CODATA SYSTEMS CORPORATION

68000 Central Processing Unit Hardware Reference Manual

68000 CENTRAL PROCESSING UNIT

DESCRIPTION, OPERATION AND MAINTENANCE

CONTENTS PAGE

1. GENERAL 3
   Features 3

2. PHYSICAL DESCRIPTION 5
   Options 5
   Test Points 5

3. FUNCTIONAL DESCRIPTION 7
   Overview 7
   68000 Microprocessor 10
   Bus Structure 10
   Addressing 11
   ROM 11
   Memory Management —
      General 13
      Context Register 13
      Segment Map 13
      Segment Level Protection 14
      Page Map 15
      Page Control 15
      Virtual Memory 15
   Allocation of Logical Address Space . . 15
   On-Card 256k RAM 15
   Interrupt 18
   Dual UART 20
   Five Channel Timer 21
   796 Bus Interface 21
   Parallel Input Port 22
   Device Decode 25
   Power-On-Reset 25
   System Timing 26

4. SPECIFICATIONS 33

5. OPERATION AND PROGRAMMING . . 37
   Options 37
   RS-423A Port 37
   Power-On-Reset — Entering Boot State . . 37
   Memory Map Initialization 38
   Copying the Exception and Interrupt
   Vectors to RAM 38
   On-Card RAM Initialization 38
   RAM Refresh 38
   Exiting Boot State 38

CONTENTS PAGE

   UART Programming —
      General 38
      Control Register 39
      Status Register 44
      Transmit/Receive Data 44
   16-Bit Parallel Input Port 45
   Exceptions 45
   System Space Error 45
   Segment Map Error 45
   Page Map Error 45
   Timeout Error 45
   Parity Error 45
   Exception Handling 45
   Timer Programming
      General 46
      Initialization 46
      Watchdog Timer Set Up 47
      RTC and Refresh Timer Set Up . . . 47
      UART Timer Set Up 49

6. MAINTENANCE 51
   Diagnostics 51
   Warranty Service 51
   Returning Material For Repair 51

7. REFERENCE 52
   Logic Diagram and Replaceable Parts List . 52
   IEEE 796 Microcomputer Bus 52
   I/O Ports 52
   Technical Manual Revisions 52

ILLUSTRATIONS

ILLUSTRATIONS	PAGE
Figure 2-1 — 68000 Central Processing Unit — 92-1012-xx	4
Figure 2-2 — 68000 Central Processing Unit Layout	6
Table 2-1 — Test Points	7
Figure 3-1 — 68000 Central Processing Unit Block Diagram	8
Table 3-1 — 68000 CPU Logical Address Locations	11
Figure 3-2 — Memory Map Block Diagram	12
Table 3-2 — Segment Level Protection Attributes	14
Table 3-3 — Memory Map Register Bit Map	16
Figure 3-3 — On-Card 256k Byte RAM Block Diagram	17
Figure 3-4 — Interrupt Block Diagram	19
Figure 3-5 — UART Block Diagram	20
Figure 3-6 — Timer Block Diagram	21
Figure 3-7 — 796 Bus Interface Block Diagram	22
Figure 3-8 — Parallel Input Block Diagram	23
Figure 3-9 — Device Decoder Block Diagram	24
Figure 3-10 — Power-On-Reset Block Diagram	25
Figure 3-11 — System Timing Block Diagram	26
Table 3-4 — 68000 Central Processing Unit Active 796 Bus Signals	27
Figure 3-12 — On-Card RAM Read Cycle Timing	28
Figure 3-13 — On-Card RAM Write Cycle Timing	29
Figure 3-14 — On-Card ROM Read Cycle Timing	30
Figure 3-15 — Off-Card RAM and Input Output Read Cycle Timing	31
Figure 3-16 — Off-Card RAM and Input Output Write Cycle Timing	32
Table 4-1 — 68000 Central Processing Unit 92-1012-xx — Specifications	33
Table 5-1 — 68000 Central Processing Unit Options P/N 91-1012-xx	36
Figure 5-1 — UART Register Values	39
Figure 5-2 — Write Register 0 Routine and Register Map	40
Figure 5-3 — Write Register 1 Routine and Register Map	40
Figure 5-4 — Write Register 2 Routine and Register Map	41
Figure 5-5 — Write Register 3 Routine and Register Map	41
Figure 5-6 — Write Register 4 Routine and Register Map	42
Figure 5-7 — Write Register 5 Routine and Register Map	43
Figure 5-8 — Read UART A or B Status Routine and Register Map	44
Figure 5-9 — Send Data to UART A or B Routine and Register Map	44
Figure 5-10 — Receive Data from UART A or B Routine	44
Figure 5-11 — Timer Initialization Routine	46
Figure 5-12 — Watchdog Timer Set Up Routine	47
Figure 5-13 — RTC and Refresh Timer Set Up Routines	47
Figure 5-14 — RTC and Refresh Timer Clear Routine	48
Figure 5-15 — Values for Period Between Interrupts on Resets	48
Figure 5-16 — Values for Divisor Constant	48
Figure 5-17 — UART Timer Set Up Routines	49
Figure 5-18 — Values of Constants Used to Program Timer Device	50
Figure 7-1 — 68000 Central Processing Unit Logic Diagram	53
Table 7-1 — 68000 Central Processing Unit Replaceable Parts List	58
Table 7-2 — Pin Assignments on 796 Bus Board Connector (P1)	63
Table 7-3 — Pin Assignments on 796 Bus Board Connector (P2)	64
Table 7-4 — 68000 CPU Connector P1 Pin Assignments	65
Table 7-5 — 68000 CPU Connector P2 Pin Assignments	67
Table 7-6 — Pin Assignments of RS-423 Serial IO Board Connector (J1)	69
Table 7-7 — Pin Assignments of 16-Bit Parallel Input Port (J2)	70

1. GENERAL

1.01 This manual provides a physical description, functional description and operating theory for effective maintenance of the Codata Systems Corp. 68000 Central Processing Unit, 92-1012-xx. This manual revision furnishes information pertaining to design enhancements to the 68000 Central Processing Unit resulting in higher processing speed.

1.02 The 68000 Central Processing Unit (CPU) is supplied as a single printed circuit assembly (PCA) for use as a system component in the Codata Systems Corp:

(1) CTS-Series Mainframe,

(2) CTW-Series Mainframe.

Features

1.03 The 68000 CPU is a powerful single card processor designed around the MCL68000L microprocessor (μP) device. CPU features include:

- The 68000 μP operates at 8 MHz.
- IEEE 796 Microcomputer Bus compatible.
- Multimaster capability.
- The entire CPU is on a single PCA.
- 20-bit 796 Bus providing 1M byte addressing.
- A segmented, paged, memory management method.
- Up to 256k bytes of on-card parity-checked dynamic Random Access Memory (RAM). The RAM operates without wait states.
- Up to 32k bytes of on-card Read Only Memory (ROM).
- Two universal asynchronous receiver transmitters (UARTs) for serial input output (I/O). EIA RS-423A compatible.
- Five 16-bit timer channels.
- One 16-bit parallel input port.
- Seven level interrupt with priority set by option jumpers.
- Single +5 Vdc power requirement.

NOTE

The following reference notations apply in this technical manual:

*(1) A * suffix to a single name indicates logical NOT and active low.*

(2) In and out references are in respect to CPU or bus master.

(3) 1k byte equals 1,024 bytes, i.e., 64k bytes equals 65,536 bytes.

(4) Codata Systems part numbers are made up of eight digits, e.g., the part number of this manual is 05-0004-01.

(5) A suffix -xx to a part number indicates the part or assembly may have more than one configuration in production, i.e., the 68000 Central Processing Unit is 92-1012-xx.

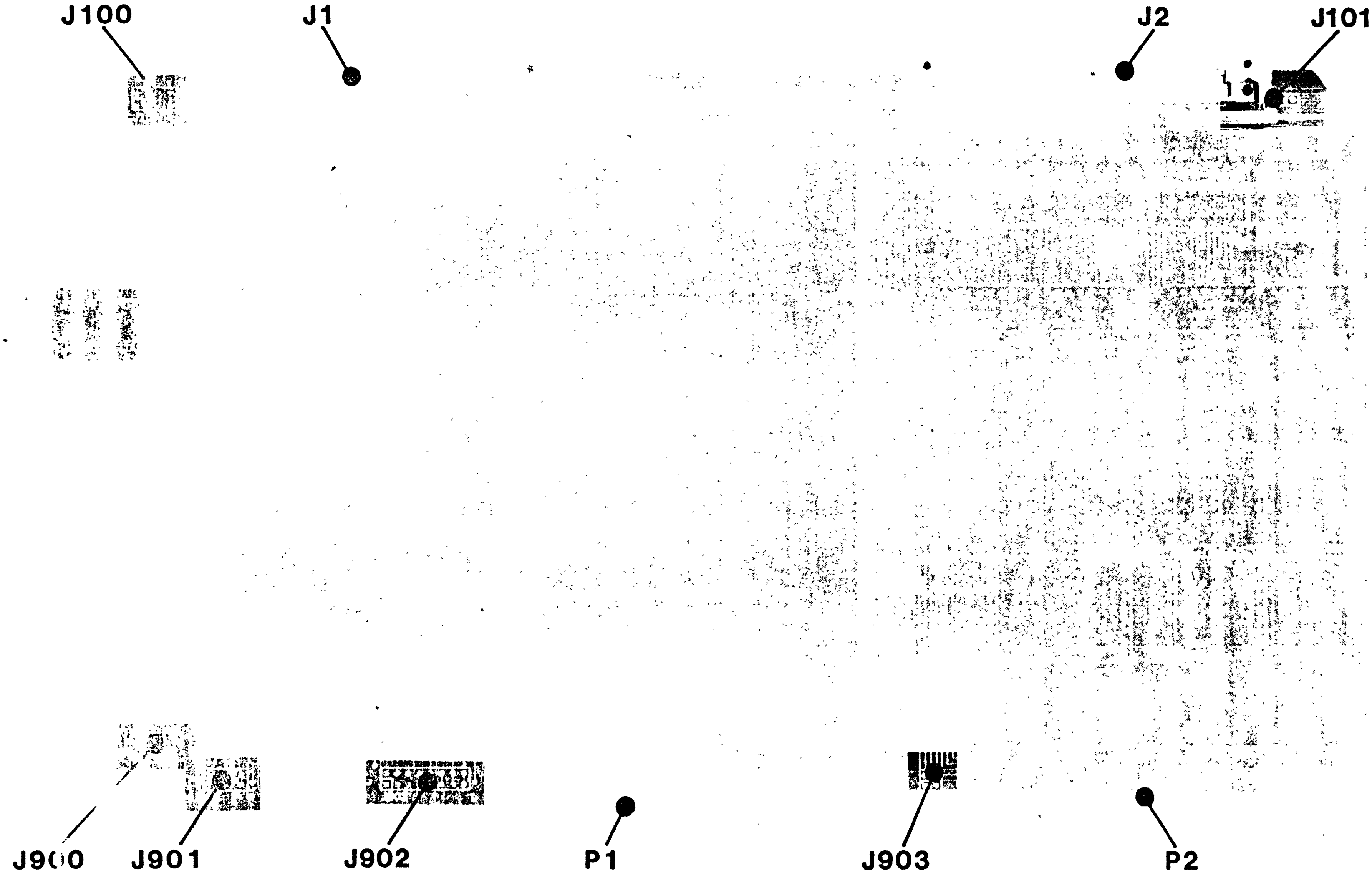


Figure 2-1 — 68000 Central Processing Unit — 92-1012-xx

2. PHYSICAL DESCRIPTION

2.01 The 68000 Central Processing Unit (CPU) 92-1012-xx is an integrated system component incorporating all the necessary component parts to provide the Mainframe with a single PCA CPU. Figure 2-1 illustrates the 68000 CPU. The PCA contains:

(1) A 68000 μP section. This is a 16-bit μP operating at 8 MHz.

(2) A Memory Management section.

(3) A Memory Control section.

(4) A 256k byte RAM section providing the μP with up to 256k bytes of dynamic memory independent from the Mainframe RAM.

(5) A 32k ROM section providing the μP with up to 32k firmware.

(6) A UART section. This provides two RS-423A serial I/O ports for the Mainframe.

(7) A Timer section providing the programmer with five 16-bit programmable timers.

(8) A 796 Bus Interface section.

(9) A 16-bit Parallel Input Port section.

(10) A Clocks and Logic section.

Figure 2-2 illustrates the physical locations of these sections on the PCA.

2.02 The PCA measures 6.0 inches by 12.0 inches. A pair of edge-type pc connectors, P1 and P2, mate with the 796 Bus Backplane connectors.

(1) P1 is a dual 43-position, 86-conductor pc connector. The pin assignments conform to the 796 Bus specification.

(2) P2 is a dual 30-position, 60-conductor pc connector. These pins are used for off-card RAM expansion. The pin assignments do not conform to the 796 Bus specification.

IMPORTANT

> ***The 68000 should not be installed in a backplane having the connector mating with P2 wired to the 796 Bus specification.***

2.03 A pair of flat ribbon cable recepticals are provided at the top of the PCA for connection to external I/O devices.

(1) The J1 connector provides data and status/control lines from two RS-423A data communication lines (DCL). The 50-conductor interconnect cable is terminated by two DB-25S connectors on the Mainframe Rear Panel. Refer to Table 7-6 for individual PCA pin assignments.

(2) The J2 connector provides data input lines for the 16-bit Input Port and lines for an external reset switch. Refer to Table 7-7 for individual PCA pin assignments.

2.04 Distinctive white silkscreen marking has been provided on the component side of the PCA. Component reference designators are marked where practical. They facilitate locating individual parts on the logic diagram or replaceable parts list.

Options

2.05 Several alternate features can be configured through option jumpers on the PCA. Refer to Figure 2-1.

(1) J100 configures UART B as a DCE or DTE port and selects the ROM size,

(2) J800 provides connection of an interrupt from the SIO and the Timer to interrupt levels 5 and 6,

(3) J900 generates the 796 Bus control signals with or without using the 8289 device,

(4) J901 selects the source or destination of INIT* and Bus Clocks,

(5) J902 selects the interrupt levels.

Test Points

2.06 Test points for the 68000 CPU have been provided on the PCA for repair and maintenance. Table 2-1 tabulates these by location and function.

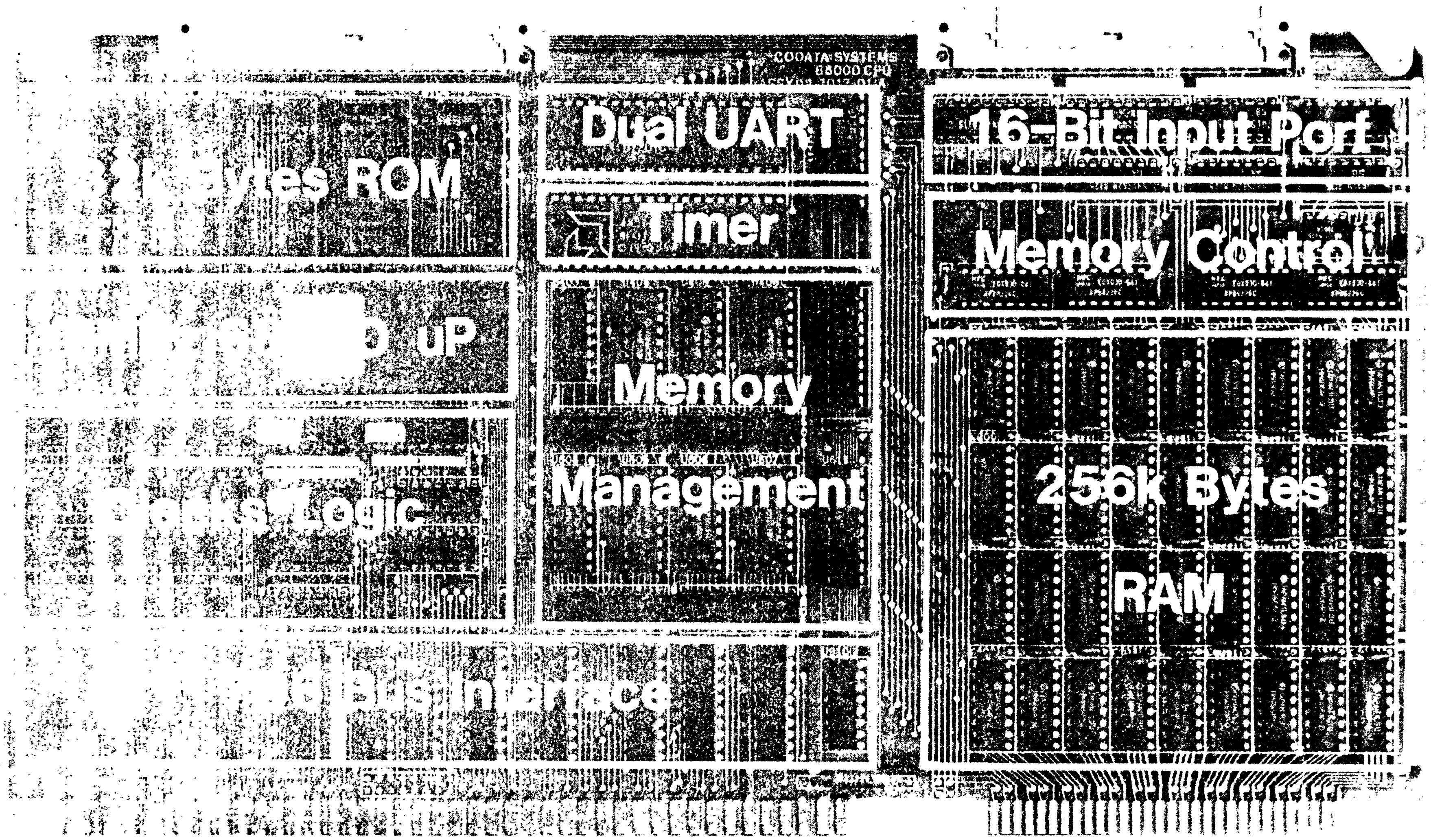


Figure 2-2 – 68000 Central Processing Unit Layout

Table 2-1 — Test Points

Reference	Mnemonic	Function
	Halt Indicator	
J101.1	VCC	+5 Vdc
J101.2	M.REF*	Memory Refresh — Halt
	NOTE	
	An LED with an internal current limiting resistor may be installed between J101.1 and J101.2 to act as an indicator that the μP is HALTED. An LED is not supplied in the standard PCA configuration.	
	General Test Points	
J400.1	VCC	+5 Vdc
J400.2	SYS.ACCESS*	System Access
J400.3	C62.0-31	16 MHz Clock
J400.4	TIMEOUT*	Timeout
J400.5	DTACK*	Data Acknowledge
J400.6	BERR	Bus Error
J400.7	GND	Signal Ground
J400.8	GND	Signal Ground
	Memory Column Address Strobe Test Points	
J903.1	M.CAS0*	Memory CAS0
J903.2	M.CAS1*	Memory CAS1
J903.3	M.CAS2*	Memory CAS2
J903.4	M.CAS3*	Memory CAS3

3. FUNCTIONAL DESCRIPTION

Overview

3.01 The basic function of a central processing unit (CPU) in a computing system is to accept data and processing instructions, perform processing operations and deliver the processed data. Several additional functions are provided by the 68000 CPU besides this basic function. Figure 3-1 illustrates each function in block diagram.

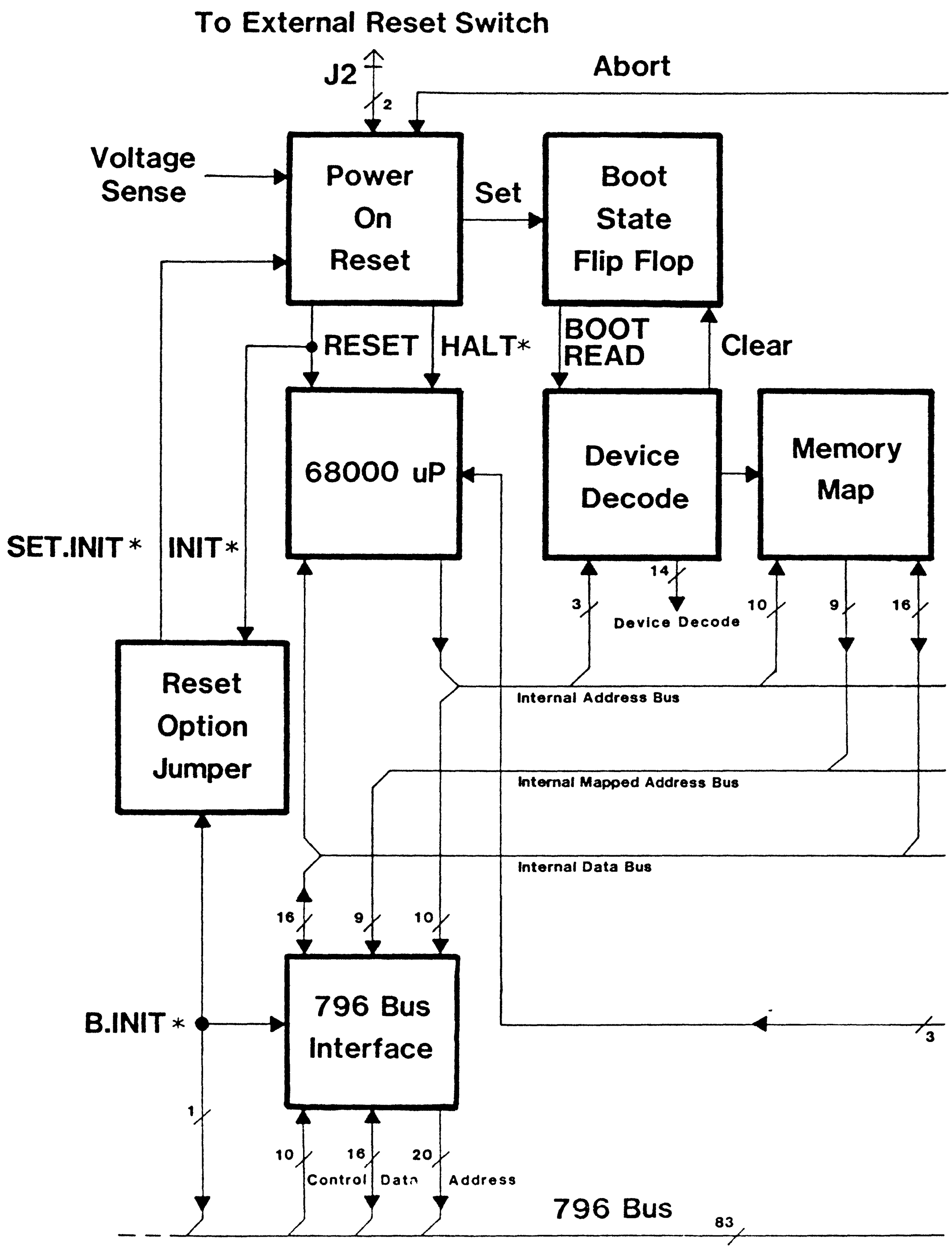


Figure 3-1 — 68000 Central Processing Unit Block Diagram

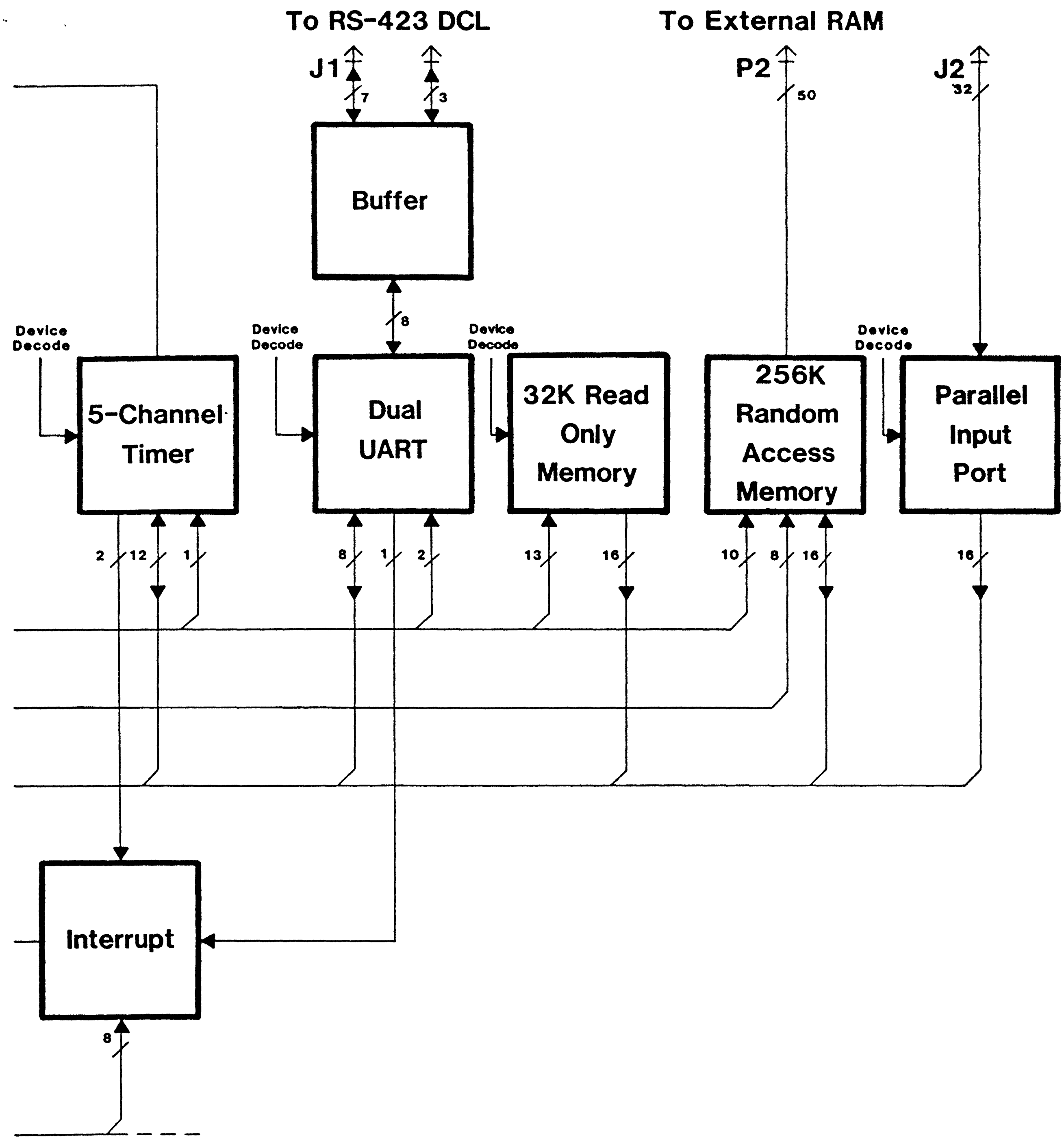


Figure 3-1 — 68000 Central Pricessing Unit Block Diagram (Cont.)

