Please do not upload this copyright pdf document to any other website. Breach of copyright may result in a criminal conviction.

This Acrobat document was generated by me, Colin Hinson, from a document held by me. I requested permission to publish this from Texas Instruments (twice) but received no reply. It is presented here (for free) and this pdf version of the document is my copyright in much the same way as a photograph would be. If you believe the document to be under other copyright, please contact me.

The document should have been downloaded from my website https://blunham.com/Radar, or any mirror site named on that site. If you downloaded it from elsewhere, please let me know (particularly if you were charged for it). You can contact me via my Genuki email page: https://www.genuki.org.uk/big/eng/YKS/various?recipient=colin

You may not copy the file for onward transmission of the data nor attempt to make monetary gain by the use of these files. If you want someone else to have a copy of the file, point them at the website. (https://blunham.com/Radar). **Please do not point them at the file itself as it may move or the site may be updated.**

It should be noted that most of the pages are identifiable as having been processed by me.

I put a lot of time into producing these files which is why you are met with this page when you open the file.

In order to generate this file, I need to scan the pages, split the double pages and remove any edge marks such as punch holes, clean up the pages, set the relevant pages to be all the same size and alignment. I then run Omnipage (OCR) to generate the searchable text and then generate the pdf file.

Hopefully after all that, I end up with a presentable file. If you find missing pages, pages in the wrong order, anything else wrong with the file or simply want to make a comment, please drop me a line (see above).

It is my hope that you find the file of use to you personally – I know that I would have liked to have found some of these files years ago – they would have saved me a lot of time !

Colin Hinson
In the village of Blunham, Bedfordshire.

TMS 9995 MICROPROCESSOR SYSTEM EMULATOR (9995 SE) SPECIFICATION

PRELIMINARY : 6/13/80

REVISION 1 : 2/14/81

REVISION 2: 6/12/81

Prepared by:

John Schabowski,
Microprocessor Systems
Engineer

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT.)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (CONT.)

1.0 SCOPE

This specification establishes the characteristics and performance requirements of the TMS 9995 System Emulator (9995 SE).

1.1 Description

The 9995 SE is a single-chip implementation (N-channel silicon-gate MOS technology) of the CPU (central processing unit) of the TMS 9995 microprocessor. The 9995 SE, when combined with external circuitry that implements the on-chip "peripheral" functions of the TMS 9995 is intended to be the vehicle by which a full-function, real-time, completely controllable in-circuit emulator design can be executed.

1.2 Characteristics

The characteristics of the 9995 SE are identical to those of the TMS 9995 EXCEPT that the 9995 SE has:

- ✓ No on-chip RAM
- ✓ No on-chip Timer/Event Counter (Decrementer)
- ? No on-chip Flags
- ✓ 16-Bit Memory Data Bus (9995 has an 8-bit DATA BUS)
- 18 Prioritized Hardware Interrupts (9995 has 7)
- Extra Control Signals For Emulation

2.0 ARCHITECTURE

2.1 Memory Allocation

Addressing of 16-bit words and 8-bit bytes using the 16-bit data bus and the 16-bit byte address is

The basic word of the 9995 SE architecture is 16 bits in length. ~~These 16 bits are folded into the 8 bit data path to external memory in the manner~~ shown in Figure 1. ~~A word is therefore defined as two consecutive 8-bit bytes in memory.~~ All words (Instruction opcodes, operand addresses, word-length data, etc.) are restricted to be on even address boundaries (i.e. the most significant half or 8 bits resides at an even address and the least significant half resides at the subsequent odd address). Any memory access involving a full word that is directed by software to utilize an odd address to begin the word access with will result in the word starting with this odd address minus one to be accessed.

External logic must therefore be implemented to protect the other byte of a word when an individual byte is written to (The WORD/BYTE- signal indicates when individual bytes are accessed).

The instruction set of the 9995 SE allows both word and byte operations. Byte instructions may address either byte as necessary. ~~A byte access of this type will not effect the other byte of the word involved since the other byte will not be accessed during the execution of the byte instruction.~~

The 9995 SE memory map is shown in Figure 2. Shown are the locations in the memory address space for the Reset, NMI, other interrupt and XOP trap vectors, ~~and the dedicated address segments for the on-chip RAM and the on-chip memory mapped I/O.~~

2.2 [redacted] Organization

The block diagram of the 9995 SE is shown in Figure 3. A flow chart, representative of the 9995 SE functional operation, is shown in Figure 4.

