MOTOROLA

Microprocessors and Memory
Technologies Group

MC68EN302

Integrated Multiprotocol Processor with Ethernet Reference Manual

Literature Distribution Centers:
USA/EUROPE: Motorola Literature Distribution; P.O. Box 20912, Arizona 85036.
JAPAN: Nippon Motorola Ltd.; 4-32-1, Nishi-Gotanda, Shinagawa-ku, Tokyo 141 Japan.
ASIA-PACIFIC: Motorola Semiconductors H.K. Ltd.; Silicon Harbour Center, No. 2 Dai King Street, Tai Po Industrial Estate,

Preface

The complete documentation package for the MC68EN302 consists of MC68EN302RM/AD, the *MC68EN302 Integrated Multiprotocol Processor with Ethernet,* MC68302UM/AD, the *MC68302 Integrated Multiprotocol Processor,* MC68EN302/D, the *MC68EN302 Integrated Multiprotocol Processor with Ethernet Product Brief*; and M68000PM/AD, the *M68000 Programmer's Reference Manual*.

The *MC68EN302 Integrated Multiprotocol Processor with Ethernet User's Manual* describes the programming, capabilities, registers, and operation of the MC68EN302 that differs from the MC68302; the *MC68302 Integrated Multiprotocol Processor describes the original MC68302,* the *MC68EN302 Integrated Multiprotocol Processor with Ethernet Product Brief* provides a brief description of the MC68EN302 capabilities; and the *M68000 Programmer's Reference Manual* describes programming and the instruction set for the IMP processor.

This user's manual is organized as follows:

Section 1	Introduction
Section 2	Module Bus Controller
Section 3	DRAM Control Module (DCM)
Section 4	Ethernet Controller
Section 5	Signal Description
Section 6	Applications
Section 7	IEEE 1149 Test Access Port (TAP)
Section 8	Electrical Specifications
Section 9	Ordering Information and Mechanical Data

Applications and Technical Information

For questions or comments pertaining to technical information, questions, and applications, please contact one of the following sales offices nearest you.

— Sales Offices —

UNITED STATES

ALABAMA, Huntsville	(205) 464-6800
ARIZONA, Tempe	(602) 897-5056
CALIFORNIA, Agoura Hills	(818) 706-1929
CALIFORNIA, Los Angeles	(310) 417-8848
CALIFORNIA, Irvine	(714) 753-7360
CALIFORNIA, Rosevllle	(916) 922-7152
CALIFORNIA, San Diego	(619) 541-2163
CALIFORNIA, Sunnyvale	(408) 749-0510
COLORADO, Colorado Springs	(719) 599-7497
COLORADO, Denver	(303) 337-3434
CONNECTICUT, Wallingford	(203) 949-4100
FLORIDA, Maitland	(407) 628-2636
FLORIDA, Pompano Beach/ Fort Lauderdal	(305) 486-9776
FLORIDA, Clearwater	(813) 538-7750
GEORGIA, Atlanta	(404) 729-7100
IDAHO, Boise	(208) 323-9413
ILLINOIS, Chicago/Hoffman Estates	(708) 490-9500
INDIANA, Fort Wayne	(219) 436-5818
INDIANA, Indianapolis	(317) 571-0400
INDIANA, Kokomo	(317) 457-6634
IOWA, Cedar Rapids	(319) 373-1328
KANSAS, Kansas City/Mission	(913) 451-8555
MARYLAND, Columbia	(410) 381-1570
MASSACHUSETTS, Marborough	(508) 481-8100
MASSACHUSETTS, Woburn	(617) 932-9700
MICHIGAN, Detroit	(313) 347-6800
MINNESOTA, Minnetonka	(612) 932-1500
MISSOURI, St. Louis	(314) 275-7380
NEW JERSEY, Fairfield	(201) 808-2400
NEW YORK, Fairport	(716) 425-4000
NEW YORK, Hauppauge	(516) 361-7000
NEW YORK, Poughkeepsie/Fishkill	(914) 473-8102
NORTH CAROLINA, Raleigh	(919) 870-4355
OHIO, Cleveland	(216) 349-3100
OHIO, Columbus Worthington	(614) 431-8492
OHIO, Dayton	(513) 495-6800
OKLAHOMA, Tulsa	(800) 544-9496
OREGON, Portland	(503) 641-3681
PENNSYLVANIA, Colmar	(215) 997-1020
Philadelphia/Horsham	(215) 957-4100
TENNESSEE, Knoxville	(615) 690-5593
TEXAS, Austin	(512) 873-2000
TEXAS, Houston	(800) 343-2692
TEXAS, Plano	(214) 516-5100
VIRGINIA, Richmond	(804) 285-2100
WASHINGTON, Bellevue	(206) 454-4160
Seattle Access	(206) 622-9960
WISCONSIN, Milwaukee/Brookfield	(414) 792-0122

Field Applications Engineering Available Through All Sales Office

CANADA

BRITISH COLUMBIA, Vancouver	(604) 293-7605
ONTARIO, Toronto	(416) 497-8181
ONTARIO, Ottawa	(613) 226-3491
QUEBEC, Montreal	(514) 731-6881

INTERNATIONAL

AUSTRALIA, Melbourne	(61-3)887-0711
AUSTRALIA, Sydney	(61(2)906-3855
BRAZIL, Sao Paulo	55(11)815-4200
CHINA, Beijing	86 505-2180
FINLAND, Helsinki	358-0-35161191
Car Phone	358(49)211501
FRANCE, Paris/Vanves	33(1)40 955 900
GERMANY, Langenhagen/ Hanover	49(511)789911
GERMANY, Munich	49 89 92103-0
GERMANY, Nuremberg	49 911 64-3044
GERMANY, Sindelfingen	49 7031 69 910
GERMANY,Wiesbaden	49 611 761921
HONG KONG, Kwai Fong	852-4808333
Tai Po	852-6668333
INDIA, Bangalore	(91-812)627094
ISRAEL, Tel Aviv	972(3)753-8222
ITALY, Milan	39(2)82201
JAPAN, Aizu	81(241)272231
JAPAN, Atsugi	81(0462)23-0761
JAPAN, Kumagaya	81(0485)26-2600
JAPAN, Kyushu	81(092)771-4212
JAPAN, Mito	81(0292)26-2340
JAPAN, Nagoya	81(052)232-1621
JAPAN, Osaka	81(06)305-1801
JAPAN, Sendai	81(22)268-4333
JAPAN, Tachikawa	81(0425)23-6700
JAPAN, Tokyo	81(03)3440-3311
JAPAN, Yokohama	81(045)472-2751
KOREA, Pusan	82(51)4635-035
KOREA, Seoul	82(2)554-5188
MALAYSIA, Penang	60(4)374514
MEXICO, Mexico City	52(5)282-2864
MEXICO, Guadalajara	52(36)21-8977
Marketing	52(36)21-9023
Customer Service	52(36)669-9160
NETHERLANDS, Best	(31)49988 612 11
PUERTO RICO, San Juan	(809)793-2170
SINGAPORE	(65)2945438
SPAIN, Madrid	34(1)457-8204
or	34(1)457-8254
SWEDEN, Solna	46(8)734-8800
SWITZERLAND, Geneva	41(22)7991111
SWITZERLAND, Zurich	41(1)730 4074
TAIWAN, Taipei	886(2)717-7089
THAILAND, Bangkok	(66-2)254-4910
UNITED KINGDOM, Aylesbury	44(296)395-252

FULL LINE REPRESENTATIVES

COLORADO, Grand Junction	
Cheryl Lee Whltely	(303) 243-9658
KANSAS, Wichita	
Melinda Shores/Kelly Greiving	(316) 838 0190
NEVADA, Reno	
Galena Technology Group	(702) 746 0642
NEW MEXICO, Albuquerque	
S&S Technologies, Inc.	(505) 298-7177
UTAH, Salt Lake City	
Utah Component Sales, Inc.	(801) 561-5099
WASHINGTON, Spokane	
Doug Kenley	(509) 924-2322
ARGENTINA, Buenos Aires	
Argonics, S.A.	(541) 343-1787

HYBRID COMPONENTS RESELLERS

Elmo Semiconductor	(818) 768-7400
Minco Technology Labs Inc.	(512) 834-2022
Semi Dice Inc.	(310) 594-4631

DOCUMENTATION FEEDBACK

FAX 512-891-8593—Documentation Comments Only (no technical questions please)
http: / / www.mot.com/hpesd/docs_survey.html—Documentation Feedback Only

The Motorola Technical Communications Department welcomes your suggestions for improving our documentation and encourages you to complete the documentation feedback form at the World Wide Web address listed above. In return for your efforts, you will receive a small token of our appreciation. Your help helps us measure how well we are serving your information requirements.

We also provide a fax number for you to submit any questions or comments about this document or how to order other documents. Please provide the part number and revision number (located in upper right-hand corner of the cover) and the title of the document. When referring to items in the manual, please reference by the page number, paragraph number, figure number, table number, and line number if needed. Please do not fax technical questions to this number.

When sending a fax, please provide your name, company, fax number, and phone number including area code.

For Internet Access:

Telnet: pirs.aus.sps.mot.com (Login: pirs)
WWW: http: / / pirs.aus.sps.mot.com/aesop/hmpg.html
Query By Email: aesop_query@pirs.aus.sps.mot.com
(Type ‘‘HELP’’ in text body.)

For Dial-Up:

Phone: +1-512-891-3650
Phone (US or Canada): 1-800-843-3451
Connection Settings: N/8/1/F
Data Rate: < 14,400 bps
Terminal Emulation: VT100
Login: pirs

For AESOP Questions:

FAX: +1-512-891-8775
EMAIL: aesop_sysop@pirs.aus.sps.mot.com

For Hotline Questions:

FAX (US or Canada): 1-800-248-8567
EMAIL: aesop_support@pirs.aus.sps.mot.com

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Paragraph Number	Title	Page Number

SECTION 1
INTRODUCTION

The MC68EN302 is a multiprotocol integrated communications controller based on the MC68302. The original MC68302 provided multiple WAN and ISDN support with three serial communcations channels, glueless memory control for SRAM and EPROM and various system integration features. The MC68EN302 builds upon the success of the MC68302 by adding an Ethernet controller which is completely independent of the three on-board serial channels as well as a DRAM control and a JTAG interface. No communications related features of the original 302 are lost when using either the Ethernet controller or the DRAM controller of the MC68EN302.

The Ethernet controller provides a 16-bit interface and provides complete IEEE 802.3 compatibility. The programming model for the Ethernet controller is based on the standard MC68302 programming model. Buffer descriptors for the Ethernet controller are compatible wiith the buffer descriptors used by the MC68360 QUICC Ethernet controller.

The DRAM controller is based upon other 300 family memory controllers with specific enhancements provided for supporting parity and external bus masters.

The JTAG interface is the standard IEEE1149.1 test interface.

1.1 FEATURE LIST

The following MC68EN302 features are in addition to the MC68302 feature list:

- Full complement of existing three SCC’s plus Ethernet channel
- Ethernet channel fully compliant with IEEE 802.3 MAC Specification.
  - — Supports data rates up to 10 Mbps.
  - — Supports the MC68302 style programming model.
  - — Bus bandwidth requirements reduced through 128 on-chip buffer descriptors.
  - — Independant 128 byte transmit and receive FIFO’s.
  - — 64 entry CAM for Address Recognition.
  - — Ethernet collision results in retransmission from TX FIFO (no external bus access).
  - — Runt frames automatically cause RX FIFO to flush internally.
  - — Interfaces to MC68160 for 10Base-T or AUI Connection.
- Dynamic Bus Sizing
- Glueless ROM and SRAM interface
- DRAM Controller
  - — Glueless DRAM interface for internal bus master

— AMUX signal provided for external bus master use
— Parity generation/checking on a per byte basis

- Fully IEEE 1149.1 JTAG Compliant
- 144 TQFP package

1.2 BLOCK DIAGRAM

The MC68EN302 adds functionality to the pre-existing MC68302 by providing additional blocks external to the MC68302 which arbitrate for use of the 68000 bus for access to off-chip resources such as memory or other peripheral devices. This modular approach is shown in Figure .

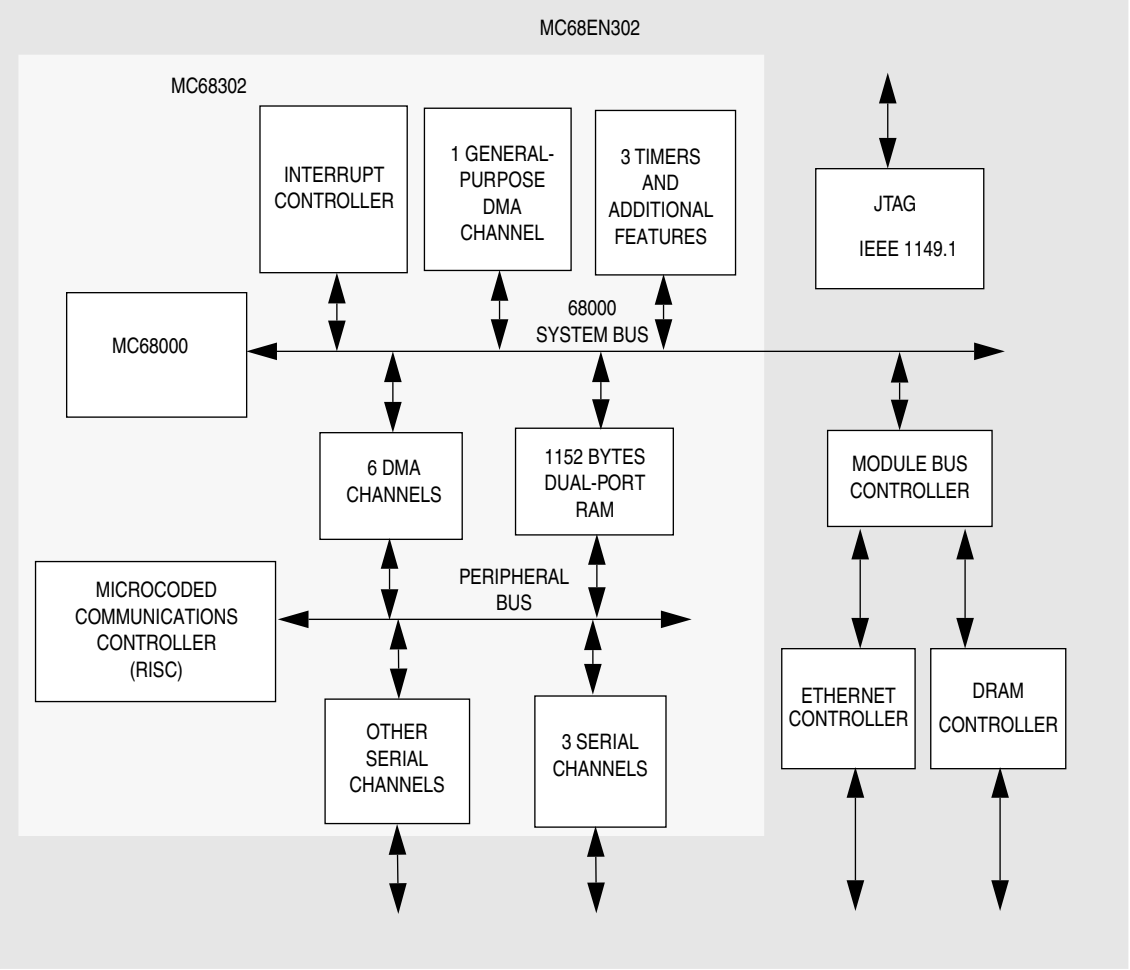


Figure 1-1. MC68EN302 Block Diagram

1.3 MEMORY MAP

The MC68EN302 memory map does not change the MC68302 memory map, but rather adds a new 4K module block. This is in addition to the 4K module block of the MC68302. Because of the additional register block, there are two Base Address Registers to program

in the MC68EN302. The BAR register is identical to the MC68302 BAR, and the Module Controller Base Address Register is specific to the MC68EN302.

1.3.1 Module Controller Base Address Register (MOBAR) Address ($EE)

The Module Controller Base Address Register (MOBAR) sets the base address for the MC68EN302 registers which are in addition to the register set of the MC68302. The MOBAR is located at address $0EE and its configuration and operation match the existing MC68302 BAR. The value of MOBAR after reset defaults to $BFFE which places the Module Controller Block directly below the MC68302 Block. The MC68EN302 must be in supervisor mode for MOBAR to be written with a new value.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FC2	FC2	FC0	CFC	BA23	BA22	BA21	BA20	BA19	BA18	BA17	BA16	BA15	BA14	BA13	BA12

FC2–FC0—Function Code 2–Function Code 0

FC2-0 sets the address space of the 4 kbyte Module Controller Block. Depending on the value of the CFC bits, the MC68EN302 address compare logic uses these bits to cause an address match within its address space.DO NOT assign FC2-0 to the M68000 interrupt acknowledge space (FC2-0 = 111b).

CFC—Compare Function Code

When cleared, the FC bits in the MOBAR are ignored and accesses to the Module Controller Block occur without comparing the FC bits. When set, the address space compare logic uses the FC bits in MOBAR to detect address matches.

MOBA—Module Controller Base Address.

The high address field is contained in bits 11-0 of the MOBAR and sets the starting address of the Module Controller Block.

1.4 REGISTER OVERVIEW

The control and status registers for the Ethernet controller are all 16 bits with an address range of MOBA+$000 to MOBA+$FFF. The MC68EN302 registers in addition to the 302 register set are shown in Table 1-1. Note that even though the entire 302 register set is provided, special care must be taken when initializing the pre-existing 302 registers so there are no contention or compatibility issues during internal arbitration. The registers that require particular attention are:

- OR
- GIMR
- SCR.

Also, notice that DTACK is not returned for accesses to unimplemented CSRs in the MOBA address space.

Table 1-1. MC68EN302 Additional Registers

ADDRESS	NAME	MNEMONIC	TYPE
MOBA + 000	Module Bus Control	MBC	Read/Write
MOBA + 002	Interrupt Extension Register	IER	Read/Write
MOBA + 004	Chip Select 0 Extension Register	CSER0	Read/Write
MOBA + 006	Chip Select 1 Extension Register	CSER1	Read/Write
MOBA + 008	Chip Select 2 Extension Register	CSER2	Read/Write
MOBA + 00A	Chip Select 3 Extension Register	CSER3	Read/Write
MOBA + 00C	Parity Control & Status Register	PCSR	Read/Write
MOBA + 010	DRAM Configuration Register	DCR	Read/Write
MOBA + 012	DRAM Refresh Register	DRFRSH	Read/Write
MOBA + 014	DRAM Bank 0 Base Address Register	DBA0	Read/Write
MOBA + 016	DRAM Bank 1 Base Address Register	DBA1	Read/Write
MOBA + 800	Ethernet Control Register	ECNTRL	Read/Write
MOBA + 802	Ethernet DMA Configuration Register	EDMA	Read/Write
MOBA + 804	Maximum Receive Buffer Length	EMRBLR	Read/Write
MOBA + 806	Interrupt Vector Register	INTR_VECT	Read/Write
MOBA + 808	Interrupt Event	INTR_EVENT	Read/Write
MOBA + 80A	Interrupt Mask Register	INTR_MASK	Read/Write
MOBA + 80C	Ethernet Configuration	ECNFIG	Read/Write
MOBA + 80E	Ethernet Test Register	ETHER_TEST	Read/Write
MOBA + 810	Address Recognition Control Register	AR_CNTRL	Read/Write
MOBA + A00 MOBA + BFF	CAM Entry Table	CET	Read/Write
MOBA + C00 MOBA + FFF	Buffer Descriptors Table	EBD	Read/Write

SECTION 2
MC68EN302 MODULE BUS CONTROLLER

2.1 INTRODUCTION

The model of the MC68EN302 is such that the internal 302 functions are unaffected by the addition of an Ethernet controller and the DRAM controller. The 302 core sees that it must arbitrate with other bus masters for access to the ‘external’ bus. In the MC68EN302, the Module Bus Controller provides the arbitration between the 302 core and the other modules (Ethernet and DRAM) for access to the bus external to the MC68EN302. The functions provided by the Module Bus Controller (MBC) are as follows:

- Interfaces between internal 68000 bus and the Module Bus.
- Performs Dynamic Bus sizing utilizing the chip select logic of the internal 68302.
- Provides Interrupt handling for Module Bus modules.
- Performs bus arbitration between external sources, the Module Bus, and the 68302 core.

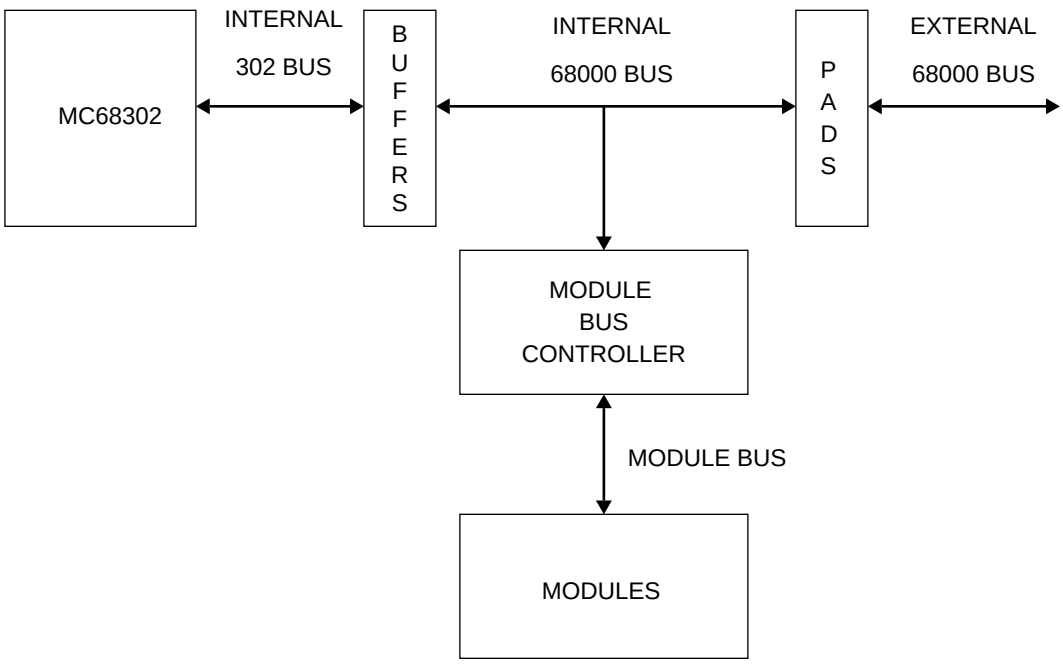


Figure 2-1. Top Level Bus Structure

2.2 TOP LEVEL MEMORY MAP

A top level diagram of register allocation for the modules in the MC68EN302 is shown in Table 2-1. A description of the DRAM control registers and the Ethernet Controller Registers are contained in the description of those modules.

Table 2-1. High Level Memory Map of MBC and MB Modules

Address	Block
MOBA + 000 MOBA + 00D	MBC Registers
MOBA + 010 MOBA + 019	DRAM Controller Registers
MOBA + 800 MOBA + FFF	Ethernet Controller Registers

2.3 MBC REGISTERS

A memory map of the MBC control registers is shown in Table 2-2.

Table 2-2. Module Bus Controller Register Set

ADDRESS	REGISTER NAME	MNEMONIC
MOBA + 000	Module Bus Control Register	MBCTL
MOBA + 002	Interrupt Extension Register	IER
MOBA + 004	Chip Select 0 Extension Register	CSER0
MOBA + 006	Chip Select 1 Extension Register	CSER1
MOBA + 008	Chip Select 2 Extension Register	CSER2
MOBA + 00A	Chip Select 3 Extension Register	CSER3
MOBA + 00C	Parity Control & Status Register	PCSR

2.4 MODULE BUS CONTROL (MBCTL)

The Module Bus Control register (MBCTL) provides the user control over the system level functionality of the MBC. This register defaults to $0x5000 upon hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BCE	MFC2	MFC1	MFC0	BB	PPE	PM9	PM8	PM7	PM6	PM5	PM4	PM3	PM2	PM1	PM0

BCE—Bus Clear Enable. This bit controls the way in which the MBC responds to the Bus Clear function inside the MC68EN302. If BCE is zero, the MBC ignores the Bus Clear ($\overline{\text{BCLR}}$) signal giving the DRAM and Ethernet modules priority over the 302 core. If this bit

is set, the MBC relinquishes the bus when it detects $\overline{\text{BCLR}}$, allowing the internal 302 core priority over the DRAM and Ethernet controllers.

MFC—Module Function Code (MFC2-MC0). These bits determine the function code put out when the Ethernet DMA machine is active.

BB—Bus Error Byte. This status bit (read-only) is the state of Address 0 upon the last generated bus error. This information is useful when performing exception processing to determine the cause of bus errors generated when the 8-bit dynamic bus sizing option is used with the Chip Selects.

PPE—Parity Pin Enable. This bit, if set, enables parity on the appropriate pins. The parity signals are muxed on three MC68EN302 configuration pins which are sampled at hard reset to determine device operation. Once out of reset, the parity function may be enabled by the PPE bit. See 2.10.4 Parity Pin Enable for more details.

PM—Pin Muxes PM9–PM0. Depending upon the setting of these bits, the MC68EN302 is able to provide some enhancements over the 68302. Because many of these enhancements are with existing 68302 pins, the enhancements are provided as programmable options. Table 2-3 shows the effect of the PM bits. All PM bits are cleared at hardware reset.

Table 2-3. Pin Muxing Operation

Mux Bit	Bit = 0 Pin Function	Bit = 1 Pin Function
PM0	$\overline{\text{AMUX}}$	BRG1
PM1	$\overline{\text{RAS0}}$	BRG2/SDS2/PA7
PM2	$\overline{\text{RAS1}}$	BRG3/PA12
PM3	$\overline{\text{CAS0}}$	PB0/$\overline{\text{IACK7}}$
PM4	$\overline{\text{CAS1}}$	PB1/$\overline{\text{IACK6}}$
PM5	$\overline{\text{DRAM\_RW}}$	PB2/$\overline{\text{IACK1}}$
PM6	A0	$\overline{\text{TOUT1}}$/PB4
PM7	$\overline{\text{DREQ}}$/PA13	$\overline{\text{WEL}}$
PM8	$\overline{\text{DACK}}$/PA14	$\overline{\text{WEH}}$
PM9	$\overline{\text{OE}}$	$\overline{\text{DONE}}$/PA15

2.5 INTERRUPT EXTENSION REGISTER (IER)

This register replaces the MOD, ET7, ET6, and ET1 bits in the pre-existing 302 GIMR (Global Interrupt Mode Register) requiring that when writing to the internal 302 core GIMR, the corresponding bits must be written as a zero.This register is $0x0000 upon hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMOD	0	0	0	MIL	IET7	IET6	IET1	0	0	0	0	0	0	0	0

IMOD—Interrupt Mode. This bit determines if the 3 interrupt inputs are configured as IPL pins or IRQ pins for the MC68EN302 and replace the MOD bit functionality in the internal

302 GIMR. For proper operation of the MC68EN302, the MOD bit must be zero in the internal 302 core.

0 = Configures the pins as $\overline{\text{IPL2}}$-$\overline{\text{IPL0}}$.
1 = Configures the pins as $\overline{\text{IRQ7}}$, $\overline{\text{IRQ6}}$, and $\overline{\text{IRQ1}}$.

Bits 14–12—Reserved. Should be written as zero. These bits are always read as zero.

MIL—Module Interrupt Level. This bit determines the interrupt level at which Module Bus Controller interrupts are generated. Because the interrupt level of the internal 302 core is set at 4, and this bit predetermines the Module Bus Controller interrupt at either level 3 or 5, external interrupts should not be generated at level 4 or the level preset by MIL.

0 = Interrupts are generated at level 5
1 = Interrupts are generated at level 3.

IET7—Interrupt Edge Trigger Level 7. This bit has no effect unless IMOD=1 and replaces the operation of the ET7 bit in the Global Interrupt Mode Register (GIMR) of the internal 302 core. The ET7 bit in the GIMR register must equal zero for correct interrupt operation

0 = An interrupt is made pending when $\overline{\text{IRQ7}}$ is low.
1 = An interrupt is made pending when $\overline{\text{IRQ7}}$ changes from a one to a zero (falling edge) of the MC68EN302.

IET6—Interrupt Edge Trigger Level 6. This bit is has no affect unless IMOD is one. This bit replaces the functionality of the ET6 bit in the Global Interrupt Mode Register (GIMR) of the internal 302 core. The ET6 bit in the GIMR register must be set to zero.

0 = An interrupt is made pending when $\overline{\text{IRQ6}}$ is low.
1 = An interrupt is made pending when $\overline{\text{IRQ6}}$ changes from a one to a zero (falling edge).

IET1—Interrupt Edge Trigger Level 1. This bit is has no effect unless IMOD is one. This bit replaces the functionality of the ET1 bit in the Global Interrupt Mode Register (GIMR) of the internal 302 core. The ET1 bit in the GIMR register must be set to zero.

0 = An interrupt is made pending when $\overline{\text{IRQ1}}$ is low.
1 = An interrupt is made pending when $\overline{\text{IRQ1}}$ changes from a one to a zero (falling edge).

Bits 7–0—Reserved. Should be written as zero. These bits are always read as zero.

2.6 CHIP SELECT EXTENSION REGISTERS (CSER3–CSER0)

These registers provide additional functionality above the 68302 chip selects including 8-bit bus operation and parity generation and checking. Before setting the FCE, DT2–DT0 or EN8 bits, be sure that an external $\overline{\text{DTACK}}$ is supported by programming the 302 DTACK field in the corresponding OR register to 111. These registers are initialized to 0x000C or 0x000D upon hard reset (refer to the EN8 bit for more detail).

If at $\overline{\text{RESET}}$, the 8-bit mode is selected through use of the PARITY1/BUSW pin, the DTACK field in OR0 of the 302 core is forced to 111. This results in the DT2–DT0 field of CSER0 controlling DTACK.

During reset, CS1, CS2 and CS3 are disabled via the EN bit in the BR1, BR2 and BR3 registers.

Note that when in disable CPU mode, the CS0 function is replaced by IOUT2.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	CSPE	0	0	FCE	DT2	DT1	DT0	EN8

Bits 15–8—Reserved. Should be written to zero by the host processor. These bits are always read as zero.

CSPE—Chip Select Parity Enable. This bit enables parity checking and generation when the corresponding Chip Select is generated. Unless the corresponding chip select is set to 8-bit operation, parity is generated and checked on both bytes.

Bits 6–5—Reserved. Should be written to zero by the host processor. This bit is always read as zero.

FCE—Fast Cycle Enable. This bit enables fast mode operation. When using fast cycles, $\overline{CS}$ and $\overline{AS}$ are not negated between 8 bits of a 16 bit transfer, allowing a 16-bit transfer to occur on an 8-bit bus in 5 clocks rather than 6.

DT2–DT0—DTACK. These bits are used to determine whether $\overline{DTACK}$ is generated internally with a programmable number of wait states or externally by the peripheral. When done internally, the MC68EN302 provides the option of allowing 16-bit accesses to take place in two-three clock external 8-bit accesses. The 68000 only sees a single six clock access internally during this mode of operation. This functionality is also referred to as ‘minus one wait state option.’ Note that an 8-bit operand access requires a 4 clock bus cycle. Table 2-4 shows how the bits are encoded.

Table 2-4. DT Bit Encoding

DT BIT ENCODING	WAIT STATES
000	-1
001	0
010	1
011	2
100	3
101	4
110	5
111	No $\overline{DTACK}$

EN8—Enable 8-bit chip select. When set to a one, the 8-bit chip select operation is enabled. If the system is booted from an 8-bit memory, the system must drive the BUSW pin low during system reset which sets the EN8 bit for all four CSER registers. This assures that the device is able to access 8-bit memories as well as 16-bit memories. In 8-bit mode bits D15–D8 of the data bus are used.

2.7 PARITY CONTROL AND STATUS REGISTER (PCSR)

This register controls and gives the status of the parity checking portions of the parity circuitry. This register is set to 0x0000 upon hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIE	OPAR	PEC3	PEC2	PEC1	PEC0	PED1	PED0	PIV7	PIV6	PIV5	PIV4	PIV3	PIV2	PIV1	PIV0

PIE—Parity Error Interrupt Enable. This bit determines if an interrupt is generated when a parity error is detected.

0 = No interrupt is generated.
1 = Either a level 3 or level 5 interrupt is generated, depending upon the encoding of the MIL bit in the IER register.

OPAR—Odd Parity. This bit is used to determine if odd or even parity is used.

0 = Parity is even.
1 = Parity is odd.

PEC—Parity Error Chip Selects (PEC3–PEC0). These status bits indicate that there was a parity error in the corresponding Chip Select Bank. If one of the three bits is set to one, a parity error is detected in the corresponding bank. If the PIE bit is set, a level 5 (or 3) interrupt is driven to the processor as long as one of the PEC3–PEC0 bits are set. PEC3–PEC0 are sticky bits which are cleared when a one is written to them or upon hardware reset. Writing a zero does not change the value of the PEC bits. The $\overline{\text{PARITYE}}$ pin is asserted until the PEC3–PEC0 bits are all cleared.

NOTE

If the Parity Pin Enable bit (PPE in MBC CSR) = 0 and parity is enabled with CSPE in CSER3–CSER0, then a parity error will be reported on the associated PEC bit.

PED—Parity Error DRAM (PED1–PED0). These status bits indicate that there was a parity error in the corresponding DRAM bank. If one of the two bits is set to one, a parity error is detected in the corresponding bank. If the PIE bit is set, a level 5 (or 3) interrupt is driven to the processor as long as one of the PED bits is set. PED are sticky bits which are cleared when a one is written to them or upon hardware reset. Writing a zero does not change the value of the PED bits. Writing a zero does not change the value of the PED bits. The $\overline{\text{PARITYE}}$ pin is asserted as long as a PED bit is set.

NOTE

If the Parity Pin Enable bit (PPE in MBC CSR) = 0 and parity is enabled on the DRAM interface (PE1 and/or PE0 = 1 in DCR) then a parity error will be reported on PED1–PED0.

PIV—Parity Error Interrupt Vector (PIV7–PIV0). If the PIE bit is set, a parity error generates a level 5 (or 3) interrupt. The PIV bits determine what interrupt vector is returned in response to a level 5 (or 3) parity error interrupt.