(1) 68000 microprocessor (μP),

(2) Memory Map,

(3) Random Access Memory,

(4) Interrupt,

(5) Dual UART — Serial Input Output (I/O),

(6) Five-Channel Timer,

(7) 796 Bus Interface,

(8) Parallel Input Port,

(9) Device Decode,

(10) Power-On-Reset,

(11) System Timing.

The following paragraphs will discuss each function in detail. Figures 3-2 through 3-11 furnish detailed block diagrams of the sections. CPU timing is illustrated in Figures 3-12 through 3-16. These figures should be used in conjunction with the logic diagram, Figure 7-1, for the descriptions which follow.

68000 Microprocessor

3.02 The principle device on the CPU is the 68000 μP. This is a high-performance μP with 32-bit architecture and a large uniform memory space. This μP features sixteen 32-bit registers divided into two sets of eight address registers and eight data registers.

3.03 The μP instruction set and addressing modes are both extremely regular in their implementation with a minimum of special cases thus making high-level language code generation fairly simple.

3.04 The μP manipulates three major data formats:

(1) 8-bit words,

(2) 16-bit words,

(3) 32-bit words.

The μP can operate in supervisor or user states assuring a secure operating system. The 68000 CPU has been designed to fully utilize the high performance of the μP by providing on-card RAM that will operate without wait states at the 8 MHz speed of the system.

Bus Structure

3.05 The 68000 CPU has two principle data busses:

(1) An internal 16-bit synchronous bus to communicate with on-card RAM/ROM and I/O devices. Since on-card accesses do not require the 796 Bus, the 796 Bus is available for use by other 796 Bus Masters, e.g., Diskette or Winchester Disk Controllers.

(2) The 796 Bus for accessing off-card RAM and I/O devices.

NOTE

> *While the CPU has complete access to the 796 Bus, the 796 Bus cannot access on-card memory or I/O devices.*

Accesses to 796 Bus devices are slower than on-card devices. The μP cycles are *stretched* by an amount appropriate to the 796 Bus device being accessed. Refer to Figures 3-15 and 3-16.

3.06 The 68000 CPU is initialized through a reset which can be activated through several channels. The reset logic is detailed in 3.54 below.

3.07 The μP is reset when both the HALT* and RESET* lines are held low. A card reset can be initiated by the μP noting the logical condition of these two lines and holding the RESET* line low.

3.08 Several operations take place after a reset:

(1) The *Boot State Flip-Flop* is set and the boot state is entered.

(2) The *Device Decode* enables the *Read Only Memory* (ROM).

(3) Instructions stored in ROM are overlaid into address space normally occupied by RAM starting at location 000 000H. These instructions form what is usually called the Boot Strap process. During this process, the exception vectors located at address 000 008H to 000 0FFH are copied to RAM by reading the data from ROM and writing the data to RAM. This process is called shadow RAM.

(4) The last instruction in the boot strap program is a write to ROM location 200 000H causing the *Device Decode* to clear the *Boot State Flip Flop*.

(5) After the *Boot State Flip Flop* has cleared the ROM, instructions are removed from 000 000H and the RAM locations become available for program variables, e.g., exception vectors and program. The boot state is exited.

Addressing

3.09 Table 3-1 lists the 68000 CPU address mapping for memory-managed RAM, and all on-card devices. All addresses above 200 000H are not memory managed and are absolute addresses for the named devices.

ROM

3.10 Up to 32k bytes of ROM may be installed on the 68000 CPU in two separate groups, ROM 0 and ROM 1, whose addresses begin at 200 000H and 400 000H, respectively. The ROM 0 group is also addressed starting at location 000 000H while the 68000 CPU is in boot state.

3.11 Sockets on the printed circuit assembly provide for three ROM types:

(1) The 2716 device for two groups of 2k x 16-bit words or 8k bytes.

(2) The 2732 device for two groups of 4k x 16-bit words or 16k bytes.

(3) The 2764 device for two groups of 8k x 16-bit words or 32k bytes.

Table 3-1 — Logical Address Locations

Address		Function
From	To	
000 000H	1FF FFFH	Mapped RAM and I/O[1]
200 000H	3FF FFFH	ROM 0
400 000H	5FF FFFH	ROM 1
600 000H	7FF FFFH	UART A and B
800 000H	9FF FFFH	Five Channel Timer
A00 000H	BFF FFFH	Page Map (read/write)
C00 000H	DFF FFFH	Segment Map (read/write) Context Register (read)
E00 000H	FFF FFFH	Context Register (write) 16-bit Input Port (read)

Note:

(1) During Boot State, Boot Strap instructions stored in ROM are shadowed into this area starting at 000 000H.

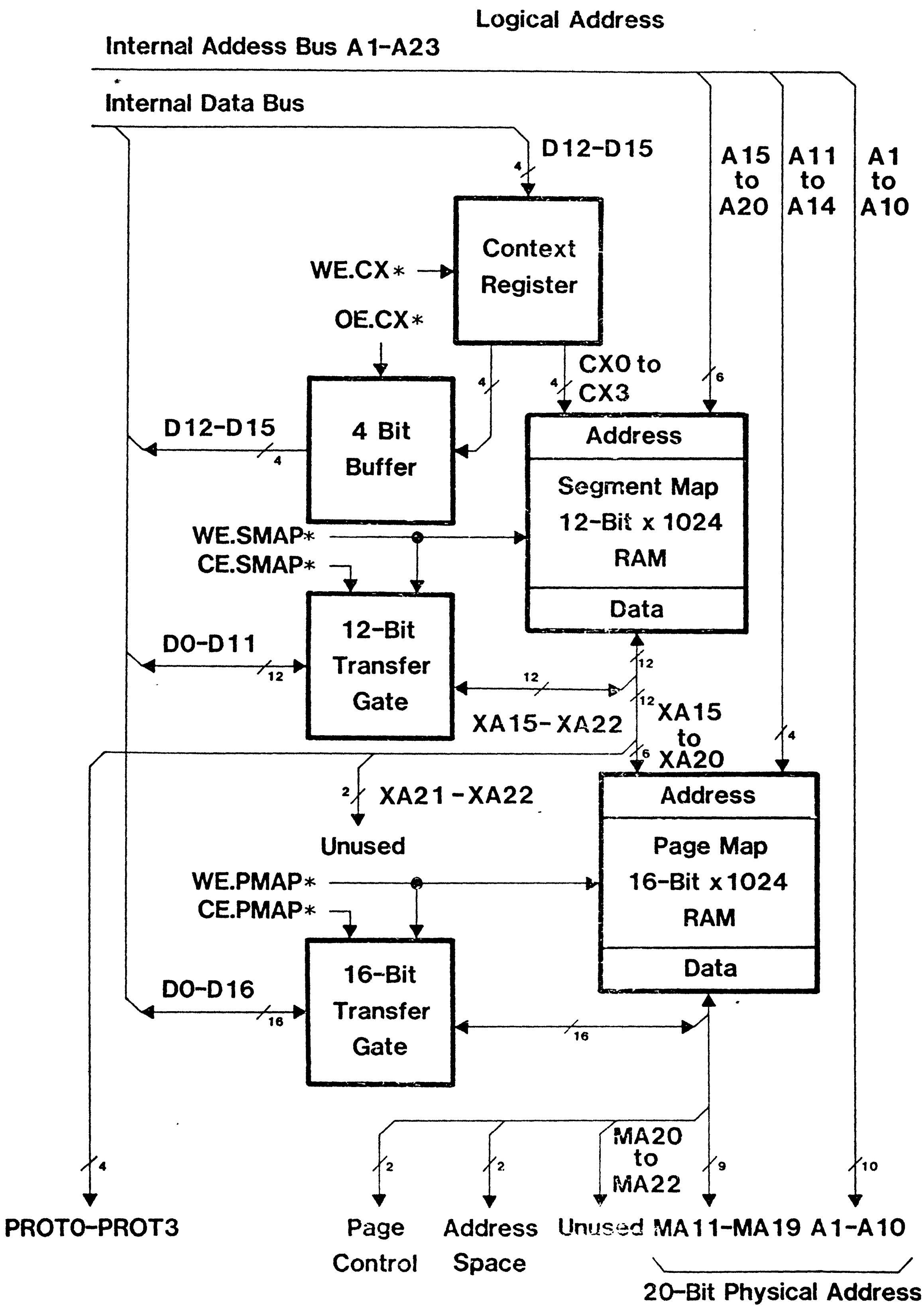


Figure 3-2 — Memory Map Management Block Diagram

Memory Management – General

3.12 The 68000 CPU employs a segmented, paged memory management method to facilitate the effective use of memory in large and complex programs. The μP is aware of an address space which is 24 bits wide. Thus the μP can directly address 16M bytes of memory. Multi-user 16M byte systems, however, require addressing methods which are more sophisticated than just a linear array of bytes ranging from 0 to 16M bytes. There is a need to partition user programs into separate logical address spaces such as execute-only code, read-only data, stack area and so on. This need is met by dividing programs into segments. There is a need for efficient management of the physical layout and allocation of such large address spaces and support features such as demand paging in a virtual memory system. This need is met by dividing the address spaces into pages.

3.13 A multi-tasking operating system needs a means to switch quickly between contexts, that is, to have the CPU work on a new program while a previous program is suspended, e.g., waiting for some peripheral transfer to complete. This need is met by providing separate contexts, addressed through a context register, which points the CPU at a fresh set of segments and pages. Figure 3-2 illustrates the Memory Management in block diagram.

3.14 The Memory Management section provides address translation, sharing and memory allocation control for multiple processes executing on the CPU. The address space is divided into pages of 2k bytes each. The page address bits, A0 through A10, pass through the translation process unmodified. Address bits, A11 through A20, are subject to translation. Bits A21 through A23 are reserved for special system functions and take no part in the address translation. The maximum logical address space for a process on the CPU is thus 21 bits or 2M bytes. This 21-bit address is further extended with a four-bit *Context Register* also known as the process or user number.

3.15 The 23-bit logical addresses, A1 through A23, from the μP are translated into 20-bit physical addresses in two stages. In the first stage, the logical address from the μP is translated by the ***Segment Map*** look-up table into a virtual address, XA15-XA20. In the second stage, this virtual address is translated by the ***Page Map*** look-up table into a 20-bit physical address, A1-A10 and MA11-MA19.

3.16 A0 is not generated by the μP. The μP uses LDS* and UDS* to select the appropriate byte or bytes from a 16-bit word addressed by A1-A23. A0 is generated by other hardware on the 68000 CPU only for use in 796 Bus accesses.

3.17 Protection is associated with the *Segment Map*. Four protection bits, PROT0-PROT3, are provided or disallow read, write and execute access to two levels, the system level and the user level. Refer to Table 3-2.

3.18 Page access control and address space control are provided at the page map level. Page access control consists of two bits which remember that a page has been referenced, used and written to, dirty. Address space control determines in which physical address space, on-card or off-card 796 Bus, a page is located as well as whether it references memory or input/output. Since no input/output addressing is done on card by the Memory Management System, this designation is interpreted as an invalid page in which case a reference to a word in that page causes a page default.

Memory Management – Context Register

3.19 In a system with multiple executing processes, it is important to be able to switch quickly between processes without having to reload all the state information relating to the address translation for a particular process. The *Context Register* is a four-bit register, writable and readable under supervisor controls, that selects one of 16 unique sections of the *Segment Map*. This memory management method can thus contain the maps for 16 distinct process or user translations at the same time.

Memory Management – Segment Map

3.20 The *Segment Map* is a 1024 entry table indexed by the four-bit *Context Register* and the six most significant bits of the logical address, A15-A20. The output of the *Segment Map* is six virtual address bits, XA15-XA20, and four protection bits, PROT0-PROT3. Each context thus has up to 64 segments and each segment has

Table 3-2 — Segment Level Protection Attributes

Protect Code	Access Allowed					
	PR3	PR2	PR1	PR0	System	User
0	0	0	0	0	- - -	- - -
1	0	0	0	1	- - x	- - -
2	0	0	1	0	r - -	- - -
3	0	0	1	1	r - x	- - -
4	0	1	0	0	r w -	- - -
5	0	1	0	1	r w x	- - -
6	0	1	1	0	r - -	r - -
7	0	1	1	1	r w -	r - -
8	1	0	0	0	r - -	r w -
9	1	0	0	1	r w -	r w -
10	1	0	1	0	r w -	r - x
11	1	0	1	1	r w -	r w x
12	1	1	0	0	r - x	r - x
13	1	1	0	1	r w x	r - x
14	1	1	1	0	r w x	- - x
15	1	1	1	1	r w x	r w x

Notes:

(1) r = read.

(2) w = write.

(3) x = execute

(4) - = attribute not enabled.

individual protection attributes. Segments may be kept private to a process or shared with other processes. The six-bit virtual address from a segment entry refers to a block of 16 consecutive page entries in the *Page Map*. A segment can be as large as 32k bytes by using all 16 of the associated Page Map entries. A segment may be as small as 2k bytes by invalidating the unused page entries in the *Page Map*. By concatenating consecutive *Segment Map* entries, a process can have a single address space of 2M bytes.

Memory Management — Segment Level Protection

3.21 Each entry in the segment table contains four bits of protection information which may be used to control the access rights of that specific portion of the logical address space. The access codes are assigned to the Unix® notation *rwx* where:

(1) r is read access allowed,

(2) w is write access allowed,

(3) x refers to execute-only access allowed,

(4) – denote absence of that privilege.

Full access is denoted rwxrwx where the first rwx applies to system access and the second rwx to user access. The assignment of the four-bit protection code to the six-level protection is illustrated in Table 3-2.

Memory Management — Page Map

3.22 In the *Page Map* the six-bit virtual address from the *Segment Map* and the next four logical address bits from the μP are translated into a physical address and a physical address space. Each segment virtual address refers to a block of 16 consecutive page entries in the *Page Map*.

3.23 The output of the *Page Map* is the upper nine bits of the physical address which is concatenated with the lower 11 bits of the logical address to form a 20-bit Physical Address.

3.24 As well as determining the upper nine bits of the physical address, a page entry also determines to which physical address space the address belongs. By setting the address space control bits appropriately, a page may be declared to be in one of these address spaces:

(1) 0 — On-card memory space,

(2) 1 — Invalid page,

(3) 2 — 796 Bus RAM,

(4) 3 — 796 Bus I/O.

Notice that each of these address spaces is 20 bits or 1M bytes even though the on-card memory is at most 256k bytes and the off-card memory is at most 1M byte. It is up to the supervisory software to initialize the memory management segment and page maps correctly for a particular system configuration.

Memory Management — Page Control

3.25 Each *Page Map* entry has two bits of page access control information. The referenced bit, often called the *used* bit, indicates that this page has been referenced:

(1) Data read reference,

(2) Data write reference,

(3) Execute reference.

The modified bit, often called the *dirty* bit, indicates that this page has been written to. These bits are automatically updated on every valid mapped reference. These bits are intended for future use in virtual memory systems as described below. Refer to Table 3-3.

Memory Management — Virtual Memory

3.26 The page map organization, together with the page control bits, provide enough information to implement virtual memory and demand paging.

3.27 The current implementation of the μP cannot recover from page faults to the extent required. Specifically, it does not store enough internal state information to be able to restart an instruction which was aborted because of a page fault. However, by limiting the set of operations that can cause page faults, it is possible to provide a limited form of virtual memory capability. For example, limiting virtual memory access to load and store operations makes recovery possible. Thus virtual data spaces can be achieved. The current version of the CPU does not yet employ virtual memory.

Allocation of Logical Address Space

3.28 The 68000 CPU does not provide the capability to access the full 16M bytes of memory that the μP address lines will accommodate. Rather, the logical address space of the μP has been allocated to various device functions. The address allocation is described in Table 3-1 and 3-3. Note the dual functions of the *Segment Map* and the *Context Register* locations. On a write to an entry, the upper four bits of the data are ignored and only the lower twelve bits are used to write to a segment entry. On a read from a *Segment Map* entry, the upper four bits are the contents of the *Context Register* and the lower twelve bits are the contents of the addressed *Segment Map*. On a write to location E00 000H, data is written to the *Context Register*. On a read from location E00 000H, data is read from the *16-Bit Input Port*.

On-Card 256k RAM

3.29 Figure 3-3 illustrates the 256k Random Access Memory in block diagram. 64k dynamic RAM devices are used to implement the on-card RAM. The RAM is organized as follows:

(1) Two 64k x 16-bit word banks, 0 and 1, on-card with provision to expand to an additional two banks off-card,

(2) Each bank is divided into an upper and lower byte.

Table 3-3 — Memory Management Register Bit Map

Address	Attribute (1)	Register	Data Bus Bit Specifications															
			15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
A00 000H	R/W	Page Map	D	U	PD 1	PD 0	MA 22	MA 21	MA 20	MA 19	MA 18	MA 17	MA 16	MA 15	MA 14	MA 13	MA 12	MA 11
C00 000H	W	Segment Map	na	na	na	na	PR 03	PR 02	PR 01	PR 00	XA 22	XA 21	XA 20	XA 19	XA 18	XA 17	XA 16	XA 15
C00 000H	R	Segment Map and Context	CX 03	CX 02	CX 01	CX 00	PR 03	PR 02	PR 01	PR 00	XA 22	XA 21	XA 20	XA 19	XA 18	XA 17	XA 16	XA 15
E00 000H	W	Context	CX 03	CX 02	CX 01	CX 00	na	na	na	na	na	na	na	na	na	na	na	na
E00 000H	R	Parallel Input	IN 15	IN 14	IN 13	IN 12	IN 11	IN 10	IN 09	IN 08	IN 07	IN 06	IN 05	IN 04	IN 03	IN 02	IN 01	IN 00

Notes:

(1) Attribute R = Read.
W = Write.
R/W = Read or Write.

(2) D = Dirty means this page has been written to.

U = Used means this page has been accessed.

(3) PDnn = Page definition Bits 0-1.

PD1	PD0	Definition
0	1	On-Card RAM
0	0	Invalid Page
1	0	796 Bus RAM
1	1	796 Bus I/O.

(4) MAnn = Translated or mapped page address bits 11-22.

(5) PRnn = Segment protect code bits 0-3. Refer to Table 3-2.

(6) XAnn = Translated segment address bits 15-22.

(7) CXnn = Context register bits 0-3.

(8) INnn = 16-bit parallel input register bits 0-15.

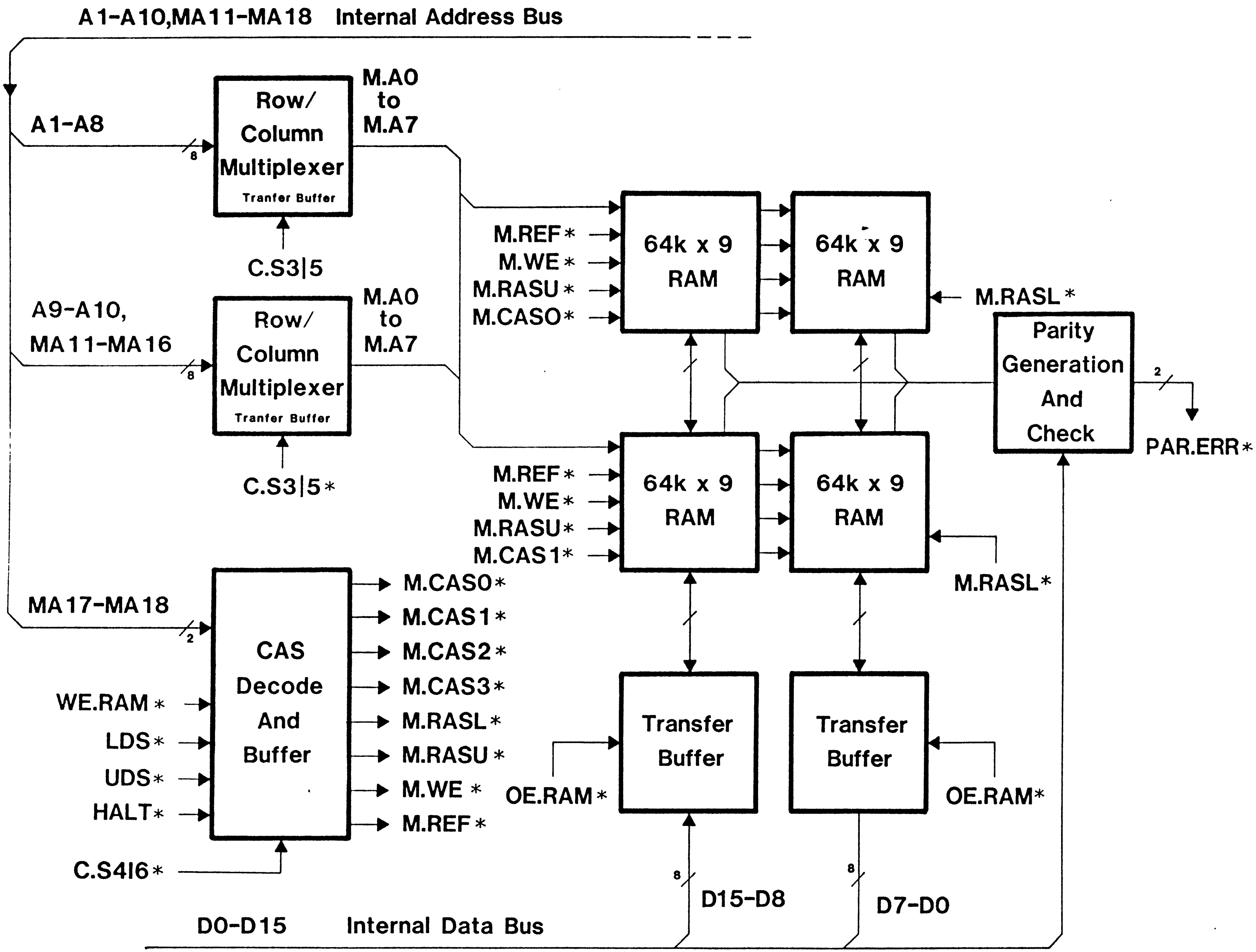


Figure 3-3 — On-Card 256k Byte RAM Block Diagram

05-0004-01

Bit 9 is used for byte parity checking. The organization of memory is in bytes. Read and write operations are performed in words.

