLCR6<5>, SB/SW, the "system byte or system word" configuration bit. Depending on the type of host CPU, NICE will supply the data order, MSB or LSB first according to the setting of DCLR7<0> as described in the detailed register descriptions. NICE supports I/O-mapping, memory mapping, and burst or single transfer DMA modes. An interrupt output, $\overline{INT}$, is provided which may be programmed by the user to inform the CPU of transmit and receive status conditions requiring host processing.

Three sets of user-accessible registers are contained within the MB86960. All registers are accessible as bytes or words.

Register Access

All control and status registers on the NICE chip are accessible through its bus interface port, which can be I/O or memory mapped in the system. Eight of the registers in the set, whose addresses are xxx0H through xxx7H, are always directly accessible. For the remaining physical addresses, three different banks of registers can be accessed through indirect addressing of the banks (bank switching). The bank switching bits are part of the first eight registers which are permanently resident.

The bank-switched register group consists of three sets or "banks" of registers. One of the sets is for Node ID (Ethernet Address) and TDR diagnostics, another is the Hash Table for multicast address filtering, and the third is for buffer memory access. This third bank is normally selected most of the time, except during initialization or diagnostic routines, as access to the other registers is not needed during normal operation.

Buffer Access

Buffer Memory Port Register 8 (BMPR<8>) of the buffer memory port register set provides serial access to the receive and transmit buffers through on-chip FIFO's. This port can be accessed with 8-bit or 16-bit wide data. There is a separate FIFO for each direction of data transfer, so there is no complicated direction control needed. Writes to the transmit buffer can be interleaved with reads from the receive buffer if desired. All buffer memory pointers are automatically maintained by the chip, eliminating software overhead normally needed for this.

This port can be accessed with I/O instructions using register address xxx8H, or by using DMA. In the latter case, assertion of the $\overline{DACK}$ input is sufficient to select the port. Thus data can be transferred from host memory to the transmit buffer, or from the receive buffer to host memory using CPU string moves, single-transfer programmed I/O moves, or DMA. The choice should be made according to which is most efficient at a system level, taking into account that a speedy transfer process will result in the best performance. A slow transfer process may not be satisfactory because it might result in poor throughput and performance, and might allow the receive buffer to overflow, losing packets.

DMA Operation

The MB86960 supports both single cycle and burst DMA operation for transfers of data between the host system and the dedicated buffer memory. The DREQ and $\overline{DACK}$ signals are used for handshaking between the external DMA and NICE. There is also an "end of process" input pin which, when asserted by the system DMA controller during a transfer cycle, will terminate the DMA activity after the current cycle completes. If enabled for DMA interrupt, upon completion of the DMA activity, the chip will generate an interrupt.

Usually only one DMA operation will be run at a time, although the NICE chip could run two interleaving operations, one reading and one writing. There is only one DMA EOP bit, and only one DREQ pin and one $\overline{DACK}$ pin, so most hosts could not support more than one DMA operation at a time with NICE.

DMA Write (Transmit)

Transmit DMA Enable, TX DMA EN, BMPR12<0>, is set high to enable DMA operation for transfers of data packets from the host memory to NICE's transmit buffer. Burst transfers can also be enabled by invoking the DMA burst control register BMPR13<1:0>. When NICE is ready to begin to accept data from the host, NICE will assert its DMA request output, DREQ. The host responds by asserting DMA Acknowledge, DACK, followed by write enable, WE, and placing the data on the data bus. NICE will assert its RDY(RDY) output when it is ready to complete the current data transfer cycle (polarity of RDY(RDY) and EOP(EOP) inputs are independently programmable). NICE accepts that data byte/word into its bus write FIFO, and later moves it into buffer memory. At the close of a transfer cycle, the host negates WE. In burst mode, NICE will negate DREQ two cycles before the end of the burst. The host DMA will then complete the last two transfer cycles, then negate DACK to close the burst. To start another burst, NICE will re-assert DREQ. The number of DMA write cycles within one burst can be 1, 4, 8, or 12 data transfers (bytes or words) depending on the burst control bits BURST1, BURST0, BMPR13<1:0>.

The DMA controller may assert the end-of-process input, EOP(EOP), concurrently with the last data transfer cycle to indicate that the entire transfer process has been completed. This sets the DMA EOP bit in NICE which causes NICE to discontinue making further data requests. If enabled, the EOP(EOP) signal assertion can also generate an interrupt. When the DMA EOP bit, DLCR1<5>, is set, the INT pin will assert if DLCR3<5>, interrupt enable for DMA EOP , is high. This interrupt can be used by the host to initiate the actions for closing the process. Upon servicing the interrupt, if DMA EOP is high, the host should close the DMA process, reset the NICE chip's DMA logic and clear the interrupt by writing 00H to BMPR12. *Note: Clearing TX DMA EN must be done to close the transmit DMA process before attempting another DMA process. This is accomplished by writing 00H to BMPR12. When this is done, the DMA EOP bit will clear automatically, clearing the EOP status and interrupt, so it is not necessary to clear the interrupt separately.*

After finishing the loading of packets into the buffer, the host initiates packet transmission. This is done by loading the number of packets to be transmitted into the Transmit Packet Count Register, BMPR10<6:0>, and asserting the Transmit Start bit, TX START, of the same register, BMPR10<7>.

DMA Read (Receive)

NICE will indicate when it has receive packets to be read with status bits and/or interrupts. Before attempting to read a packet, the host processor first reads the RX BUF EMPTY bit, DLCR5<6>. If this bit is 0, there are one or more packets in the receive buffer to read. After reading each packet, the host will check this bit again to see if there are more.

Prior to beginning the transfer of a packet from NICE's receive buffer to host memory via DMA, the host must first read the four-byte receive packet header from the buffer to obtain the packet status and the length of the packet in bytes. Calculating from the packet length the number of DMA cycles needed to read the packet, the host will load that number into the cycle counter of the host DMA controller. The starting address in system memory will also be loaded into the DMA controller. Next, RX DMA EN, BMPR12<1>, is set high to enable DMA read operation to transfer the packet to host memory. When it is ready to begin, NICE asserts its DMA Request output, DREQ. The host responds by asserting DMA Acknowledge, DACK, followed by Read Enable, RD. NICE will assert its RDY output when it has placed the byte/word on the data bus and is ready to complete the data transfer cycle. The system memory will accept the data, then the host negates RD. NICE shifts the data down into its bus read FIFO, then moves its internal bus read pointer to point to the next byte/word in the buffer, moving it into the FIFO.

NICE will negate DREQ two cycles before the end of the burst. After the host negates DACK, if NICE can transfer more data, NICE will re-assert DREQ to repeat the process. The number of DMA read cycles in a burst can be 1, 4, 8, or 12 transfer cycles of data (bytes or words), depending on the burst control bits BURST1, BURST0, BMPR13<1:0>. The DMA controller may assert the end-of-process input, EOP(EOP), concurrently with the last byte/word data transfer to indicate that the entire process has completed. NICE will then discontinue making further data requests. RX DMA EN must be cleared when the DMA process is completed, and set again when the host desires to begin reading another packet from the receive buffer using DMA.

When EOP(EOP) is asserted by the host DMA controller, the DMA EOP bit, DLCR1<5>, will be set high, and an interrupt will also be generated, provided it is enabled by a high, DLCR3<5>. This interrupt can be used by the host to initiate the final actions to close the DMA process. The interrupt is cleared and the DMA is disabled and reset by writing 00H to the DMA Enable Register, BMPR12. *Note: Clearing RX DMA EN must be done to close the*

receive DMA process before attempting another DMA process. This is accomplished by writing 00H to BMPR12. When this is done, the DMA EOP bit will clear automatically, clearing the EOP status and interrupt, so it is not necessary to clear the interrupt separately.

CONTROL AND STATUS REGISTERS

The control and status registers on the NICE chip are accessed through direct register addresses xxx0H through xxxFH, and indirect or bank-switching address bits RBS1, RBS0, DLCR7<3:0>. Table 1 summarizes the addressing scheme. In system word mode, data can be transferred 16-bits at a time on the system bus, or 8-bits at a time by using the byte lane controls of NICE. When transferring in 16-bit mode to/from the registers, even direct addresses are used to select the registers. For example, to access the Transmit/Receive Status Registers, address xxx0H would be used. The transmit status would be on the low byte and the receive status on the high byte. Separate access of high and low bytes is achieved by using the appropriate byte-access processor instructions.

Table 1. Internal Register Address Map

RBS1,RBS0	SA3	SA2	SA1	SA0	ADDRESS	DESCRIPTION
XX	0	0	0	0	DLCR0[1]	TRANSMIT STATUS
XX	0	0	0	1	DLCR1	RECEIVE STATUS
XX	0	0	1	0	DLCR2	TRANSMIT INT ENABLE
XX	0	0	1	1	DLCR3	RECEIVE INT ENABLE
XX	0	1	0	0	DLCR4	TRANSMIT MODE
XX	0	1	0	1	DLCR5	RECEIVE MODE
XX	0	1	1	0	DLCR6	CONFIG 0
XX	0	1	1	1	DLCR7	CONFIG 1
00	1	0	0	0	DLCR8	NODE ID 0
00	1	0	0	1	DLCR9	NODE ID 1
00	1	0	1	0	DLCR10	NODE ID 2
00	1	0	1	1	DLCR11	NODE ID 3
00	1	1	0	0	DLCR12	NODE ID 4
00	1	1	0	1	DLCR13	NODE ID 5
00	1	1	1	0	DLCR14	TDR 0 (LSB)
00	1	1	1	1	DLCR15	TDR 1 (MSB)
01	1	0	0	0	HT8	HASH TABLE 0
01	1	0	0	1	HT9	HASH TABLE 1
01	1	0	1	0	HT10	HASH TABLE 2
01	1	0	1	1	HT11	HASH TABLE 3
01	1	1	0	0	HT12	HASH TABLE 4
01	1	1	0	1	HT13	HASH TABLE 5
01	1	1	1	0	HT14	HASH TABLE 6
01	1	1	1	1	HT15	HASH TABLE 7
10	1	0	0	0	BMPR8[2]	BUFFER MEMORY PORT
10	1	0	0	1	BMPR9	RESERVED
10	1	0	1	0	BMPR10	TRANSMIT START
10	1	0	1	1	BMPR11	16 COLLISIONS
10	1	1	0	0	BMPR12	DMA ENABLE
10	1	1	0	1	BMPR13	DMA BURST
10	1	1	1	0	BMPR14	SKIP PACKET
10	1	1	1	1	BMPR15	RESERVED
11	X	X	X	X	—	RESERVED

1. All registers are both word and byte accessible. In word mode, register bytes are paired to form words starting with registers 0 and 1. The odd-addressed byte becomes the high byte of the word
2. In word mode, BMPR8 is a 16-bit port. In byte mode, it is an 8-bit port. The byte ordering Is determined by DLCR7<0>

Table 2. Summary of Control and Status Bits: DLCR0-7, BMPR8-15

Register	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
TX STATUS DLCR0	TX DONE	NET BSY	TX-RX	CR LOST	0	COL	16 COL	0
RX STATUS DLCR1	RX PKT	BUS RD ERR	DMA EOP	RMT 0900H	SHORT ERR	ALIGN ERR	CRC ERR	RX BUF OVERFLO
TX INT ENABLE DLCR2	INT EN	0	0	0	0	INT EN	INT EN	0
RX INT ENABLE DLCR3	INT EN	INT EN	INT EN	INT EN	INT EN	INT EN	INT EN	INT EN
TX MODE DLCR 4	COL CTR 3	COL CTR 2	COL CTR 1	COL CTR 0	0	CNTRL	LBC	EN TX DEFER
RX MODE DLCR5	0	RX BUF EMPTY	ACPT BAD PKTS	40 BIT ADDR	ACPT SHORT PKTS	1	AF1	AF0
CONFIG0 DLCR6	DLC EN	1	SB/SW	BB/BW	TBS 1	TBS 0	BS 1	BS 0
CONFIG1 DLCR7	E/D CNF 1	E/D CNF 0	PWRDN	RDYPOL	RBS 1	RBS 0	EOPPOL	M..L/ L..M
BUF MEM PORT BMPR8	7	6	5	4	3	2	1	0
BUF MEM PORT BMPR9	15	14	13	12	11	10	9	8
TX START BMPR10	TX START	TX PKT CNT 6	TX PKT CNT 5	TX PKT CNT 4	TX PKT CNT 3	TX PKT CNT 2	TX PKT CNT 1	TX PKT CNT 0
16 COLLISIONS BMPR11	0	0	0	0	0	16 COL CNTRL 2	16 COL CNTRL 1	16 COL CNTRL 0
DMA ENABLE BMPR12	0	0	0	0	0	0	RX DMA ENABLE	TX DMA ENABLE
DMA BURST BMPR13	0	0	0	0	0	0	BURST 1	BURST 0
SKIP PACKET BMPR14	0	0	0	0	0	SKIP PKT	0	0
RESERVED BMPR15	0	0	0	0	0	0	0	0

Table 3. Summary of Control and Status Bits:DLCR8-15, HT8-15, Packet Buffer Headers

Register	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
NODE ID 0 DLCR8	7	6	5	4	3	2	1	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
NODE ID 5 DLCR13	47	46	45	44	43	42	41	40
TDR 0 DLCR14	7	6	5	4	3	2	1	0
TDR1 DLCR15	N/A (0)	N/A (0)	13	12	11	10	9	8
HASH TABLE 0 HT8	7	6	5	4	3	2	1	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
HASH TABLE 7 HT15	63	62	61	60	59	58	57	56
TX LENGTH LSB (TX HEADER 0)	7	6	5	4	3	2	1	0
TX LENGTH MSB (TX HEADER 1)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	10	9	8
PKT STATUS (RX HEADER 0)	N/A (0)	N/A (0)	GOOD PKT	RMT 0900H	SHORT ERR	ALIGN ERR	CRC ERR	RBUF OVRFLO
RESERVED (RX HEADER 1)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	N/A (0)
RX LENGTH LSB (RX HEADER 2)	7	6	5	4	3	2	1	0
RX LENGTH MSB (RX HEADER 3)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	N/A (0)	10	9	8

Type Descriptions (TYPE)

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

R: Readable bit

W: Writeable 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

DATA LINK CONTROL REGISTERS

Status and control bits for the transmitter and receiver sections of the NICE chip are located in the first eight data link control registers, DLCR0–7. See Tables 4–11 for details.

Transmit Status Register

This register provides transmit status for the host processor. The system can enable interrupts based on the assertion (going high) of bits 7, 2, and/or 1 of this register by setting the corresponding enable bits in the Transmit Interrupt Enable register. See Table 4.

Bits 7, 2 and 1 in this register, the bits that can generate interrupts, are cleared by writing 1 to the bit. Writing 0 to the bit has no effect, only the NICE chip control logic can set these bits. Clearing the bit that caused the interrupt will clear both the bit itself and the interrupt. Since two or more status conditions can occur at one time, the interrupt routine must read all of them and act on all of those that are set.