2.8 BUS INTERFACE

The MBC is responsible for determining the source of the bus mastership (module bus, external 68K bus or internal 302 core) and for controlling the direction of the buses. The layer of buffering between the internal 302 and internal 68000 buses mimics the operation of the 68302 external bus, giving the module bus the appearance of an external master from the viewpoint of the internal 302 core. The MBC does not affect the operation of the bus outside the MC68EN302 unless it arbitrates for that bus and is given bus mastership. The operation of the MBC as bus master is such that the bus external to the MC68EN302 operates as if an existing 302 peripheral is bus master. This is accomplished by:

- Providing I/O control at the pad which overrides the existing 68302 I/O control.
- Intercepting the external arbitration signals and merging them with the MBC arbitration.

2.8.1 Bus Arbitration

The MBC provides the circuitry which prioritizes the bus mastership requests in the following order (highest priority to lowest):

- External bus requests
- Module bus requests (Ethernet module)
- internal 302 core

2.9 DYNAMIC BUS SIZING

The MBC accommodates dynamic bus sizing by providing control to operate on the internal 68000 bus as a 16-bit device while simultaneously providing an 8-bit option externally. Control is provided via the EN8 bit in the CSER3–CSER0 registers and via the BUSW pin. The MBC routes data into the proper byte of the word D15–D8, increments the address, and runs a second bus cycle.

When reading and writing, the MBC will assure proper operation when performing even byte accesses, even word accesses, and odd byte accesses. Figure 2-4 shows the read cases, and Figure 2-5 shows the write cases. The even word access is the only case requiring two bus cycles. When an internal bus cycles is taking place, both $\overline{AS}$ and the appropriate chip select will be in operation (except when FAST CYCLES are used). The address is incremented only in the case of an even address access. Because of this, the address increment only involves setting A0 to one.

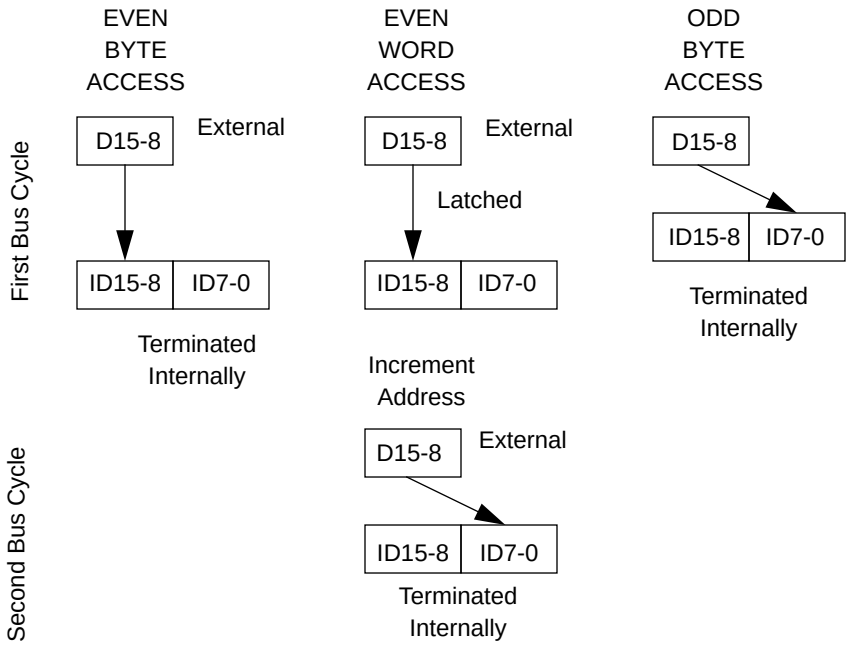


Figure 2-2. 8-bit External to 16-bit Internal Read

Note that an external access of 16-bits to an 8-bit port requires an external address increment as well as data muxing. It is generally recommended that external accesses using the 8-bit chip select extensions access only 8-bits at a time and access that data on the upper 8-bits of the bus.

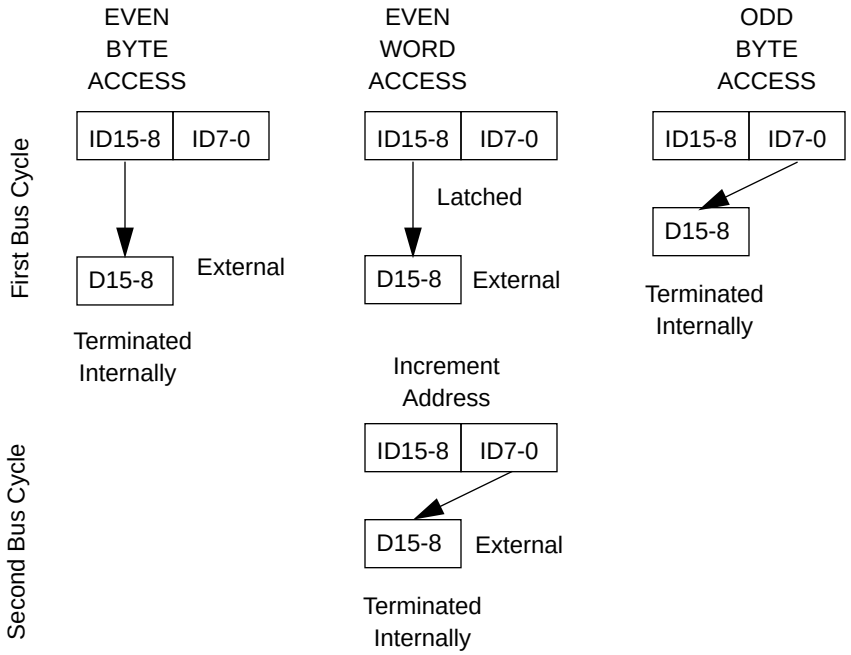


Figure 2-3. 16-bit Internal to 8-bit External Write

2.9.1 Bus Cycle Timing

The 8-bit extension logic for dynamic sizing is programmable from -1 to 5 wait states, and allows the external logic to terminate the cycle. Even word accesses are a special case which require specific control operations to allow the device to provide two external bus cycles. In some cases, the timing generated looks like two comparable 68000 cycles.There are two exceptions to this. First, the -1 wait state timing is a special case, since the 68000 bus is defined as a minimum 4 clock bus. The second exception is the Fast Cycle case. The -1 wait state case appears externally as two 16-bit 3 clock bus cycles. Figure 2-4 shows the timing for the read case, while Figure 2-5 shows the timing for writes.

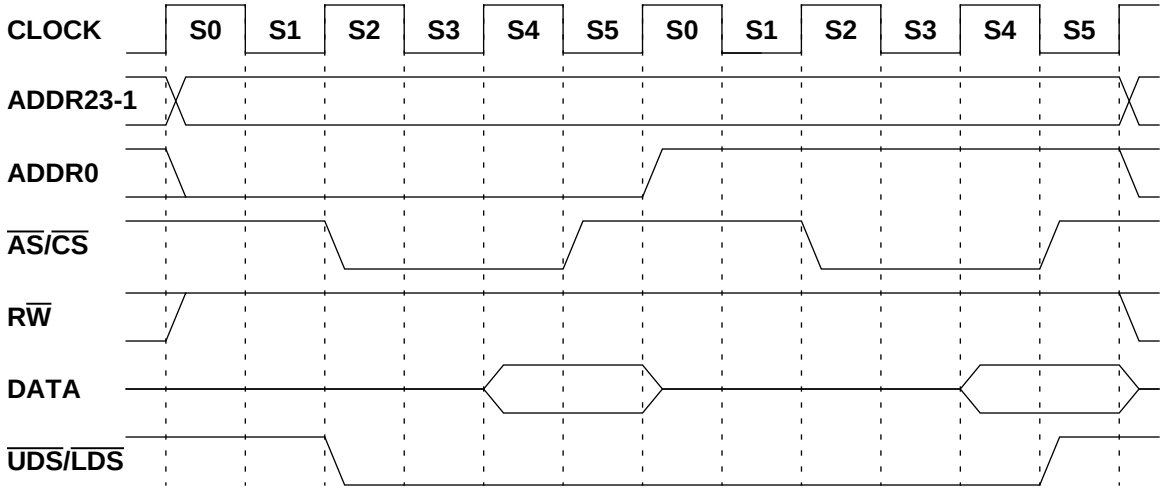


Figure 2-4. Word Read with 3-Clock 8-Bit Accesses

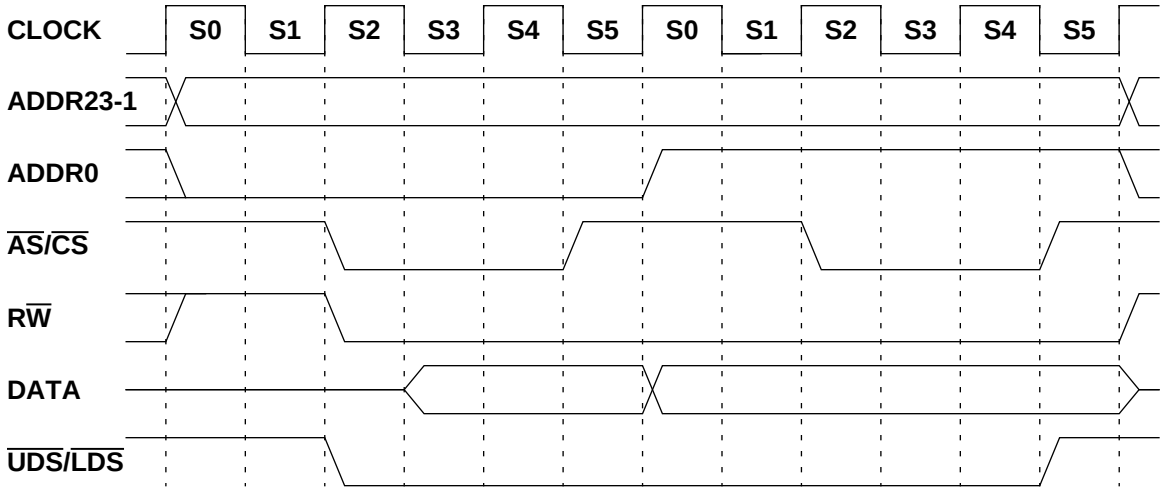


Figure 2-5. Word Write with 3-Clock 8-Bit Accesses

The fast cycle case differs from the normal cycle in that $\overline{AS}$ and chip select do not negate between the first and second half of the 16-bit access, allowing the two bus cycles to be

reduced one extra clock. A0 is incremented one half clock earlier in this scheme to allow sufficient address access in the second portion of the bus cycle. Figure 2-6 shows the fast cycle read case, while Figure 2-7 shows the fast cycle write case. In order to use fast cycles with SRAM, the $\overline{CS}$ signal is high between cycles, accommodating a write to SRAMs.

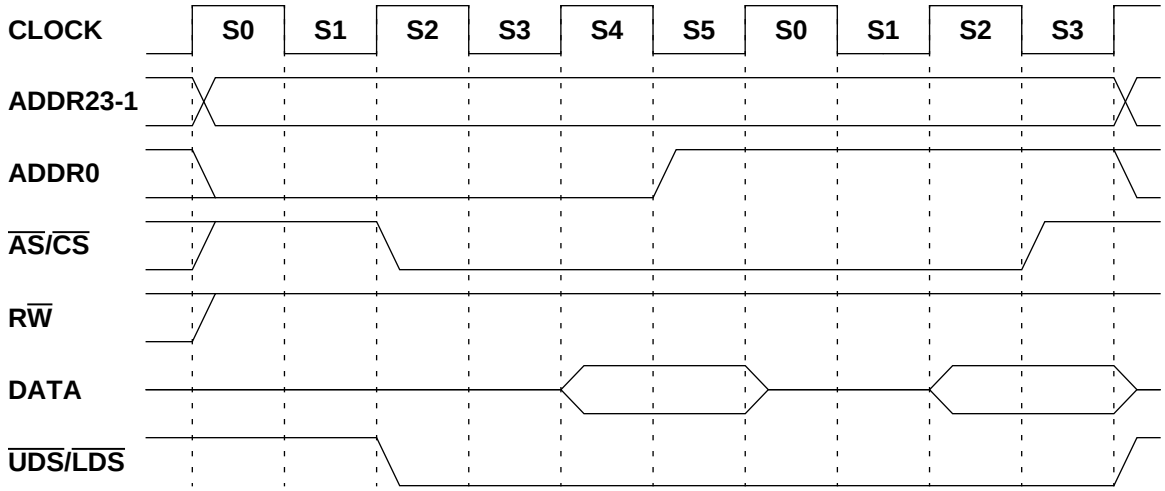


Figure 2-6. Fast Cycle Word Read with –1 Wait State

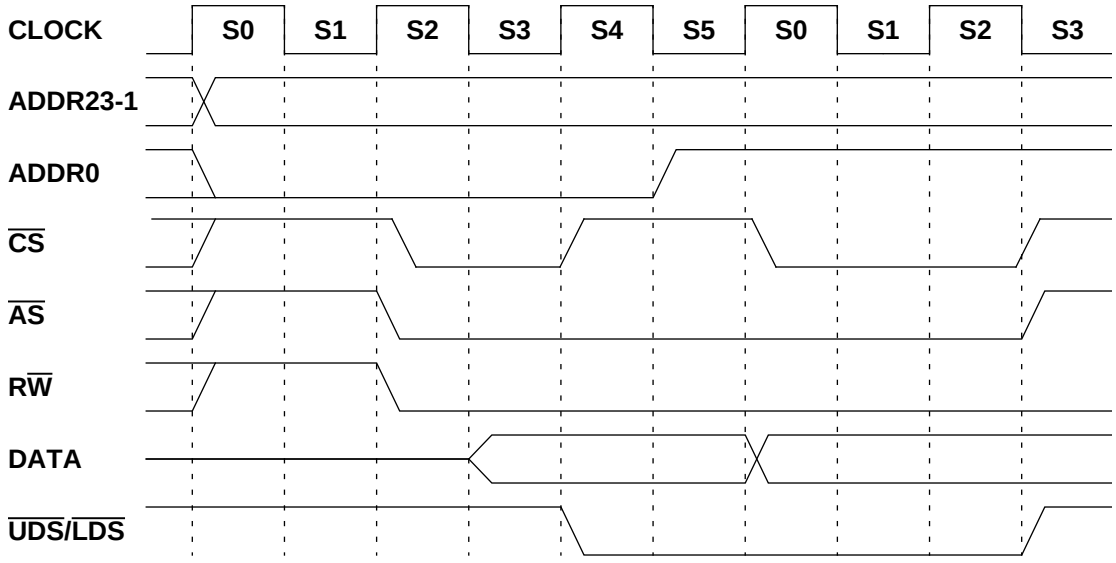


Figure 2-7. Fast Cycle Word Write with -1 Wait State

2.9.2 Bus Error Handling

Since the external bus may operate as an 8-bit bus, at the same time the internal bus operates as 16-bits, a bus error passed from the external bus to the internal bus does not specify which byte has been faulted. To assure that information is not lost, the Bus Error

Byte (BB) bit is provided in the MBC register. This bit reflects the state of A0 during the last bus error caused by an access to a byte peripheral.

2.9.3 Retry Handling

In most cases, an MC68EN302 retry is identical to 302 retry operation. If however, a retry occurs during the second bus cycle of a word access to an 8-bit port, the retry signal is passed to the initiating master. This causes both of the cycles to be retried, instead of just the second cycle.

2.10 PARITY LOGIC

The MC68EN302 provides parity support to generate, check, and report parity and parity errors.

2.10.1 Parity Generation

The MC68EN302 provides the option of generating and checking parity for the 4 chip selects and the 2 DRAM banks. In the case of a write, parity is generated with one bit of parity per byte of data. The parity is output on the parity pins and delayed from other data by the propagation delay through the parity generator.

2.10.2 Parity Checking

Parity checking is performed on read accesses. If the 8-bit option of the Chip Select logic is used, parity is checked on only the upper 8-bits. In all other options, parity is checked on both bytes.

2.10.3 Parity Error Reporting

Parity error reporting is accomplished via three mechanisms.

- Parity Error Pin

This pin is asserted when a parity error is detected. Parity error detection does not occur with enough time to generate a bus error on the affected cycle. The parity error pin may be used with external circuitry to facilitate parity error handling. This pin is not negated until all the parity error register bits are cleared.

- Parity Error Status Bits

There are 6 PCSR register bits dedicated to providing status on parity errors, corresponding to the 2 DRAM banks and the 4 Chip Selects. If a parity error is detected, the bit that corresponds to the module that generated the error is set. These bits are reset to zero and are cleared by writing a one.

- Parity Error Interrupt

The PIE bit in the PCSR register is provided to allow the option of generating a level 5 (or 3) interrupt in the event that a parity error is generated. If this option is selected, the interrupt is driven after the error is detected until the Parity Error Status Bits are cleared.

2.10.4 Parity Pin Enable

During hardware reset, the parity pin enable bit (PPE) in the MBC register (see 2.4 Module Bus Control (MBCTL)) is cleared, which results in the parity pins becoming inputs. Each of the three pins is sampled for a different function, as shown in Table 2-5. After exiting

Table 2-5. Parity Pin Enable Operation

PPE = 0 PIN FUNCTION	PPE = 1 PIN FUNCTION
DISCPU	PARITY0
BUSW	PARITY1
THREES	$\overline{\text{PARITYE}}$

hardware reset, these pins are sampled to determine chip functions. Pullup or pulldown resistors are required for presetting the desired state if the parity pins are to be later programmed as input/output pins. After hardware reset, the PPE bit can be set to enable the parity pins as outputs. The PPE bit should be set to enable parity even on reads.

2.11 INTERRUPT SUPPORT

All module bus and module bus controller interrupts are at level 5 or at level 3 if MIL is set (See 2.5 Interrupt Extension Register (IER)). There are two sources of interrupts in the MBC: One is the Ethernet controller; the second source is the parity error interrupt. The parity error interrupt is the higher priority of the two. If an interrupt acknowledge cycle is generated when both interrupts are asserted, the MBC responds to the parity error interrupt by driving its vector onto the internal 68000 bus. Only after the parity error interrupt is cleared will the Ethernet controller respond to an IACK cycle.

In order to accommodate an additional interrupt source within the MC68EN302, an additional Interrupt Mode (IMOD) bit is provided in the MBC. This bit configures the Interrupt pins as $\overline{\text{IRQ}}$ or $\overline{\text{IPL}}$ lines. This replaces the MOD bit in the Global Interrupt Mode Register (GIMR). This means that the existing MOD bit in the Global Interrupt Mode Register (GIMR) must always remain at zero. The IMOD bit in the IER register duplicates this function in the MBC.

Since the MBC generates a level 5 (or level 3) interrupt, and there is no way to resolve IACK conflicts with the external circuitry, a level 5 (or level 3) interrupt should not be asserted externally. Figure 2-8 summarizes the interrupt configuration and priorities.

Priority Level	Normal Mode $\overline{IPL2}$–$\overline{IPL0}$	Dedicated Mode $\overline{IRQ7}$, $\overline{IRQ6}$, $\overline{IRQ1}$	Interrupt Source
7 (Highest)	000	$\overline{IRQ7}$	External
6	001	$\overline{IRQ6}$	External
5	**	*	MBC/External
4	*	*	Internal 302
3	**	*	External/MBC
2	101	*	External
1 (Lowest)	110	$\overline{IRQ1}$	External

*Priority level not available to an external device in this mode
** The level not selected by MIL is available but not both level 3 and level 5.

Figure 2-8. External and Internal Interrupt Prioritization

The MBC passes the interrupt vector (see 4.1.4 Interrupt Vector Register (IVEC)) from the Module bus to the 68000 bus.

SECTION 3
MC68EN302 DRAM CONTROL MODULE

3.1 INTRODUCTION

The MC68EN302, like its predecessor the MC68302, can be connected with DRAM-type memories easily. The difference in the MC68EN302 lies in the DRAM Control Module (DCM), which was developed to provide seamless integration of the 68000 core with DRAM memories. The MC68EN302 DRAM controller is able to support up to two 16-bit wide banks and an address range from 128kbytes to 8Mbytes. Selection between the two banks occurs externally through the MC68EN302 $\overline{RAS1}$–$\overline{RAS0}$ signals, and byte selection occurs via the $\overline{CAS1}$–$\overline{CAS0}$ signals. The user is able to select cycle lengths ranging in duration from 4 to 7 clocks. The MC68EN302 also provides programmable refresh rates which can range anywhere from 16 to 4096 system clocks, or be disabled altogether.

3.2 MEMORY MAP

Table 3-1 shows the basic memory map of the DRAM Control Module registers.

Table 3-1. DRAM Controller Registers

ADDRESS	NAME	MNEMONIC	TYPE	FC
MOBA + 010	DRAM Configuration Register	DCR	Read/Write	S
MOBA + 012	DRAM Refresh Register	DRFRSH	Read/Write	S
MOBA + 014	DRAM Bank 0 Base Address Register	DBA0	Read/Write	S
MOBA + 016	DRAM Bank 1 Base Address Register	DBA1	Read/Write	S

3.3 DRAM CONFIGURATION REGISTER (DCR)

This register controls the specific operation of each bank of DRAM and is initialized to zero at hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	E1	E0	PE1	PE0	P1	P0	W1	W0	WP1	WP0	S/U1	S/U0

Bits 15–12—Reserved. Should be written to zero by the host processor. These bits are always read as zero.

E1-E0—Refresh Enable Bits.

0 = Disable refresh operation in the corresponding DRAM bank
1 = Enable refresh operation in the corresponding bank.

PE1-PE0—Enable Parity.

0 = Parity is generated but not checked
1 = Parity is generated on writes, and parity is checked on reads in the corresponding bank. If a parity error is detected the bus cycle is terminated with a bus error.

NOTE

If the Parity Pin Enable bit (PPE in MBC CSR) = 0 and parity is enabled on the DRAM interface (PE1 and/or PE0 = 1 in DCR) then a parity error will be reported on PED1–PED0.

P1-P0—RAS Precharge bits. These bits control the minimum number of clocks the RAS signal is precharged between bus cycles. Table 3-3 shows the encoding for these bits.

Table 3-2. Precharge Bit Encodings

P1	P0	PRECHARGE CLOCKS
0	0	2
0	1	3
1	0	4
1	1	5

W1-W0—Wait state bits. These bits control the number of wait states required for DRAM bank accesses. Table 3-3 shows the wait state bit encodings.

Table 3-3. Wait State Bit Encodings

W1	W0	WAIT STATES
0	0	0
0	1	1
1	0	2
1	1	3

WP1-WP0—Write Protect. This bit enables and disables write protection to a corresponding DRAM bank.

0 = The corresponding DRAM bank may be written.
1 = Write access to the corresponding DRAM bank returns a bus error.

S/U1-S/U0—Supervisor/User. This bit determines whether the given DRAM bank decodes to Supervisor Space (FC = 6 & 5) or both Supervisor and User (FC = 6 & 5 & 1 & 2) Space.

0 = Respond to Supervisor accesses only
1 = Respond to Supervisor and User Space.

3.4 DRAM REFRESH REGISTER (DRFRSH)

This register controls the operation of the refresh circuitry and is initialized to zero on hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	R7	R6	R5	R4	R3	R2	R1	R0

Bits 15–8—Reserved. Should be written to zero by the host processor. These bits are always read as zero.

R7-R0—Refresh Count Bits. The value stored in these bits is multiplied by 16 system clocks to determine the refresh period. The divide by 16 scheme provides sufficient range to address systems operating with standard DRAM at frequencies less than 2 Mhz as well as systems utilizing low power DRAM running at frequencies greater than 25 Mhz. All zeroes correspond to 4096 system clocks.

3.5 DRAM BASE ADDRESS REGISTER (DBA1-DBA0)

The base address registers for DRAM are two 16-bit registers which are initialized to zero at hardware reset. These registers hold both the base address of each bank and mask bits for determining which address bits initiate bus cycle accesses to the DRAM banks.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A23	A22	A21	A20	A19	A18	A17	0	0	M22	M21	M20	M19	M18	M17	V

A23-A17—Base Address Bits. The Base Address Bits determine where the DRAM bank is located on 128 kbyte boundaries. These bits are compared with the corresponding addresses generated by the MC68EN302 to determine if a given bus cycle accesses a particular DRAM bank. These bits are used in conjunction with the Mask Bits to determine the size and location of a given DRAM bank.

Bits 8–7—Reserved. Should be written to zero by the host processor. These bits are always read as zero.

M22-17—Mask Bits. These bits are used in conjunction with the Base Address Bits to determine the size and location of a given DRAM bank.

0 = The corresponding address bit is ignored.
1 = The address compare logic uses the corresponding address bit when determining if a bus cycle access occurs within the DRAM bank.

V—Valid Bit. This bit is cleared to 0 at hardware reset.

0 = This DRAM bank is not valid
1 = Data for the corresponding DRAM bank data is valid, and DRAM accesses are decoded by that bank’s circuitry.

3.6 DRAM CONTROL MODULE OPERATION

3.6.1 Reset Operation

Refresh accesses continue if the Ethernet module is reset and during the Reset Instruction (soft reset), but not during system (hardware) reset.

3.6.2 Read and Write Cycle Operation

Figure 3-1 shows two consecutive 4-clock accesses with a 2-clock precharge of $\overline{RAS}$. $\overline{RAS}$ is negated one clock before $\overline{CAS}$ to provide a longer precharge time between consecutive hits in a given bank.

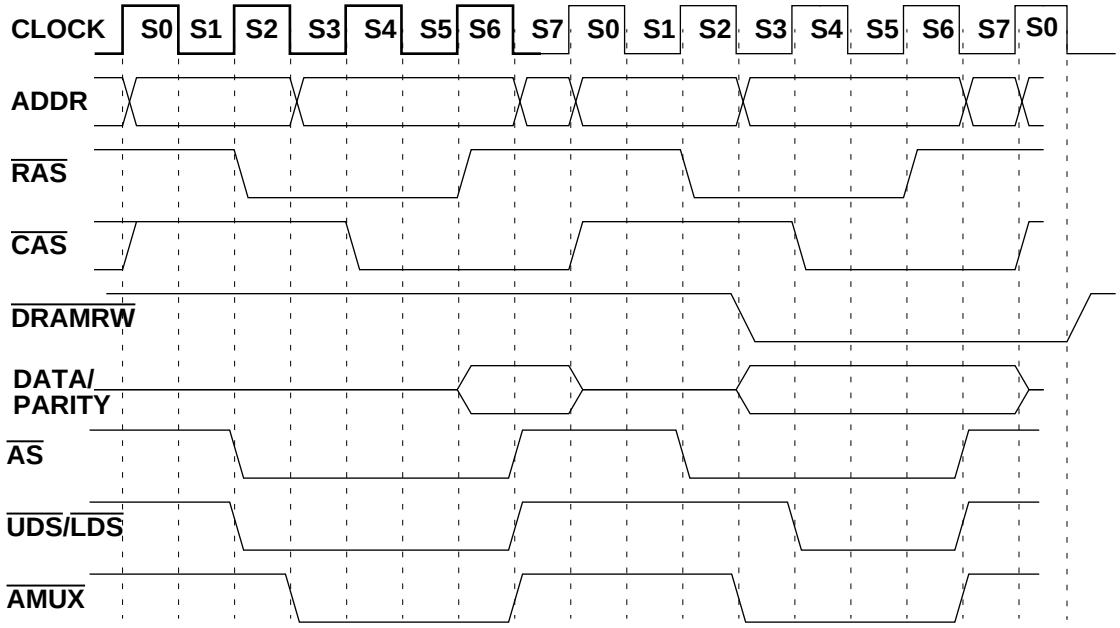


Figure 3-1. Consecutive Four-Clock Accesses

In some cases, a DRAM access is not possible in a 4-clock cycle. In still other cases, the precharge time of 2 clocks is not enough for the given DRAM. Figure 3-2 shows two consecutive 5-clock accesses, with a 3-clock $\overline{RAS}$ precharge. Note that the DRAM signals are delayed in the second cycle with respect to the processor bus signals, allowing a longer $\overline{RAS}$ precharge time.

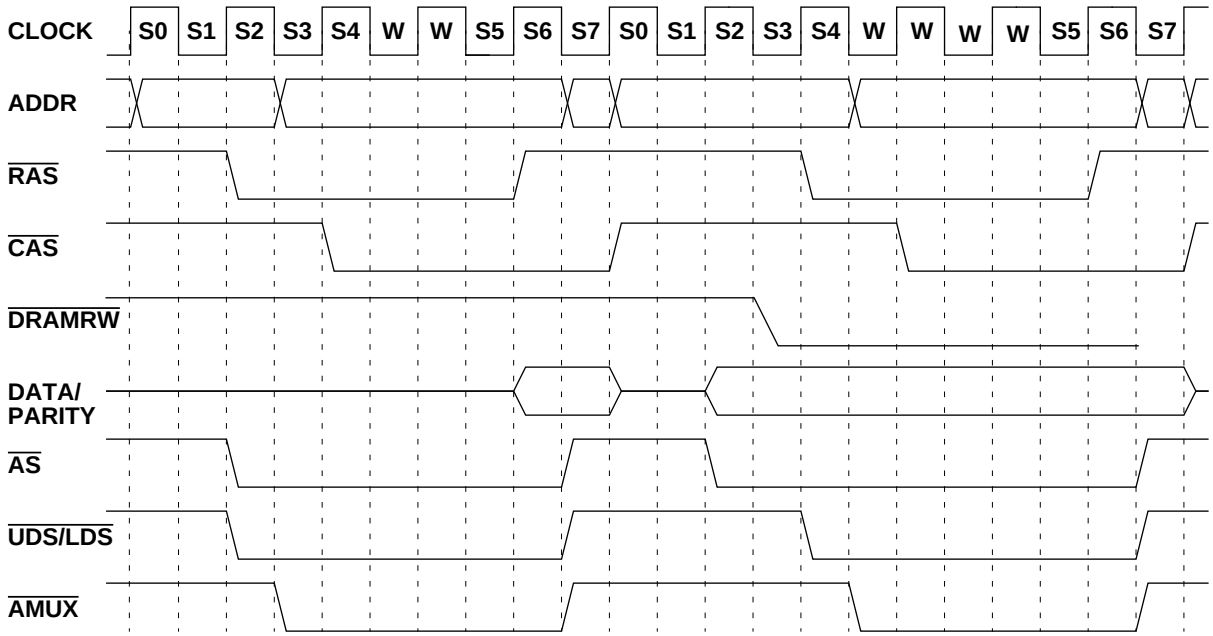


Figure 3-2. Five-Clock Accesses with Three-Clock Precharge

3.7 REFRESH OPERATION

The MC68EN302 supports $\overline{\text{CAS}}$ before $\overline{\text{RAS}}$ refresh but note that refresh operation is not synchronized to the bus activity. A special $\overline{\text{DRAMRW}}$ (Read/Write) pin is provided so that refresh may occur regardless of the state of the processor bus. Only active bus cycles operating in the DRAM banks will prevent a refresh cycle. Refresh occurs in both banks simultaneously.

DRAM refresh is initiated during an idle state between bus cycles or during a bus cycle which does not access the DRAM. If refresh is required concurrent with a DRAM access, the MC68EN302 will perform the access while holding off the refresh. After the MC68EN302 initiates a refresh cycle, it must hold off DRAM accesses by inserting wait states until the $\overline{\text{RAS}}$ precharge is complete following the refresh cycle.

Figure 3-3 shows a refresh cycle. In this case, there is a DRAM access waiting on the bus and the DRAM access must wait until after $\overline{\text{RAS}}$ precharge.

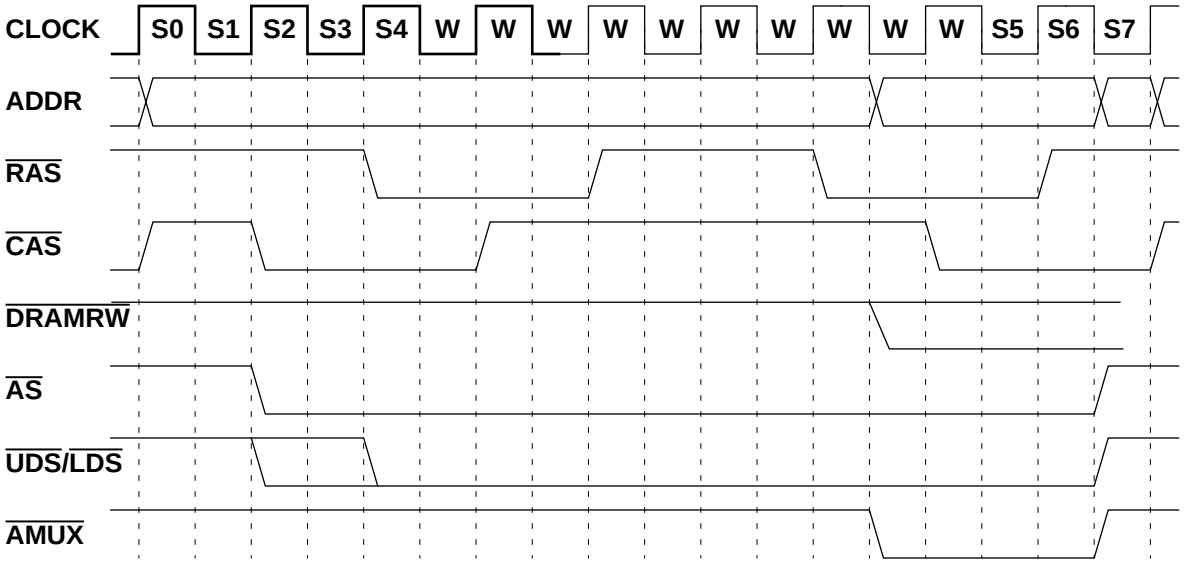


Figure 3-3. Precharge With DRAM Access Active

3.8 DRAM CONTROLLER I/O

3.8.1 Control Signal Pins

The EN302 contains 8 DRAM specific signal pins: $\overline{\text{CAS1}}$–$\overline{\text{CAS0}}$, $\overline{\text{RAS1}}$–$\overline{\text{RAS0}}$, $\overline{\text{AMUX}}$, and $\overline{\text{DRAMRW}}$.