NOTE

> *The 796 Bus provides both byte and word addressing. The 68000 μP performs full word addressing, i.e., the least significant bit, A0, is not used for internal addressing operations.*

3.30 The internal data and address lines are passed through Transfer Buffers. The data Transfer Buffers are transceivers. These *Transfer Buffers* are provided for several reasons:

(1) Testing of memory is facilitated by isolating RAM from the internal address and data busses.

(2) They furnish load buffering for internal address and data busses.

3.31 Internal Address Lines, MA17-MA18, are decoded and buffered to form column address strobe lines, M.CAS0*-M.CAS3*. These lines are also used to select the bank.

3.32 μP Control Signals:

(1) UDS*, upper data strobe, becomes M.UDS* and selects the upper byte of the bank,

(2) LDS*, lower data strobe, becomes M.LDS* and selects the lower byte of the bank.

3.33 Device decoder signal write enable RAM, WE.RAM*, becomes M.WE* the strobe for writing into RAM.

3.34 To read a word from RAM:

(1) UDS* and LDS* are asserted,

(2) M.CAS0*-M.CAS3* selects the bank,

(3) OE.RAM* active.

Refer to Figure 3-12 for the timing relationships.

3.35 To write a word to RAM:

(1) UDS* and LDS* are asserted,

(2) M.CAS0*-MCAS3* selects the bank,

(3) M.WE* is asserted.

Refer to Figure 3-13 for the timing relationships.

3.36 The row address strobe lines, M.RASU* and M.RASL*, which are associated with the upper and lower byte are common to banks.

3.37 The *Parity Generation and Check* generates the parity for the upper and lower byte simultaneously and stores the parity bits in bit positions designated DL and DU for lower and upper bytes respectively.

3.38 Parity checking is performed by checking parity of bytes, D0-D7, and D8-D15, and comparing with the respective parity bits DL and DU. A detected parity error activates PAR.ERR* which causes a bus error. Refer to 5.33.

Interrupt

3.39 The 68000 CPU has seven interrupt levels numbered 1 through 7. Level 7 is the highest priority and level 1 is the lowest priority. At any time the 68000 CPU has an interrupt priority number set as part of the μP status register. Interrupts are acknowledged for all priority levels greater than the current μP priority contained in the μP status register. Interrupts are prohibited for all priority levels less than or equal to the current μP priority contained in the process status register. When an interrupt is acknowledged, the μP priority is set to the level of the interrupt request. Figure 3-4 illustrates the interrupt in block diagram.

3.40 A level 7 interrupt is special in that it is acknowledged even if the mask in the 68000 μP status register is set to 7. This means that the level 7 interrupt is a *non-maskable* interrupt. A level 7 interrupt is acknowledged every time the interrupt request changes from a lower level to a level 7, that is, level 7 interrupts are *edge triggered*.

IMPORTANT

> *To avoid confusion for MCL68000L device programmers, the number designation of the interrupt lines of the 796 Bus and the interrupt priorities were made to correspond to the definition of the MCL68000L device. INT7* on the 796 Bus is the highest priority interrupt, and INT1* is the lowest priority. INT0* is not implemented. INT7* is non-maskable and edge triggered, whereas all other interrupts are maskable and level triggered.*

3.41 The 796 Bus Specification defines eight interrupt lines, INT0* through INT7* with INT0* being the highest priority. The standard also recommends that interrupts be level triggered instead of edge triggered to provide for multiple interrupt sources on each interrupt line.

3.42 Option jumpers are provided if alternate interrupt assignments are needed.

3.43 Three interrupt lines INT7*, INT6* and INT5* are option jumpered to on-card devices. The interrupt lines available for system use are:

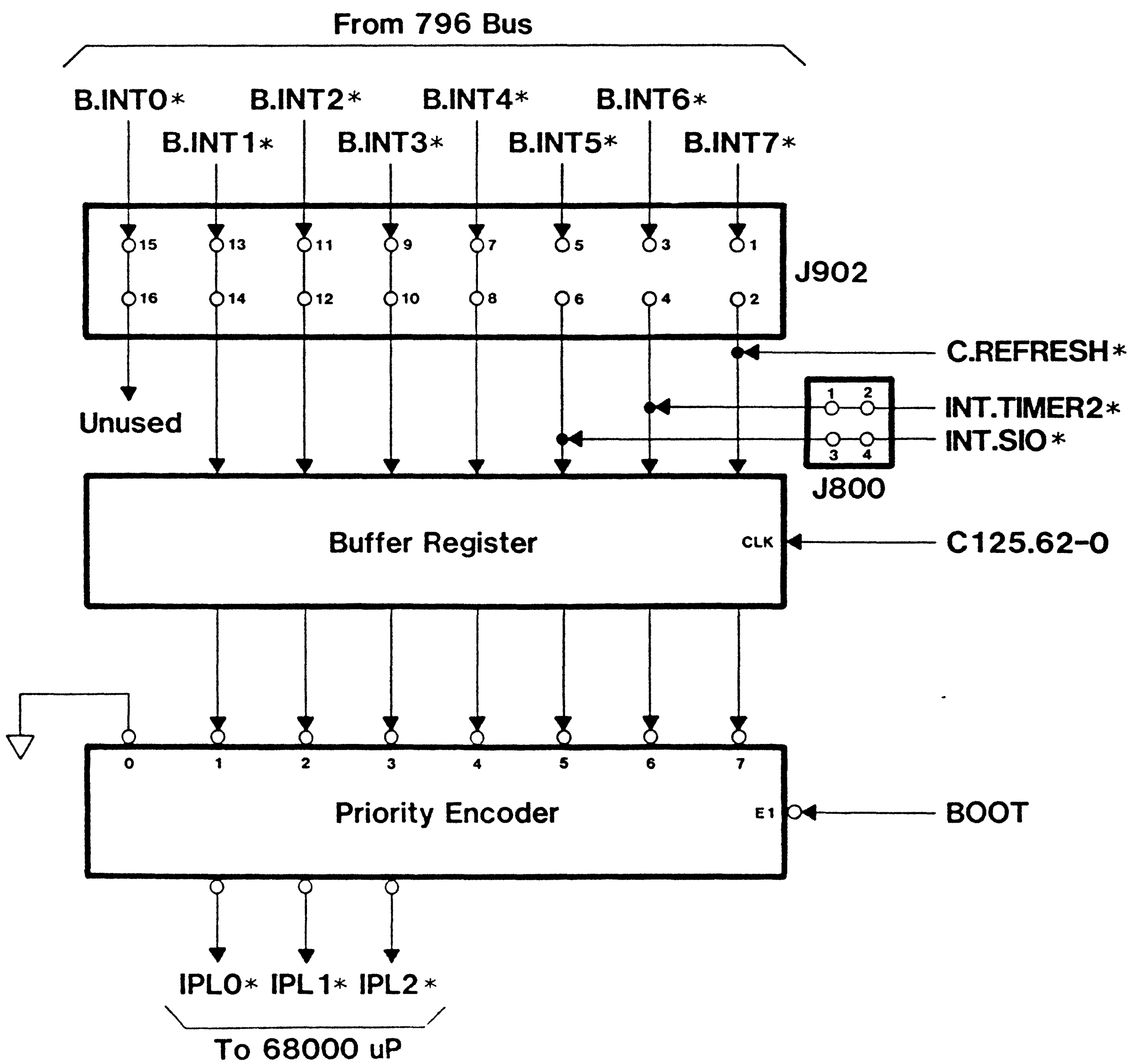


Figure 3-4 -- Interrupt Block Diagram

(1) INT7* — Refresh Timer, C.REFRESH*, highest priority, non-maskable,

(2) INT6* — User Timer, C.TIMER2*,

(3) INT5* — UART A and B, INT.SIO*,

(4) INT4* — unassigned,

(5) INT3* — unassigned,

(6) INT2* — unassigned,

(7) INT1* — unassigned lowest priority, and

(8) INT0* — not available.

3.44 The seven interrupt lines are clocked through the *Buffer Register* to the *Priority Encoder* and output as three encoded lines, IPL0*-IPL2*, to the μP. BOOT is asserted during the boot state to inhibit interrupts.

3.45 The 68000 CPU acknowledges interrupts in an *auto-vector* mode. That is, the 68000 CPU generates the interrupt vector internally rather than it being supplied by the device. Thus the INTA* signal of the 796 Bus is never asserted and the 796 Bus vectored interrupt capabilities are not used.

3.46 The *Dual UART* device provides two asynchronous serial I/O channels to the *RS-423A Drivers* and *Receivers*. Refer to Figure 3-5. Jumper option, J100, provides for Channel B to be configured as a DCE or DTE port. Channel A provides RS-423A control signals RTS, DTR, CTS, and DSR, as well as the serial data lines. Channel B provides the serial data lines only.

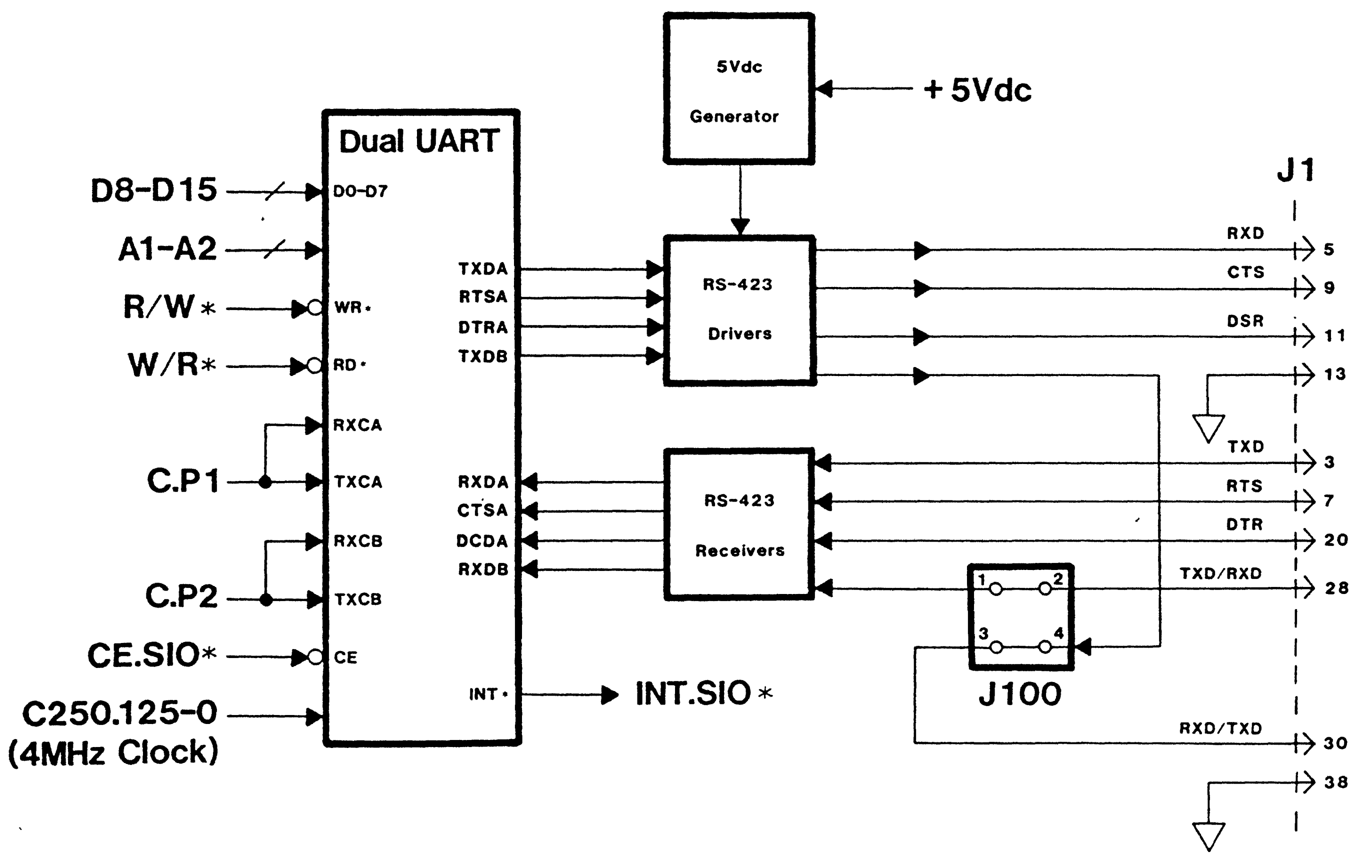


Figure 3-5 — UART Block Diagram

3.47 The use of both UART channels is almost identical, the difference being in the raising of interrupt pending status and the use of the RS-423A control lines on Channel A.

3.48 A 7201 Multi-Protocol Serial Controller device is used for the UART. This device can handle a number of different signal formats and is software programmable.

3.49 The data rate of Channels A and B can be set independently under software control. Two channels from the *Timer*, CP.1 and CP.2, furnish the clocking to the UARTs. Refer to 5.14 and 5.37 for programming of UARTs and Timer.

3.50 The EIA RS-423A specification was selected for several reasons:

(1) The RS-423A is downward compatible with RS-232C,

(2) The RS-423A allows higher data rates than RS-232C,

(3) The RS-423A can support longer cable lengths than RS-232C.

Five Channel Timer

3.51 A 9513 Counter/Timer device is used to implement the *Five Channel Timer*. Refer to Figure 3-6. Four of the five timers are pre-assigned to specific functions on the 68000 CPU. One timer is available for user-programmed timing functions. The timer channels are assigned as follows:

(1) Watchdog Timer, C.TIMER1, furnishes a programmable abort/reset capability in case the μP should unexpectedly halt,

(2) RTC Timer, C.TIMER2, furnishes a user interrupt on a programmably selectable time base. The interrupt may be optionally removed through a jumper option.

(3) Refresh Timer, C.REFRESH, furnishes an interrupt to execute dynamic RAM refresh program,

(4) UART A, C.P1, furnishes UART Channel A clock,

(5) UART B, C.P2, furnishes UART Channel B clock.

3.52 The *Five-Channel Timer* clock, C250.125-0, is a 4 MHz clock derived from a 16 MHz crystal oscillator in the *System Timing* section.

796 Bus Interface

3.53 Figure 3-7 illustrates the 796 Bus Interface in block diagram.

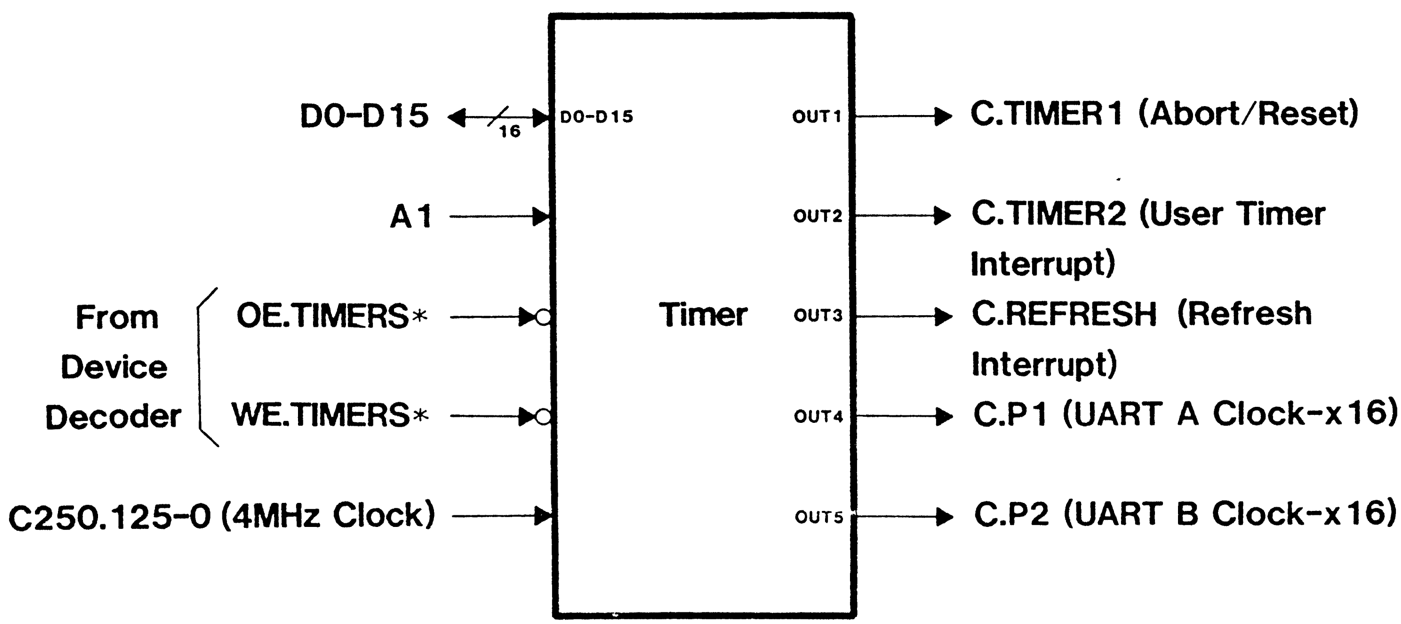


Figure 3-6 – Timer Block Diagram

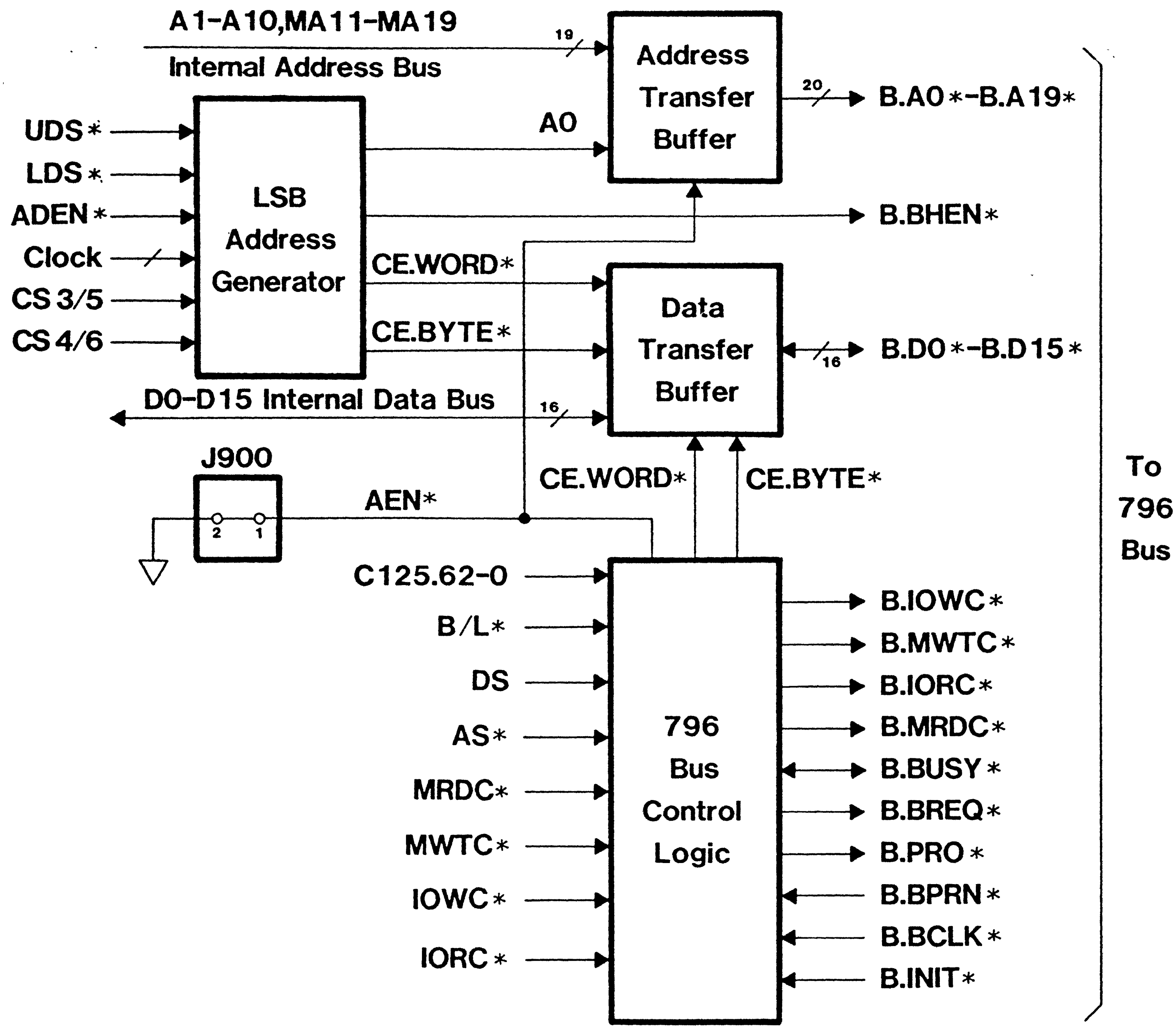


Figure 3-7 — 796 Bus Interface Block Diagram

Parallel Input Port

3.54 The *16-Bit Parallel Input Port* can be used as a general purpose 16-bit input port. Refer to Figure 3-8.

3.55 Several auxiliary lines are extended out to PCA connector, J2, for user applications:

(1) +5 Vdc,

(2) Signal ground,

(3) SET.INIT* pulled low, e.g., external switch closure will initiate a 68000 CPU reset,

(4) M.REF* active low indicates the μP is halted.