One method for clearing interrupts is to read the contents of the status register, then write the same value back to the register, thus clearing all bits which were set. Another technique is to clear each status bit separately by writing its mask to the register. This might be done as the corresponding interrupt service is performed. *Note: Wholesale clearing of all status bits by writing FFH to the register is not recommended, since this may clear status which has just been set and not yet read by the system. The transmitter must be idle and TX DONE, DLCR0<7>, must be cleared by writing 1 to it before starting the transmitter (by writing 1 to TX START, BMPR10<7>).*

Table 4. DLCR0 — Transmit Status Register

BIT	SYMBOL	TYPE	DESCRIPTION
7	TX DONE	R C 0	**TRANSMIT DONE:** This bit is set high when all packets in the active transmit buffer have been successfully transmitted to the LAN media, or skipped due to excessive collisions. Can generate interrupt if enabled by DLCR2<7>.
6	NET BSY	R	**NET BUSY:** This is a real-time image of the Carrier Sense signal of the receiver.
5	TX-RX	R 0	**TRANSMIT PACKET RECEIVED:** Indicates that a good packet was received by NICE shortly after transmission was completed. This is used to indicate self-reception of the packet. This bit is cleared as each transmission begins.
4	CR LOST	R 0	**CARRIER LOST:** This bit is set if the receive carrier sense input is negated during a packet transmission. This can be caused by a collision or a shorted LAN medium. It is automatically cleared as each transmission begins.
3	0	N 0	**RESERVED:** Write 0.
2	COL	R C 0	**COLLISION:** This bit will assert during transmission of a data packet if a collision occurs on the network. The buffer controller will automatically retransmit the current packet after collisions up to 15 times. The user may read the number of consecutive collisions in the collision counter, DLCR4<7:4>. Can generate interrupt if enabled by DLCR2<2>.
1	16 COL	R C 0	**16 COLLISIONS:** This bit is set after the sixteenth unsuccessful transmission of the same packet. Can generate interrupt if enabled by DLCR2<1>.
0	0	R O	**RESERVED:** Write 0.

Receive Status Register

This register contains eight status bits which can generate interrupts if enabled by the corresponding bit in DLCR3. Five of these bits report the status of the most recently received packet that was accepted for storage in the receive buffer. Bit 7, RX PKT, is set whenever a new packet is successfully received and stored in the buffer. One bit reports reception of a special packet with 0900H in its 'type' field. Other bits in this register report buffer overflow, DMA end-of-process and bus read error. Bits 1, 2 and 3 indicate errors, if any, detected in the packet. If ACPT BAD PKTS, DCLR5<5> and/or ACPT SHORT PKTS, DLCR5<3> are set to 1 allowing acceptance of a bad packet, these error bits will be stored in the status byte of the receive packet header. If DLCR5<5> and DLCR5<3> are both 0, all packets with detected errors will be discarded automatically, and removed from the buffer.

The bits in this register are cleared by writing 1 to the bit. Writing 0 to the bit has no effect. Only the NICE chip control logic can set these bits. Clearing the bit that caused the interrupt will clear both the bit itself and the interrupt. Since two or more status conditions can occur at one time, the interrupt routine must read all of them and act on all of those that are set. See Table 5.

One method for clearing interrupts is to read the contents of the Transmit and Receive Status Registers, then write the same value back to the registers, thus clearing all bits which were set. Another technique is to clear each status bit separately by writing its mask to the register as the corresponding interrupt service is performed. *Note: Wholesale clearing of all status bits by writing FFH to the register is not recommended, since this may clear status which has just been set and not yet read by the system.*

Table 5. DLCR1 — Receive Status Register

BIT	SYMBOL	TYPE	DESCRIPTION
7	RX PKT	R C 0	**RECEIVE PACKET:** Set when a new receive packet is stored in the Receive Buffer. Can generate interrupt if enabled by DLCR3<7>.
6	BUS RD ERR	R C 0	**BUS READ ERROR:** Set when a ready response cannot be issued within 2.4 µs after the $\overline{RD}$ signal is asserted. Occurs when reading an empty buffer. Can generate interrupt if enabled by DLCR3<6>.
5	DMA EOP	R C 0	**DMA END OF PROCESS:** Set when the host DMA asserts the EOP pin indicating that the process is finished. When set, inhibits further assertion of DREQ. Cleared by writing 00H to BMPR12. *Do not clear by writing 1 to this bit as this may result in unwanted DREQ.* Can generate interrupt if enabled by DLCR3<5>.
4	RMT 0900H	R C 0	**REMOTE CONTROL PACKET RECEIVED:** This bit is set if a packet is received with 0900H in its Length/Type Field (two bytes following the source address, received MSB first). Can generate interrupt if enabled by DLCR3<4>.
3	SHORT ERR	R C 0	**SHORT PACKET ERROR:** Set when a packet is received with less than 60 bytes, excluding its Preamble and CRC fields. Such a packet usually indicates a collision has truncated its original length, since IEEE 802.3 minimum length is 60 bytes. Can generate interrupt if enabled by DLCR3<3>. See also Table 8.
2	ALIGN ERR	R C 0	**ALIGNMENT PACKET ERROR:** This bit will assert if a packet is received with an alignment error, meaning there were 1 to 7 extra bits at the end of the packet. Such an occurrence usually indicates a collision, or a faulty transceiver. Can generate interrupt if enabled by DLCR3<2>. See also Table 8.
1	CRC ERR	R C 0	**CRC PACKET ERROR:** This bit is set if a packet is received with a CRC error. This usually indicates a collision has corrupted the packet. Can generate interrupt if enabled by DLCR3<1>. See also Table 8.
0	RX BUF OVRFLO	R C 0	**RECEIVE BUFFER OVERFLOW:** This bit will be set if the receive buffer becomes full and must reject a packet for lack of space. Can generate interrupt if enabled by DLCR3<0>. Does not get set in loopback mode.

Transmit Interrupt Enable Register

This register contains the bits which enable or mask the status bits in DLCR0 from generating interrupts. Only bits 7, 2 and 1 can generate interrupts. The other bits are not used. See Table 6.

Table 6. DLCR2 — Transmit Interrupt Enable Register

BIT	SYMBOL	TYPE	DESCRIPTION
7	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables TX DONE to generate interrupt. See also DLCR0<7>.
6	0	N 0	**RESERVED:** Write 0.
5	INT EN	N 0	**RESERVED:** Write 0.
4	0	N 0	**RESERVED:** Write 0.
3	0	N 0	**RESERVED:** Write 0.
2	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables COL to generate interrupt. See also DLCR0<2>.
1	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables 16 COL to generate interrupt. See also DLCR0<1>.
0	0	N 0	**RESERVED:** Write 0.

Receive Interrupt Enable Register

This register provides control for enabling or masking interrupts based on the assertion of status bits in DCLR1, the Receive Status Register. See Table 7.

Transmit Mode Register

This register contains two control bits associated with transmission, a general-purpose control bit which drives a pin on the chip, and a collision counter. See Table 9.

Table 7. DLCR3 — Receive Interrupt Enable Register

BIT	SYMBOL	TYPE	DESCRIPTION
7	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables RX PKT to generate interrupt. See also DLCR1<7>.
6	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables BUS RD ERR to generate interrupt. See also DLCR1<6>.
5	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables DMA EOP to generate interrupt. See also DLCR1<5>.
4	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables RMT 0900H to generate interrupt. See also DLCR1<4>.
3	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables SHORT ERR to generate interrupt. See also DLCR1<3> and Table 8.
2	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables ALIGN ERR to generate interrupt. See also DLCR1<2> and Table 8.
1	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables CRC ERR to generate interrupt. See also DLCR1<1> and Table 8.
0	INT EN	R W 0	**INTERRUPT ENABLE:** When high, enables RBUF OVRFLO to generate interrupt. See also DLCR1<0>.

Table 8. Network Error Monitoring Modes

ACPT BAD PKTS DLCR5<5>	ACPT SHORT PKTS DLCR5<3>	INT EN SHORT ERRORS DLCR3<3>	INT EN ALIGN ERRORS DLCR3<2>	INT EN CRC ERRORS DLCR3<1>	Mode Description
0	0	0	0	0	Normal non-monitor mode.
0	0	1/0	1/0	1/0	Error interrupts only, if enabled.
0	1	0	1/0	1/0	Save short packets if otherwise error free in buffer; interrupts only for alignment and CRC errors, if enabled. RX PKT will be set high if short packet received.
1	0	0	0	0	Save packets with short, alignment and/or CRC errors in buffer; RX PKT will be set high if packet with error received.
All others					Do Not Use.

Note: Packet acceptance requires both error filter acceptance and address filter acceptance.

Table 9. DLCR4 — Transmit Mode Register

BIT	SYMBOL	TYPE	DESCRIPTION
7	COL CTR 3	R 1	**COLLISION COUNT 3:** DLCR4<7:4> plus 1 indicates the number of consecutive collisions encountered by the current transmit packet. (Read only). See Table 10.
6	COL CTR 2	R 1	**COLLISION COUNT 2:** DLCR4<7:4> plus 1 indicates the number of consecutive collisions encountered by the current transmit packet. (Read only). See Table 10.
5	COL CTR 1	R 1	**COLLISION COUNT 1:** DLCR4<7:4> plus 1 indicates the number of consecutive collisions encountered by the current transmit packet. (Read only). See Table 10.
4	COL CTR 0	R 1	**COLLISION COUNT 0:** DLCR4<7:4> plus 1 indicates the number of consecutive collisions encountered by the current transmit packet. (Read only). See Table 10
3	0	N 0	**REVERSED:** Write 0.
2	CNTRL	R W 1	**CONTROL OUTPUT:** The inverse of this is bit is output for general use on pin 95.
1	LBC	R W 1	**LOOPBACK CONTROL:** This bit controls the loopback function of the NICE encoder/decoder. A 0 in this bit places the chip in internal loopback mode.
0	EN TX DEFER	R W 0	**ENABLE TRANSMIT DEFER:** Program this bit low for normal network operation. When high, the transmitter will not defer to traffic on the network.

Table 10. Collision Count

Collision Count	16 COL DLCR0<1>	DLCR4<7>	DLCR4<6>	DLCR4<5>	DLCR4<4>	COL DLCR0<2>
0	0	0	0	0	0	0
1	0	0	0	0	1	1
2	0	0	0	1	0	1
3	0	0	0	1	1	1
. . .	. . .	. . .	. . .	. . .	. . .	. . .
15	0	1	1	1	1	1
16	1	0	0	0	0	1

Receive Mode Register

This register contains six bits which control receiver function, and one receive buffer status bit. See Table 11.

The status bit, RX BUF EMPTY (Receive Buffer Empty), is very important to the software routine which reads receive packets from the buffer. It tells the host routine whether there are any packets in the receive buffer which are complete and ready to read. In a multi-tasking system, this indicator would be used in conjunction with an interrupt when RX PKT asserts, which means a packet has arrived in memory. The interrupt would be used to start the routine which reads packets from the buffer. As this routine begins, the interrupt on RX PKT can be disabled to prevent unneeded interrupts. After the first packet is read from the buffer, the RX BUF EMPTY bit would be read to see if any more packets have come in (packets may, at times, arrive in bursts). If the buffer is not empty, another packet would be read out, and this procedure repeated until the buffer is empty. After emptying the buffer, the host clears RX PKT, then re-enables interrupts on RX PKT, checks the buffer status one more time (since a packet can come in at any time), then exits to do other tasks.

Two of the control bits allow reception of packets with certain types of errors. The ACPT BAD PKTS bit, when set, causes the receiver to retain and store in the buffer packets with CRC, alignment and/or short length errors provided there was no indication of collision during reception. Likewise, the ACPT SHORT PKTS bit, when set, allows the retention of short packets down to and including only six bytes in length, excluding preamble and CRC, provided there was no indication of collision during reception and no alignment or CRC error. Under normal operation, packets with less than 60 bytes, the IEEE 802.3 lower limit, would be discarded. These functions are provided for diagnostic purposes. Packets are accepted only if both the address filter and the error filter are passed. Packets with no content errors, i.e., short, alignment or CRC, are accepted without regard to collision indicators.

Table 11. DLCR5 — Receive Mode Register

<table>
<tr><th>BIT</th><th>SYMBOL</th><th>TYPE</th><th>DESCRIPTION</th></tr>
<tr><td>7</td><td>0</td><td>N
0</td><td>RESERVED: Write 0.</td></tr>
<tr><td>6</td><td>RX BUF EMTY</td><td>R
1</td><td>RECEIVE BUFFER EMPTY: Status bit which indicates that the receive buffer does not have any complete packets to read. (Read only).</td></tr>
<tr><td>5</td><td>ACPT BAD PKTS</td><td>R
W
0</td><td>ACCEPT PACKETS WITH ERRORS: When set high, allows packets with CRC, alignment and/or short length errors to be saved into the receive buffer for analysis, provided there was no indication of collision during reception. Otherwise such packets would be discarded automatically by the receiver and removed from the buffer. See also Table 8.</td></tr>
<tr><td>4</td><td>40 BIT ADDR</td><td>R
W
0</td><td>40 BIT ADDRESS: When set high, instead of the customary 48-bit NODE ID address filter, only the first 40 bits of the NODE ID are compared (NODE ID 0–4).</td></tr>
<tr><td>3</td><td>ACPT SHORT PKTS</td><td>R
W
0</td><td>ACCEPT SHORT PACKETS: When set high, allows short packets (packets with less than 60 bytes, excluding Preamble and CRC, i.e. below IEEE minimum length) down to 6 bytes, excluding preamble and CRC, to be saved into the receive buffer, provided there was no CRC or alignment error and no indication of collision during reception. Otherwise such packets would be discarded automatically by the receiver and removed from the buffer. See also Table 8.</td></tr>
<tr><td>2</td><td>1</td><td>N
0</td><td>RESERVED: Write 1.</td></tr>
<tr><td>1</td><td>AF1</td><td>R
W
0</td><td>ADDRESS FILTER MODE: AF1 and AF0, DLCR5<1:0>, control the address filtering on incoming packets. See table below under AF0.</td></tr>
<tr><td>0</td><td>AF0</td><td>R
W
1</td><td>ADDRESS FILTER MODE: AF1 and AF0, DLCR5<1:0>, control the address filtering on incoming packets:
<table>
<tr><th>AF1</th><th>AF0</th><th>Description of Address Accepted</th></tr>
<tr><td>0</td><td>1</td><td>NODE ID, Broadcast, Multicast & 2nd-24th bits of NODE ID</td></tr>
<tr><td>1</td><td>0</td><td>NODE ID, Broadcast, Multicast & Hash Table</td></tr>
<tr><td>0</td><td>0</td><td>Reject all packets</td></tr>
<tr><td>1</td><td>1</td><td>Accept all packet</td></tr>
</table>
Note: Self reception of broadcast and multicast packets prohibited except in “Accept all packets” mode. When LBC is low (loopback mode) broadcast packets can be self-received, except in “Reject all packets” mode.</td></tr>
</table>

Configuration Registers 0 and 1

Basic system configuration bits are found in these two registers. Among the configuration controls found here are physical packet buffer memory size, partitioning between transmit and receive buffers, widths of memory and system buses, byte lane control, chip configuration and power down control. Most of these configuration parameters will be programmed only during initialization after power start and hardware reset. See Tables 12 and 13.