2.2.1 Arithmetic Logic Unit

The arithmetic logic unit (ALU) is the computational component of the 9995 SE. It performs all arithmetic and logic functions required to execute instructions. The functions include addition, subtraction, AND, OR, exclusive OR, and complement. A separate comparison circuit performs the logic and arithmetic comparisons to control bits 0 through 2 of the status register. The ALU is arranged in two 8-bit halves to accommodate byte operations. Each half of the ALU operates on one byte of the operand. During word operand operations, both halves of the ALU function in conjunction with each other. However, during byte operand processing, results from the least significant half of the ALU are ignored. The most significant half of the ALU performs all operations on byte operands so that the status circuitry used in word operations is also used in byte operations.

circuitry used in word operations is also used in byte operations.

2.2.2 Internal Registers

The following three (3) internal registers are accessible to the user (programmer) :

(A) Program Counter (PC)
(B) Status Register (ST)
(C) Workspace Pointer (WP)

Other internal registers which are utilized during instruction acquisition or execution are inaccessible to the user.

2.2.2.1 Program Counter

The Program Counter (PC) is a 15-bit counter that contains the word address of the next instruction following the instruction currently executing. The microprocessor references this address to fetch the next instruction from memory and increments the address in the PC when the new instruction is executing. If the current instruction in the microprocessor alters the contents of PC, then a program branch occurs to the location specified by the altered contents of PC. All context switching operations plus simple branch and jump instructions affect the contents of PC.

2.2.2.2 Status Register

The status register (ST) is a fully implemented 16-bit register that reports the results of program comparisons, indicates program status conditions, and supplies the arithmetic overflow enable and interrupt mask level to the interrupt priority circuits. Each bit position in the register signifies a particular function or condition that exists in the microprocessor. Figure 5 illustrates the bit position assignments. Some instructions use the status register to check for a prerequisite condition; others affect the values of the bits in the register; and others load the entire status register with a new set of parameters. Interrupts also modify the status register. The description of the instruction set later in this document details the effect of each instruction on the status register (see section 3.5). Table 1 lists each bit and identifies what conditions affect that bit.

2.2.2.3 Workspace

The 9995 SE uses blocks of memory words, called workspaces, for instruction operand manipulation instead of internal hardware registers. A workspace occupies 16 contiguous words in any part of memory that is not reserved for other use. The individual workspace registers may contain data or addresses, and function as operand registers, accumulators, address registers, or index registers. Some workspace registers take on special significance during execution of certain instructions. Table 2 lists each of these dedicated workspace registers and the instructions that use them. Figure 6 defines the workspace registers that are allowed

to be used as index registers. A larger number of workspaces may exist in memory simultaneously to provide a high degree of software flexibility.

2.2.2.3.1 Workspace Pointer

To locate the workspace in memory, a hardware register called the workspace pointer (WP) is used. The workspace pointer is a 16 bit register that contains the memory address of the first word in the workspace. The address is left justified with the 16th bit (LSB) hardwired to logic zero. The 9995 SE accesses each register in the workspace by adding two times the register number to the contents of the workspace pointer and initiating a memory request for that word. Figure 7 illustrates the relationship between the workspace pointer and its corresponding workspace in memory.

For instructions performing byte operations, use of the workspace register addressing mode (see section 3.2) will result in the most significant byte of the workspace register involved to be used as the operand for the operation. Since the workspace is also addressable as a memory address, however, the least significant byte may be directly addressed, if desired, using any of the more general memory addressing modes.

2.2.2.3.2 Context Switching

The workspace concept is particularly valuable during operations that require a context switch, which is a change from one program environment to another, as in the case of a subroutine or an interrupt service routine. Such an operation using a conventional multi-register arrangement requires that at least part of the contents of the register file be stored and reloaded using a memory cycle to store or fetch each word. The 9995 SE accomplishes this operation by changing the workspace pointer. A context switch requires only three store cycles and [illegible] fetch +w cycles, exchanging the program counter, status register and workspace pointer. After the switch, the workpace pointer contains the starting address of a new 16-word workspace in memory for use in the new routine. A corresponding time saving occurs when the original context is restored. Instructions in the 9995 SE that result in a context switch include:- Branch and Load Workspace Pointer (BLWP), Return from Subroutine (RTWP) and the Extended Operation (XOP) instruction. All interrupts also cause a context switch by forcing the 9995 SE to trap to a service subroutine.

2.3 [illegible] INTERFACES

Each 9995 SE system interface uses one or more of the signals from one or more of the signal groupings given in the pin description list in Table 3. Each interface is described in detail in the following paragraphs.

2.3.1 Memory Interface

The signals used in the 9995 SE basic interface to system memory are shown in Figure 8.

2.3.1.1 Memory Read Operations

Timing relationships of the memory read sequence are shown in Figure 9. Completion of a memory read cycle and/or generation of Wait states is determined by the READY input as detailed in Section 2.3.4. Note that MEMEN- remains active (low) between consecutive memory operations. Also during memory read cycles that an instruction opcode is being read, IAQ will be asserted as shown in Figure 9.