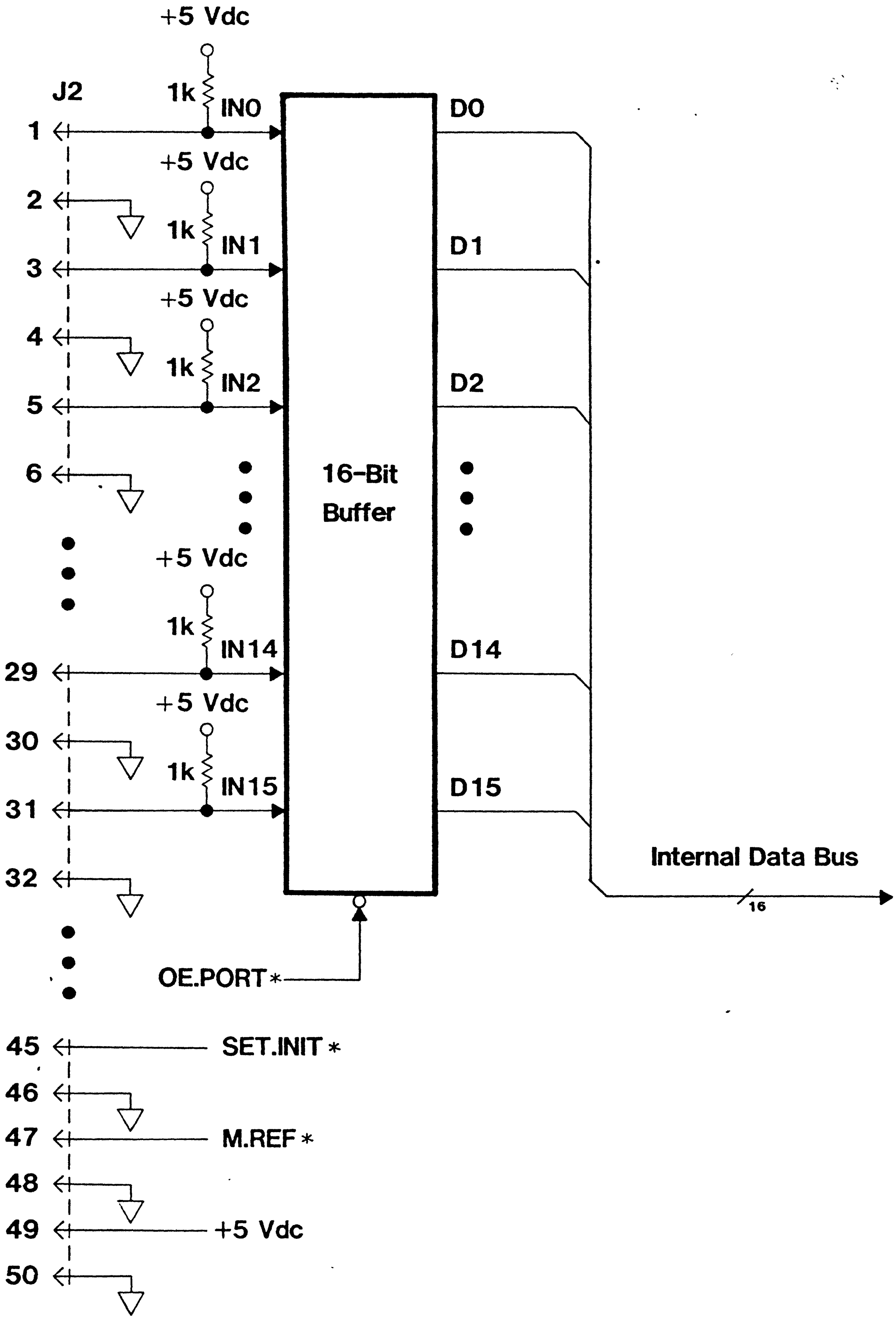


Figure 3-8 — Parallel Input Block Diagram

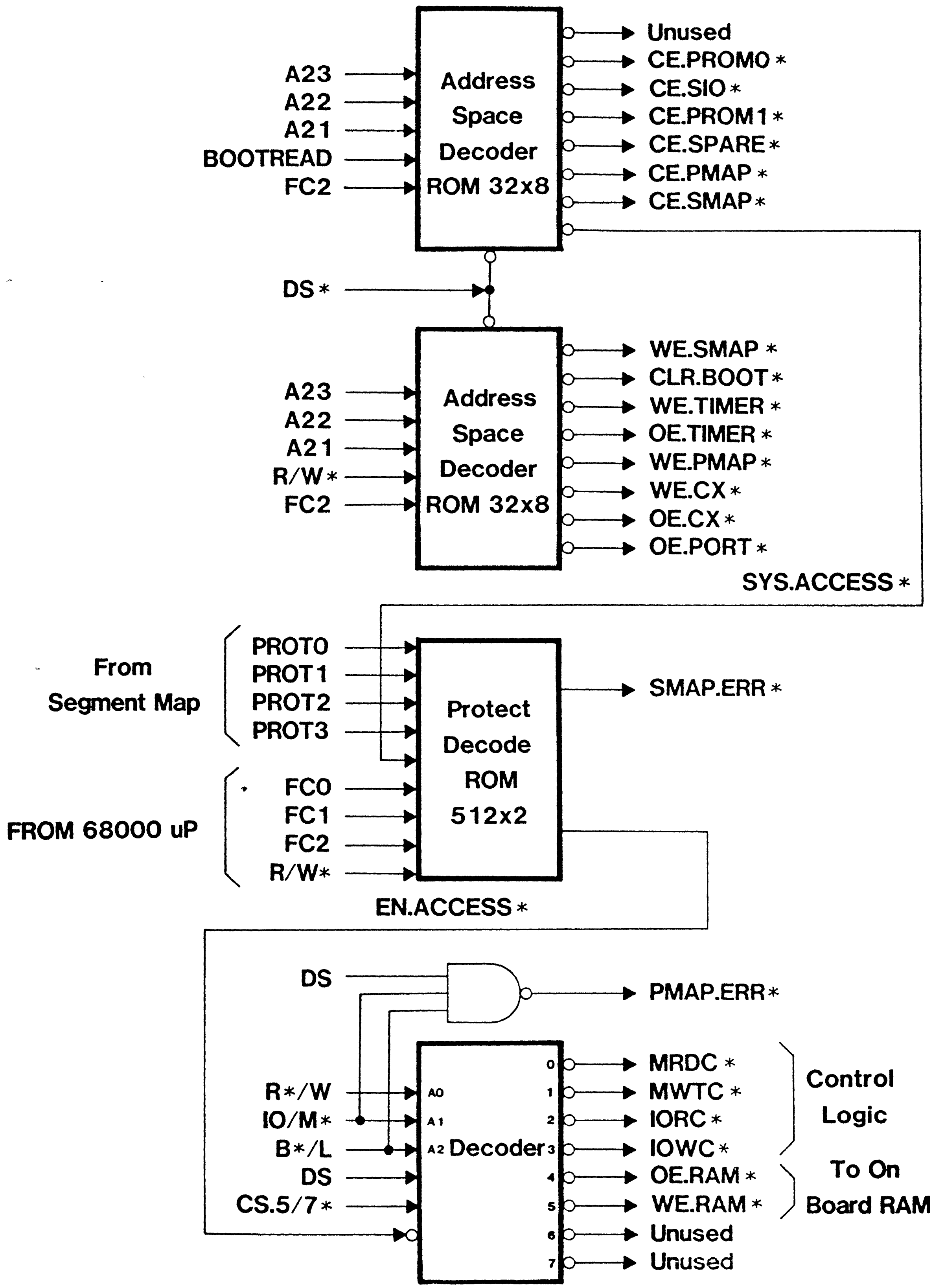


Figure 3-9 — Device Decoder Block Diagram

Device Decode

3.56 The *Device Decoder* is detailed in Figure 3-9. The 68000 CPU internal address and control lines are decoded and used to enable memory management functions:

(1) Write Enable Control Register, WE.CX*,

(2) Output Enable Control Register, OE.CX*,

(3) Write Enable Segment Map, WE.SMAP*,

(4) Chip Enable Segment Map, CE.SMAP*,

(5) Write Enable Page Map, WE.PMAP*,

(6) Chip Enable Page Map, CE.PMAP*,

and to enable five-channel timer functions:

(7) Write Enable Timer, WE.TIMER*,

(8) Output Enable Timer, OE.TIMER*,

and to enable 32k ROM:

(9) Chip Enable ROM0, CE.PROM0*,

(10) Chip Enable ROM1, CE.PROM1*,

and clear the boot state:

(11) Clear Boot, CLR.BOOT*,

and enable the 16-Bit Parallel Input Port:

(12) Output Enable Port, OE.PORT*.

Power-On-Reset

3.57 Refer to Power-On-Reset block diagram, Figure 3-10, for the description which follows. A reset, RESET* and HALT*, can be initiated through several channels.

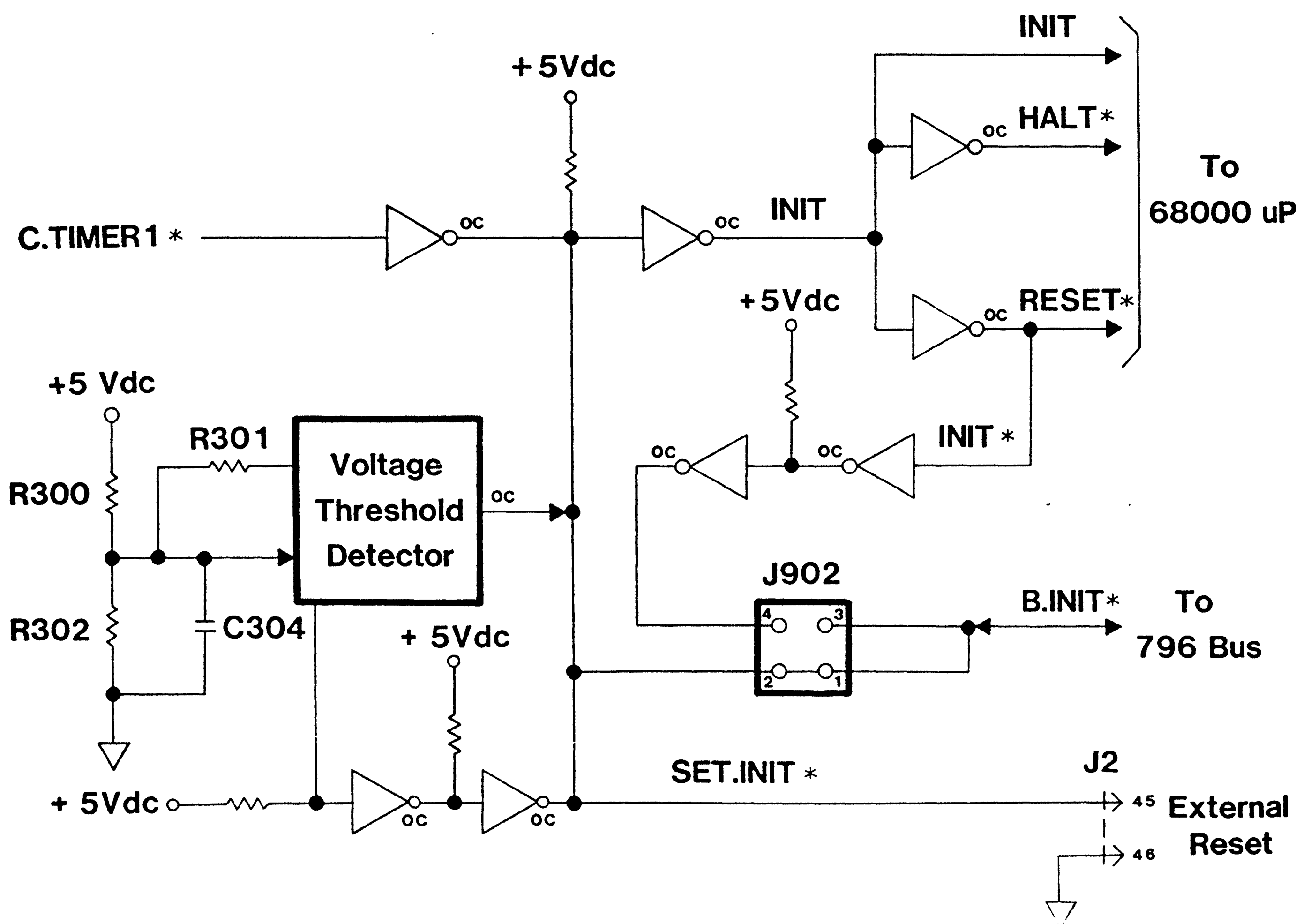


Figure 3-10 — Power-On-Reset Block Diagram

(1) The *Voltage Threshold Detector* monitors the +5 Vdc supply for a change. When the voltage is greater than 4.65 V, the reset is removed. If the voltage falls below 4.25 V, a reset is issued, e.g., when the Mainframe is powered on or the line voltage suddenly drops,

(2) The Watchdog Timer determines that the μP has halted and activates C.TIMER1*,

(3) An INIT* is issued from the 796 Bus. Jumper Option J902 provides for issuing an INIT* to the 796 Bus,

(4) An external switch closure from J2 the *Parallel Input Port Connector*.

System Timing

3.58 A 16 MHz crystal-controlled oscillator is used in the *System Timing* to count down to:

(1) 8 MHz — C125.62-0 for the μP clock,

(2) 4 MHz — C250.125-0 for the *Five Channel Timer*.

Refer to Figure 3-11.

3.59 C.S3|5 through C.S10|12 are generated in a 8-bit Shift Register clocked by the buffered 16 MHz line.

3.60 796 Bus TIMEOUT* is issued from the *4-Bit Counter*.

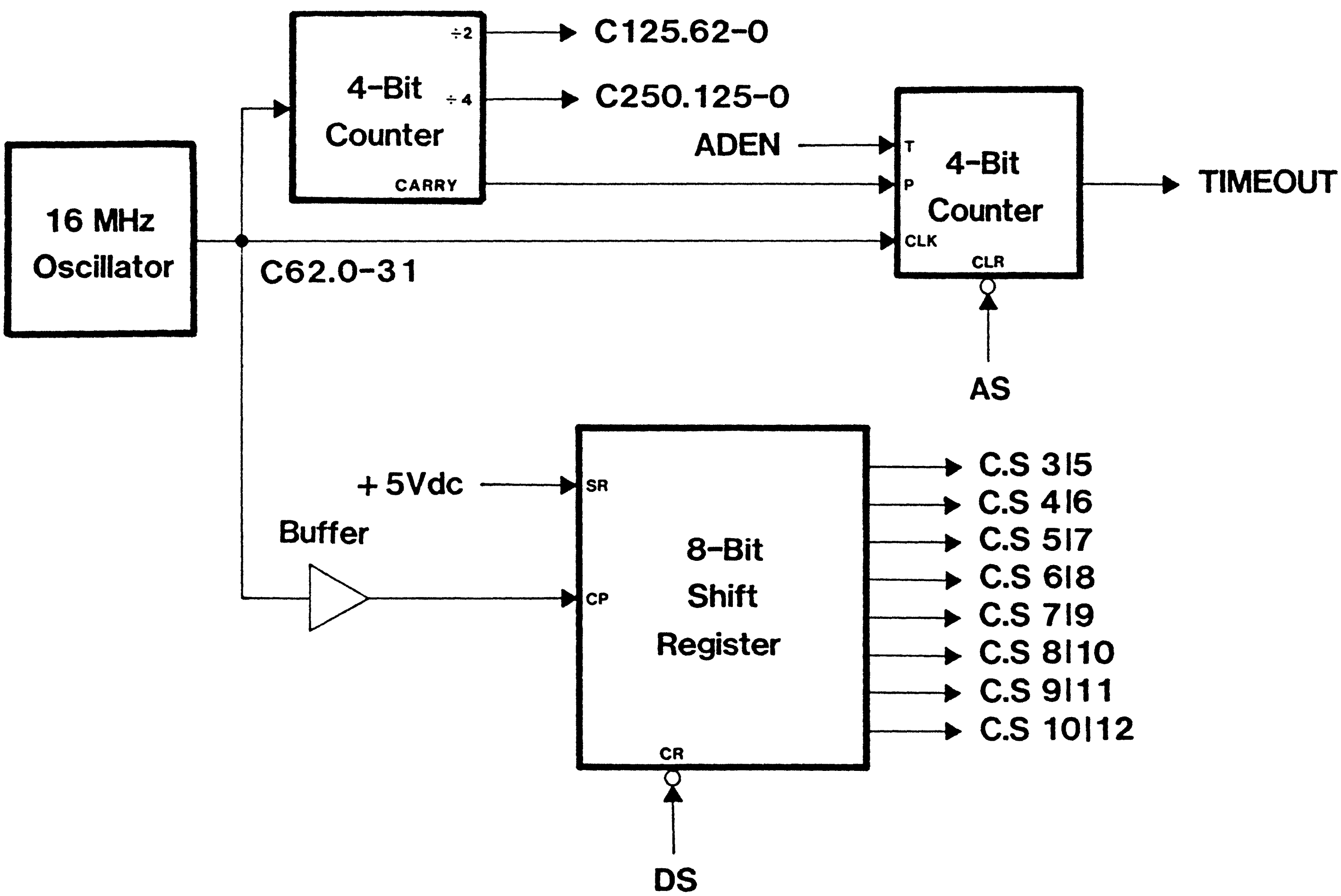


Figure 3-11 — System Timing Block Diagram

Table 3-4 — 68000 Central Processing Unit Active 796 Bus Signals

Diagram Mnemonic	796 Bus Mnemonic (1)	Pin	Function
B.BCLK*	BCLK*	13	Bus Clock
B.INIT*	INIT*	14	Initialize
B.BPRN*	BPRN*	15	Bus Priority In
B.BPRO*	BPRO*	16	Bus Priority Out
B.BUSY*	BUSY*	17	Bus Busy
B.BREQ*	BREQ*	18	Bus Request
B.MRDC*	MRDC*	19	Memory Read Command
B.MWTC*	MWTC*	20	Memory Write Command
B.IORC*	IORC*	21	I/O Read Command
B.IOWC*	IOWC*	22	I/O Write Command
B.BHEN*	BHEN*	27	Byte High Enable
B.CRQ*	CBRQ*	29	Common Bus Request
B.CCLK*	CCLK*	31	Constant Clock
B.INT0 — 7	INT0* — INT7*	35—42	Parallel Interrupt Requests
B.A0* — B.A19*	ADR0* — AD10*	Various	20-Bit Address Bus
B.D0* — B.D16*	DAT0* — DATF*	Various	16-Bit Data Bus

Note:

(1) Address and data bus lines are in hexadecimal notation.