Table 12. DLCR6 — Configuration Register 0

<table>
<tr><th>BIT</th><th>SYMBOL</th><th>TYPE</th><th>DESCRIPTION</th></tr>
<tr><td>7</td><td>DLC EN</td><td>R
W
1</td><td>ENABLE DATA LINK CONTROLLER: When low, enables the receiver and transmitter sections of the NICE chip. This bit must be set high during initialization and later set low to enable loopback testing and operation of the network. Program NODE ID only when this bit is high.</td></tr>
<tr><td>6</td><td>1</td><td>N
0</td><td>RESERVED: Write 1.</td></tr>
<tr><td>5</td><td>SB/SW</td><td>R
W
1</td><td>SYSTEM BYTE/WORD BUS WIDTH: When high, system bus will operate in 8-bit data mode; when low, 16-bit data mode is selected. See also BB/BW below.</td></tr>
<tr><td>4</td><td>BB/BW</td><td>R
W
1</td><td>BUFFER BYTE/WORK WIDTH: When high, buffer memory will operate in 8-bit data mode; when low, 16-bit data mode is selected. See table for allowable combinations with SB/SW:
<table>
<tr><th>SB/SW</th><th>BB/BW</th><th>SYSTEM</th><th>BUFFER</th></tr>
<tr><td>0</td><td>0</td><td>word</td><td>word</td></tr>
<tr><td>0</td><td>1</td><td>word</td><td>byte</td></tr>
<tr><td>1</td><td>0</td><td colspan="2">Do not use</td></tr>
<tr><td>1</td><td>1</td><td>byte</td><td>byte</td></tr>
</table></td></tr>
<tr><td>3, 2</td><td>TBS 1, 0</td><td>R
W
01</td><td>TRANSMIT BUFFER SIZE: Selects size of Transmit Buffer(s). See table:
<table>
<tr><th>TBS
1,0</th><th>No. TX
BUFS</th><th>Size each
TX BUF</th><th>Total Size
TX BUF</th></tr>
<tr><td>00</td><td>1</td><td>2KB</td><td>2KB</td></tr>
<tr><td>01</td><td>2</td><td>2KB</td><td>4KB</td></tr>
<tr><td>10</td><td>2</td><td>4KB</td><td>8KB</td></tr>
<tr><td>11</td><td>2</td><td>8KB</td><td>16KB</td></tr>
</table></td></tr>
<tr><td>1, 0</td><td>BS1, 0</td><td>R
W
10</td><td>BUFFER SIZE: Selects physical size of total SRAM buffer memory for both transmit and receive functions. See table:
<table>
<tr><th>BS1</th><th>BS0</th><th>SRAM Size</th></tr>
<tr><td>0</td><td>0</td><td>8KB</td></tr>
<tr><td>0</td><td>1</td><td>16KB</td></tr>
<tr><td>1</td><td>0</td><td>32KB</td></tr>
<tr><td>1</td><td>1</td><td>64KB</td></tr>
</table></td></tr>
</table>

Table 13. DLCR7 — Configuration Register 1

<table>
<tr><th>BIT</th><th>SYMBOL</th><th>TYPE</th><th>DESCRIPTION</th></tr>
<tr><td>7, 6</td><td>E/D
CNF
1,0</td><td>R
W
00</td><td>ENCODER/DECODER CONFIGURATION: Selects the operating mode of the controller-encoder/decoder functions and their interface. See table below and pin descriptions for the monitor pins, TXD, TCK, TEN, LBC, RXD, RCK, CRS and COL:
<table>
<tr><th>E/D CNF 1</th><th>E/D CNF 0</th><th>Registers</th></tr>
<tr><td>0</td><td>0</td><td>Normal NICE: Internal encoder/decoder active. Monitor pins inactive.</td></tr>
<tr><td>0</td><td>1</td><td>NICE + Monitor: Internal encoder/decoder active. Monitor pins outputting all controller/encoder/ decoder interface signals</td></tr>
<tr><td>1</td><td>0</td><td>Encoder/Decoder Bypass: Internal encoder/ decoder not used. Monitor pins can be used to interface controller to external encoder/decoder.</td></tr>
<tr><td>1</td><td>1</td><td>Encoder/Decoder Test: Controller inactive, encoder/decoder active. Monitor pins used to test encoder/decoder.</td></tr>
</table></td></tr>
<tr><td>5</td><td>PWRDN</td><td>R
W
1</td><td>POWER DOWN: When set high, enables power to the chip for all functions; when set low, places chip in power down mode for power conservation.</td></tr>
<tr><td>4</td><td>RDYPOL</td><td>R
0/1</td><td>READY PIN POLARITY: Reads the state of the RDY POL pin 94.</td></tr>
<tr><td>3, 2</td><td>RBS 1, 0</td><td>R
W
00</td><td>REGISTER BANK SELECT: Provides the indirect address for selection of one of the three sets of registers to access when the physical register address is xxx8H–xxxFH.
The lower 7 registers are not bank-selectable. See table:
<table>
<tr><th>RBS1</th><th>RBS0</th><th>Registers</th></tr>
<tr><td>0</td><td>0</td><td>DLCR0-7, DLCR8-F</td></tr>
<tr><td>0</td><td>1</td><td>DLCR0-7, HT8-F</td></tr>
<tr><td>1</td><td>0</td><td>DLCR0-7, BMR8-F</td></tr>
<tr><td>1</td><td>1</td><td>RESERVED</td></tr>
</table></td></tr>
<tr><td>1</td><td>EOPPOL</td><td>R
W
0</td><td>EOP PIN SIGNAL POLARITY: When high, the EOP pin is active-high; when low, EOP is active-low.</td></tr>
<tr><td>0</td><td>M..L/
L..M</td><td>R
W
0</td><td>BYTE ORDER CONTROL: Selects byte lane ordering for packet data in the buffer (applies only in System Word Mode). In both Most..Least and Least..Most modes, the first and second bytes of the packet will appear in the same word on the system bus. When this bit is high (M..L mode), the first and all odd-numbered bytes of a packet and its header will appear on the high byte of the system bus. Note that header bytes are also swapped.</td></tr>
</table>

Note to software engineers regarding NICE/EtherStar™ compatibility: If you desire to use the same node driver for Fujitsu's NICE and EtherStar controllers, the driver can determine which chip is being used by reading DLCR7 and/or DLCR6 after hardware reset. NICE will read 30B6H or 20BGH (30 or 20 for DLCR7 and B6 for DLCR6); EtherStar will read 0000H.

Power-down mode saves power when the device is not in use. When ready to place the NICE chip in Power Down Mode, first write 1 to DLCR6<7>, DLC EN, to turn the receiver and transmitter off, then write 0 to DLCR7<5>, PWRDN. To exit the power-down mode, write 1 to PWRDN. Register contents will be preserved, unless a hardware reset is issued. Hardware reset will also terminate the power-down mode.

Byte order control provided by the Most..Least/ Least..Most bit, DLCR7<0>, provides compatibility with various higher-level protocols, such as TCP/IP and XNS. These protocols may have different transmission order of the bytes within a word. When M..L/L..M is low, the least significant byte of the word is transmitted first, followed by the most significant. When M..L/L..M is set high, the byte order is reversed. This feature applies only when the system bus is operated in 16-bit (word) mode.

The byte order control works by reversing or not reversing the bytes of all words as they pass between the buffer memory and the system bus. Thus all data stored in the transmit buffer or retrieved from the receive buffer is affected, including the nontransmitted headers. The NICE registers, other than the Buffer Memory Port, BMPR8:9, are not affected by this control bit. Care must be taken in the software driver code to reverse the header information as well as the packet data when using this feature. Examples follow.

Example of using Least..Most Byte Ordering:

System Bus

Transmit Packet:

High Byte	Low Byte
TX Length, high byte	TX Length, low byte
Destination Addr, 2nd byte	Destination Addr, 1st byte…
Source Addr, 2nd byte	Source Addr, 1st byte…
Length Field, low byte*	Length Field, high byte*
Data Field, 2nd byte	Data Field, 1st byte…

Receive Packet:

High Byte	Low Byte
Unused; reserved	Receive Packet Status
RX Length, high byte	RX Length, low byte
Destination Addr, 2nd byte	Destination Addr, 1st byte…
Source Addr, 2nd byte	Source Addr, 1st byte…
Length Field, low byte*	Length Field, high byte*
Data Field, 2nd byte	Data Field, 1st byte…

Example of using Most..Least Byte Ordering:

System Bus

Transmit Packet:

High Byte	Low Byte
TX Length, low byte *	TX Length, high byte*
Destination Addr, 1st byte	Destination Addr, 2nd byte…
Source Addr, 1st byte	Source Addr, 2nd byte…
Length Field, high byte	Length Field, low byte
Data Field, 1st byte	Data Field, 2nd byte…

Receive Packet:

High Byte	Low Byte
Receive Packet Status	Unused; reserved
RX Length, low byte *	RX Length, high byte*
Destination Addr, 1st byte	Destination Addr, 2nd byte…
Source Addr, 1st byte	Source Addr, 2nd byte…
Length Field, high byte	Length Field, low byte
Data Field, 1st byte	Data Field, 2nd byte…

Note: Asterisk indicates numerically reversed byte ordering.

NODE ID REGISTERS

The Node ID Registers are accessed in register bank "0" at register addresses xxx8H–xxxDH. During initialization of the node, the unique Ethernet address assigned to the node is loaded into these registers. The first register at xxx8H corresponds to the first byte of the Node ID, which corresponds to the first address byte to be received as a packet arrives from the network. If the chip is configured to do so in its Address Filter mode bits, DLCR5<1:0>, the destination address field of an incoming packet will be compared to the Node ID stored in these registers. If there is a match, provided the packet passes the error filter, it will be accepted.

These registers are readable as well as writable, but they should not be accessed while the receiver is enabled. To avoid interaction with the receiver, access these registers only when DLC EN is 1. It is recommended that they be written and read only during initialization before enabling the receiver, i.e. before writing 0 to DLC EN, DLCR6<7>.

The address contained in these registers is used only for receive (destination) address filtering, not for the source address of outgoing packets. Outgoing packet addresses must be provided by the system as part of the packet data.

Within each byte, the bits are transmitted and received on the network least-significant bit first. See Table 3 for the transmission bit order, which follows the bit numbering in this table.

Time Domain Reflectometry (TDR) Counter

The TDR Counter can be used to get a rough indication of the location of a fault on the network, if one exists. When a node transmits, a short or open on the network would cause a reflected signal to the node's receiver which can sometimes be detected. The reflection will cause the carrier sense to fail and/or a false collision to be detected. This affect, time domain reflectometry, can be used to estimate the distance along the network cable from the node to the fault. The TDR Counter counts the number of bits transmitted before either a collision occurs, or carrier sense is lost, whichever comes first. If neither occur during transmission of the packet, the count is cleared. The amount of elapsed time this represents is two (2) times the signal delay from the node to the fault. An open on the network will usually cause a false collision, whereas a short is more likely to cause loss of carrier sense.

The TDR Count comes from DLCR14 and 15. DLCR14 is the least-significant byte, DLCR15 the most-significant. Only the lower 14 bits of the counter are equipped, which is more than is needed for an IEEE or Ethernet LAN. (The top two bits, DLCR15<7:6>, are always 0.)

To perform the TDR test for a fault, first enable interrupts for transmitter done (TX DONE). This is done by setting DLCR2<7> high. (An alternative to using the interrupt would be polling the TX DONE bit looking for a high level.) Set the 16 Collisions Register, BMPR11, to 07H for this test (no halt, skip failed packet). Clear all status bits by writing FF86H to the Receive and Transmit Status Registers. Next, transmit, or attempt to transmit, a packet of 600 bits or more in length. Up to 16 attempts may be made automatically if collisions are being indicated. Upon completion of the transmission attempt(s) TX DONE will go high, generating an interrupt if enabled. When this occurs, read the Transmit Status Register and the TDR Register.

Interpreting the results: If the count is zero, no fault was detected. If the count is greater than zero but smaller than the packet length, it may indicate a cable fault. If the count is less than 525, there may have been a real collision occurring during the test. Real collisions normally occur within the first 65 bytes of the packet, including preamble. Take note of the error messages, COL and CR LOST. COL high suggests a cable open, whereas CR LOST suggests a short. It is best to repeat the measurement several times, then throw out the anomalous values, if any, and average the rest. A cluster of readings at about the same value is a strong indicator of a valid fault measurement. If such a cluster of readings occurs, multiply the average of the cluster by 39 feet to estimate the distance from the node to the fault. (39 ft. = (100 ns x .8 x 186,282 mi/s x 5280 ft/mi)/2 ... this assumes the network is mostly coaxial cable with signal propagation speed of approximately .8 x C, the speed of light.)

HASH TABLE

The Hash Table provides a means for filtering incoming multicast packets so that the host processor does not have to process ones that are not of interest. The principle employed in this filtering scheme was originally developed by computer science to arrange a large number of elements of an array or database in such a way that facilitates searching for elements associated with a given key or datum. The 'hash function' is a mathematical or logical function which maps all possible elements in a domain onto a smaller domain called the 'hash table'.

As an example, suppose the following hashing function is used: "Treating the multicast address as a non-negative 48-bit integer, divide this number by 64 and take the remainder." This function will map all multicast addresses into a 64-element hash table since the remainder can be only the integer values 0 through 63. Applying this hashing function results in taking the least-significant 6 bits of the multicast address as an integer. In the hash table, for each element, 0 through 63, a single bit is stored which indicates whether the address is to be accepted (1) or rejected (0). If, for example, the node belongs to three multicast groups, only three or fewer of the hash table elements will store 1's, the rest 0's. The scheme allows the acceptance of any number of the addresses, including all of them. The limitation is that there may be addresses not of interest used on the network which also fall into the 'accept' elements, so in this case the filtering is imperfect. But in any case, most of the nonapplying addresses can be filtered out in this way.

The actual hashing function used in the NICE chip is this: "Calculate the CRC on the multicast address and take the high-order 6 bits of this calculation". The six bits are used to address the elements of the hash table. If a 1 is stored in an element of the table, associated packets are accepted. The hash filter criterium is only used on multicast addresses, which all start with a 1. Node ID's, which start with a 0, are not filtered by the hash filter. The broadcast address, a special case of the multicast set wherein all the bits are 1's, will be accepted anyway unless the "Reject all packets" mode is selected.

Figure 12 shows the register core, a modified shift-right register, used in generating and checking CRC's. Whereas some controllers share a single such core between transmitter and receiver, NICE has two of these, one for the generator and one for the checker, allowing both to operate concurrently for self-receive. To begin the calculation, the register is first set to all 1's. For the generator case, as the packet is transmitted, the data is clocked serially into the left-hand end of the register starting with the 48 bits of the destination address (the preamble is skipped). After the last bit of the 'data' field is clocked into the register, the CRC calculation is finished. The feedback line is then forced low and the register becomes a simple shift-right register. Its contents are then shifted out serially and transmitted, appending the CRC to the end of the packet. For the CRC checker, the calculation starts out the same way as for CRC generation feeding the incoming data into the register. But in this case, the CRC field of the packet is also fed into the calculation. The result is a fixed constant in the register if no CRC error has occurred.

For the Hash Filter, after the last bit of the destination address has been clocked into the register, the left-hand six register bits are stored in another register used to address the Hash Table elements. The left-most bit is most significant. The left-most three bits are used as the Hash Table register address and the right-most three as the bit address within a register byte. Having selected a Hash Table element in this way, a 1 in the table will indicate the packet is to be accepted, provided it is a multicast packet (first bit of destination address must be 1), and passes the error filters.

The hash filter is only employed when the address filter mode select bits, AF1, AF0 are 0, 1, selecting the "NODE ID, Broadcast, Multicast + Hash Table" mode. Like the NODE ID registers, the Hash Table registers should only be accessed when the receiver is disabled, i.e. when $\overline{\text{DLC EN}}$ is high, to avoid interaction with the receiver. There are eight bytes of registers in the Hash Table containing the 64 1-bit elements (refer to Table 1 for location). Source code software examples showing how to calculate entries for the Hash Table are available from Fujitsu Microelectronics, Semiconductor Division, in C and assembly language.