If an instruction directs that a byte read is to be performed, WORD/BYTE- will be low and only the byte specifically addressed will be read on the half of the data bus that corresponds to the byte address.

Certain special signals are applicable to memory read operations as detailed in Section 2.3.7.

2.3.1.2 Memory Write Operations

Timing relationships of the memory write sequence are shown in Figure 10. Completion of a memory write cycle and/or generation of Wait states is determined by the READY input as detailed in Section 2.3.4. Note that MEMEN- remains active (low) between consecutive memory operations.

If an instruction directs that a byte write is to be performed, WORD/BYTE- will be low and write data will be valid only on the half of the data bus that corresponds to the byte specifically addressed. External logic must use WORD/BYTE- to maintain the integrity of the contents of the other byte in the word.

Certain special signals are applicable to memory write operations as detailed in Section 2.3.7.

~~eight-bit bytes) to external memory requires two memory write cycles that will occur back-to-back (Note: a Hold state request will not be granted between them). If an instruction directs that a byte write to external memory is to be performed, only the byte specifically addressed will be written to (one memory write cycle). External words are accessed most significant (even) byte first, followed by the least significant (odd) byte.~~

2.3.1.3 ~~2.3.1.1.3~~ Direct Memory Access

The 9995 SE Hold state allows both external devices and the 9995 SE to share a common external memory. To gain direct memory access (DMA) to the common memory, the external device first requests the 9995 SE to enter a Hold state by asserting (taking low) the HOLD- input. The 9995 SE will then enter a Hold state following completion of the cycle (either memory, CRU, external Instruction, or internal ALU cycles) that it is currently performing. Note however, that a Hold state will not be entered between the first and second ~~byte accesses of a full word accessed in the external memory address space, and a Hold state will not be entered between the first and second~~ clock cycles of a CRU cycle.

and control signal drivers into specific states,

Upon entry of a Hold state, the 9995 SE puts its address, data, ~~DBIN, and WE/CRUCLK drivers in the high impedance mode,~~ and asserts HOLDA-. The external device can then utilize these signal lines to communicate with the common memory. After the external device has completed its memory transactions, it releases HOLD, and the 9995 SE will continue instruction execution at the point where it had been suspended. Timing relationships for this sequence are shown in Figure 11.

In order to allow DMA loading of external memory on power-up, the 9995 SE will not begin instruction execution after a Reset state until HOLD has been removed if HOLD was active (low) at the time RESET was taken from low to high (RESET released).

~~External devices cannot access the internal (on chip) memory address space of the TMS 9995 when it is in the Hold state.~~

~~Since IAQ (Instruction Opcode Acquisition) and HOLDA (Hold Acknowledge) are multiplexed on a single signal, IAQ/HOLDA, this signal must be gated with MEMEN using external logic to separate IAQ and HOLDA (when MEMEN =0, IAQ/HOLDA can indicate IAQ, and when MEMEN =1, IAQ/HOLDA can indicate HOLDA).~~

~~2.3.1.2 Internal Memory Address Space~~

~~Access of the internal (on chip) memory address space is transparent to the TMS 9995 instruction set. That is, operands can be read from and written into locations in the internal memory space simply by using the appropriate addresses via any of the addressing modes in the TMS 9995 instruction set, and~~

~~Wait states cannot be generated during external instruction cycles or internal ALU/other operation cycles, and the READY input is ignored during these cycles.~~

2.3.2 ~~Multiple~~ Interrupts

The 9995 SE implements ~~seven~~ 18 prioritized, vectored interrupts, some of which ~~are dedicated to~~ have predefined functions and the rest of which are user-definable. Table 4 defines the source (internal or exernal), assignment, priority level, trap vector location in memory, and enabling/resulting status register interrupt mask values for each interrupt.

The 9995 SE will grant interrupt requests only between instructions except for (Level 0) Reset, which will be granted whenever it is requested (i.e. in the middle of an instruction). The basic sequence that the 9995 SE performs to service all interrupt requests is as follows (Note that the 9995 SE performs additional functions for certain interrupts, and these functions will be detailed in subsequent sections):

(1) Prioritize all pending requests and grant the request for the highest priority interrupt that is not masked by the current value of the interrupt mask in the status register or the instruction that has just been executed (See Section 3.5 for these instructions).

(2) Make a context switch using the trap vector specified for the interrupt being granted.

(3) Reset ST7 through ST11 in the status register to zero, and change the interrupt mask (ST12 through ST15) as appropriate for the level of the interrupt being granted.

(4) Resume execution with the instruction located at the new address contained in the PC, and using the new WP. All interrupts will be disabled until after this first instruction is executed, unless: (a) [redacted] Reset is requested, in which case it will be granted, or (b) the interrupt being granted is the MID request and the NMI interrupt is requested simultaneously, in which case the NMI request will be granted before the first instruction indicated by the MID trap vector will be executed.