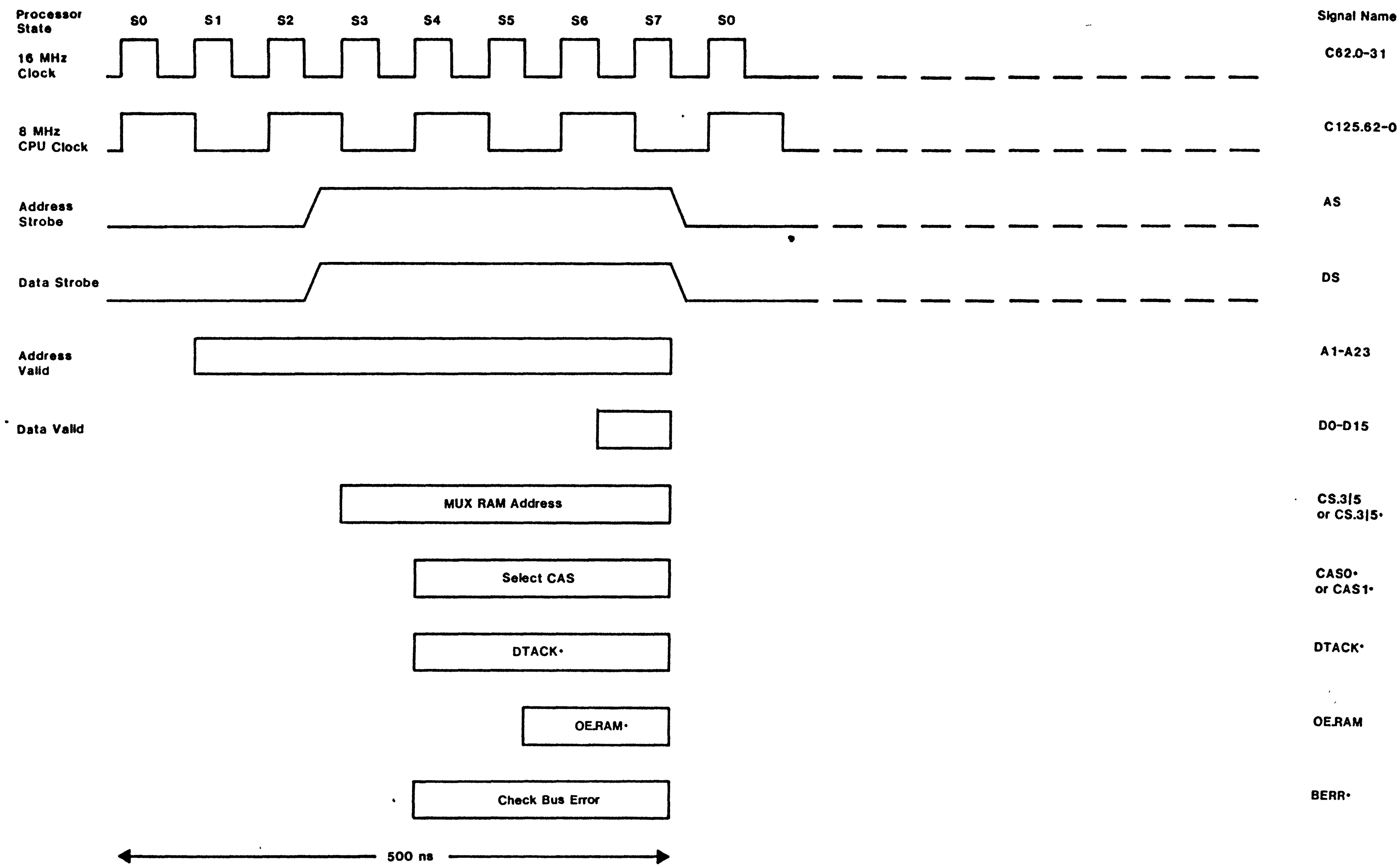


Figure 3-12 — On-Card RAM Read Cycle Timing

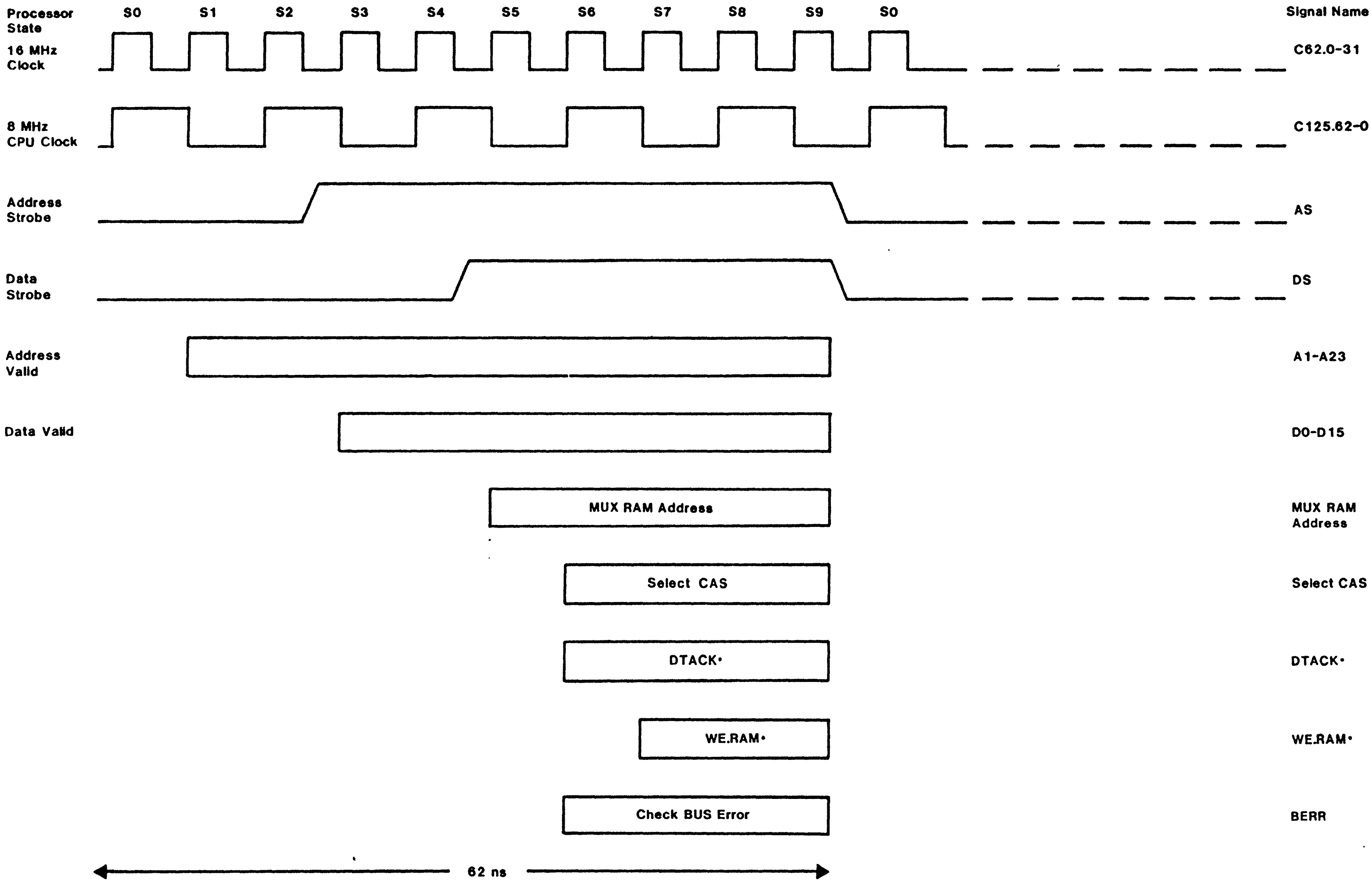


Figure 3-13 — On-Card RAM Write Cycle Timing

05-0004-01

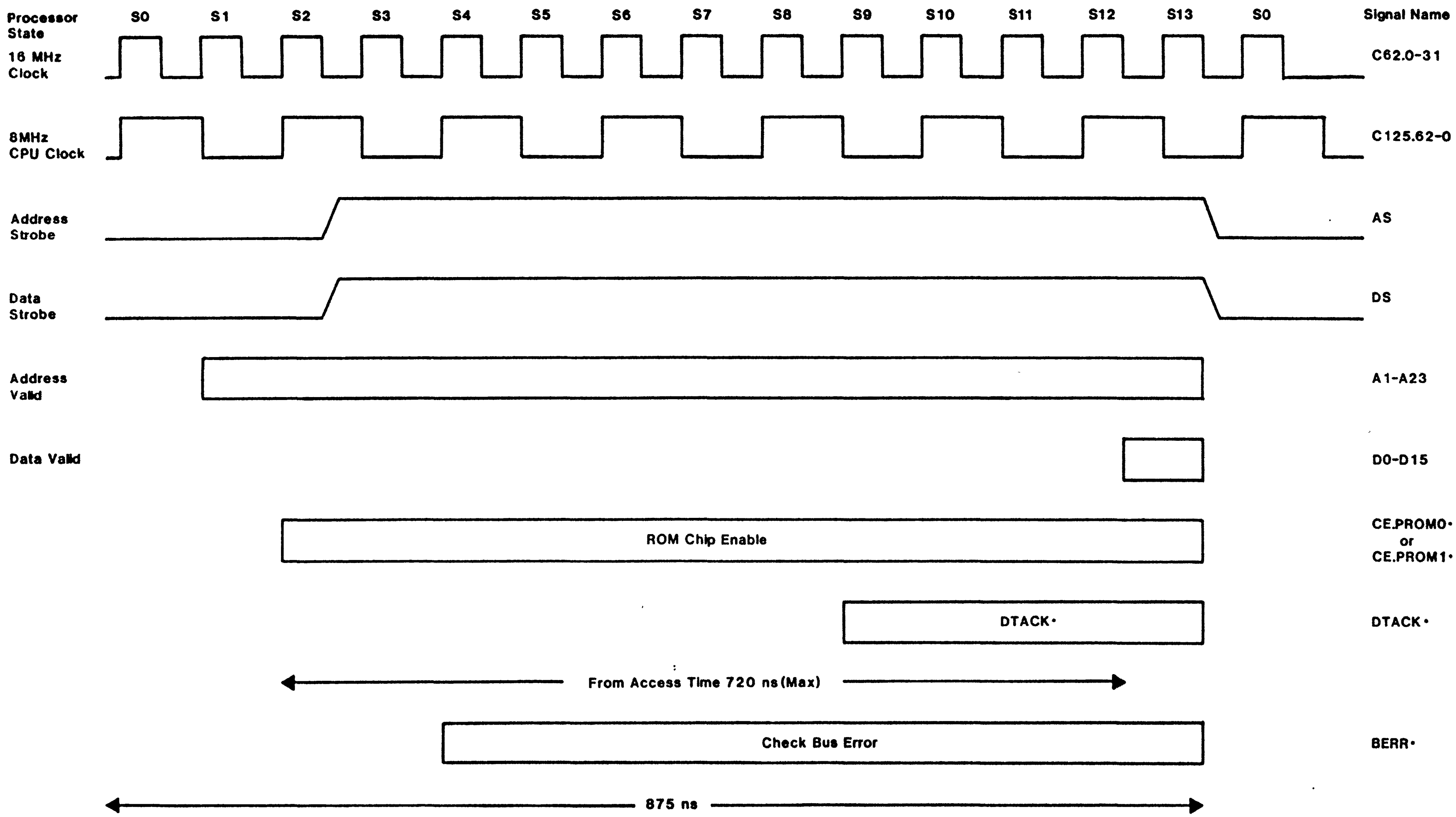


Figure 3-14 — On-Card ROM Read Cycle Timing

05-0004-01

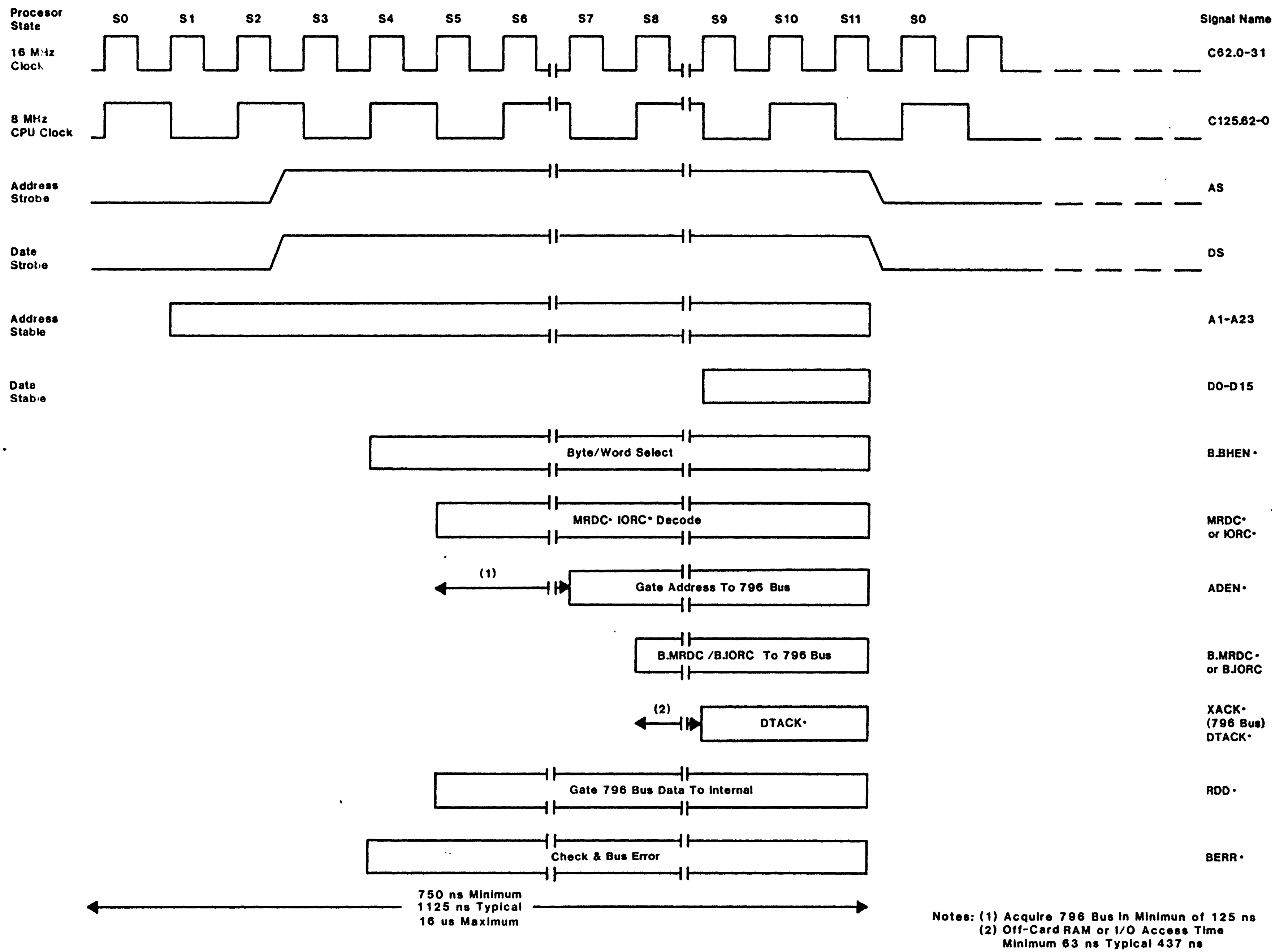
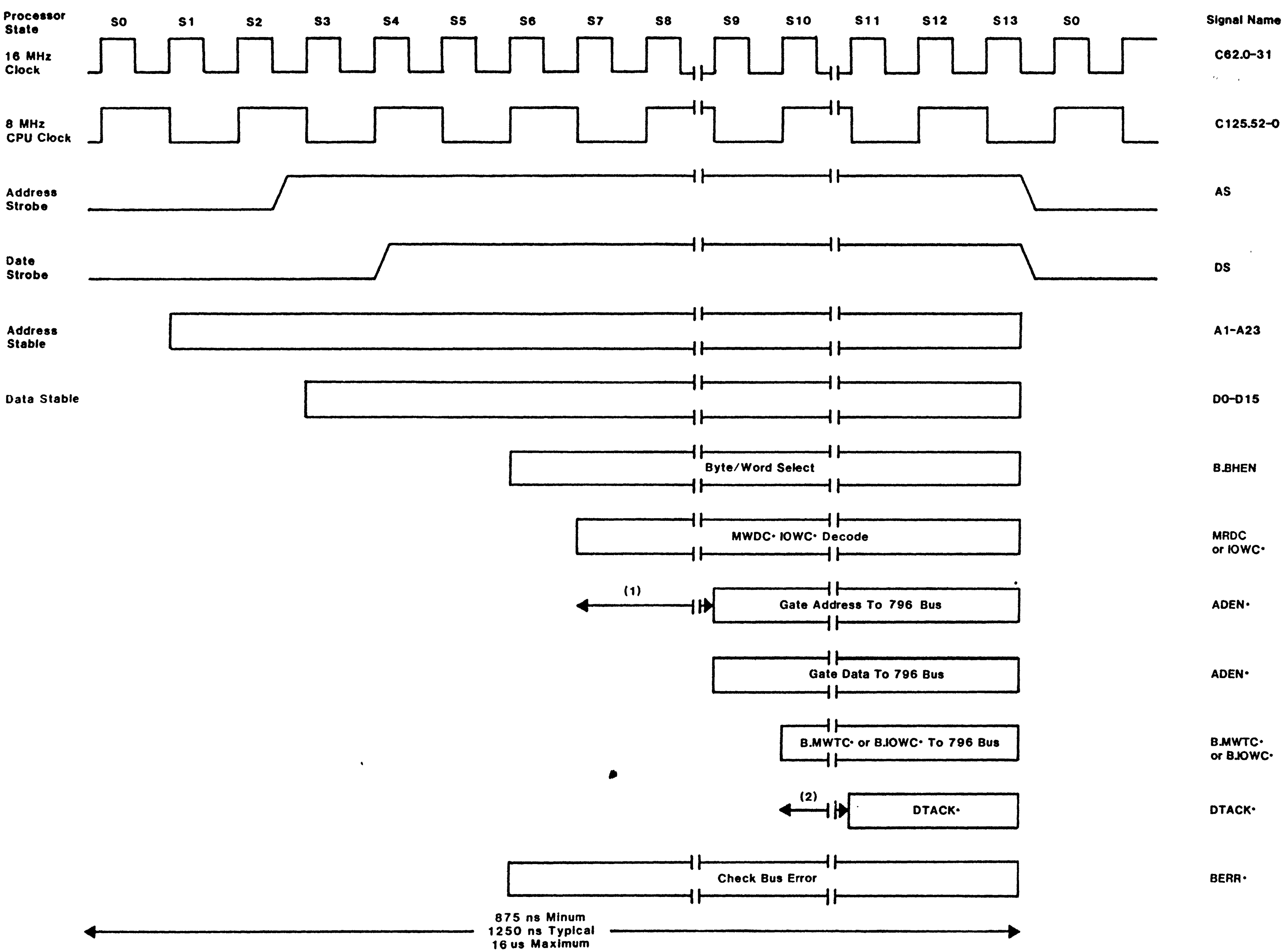


Figure 3-15 — Off-Card RAM and Input Output Read Cycle Timing



Notes: (1) Aquire 796 Bus in Minimum of 125 ns
(2) Off-Card RAM or I/O Access Time is Minimum of 63 ns and Typical 437 ns

Figure 3-16 — Off-Card RAM and Input Output Write Cycle Timing

4. SPECIFICATIONS

4.01 The following furnishes the user with information for shipping and installation and should be used to establish acceptance criteria if they are performed. Minor deviations from the specifications tabulated in Table 4-1 which do not affect the 68000 Central Processing Unit performance are excluded from the Codata Systems Corp. warranty.

Table 4-1 — 68000 Central Processing Unit 92-1012-xx Specifications

PARAMETER	CHARACTERISTICS
Microprocessor	
Device	MCL68000L or Equivalent.
Clock Rate	8 MHz.
Instruction Cycle	500 ns.
Instruction Types	56.
Memory Management	Two level; segmented and paged.
Context Switching	16 users.
Logical Address Size	2M bytes.
Physical Address Size	1M byte — 796 Bus. 256k — on card. Expandable off card to 512k bytes maximum.
Segment Size	32k byte.
Segment Protection	6 levels coded to 16 states.
Page Size	2k bytes.
Page Definition	4 levels. (1) On-card RAM. (2) Invalid Page. (3) 796 Bus RAM. (4) 796 Bus I/O.
Page Control	2 levels. (1) Used. (2) Dirty.
Interrupt Controller	
Device	SN74LS148N.
Operation	Auto-vector.
Levels	Seven.
Priority	7-Memory Refresh or 796 Bus defined. 6-Real Time Clock or 796 Bus defined. 5-UART or 796 Bus defined. 4-796 Bus defined. 3-796 Bus defined. 2-796 Bus defined. 1-796 Bus defined. 0-not available.
Bus Interface	IEEE 796 Bus specification.
Mode	Multi-master; serial or parallel priority.
Address	20 bit.
Data Width	8 or 16 bit.

Table 4-1 — 68000 Central Processing Unit 92-1012-xx Specifications (Continued)

PARAMETER	CHARACTERISTICS
Connector, P1	796 Bus pin assignments.
, P2	Pin assignments for off-board memory expansion.
Timer	Five channel.
Device	AM9513.
Operation, Timer 1	Watchdog Timer.
Operation, Timer 2	RTC Timer.
Operation, Timer 3	Refresh Timer.
Operation, Timer 4	UART A Data Rate Generator.
Operation, Timer 5	UART B Data Rate Generator.
Input Output Ports	
UART Port-Device	NEC7201.
Channels	Two.
Interface	RS-423A asynchronous.
Data Rate	75 to 125k baud.
Data Format	Programmable.
Parallel Input Port	One.
Device	SN74LS244N.
Interface, number	16 bit.
, level	TTL.
Memory	
Data Width	8 or 16 bit.
Random Access Memory	
Type	64k bit dynamic.
Size	256k byte on-card.
Expandable	256k byte off-card.
Read Only Memory	
Type 2716	8k byte.
Type 2732	16k byte.
Type 2764	32k byte.
Reset, Vcc Sense	4.65 Vdc ±1%
, Watchdog Timer	Time Interval is user programmable.
, 796 Bus INIT*	Jumper option; master or slave.
, External	Switch contact closure.
PCA Dimensions	796 Bus Specification.
Length	30.5 cm (12.0 inch).
Width	17.1 cm (6.75 inch).
Spacing	1.3 cm (0.5 inch).

Table 4-1 — 68000 Central Processing Unit 92-1012-xx Specifications (Continued)

PARAMETER	CHARACTERISTICS
Environment	
Temperature	
Operating	0°C to 55°C (32°F to 131°F).
Storage	0°C to 65°C (32°F to 149°F).
Humidity	5% to 90%, noncondensing.
Power Requirements	
+5 Vdc Bus	2.5 A.
Weight	454 g (16 oz.).

Table 5-1 68000 Central Processing Unit Options – P/N 92-1012-xx

Options		Flag 01	02	03			Description
							Serial Port
J100-1	J100-2		x				Connects P2.RXD as DTE.
J100-3	J100-4		x				Connects P2.TXD as DTE.
J100-1	J100-3						Connects P2.RXD as DCE.
J100-2	J100-4						Connects P2.TXD as DCE.
							ROM Type Select
J100-5	J100-6		x				Connects U100 . . . U104 (23) to VCC for 2716.
J100-7	J100-8						Connects U100 . . . U104 (23) to A12 for 2732/2764.
							Interrupt Select
J800-1	J800-2		x				Connects INT.TIMER* to INT6*.
J800-3	J800-4		x				Connects INT.SIO* to INT5*.
							796 Bus Signals
J900-1	J900-2		x				AEN* to GND. Connect for operation with 8289.
J901-1	J901-2		x				Receive B.INIT* from 796 Bus.
J901-3	J901-4		x				Drive B.INIT* on 796 Bus.
J901-5	J901-6		x				Drive BCLK* to 796 Bus.
J901-7	J901-8		x				Grounds BPRN* for Highest Master in Chain.
J901-9	J901-10		x				Drive CCLK* to 796 Bus.
							Interrupt Level Assignment
J902-1	J902-2		x				B.INT7* to INT7* Non-maskable Interrupt used by Refresh Timer.
J902-3	J902-4		x				B.INT6* to INT7* User Timer.
J902-5	J902-6		x				B.INT5* to INT5* UART.
J902-7	J902-8		x				B.INT4* to INT4*.
J902-9	J902-10		x				B.INT3* to INT3*.
J902-11	J902-12		x				B.INT2* to INT2*.
J902-13	J902-14		x				B.INT1* to INT1*.
J902-15	J902-16		x				(Not Used).
							Memory Expansion Board
J903-1	J903-2						Drives MCAS0* from MCAS1*.
J903-3	J903-4						Drives MCAS2* from MCAS3*.

Option	Flag 01	02	03			Description: Mnemonic	Codata Part Number
U101		x				MON-0	27-0019-01
U103		x				MON-E	27-0020-01
U602		x				P1	27-0021-01
U502		x				P0	27-0022-01
U503		x				P2	27-0023-01

5. OPERATION AND PROGRAMMING

Options

5.01 Before the 68000 CPU is installed into the Mainframe card cage, the PCA options should be verified.

(1) Check the J100, J800, J900, J901, J902 and J903 jumper options. Refer to Table 5-1.

(2) Check the U101, U102, U103, U602, U502 and U503 ROM. Refer to Table 5-1.

5.02 Install the PCA into card cage position 4 and connect the serial I/O cable to J1.

RS-423A

5.03 When the serial I/O port(s) is connected to RS-232C compatible devices some restrictions apply:

(1) Data rates and cable lengths must be restricted to those allowed under RS-232C. Data rates to 9600 baud maximum and cable lengths to 50 feet maximum.

(2) Signal level of RS-232C drivers kept to ±12 Vdc or less.

(3) Rise times of RS-423A drivers must be set to meet RS-232C specifications.

The 68000 CPU is designed to operate with RS-232C devices at data rates up to 9600 baud.

5.04 When the 68000 CPU is connected to other RS-423A compatible devices, cable lengths can be increased to 4000 feet maximum at transmission rates to 3000 baud. At higher data rates, cable length must be reduced. For example, transmission at 9600 baud requires cables no longer than 40 feet.

5.05 Transmission at high data rates requires careful design of cables and system grounding as well as adjustment of driver rise times by selecting timing capacitors on the 68000 CPU. The standard configuration of the 68000 CPU allows transmission at rates up to 40k baud with cable lengths up to 250 feet. For longer cables or higher baud rates, the standard capacitor values must be changed to change the driver rise and fall times. The system designer is strongly urged to consult EIA RS-423A specification to select the optimum value needed for the particular application.

Power-On-Reset — Entering Boot State

5.06 Each time the Mainframe is powered on or an operator keys the Mainframe reset switch, the ROM software on the 68000 CPU is used to correctly initialize the system. The term used to describe the state of the system after reset is called *Boot State*. Boot State is only entered through hardware reset. The system exits boot state by executing a *Clear Boot State* operation by writing a data word of 0001H to memory location 200 000H. At the time boot state is cleared, the parity checking function of the 68000 CPU is enabled by setting bit D0 to a 1 while issuing a write to memory location 200 000H. If D0 is set to 0 while this write occurs, the parity checking circuit will be disabled. The only time the parity checking circuit is enabled is at the time of clearing Boot State.

NOTE

The current release of the 68000 CPU does not require any particular data value to the output during the write to memory location 200 000H. Future releases of the 68000 CPU provide for enable/disable of the parity checking function with bit D0, 1 = enable. To maintain software compatibility with future board configurations, it is advised that bit D0 be set when issuing the command to exit boot state.

5.07 During the boot state, the 68000 CPU operation differs from the non-boot state.

(1) One pair of ROMs, designated ROM0 at address space 200 000H through 3FF FFFH, overlays RAM starting at location 000 000H. Thus the initial program counter and stack pointer are fetched from ROM0 locations 0 through 7H. ROM0 is still accessible at its regular address. The bootstrap code may execute from normal ROM addresses and thus be used in the non-boot state as well. ROM0 must also contain a valid set of exception vectors and the firmware to handle the exceptions in the low order locations in case an exception occurs during the boot state.

(2) Access to the on-card RAM and the 796 Bus are disabled except for write access to the on-card RAM. This provides a way to move the exception and interrupt vectors from ROM to RAM during the boot state sequence. In addition, RAM can also be initialized to parity error free data values before RAM read accesses are allowed. The Context Register, Segment Map and Page Map must be initialized at a minimum of Page 0 Segment 0. These must be defined before the exception vectors are copied to RAM.

(3) All interrupts, including the non-maskable interrupt, are disabled by the hardware. After exiting from the boot state, the non-maskable interrupt can occur any time, and maskable interrupts can occur as soon as the interrupt mask in the μP status register is lowered to allow them.

Memory Map Initialization

5.08 The memory maps of the Memory Management must be initialized with valid protect codes from logical addresses to virtual addresses to physical addresses. The minimum assignment is logical Page 0 of logical Segment 0. All context values must be defined before the exception and interrupt vectors can be moved to on-card RAM or 796 Bus RAM space during the boot state.

5.09 Usually the logical address to physical address mapping allowing access to on-card RAM and 796 Bus RAM and I/O space is first defined during boot state to provide a means to initialize all RAM in the system to a parity error-free state.

Copying The Exception and Interrupt Vectors to RAM

5.10 Boot state ROM0 must contain the exception and interrupt vectors. These are copied into RAM starting at 000 000H. The copying procedure is:

(1) Read from ROM.

(2) Write to RAM at the same address for read and write. Refer to 3.09.

On-Card RAM Initialization

5.11 The on-card RAM is byte parity checked on each read [illegible] the CPU is in boot state, all read accesses are inhibited and the parity checking disabled. As soon as the boot state is exited, any reads of the RAM for instructions, exception or interrupt vectors, stack reads or data variable reads may result in a parity error if the RAM has not been initialized by a prior write operation. The entire RAM should be written to ensure that parity is set properly.

RAM Refresh

5.12 The on-card RAM is dynamic and must be periodically refreshed. This refresh operation must be performed with a software routine started every 2 ms by an interrupt from Timer Channel 3. Refer to 5.35 for the Timer programming instructions. The RAM refresh routine consists of a series of 127 NO-OPS with a return from exception (RTE) instruction at the end. The vector to the interrupt routine is located at logical address 000 07CH and points to the actual refresh routine which may be located in on-card RAM, ROM or 796 Bus RAM.

Exiting Boot State

5.13 To exit from the boot state, a write operation is performed to location 200 000H. Once this has been performed, the previously set up refresh interrupt will occur every 2 ms and on-card RAM can now be read/write accessed and 796 Bus RAM and I/O can be accessed.

UART Programming – General

5.14 Each channel of the Dual Channel UART must be set up individually to establish:

(1) Asynchronous Mode,

(2) Word length,

(3) Number of stop bits,

(4) If parity is to be used and the type,

(5) If interrupts are to be used and how,

(6) Multiply factor for reference clock to data rate.

In addition, Timer Channels 4 and 5 must [illegible] programmed to provide the proper reference clock for each UART channel. Refer to 5.35 for Timer programming instructions.

5.15 The device used to perform the UART function is a NEC PD7201. This device is capable of both asynchronous and bisynchronous operation. This particular hardware implementation on the 68000 CPU requires the device to operate in only the asynchronous mode.

UART Programming — Control Registers

5.16 After a system reset or a program-issued reset, the control registers must be rewritten before data is transmitted or received. The Tx outputs will be in the marking state after a reset.

5.17 The control information is entered into the control registers of the UART in sets of two consecutive bytes and stored into either of the control registers of Channel A or Channel B. Figure 5-1 lists the UART address for the descriptions which follow.