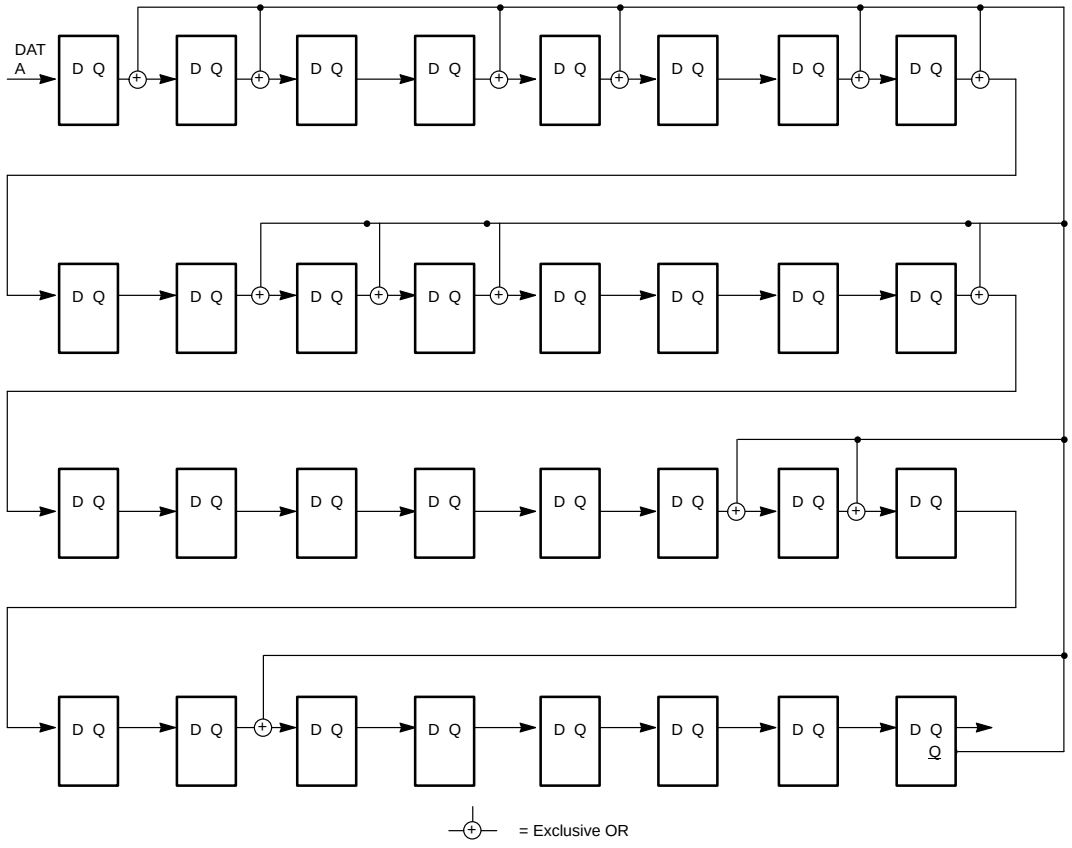


Figure 10. CRC Register Core

TRANSMIT AND RECEIVE PACKET HEADERS

Both transmit and receive packets have headers stored with them in the buffer memory which are not part of the transmitted packet. These headers precede the packet data.

Transmit Packet Length

An 11-bit integer indicating the number of bytes in the packet to be transmitted, excluding preamble and CRC fields which are generated by the NICE chip. See Table 3 for bit locations, and description for DLCR7<0>, M..L/L..M.

Receive Packet Status

The receive packet header consists of one byte of packet status, an unused byte and two bytes (11 bits) for packet length in bytes. See Table 3 for location of the bits, and description for DLCR7<0>, M..L/L..M. Bits 1, 2, 3 and 4 of the status byte are an image of the same bits in the Receive Status Register, DLCR1, with respect to the packet that follows. Bit 5 is the GOOD PKT bit. A 1 in this bit location indicates no errors were detected in the packet. Bits 0, 6 and 7 are unused and are always 0.

Receive Packet Length

The third and fourth bytes of the receive packet header indicate the total number of bytes in the stored packet data. See Table 3 for location of the bits, and description for DLCR7<0>, M..L/L..M.

BUFFER MEMORY PORT REGISTERS

The Buffer Memory Port Registers, BMPR8–15, provide the host access to the buffer memory and certain control functions. It is recommended that this bank be selected most of the time the chip is on the network for convenience in accessing the buffer memory and the control bits (Register Bank Select = 11 binary). Refer to Table 1 for location of these registers.

Writing a byte/word to BMPR8, the buffer port, transfers that data to the currently addressed location in the transmit buffer and increments the transmit buffer pointer to point to the next byte/word. Reading a byte/word from this port transfers the contents of the currently-addressed location in the receive buffer to the host and increments the receive buffer pointer to point to the next byte/word. BMPR9 is used only in word mode as the high byte of the word. In word mode, all transfers must be 16-bits wide, as the Buffer Port does not support byte-wide transfers in this mode. All other registers can be accessed word-wide, high byte only or low byte only as desired.

Table 14. BMPR10 — Transmit Start Register

BIT	SYMBOL	TYPE	DESCRIPTION
7	TX START	W 0	**TRANSMITTER START:** Writing 1 to this bit commands transmitter to start transmitting the packet(s) loaded into the transmit buffer. Before doing so, the transmitter must be idle (not busy with another buffer). The TX DONE bit is used to determine the required transmitter status, idle or busy.
6–0	TX PKT CNT	R W 0	**TRANSMIT PACKET COUNT:** A binary integer written by the system to indicate the number of packets contained in the transmit buffer for transmission. This information can be loaded at the same time the TX START bit is set high. As the transmitter finishes transmitting each packet, this counter is decremented. The value can be read by the system to see how many packets remain to be transmitted.

Transmit Start Register

The Transmit Start Register, BMPR10, contains the TX START bit and the TX PKT CNT. Writing 1 to the TX START bit immediately starts the transmitter. The Transmit Packet Count is a 7-bit binary integer written by the host to indicate the number of packets in the transmit buffer to be transmitted. See Table 14. This register should only be written when the transmitter is idle. It can be read at any time. TX START will always read as 1.

16 Collisions Control Register

This is a command register to control the actions taken when 16 consecutive attempts to transmit a packet are all met with collision. Table 15 summarizes the use of this register. It serves as both a mode setup register and an action command register. As a mode select register, two functions are selectable. Firstly, automatic continuation or halt after 16 collisions can be selected. If automatic continuation is selected, there is an option to continue attempting to transmit the same packet, or skip to the next packet. If halt is enabled, the Transmitter is restarted after a halt by writing any action code listed in Table 15 to this register.

DMA Enable Register

This is a write-only register which is used to enable and clear either Receive Read DMA or Transmit Write DMA. Refer to Table 14 for the codes.

DMA Burst Register (BMPR13)

This register is used to select the burst length for DMA operation. The burst length can be 1, 4, 8 or 12 transfers. Each transfer is one byte or one word, depending on the mode selected, System Byte or System Word (see SB/SW in DLCR6).

CODE	No. of Transfers
00H	1
01H	4
02H	8
03H	12

Table 15. 16 Collision Action Codes (written to BMPR11)

ACTION CODE	DESCRIPTION
02H or 03H	**MODE SETUP:** Halt after 16 Collisions.
02H	**COMMAND:** Resume transmitting, repeat failed packet (for use following a halt). Terminates the halt. Instructs transmitter to resume transmitting by repeating the failed packet. The collision counter is reset, allowing up to 16 additional attempts to be made. Halt after 16 collisions.
03H	**COMMAND:** Resume transmitting, skip failed packet (for use following a halt). Terminates the halt. Instructs transmitter to skip the failed packet and resume transmitting with the next packet in the buffer. The collision counter is reset, allowing up to 16 additional attempts to be made. If there is no next packet, the transmitter will deactivate, setting TX DONE as it does so. Halt after 16 collisions.
06H	**MODE SETUP:** Continue automatically after 16 collisions, repeat failed packet. Warning: If the network medium is disconnected, transmission attempts will usually result in false collision detection. Under this condition, this mode will cause the transmitter to continue re-attempting transmission of the same packet indefinitely. Interrupt or periodic polling of the status bits should be used to detect this condition.
07H	**MODE SETUP:** Continue automatically after 16 collisions, skip failed packet. Warning: This mode will result in failure to transmit some packets, since it skips a packet which has had 16 collisions. This condition is rare on a healthy network, but it does happen. To avoid this, use mode 06H.

Table 16. DMA Enable Register (BMPR12)

ACTION CODE	DESCRIPTION
01H	Enables Transmit Write DMA
02H	Enables Receive Read DMA
00H	Clear or terminate DMA activity, DMA EOP status bit and associated interrupt, if any. Normally used as response to End of Process (DMA EOP) interrupt.

Skip Packet Register (BMPR14)

Only one bit in this register is active, bit 2, the rest are 0. Writing 04H to this register commands the buffer controller to skip the balance of the current receive packet in memory. The bit can then be read to see when the skip process is complete (within 300 ns). The bit returns to 0 when the chip is ready to read the next packet, if there is another packet, or stop reading if there is not. Limitation of use: Do not use this feature before reading at least four (4) times from the beginning of the packet, nor if there are only eight (8) or fewer bytes left of the packet in the buffer. Doing so may corrupt the receive buffer pointers.

BMPR15 is unused and reserved for possible future use. Write only 0's to this register.

TRANSMITTER CIRCUITS

Circuits within the transmitter include a transmitter state machine, a small FIFO for pipe-lining the packet data, preamble generator, CRC generator, parallel to serial converter, backoff generator, inter-packet gap timer and a time domain reflectometer (TDR) counter.

The transmitter state machine provides sequencing of events for the transmitter, including idle, preamble, data, CRC, inter-packet gap, jam and backoff. It detects various transmit error conditions and sets appropriate bits within the DLCR registers.

The pipeline FIFO provides elastic buffering that the buffer controller can load with data to be transmitted. NICE's CRC generator calculates the Ethernet 32-bit CRC on the destination and source address, the length field and the data field as specified by the ISO/ANSI/ IEEE 8802-3 specification for Ethernet. This value is appended to the end of the packet.

Transmit Error Processing

NICE has four transmit error status bits in its Transmit Status Register (DLCR0) for reporting the three possible transmit errors. The errors are: 1) loss of carrier during transmission, which usually indicates a medium fault or a collision, 2) collision, and 3) 16 consecutive collisions. The latter two can be enabled separately to generate interrupts.

If NICE detects a collision during transmission, it will automatically try to retransmit the packet until sixteen attempts have been made. Collision counter DLCR4<7:4>, automatically increments after each collision up to the sixteenth collision, at which time it rolls over to zero. (Bit 7 is the most-significant of the four bits.) Appropriate status bits in the Transmit Status Register and Transmit Mode Register are set in case of a collision-terminating transmission. Another status bit (16 COL) indicates that sixteen consecutive attempts to transmit a packet have been made and all have been terminated by collision. This case may indicate a network problem. For example, a disconnected cable or terminator will produce false collisions. But 16 collisions can occur normally, although rarely. A pseudo-random number generator provides the collision backoff function. This is clocked at the bit rate, 10 MHz, so that distances between stations become part of the randomizing function. It is sampled at the time of collision, masking all but the appropriate number of bits specified by the 8802–3 backoff algorithm. This value is then counted down at the slot-time rate (512 bits) to generate the backoff interval. For a first collision, only one bit is used, giving a backoff of either 51.2 microseconds or 0. For a second consecutive collision, two bits are used, and so forth, up to ten bits. From the tenth to the 16th collision, 10 bits are used. This gives a pseudo-random backoff interval of from 0 to 52.38 ms, the so-called 'binary exponential backoff' for collisions per 8802-3.

Time Domain Reflectometry

The TDR function provided counts the actual number of bits transmitted for each packet before an indication of either collision or carrier loss occurs, or the transmission completes. If a transmission completes without error indications, the TDR counter is cleared. See also the register description for DLCR14 and DLCR15.

Media Access Control

NICE's transmitter state machine implements the media access protocol for 8802-3 networks called CSMA/CD, Carrier Sense, Multiple Access with Collision Detection. The 'carrier sense' part means that the controller monitors the network for carrier from other nodes, and defers transmission while other nodes are transmitting (collision avoidance). But collisions can still occur when two nodes, perhaps separated on the network by several microseconds, start to transmit at nearly the same time. This is handled by the 'collision detection' part. All nodes are required to monitor the network for collisions and, when involved in one, transmit a 32-bit 'Jam' to reinforce the collision, then terminate transmission. Later, after waiting a pseudo-random backoff interval, the node automatically re-attempts to transmit the packet.

Between packets, there must be a gap of at least 9.6 microseconds during which time the trunk cabling is idle. NICE's transmitter state machine measures this interval starting from the end of a packet on the network. It will not transmit until this interval has expired. During the

first 2/3 of the interval, if for some reason carrier reappears on the network, NICE resets its interval timer to re-time the interval from the end of the new transmission. Such an event can occur during a collision, since data and carrier indications can be corrupted by the superimposition of the two packets. During the last 1/3 of the interval, NICE will ignore a carrier indication if it occurs. This is in accordance with 8802–3, intended to assure fairness and equality in access to the network. If one station starts to transmit slightly ahead of another, no advantage will be gained by the slightly earlier start. Both nodes will transmit, a collision will occur, and the media-access contention will be resolved by backoff interval differentials.

Data Encoder

NICE serializes the data for transmission, and converts each bit to Manchester Code, the format used on the network media. Manchester Code for '1' is a 100 ns interval starting with a low, ending with a high, with a low-to-high transition at the 50 ns point. Manchester Code for '0' is the inverse of this. See Figure 14 for a block diagram of the encoder/decoder section.

Operating with an External Encoder/Decoder

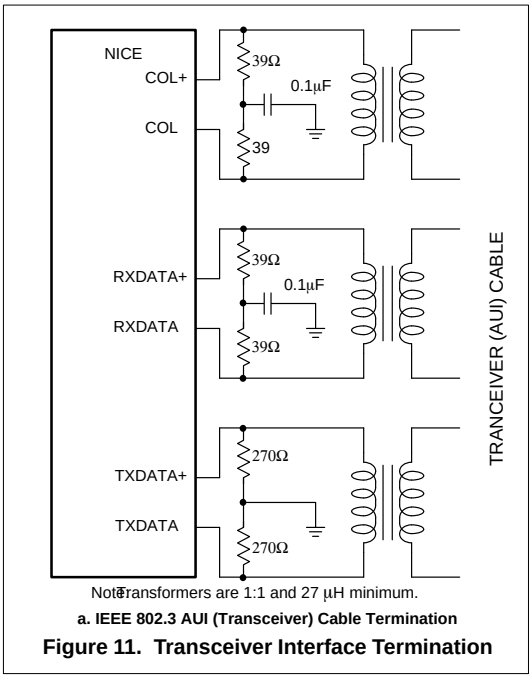
An option is provided on NICE to disable the on-chip encoder/decoder circuits and use an external encoder/decoder. Specific details are given in the register description section in Table 13, Configuration Register 1 (see DLCR7<7:6>).

Transmit Packet Processing

To transmit one or more packets, the host system first loads the packet(s), preceded by a 2-byte header giving their lengths, into a transmit buffer by writing the data to the Buffer Memory Port Register, BMPR<8:9>. Only the destination address, source address, length field and data field of the packets are loaded by the system, NICE generates the rest. When the packets are loaded, the system turns the transmitter on to initiate transmission. This enables NICE to transmit. Observing the media access protocol, NICE defers transmitting to carrier from other nodes, minimum inter-packet gap intervals and backoff intervals if any, then begins to transmit. NICE serializes the data and encodes it in Manchester Code. It generates the preamble field at the beginning, and calculates and appends the CRC field at the end, followed by the End-of-Packet Delimiter, which is a non-Manchester code. The Manchester-encoded signals are output through a differential driver to the TX DATA± pins to the external transceiver.

The driver is capable of driving a 50-meter segment of 78-Ohm transceiver cable, as specified in the 8802–3 standard. 270-Ohm resistors to GND are required externally to pull down TXDATA+ and TXDATA–. See Figures 11 and 12 for suggested cable interfacing.