Note that this sequence has several important characteristics. First of all, for those interrupts that are maskable with the interrupt mask in the status register, the mask will get changed to a value that will permit only interrupts of higher priority to interrupt their service routines. Secondly, status bit ST10 (overflow interrupt enable) gets reset to zero by the servicing of any interrupt so that overflow interrupt requests cannot be generated by an unrelated program segment. Thirdly, the disabling of other interrupts until after the first instruction

of the service routine is executed permits the routine to disable other interrupts by changing the interrupt mask with the first instruction (the exception with MID and NMI is explained in Section 2.3.2.2X). Lastly, the vectoring and prioritizing scheme of the 9995 SE permits interrupts to be automatically nested in most cases. If a higher priority interrupt occurs while in an interrupt service routine, a second context switch occurs to service the higher priority interrupt. When that routine is complete, a return instruction (RTWP) restores the saved context to complete processing of the lower priority interrupt. Interrupt routines should therefore terminate with the return instruction to restore original program parameters.

Additional details of the 9995 SE interrupts are supplied in the following paragraphs.

2.3.2.1 External Interrupt Requests

Each of these interrupts is requested when the designated signal is supplied to the 9995 SE.

2.3.2.1.1 Interrupt Level 0 (RESET-)

Interrupt Level 0 is dedicated to the RESET- input of the 9995 SE. When active (low), RESET- causes the 9995 SE to stop instruction execution and to inhibit (take to logic level) ~~high~~ ~~MEMEN-, DBEN-, and WE-/CRUCLK-~~ The 9995 SE will remain in this Reset state as long as RESET- is active.

all control signals.

specific

When RESET- is released (low-to-high transistion), the 9995 SE performs a context switch with the Level 0 interrupt trap vector (WP and PC of trap vector are in hex memory word addresses 0000X and 0002X, respectively). Note that the old WP, PC and ST are stored in registers 13, 14, and 15 of the new workspace. The 9995 SE then resets all status register bits, ~~the internal interrupt request latches (see Sections 2.3.2.1.3 and 2.3.2.2.3 for details of these latches), Flag Register bits FLAG0 and FLAG1 (see Section 2.3.3.2.1 for details of the Flag Register), and the MID Flag (See Section 2.3.3.2.2).~~ After this the 9995 SE starts execution with the new PC.

Timing relationships of the RESET- signal are shown in Figure ~~X~~ 12.

Release of the RESET- signal is also the time at which the Automatic First Wait State function of the 9995 SE can be invoked (see Section ~~2.3.1.3~~ 2.3.4).

2.3.2.1.2 Non-Maskable Interrupt (NMI-)

The NMI- signal is the request input for the NMI level interrupt and allows ROM loaders, single-step/breakpoint/maintenance panel functions, or other user-defined functions to be implemented for the 9995 SE. This signal and its associated interrupt level are

named "LOAD" in previous 9900 Family products.

13 NMI- being active (low) according to the timing illustrated in Figure ~~X~~ constitutes a request for the NMI level interrupt. The 9995 SE services this request exactly according to the basic sequence previously described, with the priority level, trap vector location, and enabling/resulting status register interrupt mask values as defined in Table 4. Note that the 9995 SE will always grant a request for the NMI level interrupt immediately after execution of the currently executing instruction is completed since NMI is exempt from the interrupt-disabling-after-execution characteristic of certain instructions and also the current value of the interrupt mask.

2.3.2.1.3 Interrupt Levels 1 Through 15

INTREQ- and IC0 through IC3 are the request inputs for interrupt levels 1 through 15. A request is initiated by taking INTREQ- active (low) and simultaneously supplying the code for the desired interrupt level on IC0 through IC3. The 9995 SE services each of these requests exactly according to the basic sequence previously described, with the priority levels, trap vector locations, and enabling/resulting status register interrupt mask values as defined in Table 4. All interrupt requests should remain active until recognized by the processor in the interrupt service routine. The individual service routines then reset the interrupt requests prior to returning from the routines.

2.3.2.2 Internally Generated Interrupts

Each of these interrupts is requested when the designated condition has occured in the 9995 SE.

2.3.2.2.1 Macro Instruction Detection (MID) Interrupt

The acquisition and attempted execution of an MID interrupt opcode will cause the MID level interrupt to be requested before execution of the next instruction will begin (MID interrupt opcodes are defined in Section 3.5.15). In addition to requesting the MID level interrupt, the MID ~~Flag is set to one~~ signal is pulsed (see Section ~~2.3.3.2.2~~ 2.3.7.5). The 9995 SE services this request exactly according to the basic sequence previously described, with the priority level, trap vector location, and enabling/ resulting status register interrupt mask values as defined in Table 4. Note that the 9995 SE will always grant a request for the MID level interrupt since MID is not affected by the interrupt mask and is higher in priority than any other interrupt except for Level 0, Reset. If the NMI interrupt is requested during an MID interrupt context switch, the MID interrupt context switch will be immediately followed by the NMI interrupt service sequence before the first instruction indicated by the MID interrupt is executed. This is done so that the NMI interrupt can be used for a single-step function with MID opcodes. Servicing the MID interrupt request is viewed as "execution" of an MID interrupt opcode, and NMI will allow the 9995 SE to be halted immediately after encountering an MID opcode.