NOTE

> ***All addressing to the UART should be made only in byte mode because the UART is an 8-bit wide device.***

```
UartdatA        equ        $600000   ;UART A Data Register
UartdatB        equ        $600004   ;UART B Data Register
UartAc          equ        $600002   ;UART A Control Register
UartBc          equ        $600006   ;UART B Control Register
```

The following values represent a typical asynchronous case:

```
WReg1           equ        $0        ;No interrupts on Rx or Tx
WReg2           equ        $0        ;Non-dma mode
WReg3           equ        $E1       ;8-bit Rx, Rx enabled
WReg4           equ        $44       ;16x clock, 1 stop bit, no parity
WReg5           equ        $E8       ;8-bit Tx, no break character
```

Figure 5-1 — UART Register Values

```
move.b      #$18,Uartac.l       ;To reset Channel A
move.b      #$18,Uartac.l       ;Issued twice to insure
                                  synchronization
```

or

```
move.b      #$18,Uartbc.l       ;To reset Channel B
move.b      #$18,Uartbc.l       ;Issued twice to insure
                                  synchronization
```

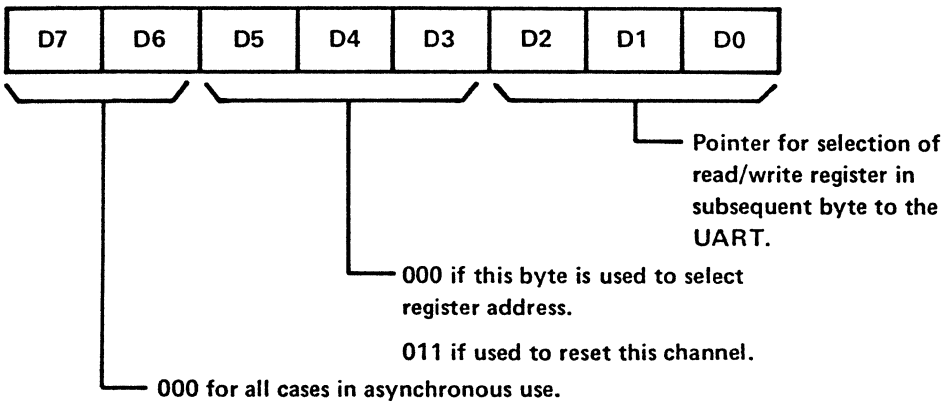


Figure 5-2 — Write Register 0 Routine and Register Map

```
move.b      #1,Uartac.l         ;Selects Register 1, Channel A
move.b      #WReg1aUartac.l     ;Sets contents in Write Register 1,
                                  Channel A synchronization
```

or

```
move.b      #1,Uartbc.l         ;Selects Register 1, Channel B
move,b      WReg1,Uartbc.l      ;Sets contents in Write Register 1,
                                  Channel B synchronization
```

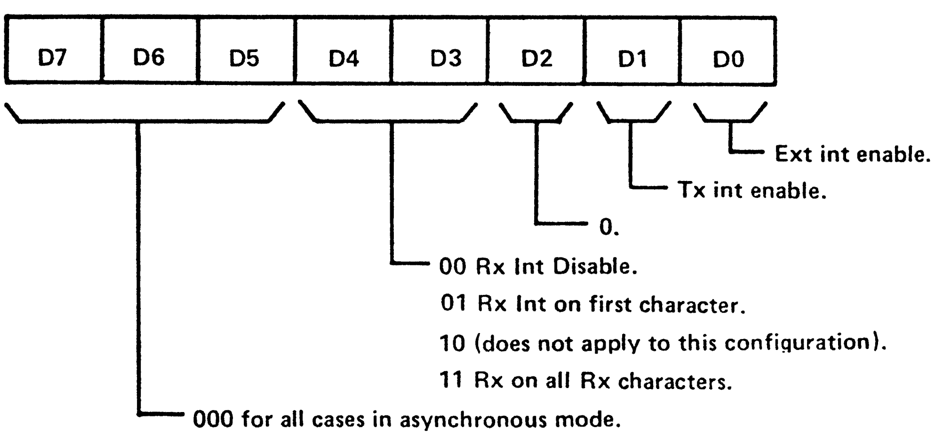


Figure 5-3 — Write Register 1 Routine and Register Map

```
move.b      #2,Uartac.l          ;Select Register 2
move.b      #WReg2,Uartac.l      ;Set contents of Write Register 2
```

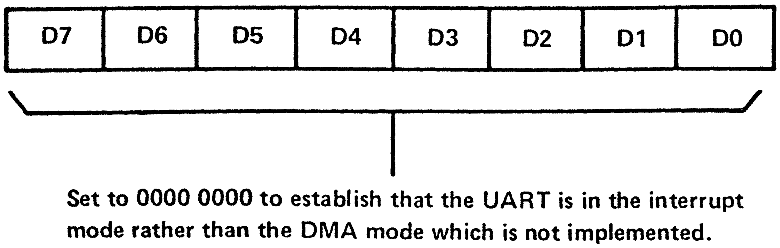


Figure 5-4 — Write Register 2 Routine and Register Map

5.18 Write Register 0 is used to perform a reset for the selected channel or to provide the register address for the second byte of a two byte control set. Refer to Figure 5-2.

5.19 Write Register 1 is used to establish the time when interrupts will be generated if interrupts are enabled for this channel. Refer to Figure 5-3.

5.20 Write Register 2 is used to specify that both channels are in the interrupt mode. This register is accessed only through UART A. Refer to Figure 5-4.

5.21 Write Register 3 establishes word length of received data and allows enable/disable of the receive function. Refer to Figure 5-5.

```
move.b      #3,Uartac.l          ;Select Register 3 for Channel A
move.b      #WReg3,Uartac.l      ;Set contents of Write Register 3
```

or

```
move.b      #3,Uartbc.l          ;Select Register 3 for Channel B
move.b      #WReg3,Uartbc.l      ;Set contents of Write Register 3
```

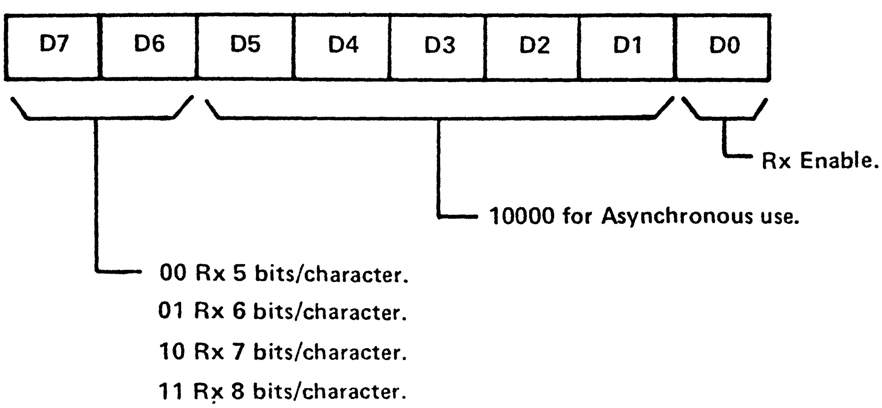


Figure 5-5 — Write Register 3 Routine and Register Map

5.22 Write Register 4 is used to enable parity, define parity type, define number of stop bits used and select the multiplier factor between the data rate and the incoming reference clock. Refer to Figure 5-6.

```
move.b     #4,Uartac.l          ;Select Register 4, Channel A
move.b     #WReg4, Uartac.l     ;Set contents of Write Register 4
```

or

```
move.b     #4,Uartbc.l          ;Select Register 4, Channel B
move.b     #WReg4, Uartbc.l     ;Set contents of Write Register 4
```

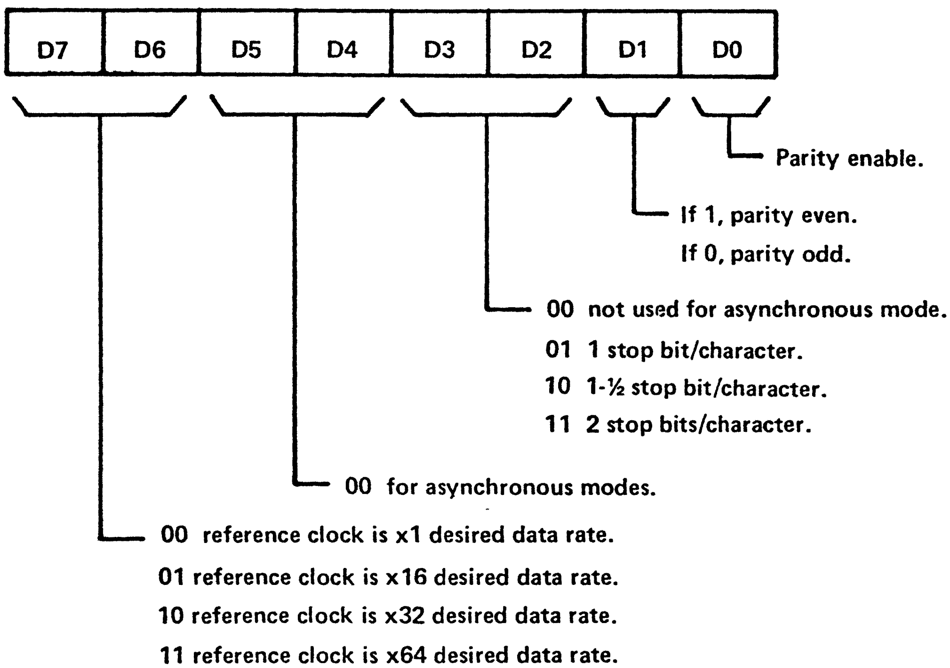


Figure 5-6 — Writer Register 4 Routine and Register Map

5.23 Write Register 5 establishes the word length of transmitted data and allows enable/disable of the transmit function. Refer to Figure 5-7.

```
   move.b     #5,Uartac.1            ;Select Register 5, Channel A
   move.b     #WReg5,Uartac.1        ;Set contents of Write Register 5
or

   move.b     #5,Uartbc.1            ;Select Register 5, Channel A
   move.b     #WReg5,Uartbc.1        ;Set contents of Write Register 5
```

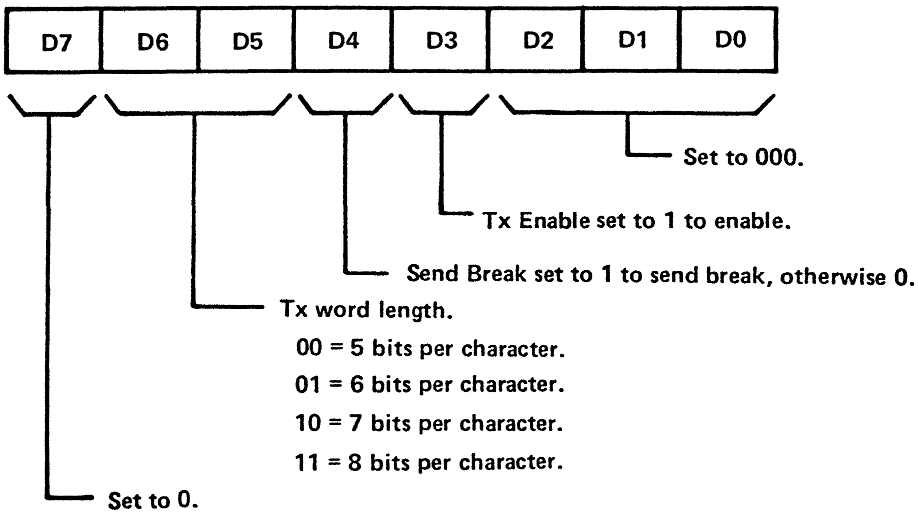
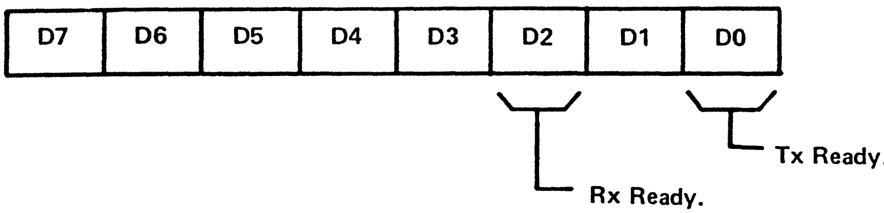


Figure 5-7 — Write Register 5 Routine and Register Map

UART Programming — Status Register

5.24 The status of each UART is obtained with the routine listed in Figure 5-8.

```
    move.b      Uartac.l,RO             ;Read UART A status
or
    move.b      Uartbc.l,RO             ;Read UART B status
```



Note:

(1) Bit positions not specified have undefined values.

Figure 5-8 — Read UART A or UART B Status Routine and Register Map

UART Programming — Transmit/Receive Data

5.25 Data is sent to UART A or UART B with the routine listed in Figure 5-9.

```
    move.b      DO,UartDatA.l           ;Output data to UART A from
                                         Register 0
or
    move.b      DO,UartDatB.l           ;Output data to UART B from
                                         Register 0
```

Figure 5-9 — Send Data To UART A or UART B Routine

5.26 Data is received from UART A or UART B with the routine listed in Figure 5-10.

```
    move.b      UartDatA.l,DO           ;Input data from UART A to
                                         Register 0
or
    move.b      UartDatB.l,DO           ;Input data from UART B to
                                         Register 0
```

Figure 5-10 — Receive Data From UART A or UART B Routine

16-Bit Parallel Input Port

5.27 This port can be used for a general purpose input port of 16 TTL-compatible lines or switch closures to ground. Each line is pulled up to +5 Vdc through a 1k Ohm resister. The port is accessed by reading from location E00 000H. Note that this is the same address used to write the Context Register. Refer to Figure 3-3, Register Bit Map.

Exceptions

5.28 When a μP cycle cannot be completed normally, an exception process is performed. In addition to the exceptions caused by the internal processes of the 68000 CPU such as divide-by-zero or a word reference to a byte address, a number of external conditions can abort the current instruction or bus cycle. External or Bus Error exceptions arise from one of five conditions:

(1) System space error,

(2) Segment map error,

(3) Page map error,

(4) Timeout error,

(5) Parity error.

System Space Error

5.29 The on-card system facilities such as the Page Map, Timer and UART, etc., are only accessible by a process running in the supervisor state. Any attempt to use a logical address greater than 1FF FFFH in user mode causes an exception.

Segment Map Error

5.30 Segment Map Error occurs when the type of access to a particular segment is incompatible with the access attributes associated with that segment. For example, a process may try to write into a segment that has execute-only access associated with it.

Page Map Error

5.31 A Page Map Error occurs when any access is attempted and its associated Page Map entry has the address space control bits set to invalid page.

Timeout Error

5.32 Timeouts occur for off-card accesses to the 796 Bus that are not acknowledged within 15 μs. The most common reasons for this error are that non-existent memory or input output devices have been accessed. There are no timeouts for on-card memory references because in the synchronous on-card bus, all cycles are acknowledged.

Parity Error

5.33 The on-card 256k RAM is 16 bits wide and is divided into two 8-bit bytes with a parity bit appended to each byte. Odd parity for each byte is set on all write to on-card RAM. Each time a read from on-card RAM is performed, the parity of each 8-bit byte is checked. If the parity is incorrect, a Parity Error occurs. The parity error is clocked into a flip flop to carry over the parity error information into the next 68000 cycle forcing a bus error on that next cycle. The parity check functions on both read and read/write memory cycles. Parity errors are cleared on any subsequent memory write cycle, such as, the stacking sequence in response to the parity caused bus error. The parity circuit is enabled by setting a 1 in bit D0 while performing the clear Boot State operation of writing to location 200 000H.

NOTE

> *After power-on or hardware reset, the RAM is in a random state. To avoid parity errors from reading previously unwritten memory, the entire memory should be written to a known state during the initialization sequence.*

Exception Handling

5.34 When a bus error occurs, the source of the error can be determined by the following algorithm:

(1) If the access was to system input output, the only possible exception is that the access was attempted in user mode.

(2) Then check if the operation violated segment access attributes,

(3) Then check if a nonresident page was accessed,

(4) If none of the above conditions caused the exception, then the cause depends upon the setting of the address space control bits in the Page Map.

(5) If the address was an off-card 796 Bus access, the access was aborted due to bus timeout, no XACK* with 15 μs.

(6) If the access was an on-card access, a parity error occurred in the previous read cycle.

Timer Programming – General

5.35 The Five-Channel Timer section of the 68000 CPU provides the following dedicated functions:

(1) Timer 1 — User Programmable Watchdog Timer,

(2) Timer 2 — User Real Time Clock (RTC),

(3) Timer 3 — Memory Refresh Clock, nonmaskable,

(4) Timer 4 — UART A Clock,

(5) Timer 5 — UART B Clock.

5.36 The device used for the Timer is an AMD 9513. The AMD 9513 is a general purpose counter/timer and has many possible operating modes. The particular hardware implementation of this timer on the 68000 CPU requires that the timers be used in only one of three ways:

(1) The Watchdog Timer is used as a programmable, retriggerable timer.

(2) The RTC and Memory Refresh Timers are used as programmable interval timers between interrupts.

(3) The UART A and B Timer is used as a programmable square wave generator.

Timer Programming – Initialization

5.37 The timer device should be set to a known state after a Power-On-Reset. Figure 5-11 illustrates a routine for performing this operation.

```
CtrInit

  move.w     #CtReset,CtrCmd.l      ;Reset Timer device
  move.w     #LoadAll,CtrCmd.l      ;Set all Timers to 0
  move.w     #Ct16Bus,CtrCmd.l      ;Set Timer to 16 bit mode
```

Figure 5-11 – Timer Initialization Routine

Timer Programming — Watchdog Timer Setup

5.38 The Watchdog mode establishes a time period after which an Abort/Reset will be issued to the 68000 CPU. Normally the operating software restarts the Watchdog Timer before this period has expired. The setup or restart routine for the Watchdog mode is illustrated in Figure 5-12.

```
WatchSet

   move.w     #Ct1Load,CtrCmd.l      ;Address Timer 1
   move.w     #CtrMode,CtrDat.l      ;Set to square wave mode
   move.w     #CtrPrd,CtrDat.l       ;Set to appropriate period
   move.w     #Ct1LdArm,CtrCmd.l     ;Load & arm Timer 1
   move.w     #Ct1Clr,CtrCmd.l       ;Clear Timer 1 output bit
```

Figure 5-12 — Watchdog Timer Set Up Routine

Timer Programming — RTC and Refresh Timer Setup

5.39 Figure 5-13 illustrates the routines for setting up the two interrupt timers.

```
RTCSet                               ;Real Time Clock

   move.w     #Ct2Load,CtrCmd.l      ;Address Timer 2
   move.w     #CtrMode,CtrDat.l      ;Set to Timer mode
   move.w     #CtrPrd,CtrDat.l       ;Set to appropriate period
   move.w     #Ct2LdArm,CtrCmd.l     ;Load & arm Timer 2
   move.w     #Ct2Clr,CtrCmd.l       ;Clear Timer 2 output

RefshSet                             ;Refresh Timer

   move.w     #Ct3Load,CtrCmd.l      ;Address Timer 3
   move.w     #CtrMode,CtrDat.l      ;Set to Timer mode
   move.w     #CtrPrd,CtrDat.l       ;Set to appropriate period
   move.w     #Ct3LdArm,CtrCmd.l     ;Load & arm Timer 3
   move.w     #Ct3Clr,CtrCmd.l       ;Clear Timer 3 output
```

Figure 5-13 — RTC and Refresh Timer Set Up Routines

5.40 After an interrupt from the levels associated with these two timers:

(1) RTC — Level 6,

(2) Refresh Memory — Level 7.

The output bit should be cleared to remove the interrupt, but the particular timer should not be reloaded and rearmed. This operation is performed automatically by the timer device. To clear the output bit, use the routine illustrated in Figure 5-14.

```
  move.w     #Ct2Clr,CtrCmd.1       ;For the RTC
or
  move.w     #Ct3Clr,CtrCmd.1       ;For the Refresh interrupt
```

Figure 5-14 — RTC and Refresh Timer Clear Routine

5.41 The period between interrupts or resets is calculated by the method and examples illustrated in Figure 5-15.

```
  Period = CtrPrd/4                Where the period is measured
                                     in microseconds

  CtrPrd = Period x 4              Where CtrPrd is an integer in
                                     the range 1 to 65536

                    CtrPrd                     Time

  P1ms        equ        4000      ;1 millisecond
  P2ms        equ        8000      ;2 millisecond
  P10ms       equ       40000      ;10 millisecond
```

Figure 5-15 — Values For Period Between Interrupts Or Resets

```
  Data Rate              CtrK      Fout = Data Rate x 16

  B110        equ        1136      ;Fout = 1761 Hz
  B150        equ         832      ;Fout = 2404 Hz
  B300        equ         416      ;Fout = 4808 Hz
  B600        equ         208      ;Fout = 9615 Hz
  B1200       equ         104      ;Fout = 19231 Hz
  B2400       equ          52      ;Fout = 38462 Hz
  B4800       equ          26      ;Fout = 76923 Hz
  B9600       equ          13      ;Fout = 153846 Hz
```

Figure 5-16 — Values For Divisor Constant

Timer Programming — UART Timer Set Up

5.42 The UART Timer output(s) is a square wave. The frequency is determined by the following formulas:

(1) Fout = 2×10^6/CtrK — where Fout is Hz.

(2) CtrK = 2×10^6/Fout — where CtrK, the division constant, is an integer in the range of 1 to 65,535.

5.43 Figure 5-16 lists the values for divisor constant, CtrK, to generate data rates in general use. To program the UART Timer(s) for the desired frequency or change the frequency, the routines illustrated in Figure 5-17 should be used.

```
SetTimr4                                  ;UART A Txc/Rxc

   move.w     #Ct4Load,TimrCmd.l          ;Address Timer 4
   move.w     #CtrMode,TimrDat.l          ;Set to square wave mode
   move.w     #CtrK,TimrDat.l             ;Set appropriate divisor
   move.w     #Ct4LdArm,TimrCmd.l         ;Load & arm Timer 4

SetTimr5                                  ;UART B Txc/Rxc

   move.w     #Ct5Load,TimrCmd.l          ;Address Timer 5
   move.w     #CtrMode,TimrDat.l          ;Set to square wave mode
   move.w     #CtrK,TimrDat.l             ;Set appropriate divisor
   move.w     #Ct5LdArm,TimrCmd.l         ;Load & arm Timer 5
```

Figure 5-17 — UART Timer Set Up Routines

5.44 Figure 5-18 lists the values of constants used to program the timer device.

```
CtrCmd      equ     $800002     ;Timer Command Resister
CtrDat      equ     $800000     ;Timer Data Resister
CtrMode     equ     $0B22       ;Mode for timing
CtReset     equ     $FFFF       ;Reset Timer device
LoadAll     equ     $FF5F       ;Reset all Timers
Ct16Bus     equ     $FFEF       ;Set Timer to 16 bit mode

CT1Load     equ     $FF01       ;Address Timer 1
Ct1LdArm    equ     $FF61       ;Load & arm Timer 1
Ct1Clr      equ     $FFE1       ;Clear Timer 1 output

Ct2Load     equ     $FF02       ;Address Timer 2
Ct2LdArm    equ     $FF62       ;Load & arm Timer 2
Ct2Clr      equ     $FFE2       ;Clear Timer 2 output

Ct3Load     equ     $FF03       ;Address Timer 3
Ct3LdArm    equ     $FF64       ;Load & arm Timer 3
Ct3Clr      equ     $FFE3       ;Clear Timer 3

Ct4Load     equ     $FF04       ;Address Timer 4
Ct4LdArm    equ     $FF68       ;Load & arm Timer 4
Ct4Clr      equ     $FFE4       ;Clear Timer 4

Ct5Load     equ     $FF05       ;Address Timer 5
Ct5LdArm    equ     $FF70       ;Load & arm Timer 5
Ct5Clr      equ     $FFE5       ;Clear Timer 5
```

Figure 5-18 — Values of Constants Used to Program Timer Device.

6. MAINTENANCE

6.01 The 68000 Central Processing Unit is a result of several years of design, development and modern electronic manufacturing. The system components are designed with the latest semiconductors and integrated circuits. They operate at relatively low power levels with adequate cooling. Each 68000 Central Processing Unit is operated under power and functionally tested in the Codata Systems Corp. factory for a minimum of 72 hours before shipment. The 68000 Central Processing Unit can be expected to operate at peak performance for long intervals.

6.02 No routine maintenance should be performed to the 68000 Central Processing Unit.

Diagnostics

6.03 68000 CPU diagnostic software is under development and not released for production at this manual revision.

Warranty Service

6.04 Codata Systems Corp. Customer Service is available by telephone for assistance in troubleshooting and recommendations for repairs. All communications and material should be directed to:

Codata Systems Corp.
Customer Service Manager
285 North Wolfe Road
Sunnyvale, CA. 94086
(408) 735-1744
TWX 171119

Returning Material For Repair

6.05 The Mainframe Hardware Reference Manual outlines the procedure for returning material.

7. REFERENCE

Logic Diagram and Replaceable Parts List

7.01 Figure 7-1 will furnish the service technician with the logic diagram of the 68000 CPU. Table 7-1 is the replaceable parts list for the 68000 CPU indexed by reference designator appearing on the logic diagram. Enough information is furnished so the maintenance technician should be able to purchase replaceable parts from a local supplier or make a substitution if necessary, 68000 CPU PCAs, ROMs and I/O cables should be ordered directly from Codata Systems Corp. Customer Service.