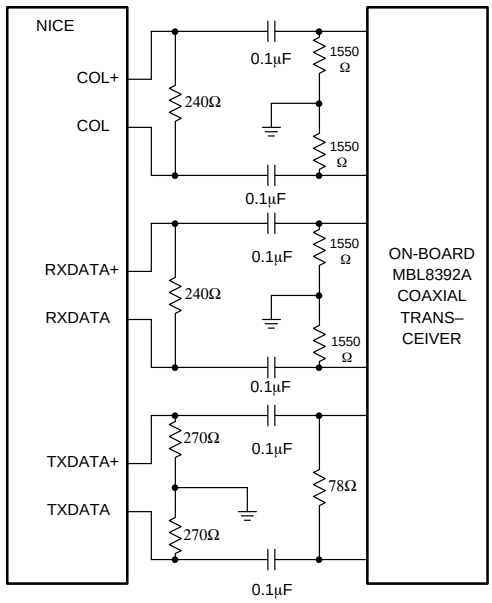
The host system activates the transmitter by writing 1 to the TX START bit and the packet count to TX PKT CNT. After this is done, the transmitter will transmit each packet in the buffer in the order that they were loaded. If a collision occurs, the transmitter automatically retransmits the packet until successful, or until 16 consecutive attempts have ended in collision. In the latter case, depending on the mode selection made at initialization time, the Transmitter will either 1) continue to try to transmit the same packet, starting again with a collision count of zero, 2) skip the current packet and try to transmit the next packet, starting with a collision count of zero, or 3) halt and wait for instruction from the host. In the third case the host can select to either 1) terminate by setting DLC EN high, 2) continue to attempt to transmit the same packet (collision counter gets reset) or 3) skip the current packet and try to transmit the next packet (collision count = 0).



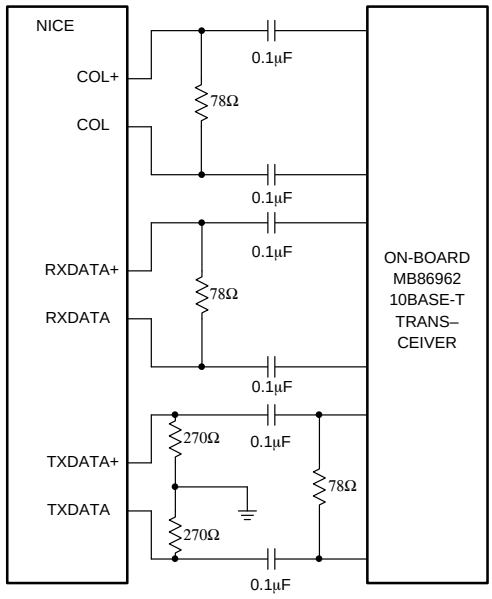
Note: Transformers are 1:1 and 27 µH minimum.

a. IEEE 802.3 AUI (Transceiver) Cable Termination

Figure 11. Transceiver Interface Termination



b. Simplified Termination for Coaxial Transceiver



c. Simplified Termination for on-board 10BASE-T Transceiver

Figure 12. Transceiver Interface Termination

Collision Signal Processing

As collisions are detected on the network media, the external transceiver generates a 10 MHz signal on the COL± differential inputs to the NICE's encoder/decoder section. When this signal is detected by NICE, it asserts the internal collision line, and in some modes the COL pin, pin 92, is also asserted.

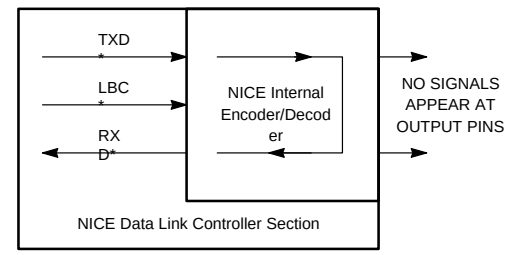
The COL± differential driver inputs require termination like that for the RXDATA± inputs. See Figures 11 and 12 for suggested cable termination.

Loopback

Loopback capability is provided to allow testing of NICE without sending signals onto the LAN media. The loopback function is invoked by clearing the LBC bit, DLCR4<1>. Loopback operation is illustrated in Figure 13.

Data is routed from the transmit buffer through a FIFO to the transmit section of the data link controller, through the internal Manchester encoder, back through the Manchester decoder, through the receiver section of the data link controller, and is then stored in a receive buffer. The test software can then read and check the received packet which has traveled through nearly all transmit and receive sections of the chip.

When an external encoder/decoder is used, the data is output on TXD and received at RXD. The external Manchester encoder/decoder should respond to assertion of its LBC input by looping its transmitter output to its receiver input internally, and should block the transmit data from appearing at its network output pin. Fujitsu's MB86951 and MB502A Encoder/Decoders, and MB86961 Encoder/Decoder with 10BASE-T Transceiver, for example, all respond in this way.



* In "NICE" mode, these are internal signals and no signals appear on the corresponding pins.

Figure 13. NICE in Loopback Mode

RECEIVER CIRCUITS

The receiver includes a receive state machine, serial to parallel conversion, pipe-line FIFO, preamble recognition, bit and byte framing, address filtering, CRC and other error checking and 'end-of-packet' symbol recognition.

The receiver state machine provides sequencing of events for the receiver, including idle, busy, address filtering, data storage, etc. It detects various receive error conditions and sets appropriate bits within the DLC Registers.

A small data FIFO provides elastic buffering for synchronization with the buffer controller timing, and buffering data while the buffer controller is servicing other buffer memory access requests.

All received bytes are delayed by four bytes before storing in the receive buffer so that the last four bytes of the packet can be stripped and checked for correct CRC. (The CRC bytes are not transferred to the receive buffer.)

During reception, packets are automatically rejected if space in the receive buffer is insufficient to hold the entire received packet. Status bits in the receive status register are set to indicate this and other errors. The receive errors are: 1) bus read error, which occurs if the host system attempts to read from an empty receive buffer (this need never occur if the RX BUF EMPTY bit is checked), 2) short packet error, 3) alignment error (incomplete byte fragment at end of packet), 4) CRC error and 5) buffer overflow. There is one additional possible receive error which the chip leaves to the software to check -- length error. When the length of the packet does not match the value in the Length Field of an 8802–3 packet, this is a length error. Some protocols use the length field for other purposes, for example, the DIX protocol uses it for a packet type code. In this case, allowed type codes do not overlap allowed packet length values, providing a means to distinguish which protocol is being used (if length value >1500, it's DIX type code). Length check can be made conditional on protocol type if necessary to support other protocols like DIX.

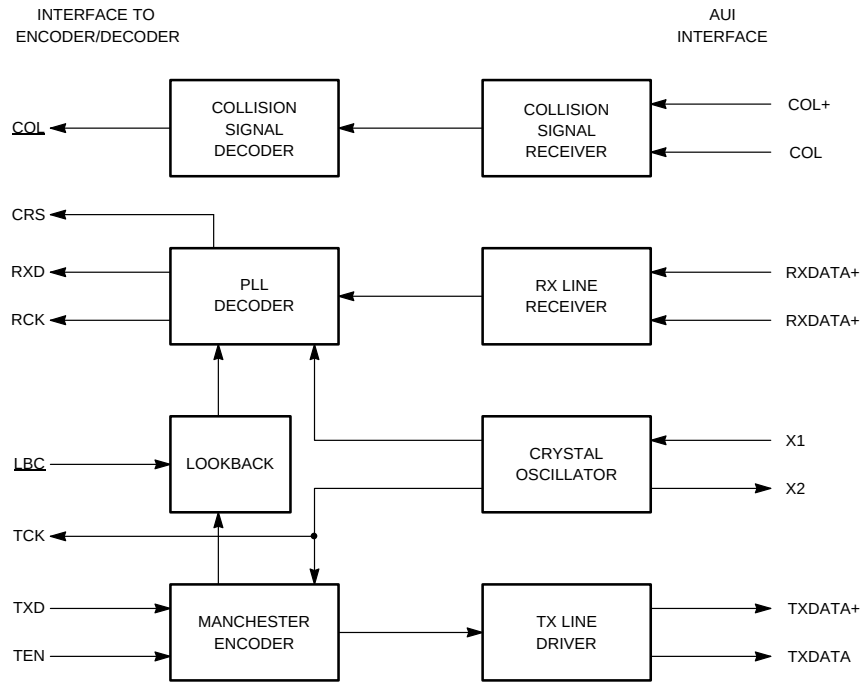


Figure 14. Encoder/Decoder Block Diagram

Decoder Functions

The data decoder section performs three functions on the data received at the differential receive inputs (RXDATA±) from the transceiver: clock recovery, carrier detection and Manchester data decoding.

Clock recovery and data separation are accomplished by the use of a phase locked loop. Use of proprietary techniques in the PLL allows lock-on to be accomplished within 6–7 bit times of the beginning of the preamble, and permits stable operation with input signal jitter of up to ±18 ns. Carrier detection is indicated to the controller by assertion of the CRS signal, which occurs shortly after a signal appears at RXDATA±.

The recovered clock is supplied to the controller on RCK, and is also used to convert the Manchester encoded data to NRZ format. NRZ data is output on RXD. Transitions in the state of RXD are synchronous with the falling edge of RCK. During idle periods, RCK is a free-running 10 MHz clock.

The RXDATA± differential inputs are usually terminated with two 39-Ohm resistors in series and an 0.1μF bypass capacitor to ground at their junction, as shown in Figure 11.

Monitoring the Network

Whenever the data link section is enabled (DLC EN = 0), the receiver is constantly monitoring the network for carrier. Signals which exceed the AC and DC squelch thresholds of the RXDATA± input section cause the internal carrier sense (CRS) line to assert, which in turn causes the receiver to attempt to receive a packet. Refer to Figure 2 for a block diagram of the encoder/decoder section. (The carrier sense function is also used by the transmitter to defer to transmissions from other nodes, except when DLCR4<0> is high.)

After the PLL decoder acquires bit-synchronization with the incoming signal, the receiver monitors the data stream for the end-of-preamble bit pattern, two consecutive 1's ending the preamble's pattern of alternating 1's and 0's. This pattern gives the receiver byte and field synchronization, because the bit immediately following the two 1's is the first bit of the first byte of the packet's destination address field.

When packet transmission is unflawed, CRS will remain asserted for the duration of the packet, negating just after the last bit has been received. As a packet is coming in, the decoder's carrier sense function monitors the data stream for the end-of-packet symbol, a special non-Manchester code element at the end of the packet. Upon detecting this symbol, the carrier sense line will be negated. Loss of carrier will also result in negation of the carrier sense line, for example, when a collision occurs.

Receive Packet Processing

As a receive packet comes in from the network, its destination address field is tested for the various address filter criteria selected by the Address Filter Mode bits (AF1, AF0) and the Hash Table. See Figure 15 for 8802–3 packet format. Only if the address meets the filter criteria selected will the packet be accepted for storage in the receive buffer. In addition, the packet must be error-free, unless the chip has been enabled to receive flawed packets for diagnostic purposes. If these conditions are met, the packet reception results in the packet being stored in the buffer, its 4-byte header being updated at the end of reception, the RX BUF EMPTY bit being cleared, the RX PKT bit being set high and an interrupt being generated (if enabled). Otherwise the packet will be discarded and pointers will be reset to reuse the same portion of memory for the next packet to arrive. If a flawed packet is accepted for storage for diagnostic purposes, its error(s) will be reported in the PKT STATUS byte of its header (refer to Table 3 for byte and bit positions).

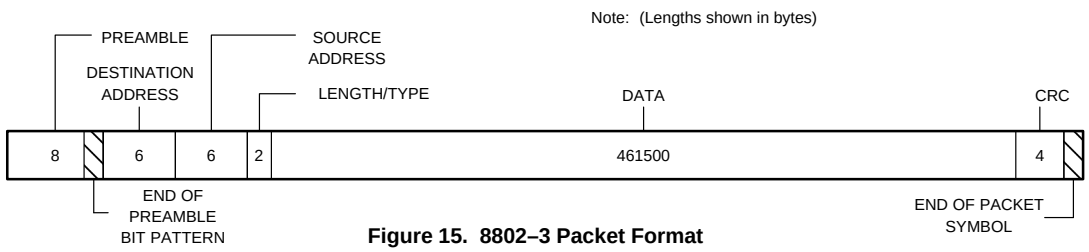


Figure 15. 8802–3 Packet Format

Network Management

Error, traffic and performance statistics can be collected continuously or on a sampled basis. The Receive Status Register and Transmit Status Register indicate any errors detected. Such data can be collected in two ways. Either interrupts can be used after each packet, and the status read from the status register by the interrupt service routine, or, for the receive case, the packets can be accepted for storage in the receive buffer, allowing their contents and error statuses (stored in the header) to be read later in batch mode. To get maximum statistics for the network, the "Accept all packets" mode can be used. In this mode, all packets can be counted, including their lengths. But of course, this use will maximize host overhead, so it should be used sparingly in user terminal equipment.

Carrier detection can be sampled as a means of estimating network bandwidth utilization. This bit is available in the Transmit Status Register (NET BSY).

An estimate of the average media-access waiting time can be calculated from the elapsed time between starting the Transmitter and TX DONE going high.

The transmit collision count (DLCR4<7:4>) can be used to determine the number of collisions encountered by the last outgoing packet. (The counter is reset at the start of transmitting each new packet.)

ELECTRICAL CHARACTERISTICS

Table 17. ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter Description	Min.	Max.	Units
V_{CC}	Supply voltage	–0.5	6.0	V
V_{IN}	Input voltage	–0.5	V_{CC} + 0.5	V
V_{OUT}	Output voltage	–0.5	V_{CC} + 0.5	V
I_{ODF}	Differential output current on TXDATA± pins		–40	mA
V_{IDC}	Input DC voltage on RXDATA± and COL±	–0.5	16	V
V_{ODC1}	Output DC voltage on TXDATA± w/o transformer	–0.5	14	V
V_{ODC2}	Output DC voltage on TXDATA± with transformer	–0.5	16	V
T_{BIAS}	Temperature under bias	–25	+85	°C
T_{STG}	Storage temperature	–40	+125	°C
PWR	Power dissipation		425	mW

1. Permanent device damage may occur if absolute maximum ratings are exceeded. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.
2. Not more than one output may be shorted to ground or V_{CC} at a time for a maximum duration of one second.

Table 18. RECOMMENDED OPERATING CONDITIONS

Symbol	Parameter Description	Min.	Typ.	Max.	Units
V_{DD}	Supply voltage	4.75		5.25	V
V_{IH}	Logic input high voltage	2.2			V
V_{IL}	Logic input low voltage			0.8	V
R_L	Driver load resistors on TXDATA± pins to ground	250	270	290	Ohm
R_T	Termination resistors (2 in series across RXDATA± and COL±)	38.6	39	39.4	Ohm
C_T	Termination bypass capacitors (between junction of termination resistors and ground)	0.1			µF
C_{OSC}	Oscillator load capacitance, including stray capacitance (see Figure 3)	12	20	38	pF
f_{XTAL}	Crystal oscillator frequency	19.999	20	20.001	MHz
T_A	Operating temperature	0		70	°C

Table 19. DC SPECIFICATIONS (At recommended operating conditions unless otherwise noted.)