It should also be noted that the MID interrupt shares its trap vector ~~with Level 2~~ with the Arithmetic Overflow interrupt ~~(See Section 2.3.2.2.2)~~. The interrupt subroutine beginning with the PC of this vector ~~should examine the MID Flag~~ to determine the cause of the interrupt. ~~If the MID Flag is set to one, an MID interrupt has occured, and if the MID Flag is set to zero, an Arithmetic Overflow interrupt has occurred. The portion of this interrupt subroutine that handles MID interrupts should always, before returning from the subroutine, reset the MID Flag to zero.~~

and also the external Level 2 request.

must therefore do some inactivation...

The MID interrupt has basically two applications. The MID opcodes can be considered to be illegal opcodes and the MID interrupt is then used to detect errors of this nature. The second, and primary application of the MID interrupt is to allow the definition of additional instructions for the 9995 SE. MID opcodes are used as the opcodes for these macro instructions and software in the MID interrupt service routine emulates the execution of these instructions. The benefit of this implementation of macros is that the macro instructions can be implemented in microcode in future processors and software will then be directly transportable to these future processors.

It should be noted that the 9995 SE interrupt request processing sequence does create some difficulties for re-entrant usage of

MID interrupt macro instructions. In general, to avoid possible errors, MID interrupt macro instructions should not be used in the NMI and Level 1 interrupt subroutines, and should only be used in the Reset subroutine if Reset is a complete initialization of the system.

2.3.2.2.2 Arithmetic Overflow Interrupt

The occurrence of an arithmetic overflow condition, defined as status register bit 4 (ST4) getting set to one (See Table 1 for those conditions that set ST4 to one) can cause the Level 2 interrupt to be requested. Note that this request will occur at a time such that if the request is granted, it will be granted immediately after the instruction that caused the overflow condition. The 9995 SE services this request exactly according to the basic sequence previously described, with the priority level, trap vector location, and enabling/resulting status register interrupt mask values as defined in Table 4.

In addition to being maskable with the interrupt mask, the Level 2 overflow interrupt request is enabled/disabled by status register bit 10 (ST10), the Arithmetic Overflow Enable Bit (i.e., ST10=1 enables overflow interrupt request; ST10=0 disables overflow interrupt request). If servicing the overflow interrupt request is temporarily overridden by the servicing of a higher priority interrupt, the occurrence of the overflow condition will be retained in the contents of the status register (i.e. ST4=1), which is saved by the higher priority context switch. Returning from the higher priority interrupt subroutine via an RTWP instruction causes the overflow condition to be reloaded into status register bit ST4 and the overflow interrupt to be requested again (upon completion of RTWP instruction). The arithmetic overflow interrupt subroutine must reset ST4 or ST10 to zero in the status word saved in register 15 before the routine is complete to prevent generating another overflow interrupt immediately after the return.

It should also be noted that the Level 2 arithmetic overflow interrupt shares its trap vector with the MID interrupt~~. Section 2.3.2.2.1 describes how the interrupt subroutine beginning with the PC of this vector can determine the cause of the interrupt.~~ and the external Level 2 request, as discussed in the previous section.

~~2.3.2.2.3 Decrementer Interrupt~~

~~The occurrence of an interrupt request by the decrementer (See Section 2.3.1.2.2) will cause the Level 3 internal interrupt request latch to get set. This latch is similar to those for Levels 1 and 4 in that it gets reset by servicing a Reset interrupt or when the context switch for its associated interrupt level occurs (Figure 17).~~

~~The Level 3 internal interrupt request latch being set constitutes a request for a level 3 interrupt, and the TMS 9995~~

~~services this request exactly according to the basic sequence previously described, with the priority level, trap vector location, and enabling/resulting status register interrupt mask values as defined in Table 4.~~

2.3.3 Communication Register Unit Interface

The 9995 SE communication register unit (CRU) is an instruction (software) driven bit-oriented I/O interface. The CRU can directly address, in bit-fields of one to sixteen-bits, up to 32768 input bits and 32768 output bits. The 9995 SE executes three single-bit and two multiple-bit instructions. The single-bit instructions are: TEST BIT (TB), SET BIT TO ONE (SBO), and SET BIT TO ZERO (SBZ); the multiple-bit instructions are: LOAD CRU (LDCR), and STORE CRU (STCR). Details of these instructions are given in Sections 3.5.6 and 3.5.7.