IEEE 796 Microcomputer Bus

7.02 Tables 7-2 and 7-3 tabulate connectors P1 and P2 pin assignments for the 796 Bus specification.

7.03 The 68000 Central Processing Unit was developed several years prior to adoption of the IEEE 796 Bus Specification. The logic diagram, Figure 7-1, uses references, mnemonics and conventions in use prior to the 796 Bus Specification. Table 7-4 tabulates the pin assignments for the P1 connector and cross references mnemonics to the 796 Bus. The P1 connector is an 86-conductor connector meeting the 796 Bus physical and signal specifications. In some cases a standard 796 Bus signal is not used by the 68000 CPU and is indicated in the comments column.

7.04 Table 7-5 tabulates the pin assignments for the P2 connector. The P2 connector is a 60-conductor connector dedicated to expansion of on-card RAM and is a non-standard use of the 796 Bus.

I/O Ports

7.05 PCA J1 connector provides two serial I/O data channels. The PCA pin assignment is arranged to mate with a 50-conductor serial I/O cable. The cable is split into two 25-conductor groups. Each 25-conductor group is terminated in a DB-25S connector. The DB-25S connector is mounted to the Mainframe rear panel.

7.06 A correlation between J1 pin outs to DB-25 pins has been made to Table 7-6.

7.07 PCA J2 connector provides for the 16-bit input port. Table 7-7 tabulates the pin assignments.

Technical Manual Revisions

7.08 The following summarizes the change history for this technical manual.

(1) Revision A, the initial release, May, 1982.

(2) Revision B, July, 1983. Incorporates design enhancements for and creates 92-1012-02. Changes include PCB wiring and changing 796 Bus Interface device from an 8218 to an 8289.

7.09 Codata Systems Corp. makes changes to drawings and products through engineering change notices (ECN)s. Before a change to a product is approved or made:

(1) The implications to systems in the field are determined,

(2) Rework instructions are included for the equipment in the field when appropriate. Codata Systems Customer Service receives copies of all ECNs.

7.10 There are no pertinent ECNs affecting this 68000 CPU at this manual revision.

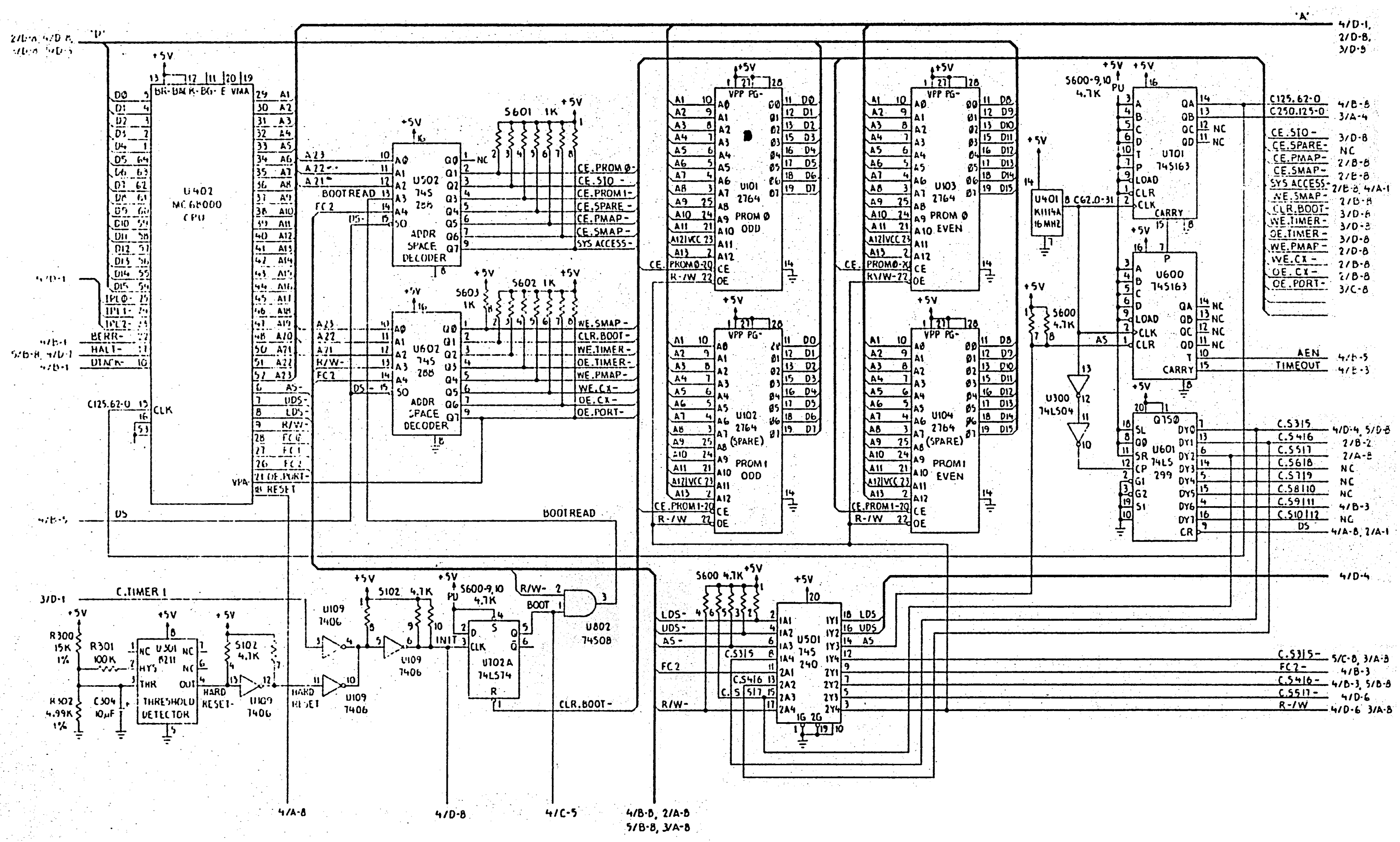


Figure 7-1 — 68000 Central Processing Unit Logic Diagram

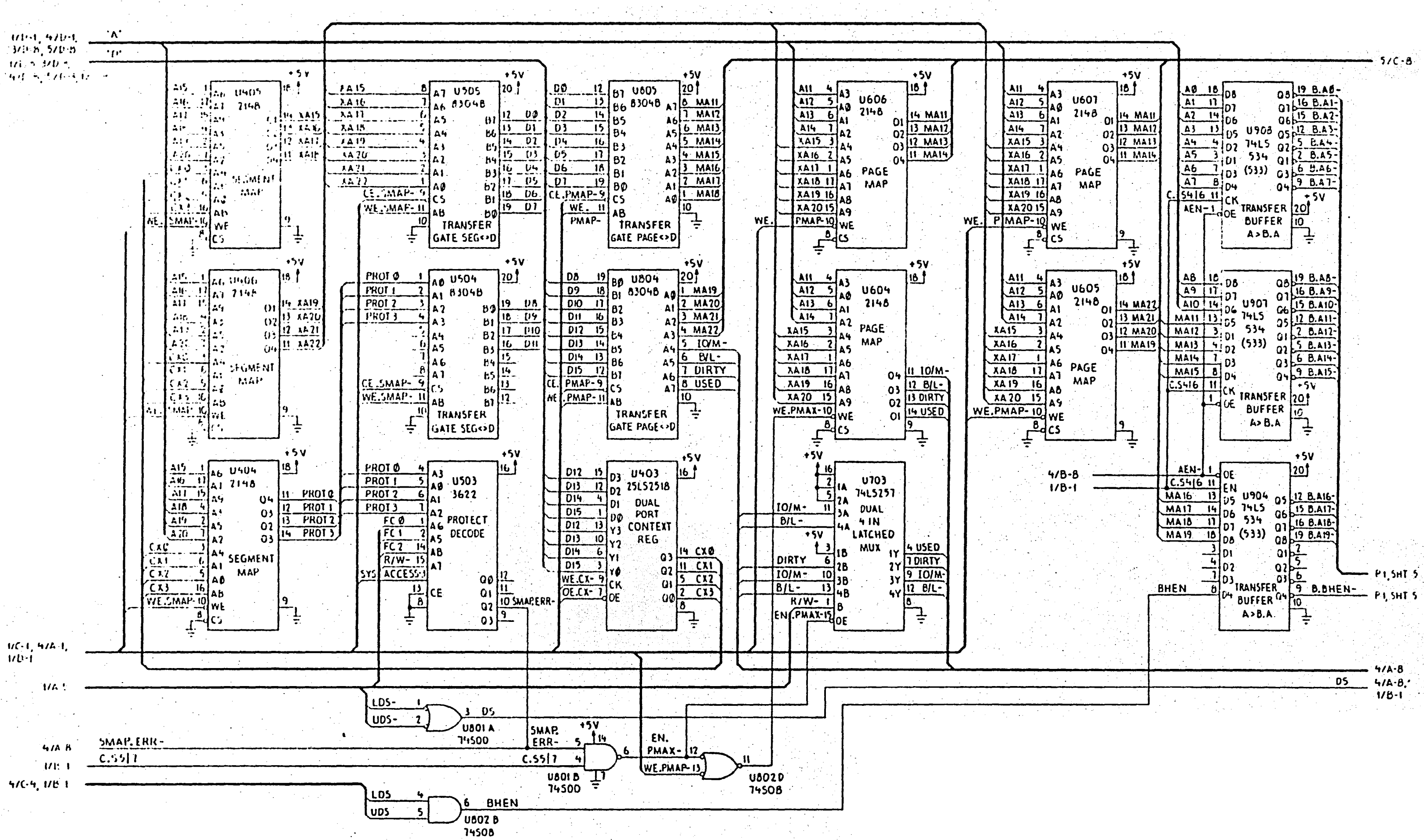


Figure 7-1 — 68000 Central Processing Unit Logic Diagram (Continued)

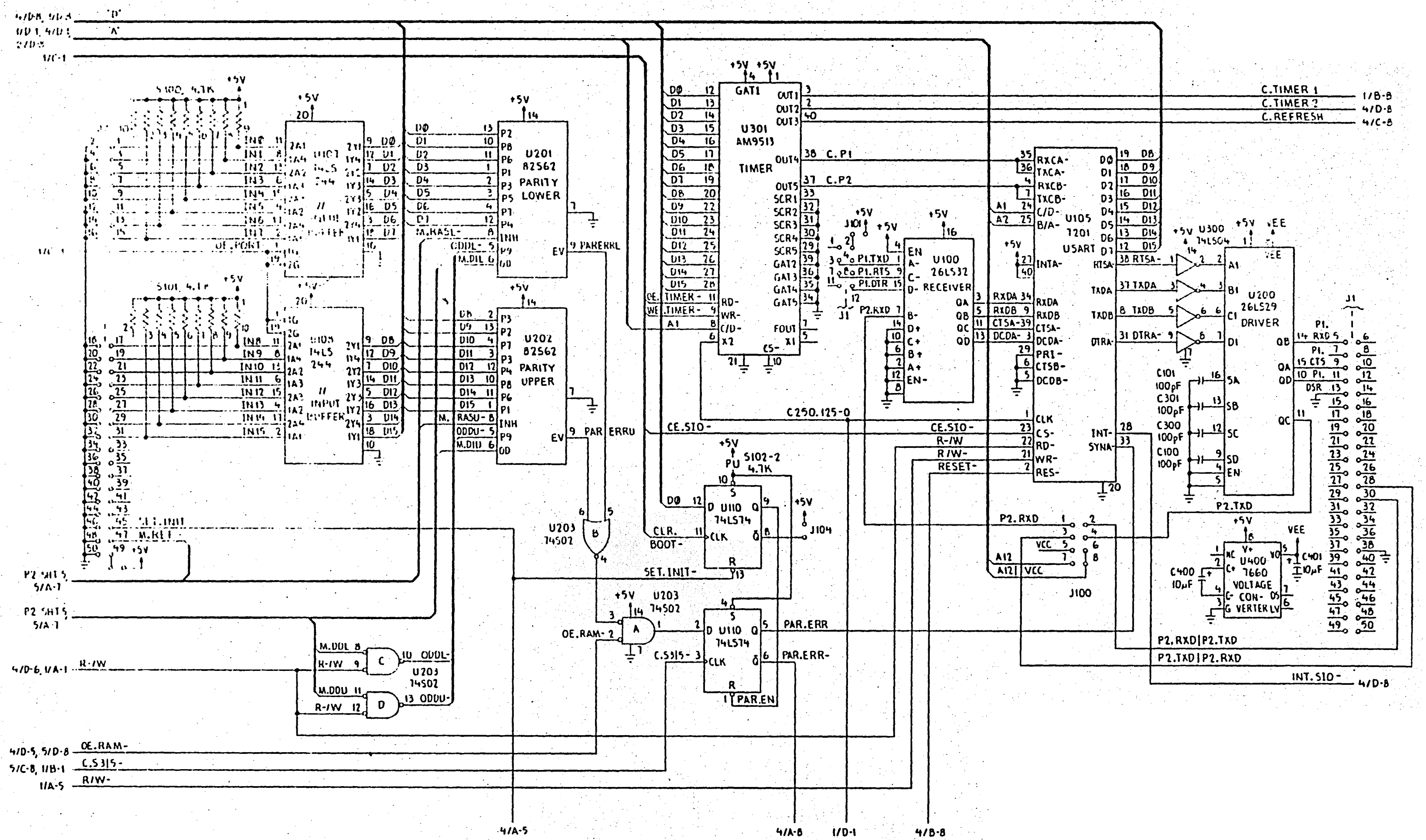


Figure 7-1 — 68000 Central Processing Unit Logic Diagram (Continued)

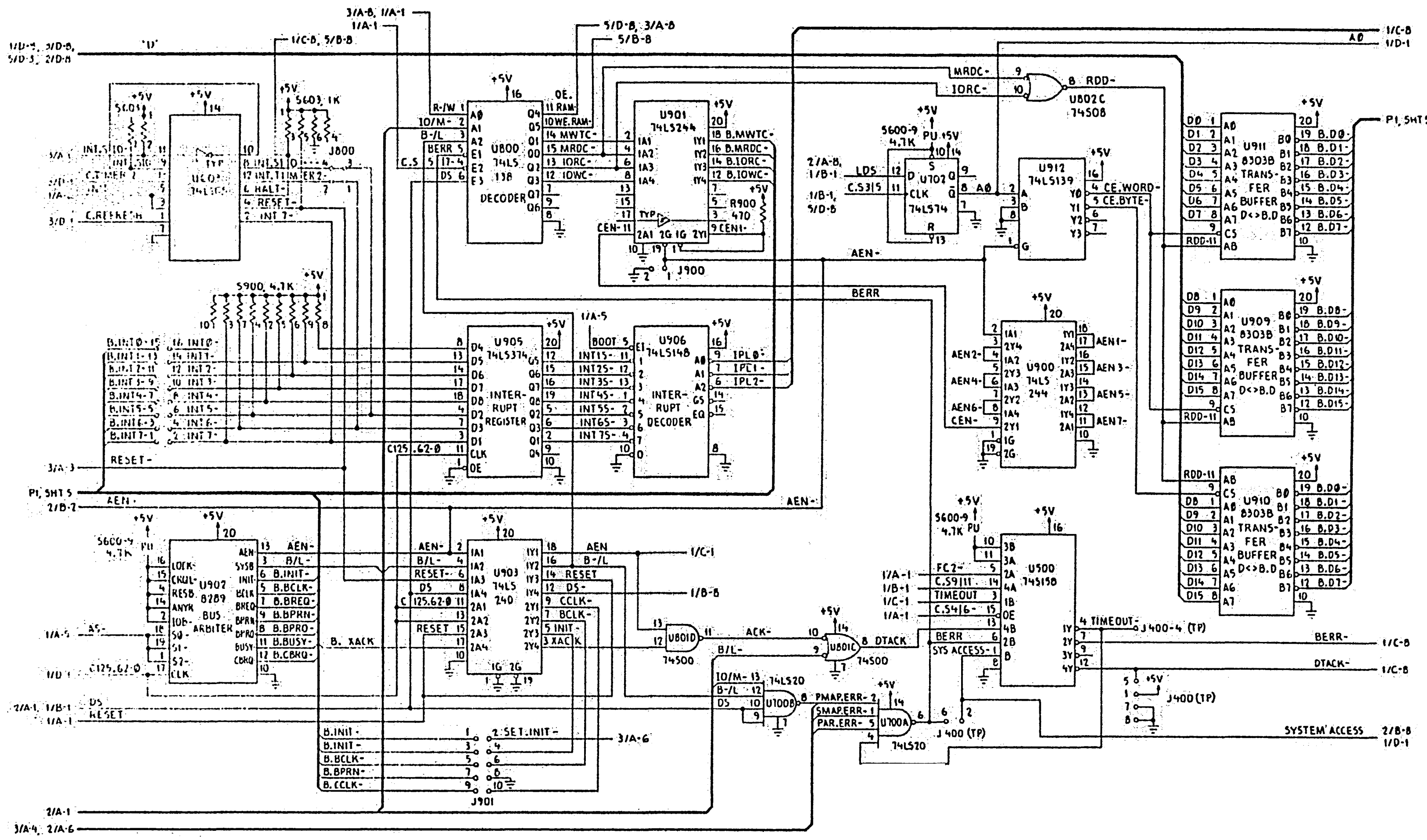


Figure 7-1 — 68000 Cental Processing Unit Logic Diagram (Continued)

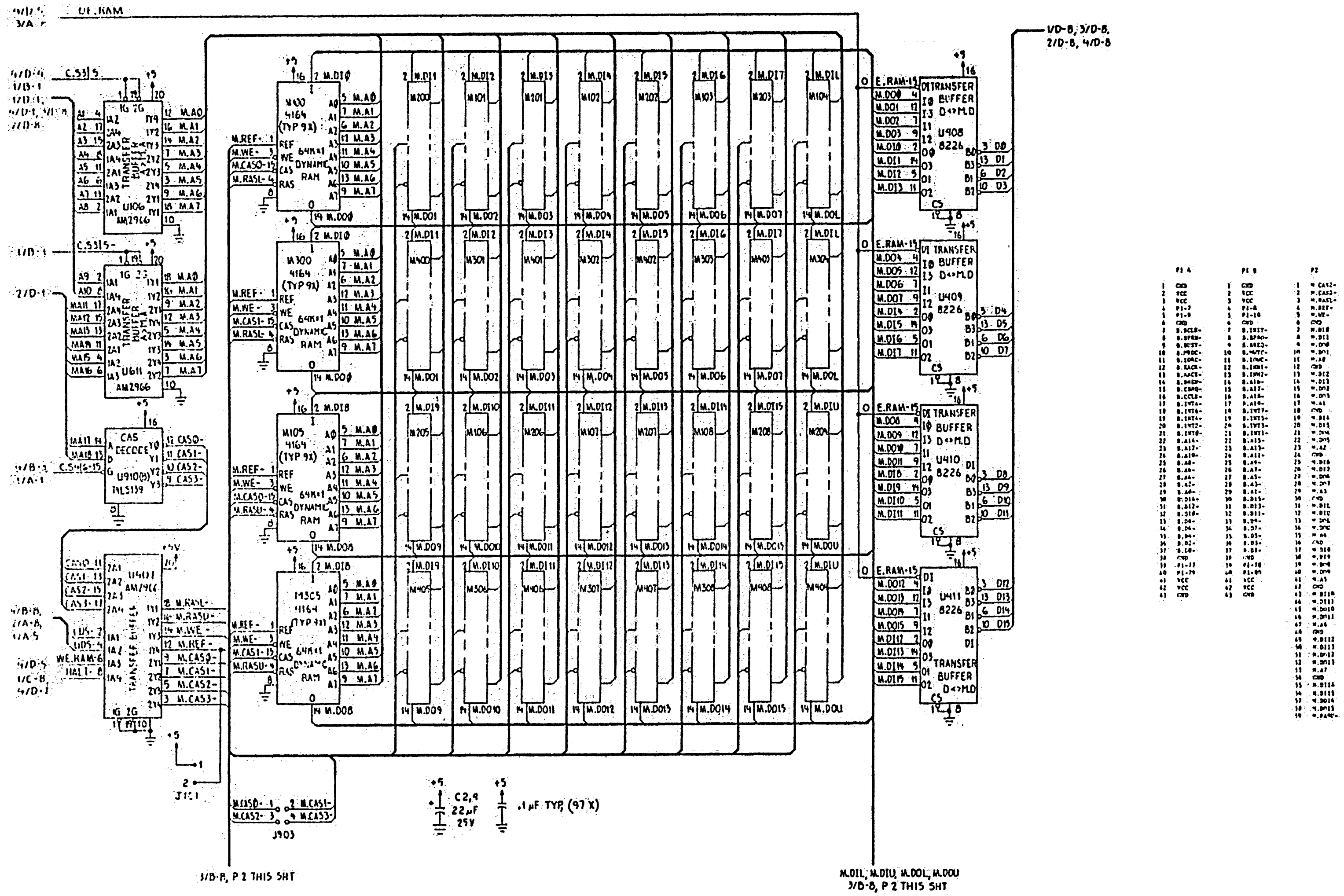


Figure 7-1 — 68000 Central Processing Uint Logic Diagram (Continued)

Table 7-1 — 68000 Central Processing Unit Replaceable Parts List

Reference	Description	Manufacturer	Manufacturer's Part Number	Codata Part Number
	PCA: CPU 68000	Codata Sys	92-1012-02	92-1012-02
C 02	C: Fxd Tant 25V 10% 22uF	Kemet	T110B15620AS	18-0197-01
C 04	C: Fxd Tant 25V 10% 22uF	Kemet	T110B15620AS	18-0197-01
C 100	C: Fxd Mica 50V 10% 100pF	CD		18-0040-01
C 101	C: Fxd Mica 50V 10% 100pF	CD		18-0040-01
C 300	C: Fxd Mica 50V 10% 100pF	CD		18-0040-01
C 301	C: Fxd Mica 50V 10% 100pF	CD		18-0040-01
C 304	C: Fxd Tant 25V 10% 10uF	Sprague	1960106K0025KAI	18-0186-01
C 400	C: Fxd Tant 25V 10% 10uF	Sprague	1960106K0025KAI	18-0186-01
C 401	C: Fxd Tant 25V 10% 10uF	Sprague	1960106K0025KAI	18-0186-01
J 01	Connector: 50-Conductor	3M	3433-1002	21-1026-02
J 02	Connector: 50-Conductor	3M	3433-1002	21-1026-02
K 100	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 200	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 201	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 300	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 400	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 402	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 500	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 501	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 502	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 700	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 800	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 801	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 802	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 900	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 901	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
K 902	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
M 100	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 101	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 102	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 103	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 104	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01

Table 7-1 — 68000 Central Procesing Unit Replaceable Parts List (Continued)

Reference	Description	Manufacturer	Manufacturer's Part Number	Codata Part Number
M 105	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 106	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 107	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 108	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 200	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 201	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 202	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 203	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 204	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 205	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 206	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 207	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 208	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 300	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 301	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 302	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 303	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 304	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 305	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 306	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 307	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 308	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 400	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 401	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 402	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 403	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 404	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 405	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 406	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 407	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
M 408	IC: Random Access Memory 64k x1	Fujitsu	MB8264-20	17-7009-01
R 300	R: Fxd MF 0.25W 1% 15k Ohm	Bourns	RN5501502F	20-3011-01
R 301	R: Fxd CF 0.25W 5% 1M Ohm	Rohm	RC07GF105J	20-0144-01
R 302	R: Fxd MF 0.25W 1% 4.7k Ohm	Bourns	RN5504641F	20-3010-01
S 100	R: SIP MF 0.25W 5% 2.2k Ohm	CTS	750-101-R2.2K	20-1003-01

Table 7-1 — 68000 Central Processing Unit Replaceable Parts List (Continued)