Symbol	Parameter Description	Conditions	Min.	Typ.	Max.	Units
V_{IL}	Low level input voltage		0.0		0.8	V
V_{IH}	High level input voltage		2.2		V_{CC}	V
V_{OL1}	Low level output voltage, all outputs except DREQ	I_{OL} = –3.2 mA	0.0		0.4	V
V_{OL2}	Low level output voltage, DREQ only	I_{OL} = –12 mA	0.0		0.4	V
V_{OH1}	High level output voltage, all outputs except DREQ	I_{OH} = +2 mA	4.2		V_{CC}	V
V_{OH2}	High level output voltage, DREQ only	I_{OH} = +4 mA	4.2		V_{CC}	V
V_{OP}	TXDATA± peak output	R_L = 270 Ohms	±0.5		±1.3	V
V_{ACCM}	Output AC common mode on TXDATA±	R_L = 270 Ohms, R_T = 78 Ohms			±40	mV
V_{DCCM}	Output DC common mode on TXDATA±	R_L = 270 Ohms	2.4	3.4	4.4	V
V_{SQ}	Squelch threshold	AC/DC = 1 AC/DC = 0	–300 –80	–220 0	–140 +80	mV mV
I_L	Input leakage current	V_I = 0 – V_{CC}	–10		10	µA
I_{PWRDN}	Power down V_{CC} current	No output loads		6	10	mA
I_{CC}	Operating V_{CC} current	No output loads		40	85	mA

Table 20. GENERAL CAPACITANCE (T_A = 25°°C, V_{DD} = V_I = 0 V, f = 1 MHz)

Symbol	Parameter Description	Min.	Max.	Units
C_{IN}	Input pin capacitance		16	pF
C_{OUT}	Output pin capacitance		16	pF
$C_{I/O}$	I/O pin capacitance		16	pF

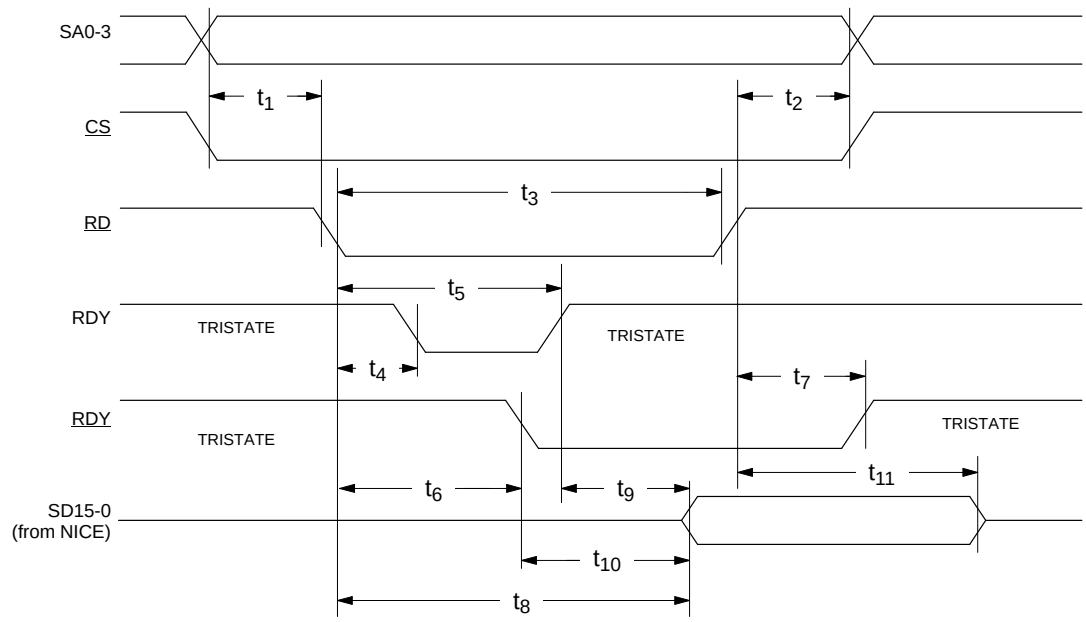


Figure 16. Read Cycle

Table 21. Read Cycle

Symbol	Parameter Description	Min.	Max.	Units
t_1	CS low to RD low; SA3–0 valid to RD low	3		ns
t_2	RD high to CS high; RD high to SA3–0 invalid	3		ns
t_3	RD low pulse width	30		ns
t_4	RD low to RDY low	0	26	ns
t_5	RD low to RDY TRISTATE [1]		175	ns
t_6	RD low to RDY low [2]	0	175	ns
t_7	RD high to RDY TRISTATE		28	ns
t_8	RD low to SD15–0 valid (except Buffer Memory Port)		44	ns
t_9	RDY TRISTATE to SD15–0 valid (buffer port)		8	ns
t_{10}	RDY low to SD15–0 valid		10	ns
t_{11}	RD high to SD15–0 invalid (data hold)	15		ns

1. 0 ns maximum for registers, and for Buffer Memory Port when port is ready before the read cycle begins. For port access only, 175 ns maximum may occur if system makes contiguous system read cycles at less than 100 ns intervals, and both the transmitter and receiver are active in “loopback” reception. 2.15 µs max for bus read error.
2. 28 ns maximum for all registers. For port access only, 175 ns maximum may occur if system makes contiguous system read cycles at less than 100 ns intervals, and both the transmitter and receiver are active in “loopback” reception. 2.15 µs max for bus read error.

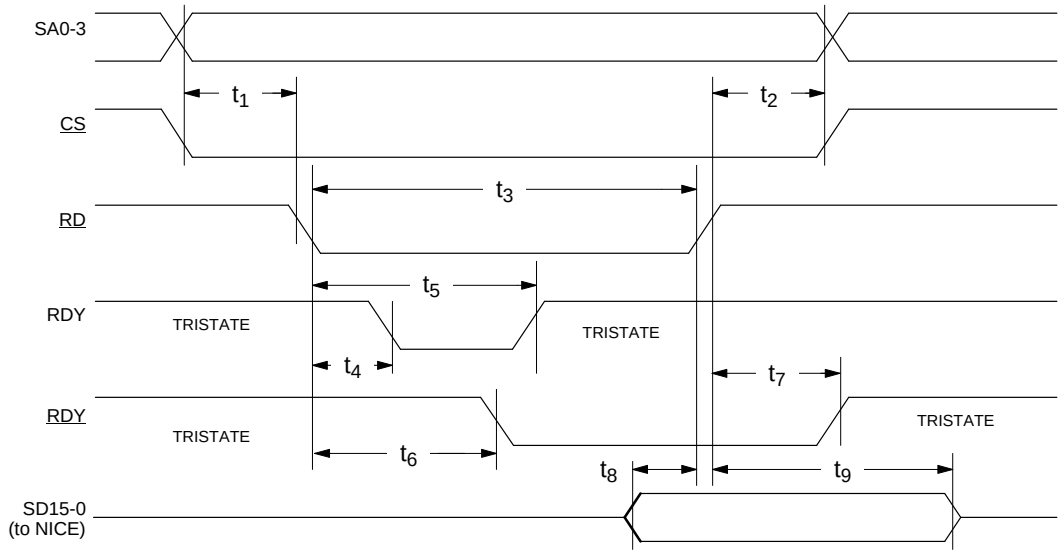


Figure 17. Write Cycle

Table 22. Write Cycle

Symbol	Parameter Description	Min.	Max.	Units
t_1	CS low to WE low; SA3–0 valid to WE low	3		ns
t_2	WE high to CS high; WE high to SA3–0 invalid	3		ns
t_3	WE low pulse width	36		ns
t_4	WE low to RDY low	0	26	ns
t_5	WE low to RDY TRISTATE [1]		175	ns
t_6	WE low to RDY low [2]	0	175	ns
t_7	WE high to RDY TRISTATE		28	ns
t_8	SD15–0 valid to WE high (data setup)	5		ns
t_9	WE high to SD15–0 invalid (data hold)	6		ns

1. 0 ns maximum for registers, and for Buffer Memory Port when port is ready before the write cycle begins. For port access only, 175 ns maximum may occur if system makes contiguous system write cycles at less than 100 ns intervals, and both the transmitter and receiver are active in “loopback” reception.
2. 28 ns maximum for all registers. For port access only, 175 ns maximum may occur if system makes contiguous system write cycles at less than 100 ns intervals, and both the transmitter and receiver are active in “loopback” reception.

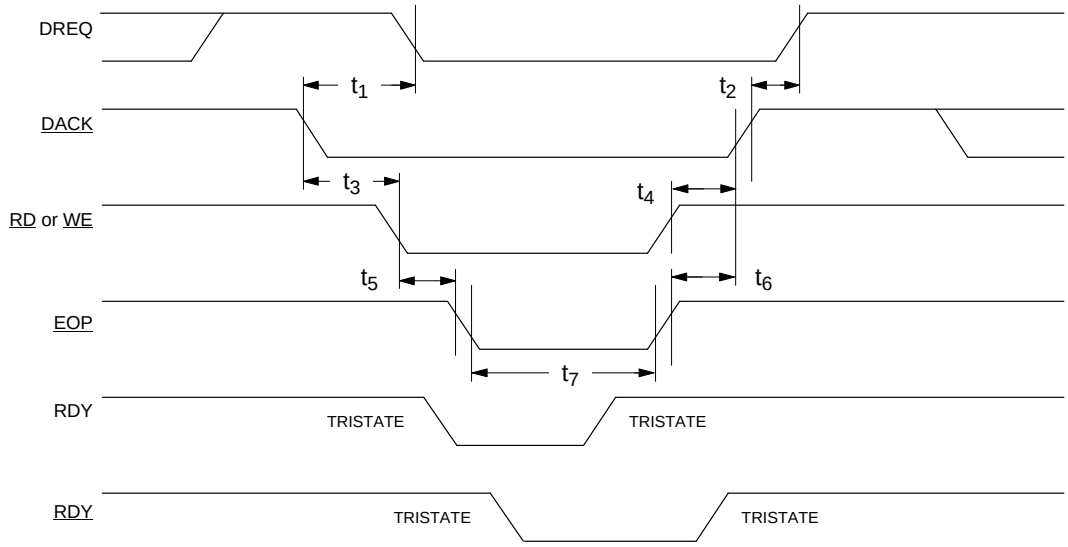


Figure 18. Single-Cycle DMA Timing

Table 23. Single-Cycle DMA Timing

Symbol	Parameter Description	Min.	Max.	Units
t_1	DACK low to DREQ low	0	21	ns
t_2	DACK high to DREQ high	0	19	ns
t_3	DACK low to RD or WE low	0		ns
t_4	RD or WE high to DACK high	3		ns
t_5	RD or WE low to EOP low	0		ns
t_6	EOP high to DACK high	3		ns
t_7	EOP low pulse width	10		ns

1. An asserted EOP terminates any further DREQ after DACK returns high.
2. The DMA cycle uses DACK as the chip select. DACK overrides CS and SA3-0 if they are both asserted at the same time, forcing selection of the Buffer Memory Port as in a DMA cycle.
3. For RDY(RDY) timing and SD15-0 timing, see Figure 16, t_4-t_{11}, and Figure 17, t_4-t_9.

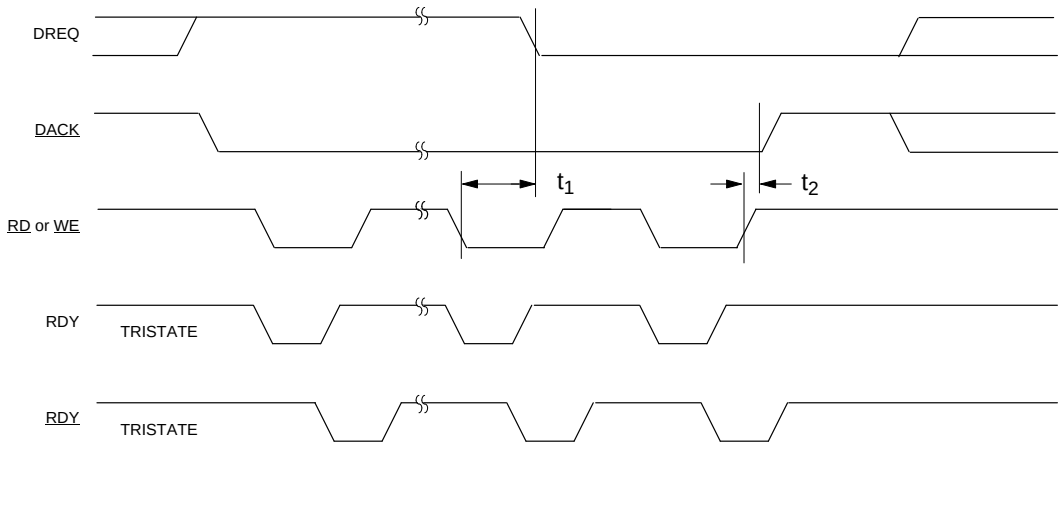


Figure 19. Burst DMA Timing

Table 24. Burst DMA Timing

Symbol	Parameter Description	Min.	Max.	Units
t_1	RD or WE low to DREQ low		32	ns
t_2	RD or WE high to DACK high	3		ns

1. DREQ goes low during the next-to-last transfer of the burst. DACK should not go high until after the RD or WE pulse of the last transfer cycle goes high.
2. The DMA cycle uses DACK as the chip select. DACK overrides CS and SA3-0 if they are both asserted at the same time, forcing selection of the Buffer Memory Port as in a DMA cycle.
3. For RDY(RDY) timing and SD15-0 timing, see Figure 16, t_4-t_{11}, and Figure 17, t_4-t_9.

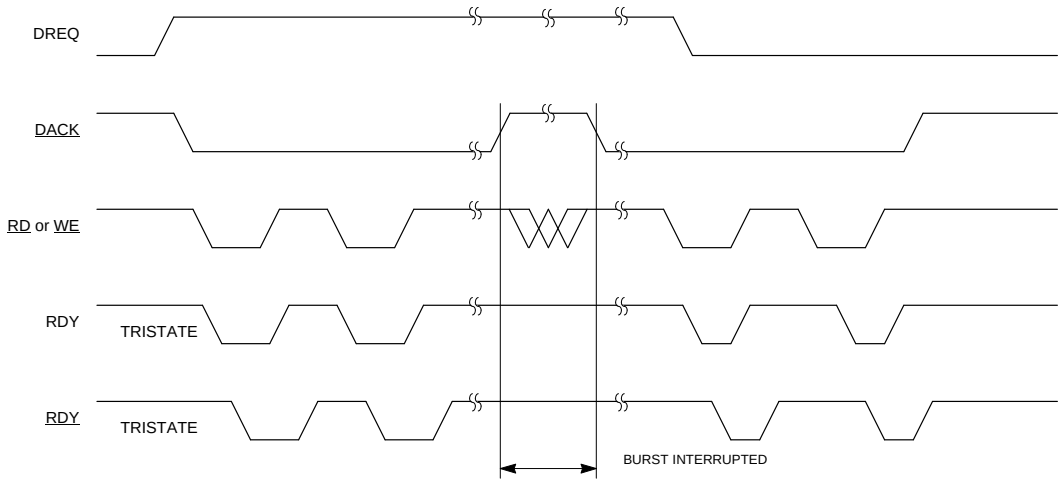


Figure 20. Burst DMA Interrupted by DACK

Notes: Burst can be interrupted by DACK high-going pulse during the burst. Burst will resume when DACK returns low.

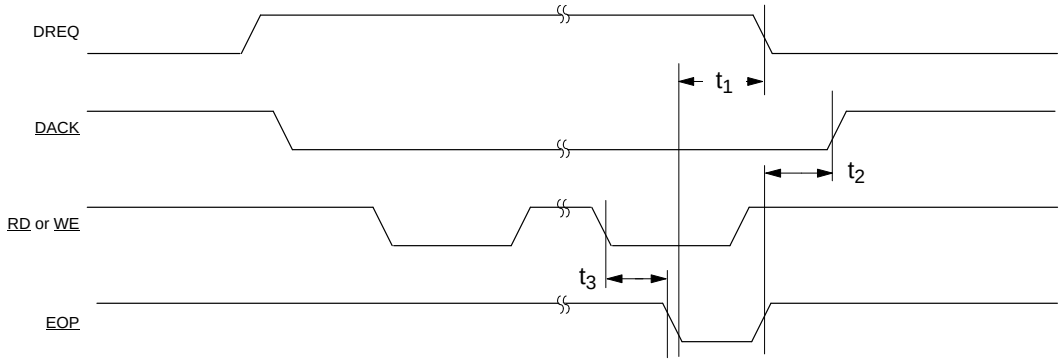


Figure 21. Burst DMA Terminated by EOP

Table 25. Burst DMA Terminated by EOP

Symbol	Parameter Description	Min.	Max.	Units
t_1	EOP low to DREQ low	4	28	ns
t_2	EOP high to DACK high	3		ns
t_3	RD or WE low to EOP low	0		ns

Note: EOP can be asserted during any transfer of the burst to terminate the process following that transfer.