The signals used in the 9995 SE interface to the CRU are shown in Figure ~~28~~ 14.

~~2.3.3.1~~ Timing relationships of the CRU input cycle and the CRU output cycle are shown in Figures 15 and 16, respectively. Completion of each CRU cycle and/or generation of Wait states is determined by the READY input as detailed in Section 2.3.4.

~~30~~

~~To output a data bit to an external (off chip) CRU device, the TMS 9995 first outputs the appropriate address on A0-A14. The TMS 9995 leaves MEMEN high, outputs logic zeroes on D0-D2, outputs the data bit on A15/CRUOUT, and strobes WE /CRUCLK. Completion of each CRU output cycle and/or generation of Wait states is determined by the READY input as detailed in Section 2.3.4.3. Timing relationships of the CRU output cycle are shown in Figure 21.~~

(LDCR or STCR instructions)

For multiple bit transfers, these input and output cycles are repeated (with the bit address incremented each cycle) until transfer of the entire field of data bits specified by the CRU instruction being executed has been accomplished. Hold states can occur between these cycles, but will never be granted during any bit cycle.

2.3.3.1 Single Bit CRU Operations

The 9995 SE performs three single-bit CRU functions: TEST BIT (TB), SET BIT TO ONE (SBO), and SET BIT TO ZERO (SBZ). The SBO instruction performs a CRU output cycle with logic one for the

data bit, and the SBZ instruction performs a CRU output cycle with logic zero for the data bit. A TB instruction transfers the addressed CRU bit from the CRUIN input line to bit 2 of the status register (bit ST2, the EQUAL bit).

The 9995 SE develops a CRU bit address for the single-bit operations from the CRU base address contained in workspace register 12 and the signed displacement count contained in bits 8 through 15 of the instruction. The displacement allows two's complement addressing from base minus 128 bits through base plus 127 bits. The base address from WR12 is added to the signed displacement specified in the instruction and the result is placed onto the address bus. Figure ~~23~~ 17 illustrates the development of a single-bit CRU address.

2.3.3.2 ~~2.3.3.1.2~~ Multiple Bit CRU Operations

The 9995 SE performs two multiple-bit CRU operations: store communications register (STCR) and load communications register (LDCR). Both operations perform a data transfer from the CRU-to-memory or from memory-to-CRU as illustrated in Figure ~~23~~ 18. Although the figure illustrates a full 16-bit transfer operation, any number of bits from 1 through 16 may be involved.

The LDCR instruction fetches a word from memory and right shifts it to serially transfer it to CRU output bits. If the LDCR involves eight or fewer bits, those bits come from the right-justified field within the addressed byte of the memory word. If the LDCR involves nine or more bits, those bits come from the right-justified field within the whole memory word. Register 12, bits 0 through 14, defines the starting bit address. When transferred to the CRU interface, each successive bit receives an address that is sequentially greater than the address for the previous bit. This addressing mechanism results in an order reversal of the bits; that is, bit 15 of the memory word (or bit 7) becomes the lowest addressed bit in the CRU and bit 0 becomes the highest bit in the CRU field.

A STCR Instruction transfers data from the CRU to memory. If the operation involves a byte or less transfer, the transferred data will be stored right-justified in the memory byte with leading bits set to zero. If the operation involves from nine to 16 bits, the transferred data is stored right-justified in the memory word with leading bits set to zero. When the input from the CRU device is complete, the lowest addressed bit from the CRU is in the least significant bit position in the memory word or byte.

~~2.3.3.2 Internal CRU Devices~~

~~Access of internal (on-chip) CRU devices is transparent to the TMS 9995 CRU instructions. That is, data can be input from and output to the bits of the internal CRU devices simply by using~~

2.3.4 Wait State Generation

Wait states can be generated for ~~memory~~ and external instruction cycles memory cycles, ~~and external~~ CRU cycles for the 9995 SE using the READY input. A Wait state is defined as extention of the present cycle by one CLKOUT cycle. The timing relationship of the READY input to the memory interface and the CRU interface signals is shown in Figure 19. ~~X. Note that Wait states cannot be generated for memory cycles that access the internal memory address space or for CRU cycles that access the internal CRU address space, as the READY input will be ignored during these cycles.~~

The Automatic First Wait State Generation feature of the 9995 SE allows a Wait state to be inserted in each external memory cycle, regardless of the READY input, as shown in Figure ~~X~~. The Automatic First Wait State Generation feature can be invoked when RESET is asserted. If READY is active (high) when RESET goes through a low-to-high transition, the first Wait state in each ~~external~~ memory cycle will be automatically generated. If READY is inactive (low) when RESET goes through a low-to-high transition, no Wait state will be inserted automatically in each ~~external~~ memory cycle. There is a one and one-half CLKOUT cycle time minimum setup time requirement on READY before the RESET low-to-high transition. The recommended external circuitry for invoking or inhibiting the Automatic First Wait State Generation feature is shown in Figure ~~X~~. Note that this feature does not apply to ~~either internal memory address space accesses or any~~ CRU cycles or external instruction cycles.