3.8.2 Column Address Strobes ($\overline{\text{CAS1}}$–$\overline{\text{CAS0}}$)

These active low pins allow seamless interface to column address strobe ($\overline{\text{CAS}}$) inputs on industry standard DRAM, providing $\overline{\text{CAS}}$ for both bank 0 and bank 1 of the DRAM controller. Two strobes support byte operations on the external 16-bit bus. $\overline{\text{CAS0}}$ corresponds to data pins D15-D8. $\overline{\text{CAS1}}$ corresponds to data pins D7–D0.

3.8.3 Row Address Strobes ($\overline{\text{RAS1}}$–$\overline{\text{RAS0}}$)

These active low pins allow seamless interface to row address strobe ($\overline{\text{RAS}}$) inputs on industry standard DRAM, providing $\overline{\text{RAS}}$ for both bytes of a given DRAM bank. A particular bank corresponds to specific base address and control information programmed in the MC68EN302 DRAM control registers (see 3.2 Memory Map for a description). $\overline{\text{RAS0}}$ corresponds to bank 0 and $\overline{\text{RAS1}}$ corresponds to Bank 1.

3.8.4 DRAM Read/Write ($\overline{\text{DRAMRW}}$)

This active low pin is asserted to signify that a DRAM write cycle is occurring. It is separate from the processor bus R/W so that precharge takes place without regard to the state of R/$\overline{\text{W}}$.

3.8.5 Address Mux ($\overline{\text{AMUX}}$)

The $\overline{\text{AMUX}}$ pin is provided for implementing external address muxing circuitry so that external masters may access DRAM modules controlled by the MC68EN302 DRAM controller. External address muxing must take place in this situation since an access to the MC68EN302 as a slave always results in the addresses driven as an input, and does not output addresses to the DRAM module.

Another use for the $\overline{\text{AMUX}}$ pin would be implementations in which a linear DRAM space is required.

3.8.6 Parity (PARITY1–PARITY0)

These two pins are provided to support parity checking of DRAM. If enabled, parity is generated on writes and checked on reads. A parity error on a read generates a bus error. PARITY0 is used in connection with D15–D8 and PARITY1 is used in connection with D7–D0.

Parity checking/generation is not supported for external bus masters.

3.8.7 Muxing Scheme

To provide a simplified implementation of the Address Mux, a unique muxing scheme is provided. Rather than providing programmability to change which addresses are muxed on a particular signal, a generic muxing scheme is provided so that one muxing scheme may be utilized by all supported DRAM bank sizes. Table 3-3 shows the DRAM muxing scheme. The usage listed in the table is for typical operation. It is possible that some users may utilize the Base Address Registers and the Mask bits in a non-standard way.

Table 3-4. Address Muxing Scheme

PROCESSOR ADDRESS	ROW ADDRESS	COLUMN ADDRESS	USAGE
A9	9	1	Used for all Bank Sizes
A10	10	2	
A11	11	3	
A12	12	4	
A13	13	5	
A14	14	6	
A15	15	7	
A16	16	8	
A18	18	17	Used for 512K and up
A20	20	19	Used for 2M and up
A22	22	21	Used for 8M

SECTION 4
ETHERNET CONTROLLER

The MC68EN302, like the MC68360 QUICC, provides full duplex Ethernet along with multiprotocol support on the SCC channels. On the MC68EN302, the Ethernet controller is independent of the CPM and therefore the MC68EN302 provides the Ethernet channel in addition to the pre-existing 302 SCC channels without sacrificing any SCC pins. The Ethernet controller consists of an 802.3/Ethernet MAC layer protocol machine with an internal CAM, transmit and receive FIFOs, buffer descriptor memory and a dual channel DMA controller. The buffer descriptor memory, CAM and DMA controller all interface to the module bus. A block diagram of the Ethernet controller is shown in Figure 4-1.

Features are:

- 802.3 Ethernet compliant MAC layer (1-10 Mbps) with industry standard interface
- Full-duplex operation
- 128 byte transmit and receive FIFOs
- Collision retry does not generate extra bus bandwidth
- Collision fragments automatically discarded
- 64 entry CAM with optional hash mode
- Buffer descriptor memory on-chip

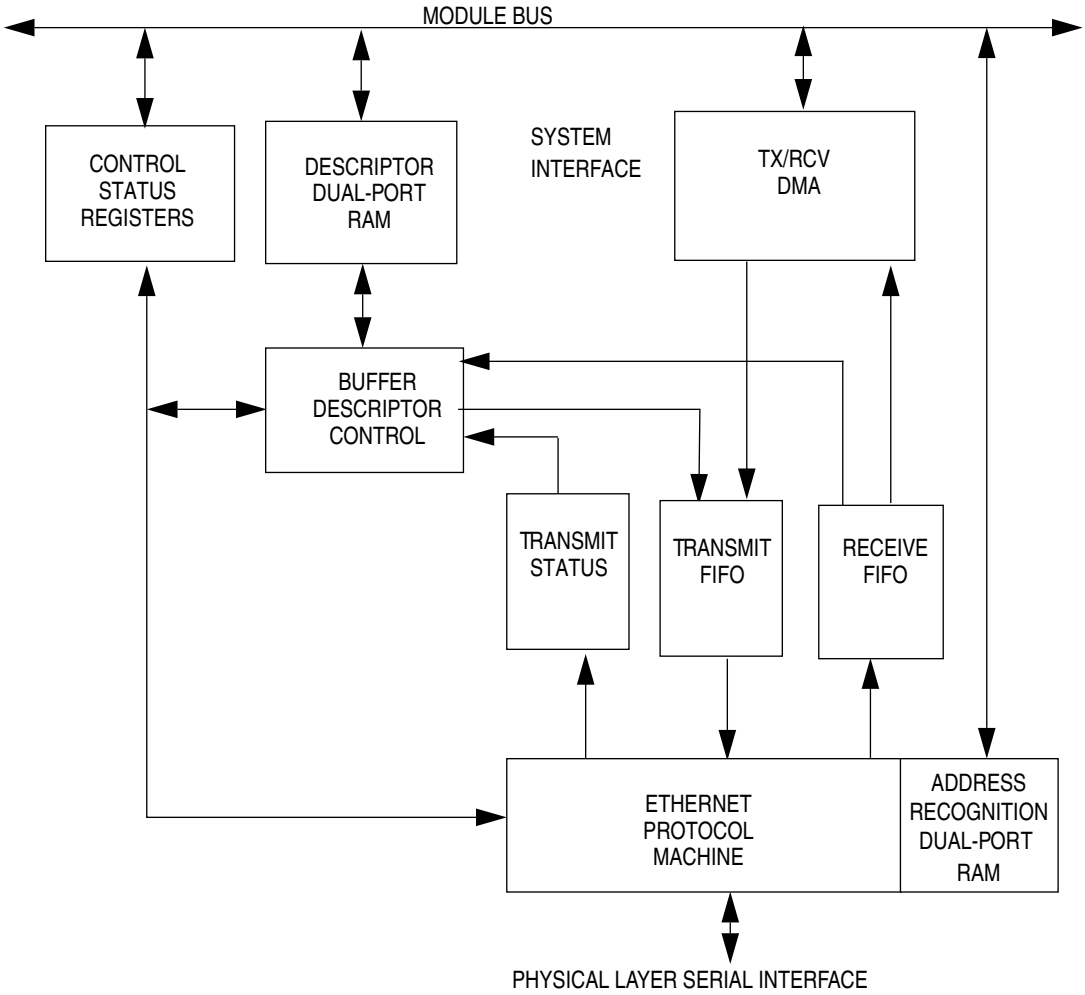


Figure 4-1. Ethernet Controller Block Diagram

4.1 REGISTER DESCRIPTION

Table 4-1 describes the Ethernet controller memory map.

Table 4-1. Ethernet Controller Memory Map

ADDRESS	NAME	MNEMONIC	TYPE
MOBA + 800	Ethernet Control Register	ECNTRL	Read/Write
MOBA + 802	Ethernet DMA Configuration Register	EDMA	Read/Write
MOBA + 804	Maximum Receive Buffer Length	EMRBLR	Read/Write
MOBA + 806	Interrupt Vector Register	INTR_VECT	Read/Write
MOBA + 808	Interrupt Event	INTR_EVENT	Read/Write
MOBA + 80A	Interrupt Mask Register	INTR_MASK	Read/Write
MOBA + 80C	Ethernet Configuration	ECNFIG	Read/Write
MOBA + 80E	Ethernet Test Register	ETHER_TEST	Read/Write
MOBA + 810	Address Recognition Control Register	AR_CNTRL	Read/Write
MOBA + (812-9FF)	Reserved		
MOBA + A00 MOBA + BFF	CAM Entry Table	CET	Read/Write
MOBA + C00 MOBA + FFF	Buffer Descriptors Table	EBD	Read/Write

Bits in the registers are R/W unless noted otherwise. Unimplemented bits will return 0 on reads. If reserved memory locations are accessed, DTACK is not returned.

4.1.1 ETHERNET CONTROL REGISTER (ECNTRL)

The ECNTRL register controls MC68EN302 Ethernet controller operation. All implemented bits in this register are R/W. This register is $0000 following system reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	GTS	ETHER_EN	RESET

15–3 Reserved. Should be written to zero by the host processor. These bits are always read as zero.

GTS—Graceful Transmit Stop.

0 = No change in Ethernet controller operation
1 = The Ethernet controller will stop transmission after all frames that are currently being transmitted have completed. See GRA in 4.1.5 Interrupt Event Register (INTR_EVENT).

ETHER_EN—Ethernet Enable.

0 = Reception is immediately stopped and transmission ends following the appending of a bad CRC to any frame currently being transmitted. Buffer descriptor(s) corresponding to an aborted transmit frame are not updated following ETHER_EN deassertion. In this situation, the DMA, buffer descriptor and FIFO control logic is reset along with the buffer descriptor and FIFO pointers.
1 = The Ethernet controller is enabled and reception and transmission of frames may occur.

RESET—Ethernet Controller Reset.

0 = A reset is performed locally within the Ethernet controller. ETHER_EN is cleared and all other Ethernet controller registers take their reset values. During Ethernet controller reset, the Buffer Descriptor Table and the CAM Entry Table can not be read or written. Any transmission/reception currently in progress is abruptly aborted.
1 = The MC68EN302 Ethernet controller operates normally

4.1.2 ETHERNET DMA CONFIGURATION STATUS REGISTER (EDMA)

The EDMA register allows user control of the DMA unit and may be written only when the ETHER_EN bit in the ECNTRL register is cleared. This register is cleared by a hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDERR<6:0>							0	BDSIZE<1:0>		TSRLY	WMRK<1:0>		BLIM<2:0>		

BDERR 6–0—Buffer descriptor error number.

This read only field is the buffer descriptor number that was being accessed when a bus error occurred. See 4.1.5 Interrupt Event Register (INTR_EVENT) for a description of the bus error handling.

Bit 8—Reserved. Should be written to zero by the host processor. This bit is always read as zero.

BDSIZE1-0—Buffer descriptor size. (R/W)

00 = 8 transmit buffer descriptors, 120 receive buffer descriptors
01 = 16 transmit buffer descriptors, 112 receive buffer descriptors
10 = 32 transmit buffer descriptors, 96 receive buffer descriptors
11 = 64 transmit buffer descriptors, 64 receive buffer descriptors

BDSIZE controls the allocation of the44 128 on-chip buffer descriptors between the transmit and receive operations. Typical implementations will set BDSIZE(1–0) to 01 allowing 16 transmit buffer descriptors and 112 receive descriptors.

TSRLY—Transmit start early. (R/W)

TSRLY controls when the transmission of a frame will begin. Typical applications will set TSRLY to 0.

0 = Frames do not begin transmitting until the transmit FIFO has only WMRK bytes available (empty), where WMRK ranges from 96 to 120 bytes.
1 = The frame will begin transmitting after the WMRK number of bytes have been written to the transmit FIFO where WMRK ranges from 8 to 32 bytes. This requires low bus latency to avoid transmit FIFO underrun.

TSRLY	WMRK<1:0>	BYTES IN TRANSMIT FIFO AT START OF TRANSMISSION
0	00	120
0	01	112
0	10	104
0	11	96
1	00	8
1	01	16
1	10	24
1	11	32

WMRK1–0—FIFO Watermark. (R/W)

00 = 8 FIFO bytes present or available
01 = 16 FIFO bytes present or available
10 = 24 FIFO bytes present or available
11 = 32 FIFO bytes present or available

The FIFO Watermark is used to control the start of a DMA burst. In the receive direction, the DMA state machine waits for either an end-of-frame (EOF) or a WMRK number of bytes to be in the receive FIFO prior to beginning a DMA burst of data out of the MC68EN302 to the host bus. In the transmit direction, the DMA state machine waits for WMRK number of bytes

of space to be available in the transmit FIFO prior to beginning a DMA burst into the MC68EN302 transmit FIFO. WMRK is typically set to 16.

WMRK1–WMRK0, when used in conjunction with BLIM2–BLIM0, allows the system designer to configure the MC68EN302 device for expected bus latency.

BLIM2–BLIM0—Burst Limit. (R/W)

BLIM2–BLIM0 controls the maximum length of a DMA burst in accesses from the bus interface unit. BLIM is typically set to 8 for 16 bit systems.

000 = 1 Access
001 = 2 Accesses
010 = 4 Accesses
011 = 8 Accesses
100 = 16 Accesses
101 = 32 Accesses
110 = 64 Accesses
111 = Unlimited

4.1.3 ETHERNET MAXIMUM RECEIVE BUFFER LENGTH (EMRBLR)

The EMRBLR register determines the maximum size of all receive buffers. Because the maximum frame is limited to 1518, only bits 10–0 are written by the user. The value written into the maximum receive buffer length register must account for the receive CRC which is always written into the last receive buffer. To allow one maximum size frame per buffer, EMRBLR must be set to 0000010111101110 or larger. The EMRBLR must be evenly divisible by 2. To ensure this, bit 0 is forced low. Only non-zero values are considered to be valid, therefore this register should be written after reset, but before Ethernet operation is enabled. All implemented bits are R/W. This register is cleared upon a hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	MAXIMUM RECEIVE BUFFER LENGTH										0

5–11—Reserved.

Should be written to zero by the host processor. These bits are always read as zero.

MRBL—Maximum receive buffer length.

Must be programmed to a non-zero value for operation.

0—Reserved.

Must be written as zero by the host processor. This bit is always read as zero.

4.1.4 INTERRUPT VECTOR REGISTER (IVEC)

The IVEC register controls the interrupt vector generated by the Ethernet controller during an interrupt acknowledge cycle. This register can only be written when the ETHER_EN bit in the ECNTRL register is cleared. This register is reset to $000F.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	INV<7:0>							

15–8—Reserved.

Should be written to zero by the host processor. These bits are always read as zero.

VG—Vector Granularity.

0 = The interrupt vector is not modified to reflect the cause of the interrupt.

1 = The interrupt vector is modified to indicate the cause of the interrupt, replacing the lower two bits of the interrupt vector according to the following table:

INTERRUPT VECTOR1–0	CAUSE	EXAMPLES
00	Receive Interrupt	RFINT, RXB
01	Transmit Interrupt	TFINT, TXB
10	Non-Time Critical Interrupt	HBERR, BABR, BABT, GRA, BOD, EBERR
11	Time Critical Interrupt	BSY

If multiple interrupt sources are present simultaneously and VG = 1, the INV bits will be set based on the following priority (highest ot lowest);

1. Time critical interrupt.
2. Receive interrupt
3. Transmit interrupt
4. Non-time critical interrupt.

For example, if both RXB and TFINT interrupts are asserted, INV1–INV0 will equal 00.

Interrupt Vector1–0 represent the values of the two lower bits placed on the data bus during an interrupt acknowledge cycle. VG is cleared by reset.

INV7–0—Interrupt Vector.

INV is the eight bit vector that the Ethernet controller places on the low byte of the data bus during an interrupt acknowledge cycle.

4.1.5 INTERRUPT EVENT REGISTER (INTR_EVENT)

When an event occurs that sets a bit in the Interrupt Event Register, and the corresponding bit in the interrupt mask register (INTR_MASK) is set, an interrupt will be generated. To clear

a bit in INTR_EVENT, a one must be written to that bit position. Writing a zero will not change the value of the bit. This register is cleared upon a hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	HBERR	BABR	BABT	GRA	BOD	EBERR	TFINT	RFINT	BSY	TXB	RXB

15–11—Reserved. Should be written to zero by the host processor.

HBERR—Heartbeat Error.

When HBC is set, a Heartbeat was not detected within the Heartbeat window following a transmission.

BABR—Babbling Receiver Error.

Indicates a frame longer than 1520 bytes was received. According to 802.3, frames should not exceed 1518 bytes but two bytes of slop is allowed. Receive frames exceeding 1520 bytes in length are truncated to prevent receive buffer overflow.

BABT—Babbling Transmitter Error.

The transmitted frame length has exceeded 1520 bytes. This condition is usually caused by a frame that is too long being placed into the transmit data buffer(s).

GRA—Graceful Stop Complete.

A graceful stop, initiated by the setting of GTS, is now complete. Once the frame that was in progress when GTS was set has transmitted, this bit is set. If the start of a second frame is in the FIFO, GRA will be set after the transmission of the second frame. GRA is also set after EBERR.

BOD—BackOff Done.

Indicates that the backoff timer has expired. This interrupt is used only for production testing and should normally be ignored. (Set BODEN = 0)

EBERR—Ethernet Bus Error occurred.

Indicates that a bus error occurred when the Ethernet controller was bus master. The BDERR bits in the EDMA register indicate which buffer descriptor was being used at the time of the bus error. Once any frames currently in the transmit FIFO have completed transmission and their status is written to the appropriate buffer descriptor, the GRA bit is set. If no frames or only a partial frame is in the transmit FIFO, the GRA bit is set immediately causing the partial frame to become an underrun truncated with a bad CRC.

TFINT—Transmit Frame Interrupt.

Indicates that a frame has been transmitted and that the last corresponding buffer descriptor has been updated.

RFINT—Receive Frame Interrupt.

Indicates that a frame has been received and that the last corresponding buffer descriptor has been updated.

BSY—Busy Interrupt.

Indicates that the start of a frame was detected but a receive buffer was not available in which to put the frame. If not corrected, this results in a FIFO overflow, which is indicated by the OV bit in the receive buffer descriptor.

TXB—Transmit Buffer Interrupt.

Indicates that a transmit buffer descriptor with the I bit set in its status word has been updated.

RXB—Receive Buffer Interrupt.

Indicates that a receive buffer descriptor with the I bit set in its status word has been updated.

4.1.6 INTERRUPT MASK REGISTER (INTR_MASK)

The Interrupt Mask register provides control over which Ethernet controller events generate an actual interrupt. The Interrupt Mask register is cleared on a hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	HBEEN	BREN	BTEN	GRAEN	BODEN	EBERREN	TFIEN	RFIEN	BSYEN	TBIEN	RBIEN

15–11—Reserved.

Must be written to zero by the host processor.

HBEEN—Heartbeat Error Interrupt Enable.

Enable interrupts when HBERR is set

BREN—Babbling Receiver Interrupt Enable.

Enable interrupts when BABR is set

BTEN—Babbling Transmitter Interrupt Enable.

Enable interrupts when BABT is set.

GRAEN—Graceful Stop Interrupt Enable.

Enable interrupts when GRA is set

BODEN—BackOff Done Enable.

Enable interrupts when BOD is set. This bit should always be cleared.

EBERREN—Ethernet Controller Bus Error Enable.

Enable interrupts when EBERR is set.

TFIEN—Transmit Frame Interrupt Enable.

Enable interrupts when TFINT is set.

RFIEN—Receive Frame Interrupt Enable.

Enable interrupts when RFINT is set.

BSYEN—Busy Interrupt Enable.
Enable interrupts when BSY is set.

TBIEN—Transmit Buffer Interrupt Enable.
Enable interrupts when TBINT is set.

RBIEN—Receive Buffer Interrupt Enable.
Enable interrupts when RBINT is set.

4.1.7 ETHERNET CONFIGURATION (ECNFIG)

The Ethernet Configuration register provides protocol configuration information to the Ethernet controller and can be written only when the ETHER_EN bit in the ECNTRL register is clear. This register is cleared on a hardware reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	RDT	HBC	FDEN	LOOP

15–4—Reserved.
Should be written to zero by the host processor. These bits are always read as zero.

RDT—Receive Disable on Transmit
- 0 = Receive path operates independently of transmit.
- 1 = Disable reception of frames while transmitting.

The purpose of this bit is to block reception of frames while this node is transmitting. If FDEN = 1, this bit must be 0.

If FDEN = 0, this bit should be 1, except when:

1. LOOP = 1 to select internal feedback.
2. An external loopback is being performed.
3. This node wants to receive its own transmit frames for monitoring purposes.

HBC—HeartBeat Control.
- 0 = The heartbeat check is not performed. Following end of transmission, the HB bit in the TxBD will be cleared.
- 1 = The heartbeat check is performed. Following end of transmission, if CLSN is not asserted during the heartbeat window, the HB bit in the TxBD will be set.

FDEN—Full Duplex Enable.
- 0 = Carrier Sense and Collision inputs are active, deference and collision handling is according to IEEE 802.3
- 1 = Frames are transmitted independent of Carrier Sense and Collision inputs.

LOOP—Loopback.

0 = No loopback operation is performed
1 = Transmitted frames are looped back internal to the device and TENA remains inactive (low).

4.1.8 ETHERNET TEST (ETHER_TEST)

The Ethernet Test register controls various manufacturing test modes. Test modes may be useful to some users, but in general it is not suggested that the user set these modes in normal operation. This register can only be written when the ETHER_EN bit in the ECNTRL register is cleared and it is cleared on a reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	RNGT	TBO	TRND	SLOT	COLL	DRTY	RWS	TWS

15–8—Reserved.

Must be written to zero by the host processor. These bits are always read as zero.

RNGT—Random number generator test.

This bit allows testing of the random number generation logic used in the transmit backoff process. To run the random number generator test, write $0080 to this register and then poll repeatedly. This bit will be changed by hardware from 1 to 0 within 15 μsec if the random number generator is operating properly.

TBO—Test Backoff.

0 = Normal operation of the random number generator logic.
1 = The random number generated by the transmit backoff logic is all ones.

TRND—Transmit random number control.

0 = An internal oscillator clocks the random number generator (normal mode).
1 = The transmit clock is used to generate the Ethernet random number.

SLOT—Slot Time Length.

Specifies the number of bytes in a slot time used in backoff determination.

0 = The number of bytes in a slot time is 64.
1 = The number of bytes in a slot time is 2.

COLL—Force Collision.

COLL allows the collision logic to be tested. The device must be in the internal loopback mode for COLL to be valid.

0 = No collision is forced.
1 = A collision is forced during the subsequent transmission attempt. This results in 16 total transmission attempts with a retry error reported in the transmit descriptor.

DRTY—Disable Retry.

0 = The MC68EN302 performs retry error reporting normally.
1 = The MC68EN302 does not perform any retries of a frame before reporting a retry error in the transmit descriptor for the frame.

RWS—Receive Watchdog Timer Short.

0 = The Receive Watchdog Timer operates normally
1= The Receive Watchdog Timer is short cycled for test purpose causing the Receive Watchdog Timer to expire, and a BABR interrupt to be generated if more than 100 bytes in a frame are received.

TWS—Transmit Watchdog Timer Short.

0 = The Transmit Watchdog Timer operates normally.
1 = The Transmit Watchdog Timer is short cycled for test purposes causing the Transmit Watchdog Timer to expire and a BABT interrupt to be generated if more than 100 bytes in a frame are transmitted.

4.1.9 AR CONTROL REGISTER (AR_CNTRL)

The AR Control register controls AR memory operation and can be written only when the ETHER_EN bit in the Control register is clear. Bits in this register are not changed if the ETHER_EN bit in the Control register is set. This register is set to the value $0000 on a reset.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HASH_EN	INDEX_EN	MULT1–MULT0		PA_REJ	PROM	0	0	0	0	0	0	0	0	0	0

HASH_EN—Hash mode enable.

0 = Perfect-entry match mode is selected for all entries
1= Hash mode is enabled instead of perfect matches on entries 62 and 63.

INDEX_EN—Index enable.

0 = The receive buffer data pointer is unmodified
1= Pass either line number or hash index into the upper byte of the receive buffer descriptor’s data pointer.

MULT1–MULT0—Multicast and broadcast reception control.

MULT controls whether multicast frames are received. Broadcast is the special multicast address of all ones.

MULT1–MULT0]	ACCEPT MULTICAST?	ACCEPT BROADCAST?
00	If match in tables	Yes
01	If match in tables	No
10	Yes, all	Yes
11	No, none	No

PA_REJ—Physical address reject.

0 = Frames with physical addresses are accepted if there is a table match, either perfect or hash.
1 = Frames with physical addresses are accepted if there is no perfect match. If a physical address has a hash match but not a perfect match, the frame will be accepted. This bit has no effect on frames with a multicast address.

PROM—Promiscuous mode.

0 = Frames are accepted only if they meet the hashing, perfect address match, or MULT1–MULT0 criteria
1 = All frames are accepted regardless of address matching or settings of MULT1–MULT0.

9–0—Reserved.

Should be written as zero by the host processor. These bits are always read as zero.

4.2 ETHERNET BUFFER DESCRIPTORS

The data for the Ethernet frames must reside in memory external to the MC68EN302 device and is placed in one or more buffers. Buffer descriptors contain pointers to each buffer and contain the current state of the buffer. The BDs are located inside the MC68EN302 in the dual port Buffer Descriptor RAM so that the load on the processor bus is minimized.

Software “produces” buffers by allocating/initializing memory and initializing buffer descriptors in the BDRAM. Setting the most significant bit (R for transmit and E for receive) in the most significant word of the buffer descriptor initializes the buffer. MC68EN302 DMA hardware constantly polls the BDs and processes the buffers after they have been initialized. Processing in the case of transmit indicates that the data in the buffers has been read into the MC68EN302 and transmitted out the Ethernet interface. Processing in the case of receive indicates that data received from the Ethernet interface has been placed into data buffers pointed to by the receive buffer descriptors. Once DMA is complete and the buffer descriptor status bits have been written, the most signficant bit of the buffer descriptor is cleared indicating that the buffer has been processed. Software may either poll the BDs or may rely on the buffer/frame interrupts to detect when the buffers have been consumed.

The ETHER_EN signal operates as a reset to the BD/DMA logic. When ETHER_EN is deasserted, the BD pointers are reset to point to the starting transmit and receive BDs. The buffer descriptors are not initialized by hardware during reset. For proper operation, before setting the ETHER_EN bit, initialize at least one transmit and receive buffer descriptor by setting the most significant word of the descriptor to $0000 (this does not result in any transmit or receive operation, but is considered to be initialization). Because the DMA polls buffer descriptor memory to determine if the R/E bits in the next available BD are set whenever ETHER_EN=1, initializing ‘n’ buffers, requires software to initialize n+1 buffer descriptors, setting the most significant bit of the $(n+1)^{th}$ descriptor to 0.

The BDSIZE field in the EDMA register allows the user to define up to sixty-four buffers for the transmit channel and up to one hundred twenty buffers for the receive channel. The total number of combined transmit and receive buffers is one-hundred-twenty-eight. Each BD table, transmit and receive, forms a circular queue with separate transmit Buffer Descriptor and receive Buffer Descriptor pointers maintained in the hardware. The length of the circular queues may also be controlled by using the W (wrap) bit in the buffer descriptors.

If the transmit FIFO empties of data before the end of the frame, an underrun occurs and a bad CRC is appended to the partially transmitted data. In addition, the UN bit is set in the last BD of the affected frame. Transmit underrun may occur if the Ethernet controller can not access the 68000 bus or if the next BD in the frame is not available.

During the receive process, if data from a frame is available but no BD is available, the BSY interrupt is generated, warning the user that data will soon be lost if a BD does not become available. If the receive FIFO overruns because there is no available BD or the Ethernet controller can not access the 68000 bus, then the last BD for the receive frame will have the OV bit set.

4.2.1 ETHERNET RECEIVE BUFFER DESCRIPTOR (RX BD)

The user initializes the E, W, I, and (optionally) RO bits in the first word and the pointer in 3rd and 4th words of the receive buffer descriptor. The Ethernet controller writes the following status bits:

- First word: E, L, F, M, LG, NO, SH, CR, OV and CL bits. The M, LG, NO, SH, CR, OV and CL bits in the first word of the buffer descriptor are only modified by the Ethernet controller when the L bit is set
- Second word: the buffer length
- Third word: the Reason and ARIndex fields if the INDEX_EN bit in the AR_CNTRL register is set.

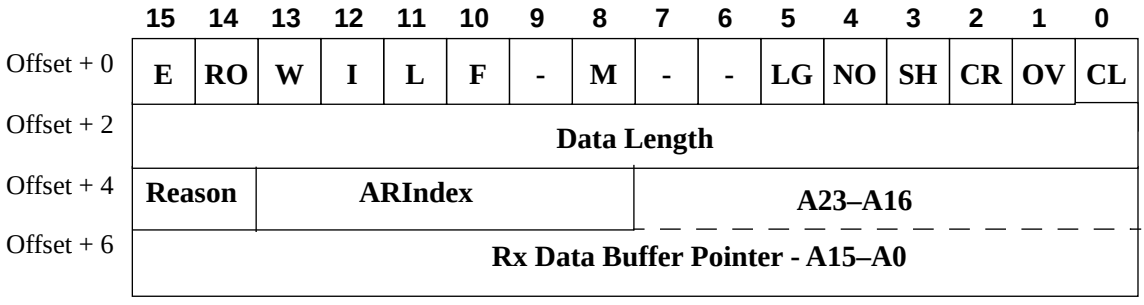


Figure 4-2. Ethernet Receive Buffer Descriptor (Rx BD)

The first word of the receive buffer descriptor contains status and control information concerning buffer descriptor handling and data flow. These status and control bits are described in the following paragraphs.

E—Empty, written by Ethernet controller (=0) and user (=1).

0 = The data buffer associated with this BD has been filled with received data. A receive buffer descriptor also has the E bit set to 0 when data reception has been aborted due to an error condition. When E is set to 0, the status, length, reason and AR Index fields are updated according to the event that just occured during reception.

1 = The data buffer associated with this BD is empty, or reception is currently in progress.

RO—Receive Buffer Software Ownership, written by user.

This bit is provided as a software ownership bit, if needed. Hardware does not alter the value of this bit.

W—Wrap (Final BD in Table), written by user.

0 = This is not the last buffer descriptor in the Rx BD table.

1 = This is the last buffer descriptor in the Rx BD table. After this buffer has been used, the Ethernet controller receives incoming data into the first BD in the table, allowing the user to use fewer buffer descriptors than the number programmed by BDSIZE.

I—Interrupt, written by user.

0 = No interrupt is generated after this buffer has been filled.

1 = The RBINT bit in the interrupt event register is set when this buffer has been completely filled by the Ethernet controller, indicating that the internal 68000 core is free to process the buffer.

L—Last in Frame, written by Ethernet controller.

0 = The buffer is not the last in a frame.

1 = The buffer is the last in a frame.

F—First in Frame, written by Ethernet controller.

0 = The buffer is not the first in a frame.

1 = The buffer is the first in a frame.

M—Miss, written by Ethernet controller.

This bit is set by the Ethernet controller when the incoming frame is not matched by the internal address recognition but is accepted because the Ethernet controller is operating in promiscuous mode (PROM=1). The user can monitor the M-bit to quickly determine whether the incoming frame was destined to this station. This bit is valid only if the L-bit is set. The M-bit is valid even if INDEX_EN is not set.

0 = An address match in the CAM or hash algorithm caused frame reception.

1 = No address match occured on the received frame - promiscuous mode operation caused frame reception.

LG—Rx Frame Length Violation, written by Ethernet controller.

A frame length greater than 1520 (maximum allowed receive frame length) was recognized. In this situation, note that only the first 1520 bytes are written to the data buffer. This bit is valid only if the L-bit is set. This frame should be discarded.

NO—Rx Nonoctet Aligned Frame, written by Ethernet controller.

The received frame contained a number of bits which is not a multiple of 8, and the CRC check that occurred at the preceding byte boundary generated an error. This bit is valid only if the L-bit is set. If this bit is set, the CR bit will not be set. This frame should be discarded.

SH—Short Frame, written by Ethernet controller.

The MC68EN302 does not support SH and this bit is always cleared. This bit indicates that a frame length less than the minimum defined for this channel was recognized. This frame should be discarded.

CR—Rx CRC Error, written by Ethernet controller.

This frame contains a CRC error and is an integral number of octets in length. This bit is valid only if the L-bit is set. This frame should be discarded.

OV—Overrun, written by Ethernet controller.

A receive FIFO overrun occurred during frame reception. During a FIFO overflow, the status bits also in this word (M, LG, NO, SH, CR, and CL) lose their normal meaning and are zero. This bit is valid only if the L-bit is set. This frame should be discarded.

CL—Collision, written by Ethernet controller.

A collision occurred during frame reception and the frame was closed. This bit is set only if a late collision occurred. This bit is valid only if the L-bit is set. This frame should be discarded.

Data Length, written by Ethernet controller.

Data length indicates the number of octets written by the Ethernet controller into this BD’s data buffer. It is written by the Ethernet controller upon the close of this BD.

Reason and ARIndex, written by Ethernet controller.

If INDEX_EN=1 in the AR_CNTRL register, then the Reason and ARIndex fields replace the most significant byte of the Rx Buffer Pointer. The Reason and ARIndex are available on all buffer descriptors for a frame when INDEX_EN is set, independent of the condition of the L and F bits. When INDEX_EN = 0 the Reason and ARIndex fields are not modified by hardware.

Rx Buffer Pointer, written by user.

The receive buffer pointer always points to the first location of the associated data buffer and must be a multiple of 2. The data buffer must reside in memory external to the Ethernet controller. When INDEX_EN=1, the most significant byte of the receive buffer pointer is replaced by a reason and index field. When INDEX_EN=0, the receive buffer pointer is not modified. See 4.6.1 Buffer Descriptor Modification for more details.

4.2.2 ETHERNET TRANSMIT BUFFER DESCRIPTOR

Transmit data is presented to the Ethernet controller through buffers referenced by transmit Buffer Descriptors. The Ethernet controller confirms transmission operation through the R bit, and indicates error conditions through the other status bits in the most signficant word of the BD. The host software must initialize the R, W, I, L, TC, and (optionally) TO bits in the first word, the length in the second word, and the buffer pointer in the third and fourth words.