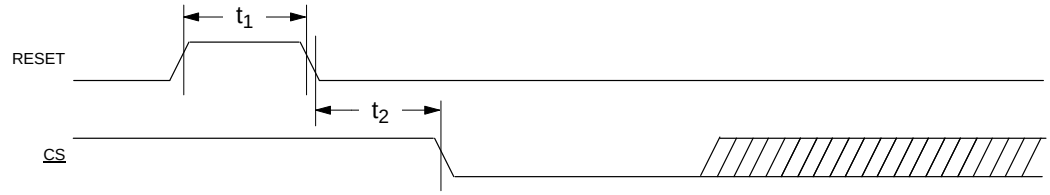


Figure 22. RESET Timing

Table 26. RESET Timing

Symbol	Parameter Description	Min.	Max.	Units
t_1	RESET pulse width	200		ns
t_2	RESET low to first CS low	300		ns

Note: Before enabling transmit and receive functions (DLC EN), wait 200 µs after reset pulse for stabilization of receiver PLL.

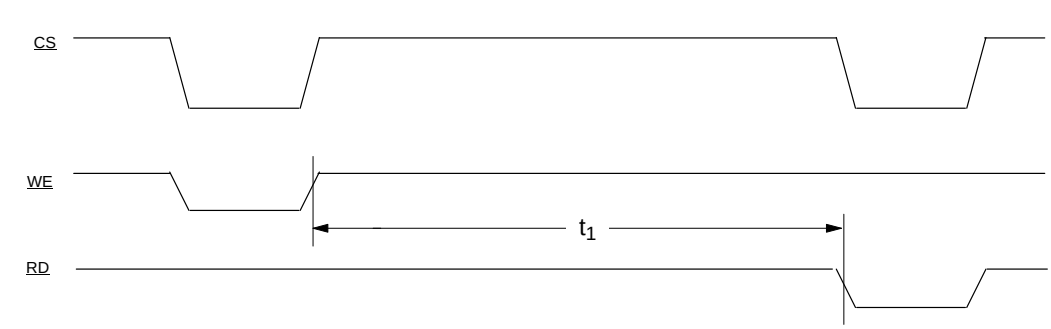


Figure 23. Skip Packet Timing

Table 27. Skip Packet Timing

Symbol	Parameter Description	Min.	Max.	Units
t_1	Writing Skip Packet high to next Buffer Memory Port read	300		ns

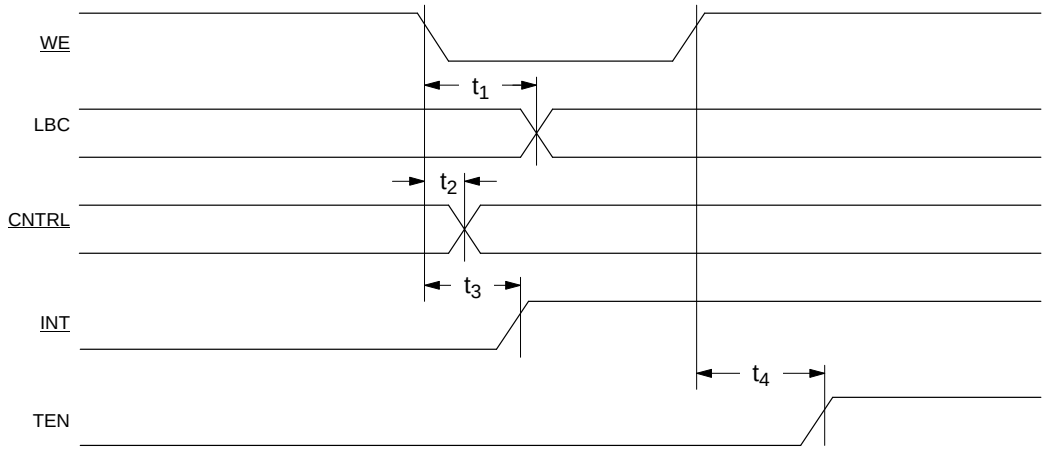


Figure 24. LBC, CNTRL, AND INT Timing

Table 28. LBC CNTRL and INT Timing

Symbol	Parameter Description	Min.	Typ.	Max.	Units
t_1	Loopback Control (LBC) delay	20		60	ns
t_2	CNTRL delay	20		60	ns
t_3	INT signal clearing delay	20		60	ns
t_4	Transmit enable delay after setting TX START high (if network free)		2.3		µs

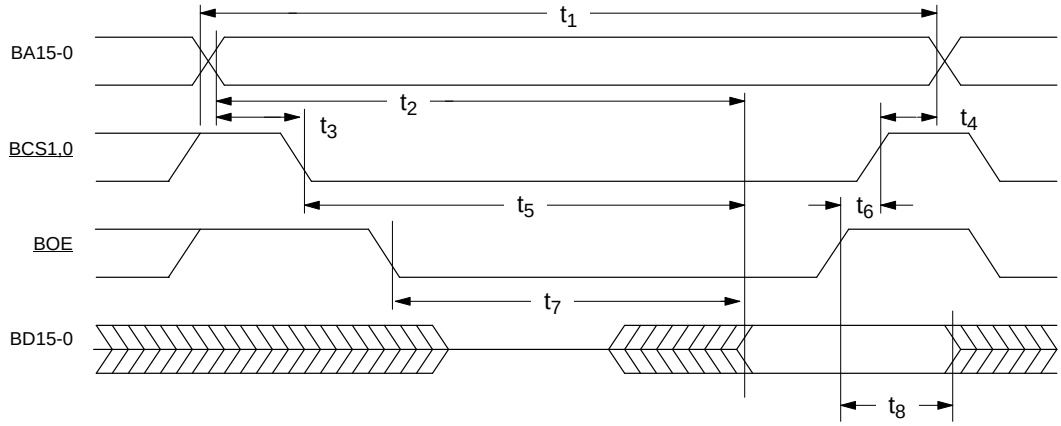


Figure 25. SRAM Read Timing

Table 29. SRAM Read Timing

Symbol	Parameter Description	Min.	Max.	Units
t_1	Read cycle	95		ns
t_2	Address access time		81	ns
t_3	Address valid to BCS1,0 low	0	8	ns
t_4	BCS1,0 high to address invalid	0		ns
t_5	Chip select access time		81	ns
t_6	BOE high to BCS1,0 high	0	2	ns
t_7	Output enable access time		49	ns
t_8	Data hold time	0		ns

Note: Use SRAM with address access time of 80 ns or less.

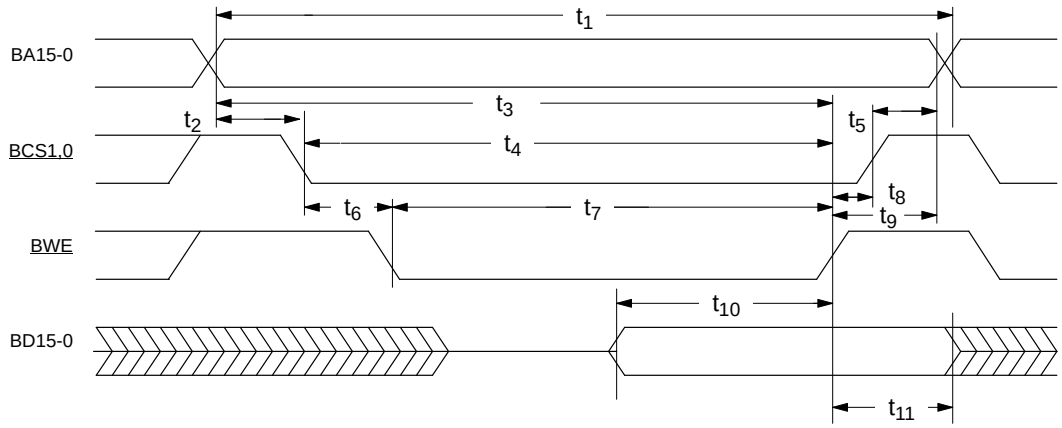


Figure 26. SRAM Write Timing

Table 30. SRAM Write Timing

Symbol	Parameter Description	Min.	Max.	Units
t_1	Write Cycle	95		ns
t_2	Address Valid to BCS1,0 low	2	8	ns
t_3	Address Valid to BWE high	71		ns
t_4	BCS1,0 low to BWE high	62		ns
t_5	BCS1,0 high to Address Invalid	0		ns
t_6	BCS1,0 low to BWE low	0		ns
t_7	BWE Pulse Width	60		ns
t_8	BWE high to BCS1,0 high	0		ns
t_9	BWE high to Address Invalid	12		ns
t_{10}	Data Setup Time	41		ns
t_{11}	Data Hold Time	14		ns

Note: Use SRAM with address access time of 80 ns or less.

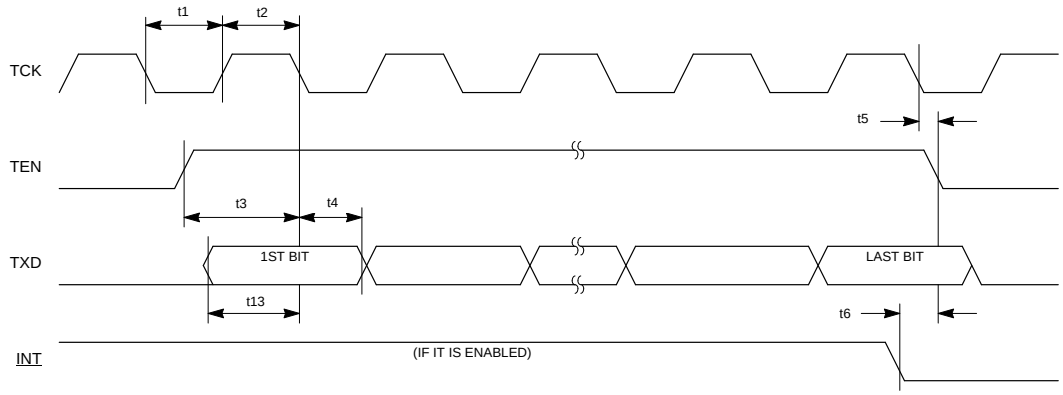


Figure 27. Transmit Timing

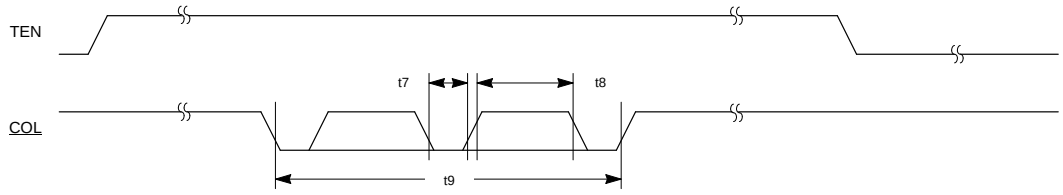


Figure 28. Transmit Timing

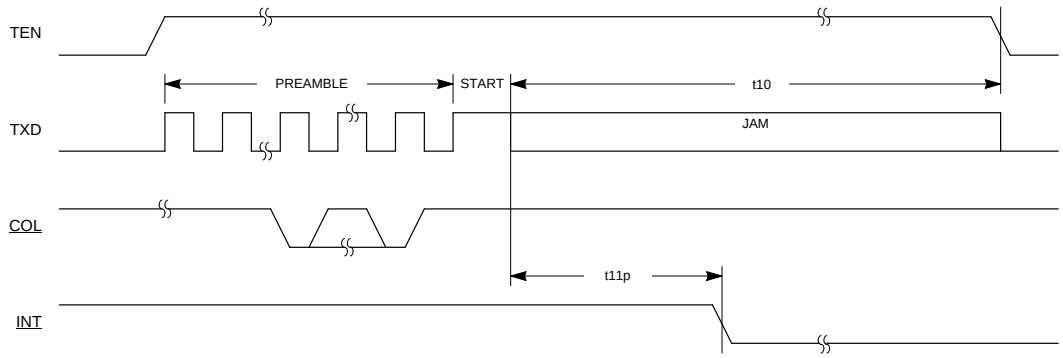


Figure 29. Transmit Timing

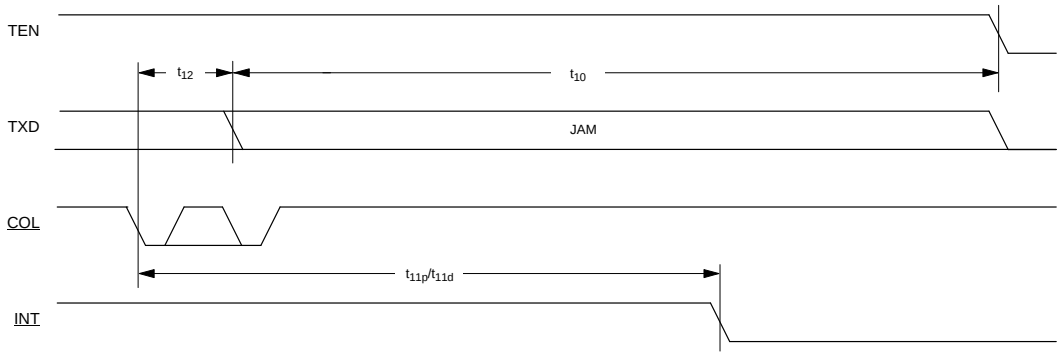


Figure 30. Transmit Timing

Table 31. Transmit Timing: Figures 27–30 (for Encoder/Decoder Bypass mode)

Symbol	Description	Min.	Typ.	Max.	Units
t_1	Transmit clock low width	40	50	60	ns
t_2	Transmit clock high width	40	50	60	ns
t_3	TEN high to TCK low	48	—	—	ns
t_4	Transmit data hold	12	—	—	ns
t_5	TCK low to TEN low	13	—	—	ns
t_6	Transmit interrupt low to transmit enable low	—	1	—	TCK cycles
t_7	Collision low pulse width	20	—	—	ns
t_8	Collision high pulse width	—	—	200	ns
t_9	Minimum collision length	520	—	—	ns
t_{10}	Jam period [1]	—	32	—	TCK cycles
t_{11p}	Transmit interrupt when collision at preamble	—	5	—	TCK cycles
t_{11d}	Transmit interrupt when collision at data field	—	16	—	TCK cycles
t_{12}	Collision to first jam bit	4	—	12	TCK cycles
t_{13}	Transmit data setup	40	—	—	ns