Wait states cannot be generated during ~~internal~~ internal ALU/other operation cycles, and the READY input is ignored during these cycles.

2.3.5 External Instructions

The 9995 SE has five external instructions that allow user-defined external functions to be initiated under program control. These instructions are CKON, CKOF, RSET, IDLE, and LREX. These mnemonics, except for IDLE, relate to functions implemented in the 990 minicomputer and do not restrict use of the instructions to initiate various user-defined functions. Execution of an IDLE instruction causes the 9995 SE to enter the Idle state and remain in this state until a request occurs for an

Note that completion of each external instruction and/or generation of Wait states is determined by the READY input as detailed in Section 2.3.4.

interrupt level that is not masked by the current value of the interrupt mask in the status register. (Note that the Reset and NMI interrupt levels are not masked by any interrupt mask value). When any of these five instructions are executed by the 9995 SE, the 9995 SE will use the CRU interface (See Figure ~~18~~ 14) to perform a cycle that is identical to a single bit CRU output cycle (See Figure 21) except for the following: (1) the address being output will be non-specific, (2) the data bit being output will be non-specific, and (3) a code, specified in Table ~~X~~ 5, will be output on D0-D2 to indicate the external instruction being executed. ~~[illegible] READY input is ignored.~~ When the 9995 SE is in the Idle state, cycles with the Idle code will occur repeatedly until a request for an interrupt level that is not masked by the interrupt mask in the status register occurs.

A Hold state can occur during an Idle state, with entry to and return from the Hold state occurring at the Idle code cycle boundaries (See Section ~~2.3.1.1.3~~ 2.3.1.3 for details of entry to and return from the Hold state).

2.3.6
~~2.3.5~~ ~~[illegible]~~ Internal ALU/Other Operation Cycles

When the 9995 SE is performing an operation internally and is not using the memory, CRU, or external instruction interfaces ~~(internal memory space and internal CRU device accesses are defined as using the memory and CRU interfaces)~~ or is not in the Hold state, the 9995 SE will, for as long (as many CLKOUT cycles) as the operation takes, do the following with its interface signals: (1) output a non-specific address on A0-A14 and A15/CRUOUT, (2) output non-specific data on ~~D0-D7~~ D0-D, (3) output logic level high on MEMEN-, DBIN-, ~~and WE-/CRUCLK~~, (4) output logic level low on ~~IAQ/HOLDA~~, and (5) ignore the READY and CRUIN inputs. The HOLD input is still active, however, as the 9995 SE can enter a Hold state while performing an internal ALU/other operation. Also, all interrupt inputs are still active.

IAQ, HOLDA, and CRUCLK

WE-, and R/W-,

2.3.7 Special Signals

The signals described in the following sections are required for emulation purposes.

2.3.7.1 9995 SE/9990- Signal

The 9995 SE/9990- signal, which is permanently connected to either VCC or ground in a system, tells the 9995 SE device which memory map to use with its internal memory control logic. When 9995 SE/9990- is connected to VCC the 9995 SE will use one cycle or two cycle memory word accesses according to the on-chip/off-chip memory addresses of the 9995, as shown in Figures 21 and 22. When 9995SE/9990- is connected to ground the 9995SE will always use one cycle memory word access as shown in Figures 21 and 22.

It should be noted that whenever a two cycle word access is taking place (WORD/BYTE- =1, 9995 SE/9990- connected to VCC, and address of the word being accessed is in the off-chip memory address space of the TMS 9995) a Hold state request will not be granted between the two consecutive cycles.

Also, whenever 9995SE/9990- is connected to VCC and the address of the word or byte being accessed is in the on-chip memory address space of the TMS 9995 the first Wait state of the access is determined by the READY input regardless of whether the Automatic First Wait State Generation feature has been invoked or inhibited.

9995SE/9990- does not affect any non-memory cycles.

2.3.7.2 DESTWRITE Signal

The DESTWRITE output becomes active (high) during an instruction opcode prefetch that occurs when the results of the currently executing instruction have yet to be written to their designated destination address. Timing relationships of this sequence are shown in Figure 23. DESTWRITE is low at all other times.

2.3.7.3 FREEZE- Signal

The FREEZE- input signal conditions the 9995 SE for an immediately following RESET-. The conditioning consists of ensuring that during the following RESET- the 9995 SE inhibits (keeps at inactive logic level) WE-, DBIN-, MEMEN-, and CRUCLK, and the 9995 SE will not change the contents of its PC, WP, or ST registers. Timing relationships of the FREEZE- signal are shown in Figure 24.