.

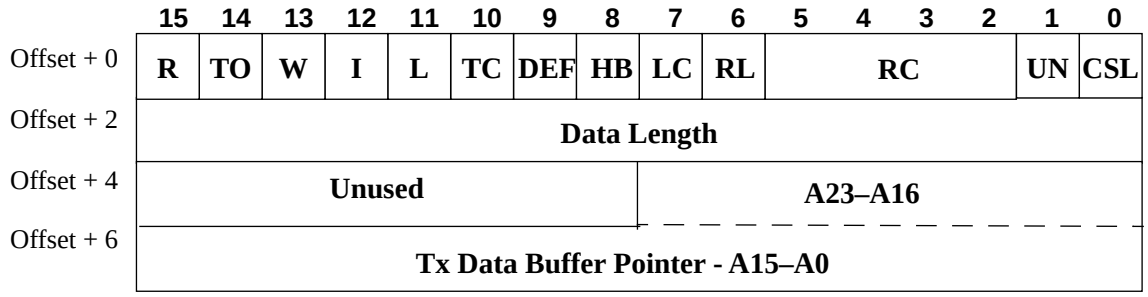


Figure 4-3. Ethernet Transmit Buffer Descriptor (Tx BD)

The Tx BD fields are detailed below. The unused bits (15-8) in Offset + 4 are not used by the hardware. These unused bits are R/W by software and are ignored by hardware.

R—Ready, written by Ethernet controller and user.

0 = The data buffer associated with this BD is not ready for transmission, leaving the software free to manipulate this BD or its associated data buffer. The Ethernet controller clears this bit after the buffer has been transmitted or after an error condition is encountered.

1 = The data buffer, which has been prepared for transmission by the user, has not been transmitted or is currently being transmitted. No fields of this BD may be written after this bit is set.

TO—Transmit Buffer Software Ownership, written by user.

This bit is provided as a software ownership bit, if needed. Hardware does not alter the value of this bit.

L— Last (Last BD for this frame)

0 = This is not the last BD for this frame and the Ethernet controller sets R= 0 when the buffer has been DMA'd into the MC68EN302. Status bits are not modified.

1 = The Ethernet controller sets R = 0 and modifies the DEF, HB, LC, RL, RC, UN and CSL status bits once the buffer has been DMA'd into the MC68EN302 and frame transmission has completed

W—Wrap (Final BD in Table), written by user.

0 = This is not the last buffer descriptor in the Tx BD table.
1 = This is the last buffer descriptor in the Tx BD table. After this buffer has been used, the Ethernet controller transmits data from the first BD in the table. The maximum number of Tx BDs in this table is programmable through the BDSIZE bits.

I—Interrupt, written by user.

0 = No interrupt is generated after this buffer has been serviced.
1 = TBINT is set in the Ethernet event register after this buffer has been serviced. If enabled through the mask register, this bit can cause interrupts to the host.

L—Last in Frame, written by user.

0 = The buffer is not the last in the transmit frame.
1 = The buffer is the last in the transmit frame.

TC—Tx CRC, written by user (only valid if L = 1).

0 = End transmission immediately after the last data byte.
1 = Transmit the CRC sequence after the last data byte.

DEF—Defer Indication, written by Ethernet controller (only valid if L = 1).

The Ethernet controller had to defer while attempting frame transmission. DEF is not set if a collision occurs during transmission.

HB—Heartbeat Error, written by Ethernet controller (only valid if L = 1).

The collision input was not asserted within the heartbeat window after transmit completion. HB may be set only if HBC is not set in the ECNFIG register.

LC—Late Collision, written by Ethernet controller (only valid if L = 1).

A collision has occurred after 56 data bytes have been transmitted. The Ethernet controller terminates the transmission.

RL—Retransmission Limit, written by Ethernet controller (only valid if L = 1).

The transmitter has failed (Retry Limit + 1) attempts to successfully transmit a message due to repeated collisions on the medium.

RC—Retry Count, written by Ethernet controller (only valid if L = 1).

These four bits indicate the number of retries required before this frame was successfully transmitted. If RC = 0, then the frame was transmitted correctly the first time. If RC = 15, then the frame was transmitted successfully while the retry count was at its maximum value. If RL is set, then RC has no meaning.

UN—Underrun, written by Ethernet controller (only valid if L = 1).

A transmit FIFO underrun occurred while transmitting one or more of the the data buffers associated with this frame. When FIFO underrun occurs, frame transmission halts once an incorrect CRC is appended. The remaining buffer(s) associated with this frame are DMA’d and dumped by the transmit logic.

CSL—Carrier Sense Lost, written by Ethernet controller (only valid if L = 1).
Carrier sense dropped out or was never asserted during a collision free frame transmission.

Data Length, written by user.

Data length is the number of octets the Ethernet controller should transmit from this BD's data buffer. It is never modified by the Ethernet controller. The value of this field must be greater than zero.

Tx Buffer Pointer, written by user.

The transmit buffer pointer containing the address of the associated data buffer, may be even or odd. The buffer must reside in memory external to the MC68EN302. This value is never modified by the Ethernet controller.

4.3 DMA AND BUFFER DESCRIPTOR LOGIC

The DMA and buffer descriptor modules transfer data between external memory and the TX/RX FIFOs.

4.3.1 BUFFER DESCRIPTOR LOGIC

Buffer descriptors are stored in the on-chip dual-port RAM. The RAM is sufficient to store 128 buffer descriptors of 4 sixteen-bit-words. The features of the BD circuitry are as follows:

- Flexible Buffer Descriptor allocation between transmit and receive;
- Multiple buffers per frame
- Transmit buffers may start on any byte boundary, Receive buffers must start on even byte boundaries.
- Maximum Receive Buffer size is user controllable;

The Buffer Descriptor space is divided between transmit and receive in various configurations depending on the value of BDSIZE in the EDMA register. Table 4-2 shows the starting and ending addresses (offset from MOBA) in the BD RAM for the four options.

Table 4-2. BD RAM Address Ranges

BDSIZE	TRANSMIT BUFFER DESCRIPTOR RANGE	RECEIVE BUFFER DESCRIPTOR RANGE	NUMBER OF TRANSMIT BUFFERS	NUMBER OF RECEIVE BUFFERS
$00	$C00 - $C3F	$C40 - $FFF	8	120
$01	$C00 - $C7F	$C80 - $FFF	16	112
$10	$C00 - $CFF	$D00 - $FFF	32	96
$11	$C00 - $DFF	$E00 - $FFF	64	64

The Maximum Receive Buffer Length field (MRBL) in the EMRBLR register determines the default length of all receive buffers besides the last buffer of a frame (the last buffer is usually shorter in length than the preceding buffers).

On the transmit side, the MC68EN302 may have up to two separate frames with open buffers at a specific point in time. While the first frame completes the transmit process, DMA

operaton begins for the second frame as the frame status of the first frame is determined. Frame status is not available until after the 4 microsecond heartbeat window at the end of transmission.

When ETHER_EN changes from 0 to 1, the transmit process starts at Buffer Descriptor+ $C00 and the receiver begins processing BDs at offset $C40, $C80, $D00, or $E00 depending on BDSIZE. When GTS (graceful transmit stop) is set, the transmitter halts once any unfinished transmit frames have completed transmission and the Buffer Descriptors have been updated. The transmitter then generates a GRA interrupt. When GTS is cleared (0), the transmitter begins transmission with the next frame in the transmit queue.

4.3.2 DMA LOGIC

The DMA block transfers data between the FIFOs and the data buffers that Buffer Descriptors point to. The DMA block must arbitrate for access to the module bus. Once the module bus controller has received a grant for the 68000 bus, a module bus grant is passed to the DMA controller. The DMA controller alternates between transmit and receive DMA, passing over either one if there is no outstanding data.

If there is receive data but no available buffer to place the data in, a BSY (BUSY) interrupt is generated.

Once a data flow direction is chosen between transmit and receive, the DMA machine continues to read a word or byte until it releases the bus and returns to the idle state. The DMA machine will release the bus on any of the following conditions:

- The burst limit counter is reached.
- An end of frame (EOF) has been reached.
- The receive FIFO has been emptied.
- The transmit FIFO has been filled.
- At the end of a buffer.

At the beginning or end of a buffer, a byte access occurs if necessary. Once the DMA machine releases bus mastership, if additional data must be moved, the DMA machine generates another bus request.

4.4 TRANSMIT AND RECEIVE FIFOS

The Ethernet controller contains separate 128-byte transmit and receive FIFOs organized as 64 locations x 18 bits each with 16 bits for data and 2 for tag information. Each FIFO has independent control logic allowing full duplex operation.

4.4.1 TRANSMIT FIFO

The transmit FIFO control logic provides flow control information to the transmit buffer descriptor logic. The timing for new transmit DMAs depends upon the WMRK and TSRLY bits in the EDMA register as well as the number of locations currently available in the FIFO.

The transmit FIFO control logic provides a signal indicating data is available to the Ethernet transmit protocol machine. If underflow occurs, the Ethernet transmit protocol machine will handle aborting the frame (append a bad CRC) and flushing the remainder of the frame from the FIFO.

If a collision occurs within the slot time in a transmit frame, the FIFO supports retry by maintaining a separate start of frame pointer (read lag pointer). New data is never written on top of start of frame data until the slot time has passed. Two control signals pass between the Ethernet transmit logic and the transmit FIFO to indicate when the slot time (collision window) has been passed (Transmit Accept) or when a collision retry must take place (Transmit Retry).

4.4.2 RECEIVE FIFO

The receive FIFO control logic provides “data available” and “receive FIFO empty” flow control signals to the receive DMA controller. The “data available” signal is asserted as a function of the number of bytes available in the FIFO and the WMRK bits from the EDMA register. If overflow occurs, the STATUS word will have the OU bit set which will be written into the receive BD. The frame should be discarded by software.

Data is written into the receive FIFO by the Ethernet receive logic in the case of status information, and by the address recognition logic if the Reason and ARIndex fields are enabled.

The receive FIFO control logic maintains a “start of frame” pointer that allows purging collision fragments from the FIFO so that they need not be DMA’d. This purging of fragments (runt frames less than 64 bytes long) is automatic and cannot be disabled.

4.5 ETHERNET PROTOCOL LOGIC

This block implements the MAC (media access control) sublayer of the IEEE 802.3 standard, supporting operation up to 10 Mbps compliant with both Ethernet and 802.3 standards. This logic is subdivided into transmit, receive and loopback/serial interface sections.

4.5.1 ETHERNET TRANSMIT

The Ethernet transmiter block performs the following functions:

- Parallel to serial conversion of data
- Encapsulation of transmit frames
  - — Generation of preamble (PA) and start of frame delimiter (SFD)
  - — Transmits serial data from the transmit FIFO interface
  - — Pads short frames (with 0’s)
  - — Appends CRC, if required
  - — Appends bad CRC if required
  - — Appends JAM pattern (all 1’s)
- Transmit Protocol

- — Guarantees minimum inter-frame gap (IFG) of 9.6 μsec between CarrierSense deasserted and next frame transmitted.
- — Provides 8 byte PA + SFD
- — Appends 32 bit JAM sequence (all 1's) and start backoff timer upon collision
- — Appends 32 bit CRC (if TC = 1) or bad CRC if aborting frame
- — Defers to CarrierSense for 6 μsec, then ignores CarrierSense for 3.6 μsec during InterFrameGap
- — Collision Retry occurs under the 802.3 truncated binary exponential backoff algorithm
- — Detects a babbling transmission and generates BABT interrupt
- — Aborts Frame transmission if Transmit FIFO underflow, ETHER_EN deassertion during frame transmission, Collision Retry Limit exceeded, Late Collision or Collision and DRTY = 1

- Provides Transmit Frame Status
  - — Generates the DEF, HB, LC, RL, RC, UN and CSL status fields written into the end of frame transmit buffer descriptor which provide status on the transmission of the frame. The definition of these fields is based on the Layer Management section of the 802.3 standard. These fields are valid after the heartbeat window following the successful transmission of a frame or if the collision retry limit (16 attempts) is exceeded. A "XmitStatusReady" signal is asserted to the transmit buffer descriptor control logic when this status is available

All logic in the Ethernet Transmit block runs synchronously with the Ethernet TCLK provided by an external Ethernet physical layer component(s).

NOTE

Deasserting ETHER_EN during frame transmission is NOT recommended as ETHER_EN is used as a reset signal in the Ethernet controller logic. The recommended procedure is to assert the GTS bit to gracefully halt transmission. Once the GRA interrupt is received indicating that transmission has completed, then deassert ETHER_EN.

4.5.2 ETHERNET RECEIVE

The receive block consists of the following submodules:

- Serial to Parallel Conversion
- Receive Protocol Control
  - — Controls data path by stripping PA, SFD, and dribble bits
  - — Detects runt frames, and signals REJECT to the receive FIFO
  - — Detects giant frames, generates the BABR interrupt and discards the rest of frame
  - — Provides count to determine frame length (in bytes)
  - — Provides for interframe recovery if a minimum receive interframe gap of approximately 2.4 μsec is provided.

- Receive Frame Status
  - — Generates the M, LG, NO, SH, CR, OV and CL status fields which are written into the end of frame receive buffer descriptor to provide status on the reception of the frame. The definition of these fields is based on the Layer Management section of the 802.3 standard.

The serial interface consists of TCLK, TENA, TX, RCLK, RENA, RX and CLSN. The polarity of the TENA, TX, RENA, RX and CLSN signals is positive (1 or asserted = Voh or Vih). Zero or more RCLK cycles are required following the deassertion of RENA at the end of a receive frame. Logic in this module will detect end of receive frame condition and switch in TCLK if necessary to complete flushing the frame through the receive data path and into the receive FIFO.

4.5.3 ETHERNET LOOPBACK

The transmit to receive loopback function is selected by the LOOP bit in the ECNFIG register. While in the internal loopback mode, TENA will not assert. Any assertion of RENA and CLSN will be ignored.

4.6 ETHERNET AR (ADDRESS RECOGNITION)

The MC68EN302 supports 64-entry internal address recognition with 48 bit address matching for receive address filtering. Address Recognition memory is written as a normal memory cycle. Note that unused entries in the AR memory map do not return $\overline{\text{DTACK}}$ if accessed.

There are two modes for address recognition: perfect entries, and hash mode. The mode selected determines the way in which memory is partitioned. When perfect-entry mode is selected, the entire memory is devoted to storing addresses for 64 perfect matches. When hash mode is selected, 8 bytes are used to store a logical address filter, and 372 bytes are used to store addresses for 62 perfect matches.

In hash mode, a logical address filter mask is used which requires the processor to perform final filtering. As the incoming data stream goes through the CRC Generator, once the 48th bit of the destination address has passed this circuitry, the six most significant bits of the CRC are sampled. Those 6 bits become an address which selects one of the 64 bits in the logical address filter mask. If the mask bit selected is a “1”, the address matches and the packet is accepted. When programming the hash table, the task of mapping a destination address to one of 64 bit positions requires a computer program to generate the CRC codes for the addresses desired. The 6 most significant bits of a given addresses’ CRC becomes the pointer into that addresses’ hash table entry. For Ethernet, the CRC polynomial is CRC32 or:

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

There is no rule on what type of address can be used in which type of address matching mode. Either physical or multicast addresses may be stored as either a perfect match or hash table entries. If an address matches both a perfect entry and a hash entry, the perfect entry takes precedence.

Broadcast and multicast frames may be either unconditionally accepted or unconditionally rejected. Note that multicast frames may be conditionally accepted based on a matching table entry, either perfect or hash. Refer to Figure 4-4 and Table 4-3 for broadcast and multicast frame address recognition.

Physical addresses are compared to perfect-match entries and either accepted or rejected. If no perfect match occurs, the addresses may then be accepted (but not rejected) on the basis of a hash match. Refer to Figure 4-4 and Table 4-4 for physical address recognition.

A physical address, but not a multicast address, may be rejected on the basis of a perfect match. A hash table match alone on a physical address is not sufficient to reject that frame.

The address recognition memory is not initialized at reset; the user must initialize it before setting ETHER_EN in ECNTRL. Perfect-match entries which are unused should be set to all 1’s (broadcast address) which is a safe value since broadcast frame handling occurs independently of any table entries. Unused hashmode entry bits must be set to 0.

4.6.1 BUFFER DESCRIPTOR MODIFICATION

Index values and reason fields may be passed into the upper byte of the receive buffer descriptor’s data pointer by setting the INDEX_EN bit. This may help improve software efficiency. If INDEX_EN is cleared, the receive buffer descriptor data pointer is not modified by the MC68EN302. If INDEX_EN is set, the index value and reason field will be written into all the BDs of a frame.

There are four reasons, not counting promiscuous mode, which cause a frame to be accepted:

1. The frame is a multicast or broadcast frame and the appropriate MULT bits are set.
2. A perfect address match occurs.
3. A hash match occurs.
4. No perfect match occurs, the frame has a physical, not multicast address, and the PA_REJ bit is set.

Table 4-3. Unicast Address Processing

CONTROL		OPERATION				
PROM	PA_REJ	AR RESULT	ACCEPT/REJ	MISS	AR REASON	INDEX
0	0	Perfect match	A	0	10	Line #
		Hash match	A	0	11	Hash index
		No match	R	–	–	–
0	1	Perfect match	R	–	–	–
		Hash match	A	0	11	Hash index
		No match	A	0	00	Hash index
1	0	Perfect match	A	0	10	Line #
		Hash match	A	0	11	Hash index
		No match	A	1	00	Hash index
1	1	Perfect match	A	1	00	Hash index
		Hash match	A	0	11	Hash index
		No match	A	0	00	Hash index

Table 4-4. Broadcast and Multicast Address Processing

CONTROL		BROADCAST				MULTICAST				
PROM	MULT1–MULT0	ACCEPT/REJECT	MISS	AR REASON	INDEX	AR RESULT	ACCEPT/REJECT	MISS	AR REASON	INDEX
0	00	A	0	01	0x3F	Perfect match	A	0	10	Line #
						Hash match	A	0	11	Hash index
						No match	R	–	–	–
0	01	R	–	–	–	Same as above entries				
0	10	A	0	01	0x3F	Perfect match	A	0	10	Line #
						Hash match	A	0	11	Hash index
						No match	A	0	01	0x00
0	11	R	–	–	–	–	R	–	–	–
1	00	A	0	01	0x3F	Perfect match	A	0	10	
						Hash match	A	0	11	
						No match	A	1	00	
1	01	A	1	00	0x2F	Same as above entries				
1	10	A	0	01	0x3F	Perfect match	A	0	10	Line #
						Hash match	A	0	11	Hash index
						No match	A	0	01	0x00
1	11	A	1	00	0x2F		A	1	00	Hash index

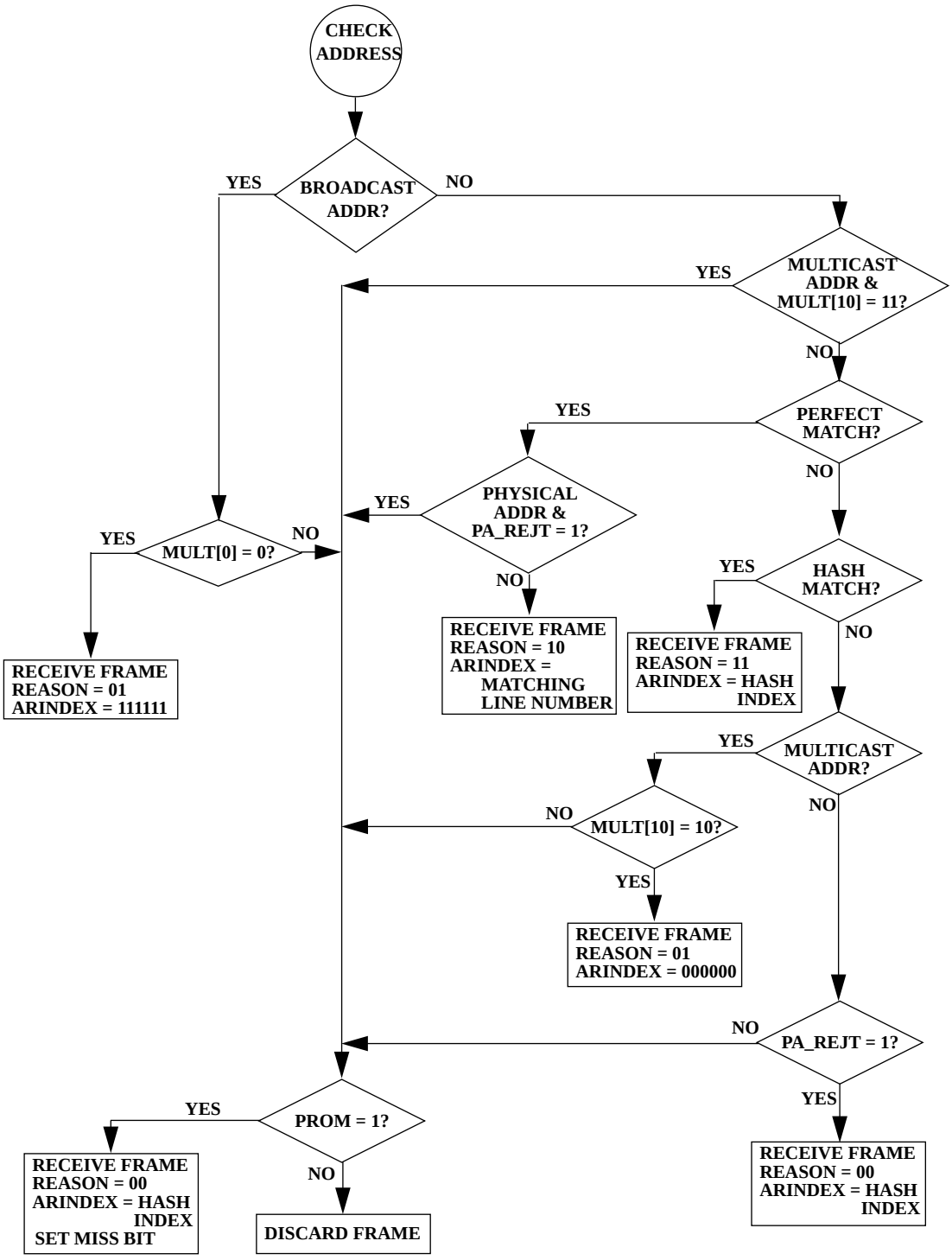


Figure 4-4. Ethernet Address Recognition Flowchart

4.6.2 WRITING ADDRESSES INTO TABLES

The address recognition block is written/read just like a normal memory cycle (word or byte). Unused locations do not return $\overline{\text{DTACK}}$ if accessed.

Because each entry in the perfect-match table is 48 bits, but no more than 16 bits can be written at a time; byte 5(or the word consisting of bytes 4 and 5 of a perfect-match entry) must be written last. This prevents an address compare from occurring on partially-written entries. When the first byte (or word) of an entry is written, that entry is temporarily disabled until byte 5 (or a word consisting of bytes 4 and 5) is written.

The Address Recognition memory map for perfect match mode is shown in Figure 4-5. The least significant bit of byte 0 (bit 8 of word MOBA + $A00, +$A08,...) corresponds to the I/G address bit - this is the first bit received off the wire. The order of the bits received starts with Byte 0-bit0 and continues through Byte0-bit7. The next byte is received as Byte1-bit0, Byte1-bit1,...Byte1-bit7 up through byte 5.

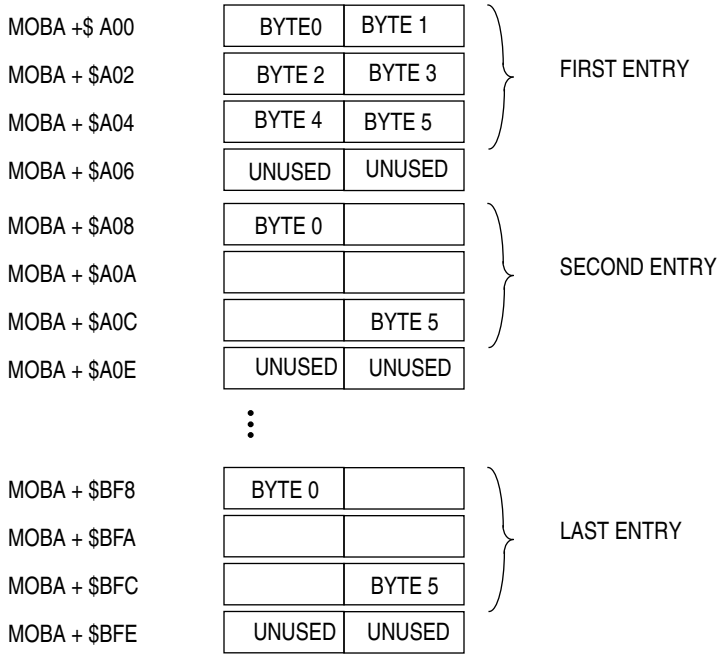


Figure 4-5. AR Memory Map - Perfect Match Mode

When HASH_EN is set, the last two entries in the table are used for the logical address filter mask bits. Hash index 0 is located in the least significant bit of byte 7 in the hash table (MOBA + BFB). Hash index 63 is bit 7 of byte 0 (MOBA + BF0) in the hash table. When writing logical address filter mask bits there is no restriction on the ordering of the writes. When HASH_EN is set, the memory map is changed as shown in Figure 4-6. When HASH_EN is set, locations MOBA + BF4, + BF6 AND MOBA + BFC, +BFE should not be read or written to ($\overline{\text{DTACK}}$ will not be returned).

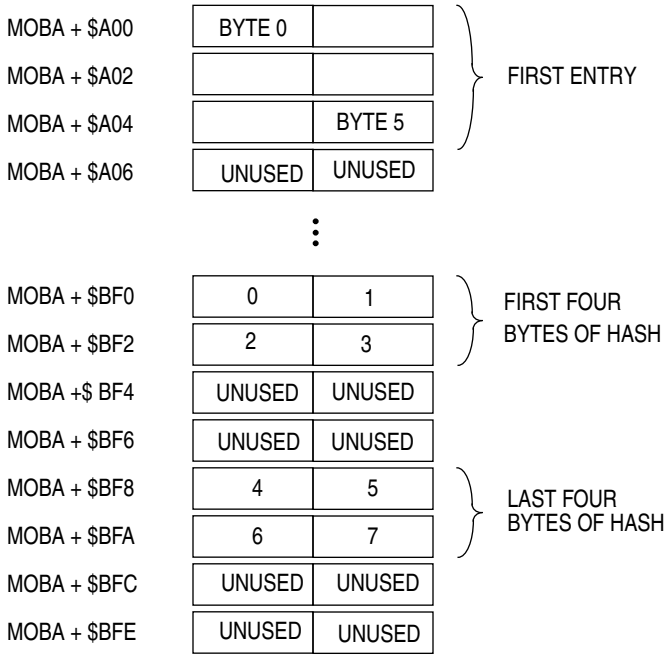


Figure 4-6. AR Memory Map - Hash Mode

4.6.3 READING ADDRESSES FROM TABLES

The address recognition block is read as a normal memory cycle. There is no restriction on the order of bytes to be read.

SECTION 5
SIGNAL DESCRIPTIONS

This section contains descriptions of the MC68EN302 signals using the MC68302 as a reference.

5.1 PIN/SIGNAL COMBINATIONS

The following table defines the MC68EN302 signals and the pinouts of the 144 pin TQFP package. The PGA package has several exclusive signals not included on the TQFP package. The signals exclusive to the PGA package are IAC, $\overline{FRZ}$ and two temperature sense pins which are internally connected to each other.

Table 5-1. MC68EN302 144-TQFP Pin/Signal Definition

TQFP PIN	SIGNAL(S)	TYPE
1	TIN2/PB5	BIDIR
2	A0/$\overline{TOUT1}$/PB4	BIDIR
3	VCC1	PWR
4	TIN1/PB3	BIDIR
5	$\overline{DRAMRW}$/PB2/IACK1	BIDIR
6	$\overline{CAS1}$/PB1/IACK6	BIDIR
7	$\overline{CAS0}$/PB0/IACK7	BIDIR
8	GND1	PWR
9	$\overline{UDS}$	BIDIR
10	$\overline{LDS}$	BIDIR
11	$\overline{AS}$	BIDIR
12	$\overline{RW}$	BIDIR
13	GND2	PWR
14	XTAL	OUT
15	EXTAL	IN
16	VCC2	PWR
17	CLKO	OUT
18	$\overline{IPL0}$/$\overline{IRQ1}$	IN
19	$\overline{IPL1}$/$\overline{IRQ6}$	IN
20	$\overline{IPL2}$/$\overline{IRQ7}$	IN
21	$\overline{BERR}$	BIDIR
22	$\overline{AVEC}$/$\overline{IOUT0}$	BIDIR
23	$\overline{RESET}$	BIDIR
24	$\overline{HALT}$	BIDIR
25	$\overline{BR}$	BIDIR
26	$\overline{BGACK}$	BIDIR
27	$\overline{BG}$	BIDIR
PGA only	TPAD1 (TEMP_SENSE)	
28	TMS	IN

Table 5-1. MC68EN302 144-TQFP Pin/Signal Definition

29	TCK	IN
30	$\overline{\text{TRST}}$	IN
31	$\overline{\text{DTACK}}$	BIDIR
32	GND3	PWR
33	VCC3	PWR
34	RCLK1/L1CLK	BIDIR
35	TCLK1/L1SY0/SDS1	BIDIR
36	TXD1/L1TXD	OUT
37	RTS1/L1RQ/GCIDCL	OUT
38	RTS3/SPTXD	OUT
39	CD3/SPCLK	BIDIR
40	$\overline{\text{AMUX}}$/BRG1	OUT
41	GND	IN
42	PARITY1/BUSW	BIDIR
43	PARITY0/DISCPU	BIDIR
44	VCC4	PWR
PGA only	FRZ	IN
45	GND4	PWR
46	$\overline{\text{PARITYE}}$/THREESTATE	BIDIR
47	WEL/$\overline{\text{DREQ}}$/PA13	BIDIR
48	WEH/$\overline{\text{DACK}}$/PA14	BIDIR
49	$\overline{\text{OE}}$/DONE/PA15	BIDIR
50	$\overline{\text{RAS1}}$/BRG3/PA12	BIDIR
51	$\overline{\text{RAS0}}$/BRG2/SDS2/PA7	BIDIR
52	GND5	PWR
53	TCLK3/PA11	BIDIR
54	RCLK3/PA10	BIDIR
55	TXD3/PA9	BIDIR
56	RXD3/PA8	BIDIR
57	VCC5	PWR
58	CD2/PA6	BIDIR
59	$\overline{\text{RTS2}}$/PA5	BIDIR
60	$\overline{\text{CTS2}}$/PA4	BIDIR
61	GND6	PWR
62	TCLK2/PA3	BIDIR
63	RCLK2/PA2	BIDIR
64	TXD2/PA1	BIDIR
65	RXD2/PA0	BIDIR
66	RXD1/L1RXD	IN
67	CTS1/L1GR	IN
68	CD1/L1SY1	IN
69	CTS3/SPRXD	IN
70	RX	BIDIR
71	RENA	BIDIR
72	TX	OUT
73	VCC6	PWR
74	GND7	PWR
75	CLSN	IN
76	RCLK	IN
77	TENA	OUT

Table 5-1. MC68EN302 144-TQFP Pin/Signal Definition

78	TCLK	IN
79	D0	BIDIR
80	D1	BIDIR
81	D2	BIDIR
82	D3	BIDIR
83	GND8	PWR
84	D4	BIDIR
85	D5	BIDIR
86	D6	BIDIR
87	D7	BIDIR
88	VCC7	PWR
89	D8	BIDIR
90	D9	BIDIR
91	D10	BIDIR
92	D11	BIDIR
93	GND9	PWR
94	D12	BIDIR
95	D13	BIDIR
96	D14	BIDIR
97	D15	BIDIR
98	GND10	PWR
99	VCC8	PWR
100	A23	BIDIR
101	A22	BIDIR
102	A21	BIDIR
103	A20	BIDIR
104	GND11	PWR
105	A19	BIDIR
106	A18	BIDIR
107	A17	BIDIR
108	A16	BIDIR
109	VCC9	PWR
110	A15	BIDIR
111	A14	BIDIR
112	A13	BIDIR
113	A12	BIDIR
114	GND12	PWR
115	A11	BIDIR
116	A10	BIDIR
117	A9	BIDIR
118	A8	BIDIR
119	A7	BIDIR
120	A6	BIDIR
121	A5	BIDIR
122	A4	BIDIR
123	GND13	PWR
124	A3	BIDIR
125	A2	BIDIR
126	A1	BIDIR
127	FC0	BIDIR

Table 5-1. MC68EN302 144-TQFP Pin/Signal Definition

128	VCC10	PWR
129	FC1	BIDIR
130	FC2	BIDIR
131	$\overline{CS0}$/$\overline{IOUT2}$	OUT
132	$\overline{CS1}$	OUT
133	GND14	PWR
134	$\overline{CS2}$	OUT
135	$\overline{CS3}$	OUT
PGA only	TPAD2 (TEMP_SENSE)	
136	TDO	OUT
137	TDI	IN
PGA only	IAC	OUT
138	PB11	BIDIR
139	PB10	BIDIR
140	PB9	BIDIR
141	PB8	BIDIR
142	$\overline{WDOG}$/PB7	BIDIR
143	GND15	PWR
144	$\overline{TOUT}$2/PB6	BIDIR

5.2 MC68EN302/MC68302 COMMON SIGNALS

The following pin/signal combinations are common between the MC68302 and MC68EN302. Any differences in functionality are noted.

A23–A1
D15–D0

In 8-bit mode bit 15–bit 8 is used on the MC68EN302 rather than bit 7–bit 0 on the MC68302. All 16 bits of the data bus are driven during an 8-bit write.