1. The 32 jam bits include eight data bits and 24 ‘0’ bits.

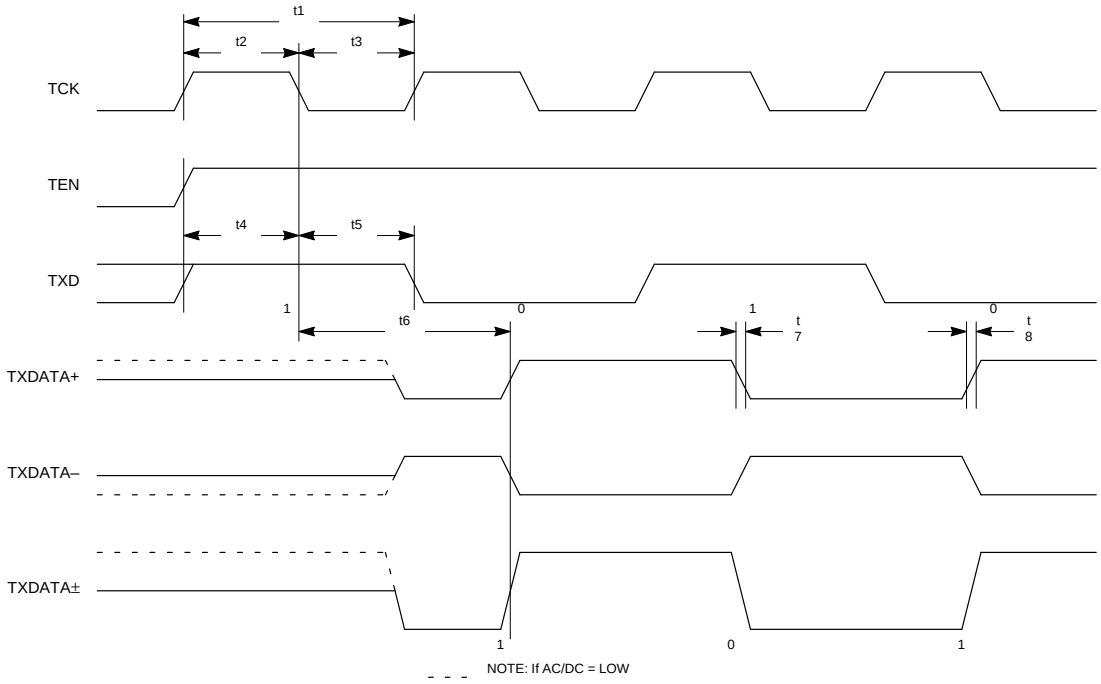


Figure 31. Transmit Start Timing

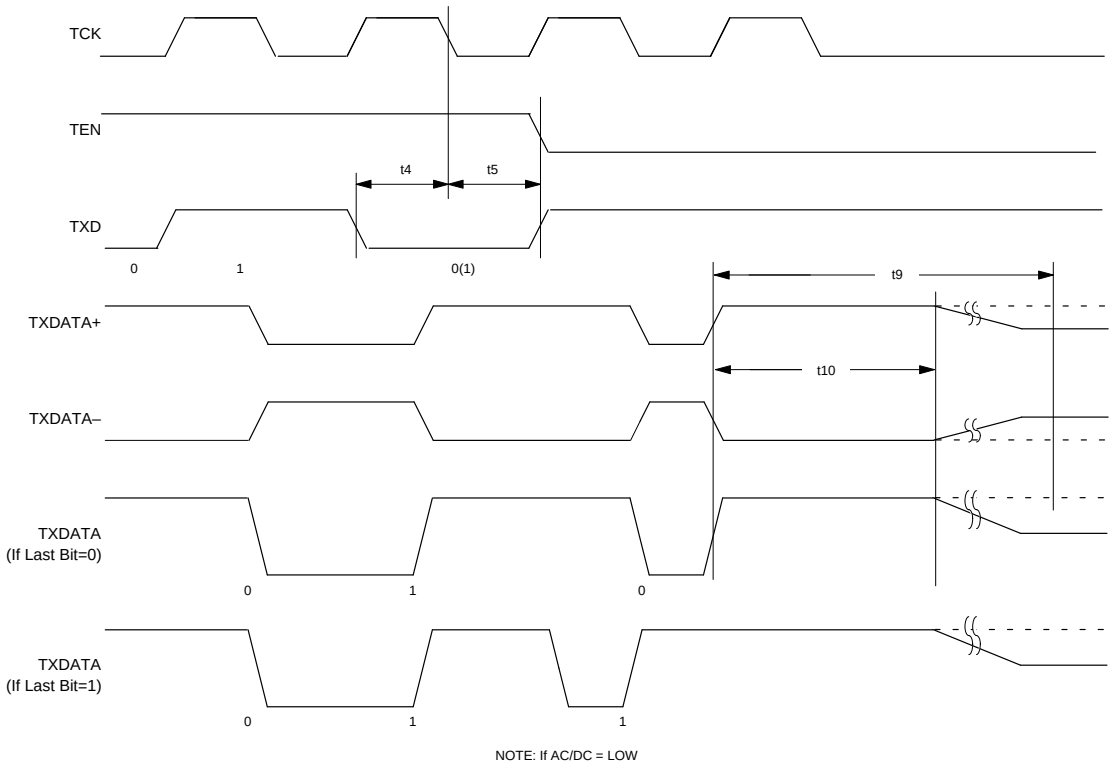


Figure 32. Transmit End Timing

Table 32. Transmit Start and End Timing: Figures 31–32 (for Encoder/Decoder Test mode)

Symbol	Figure	Parameter Description	Conditions	Min.	Typ.	Max.	Units
t_1	31	TCK cycle time		99.99	100	100.01	ns
t_2	31	TCK high width		35	50	65	ns
t_3	31	TCK low width		35	50	65	ns
t_4	31, 32	TXD, TEN setup time to TCK		20	—	—	ns
t_5	31, 32	TXD, TEN hold time from TCK		10	—	—	ns
t_6	31	TXDATA± encode time		—	95	—	ns
t_7	31	TXDATA± fall time	20% to 80%	—	2	—	ns
t_8	31	TXDATA± rise time	20% to 80%	—	2	—	ns
t_9	32	TXDATA± line voltage transition	AC/DC = High	—	—	8000	ns
t_{10}	32	TXDATA± end-of-packet delimiter	AC/DC = High	200	—	—	ns

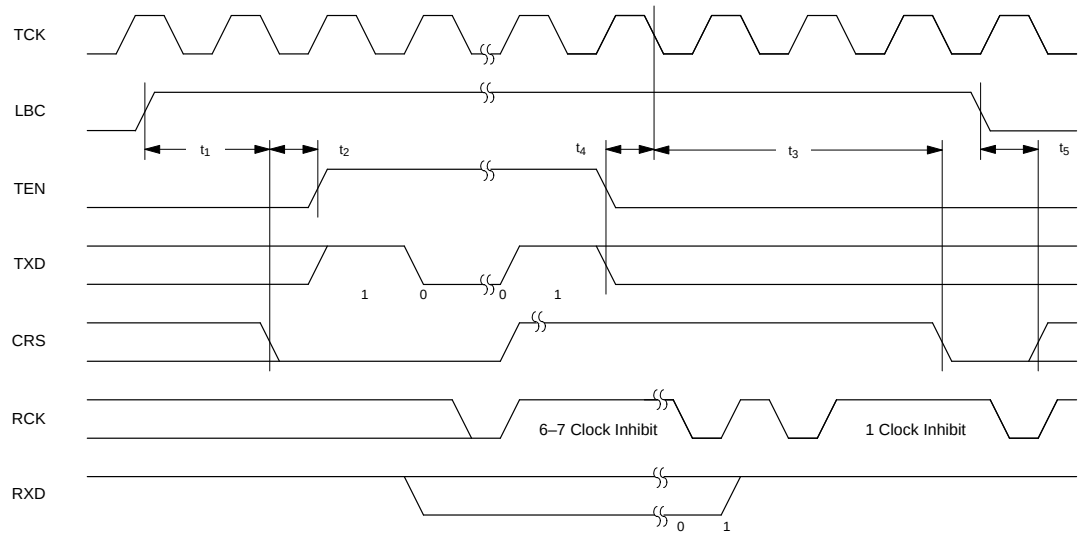
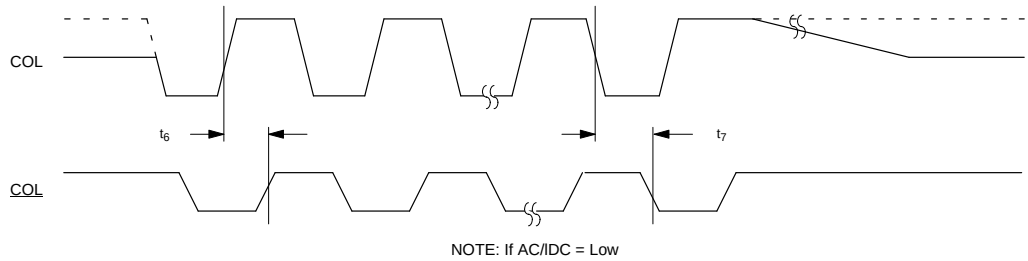


Figure 33. Loopback Timing

Table 33. Loopback Timing (for Encoder/Decoder bypass mode)

Symbol	Parameter Description	Min.	Typ.	Max.	Units
t_1	LBC receiving data purge time	—	260	—	ns
t_2	Wait time from CRS low to TEN high	9.6	—	—	s
t_3	Data through time	—	280	—	ns
t_4	TXD, TEN setup to me to TCK	0	—	—	ns
t_5	LBC receiving data accept time	—	80	—	ns



NOTE: If AC/IDC = Low

Figure 34. Collision Timing

Table 34. Collision Timing (for Encoder/Decoder Test mode)

Symbol	Parameter Description	Min.	Typ.	Max.	Units
t_6,t_7	COL± to COL propagation delay time	15	50	—	ns

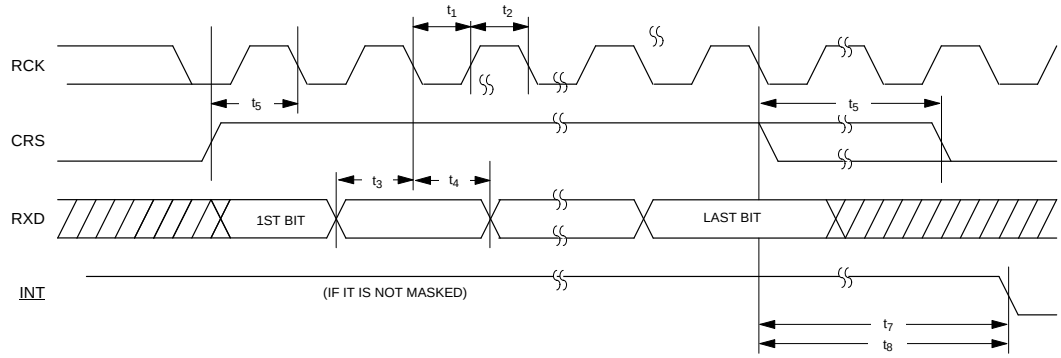


Figure 35. Receive Timing

Table 35. Receive Timing (for Encoder/Decoder Bypass mode)

Symbol	Parameter Description	Min.	Typ.	Max.	Units
t_1	Receive clock low width	35			ns
t_2	Receive clock high width	35			ns
t_3	Receive data setup	10			ns
t_4	Receive data hold	10			ns
t_5	Receive carrier sense setup	10			ns
t_6	Receive carrier sense hold			7	RCK cycles
t_7	Last bit of good packet received to interrupt		40		RCK cycles
t_8	Receive interrupt after bad packet		15		RCK cycles

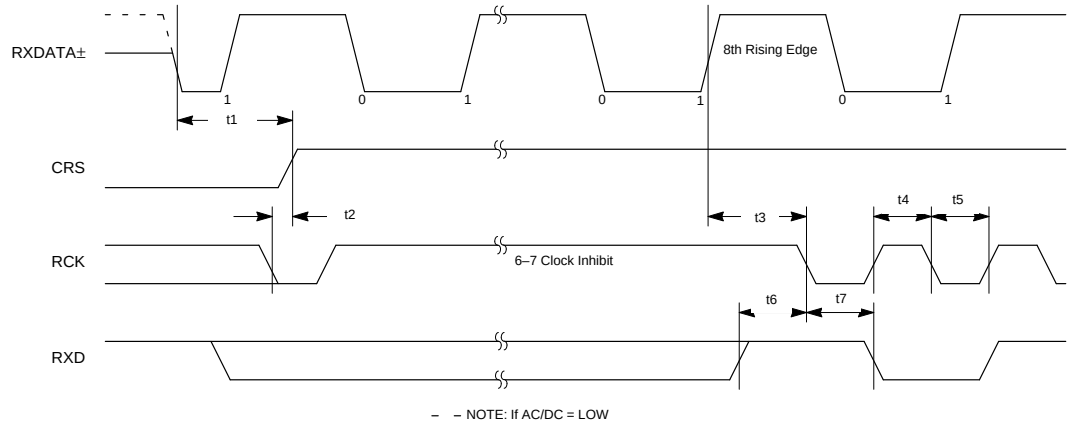


Figure 36. Receive Start Timing

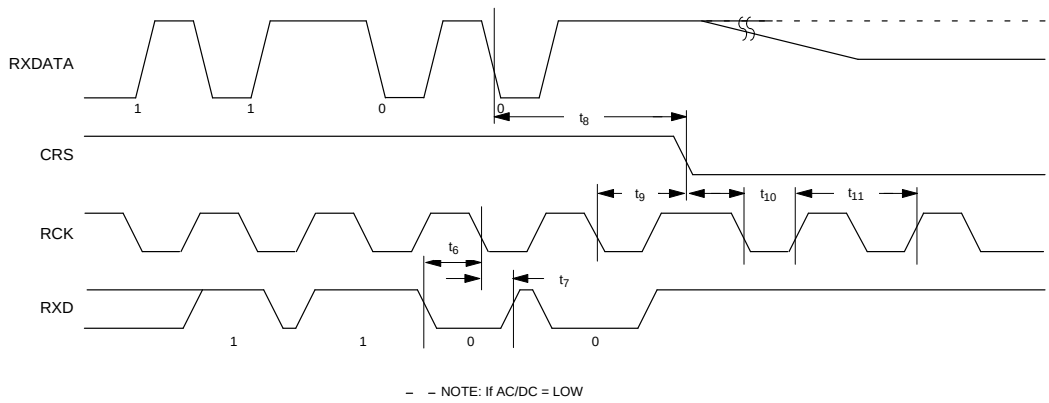


Figure 37. Receive End Timing

Table 36. Receive Timing: Figures 36–37 (for Encoder/Decoder Test mode)

Symbol	Figure	Parameter Description	Conditions	Min.	Typ.	Max.	Units
t_1	36	CRS on delay time		—	90	120	ns
t_2	36	CRS low hold time		10	—	—	ns
t_3	36	RCK delay time		—	125	—	ns
t_4	36	RCK high time		35	50	—	ns
t_5	36	RCK low time		35	50	—	ns
t_6	36, 37	RXD setup time to RCK		20	60	—	ns
t_7	36, 37	RXD hold time from RCK		10	20	—	ns
t_8	37	CRS off delay time		—	260	—	ns
t_9	37	CRS high hold time		—	130	—	ns
t_{10}	37	CRS low setup time		—	70	—	ns
t_{11}	37	RCK cycle time during idle		99.99	100	100.01	ns

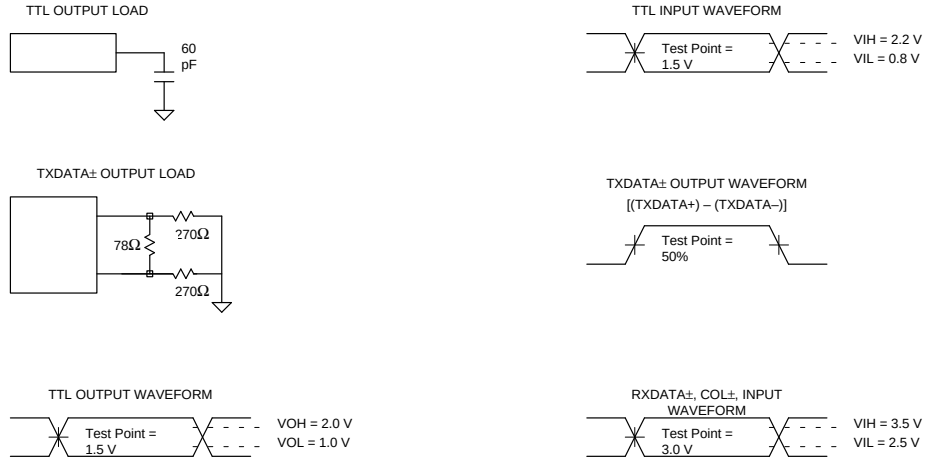
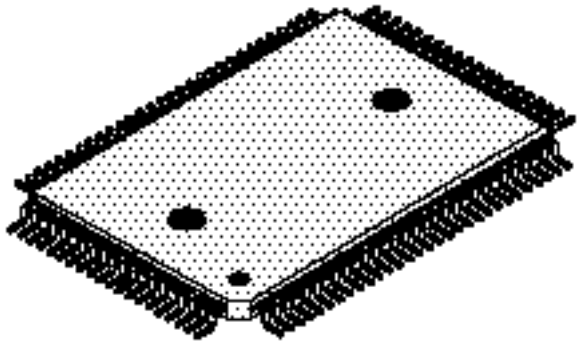


Figure 38. Test Conditions

100-Pin Plastic Flat Package



Ordering Information: MB86960APF-G

100-Lead Plastic Flat Package
(Case No. FPT-100P-M06)