Reference	Description	Manufacturer	Manufacturer's Part Number	Codata Part Number
S 101	R: SIP MF 0.25W 5% 2.2k Ohm	CTS	750-101-R2.2K	20-1003-01
S 102	R: SIP MF 0.25W 5% 2.2k Ohm	CTS	750-101-R2.2K	20-1003-01
S 600	R: SIP MF 0.25W 5% 2.2k Ohm	CTS	750-101-R2.2K	20-1003-01
S 601	R: SIP MF 0.25W 5% 1k Ohm	CTS	750-81-R1K	20-1005-01
S 602	R: SIP MF 0.25W 5% 1k Ohm	CTS	750-81-R1K	20-1005-01
S 603	R: SIP MF 0.25W 5% 1k Ohm	CTS	750-81-R1K	20-1005-01
S 900	R: SIP MF 0.25W 5% 2.2k Ohm	CTS	750-101-R2.2K	20-1003-01
U 100	IC: Quad Line Receiver	AMD	AM26LS32	17-8013-01
U 101	IC: Read Only Memory MON-0	Codata Sys	27-0019-01	27-0019-01
U 103	IC: Read Only Memory MON-E	Codata Sys	27-0020-01	27-0020-01
U 105	IC: Dual UART	NEC	D7201C	17-8011-01
U 106	IC: Octal RAM Driver	AMD	AM2966PC	17-6011-01
U 107	IC: Octal Buffer	TI	SN74LS244N	17-1244-01
U 108	IC: Octal Buffer	TI	SN74LS244N	17-1244-01
U 109	IC: Hex Inverter	TI	SN7406N	17-1006-01
U 110	IC: Dual D-Type Flip Flop	TI	SN74LS74N	17-1074-01
U 200	IC: Quad Line Driver	AMD	AM26LS29	17-8014-01
U 201	IC: 8-Bit Parity Generator	Signetics	82S62	17-6007-01
U 202	IC: 8-Bit Parity Generator	Signetics	82S62	17-6007-01
U 203	IC: Quad 2-In Nor	TI	SN74S02N	17-3002-01
U 300	IC: Hex Inverters	TI	SN74LS04N	17-1004-01
U 301	IC: Programmable Timer	AMD	AM9513DC	17-8015-01
U 302	IC: Voltage Comparator	Intersil	ICL8211CPA	17-6009-01
U 400	IC: Voltage Inverter	Intersil	IC7660CPA	17-6010-01
U 401	IC: Xtal Osc .005% 16 Mhz	Motorola	K114A	17-6012-01
U 402	IC: 16-Bit Microprocessor	Motorola	MC68000L8	17-8010-01
U 403	IC: Quad D-Register	AMD	AM25LS2518	17-6008-01
U 404	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 405	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 406	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 407	IC: Octal RAM Driver	AMD	AM2966PC	17-6011-01
U 408	IC: Bus Driver	Intel	8226	17-8004-01
U 409	IC: Bus Driver	Intel	8226	17-8004-01
U 410	IC: Bus Driver	Intel	8226	17-8004-01
U 411	IC: Bus Driver	Intel	8226	17-8004-01
U 500	IC: Quad 2-To-1-Line Data Sel/M	TI	SN74S158N	17-3158-01

Table 7-1 — 68000 Central Processing Unit Replaceable Parts List (Continued)

Reference	Description	Manufacturer	Manufacturer's Part Number	Codata Part Number
U 501	IC: Octal Buffer	TI	SN74S240N	17-3240-01
U 502	IC: Read Only Memory P0	Codata Sys	27-0022-01	27-0022-01
U 503	IC: Read Only Memory P2	Codata Sys	27-0023-01	27-0023-01
U 504	IC: 8-Bit Noninverting Transcvr	National	DP 8304	17-8017-01
U 505	IC: 8-Bit Noninverting Transcvr	National	DP 8304	17-8017-01
U 600	IC: 4-Bit Counter	TI	SN74LS163N	17-1163-01
U 601	IC: 8-Bit Bidirectional S R	TI	SN74LS299N	17-1299-01
U 602	IC: Read Only Memory P1	Codata Sys	27-0021-01	27-0021-01
U 603	IC: Hex Inverters	TI	SN74LS05N	17-1005-01
U 604	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 605	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 606	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 607	IC: Random Access Memory 4k	Intel	2148-3	17-7008-01
U 611	IC: Octal RAM Driver	AMD	AM2966PC	17-6011-01
U 700	IC: Dual 4-Input Nand	TI	SN74LS20N	17-1020-01
U 701	IC: 4-Bit Counter	TI	SN74S163N	17-3163-01
U 702	IC: Dual D-Type Flip Flop	TI	SN74LS74N	17-1074-01
U 703	IC: Quad Data Select/Mux	TI	SN74LS257N	17-1257-01
U 800	IC: 3-To-8 Decoder	Intel	P3205	17-6006-01
U 801	IC: Quad 2-In Nand	TI	SN74S00N	17-3000-01
U 802	IC: Quad 2-In And	TI	SN74S08N	17-3008-01
U 804	IC: 8-Bit Noninverting Transcvr	National	DP 8304	17-8017-01
U 805	IC: 8-Bit Noninverting Transcvr	National	DP 8304	17-8017-01
U 900	IC: Octal Buffer	TI	SN74LS244N	17-1244-01
U 901	IC: Octal Buffer	TI	SN74LS244N	17-1244-01
U 902	IC: 796 Bus Arbiter	Intel	P8289	17-6021-01
U 903	IC: Octal Buffer	TI	SN74S240N	17-3240-01
U 904	IC: Octal D-Type Flip Flop	TI	SN74LS534N	17-1534-01
U 905	IC: Octal D-Type Flip Flop	TI	SN74LS374N	17-1374-01
U 906	IC: 8-Line-To-3-Line Octal Encd	TI	SN74LS148N	17-1148-01
U 907	IC: Octal D-Type Flip Flop	TI	SN74LS534N	17-1534-01
U 908	IC: Octal D-Type Flip Flop	TI	SN74LS534N	17-1534-01
U 909	IC: Octal Inverting Transceiver	National	DP 8303	17-8016-01
U 910	IC: Octal Inverting Transceiver	National	DP 8303	17-8016-01
U 911	IC: Octal Inverting Transceiver	National	DP 8303	17-8016-01
U 912	IC: Dual 2-To-4-Line Decode/Mux	TI	SN74S139N	17-3139-01

Table 7-1 — 68000 Central Processing Unit Replaceable Parts List (Continued)

Reference	Description	Manufacturer	Manufacturer's Part Number	Codata Part Number
X 100	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 101	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 102	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 103	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 104	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 105	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 106	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 107	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 108	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 200	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 201	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 202	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 203	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 204	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 205	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 206	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 207	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 208	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 300	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 301	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 302	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 303	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 304	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 305	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 306	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 307	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 308	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 400	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 401	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 402	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 403	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 404	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 405	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 406	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 407	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01
X 408	C: Fxd Cer 50V 10% 0.1uF	Centralab	CY20C104M	18-0122-01

Table 7-2 — Pin Assignment of Bus Signals on 796 Bus Board Connector (P1)

	Pin	(Component Side)		Pin	(Circuit Side)	
		Mnemonic	Description		Mnemonic	Description
Power Supplies	1	GND	Signal GND	2	GND	Signal GND
	3	+5V	+5Vdc	4	+5V	+5Vdc
	5	+5V	+5Vdc	6	+5V	+5Vdc
	7	+12V	+12Vdc	8	+12V	+12Vdc
	9		Reserved, bussed	10		Reserved, bussed
	11	GND	Signal GND	12	GND	Signal GND
Bus Controls	13	BCLK*	Bus Clock	14	INIT*	Initialize
	15	BPRN*	Bus Pri. In	16	BPRO*	Bus Pri. Out
	17	BUSY*	Bus Busy	18	BREQ*	Bus Request
	19	MRDC*	Mem Read Cmd	20	MWTC*	Mem Write Cmd
	21	IORC*	I/O Read Cmd	22	IOWC*	I/O Write Cmd
	23	XACK*	XFER Acknowledge	24	INH1*	Inhibit 1 (disable RAM)
Bus Controls and Address	25	LOCK*	Lock	26	INH2*	Inhibit 2 (disable PROM or ROM)
	27	BHEN*	Byte High Enable	28	AD10*	
	29	CBRQ*	Common Bus Request	30	AD11*	Address Bus
	31	CCLK*	Constant Clk	32	AD12*	
	33	INTA*	Intr Acknowledge	34	AD13*	
Interrupts	35	INT6*	Parallel Interrupt Requests	36	INT7*	Parallel Interrupt Requests
	37	INT4*		38	INT5*	
	39	INT2*		40	INT3*	
	41	INT0*		42	INT1*	
Address	43	ADRE*		44	ADRF*	
	45	ADRC*		46	ADRD*	
	47	ADRA*	Address Bus	48	ADRB*	Address Bus
	49	ADR8*		50	ADR9*	
	51	ADR6*		52	ADR7*	
	53	ADR4*		54	ADR5*	
	55	ADR2*		56	ADR3*	
	57	ADR0*		58	ADR1*	
Data	59	DATE*		60	DATF*	
	61	DATC*		62	DATD*	
	63	DATA*	Data Bus	64	DATB*	Data Bus
	65	DAT8*		66	DAT9*	
	67	DAT6*		68	DAT7*	
	69	DAT4*		70	DAT5*	
	71	DAT2*		72	DAT3*	
	73	DAT0*		74	DAT1*	
Power Supplies	75	GND	Signal GND	76	GND	Signal GND
	77		Reserved, bussed	78		Reserved, bussed
	79	-12V	-12Vdc	80	-12V	-12Vdc
	81	+5V	+5Vdc	82	+5V	+5Vdc
	83	+5V	+5Vdc	84	+5V	+5Vdc
	85	GND	Signal GND	86	GND	Signal GND

Notes:

(1) All Reserved pins are reserved for future use and should not be used if upward compatibility is desired.

Table 7-3 — Pin Assignment of Bus Signals on 796 Bus Board Connector (P2)

	Pin	(Component Side) Mnemonic	Description	Pin	(Circuit Side) Mnemonic	Description
	1		Reserved, Not Bussed	2		Reserved, Not Bussed
	3		Reserved, Not Bussed	4		Reserved, Not Bussed
	5		Reserved, Not Bussed	6		Reserved, Not Bussed
	7		Reserved, Not Bussed	8		Reserved, Not Bussed
	9		Reserved, Not Bussed	10		Reserved, Not Bussed
	11		Reserved, Not Bussed	12		Reserved, Not Bussed
	13		Reserved, Not Bussed	14		Reserved, Not Bussed
	15		Reserved, Not Bussed	16		Reserved, Not Bussed
	17		Reserved, Not Bussed	18		Reserved, Not Bussed
	19		Reserved, Not Bussed	20		Reserved, Not Bussed
	21		Reserved, Not Bussed	22		Reserved, Not Bussed
	23		Reserved, Not Bussed	24		Reserved, Not Bussed
	25		Reserved, Not Bussed	26		Reserved, Not Bussed
	27		Reserved, Not Bussed	28		Reserved, Not Bussed
	29		Reserved, Not Bussed	30		Reserved, Not Bussed
	31		Reserved, Not Bussed	32		Reserved, Not Bussed
	33		Reserved, Not Bussed	34		Reserved, Not Bussed
	35		Reserved, Not Bussed	36		Reserved, Not Bussed
	37		Reserved, Not Bussed	38		Reserved, Not Bussed
	39		Reserved, Not Bussed	40		Reserved, Not Bussed
	41		Reserved, Bussed	42		Reserved, Bussed.
	43		Reserved, Bussed	44		Reserved, Bussed.
	45		Reserved, Bussed	46		Reserved, Bussed.
	47		Reserved, Bussed	48		Reserved, Bussed
	49		Reserved, Bussed	50		Reserved, Bussed
	51		Reserved, Bussed	52		Reserved, Bussed
	53		Reserved, Bussed	54		Reserved, Bussed
Address	55	ADR16*	Address Bus	56	ADR17*	Address Bus
	57	ADR14*		58	ADR15*	
	59		Reserved, Bussed	60		Reserved, Bussed

Notes:

(1) All Reserved Pins are reserved for future use and should not be used if upwards compatibility is desired.

(2) Pins 1—40 are for "SPECIAL USE". Special uses are defined in categories. Only category No. 1 is currently described in the IEEE 796 Bus Specification. Category No. 1 is unconstrained use. Other categories are expected to include higher performance busses, I/O interfaces, etc.

(3) Pins 41—60 are intended for future address, data and/or other P1-related signals.

Table 7-4 – 68000 CPU Connector P1 Pin Assignments

Logic Reference	796 Bus Pin	Mnemonic	Signal Name	Comment
PA1	1	GND	Signal Ground	
PB1	2	GND	Signal Ground	
PA2	3	VCC	+5 Vdc	
PB2	4	VCC	+5 Vdc	
PA3	5	VCC	+5 Vdc	
PB3	6	VCC	+5 Vdc	
PA4	7		+12 Vdc	Not used
PB4	8		+12 Vdc	Not used
PA5	9	--	–5 Vdc	Not used
PB5	10		–5 Vdc	Not used
PA6	11	GND	Signal Ground	
PB6	12	GND	Signal Ground	
PA7	13	B.BCLK*	Bus Clock	
PB7	14	B.INIT*	Initialize	
PA8	15	B.BPRN*	Bus Priority In	
PB8	16	B.BPRO*	Bus Priority Out	
PA9	17	B.BUSY*	Bus Ready	
PB9	18	B.BREQ*	Bus Request	
PA10	19	B.MRDC*	Memory Read Command	
PB10	20	B.MWTC*	Memory Write Command	
PA11	21	B.IORC*	I/O Read Command	
PB11	22	B.IOWR*	I/O Write Command	
PA12	23	B.XACK*	XFER Acknowledge	
PB12	24	B.INH1*	Inhibit RAM	Not used
PA13	25	B.AACK*	Adv Acknowledged	Not used
PB13	26	B.INH2*	Inhibit PROM	Not used
PA14	27	B.BHEN*	Byte High Enable	
PB14	28	B.A16*	Address Bit 16	
PA15	29	B.CBRQ*	Common Bus Request	
PB15	30	B.A17*	Address Bit 17	
PA16	31	B.BCCLK*	Constant Clock	
PB16	32	B.A18*	Address Bit 18	
PA17	33	B.INTA*	Intr Acknowledge	Not used
PB17	34	B.A19*	Address Bit 19	
PA18	35	B.INT6*	Interrupt Level 6	
PB18	36	B.INT7*	Interrupt Level 7	
PA19	37	B.INT4*	Interrupt Level 4	
PB19	38	B.INT5*	Interrupt Level 5	
PA20	39	B.INT2*	Interrupt Level 2	
PB20	40	B.INT3*	Interrupt Level 3	
PA21	41	B.INT0*	Interrupt Level 0	Not used
PB21	42	B.INT1*	Interrupt Level 1	
PA22	43	B.A14*	Address Bit 14	
PB22	44	B.A15*	Address Bit 15	

Table 7-4 — 68000 CPU Connector P1 Pin Assignments (Continued)

Logic Reference	796 Bus Pin	Mnemonic	Signal Name	Comment
PA23	45	B.A12*	Address Bit 12	
PB23	46	B.A13*	Address Bit 13	
PA24	47	B.A10*	Address Bit 10	
PB24	48	B.A11*	Address Bit 11	
PA25	49	B.A8*	Address Bit 8	
PB25	50	B.A9*	Address Bit 9	
PA26	51	B.A6*	Address Bit 6	
PB26	52	B.A7*	Address Bit 7	
PA27	53	B.A4*	Address Bit 4	
PB27	54	B.A5*	Address Bit 5	
PA28	55	B.A2*	Address Bit 2	
PB28	56	B.A3*	Address Bit 3	
PA29	57	B.A0*	Address Bit 0	
PB29	58	B.A1*	Address Bit 1	
PA30	59	B.D14*	Data Bit 14	
PB30	60	B.D15*	Data Bit 15	
PA31	61	B.D12*	Data Bit 12	
PB31	62	B.D13*	Data Bit 13	
PA32	63	B.D10*	Data Bit 10	
PB32	64	B.D11*	Data Bit 11	
PA33	65	B.D8*	Data Bit 8	
PB33	66	B.D9*	Data Bit 9	
PA34	67	B.D6*	Data Bit 6	
PB34	68	B.D7*	Data Bit 7	
PA35	69	B.D4*	Data Bit 4	
PA35	70	B.D5*	Data Bit 5	
PA36	71	B.D2*	Data Bit 2	
PB36	72	B.D3*	Data Bit 3	
PA37	73	B.D0*	Data Bit 0	
PB37	74	B.D1*	Data Bit 1	
PA38	75	GND	Signal Ground	
PB38	76	GND	Signal Ground	
PA39	77		Reserved	Not used
PB39	78		Reserved	Not used
PA40	79		−12 Vdc	Not used
PB40	80		−12 Vdc	Not used
PA41	81		+5 Vdc	
PB41	82		+5 Vdc	
PA42	83		+5 Vdc	
PB42	84		+5 Vdc	
PA43	85		Signal Ground	
PB43	86		Signal Ground	

Table 7-5 – 68000 CPU Connector P2 Pin Assignments

Logic Reference	796 Bus Pin	Mnemonic	Signal Name	Comment
PC1	1	M.CAS2*	Memory CAS 2	
PC2	2	M.CAS3*	Memory CAS 3	
PC3	3	M.RASL*	Memory RAS low order	
PC4	4	M.REF*	Memory Refresh	
PC5	5	M.WE*	Memory Write Enable	
PC6	6	GND	Signal Ground	
PC7	7	M.DI0	Memory Data In Bit 0	
PC8	8	M.DI1	Memory Data In Bit 1	
PC9	9	M.DO0	Memory Data Out Bit 0	
PC10	10	M.DO1	Memory Data Out Bit 1	
PC11	11	M.A0	Memory Address Bit 0	
PC12	12	GND	Signal Ground	
PC13	13	M.DI2	Memory Data In Bit 2	
PC14	14	M.DI3	Memory Data In Bit 3	
PC15	15	M.DO2	Memory Data Out Bit 2	
PC16	16	M.DO3	Memory Data Out Bit 3	
PC17	17	M.A1	Memory Address Bit 1	
PC18	18	GND	Signal Ground	
PC19	19	M.DI4	Memory Data In Bit 4	
PC20	20	M.DI5	Memory Data In Bit 5	
PC21	21	M.DO4	Memory Data Out Bit 4	
PC22	22	M.DO5	Memory Data Out Bit 5	
PC23	23	M.A2	Memory Address Bit 2	
PC24	24	GND	Signal Ground	
PC25	25	M.DI6	Memory Data In Bit 6	
PC26	26	M.DI7	Memory Data In Bit 7	
PC27	27	M.DO6	Memory Data Out Bit 6	
PC28	28	M.DO7	Memory Data Out Bit 7	
PC29	29	M.A3	Memory Address Bit 3	
PC30	30	GND	Signal Ground	
PC31	31	M.DIL	Memory Data In Low Order Byte	
PC32	32	M.DIU	Memory Data In High Order Byte	
PC33	33	M.DOL	Memory Data Out Low Order Byte	
PC34	34	M.DOU	Memory Data Out High Order Byte	
PC35	35	M.A4	Memory Address Bit 4	
PC36	36	GND	Signal Ground	
PC37	37	M.DI8	Memory Data In Bit 8	
PC38	38	M.DI9	Memory Data In Bit 9	
PC39	39	M.DO8	Memory Data Out Bit 8	
PC40	40	M.DO9	Memory Data Out Bit 9	
PC41	41	M.A5	Memory Address Bit 5	
PC42	42	GND	Signal Ground	
PC43	43	M.DI10	Memory Data In Bit 10	
PC44	44	M.DI11	Memory Data In Bit 11	

Table 7-5 — 68000 CPU Connector P2 Pin Assignments (Continued)

Logic Reference	796 Bus Pin	Mnemonic	Signal Name	Comment
PC45	45	M.DO10	Memory Data Out Bit 10	
PC46	46	M.DO11	Memory Data Out Bit 11	
PC47	47	M.A6	Memory Address Bit 6	
PC48	48	GND	Signal Ground	
PC49	49	M.DI12	Memory Data In Bit 12	
PC50	50	M.DI13	Memory Data In Bit 13	
PC51	51	M.DO12	Memory Data Out Bit 12	
PC52	52	M.DO13	Memory Data Out Bit 13	
PC53	53	M.A7	Memory Address Bit 7	
PC54	54	GND	Signal Ground	
PC55	55	M.DI14	Memory Data In Bit 14	
PC56	56	M.DI15	Memory Data In Bit 15	
PC57	57	M.DO15	Memory Data Out Bit 15	
PC58	58	M.DO14	Memory Data Out Bit 14	
PC59	59	M.RASU*	Memory RAS Upper Order Byte	
PC60	60	GND	Signal Ground	

Table 7-6 — Pin Assignments of RS-423 Serial IO Board Connector (J1)

PCA Pin	DB-25S Pin	Mnemonic	Description	PCA Pin	DB-25S Pin	Mnemonic	Description
1	1			26	1		
2	14			27	24		
3	2	P1.TXD	Port 1 Transmit	28	2	P2.TXD/RXD	Port 2[1]
4	15			29	15		
5	3	P1.RXD	Port 1 Receive	30		P2.RXD/TXD	Port 2[1]
6	16			31	16		
7	4		Port 1 RTS	32	4		
8	17			33	17		
9	5		Port 1 CTS	34	5		
10	18			35	18		
11	6		Port 1 DSR	36	6		
12	19		Port 1 DTR	37	19		
13	7	GND	Signal Ground	38	7	GND	Signal Ground
14	20			39	20		
15	8			40	8		
16	21			41	21		
17	9			42	9		
18	22			43	22		
19	10			44	10		
20	23			45	23		
21	11			46	11		
22	24			47	24		
23	12			48	12		
24	25			49	25		
25	13			50	13		

Notes:

(1.) Port 2 is configured as DCE or DTE through PCA jumper options. Refer to Table 5-1.

(2.) J1 mates with TB-Ansley 609-5002M.

Table 7-7 — Pin Assignments of 16-Bit Parallel Input Port Connector (J2)

Pin	Mnemonic	Description	Pin	Mnemonic	Description
1	IN0	Input Bit 0	2	GND	Signal Ground
3	IN1	Input Bit 1	4	GND	Signal Ground
5	IN2	Input Bit 2	6	GND	Signal Ground
7	IN3	Input Bit 3	8	GND	Signal Ground
9	IN4	Input Bit 4	10	GND	Signal Ground
11	IN5	Input Bit 5	12	GND	Signal Ground
13	IN6	Input Bit 6	14	GND	Signal Ground
15	IN7	Input Bit 7	16	GND	Signal Ground
17	IN8	Input Bit 8	18	GND	Signal Ground
19	IN9	Input Bit 9	20	GND	Signal Ground
21	IN10	Input Bit 10	22	GND	Signal Ground
23	IN11	Input Bit 11	24	GND	Signal Ground
25	IN12	Input Bit 12	26	GND	Signal Ground
27	IN13	Input Bit 13	28	GND	Signal Ground
29	IN14	Input Bit 14	30	GND	Signal Ground
31	IN15	Input Bit 15	32	GND	Signal Ground
33			34	GND	Signal Ground
35			36	GND	Signal Ground
37			38	GND	Signal Ground
39			40	GND	Signal Ground
41			42	GND	Signal Ground
43			44	GND	Signal Ground
45	SET.INIT*	Reset	46	GND	Signal Ground
47	M.REF*	Halt	48	GND	Signal Ground
49	+5 V	+5 Vdc	50	GND	Signal Ground

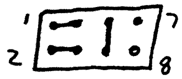
CAUTION

+5 Vdc and ground are physically adjacent pins. If switch closures are used to active IN0 – IN15, be careful NOT to reverse the connector or else the closure will short the system +5 Vdc to ground.

~~92-1012-XX~~

92-1012-02

CPU W/ FLOPPY DRIVE



INSTALL, J 1-3, 2-4, 5-6

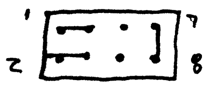
J100 (UPPER LEFT CORNER)

U101 = 27-0019-01 / MON - O

U103 = 27-0020-01 / MON - E

NOTE: when using these P.R.O.M.S., there will be 4 open pins at the top of the socket, because these are 24 pin ~~sockets~~ P.R.O.M.S

CPU W/ NO FLOPPY DRIVE
(PROM BOOT, TAPE DRIVE SYSTEM)



INSTALL, J 1-3, 2-4, 7-8

J100 (UPPER LEFT CORNER)

U 101 = 27-0042-01

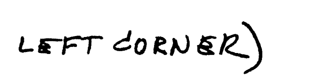
U 102 = 27-0043-01

U 103 = 27-0044-01 / E Ø

U 104 = 27-0045-01 / E 1

FOR EITHER CONFIGURATION

J901



INSTALL, J 3-4, 5-6, 9-10

(LOWER LEFT CORNER)

First to back up Unix Sys
from hard disk the command is

Dump Q B F - 1200 - /DeV/FD1 - /DeV/
RCD00

Next to Config sys once
your in Unix the command is

CD - /USR/SRC/MKconF HiT return
Then type install and answear The
QuesTions.

to go multiuser type Cntrl D

OTher DiRectory's

USR/SRC/ Stand

USR/include/local

CTS/
CTW/
SYS/
~~USR/ lib/ file~~