R/$\overline{W}$
$\overline{AS}$
$\overline{UDS}$

A0 is not multiplexed with $\overline{UDS}$ on the EN302

$\overline{LDS}$
$\overline{DTACK}$
$\overline{AVEC}$/$\overline{IOUT0}$
$\overline{IPL2}$/$\overline{IRQ7}$
$\overline{IPL1}$/$\overline{IRQ6}$
$\overline{IPL0}$/$\overline{IRQ1}$
FC2–FC0
$\overline{BR}$
$\overline{BG}$
$\overline{BGACK}$
$\overline{BERR}$
$\overline{RESET}$
$\overline{HALT}$
$\overline{TCLK3}$/PA11

$\overline{\text{RCLK3}}$/PA10
$\overline{\text{TXD3}}$/PA9
$\overline{\text{RXD3}}$/PA8
$\overline{\text{CD2}}$/PA6
$\overline{\text{RTS2}}$/PA5
$\overline{\text{CTS2}}$/PA4
$\overline{\text{TCLK2}}$/PA3
$\overline{\text{RCLK2}}$/PA2
$\overline{\text{TXD2}}$/PA1
PB11–PB8
$\overline{\text{WDOG}}$/PB7
$\overline{\text{TOUT2}}$/PB6
TIN2/PB5
TIN1/PB3
RXD1/L1RXD
TXD1/L1TXD
RCLK1/L1CLK
TCLK1/L1SY0/SDS1
$\overline{\text{CD1}}$/S1SY1
$\overline{\text{CTS1}}$/L1GR
$\overline{\text{RTS1}}$/L1RQ/GCIDCL
$\overline{\text{CD3}}$/SPCLK
$\overline{\text{CTS3}}$/SPRXD
$\overline{\text{RTS3}}$/SPTXD
CS3–CS1
CS0/$\overline{\text{IOUT2}}$
CLKO
XTAL
EXTAL

5.3 MC68302 SIGNALS REMOVED OR REDEFINED

The following signals that are present on the MC68302 are not present on the MC68EN302 or have been redefined in some way.

RMC/$\overline{\text{IOUT1}}$
IAC
$\overline{\text{BCLR}}$
FRZ
BUSW
DISCPU

5.3.1 RMC/IOUT1

This MC68302 output is not available on the MC68EN302.

5.3.2 IAC

This MC68302 output is not available on the MC68EN302 in the 144-pin TQFP package. It is available in the PGA package.

5.3.3 $\overline{\text{BCLR}}$

This MC68302 signal is not available on the MC68EN302. The additional EN302 logic monitors the $\overline{\text{BCLR}}$ signal from the 302 core, but does not drive it.

5.3.4 $\overline{\text{FRZ}}$

This MC68302 input is now pulled up internally and is not available on the MC68EN302 in the 144 pin TQFP package. It is available in the PGA package.

5.3.5 BUSW

The BUSW signal (bus width) is a dedicated pin on the MC68302, but is muxed with PARITY1 on the MC68EN302. This pin functions as BUSW only during system reset. If the BUSW signal is low during EN302 reset, the EN8 bits in the CSER3–CSER0 register are set. The BUSW signal is not passed to the 302 core on the MC68EN302, since the 302 core of the MC68EN302 always operates in 16-bit mode. In 8-bit mode, bits D15–D8 of the data bus are used. This is in contrast with the MC68302, which uses bits D7–D0 in 8-bit mode.

5.3.6 DISCPU

The DISCPU signal is a dedicated pin on the MC68302 but is muxed with PARITY0 on the MC68EN302. This pin is DISCPU only during system reset.

5.4 MC68EN302 NEW SIGNALS MUXED WITH EXISTING MC68302 SIGNALS

Several pins on the MC68EN302 have enhanced MC68302 functionality. The additional signal capability is controlled by the PM9–PM0 field in the MBCTRL (Module Bus Control) register.

Table 5-2. Pin Muxing Control

PM[I]	MC68EN302 PM[I]=0	MC68EN302 PM[I]=1	MC68302
0	$\overline{\text{AMUX}}$	BRG1	BRG1
1	$\overline{\text{RAS0}}$	BRG2/SDS2/PA7	BRG2/SDS2/PA7
2	$\overline{\text{RAS1}}$	BRG3/PA12	BRG3/PA12
3	$\overline{\text{CAS0}}$	PB0/IACK7	PB0/IACK7
4	$\overline{\text{CAS1}}$	PB1/IACK6	PB1/IACK6
5	$\overline{\text{DRAMRW}}$	PB2/IACK1	PB2/IACK1
6	A0	TOUT1/PB4	$\overline{\text{TOUT1}}$/PB4
7	$\overline{\text{DREQ}}$/PA13	$\overline{\text{WEL}}$	$\overline{\text{DREQ}}$/PA13
8	$\overline{\text{DACK}}$/PA14	$\overline{\text{WEH}}$	$\overline{\text{DACK}}$/PA14
9	$\overline{\text{OE}}$	$\overline{\text{DONE}}$/PA15	$\overline{\text{DONE}}$/PA15

5.4.1 $\overline{\text{AMUX}}$ - DRAM Address Mux

The $\overline{\text{AMUX}}$ pin is an output only pin provided for implementing external address muxing circuitry when accessing DRAM. The user may require use of this signal if external masters are utilizing the EN302 DRAM controller. When performing an access to the EN302 as a slave, the address is driven as an input, preventing the EN302 DRAM controller from driving the address bus. Because of this, external muxing must take place. The $\overline{\text{AMUX}}$ pin is also useful in implementations where a linear DRAM space is required.

The $\overline{\text{AMUX}}$ signal appears on the pin if the PM0 bit of the MBCTRL = 0. If PM0 = 1 then the pin becomes BRG1 instead.

5.4.2 $\overline{\text{RAS0}}$ - DRAM Row Address Select, Bit Zero

When the PM1 bit of the MBCTRL = 0, this active low output signal is used to select one of two banks of DRAM as determined by the DRAM Base Address Register 0 (DBA0).

If PM1 = 1 then the pin becomes BRG2/SDS2/PA7 and is bidirectional depending on the function chosen

5.4.3 $\overline{\text{RAS1}}$ - DRAM Row Address Select Bit 1

When the PM2 bit of the MBCTRL = 0, this active low output signal is used to select one of two banks of DRAM as determined by the DRAM Base Address Register1 (DBA1) CSR.

If PM2 = 1, then the pin is used as BRG3/PA12 and may be bidirectional

5.4.4 $\overline{\text{CAS0}}$ - DRAM Column Address Select Bit 0

If the PM3 bit of the MBCTRL = 0, this active low output signal is used to enable the DRAM module upper byte (bits 15–8).

If PM3 = 1 then the pin is used for PB0/IACK7 and may be bidirectional.

5.4.5 $\overline{\text{CAS1}}$- DRAM Column Address Select Bit 1

If the PM4 bit of the MBCTRL = 0, this active low output signal is used to enable the lower byte (bits 7–0) of the DRAM module.

If PM4 = 1 then the pin is used for PB1/IACK6 and may be bidirectional.

5.4.6 $\overline{\text{DRAMRW}}$- DRAM Read/Write

If the PM5 bit of the MBCTRL = 0, this pin is asserted low for a DRAM write cycle. It is separate from the processor bus R/$\overline{\text{W}}$ signal to allow precharge to take place without regard to the state of R/$\overline{\text{W}}$.

If PM5 = 1 then the pin is used for PB2/IACK1 and may be bidirectional.

5.4.7 A0

A new signal, A0 has been added for supporting dynamic bus sizing. This signal replaces the MC68302 A0 that was multiplexed onto $\overline{\text{UDS}}$. A0 is bidirectional.

The A0 signal appears on the pin if the PM6 bit of the MBCTRL = 0. If PM6= 1 then the pin is used as $\overline{\text{TOUT1}}$/PB4.

5.4.8 $\overline{\text{WEL}}$- Write Enable for Byte 1 (Bit 7–Bit 0)

If the PM7 bit of the MBCTRL = 1, then the active low signal $\overline{\text{WEL}}$ is enabled. If PM7 = 0, then the pin is used for $\overline{\text{DREQ}}$/PA13 and is bidirectional.

5.4.9 $\overline{\text{WEH}}$ - Write Enable for Byte 0 (Bit 15–Bit 8)

If the PM8 bit of the MBCTRL = 1, the active low $\overline{\text{WEH}}$ signal appears on the pin. If PM8 = 0, then the pin is used for $\overline{\text{DACK}}$/PA14 and is bidirectional.

5.4.10 $\overline{\text{OE}}$ - Output Enable

Output - asserted low, pin is bidirectional.

If the PM9 bit of the MBCTRL = 0, the active low $\overline{\text{OE}}$ signal is enabled. If PM9 = 1 then the pin is used for $\overline{\text{DONE}}$/PA15 and may be bidirectional.

5.5 MC68EN302 ONLY PIN/SIGNALS

TENA
TX
TCLK
RENA
RX
RCLK
CLSN
PARITY0/DISCPU
PARITY1/BUSW
$\overline{\text{PARITYE}}$/THREES
TMS
TCK
$\overline{\text{TRST}}$
TDO
TDI
GND

5.5.1 GND

For proper implementation of the EN302, this pin must be tied to ground.

5.5.2 $\overline{\text{TRST}}$ - JTAG Reset Signal

For normal operation the asynchronous $\overline{\text{TRST}}$ signal must be held low during system reset. This pin has an internal pullup resistor.

5.5.3 TMS - JTAG Test Mode Select

This input controls test mode operation for the EN302 test logic as defined by the IEEE 1149.1 standard. This pin has an internal pullup resistor.

5.5.4 TDO - JTAG Test Data Out

This output is used in shifting serial test instructions and test data for on-board test logic defined by the IEEE 1149.1 standard.

5.5.5 TDI - JTAG Test Data In

This input is used for shiftng serial test instructions and test data for on-board test logic defined by the IEEE 1149.1 standard. This pin has an internal pullup resistor.

5.5.6 TCK- JTAG Clock

The JTAG clock runs at a frequency no greater than 10 MHz.

5.5.7 GND

Must be tied to GND in normal operation.TX

Ethernet MAC transmit data output.

5.5.8 TENA

Ethernet MAC transmit data valid output.

5.5.9 TCLK

Ethernet transmit clock input must be 10 MHz +/- 100 ppm according to the 802.3 spec.

5.5.10 RCLK

Ethernet receive clock input.

5.5.11 RX

Ethernet MAC receive data input.

5.5.12 RENA

This input indicates that the Ethernet MAC receive data is valid.

5.5.13 CLSN

This input pin indicates a collision (or SQE Test) was detected in the ethernet physical layer.

5.5.14 PARITY0/DISCPU

Parity is controlled by the PCSR register in the module bus controller. This bidirectional pin provides even or odd parity for byte 0 (bit 15–bit 8) when not in the reset state.

The MC68EN302 DISCPU state is sampled during hardware reset just as in the MC68302. The M68000 core is disabled by asserting the DISCPU pin high during total system reset.

5.5.15 PARITY1/BUSW

Parity is controlled by the PCSR Register in the module bus controller. This bidirectional pin provides even or odd parity for byte 1 (bit 7–bit 0) when not in the reset state.

The state of BUSW is sampled during total system reset. When the BUSW is low during hardware reset, it does not put the 68000 into 68008 mode with an 8 bit bus. Instead, having the BUSW low during hardware reset will force the four EN8 bits in the CSER registers to one, enabling support for dynamic bus sizing in the chip selects. Note that because the 68000 core is in normal 16 bit mode, if the 68000 accesses memory outside of the four chip select areas, it always performs a normal 16 bit access.

5.5.16 $\overline{\text{PARITYE}}$/THREESTATE

During normal operation, this bidirectional pin is the active low $\overline{\text{PARITYE}}$ (parity error) output, and is asserted whenever one of the PED (Parity Error DRAM) bits in the PCSR is asserted.

If this pin is low during total system reset, all bidirectional pins and output pins will be put into three-state mode. This is intended for chip test purposes.

5.6 DRAM CONTROLLER I/O

5.6.1 Control Signal Pins

The EN302 contains 8 DRAM specific signal pins: $\overline{\text{CAS1}}$–$\overline{\text{CAS0}}$, $\overline{\text{RAS1}}$–$\overline{\text{RAS0}}$, $\overline{\text{AMUX}}$, and $\overline{\text{DRAMRW}}$.

5.6.2 Column Address Strobes ($\overline{\text{CAS1}}$–$\overline{\text{CAS0}}$)

These active low pins allow seamless interface to Column Address Strobe ($\overline{\text{CAS}}$) inputs on industry standard DRAM, providing $\overline{\text{CAS}}$ for both bank 0 and bank 1 of the DRAM controller. Two strobes support byte operations on the external 16-bit bus. $\overline{\text{CAS0}}$ corresponds to data pins D15-D8. $\overline{\text{CAS1}}$ corresponds to data pins D7–D0.

5.6.3 Row Address Strobes ($\overline{\text{RAS1}}$–$\overline{\text{RAS0}}$)

These active low pins allow seamless interface to Row Address Strobe ($\overline{\text{RAS}}$) inputs on industry standard DRAM, providing $\overline{\text{RAS}}$ for both bytes of a given DRAM bank. A particular bank corresponds to specific Base Address and Control information programmed in the MC68EN302 DRAM control registers (see 3.2 Memory Map for a description). $\overline{\text{RAS0}}$ corresponds to bank 0 and $\overline{\text{RAS1}}$ corresponds to Bank 1.

5.6.4 DRAM Read/Write ($\overline{\text{DRAMRW}}$)

This active low pin is asserted to signify that a DRAM write cycle is occurring. It is separate from the processor bus R/$\overline{\text{W}}$ so that precharge takes place without regard to the state of R/$\overline{\text{W}}$.

5.6.5 Address Mux ($\overline{\text{AMUX}}$)

The $\overline{\text{AMUX}}$ pin is provided for implementing external address muxing circuitry so that external masters may access DRAM modules controlled by the MC68EN302 DRAM controller. External address muxing must take place in this situation since an access to the MC68EN302 as a slave always results in the addresses driven as an input, and does not output addresses to the DRAM module.

Another use for the $\overline{\text{AMUX}}$ pin would be implementations in which a linear DRAM space is required.

5.6.6 Parity (PARITY1–PARITY0)

These two pins are provided to support parity checking of DRAM. If enabled, parity is generated on writes and checked on reads. A parity error on a read generates a bus error. PARITY0 is used in connection with D15-D8 and PARITY1 is used in connection with D7–D0.

Parity checking/generation is not supported for external bus masters.

5.6.7 Muxing Scheme

To provide a simplified implementation of the Address Mux, a unique muxing scheme is provided. Rather than providing programmability to change which addresses are muxed on a particular signal, a generic muxing scheme is provided so that one muxing scheme may be utilized by all supported DRAM bank sizes. Table 5-3 shows the DRAM muxing scheme. The usage listed in the table is for typical operation. It is possible that some users may utilize the Base Address Registers and the Mask bits in a non-standard way.

Table 5-3. Address Muxing Scheme

PROCESSOR ADDRESS	ROW ADDRESS	COLUMN ADDRESS	USAGE
A9	9	1	Used for all Bank Sizes
A10	10	2	
A11	11	3	
A12	12	4	
A13	13	5	
A14	14	6	
A15	15	7	
A16	16	8	
A18	18	17	Used for 512K and up
A20	20	19	Used for 2M and up
A22	22	21	Used for 8M

SECTION 6
APPLICATIONS

6.1 BRINGING THE MC68EN302 OUT OF RESET

The following paragraphs provide an example of how to bring the MC68EN302 out of reset and initialize the Ethernet Controller to perform internal loopback of one frame. Bank 0 of DRAM is used as packet memory.

1. Write the Base Address Register (BAR) with the desired starting point of the 302 core 4k-byte relocatable address space. Write the Module Bus Controller Base Address Register (MOBAR) with the starting point of the 4k-byte relocatable address space for the module bus controller, DRAM controller and Ethernet controller CSRS and memory.
2. Write to the Option Register (OR) to include 256K bytes of space and so that the DTACK field may be written to to change the number of wait states. Also note that to access data in the program ROM, the CFC bits should be modified.
3. OR1 affects the RAM range, controls $\overline{\text{DTACK}}$, and will also affect whether or not function code comparisons are performed.
4. BR1 will set up the RAM address, enable the RAM, and set the function code appropriately.
5. Switch from ROM location to dual-port RAM location to assure that the reset vector is supplied by the ROM, but the exception vectors all come from the RAM. This switch is performed by a short, dual-port RAM program which is summarized below, and is explained in depth in Appendix D.2 of the MC68302 User's Manual.

   MOVE.W #$A001, (Address of BR1).L

   MOVE.W #$C201, (Address or BR0).L

   JMP ($Address in ROM).L

   After the code is copied, then execute the following instruction, which will cause a jump to dual port RAM

   JMP ($Base Address).L

6. MBC - the MBC register controls bits for overall system level functionality of the Module Bus Controller. This register must be initialized to assure smooth functionality between the SIM module on the MC68EN302 and the SIM module on the internal 302 core. In the MBC register the module bus controller response to $\overline{\text{BCLR}}$ from the internal 302 is controlled, as is the parity, function code for the Ethernet specific core,

and the pin muxing that is used in the current MC68EN302 application.

Example: Write $5400 to MBC (MFC = 5, PPE = 1).

7. IER - The interrupt extension register replaces the MOD, ET7, ET6 and ET1 bits in the 302 GIMR. *The user MUST assure that the corresponding bits in the GIMR of the internal 302 are all written as zeros for proper functionality of the MC68EN302*. The IER is reset to $0000 which configures the interrupt input pins as $\overline{IPL2}$–$\overline{IPL0}$ and sets the module bus controller interrupt to level 5.
8. CSER0–CSER3. The MC68EN302 extends the functionality of that provided by the internal 302 core chip selects through the programming of this register. Additional functionality includes 8-bit bus operation as well as parity checking and generation.
9. PCSR - This register controls parity operation on the MC68EN302. Also, bits 9-8 show the result of parity on the current DRAM bank.
10. DRAM controller initialization. Assume bank 0 is to be used, parity enabled, 0 wait states.

- DRAM Configuration Register (DCR) = $0501 (Enable refresh and parity in bank 0, allow supervisor or user access)
- DRAM Refresh Register (DRFRSH) = $0000 (Refresh every 4096 system clocks)
- DRAM Base Address Registers
  DBA0 = desired DRAM base address and size, bit 0 = 1
  DBA1 = $0000 (reset value)
- To ensure correct parity, write $0000 to each memory location used before running other code.

11. Ethernet Controller Initialization. In this example the Ethernet Controller is initialized to perform internal loopback of one frame. The received frame buffer will be 4 bytes longer than the transmit buffer due to the CRC being appended by hardware.

- ECNTRL = $0001 (Release RESET to the Ethernet Controller)
- EDMA = $000B
  WMRK = 01 (16 bytes)
  BLIM = 011 (max DMA burst length of 8 bus transactions (16 bytes of data))
- EMRBLR = $0600 (1536 bytes, this allows receiving a max size frame into a single buffer).
- IVEC = $0140
  VG = 1 (bit 1–bit 0 of the interrupt vector will be modified).
  INV7–INV0 = 40
- INTR_MASK = $07BC (all interrupts enabled except BackOffDone, TransmitBuffer and ReceiveBuffer).
- ECNFIG = $0001 (enable internal loopback).
- ETHER_TEST = $0000

- AR_CNTRL = $7000
  HASH_EN = 0 (all 64 entries used for “match” mode)
  INDEX_EN = 1 (enables hardware to write “Reason” and “ARIndex” fields into the receive BD.
  MULT1–MULT0 = 11 (reject multicast and broadcast frames).
- Place the frame to be looped internally into memory. Entire frame should be in a single buffer. Allocate receive buffer memory to receive same frame plus 4 CRC bytes.
- Initialize CAM (arbitrarily select 1 of 64 entries to contain the DA of the frame to be transmitted). CAM is in perfect match mode. CAM starts at MOBA + $A00.
  Write $FF_FF_FF into all CAM entries except 1
  Write DA of transmit frame into remaining entry
- Initialize buffer descriptors. A good practice would be to initialize all locations to $0000 before putting in any specific values.
  MOBA + C48 = $0000 (clear E bit in second receive BD)
  MOBA + C46 = $LLLL (A15–A0 pointer to receive buffer)
  MOBA + C44 = $00HH (A23–A16 pointer to receive buffer)
  MOBA + C40 = $8000 (set E bit in first receive BD)
  MOBA + C08 = $0000 (clear R bit in second transmit BD)
  MOBA + C06 = $LLLL (A15–A0 pointer to transmit buffer)
  MOBA + C04 = $00HH (A23–A16 pointer to transmit buffer)
  MOBA + C02 = $0NNN (transmit buffer length)
  MOBA + C00 = $8C00 (single buffer frame, hardware appends CRC)
- ECNTRL = $0003 (assert ETHER_EN to the Ethernet Controller, this will cause the buffer descriptor and DMA state machines to start operation)
- Frame loopback should occur under hardware control. The TFINT and RFINT interrupts should occur. Once these interrupts have occurred, frame loopback can be verified by the following:
  - Receive buffer should contain frame transmitted plus 4 byte CRC
  - Transmit buffer descriptor should contain the following:
    MOBA + C00 = 0C00
    MOBA + C02, C04, C06 locations should be unchanged
  - Receive buffer descriptor should contain the following:
    MOBA + C40 = $0C00
    MOBA + C42 = value in (MOBA + C02) + 4
    MOBA + C44, C46 locations should be unchanged

6.2 MOVING A QUICC ETHERNET DRIVER TO A 68EN302 ETHERNET DRIVER

Porting an Ethernet driver written for the MC68360 QUICC to the MC68EN302 requires only that the QUICC driver be pared down to support a simpler implementation of Ethernet. In the case of register settings and counters, many of the functions requiring user initialization by the QUICC are either supported directly in the MC68EN302 hardware as dictated by the Ethernet standard, or are provided in the indications that accompany the buffer descriptors. This simplifies the initialization routines in the area of CRC calculation as well as the maximum and minimum frame lengths, DMA operations and the backoff counter

operations.The MC68EN302 buffer descriptors are a superset of the QUICC buffer descriptors. The transmit buffer descriptors are identical, while the MC68EN302 receive buffer descriptors include an additional bit in the most significant word which is used as an indication for address matching while running in promiscuous mode. Any driver that implements Ethernet on the QUICC will be able to utilize the same buffer descriptor structure and handling when running on the MC68EN302. Following the Ethernet register map from the MC68360UM/AD Revision 2.0 user's manual in the table listing the Ethernet Specific Parameters (in the Parameter RAM), the corresponding function in the MC68EN302 is listed below. Items in italics indicate that that particular parameter was not part of the QUICC initialization process and is not required for CRC.

The standard 32 bit CRC calculation for the CRC is performed in the MC68EN302 hardware, and therefore there is no need to write to the CRC value.

6.2.1 C_PRES, C_MASK:

The QUICC preset and mask options for the CRC calculation are not required in the MC68EN302 because the CRC is automatically calculated in hardware.

6.2.2 CRCEC:

In the MC68EN302, CRC errors are flagged in the buffer descriptor indication rather than keeping a running counter as in the QUICC. A software counter may be implemented to support this function with very little difficulty by incrementing the count each time the CR bit (bit 2 in the most significant long word of the buffer descriptor) is set.

6.2.3 ALEC:

The Alignment error is flagged in the buffer descriptor indication and therefore a software counter may be implemented to support this function. A counter may be incremented each time the NO bit (bit 4 in the most significant long word of the buffer descriptor) is set.

6.2.4 DISFC:

This function is not required in standard Ethernet. If frames are discarded because of error conditions, then the buffer descriptor is flagged with the error notification. If a buffer descriptor is not available, then an overrun error occurs, notifying the user that a frame was discarded because no buffer descriptors were available. If a frame is discarded because of the address filtering that is implemented on the MC68EN302, there is no indication provided since this function is generally used for collecting network statistics and does not add to the station performance.

6.2.5 PADS:

In the transmit direction the padding is automatic and is always generated as all ones.

6.2.6 RET_LIM:

Set in hardware to standard Ethernet values of 15.

6.2.7 RET_Cnt:

The RC field in the transmit buffer descriptor is the retry count (bits 5-2 of the most significant long word). If the RL bit is also set (bit 6) then the RC bits have no meaning since the retry limit has been exceeded.

6.2.8 MFLR:

The maximum frame length is set in MC68EN302 hardware to 1518.

6.2.9 MINFLR:

The minimum frame length is set in MC68EN302 hardware to 64.

6.2.10 MAXD1, MAXD2:

These operations are covered by the maximum frame length being automatically set at 1518 bytes. This is not programmable in the MC68EN302.

6.2.11 MAX_b:

The MC68EN302 supports buffer descriptor options ranging from 8 Tx and 120 RX buffer descriptors to 64 TX and 64 RX buffer descriptors.

6.2.12 GADDR1-4 / PADDR_HML / IADDR1-4:

AR_CNTL provides the address filtering. Options in the MC68EN302 are either 64 perfect address matches or 62 perfect address matches with address filtering via a hash routine.

6.2.13 P_PER:

The persistance is not programmable on the MC68EN302 but rather is a specific value set in the MC68EN302 hardware.

6.2.14 RFBD_ptr/TFBD_ptr/TLBD_ptr:

Buffer descriptors in the MC68EN302 are placed at MOBAR+$C00 through MOBAR+$FFF. The TFBD_ptr is always at MOBAR+$C00.

The TLBD_ptr can range from MOBAR+$C00 in a one buffer frame to MOBAR+$DFF in a 64 transmit buffer frame.

The RFBD_ptr is either at MOBAR+$C40, MOBAR+$C80, MOBAR+$D00 or MOBAR+$E00 depending on the setting of the BDSIZE bits.

6.2.15 TX_len:

The TX frame length counter is part of the data length field in the buffer descriptor. To come up with the total tx frame length the user must keep track of the data length in the buffer descriptors belonging to that frame.

6.2.16 BOFF_CNT:

The backoff count is set to a specific value in the MC68EN302 hardware.

6.2.17 TADDR_H/M/L:

Implemented in the 64 entry address recognition table as per user set up.

6.2.18 GSMR (QUICC Section 7.10.2)

The following table represents bits relating to Ethernet in the QUICC's GSMR register and indicate the corresponding function on the MC68EN302.

BIT(S)	MNEMONIC	FUNCTION ON MC68EN302
47/46	TCRC	Hardware function
37	RFW	Set width in hardware
28	TCI	Clocking optimized in hardware
23-21	TPL	Preamble set to 6 bytes in hardware
20	TPP	Preamble pattern defined as 10
3-0	MODE	Ethernet only

SECTION 7
IEEE 1149.1 (JTAG) TEST ACCESS PORT

The MC68EN302 provides a dedicated user-accessible test access port (TAP) that is fully compatible with the *IEEE 1149.1 Standard Test Access Port and Boundary Scan Architecture*.

The TAP consists of five dedicated signal pins, a 16-state TAP controller, boundary scan and instruction registers. A boundary scan register links I/O pins into a single shift register. The test logic, implemented utilizing static logic design, is independent of the device system logic. The MC68EN302 implementation provides the capability to:

1. Perform boundary scan operations to test circuit-board electrical continuity.
2. Bypass the MC68EN302 for a given circuit-board test by effectively reducing the boundary scan register to a single cell.
3. Sample the MC68EN302 system pins during operation and transparently shift out the result in the boundary scan register.
4. Disable the output drive to pins during circuit-board testing.

NOTE

Certain precautions must be observed to ensure that the IEEE 1149.-like test logic does not interfere with nontest operation. See 7.6 Non-Scan Chain Operation for details.

In addition to the scan-test logic, the MC68EN302 contains a signal that can be used to three-state all MC68EN302 output signals. This signal, called three-state ($\overline{\text{THREESTATE}}$), is sampled during system reset.

7.1 OVERVIEW

An overview of the MC68EN302 scan chain implementation is shown in Figure 7-1. The MC68EN302 implementation includes a TAP controller, a 4-bit instruction register, and two test registers (a 1-bit bypass register and a 163-bit boundary scan register). This implementation includes a dedicated TAP consisting of the following signals:

- TCK—a test clock input to synchronize the test logic.
- TMS—a test mode select input (with an internal pullup resistor) that is sampled on the rising edge of TCK to sequence the TAP controller's state machine.
- TDI—a test data input (with an internal pullup resistor) that is sampled on the rising edge of TCK.

- TDO—a three-stateable test data output that is actively driven in the shift-IR and shift-DR controller states. TDO changes on the falling edge of TCK.
- $\overline{\text{TRST}}$—an asynchronous reset (with an internal pullup resistor) that provides initialization of the TAP controller and other logic required by the standard.

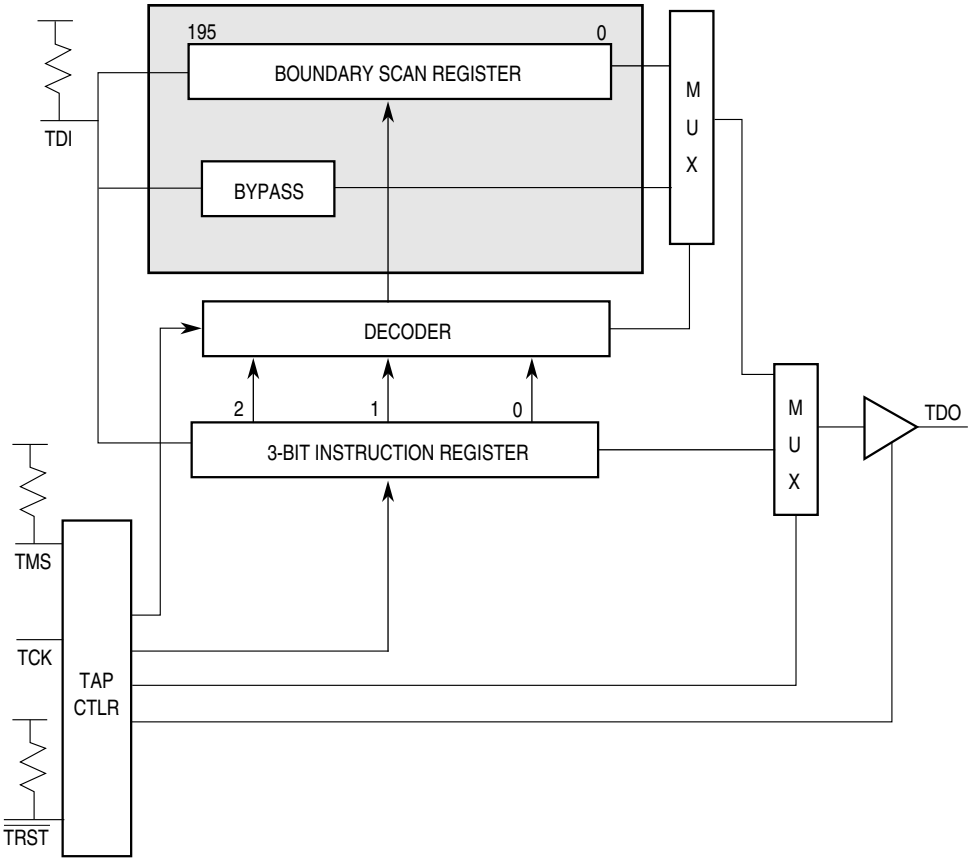


Figure 7-1. Test Logic Block Diagram

7.2 TAP CONTROLLER

The TAP controller is responsible for interpreting the sequence of logical values on the TMS signal. It is a synchronous state machine that controls the operation of the JTAG logic. The state machine is shown in Figure 7-2. The value shown adjacent to each arc represents the value of the TMS signal sampled on the rising edge of the TCK signal. For a description of the TAP controller states, refer to the IEEE 1149.1 document.

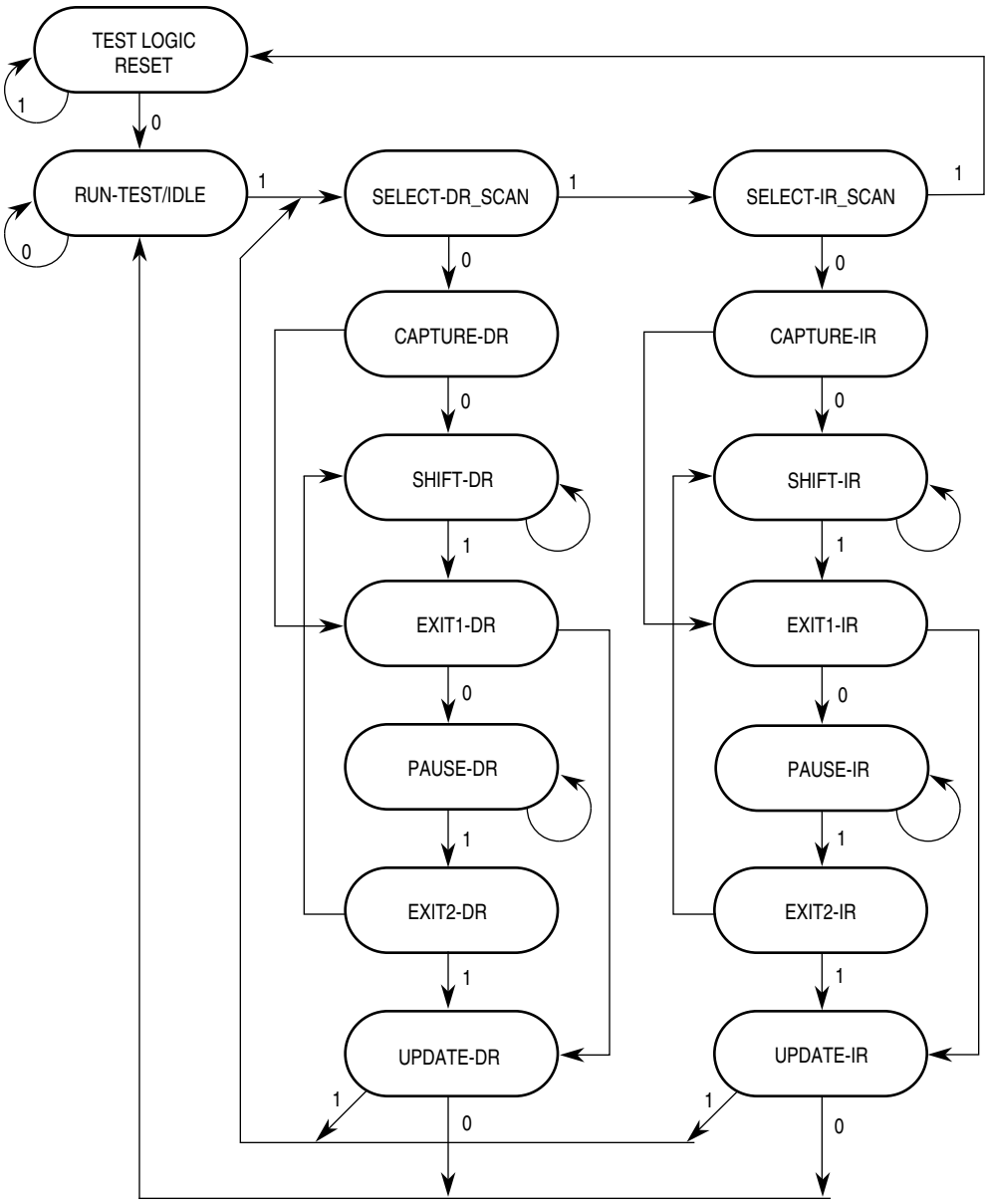


Figure 7-2. TAP Controller State Machine

7.3 BOUNDARY SCAN REGISTER

The MC68EN302 IEEE 1149.1 implementation has a 163-bit boundary scan register. This register contains bits for all device signal pins and associated control signals with the following exceptions. The $\overline{FRZ}$ and IAC signals which are not pinned out on the TQFP package are not included in the scan chain. The XTAL pin is associated with an analog signal and is not included in the boundary scan register. The EXTAL pin (clock in) is not included to minimize loading on this signal, however a boundary scan cell is included for an internal signal, SCLK which is the clock input to the MBC, Ethernet and DRAM controller logic.

All MC68EN302 bidirectional pins have a single register bit in the boundary scan register for pin data and are controlled by an associated control bit in this register. Fifty bits in the boundary scan register define the output enable signal for associated groups of bidirectional and three-stateable output pins. The control bits and their bit positions are listed in Table 7-1.

Table 7-1. Boundary Scan Control Bits

Name	Bit Number	Name	Bit Number	Name	Bit Number
out_enb	0	ras1_enb	86	berrb_out	125
drv_ext_a	5	oeb_enb	88	CLKO_enb	130
drv_ext_d	32	wehb_enb	90	drv_ext_cntl	133
rena_enb	54	welb_enb	92	drv_ext_uds	137
rx_enb	56	paritye_enb	94	cas0_enb	139
pa0_enb	62	parity_enb	96	cas1_enb	141
txd2_enb	64	cd3_enb	101	dram_rw_enb	143
pa2_enb	66	txd1_enb	105	pb3_enb	145
pa3_enb	68	tclk1_enb	107	a0_enb	147
pa4_enb	70	rclk1_enb	109	pb5_enb	149
pa5_enb	72	dtack_enb	111	pb6_enb	151
pa6_enb	74	ecpt_buf	113	wdogb_enb	153
pa8_enb	76	bgack_enb	115	pb8_enb	155
txd3_enb	78	br_enb	117	pb9_enb	157
pa10_enb	80	haltb_out	119	pb10_enb	159
pa11_enb	82	resetb_out	121	pb11_enb	161
ras0_enb	84	avec_enb	123		

The boundary scan bit definitions are listed in Table 7-2.

The first column in the table defines the bit's ordinal position in the boundary scan register. The shift register cell nearest TDO (i.e., first to be shifted out) is defined as bit 0; the last bit to be shifted out is 162.

The second column references one of the seven MC68EN302 cell types depicted in Figure 7-3 through Figure 7-9, which describe the cell structure for each type.

The third column lists the pin name for all pin-related cells or defines the name of bidirectional control register bits.

The fourth column lists the pin type for convenience, where Output indicates a three-stateable output pin, I/O indicates a bidirectional pin and Input represents an input.

The last column indicates the associated boundary scan register control bit for bidirectional and output pins.

Bidirectional pins include a single scan cell for data (bicell) as depicted in Figure 7-6. These bits are controlled by the cell shown in Figure 7-5. The value of the control bit determines whether the bidirectional pin is an input or an output. One or more bidirectional data cells

can be serially connected to a control cell as shown in Figure 7-10. Note that, when sampling the bidirectional data cells, the cell data can be interpreted only after examining the IO control cell to determine pin direction, and also note that the control cell captures the value of the following cell.

Table 7-2. Boundary Scan Bit Definition

Bit Num	Cell Type	Pin/Cell Name	Pin Type	Output CTLCell	Bit Num	Cell Type	Pin/Cell Name	Pin Type	Output CTLCell
0	encell	out_enb	-	-	39	bicell	D9	I/O	drv_ext_d
1	iocell	CS3	Output	out_enb	40	bicell	D8	I/O	drv_ext_d
2	iocell	CS3	Output	out_enb	41	bicell	D7	I/O	drv_ext_d
3	iocell	CS1	Output	out_enb	42	bicell	D6	I/O	drv_ext_d
4	iocell	CS0	Output	out_enb	43	bicell	D5	I/O	drv_ext_d
5	dicell	drv_ext_a	-	-	44	bicell	D4	I/O	drv_ext_d
6	bicell	FC2	I/O	drv_ext_a	45	bicell	D3	I/O	drv_ext_d
7	bicell	FC1	I/O	drv_ext_a	46	bicell	D2	I/O	drv_ext_d
8	bicell	FC0	I/O	drv_ext_a	47	bicell	D1	I/O	drv_ext_d
9	bicell	A1	I/O	drv_ext_a	48	bicell	D0	I/O	drv_ext_d
10	bicell	A2	I/O	drv_ext_a	49	iscell	TCLK	Input	-
11	bicell	A3	I/O	drv_ext_a	50	iocell	TENA	Output	out_enb
12	bicell	A4	I/O	drv_ext_a	51	iscell	RCLK	Input	-
13	bicell	A5	I/O	drv_ext_a	52	iscell	CLSN	Input	-
14	bicell	A6	I/O	drv_ext_a	53	iocell	TX	Output	out_enb
15	bicell	A7	I/O	drv_ext_a	54	dicell	rena_enb	-	-
16	bicell	A8	I/O	drv_ext_a	55	bicell	RENA	I/O	rena_enb
17	bicell	A9	I/O	drv_ext_a	56	dicell	rx_enb	-	-
18	bicell	A10	I/O	drv_ext_a	57	bicell	RX	I/O	rx_enb
19	bicell	A11	I/O	drv_ext_a	58	iscell	CTS3	Input	-
20	bicell	A12	I/O	drv_ext_a	59	iscell	CD1	Input	-
21	bicell	A13	I/O	drv_ext_a	60	iscell	CTS1	Input	-
22	bicell	A14	I/O	drv_ext_a	61	iscell	RXD1	Input	-
23	bicell	A15	I/O	drv_ext_a	62	dicell	pa0_enb	-	-
24	bicell	A16	I/O	drv_ext_a	63	bicell	RXD2	I/O	pa0_enb
25	bicell	A17	I/O	drv_ext_a	64	dicell	txd2_enb	-	-
26	bicell	A18	I/O	drv_ext_a	65	bicell	TXD2	I/O	txd2_enb
27	bicell	A19	I/O	drv_ext_a	66	dicell	pa2_enb	-	-
28	bicell	A20	I/O	drv_ext_a	67	bicell	RCLK2	I/O	pa2_enb
29	bicell	A21	I/O	drv_ext_a	68	dicell	pa3_enb	-	-
30	bicell	A22	I/O	drv_ext_a	69	bicell	TCLK2	I/O	pa3_enb
31	bicell	A23	I/O	drv_ext_a	70	dicell	pa4_enb	-	-
32	dicell	drv_ext_d	-	-	71	bicell	CTS2	I/O	pa4_enb
33	bicell	D15	I/O	drv_ext_d	72	dicell	pa5_enb	-	-
34	bicell	D14	I/O	drv_ext_d	73	bicell	RTS2	I/O	pa5_enb
35	bicell	D13	I/O	drv_ext_d	74	dicell	pa6_enb	-	-
36	bicell	D12	I/O	drv_ext_d	75	bicell	CD2	I/O	pa6_enb
37	bicell	D11	I/O	drv_ext_d	76	dicell	pa8_enb	-	-
38	bicell	D10	I/O	drv_ext_d	77	bicell	RXD3	I/O	pa8_enb

Table 7-2. Boundary Scan Bit Definition

78	dicell	txd3_enb	-	-	118	bicell	BR	I/O	br_enb
79	bicell	TXD3	I/O	txd3_enb	119	dicell	haltb_out	-	-
80	dicell	pa10_enb	-	-	120	bicell	HALT	I/O	haltb_ou
81	bicell	RCLK3	I/O	pa10_enb	121	dicell	resetb_out	-	-
82	dicell	pa11_enb	-	-	122	bicell	RESET	I/O	resetb_out
83	bicell	TCLK3	I/O	pa11_enb	123	dicell	avec_enb	-	-
84	dicell	ras0_enb	-	-	124	bicell	AVEC	I/O	avec_enb
85	bicell	RAS0	I/O	ras0_enb	125	dicell	berrb_out	-	-
86	dicell	ras1_enb	-	-	126	bicell	BERR	I/O	berrb_out
87	bicell	RAS1	I/O	ras1_enb	127	iscell	IPL2	Input	-
88	dicell	oeb_enb	-	-	128	iscell	IPL1	Input	-
89	bicell	OE	I/O	oeb_enb	129	iscell	IPL0	Input	-
90	dicell	wehb_enb	-	-	130	clko_encell	CLKO_enb	-	-
91	bicell	WEH	I/O	wehb_enb	131	iocell	CLKO	Output	CLKO_enb
92	dicell	welb_enb	-	-	132	iscell	SCLK (1)[1]	Input	-
93	bicell	WEL	I/O	welb_enb	133	dicell	drv_ext_cntl	-	-
94	dicell	paritye_enb	-	-	134	bicell	RW	I/O	drv_ext_cntl
95	bicell	PARITYE	I/O	paritye_enb	135	bicell	AS	I/O	drv_ext_cntl
96	dicell	parity_enb	-	-	136	bicell	LDS	I/O	drv_ext_cntl
97	bicell	PARITY0	I/O	parity_enb	137	dicell	drv_ext_uds	-	-
98	bicell	PARITY1	I/O	parity_enb	138	bicell	UDS	I/O	drv_ext_uds
99	iscell	TEST302	Input	-	139	dicell	cas0_enb	-	-
100	iocell	AMUX	Output	out_enb	140	bicell	CAS0	I/O	cas0_enb
101	dicell	cd3_enb	-	-	141	dicell	cas1_enb	-	-
102	bicell	CD3	I/O	cd3_enb	142	bicell	CAS1	I/O	cas1_enb
103	iocell	RTS3	Output	out_enb	143	dicell	dram_rw_enb	-	-
104	iocell	RTS1	Output	out_enb	144	bicell	DRAMRW	I/O	dram_rw_enb
105	encello	txd1_enb	-	-	145	dicell	pb3_enb	-	-
106	iocell	TXD1	Output	txd1_enb	146	bicell	TIN1	I/O	pb3_enb
107	dicell	tclk1_enb	-	-	147	dicell	a0_enb	-	-
108	bicell	TCLK1	I/O	tclk1_enb	148	bicell	A0	I/O	a0_enb
109	dicell	rclk1_enb	-	-	149	dicell	pb5_enb	-	-
110	bicell	RCLK1	I/O	rclk1_enb	150	bicell	TIN2	I/O	pb5_enb
111	dicell	dtack_enb	-	-	151	dicell	pb6_enb	-	-
112	bicell	DTACK	I/O	dtack_enb	152	bicell	TOUT2	I/O	pb6_enb
113	dicell	ecpt_buf	-	-	153	dicell	wdogb_enb	-	-
114	bicell	BG	I/O	ecpt_buf	154	bicell	WDOG	I/O	wdogb_enb
115	dicell	bgack_enb	-	-	155	dicell	pb8_enb	-	-
116	bicell	BGACK	I/O	bgack_enb	156	bicell	PB8	I/O	pb8_enb
117	dicell	br_enb	-	-	157	dicell	pb9_enb	-	-

Table 7-2. Boundary Scan Bit Definition

158	bicell	PB9	I/O	pb9_enb
159	dicell	pb10_enb	-	-
160	bicell	PB10	I/O	pb10_enb

161	dicell	pb11_enb	-	-
162	bicell	PB11	I/O	pb11_enb

Notes:
1. Boundary scan cell for SCLK (bit number 132 in table) is for the internal SCLK signal used in the Ethernet controller. A boundary scan cell was not included on the EXTAL clock input signal to minimize loading.

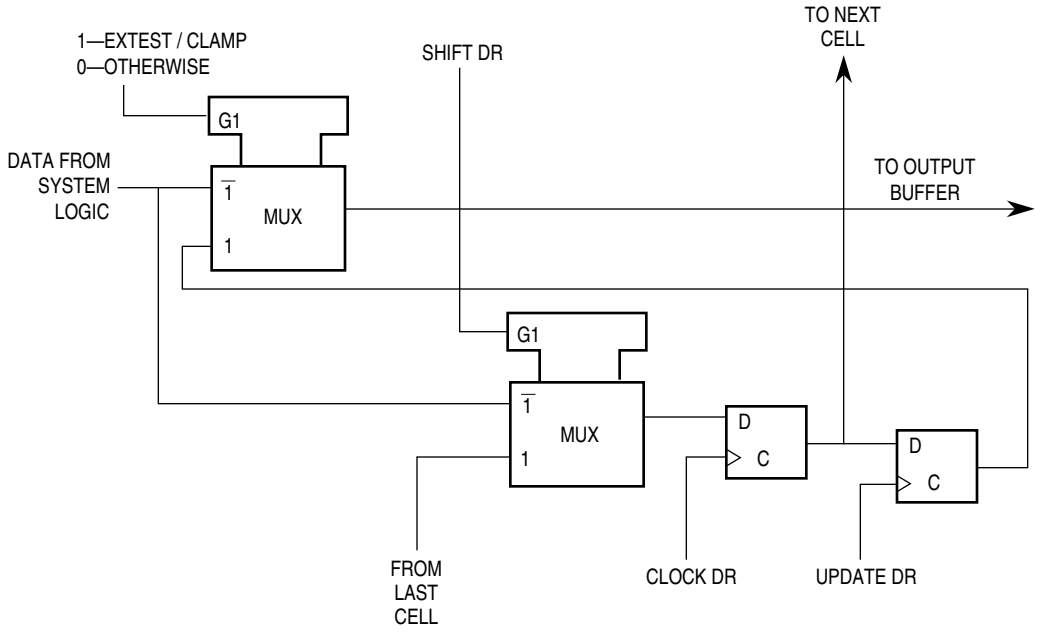


Figure 7-3. Output Latch Cell (iocell)

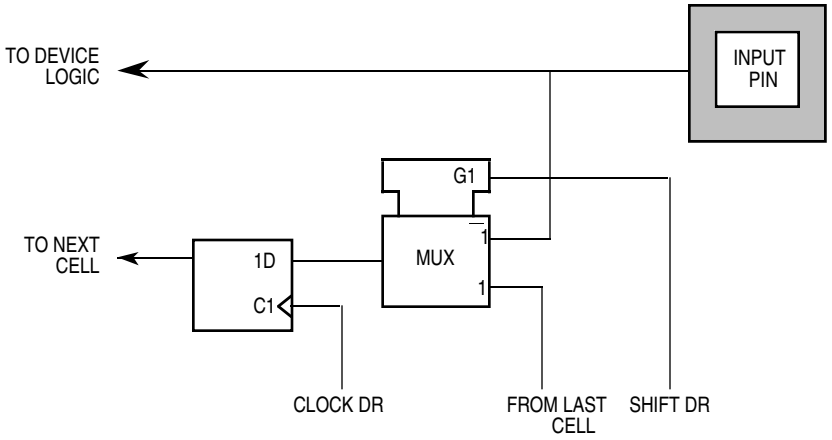


Figure 7-4. Input Pin Cell (iscell)

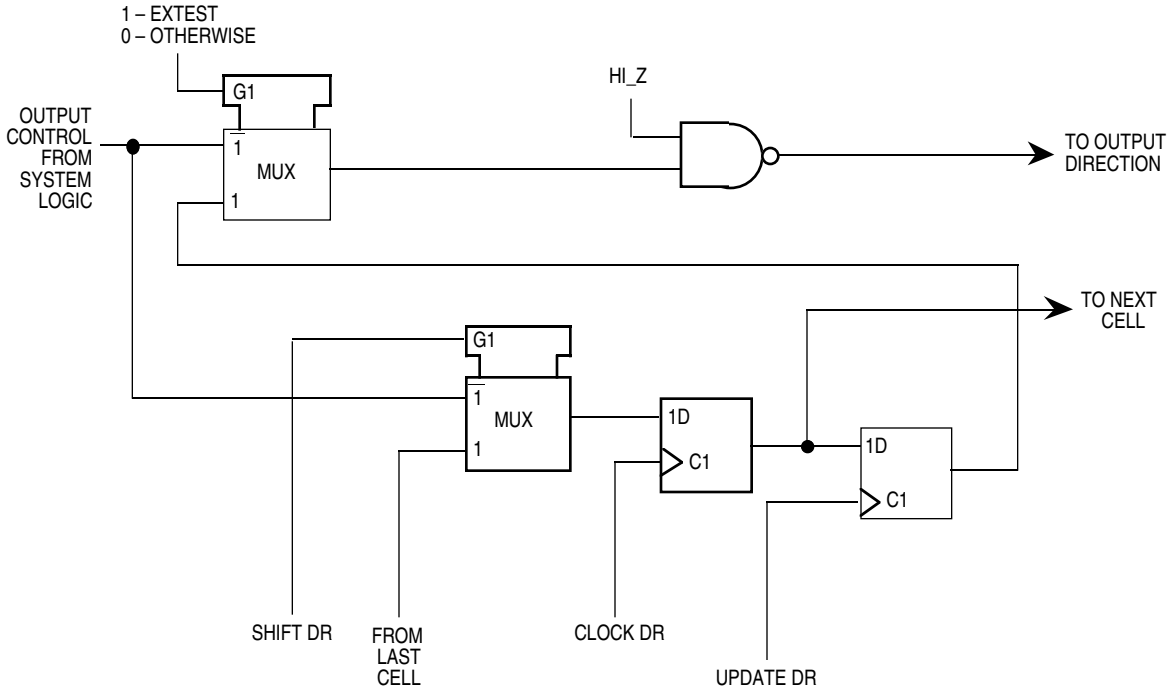


Figure 7-5. Control Cell (dicell)

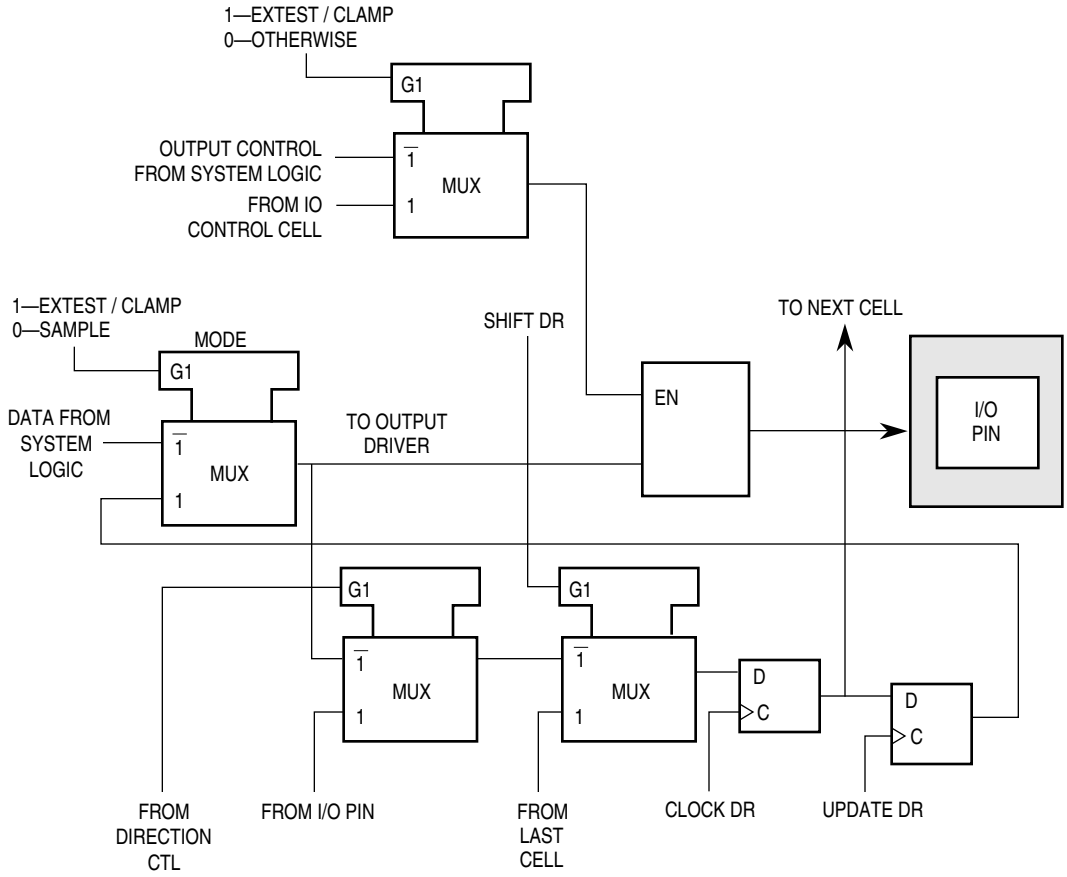


Figure 7-6. Bidirectional Cell (bicell)

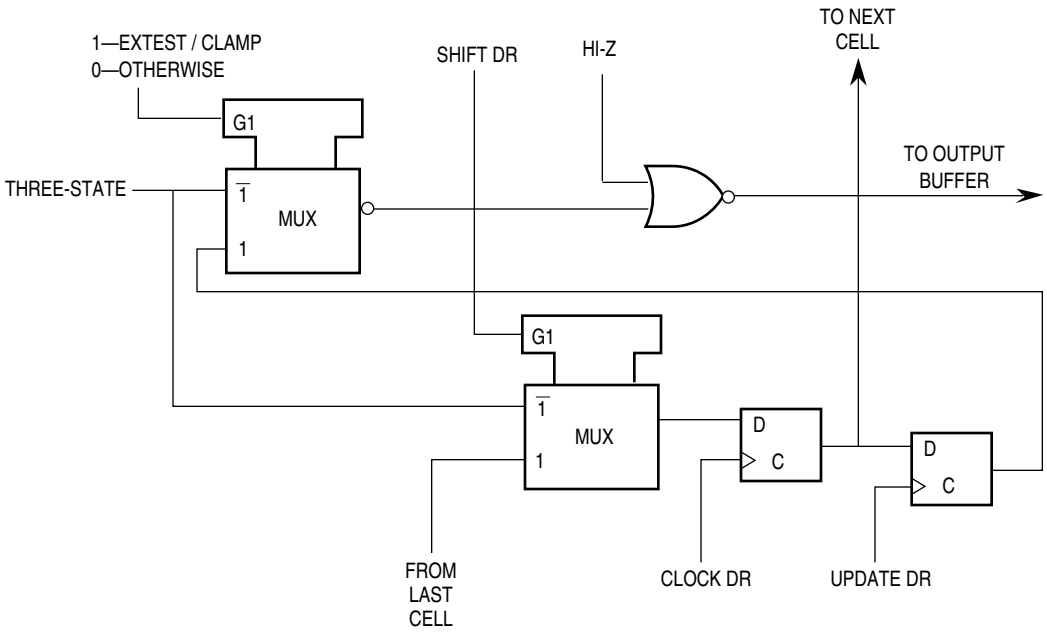


Figure 7-7. Output Enable Cell (encell)

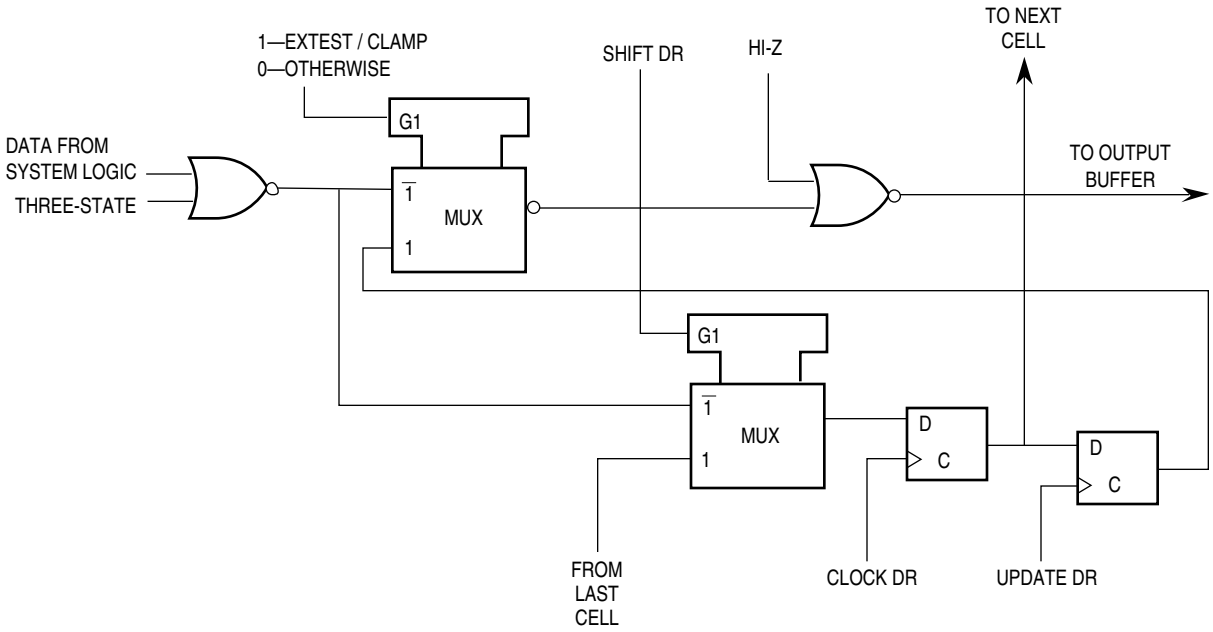


Figure 7-8. Output Enable Cell (encello)

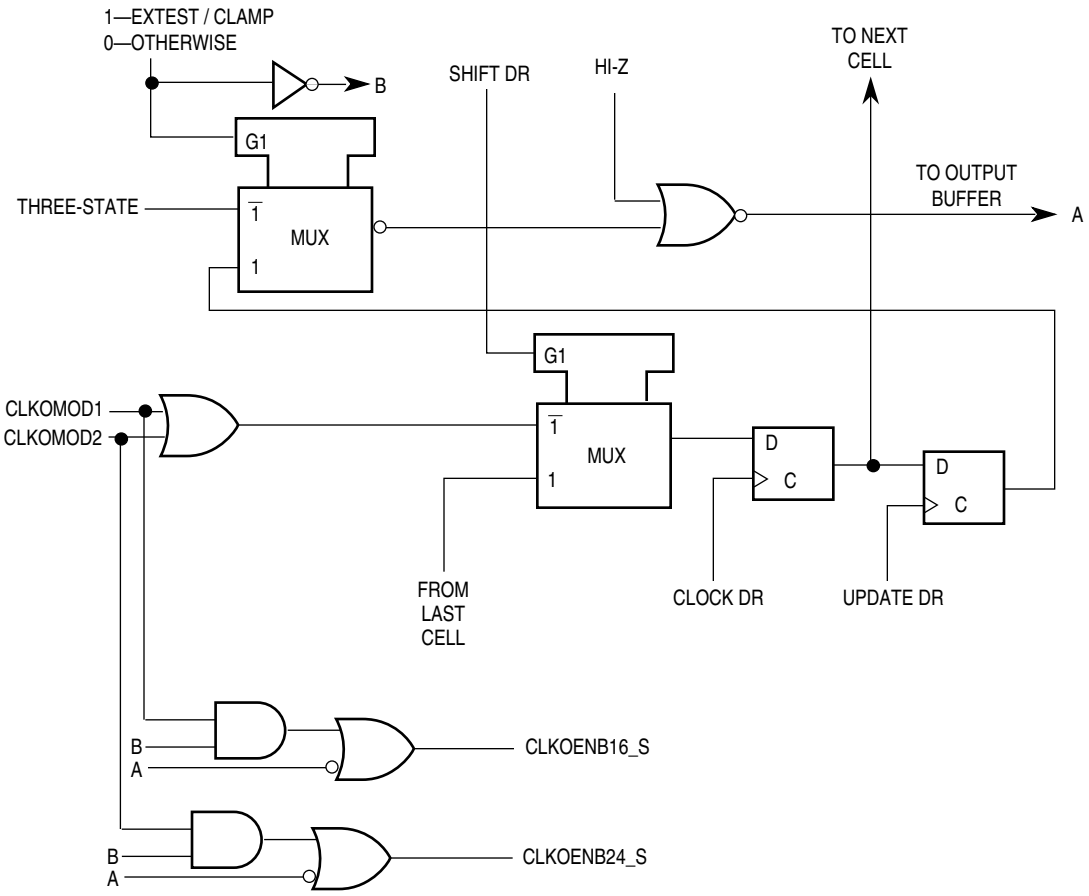
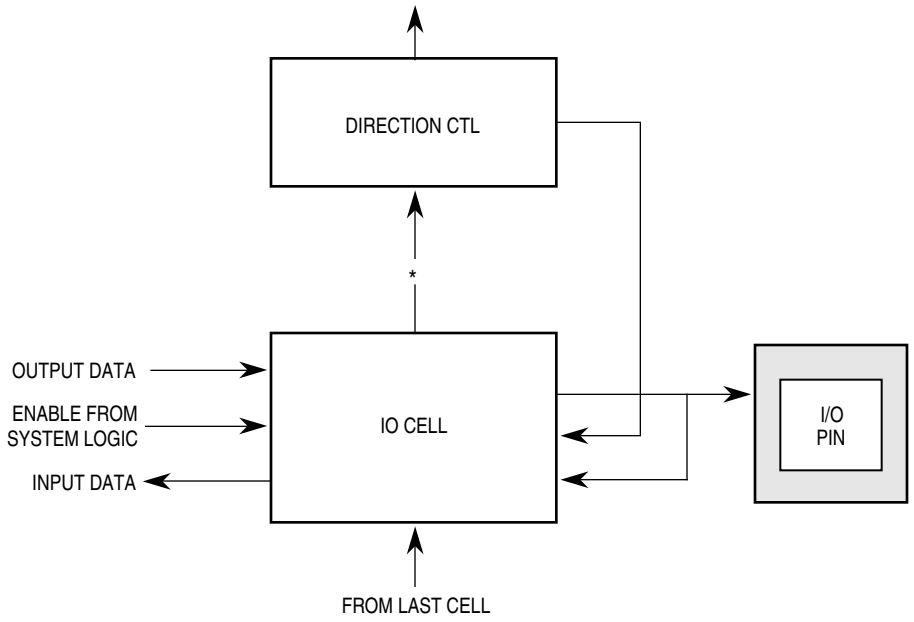


Figure 7-9. Output Enable Cell (clko_encell)



NOTE: More than one IO.Cell could be serially connected and controlled by a single IO.Ctl.

Figure 7-10. General Arrangement for Bidirectional Pins

7.4 INSTRUCTION REGISTER

The MC68EN302 IEEE 1149.1 implementation includes the three mandatory public instructions (EXTEST, SAMPLE/PRELOAD, and BYPASS), and also supports the CLAMP instruction. One additional public instruction (HI-Z) provides the capability for disabling all device output drivers. The MC68EN302 includes a 4-bit instruction register without parity. Data is transferred from the shift register to the parallel outputs during the update-IR controller state. The four bits used to decode the instructions are listed in Table 7-3.

Table 7-3. Instruction Decoding

Code				Instruction
IR[3]	IR[2]	IR[1]	IR[0]	
0	0	0	0	EXTEST
0	0	1	0	SAMPLE/PRELOAD
1	1	1	1	BYPASS
1	0	0	1	HI-Z
1	1	0	0	CLAMP
ALL OTHER CASES				BYPASS

The parallel output of the instruction register logic is reset to the equivalent to the BYPASS instruction. During the capture-IR controller state, the parallel inputs to the instruction shift register are loaded with 0001.

7.4.1 EXTEST

The external test (EXTEST) instruction selects the 163-bit boundary scan register.

By using the TAP, the register is capable of a) scanning user-defined values into the output buffers, b) capturing values presented to input pins, c) controlling the direction of bidirectional pins, and d) controlling the output drive of three-stateable output pins. For more details on the function and use of EXTEST, refer to the IEEE 1149.1 document.

7.4.2 SAMPLE/PRELOAD

The SAMPLE/PRELOAD instruction provides two separate functions. First, it provides a means to obtain a snapshot of system data and control signals. The snapshot occurs on the rising edge of TCK in the capture-DR controller state. The data can be observed by shifting it transparently through the boundary scan register.

NOTE

Since there is no internal synchronization between the scan chain clock (TCK) and the system clock (CLKO), the user must provide some form of external synchronization to achieve meaningful results.

The second function of SAMPLE/PRELOAD is to initialize the boundary scan register output cells prior to selection of EXTEST. This initialization ensures that known data will appear on the outputs when entering the EXTEST instruction.

7.4.3 BYPASS

The BYPASS instruction selects the single-bit bypass register as shown in Figure 7-11. This creates a shift register path from TDI to the bypass register and, finally, to TDO, circumventing the 163-bit boundary scan register. This instruction is used to enhance test efficiency when a component other than the MC68EN302 becomes the device under test.

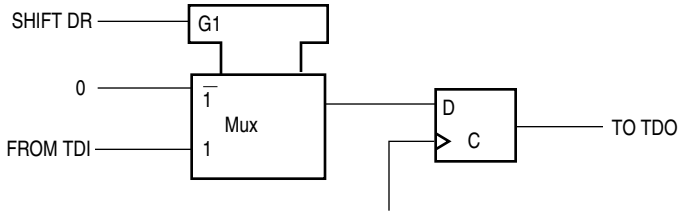


Figure 7-11. Bypass Register

When the bypass register is selected by the current instruction, the shift register stage is set to a logic zero on the rising edge of TCK in the capture-DR controller state. Therefore, the first bit to be shifted out after selecting the bypass register will always be a logic zero.

7.4.4 CLAMP

The CLAMP instruction selects the single-bit bypass register as shown in Figure 7-11, and the state of all signals driven from system output pins is completely defined by the data previously shifted into the boundary scan register (for example, using the SAMPLE/PRELOAD instruction).

7.4.5 HI-Z

The HI-Z instruction is provided as a manufacturer's optional public instruction to prevent having to backdrive the output pins during circuit-board testing. When HI-Z is invoked, all output drivers are turned off (i.e., high impedance). The instruction selects the bypass register.

NOTE

On the MC68EN302, the $\overline{\text{THREESTATE}}$ pin may also be used during system reset to perform the same function.

7.5 MC68EN302 RESTRICTIONS

The control afforded by the output enable signals using the boundary scan register and the EXTEST instruction requires a compatible circuit-board test environment to avoid device-destructive configurations. The user must avoid situations in which the MC68EN302 output drivers are enabled into actively driven networks.

7.6 NON-SCAN CHAIN OPERATION

In non-scan chain operation, there are two constraints. First, the TCK input does not include an internal pullup resistor and should not be left unconnected to preclude mid-level inputs. The second constraint is to ensure that the scan chain test logic is kept transparent to the system logic by forcing TAP into the test-logic-reset controller state. This is accomplished by asserting the $\overline{\text{TRST}}$ signal during system reset ($\overline{\text{RESET}}$ and $\overline{\text{HALT}}$ asserted) and leaving TMS unconnected or tied to VCC.

SECTION 8
MC68EN302 ELECTRICAL CHARACTERISTICS

8.1 POWER DISSIPATION

At 25 MHz, typical current will be 140 mA (with all circuitry active), max current will be tbd.

8.2 CHANGES TO EXISTING MC68302 TIMING SPECS

The timing for the MC68EN302 signals that are shared with the MC68302 are the same as specified in the MC68302 manual.

8.3 DRAM INTERFACE TIMING

Table 8-1. DRAM Interface Timing

NUM	CHARACTERISTIC	SYMBOL	20 MHZ		25 MHZ		UNIT
			MIN	MAX	MIN	MAX	
6A	CLKO Low to Column Address Valid		0	25	0	25	
9	CLKO High to $\overline{AS}$, $\overline{RASx}$ Asserted		3	25	3	20	ns
9A	CLKO High to $\overline{RASx}$ Deasserted		3	25	3	20	ns
400	$\overline{RASx}$ Asserted to Row Address Invalid		12	-	12	-	ns
401	$\overline{RASx}$ Asserted to Column Address Valid		15	-	15	-	ns
402	$\overline{RASx}$ Width Asserted (1)		85	-	75	-	ns
403	$\overline{RASx}$ Width Negated (2)		85	-	75	-	ns
404	$\overline{RASx}$ Asserted to $\overline{CAS}$ Asserted		35	-	35	-	ns
405	CLKO High to $\overline{CASx}$ Asserted		3	25	3	20	ns
405A	CLKO High to $\overline{CASx}$ Asserted (Refresh)		3	25	3	20	ns
406	CLKO High to $\overline{CASx}$ Negated		3	25	3	20	ns
407	Column Address Valid to $\overline{CASx}$ Asserted		15	-	15	-	ns
408	$\overline{CASx}$ Asserted to Column Address Negated		50	-	40	-	ns
409	$\overline{CASx}$ Asserted to $\overline{RASx}$ Negated		35	-	30	-	ns
410	$\overline{CASx}$ Width Asserted (1)		75	-	60	-	ns
411	$\overline{CASx}$ Width Negated (2)		75	-	60	-	ns
412	$\overline{CASx}$ Negated to Data, Parity-In invalid		0	-	0	-	ns
415	$\overline{DRAMRW}$ Low to $\overline{CASx}$ Asserted		30	-	25	-	ns
416	$\overline{CASx}$ Asserted to $\overline{DRAMRW}$ High		100	-	80	-	ns
417	Data-Out Valid to $\overline{CASx}$ Asserted		15	-	10	-	ns
417A	Parity-Out Valid to $\overline{CASx}$ Asserted		0	-	0	-	ns
418	$\overline{CAS}$ asserted to Data/Parity-Out Invalid (1)		100	-	80	-	ns
419	CLKO Low to $\overline{AMUX}$ Negated		3	15	3	15	ns
420	CLKO Low to $\overline{AMUX}$ Asserted		3	15	3	15	ns
421	$\overline{AMUX}$ High to $\overline{RASx}$ Asserted		50	-	40	-	ns
422	$\overline{RASx}$ Asserted to $\overline{AMUX}$ Low		10	-	10	-	ns
423	$\overline{AMUX}$ Low to $\overline{CAS}$ Asserted		15	-	15	-	ns
424	$\overline{CASx}$ Asserted to $\overline{AMUX}$ High		55	-	45	-	ns
23	CLKO Low to Data Out Valid		-	25	-	20	ns
23A	CLKO Low to Parity Out Valid		-	40	-	33	ns
27	Data-In to CLKO Low (Parity Disabled) (3)		6	-	5	-	ns
27A	Data/Parity-In to CLKO Low (Parity Enabled) (3)		16	-	12	-	ns
425	CLKO High to $\overline{PARITYE}$ Valid		-	15	-	15	ns

Any spec numbers shown in diagrams and not listed in the table are unchanged from the MC68302 User's Manual.

NOTES:
1. Width increases by clock period (Tcyc) for each wait state added.
2. Width increases by clock period (Tcyc) for each increase in P1–P0 ($\overline{RAS}$ precharge time).
3. Parity Enabled timing (spec 27A) only applies to bank(s) which have parity enabled.

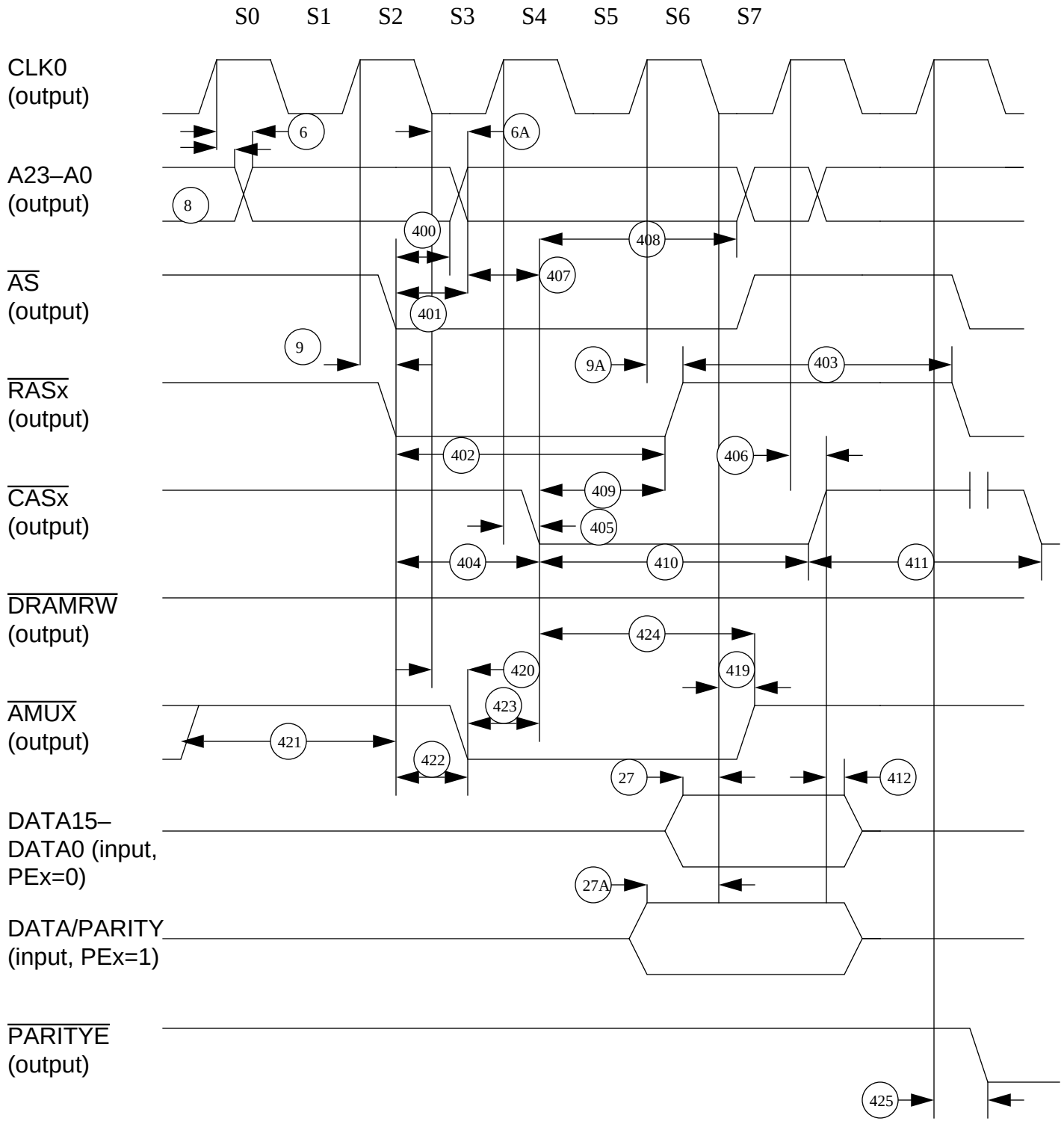


Figure 8-1. DRAM Read Cycle

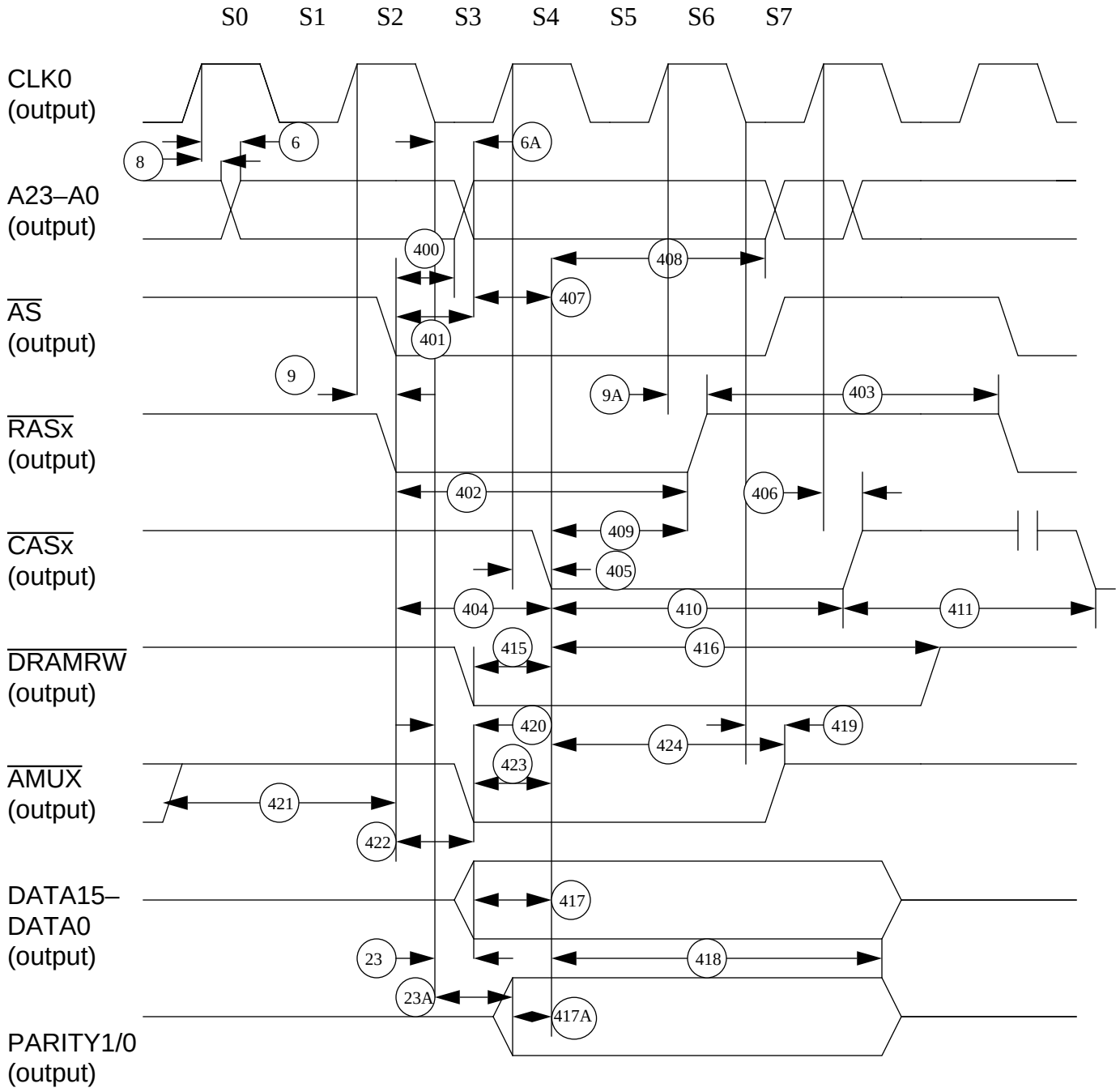


Figure 8-2. DRAM Write Cycle

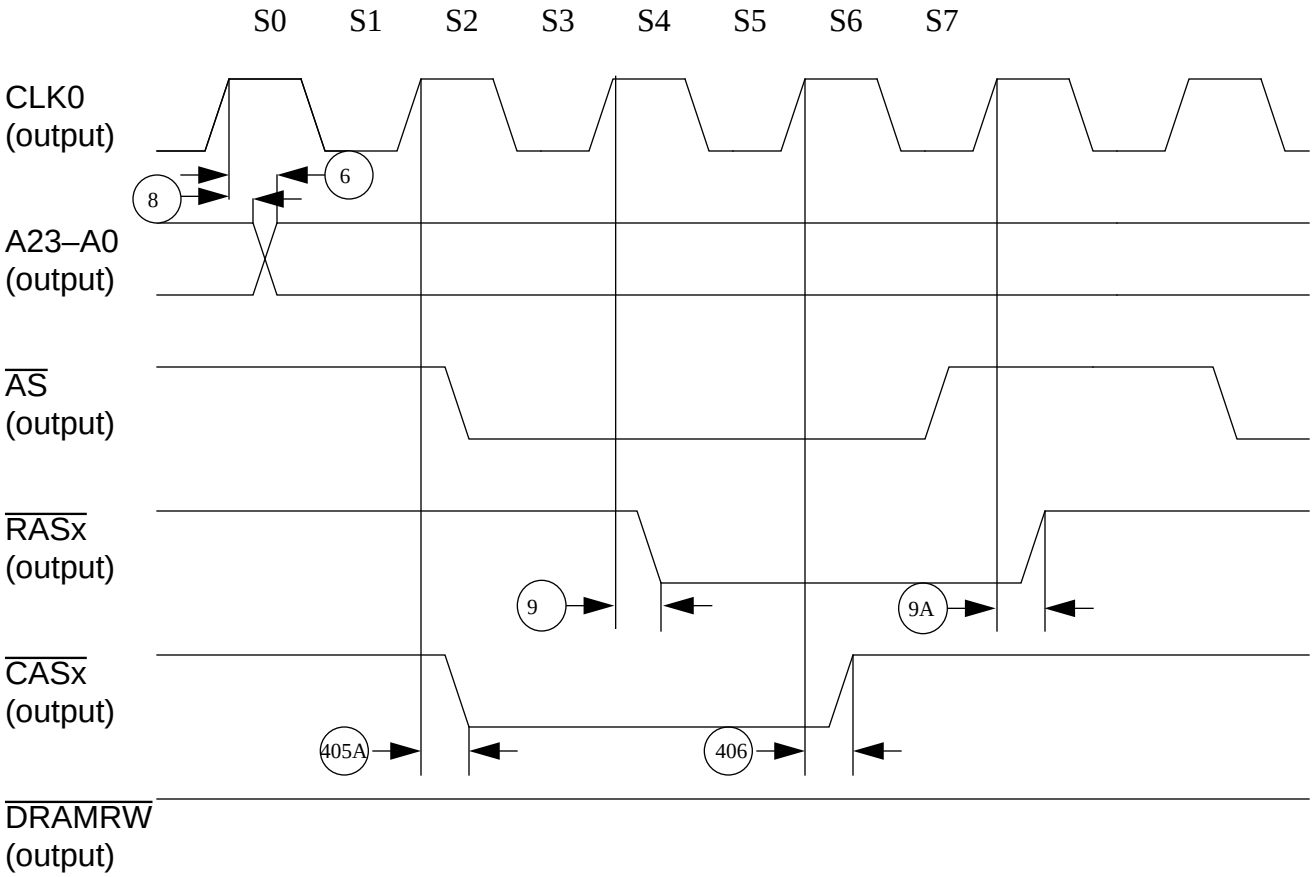


Figure 8-3. DRAM Refresh

8.4 ETHERNET TIMING

Table 8-2. Ethernet Timing

NUM	CHARACTERISTIC	SYMBOL	MIN	MAX	UNIT
520	CLSN width high		105		ns
521	RCLK Rise/Fall Time			15	ns
522	RCLK width low		40	60	ns
523	RCLK period		80	120	ns
524	RX, RENA Setup to RCLK rising edge		25		ns
525	RX Hold time from RCLK rising edge		0		ns
526	RENA Active Delay (from RCLK rising edge of the last data bit)		0		ns
527	RENA Width Low		105		ns
528	TCLK Rise/Fall Time			15	ns
529	TCLK width low		45	55	ns
530	TCLK clock period		99	101	ns
531	TCLK High to TXD, TENA Active Delay (1)		45	75	ns
532	TCLK High to TXD, TENA Inactive Delay (1)		45	75	ns

1. NOTES:
   1. TXD, TENA are actually driven by TCLK Low (falling) edge. Max delay from TCLK low to TXD, TENA change is 20 nsec.

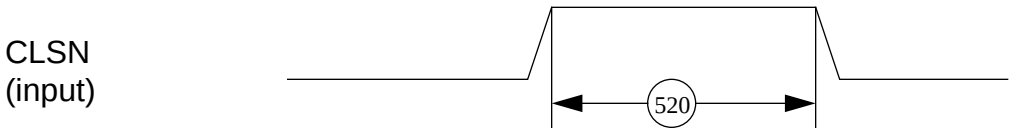


Figure 8-4. Ethernet Collision Timing

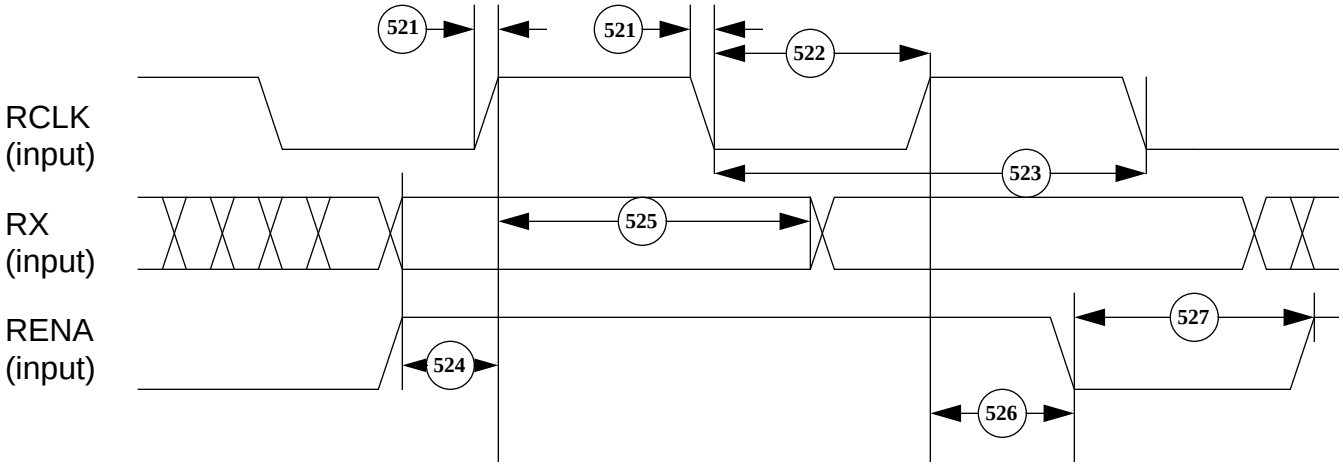


Figure 8-5. Ethernet Receive Timing

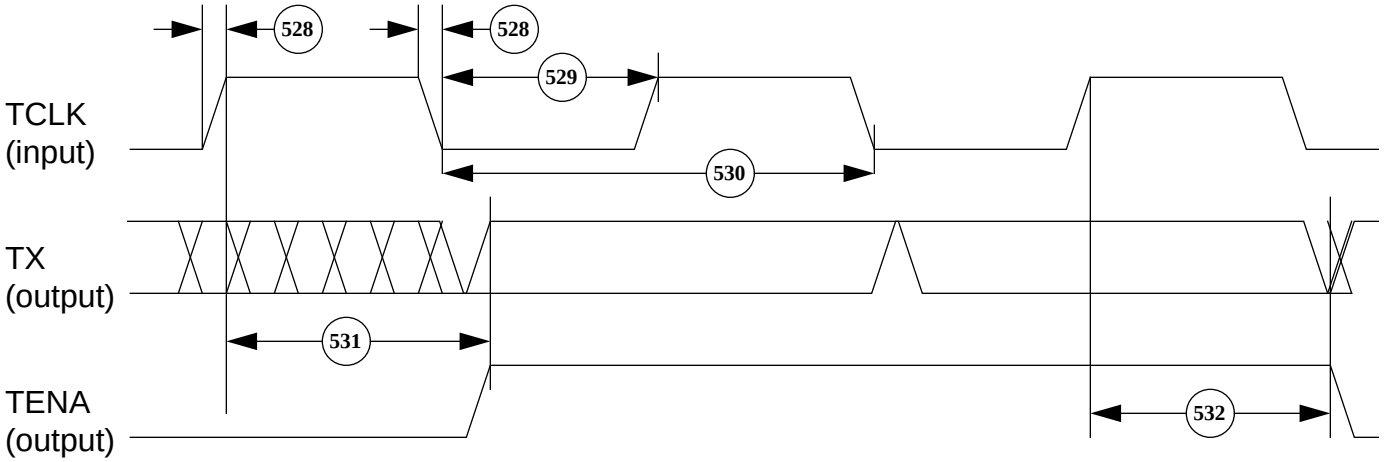


Figure 8-6. Ethernet Transmit Timing

8.5 JTAG INTERFACE TIMING

The TCK, $\overline{\text{TRST}}$, TMS, TDI, TDO are new signals added to the MC68EN302 that do not exist on the MC68302.

Num	Characteristic	Min	Max	Unit
	TCK Frequency of Operation	0	10	MHz
1	TCK Cycle Time in Crystal Mode	100	—	ns
2	TCK Clock Pulse Width Measured at 1.5 V	40	—	ns
3	TCK Rise and Fall Times	0	3	ns
6	Boundary Scan Input Data Setup Time	10	—	ns
7	Boundary Scan Input Data Hold Time	18	—	ns
8	TCK Low to Output Data Valid	0	30	ns
9	TCK Low to Output High Impedance	0	40	ns
10	TMS, TDI Data Setup Time	10	—	ns
11	TMS, TDI Data Hold Time	10	—	ns
12	TCK Low to TDO Data Valid	0	20	ns
13	TCK Low to TDO High Impedance	0	20	ns
14	$\overline{\text{TRST}}$ Assert Time	100	—	ns
15	$\overline{\text{TRST}}$ Setup Time to TCK Low	40	—	ns

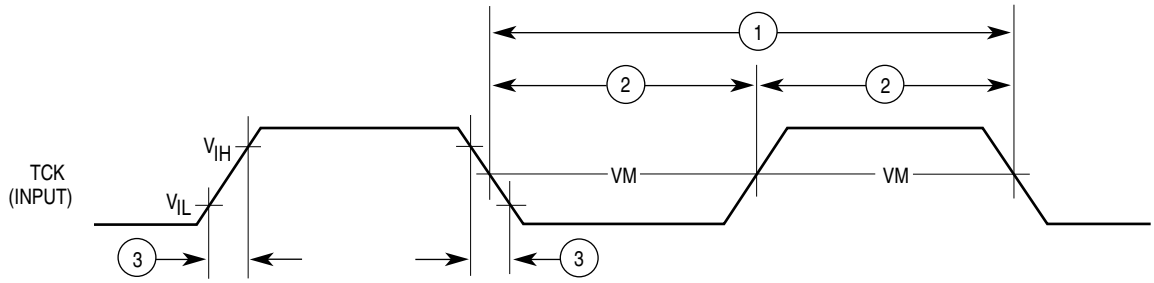


Figure 8-7. Test Clock Input Timing Diagram

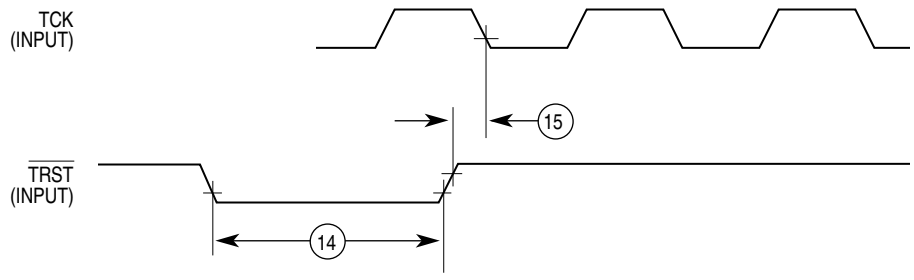


Figure 8-8. $\overline{\text{TRST}}$ Timing Diagram

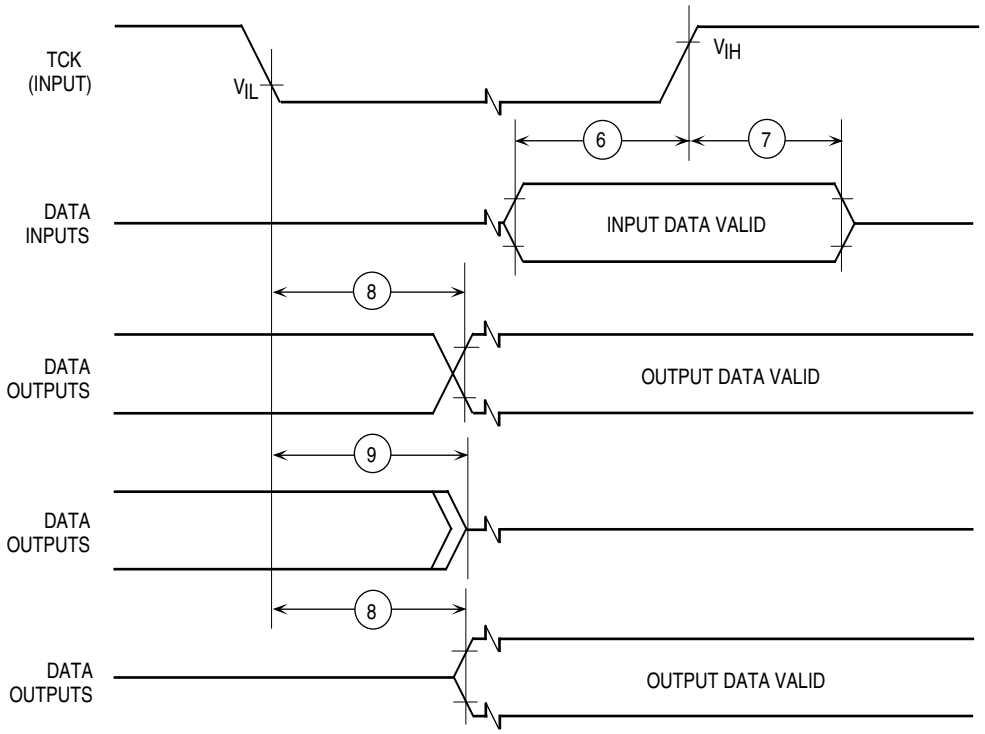


Figure 8-9. Boundary Scan (JTAG) Timing Diagram

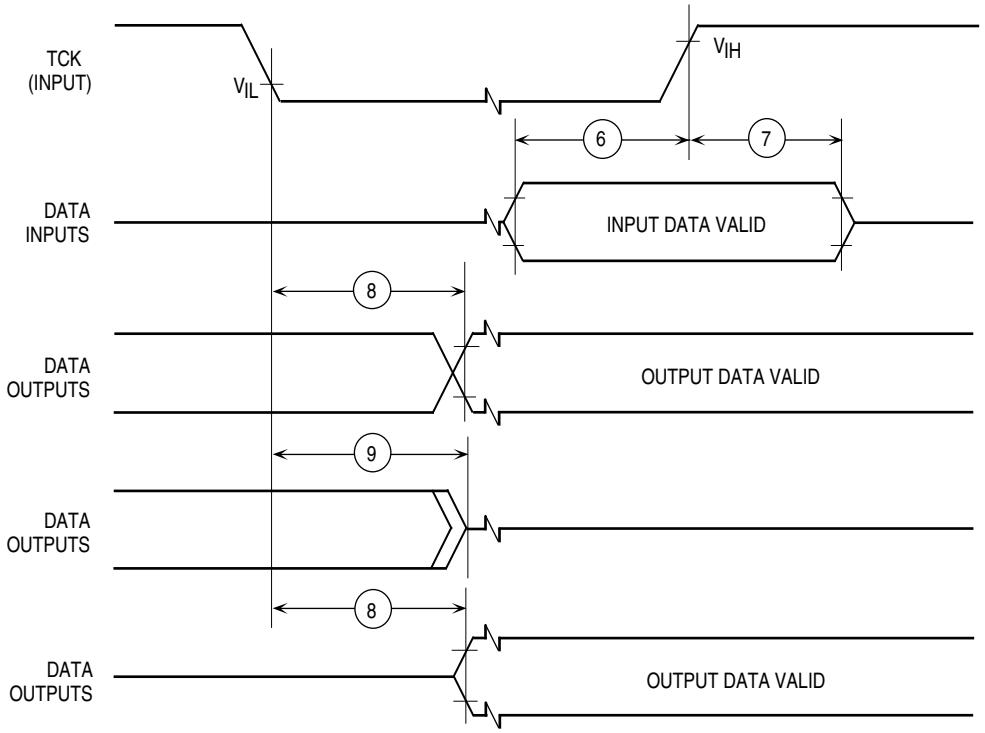


Figure 8-10. Test Access Port Timing Diagram

8.6 $\overline{OE}$, $\overline{WEL}$, $\overline{WEH}$ TIMING

These are new signals added to the MC68EN302 that do not exist on the MC68302.

8.6.1 $\overline{OE}$ Timing

During a read, the timing on the $\overline{OE}$ signal is similar to the MC68302 $\overline{UDS}$, $\overline{LDS}$ lines (assertion and deassertion reference the same clock edges as $\overline{UDS}$, $\overline{LDS}$). Reference Figure 6-2 (Read Cycle Timing Diagram) in the MC68302 User's Manual. The following MC68302 specifications define $\overline{OE}$ timing:

MC68302 Spec 9 - CLK0 High to $\overline{OE}$ Asserted

MC68302 Spec 12 - CLK0 Low to $\overline{OE}$ Negated

8.6.2 $\overline{WEL}$, $\overline{WEH}$ Timing

During a write, the timing on the $\overline{WEL}$, $\overline{WEH}$ signals is similar to the MC68302 $\overline{UDS}$, $\overline{LDS}$ lines (assertion and deassertion reference the same clock edges as $\overline{UDS}$, $\overline{LDS}$). Reference Figure 6-3 (Write Cycle Timing Diagram) in the MC68302 User's Manual. The following MC68302 specifications define $\overline{WEL}$, $\overline{WEH}$ timing:

MC68302 Spec 9 - CLK0 High to $\overline{WEL}$, $\overline{WEH}$ Asserted

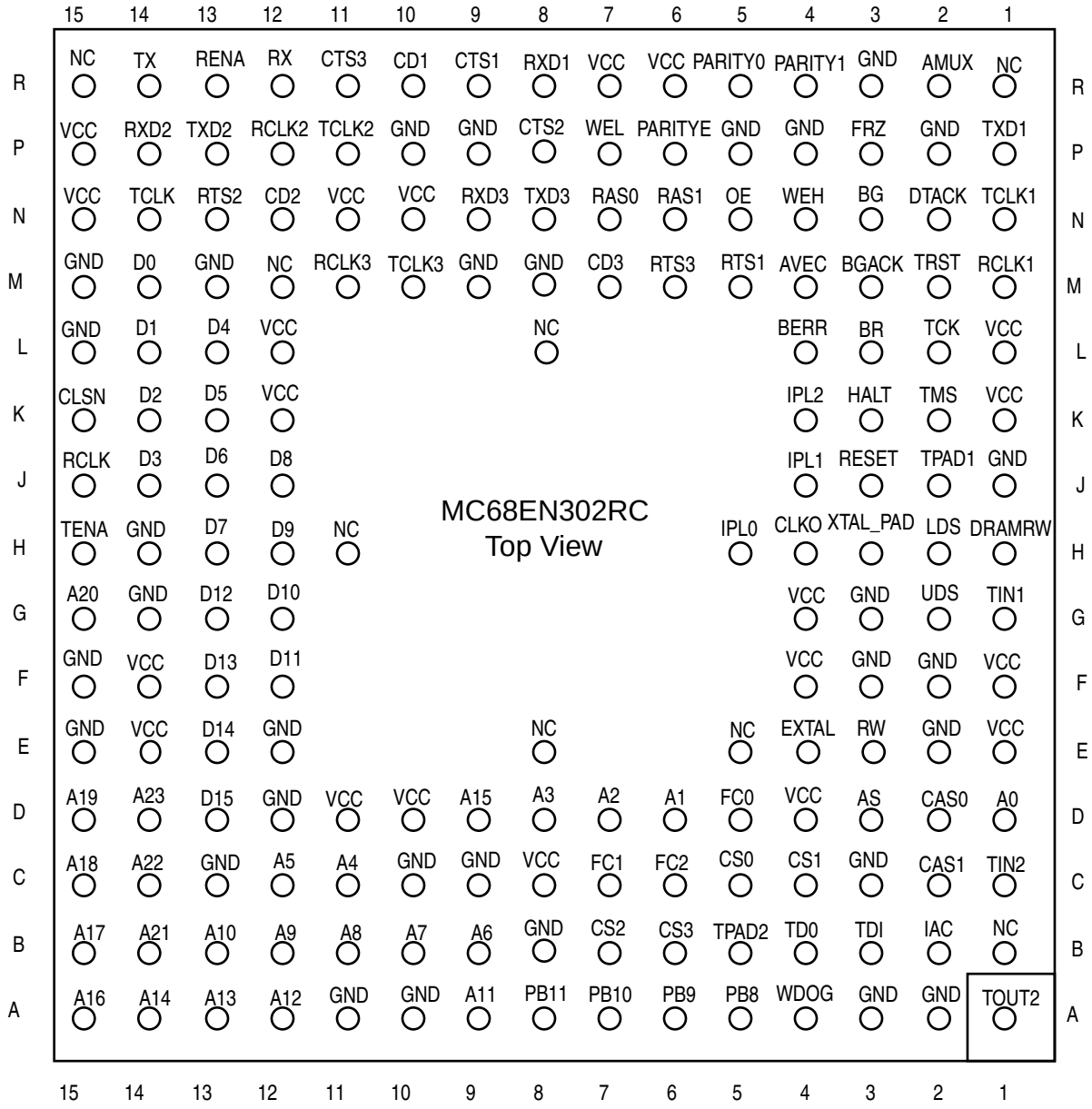
MC68302 Spec 12 - CLK0 Low to $\overline{WEL}$, $\overline{WEH}$ Negated

SECTION 9
ORDERING AND MECHANICAL INFORMATION

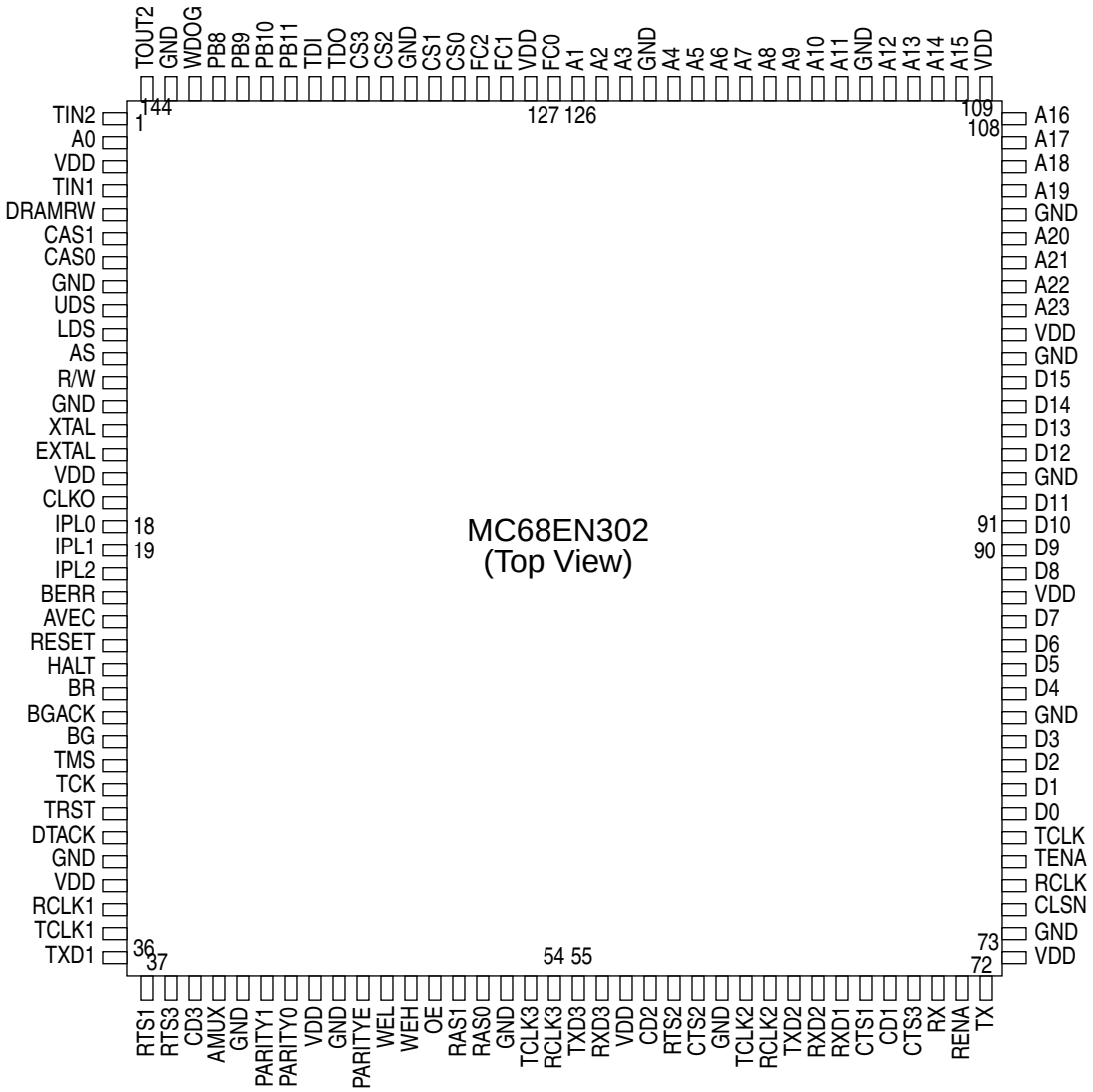
This section contains the ordering information, pin assignments, and package dimensions for the MC68EN302.

9.1 PIN ASSIGNMENT

9.1.1 Pin Grid Array (PGA)

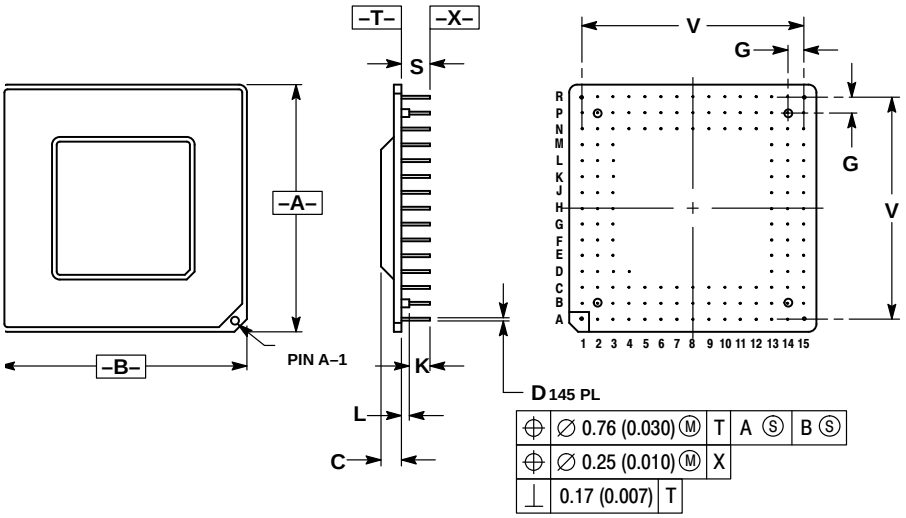


9.1.2 144 Thin Quad Flat Pack (TQFP)



9.2 PACKAGE DIMENSIONS

9.2.1 Pin Grid Array (PGA)



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION D INCLUDES LEAD FINISH.

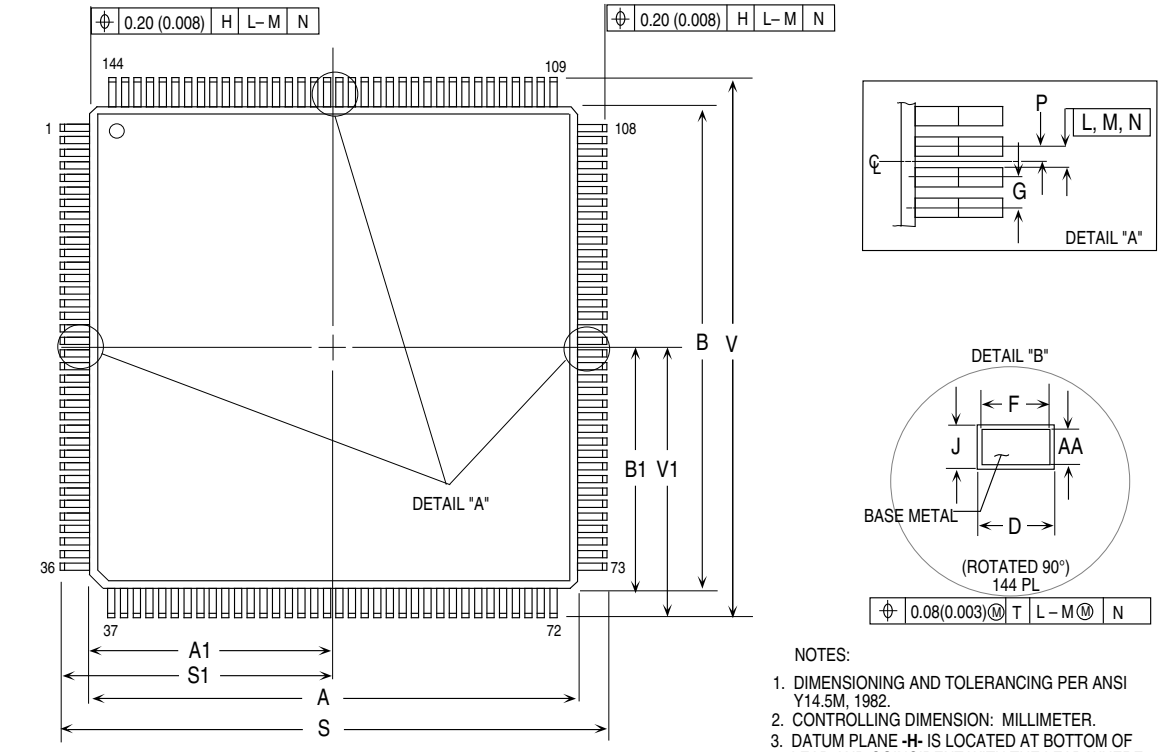
	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	1.550	1.570	39.37	39.88
B	1.550	1.570	39.37	39.88
C	0.115	0.135	2.92	3.43
D	0.017	0.022	0.43	0.55
G	0.100 BSC		2.54 BSC	
K	0.120	0.140	3.05	3.55
L	0.040	0.060	1.02	1.52
S	0.170	0.195	4.32	4.95
V	1.400 BSC		35.56 BSC	

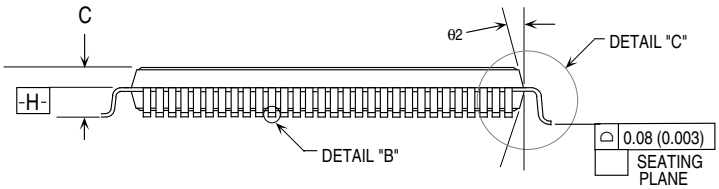
CASE 768E-01
ISSUE O

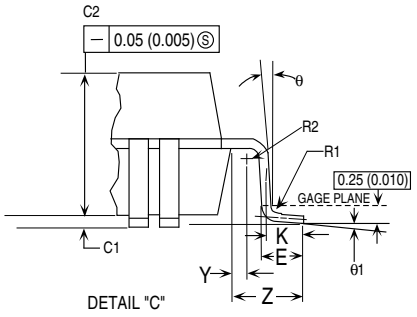
DATE 04/04/94

9.2.2 144 Thin Quad Flat Pack (TQFP)

CASE 918-02
144 TQFP







NOTES:

1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DATUM PLANE **-H-** IS LOCATED AT BOTTOM OF LEAD AND COINCIDENT WITH THE LEAD WHERE THE LEAD EXITS THE PLASTIC BODY AT THE BOTTOM OF THE PARTING LINE.
4. DATUMS **-L-**, **-M-**, AND **-N-** TO BE DETERMINED AT DATUM PLANE **-H-**.
5. DIMENSIONS S AND V TO BE DETERMINED AT SEATING PLANE **-T-**.
6. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION. ALLOWABLE PROTRUSION IS 0.25 (0.010 PER SIDE. DIMENSIONS A AND B DO NOT INCLUDE MOLD MISMATCH AND ARE DETERMINED AT DATUM LINE **-H-**.
7. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL NOT CAUSE THE D DIMENSION TO EXCEED 0.35 (0.014).

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	20.00 BSC		0.790 BSC	
A1	10.00 BSC		0.394 BSC	
B	20.00 BSC		0.790 BSC	
B1	10.00 BSC		0.394 BSC	
C	1.40	1.60	0.055	0.063
C1	0.05	0.15	0.002	0.006
C2	1.35	1.45	0.053	0.057
D	0.17	0.27	0.007	0.011
E	0.45	0.75	0.018	0.030
F	0.17	0.23	0.007	0.009
G	0.50 BSC.		0.20 BSC.	
J	0.09	0.20	0.004	0.008
K	0.50 REF		0.020 REF	
P	0.25 BSC		0.010 BSC	
R1	0.13	0.20	0.005	0.008
R2	0.13	0.20	0.005	0.008
S	22.00 BSC		0.866 BSC	
S1	11.00 BSC		0.433 BSC	
V	22.00 BSC		0.866 BSC	
V1	11.00 BSC		0.433 BSC	
Y	0.25 REF		0.010 REF	
Z	1.00 REF		0.039 REF	
AA	0.09	0.16	0.004	0.006
θ	0°		0°	
θ1	0°	7°	0°	7°
θ2	11°	13°	11°	13°

9.3 STANDARD ORDERING INFORMATION

PACKAGE TYPE	FREQUENCY (MHZ)	TEMPERATURE	ORDER NUMBER
Thin Quad Flat Pack (TQFP)	20	$0^{o}C$-$70^{o}C$	MC68EN302PV20
Thin Quad Flat Pack (TQFP)	25	$0^{o}C$-$70^{o}C$	MC68EN302PV25
Pin Grid Array (PGA) (for development only, not available in production quantity)	25	$0^{o}C$-$70^{o}C$	PC68EN302RC25

MC68302 : Integrated Multi-Protocol Processor

The MC68302 is a versatile one-chip processor that incorporates the main building blocks needed for the design of a wide variety of networking and communications products.

The MC68302 was the first device to offer the benefits of a closely coupled, industry-standard, MC68000/MC68008 microprocessor core and a flexible communications architecture. This multi-channel communications device may be configured to support a number of popular industry-standard interfaces, including those for the Integrated Services Digital Network (ISDN) basic rate and terminal adapter applications. Through a combination of architectural and programmable features, concurrent operation of different protocols is easily achieved using the MC68302. Data concentrators, modems, line cards, bridges, and gateways are examples of other suitable applications for this versatile device.

The MC68302 is an HCMOS device consisting of an MC68000/MC68008 microprocessor core, a system integration block (SIB), and a communications processor (CP).

This device is still recommended for new designs.

Page Contents

- Features
- Parametrics
- Documentation
- Design Tools
- Orderable Parts

Other Info

- FAQs
- Literature Services
- 68K/ColdFire®
- Microcontrollers
- Networking
- PowerQUICC™ Communication Processors
- 3rd Party Design Help

Link Product Picture

Link Block Diagram

MC68302 Features

Product Highlights

- MC68000/MC68008 Microprocessor Core
- Efficient architecture involves a separate RISC processor for handling communications
- Three Serial Communications Controllers (SCCs)
- Support for HDLC/SDLC, Bisync, UART, DDCMP, and Totally Transparent protocols.
- Two Serial Management Controllers (SMCs) for IDL and GCI Channel.
- Available at 16, 20, 25, and 33 MHz in three different Thin Quad Flat Pack Packages.
- Strong 3rd Party tools support.

Typical Applications

- ISDN equipment
- Data Concentrators
- Modems

- Line Cards
- Network Bridges
- Gateways

Technical Specifications

- MC68000/MC68008 Microprocessor Core (May be disabled to use the IMP as a peripheral)
- SIB Including:
  - Independent Direct Memory Access (IDMA) Controller
  - Interrupt controller with two modes of operation
  - Parallel I/O ports, some with interrupt capability
  - On-Chip 1152-bytes of dual-port RAM
  - Three timers, with a software watchdog timer
  - Four programmable chip-select lines with wait-state logic
  - Programmable address mapping of dual-port RAM and IMP registers

- On-Chip clock generator with an output clock signal
- System Control
  - Bus arbitration logic with low interrupt latency support
  - System control register
  - Hardware watchdog for monitoring bus activity
  - Low power (Standby) modes
  - Disable CPU logic (M68000)
  - Freeze control for debugging selected on-chip peripherals
  - DRAM refresh controller

- CP Including:
  - Main controller (RISC Processor)
  - Three full-duplex Serial Communication/Controllers with the following protocols:
    - HDLC/SDLC
    - Bisync
    - UART
    - DDCMP
    - Totally Transparent
    - V.110
  - Six serial DMA channels dedicated to the three SOCs
  - Capability to send/receive up to eight buffers/frames without M68000 core intervention
  - Flexible physical interface accessible by SCCs for Inter-chip Digital Link (IDL), General Circuit Interface (GCI).
  - Pulse Code Modulation (PCM), and Non-multiplexed Serial Interface (NMSI) Operation.
  - Serial Communication Port (SCP) for synchronous communication.
  - Two Serial Management Controllers (SMCs) for IDL and GCI Channel.

[top]

MC68302 Parametrics

Description	Processor Speed (MHz)	Bus Interface (Bits)	Memory	Voltage (V)	Package	Performance

Integrated Multiprotocol Processor (IMP)	16, 20	32 Data	No On-Chip MMU	5	100 TQFP, 144 TQFP	1.6 (MC68000 Core)

[top]

MC68302 Documentation

Application Note

ID	Name	Format	Size K	Rev #	Date Last Modified	Order	Availability
AN2015/D	Configuring the Chip Selects on the MC68302	pdf	62	0	6/17/1991		
AN2016/D	DRAM Control with the MC68302	pdf	266	0	3/21/1991		
AN2018/D	MC68302 Getting Started with Interrupts on the MC68302	pdf	30	0	3/14/1990		
AN2019/D	MC68302 Design Concept - Expanding Interrupts on the MC68302	pdf	13	0	7/09/1990		
AN2020/D	MC68302 Adapting a WAN Controller to a LAN Environment	pdf	515	0	1/01/1999		
AN2021/D	MC68302 Interfacing the MC68020 to a Slave MC68302	pdf	141	0	2/02/1993		
AN2023/D	MC68302 EKB Applications - Power Measurements on the MC68302	pdf	45	1	7/26/1995		
AN2024/D	MC68302 Software Performance	pdf	42	0	6/18/1990		
AN2026/D	MC68302 Evaluating EDX on the ADS302	pdf	167	0	2/27/1991		
AN2027/D	MC68302 Using the 302 Communications Peripheral for PowerPC Microprocessors	pdf	325	0	1/01/1994		
AN2028/D	MC68302 Design Advisory #1 - MC68SC302 Passive ISDN Protocol Engine	pdf	5	0	3/31/1997		
AN2049/D	MC68302, MC68360, and MPC860 Characteristics and Design Notes for Crystal Feedback Oscillators	pdf	29	0	1/01/1998		

Brochure

ID	Name	Format	Size K	Rev #	Date Last Modified	Order	Availability
8-16BITPAK/D	8-16 Bit Microcontrollers Product Portfolio	html	1	0	10/15/2002		
CFPITCHPAK/D	68K ColdFire Product Portfolio	html	1	0	10/14/2002		

Data Sheets

ID	Name	Format	Size K	Rev #	Date Last Modified	Order	Availability
MC68EN302/D	MC68EN302 Integrated Multiprotocol Processor w/ Ethernet Product Brief	pdf	55	0	1/01/1995		

MC68LC302/D	MC68LC302 Low Cost Integrated Multiprotocol Processor Product Brief	pdf	43	0	1/01/1995	
MC68PM302/D	MC68PM302 Integrated Multiprotocol Processor w/ PCMCIA Product Brief	pdf	636	0	1/01/1995	
MC68QH302/D	MC68QH302 Quad HDLC Integrated Multiprotocol Processor Tech. Summary Product Brief	pdf	21	0	11/01/1997	
MC68SC302P/D	MC68SC302 Passive ISDN Protocol Engine Product Brief	pdf	46	0	1/01/1996	

Engineering Bulletin

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
EB382/D	MC68302FC 132-Lead PQFP	pdf	31	0	3/30/2001	

Errata

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
MC68302DEC1/D	M68302 Errata Revision C.1 (Masks 0F26E, 1F26E)	pdf	33	C.1	12/10/1998	-
MC68EN302DEA1/D	M68302 Device Errata - 68EN302 Integrated Multiprotocol Processor Device	pdf	12	A.1	2/20/1997	-
MC68EN302DEB/D	M68302 Device Errata - 68EN302 Integrated Multiprotocol Processor With Ethernet Device	pdf	27	B	4/15/1999	-
MC68LC302DEB/D	M68302 Device Errata - 68LC302 Integrated Multiprotocol Processor Devices	pdf	7	B	8/02/1996	-
MC68LC302DEC/D	M68302 Device Errata - MC68LC302 Integrated Multiprotocol Processor Devices	pdf	27	C	2/09/1999	-
MC68PM302DEB1/D	M68302 Device Errata - 68PM302 Integrated Multiprotocol Processor Devices	pdf	11	B.1	10/28/1996	-
MC68PM302DEC/D	M68302 Revision C Changes - 68PM302 Integrated Multiprotocol Processor Devices	pdf	6	C	2/27/1997	-
MC68QH302DEC1/D	M68302 Device Errata - MC68QH302 Chip Errata	pdf	9	C.1	8/05/1998	-
MC68SC302DEA1/D	M68302 Device Errata - XC68SC302	pdf	6	A.1	2/18/1998	-

Fact Sheets

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
MC68302FACT/D	M68302 Integrated Multiprotocol Processor	pdf	30	0	1/01/1999	

Miscellaneous

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
M68302FADS/D	M68302 Family Application Development System	pdf	27	0	1/01/1995	
MC68302ABSPEC/D	MC68302 Auto Baud Support Package Specifications	pdf	60	2	5/01/1994	
MC68302DEB/D	MC68302 DEB	pdf	9	B	8/06/2001	-
MC68302TIMING/D	MC68302 33 MHz AC Timing Specification	pdf	31	-	8/07/2001	-
MC68EN302SCH/D	MC68302 Schematic for EN302	pdf	281	0	2/06/1996	-
MC68PM302DEC1/D	M68302 Device Errata - 68PM302 Integrated Multiprotocol Processor Devices	pdf	9	C.1	2/27/1997	-
MC68QH302SUPL/D	MC68QH302 Supplement to the MC68302 Integrated Multiprotocol Processor User's Manual	pdf	160	0	10/01/1997	

Packages & Pinouts

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
MC68302.OLB	MC68302 TQFP Part Symbol (OrCAD Capture 7.1+)	zip	3	0	10/31/1997	-
MC68302TQFP	MC68302 Pin Assignment (TQFP)	pdf	12	0	12/16/1999	-
MC68EN302.OLB	MC68EN302 Part Symbol (OrCAD Capture 7.1+)	zip	3	0	11/06/1997	-
MC68EN302TQFP	MC68EN302 Pin Assignment (TQFP)	pdf	12	0	12/16/1999	-
PBGAPRES	PBGA Customer Tutorial	pdf	2947	0	8/17/2000	-

Product Change Notices

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
PCN5656/D	MC68EN302 Capacity Expansion-PCN5656	pdf	20	-	5/25/2000	-
PCN5787/D	MC68EN302 Fab Transfer to TSC Addendum-PCN5787	pdf	21	-	8/01/2000	-
PCN5792/D	MC68LC302 Fab Transfer to TSC-PCN5792	pdf	20	-	8/07/2000	-
PCN7917	20X20 LQFP ASSY MOVE FROM SHC TO KLM	htm	24	0	8/19/2002	-

Reference Manual

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
MC68302UM/D	MC68302 Integrated Multiprotocol Processor User's Manual	pdf	1815	3	12/30/1995	-

MC68302UMAD3/D	Errata to MC68302 Integrated Multiprotocol Processor Users Manual	pdf	80	3	3/19/1992	
MC68302UMAD4/D	Errata to MC68302 Integrated Multiprotocol Processor Users Manual	pdf	32	4	1/01/1998	
MC68302UMADD33V/D	Addendum to MC68302 Integrated Multiprotocol Processor User Manual (3.3 V Specifications)	pdf	191	0.1	7/19/2000	-
MC68EN302RM/D	MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual	pdf	686	0	1/01/2000	-
MC68EN302RMAD/D	Errata to MC68EN302 Integrated Multiprotocol Processor with Ethernet Reference Manual Rev 1	pdf	27	B	11/15/1995	-
MC68LC302RM/AD	MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual	pdf	4958	0	1/01/1995	
MC68LC302RMAD/D	MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual ERRATA	pdf	15	1	2/21/1997	-
MC68LC302UM/D	MC68LC302 Low Power Integrated Multiprotocol Processor Reference Manual	pdf	766	0	8/06/2001	
MC68PM302RM/D	MC68PM302 Integrated Multiprotocol Processor with PCMCIA Reference Manual	pdf	1126	0	1/01/1995	-
MC68PM302RMAD/D	Errata to MC68PM302 Integrated Multiprotocol Processor with PCMCIA Reference Manual	pdf	25	1	2/11/1997	
MC68SC302RM /D	MC68SC302 Passive ISDN Protocol Engine User's Manual	pdf	521	0	1/01/1997	-
MC68SC302UMAD/AD	MC68SC302 Errata to MC68SC302 Users Manual	pdf	30	0	7/15/1998	-

Reports or Presentations

ID	Name	Format	Size K	Rev #	Date Last Modified	Order	Availability
ORDPARTS	Codec, Communication Processor, and ISDN Orderable Parts	pdf	61	-	7/30/2002		-

Selector Guide

ID	Name	Format	Size K	Rev #	Date Last Modified	Order	Availability
SG1006/D	Microcontrollers SPS Sales Guide	pdf	600	0	9/26/2002		
SG1007/D	Network and Communications Processors Sales Guide	pdf	161	0	9/26/2002		
SG1011/D	Software and Development Tools Sales Guide	pdf	259	1	9/26/2002		

Users Guide

ID	Name	Format	Size K	Rev #	Date Last Modified	Order Availability
MC68302FADSUM/D	MC68302FADS User's Manual	pdf	863	0	10/17/1995	
MC68302PIAFSUM/D	MC68302 PIAFS Microcode for MC68302(IMP) Family User's Manual	pdf	20	1	1/01/1997	-
MC68EN302ADPUM/D	MC68EN302 Adapter User's Manual	pdf	193	0	1/01/1999	-
MC68SC302ADSUM/D	MC68SC302ADS Application Development System User's Manual	pdf	862	1	1/01/1996	-

[top]

Design Tools

BSDL Files

ID	Name	Vendor ID	Format	Size K	Rev #
MC68EN302BSDL	MC68EN302 BSDL for JTAG Logic Ver. 0.1 (02/29/1996)	MOTOROLA	txt	20	0.1

Drivers

ID	Name	Vendor ID	Format	Size K	Rev #
MC68302DRV01	MC68302 Configuration and Driver Files (12/04/1997)	MOTOROLA	html	1	-

Microcode

ID	Name	Vendor ID	Format	Size K	Rev #
MC68302MC1	MC68302 Autobaud 2.0 Microcode - M68302SWAUT (02/22/99)	MOTOROLA	zip	5	-
MC68302MC2	MC68302 Centronics Microcode - M68302SWCEN (02/22/99)	MOTOROLA	zip	1	-
MC68302MC3	MC68302 Profibus Microcode - M68302SWPRO (02/22/99)	MOTOROLA	zip	1	-
MC68302MC4	MC68302 SS7 Microcode - M68302SWSS7 (03/06/2002)	MOTOROLA	zip	104	1.1
MC68302MC5	MC68302 V 110 Microcode - M68302SWV110 (02/22/99)	MOTOROLA	zip	5	-
MC68302MC6	MC68302 PIAFS Microcode - M68302SWPIA (01/26/00)	MOTOROLA	srx	1	-
MC68QH302MC1	Autobaud Microcode for the MC68QH302 (08/28/98)	MOTOROLA	zip	55	-

Reference Designs

ID	Name	Vendor ID	Format	Size K	Rev #
MC68QH302RD1	Firefly 2B - ISDN Active Terminal Adapter Reference Design Using MC68QH302 (Preliminary) (01/14/1997)	MOTOROLA	pdf	31	-
MC68QH302RD2	Firefly 2B - Schematics for ISDN Active Terminal Adapter Reference Design using MC68QH302 (Preliminary) (01/14/1997)	MOTOROLA	pdf	463	-
MC68SC302RD1	MC68SC302 Passive ISA U-Interface Reference Design BOM (01/16/1997)	MOTOROLA	txt	6	-
MC68SC302RD2	MC68SC302 Passive ISDN Terminal Adapter Design (Gerber) (06/27/1997)	MOTOROLA	zip	109	-
MC68SC302RD3	MC68SC302 Passive ISDN Terminal Adapter Design (Tango Pro CAD) (01/07/1997)	MOTOROLA	zip	288	-
MC68SC302RD4	MC68SC302 Passive U ISDN Card Reference Design (12/06/1996)	MOTOROLA	pdf	546	-
MC68SC302RD5	MC68SC302 PNP EEPROM Maker Script and Example Files (for Sun in PERL) (01/08/1997)	MOTOROLA	tar	160	-
MC68SC302RD6	MC68SC302 S/T-Interface Passive Terminal Adapter Reference Design (01/08/1997)	MOTOROLA	pdf	282	-

Schematics

ID	Name	Vendor ID	Format	Size K	Rev #
M68EN302FADSSCH	M68EN302 FADS Adapter (ENA) Schematics (02/06/1996)	MOTOROLA	pdf	281	-
M68XX302FADSSCH	M68xx302 FADS Schematics (12/04/1996)	MOTOROLA	pdf	630	-

Software

ID	Name	Vendor ID	Format	Size K	Rev #
M68XX302FADSSW1	M68xx302 Family Application Development System (302 FADS) Software (12/23/1997)	MOTOROLA	zip	563	0.4
MC68302ADSSW1	MC68302 Application Development System Software (12/04/1996)	MOTOROLA	html	0	-
MC68302ADSSW2	MC68302ADS IBM-PC Host Files (12/04/1996)	MOTOROLA	html	0	-
MC68302COD1	MC68302 Code for Setting Up an SCC in UART Mode (12/04/1996)	MOTOROLA	txt	8	-
MC68302COD2	MC68302 Demo Code for the 302ADS Board 08/31/1990	MOTOROLA	html	1	-

MC68302COD3	Sample Code - Configuring a UART on the MC68302 (12/04/1996)	MOTOROLA	txt	1	-
MC68302COD4	MC68302 Source Code for ISDN Smart NT1 Layer 1 (02/04/1997)	MOTOROLA	zip	6	-
MC68302COD5	MC68302 Confidence Test Software (10/22/2001)	MOTOROLA	zip	194	-
MC68EN302COD1	MC68EN302 Software Example (05/13/1997)	MOTOROLA	zip	454	-
MC68SC302SW	MC68SC302 Application Development System Software ** Not recommended for new design. Part is on EOL notification. ** (09/24/1997)	MOTOROLA	zip	271	3.1

Supporting Information

ID	Name	Vendor ID	Format	Size K	Rev #
MC68EN302_START	MC68EN302 Getting Started Hints	MOTOROLA	txt	8	-

[top]

Orderable Parts Information

PartNumber	Package Info	Life Cycle Description (code)	Remarks	Budgetary Price QTY 1000+ ($US)	Order Availability
MC68302CFC16C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	16MHz, extended temp	$16.04	
MC68302CFC20C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	20MHz, extended temp	$18.20	
MC68302CPV16VC	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	16MHz, extended temp	$19.37	
MC68302CRC16C	PGA 132	PRODUCT MATURITY/SATURATION(4)	16MHz, extended temp	$42.44	
MC68302CRC20C	PGA 132	PRODUCT MATURITY/SATURATION(4)	20MHz, extended temp	$53.05	
MC68302FC16C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	16MHz	$13.35	

MC68302FC16CR2	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	Packaged in Tape & Reel	$15.72	-
MC68302FC20C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	20MHz	$15.17	☐
MC68302FC20CR2	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	Packaged in Tape & Reel	-	-
MC68302FC25C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	25MHz	$16.97	☐
MC68302FC25CR2	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	Packaged in Tape & Reel	$20.44	☐
MC68302PV16C	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	16MHz	$12.91	☐
MC68302PV16VC	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	16MHz	$16.13	☐
MC68302PV20C	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	20MHz	$14.71	☐
MC68302PV25C	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	25MHz	$16.52	☐
MC68302PV33C	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	33MHz	$18.32	☐
MC68302RC16C	PGA 132	PRODUCT MATURITY/SATURATION(4)	16MHz	$40.32	☐
MC68302RC20C	PGA 132	PRODUCT MATURITY/SATURATION(4)	20MHz	$50.40	☐
MC68302RC25C	PGA 132	PRODUCT MATURITY/SATURATION(4)	25MHz	$63.00	☐
SPAK302CFC20C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$20.19	☐
SPAK302FC25C	PQFP 132 0.950*.950P.025	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$20.44	☐
SPAK302PV16VC	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$20.47	☐

SPAK302PV33C	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$21.79	☐
SPAK302RC25C	PGA 132	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$50.79	☐
KMC68LC302PU16CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$11.35	☐
KMC68LC302PU20CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$13.61	☐
KMC68LC302PU25CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$15.87	☐
KM68LC302PU20VCT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$15.87	☐
KMC68EN302PV20BT	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$22.73	☐
KMC68EN302PV25BT	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	part number for ordering samples only	$27.27	☐
MC68EN302CPV20BT	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	20MHz	$21.65	☐
MC68EN302PV20BT	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	20MHz	$18.04	☐
MC68EN302PV25BT	LQFP 144 20*20*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	25MHz	$21.65	☐
M68LC302CPU16VCT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	-	$13.05	☐

M68LC302CPU20VCT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	-	$13.44	
MC68LC302CPU16CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	16MHz	$8.74	
MC68LC302CPU20CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	20MHz	$10.93	
MC68LC302PU16CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	16MHz	$8.56	
MC68LC302PU16VCT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	16MHz	$12.80	
MC68LC302PU20CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	20MHz	$10.71	
MC68LC302PU20VCT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	20MHz	$13.18	
MC68LC302PU25CT	LQFP 100 14*14*1.4P0.5	PRODUCT MATURITY/SATURATION(4)	25MHz	$12.80	

[top]

Motorola Home | Semiconductors | Login | Support | Contact Us | Site Map
Products | Documentation | Tools | Design Resources | Applications