2.3.7.4 IACK Signal

The IACK signal provides a means to acknowledge (reset) interrupt stimuli. IACK is active (high) when the 9995 SE is reading the WP of the vector during an interrupt (any interrupt, including RESET, NMI and MID) context switch. IACK is low at all other times. The timing of IACK is exactly the same as A0-A14 for the memory cycle to which it pertains.

2.3.7.5 MID Signal

The MID signal provides a means of setting an externally implemented MID flag. MID is active (high) when the 9995 SE is reading the WP of the vector during an MID interrupt context switch. MID is low at all other times. The timing of MID is exactly the same as A0-A14 for the memory cycle to which it pertains.

3.0 9995 INSTRUCTION SET

3.1 Definition

Each 9995 SE instruction performs one of the following operations:

- Arithmetic, logical, comparison, or manipulation operations on data
- Loading or storage of internal registers (program counter, workspace pointer, or status)
- Data transfer between memory and external devices via the CRU
- Control functions

3.2 Addressing Modes

The 9995 SE instructions contain a variety of available modes for addressing random memory data (e.g., program parameters and flags), or formatted memory data (character strings, data lists, etc.). These addressing modes are:

- Workspace Register Addressing
- Workspace Register Indirect Addressing
- Workspace Register Indirect Auto Increment Addressing
- Symbolic (Direct) Addressing
- Indexed Addressing
- Immediate Addressing
- Program Counter Relative Addressing
- CRU Relative Addressing

The following figures graphically describe the derivation of effective address for each addressing mode. The applicability of addressing modes to particular instructions is described in Section 3.5 along with the description of the operations performed by each instruction. The symbols following the names of the addressing modes (R, *R, *R+, @LABEL or @TABLE (R) are the general forms used by 9995 SE assemblers to select the addressing modes for register R.

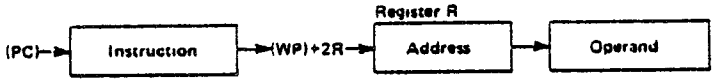
3.2.1 Workspace Register Addressing, R

Workspace Register R contains the operand

The Workspace Register addressing mode is specified by setting the two-bit T-field (T_S or T_D) of the instruction word equal to 00.

3.2.2 Workspace Register Indirect Addressing, *R

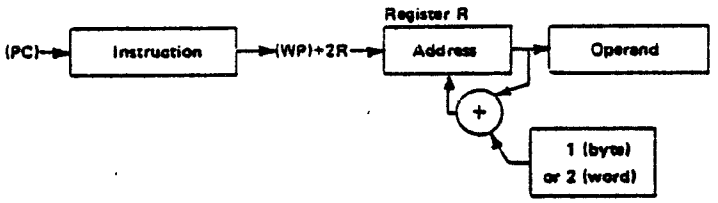
Workspace Register R contains the address of the operand.



The Workspace Register Indirect addressing mode is specified by setting the two-bit T-field (TS or TD) in the instruction word equal to 01.

3.2.3 Workspace Register Indirect Auto Increment Addressing, *R+

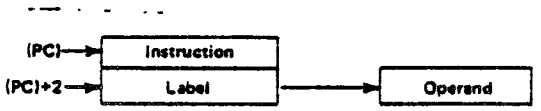
Workspace Register R contains the address of the operand. After acquiring the address of the operand, the contents of Workspace Register R are incremented.



The Workspace Register Indirect Auto Increment addressing mode is specified by setting the two-bit T-field (TS or TD) in the instruction word equal to 11.

3.2.4 Symbolic (Direct) Addressing, @LABEL

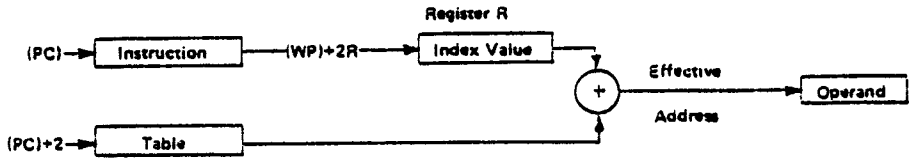
The word following the instruction contains the address of the operand.



The Symbolic addressing mode is specified by setting the two-bit T-field (TS or TD) in the instruction word equal to 10 and setting the corresponding S or D field equal to 0.

3.2.5 Indexed Addressing, @TABLE (R)

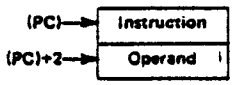
The word following the instruction contains the base address. Workspace Register R contains the index value. The sum of the base address and the index value results in the effective address of the operand.



The indexed addressing mode is specified by setting the two-bit T-field (T_S or T_D) of the instruction word equal to 10 and setting the corresponding S or D field not equal to 0. The value in the S or D field is the register which contains the index value.

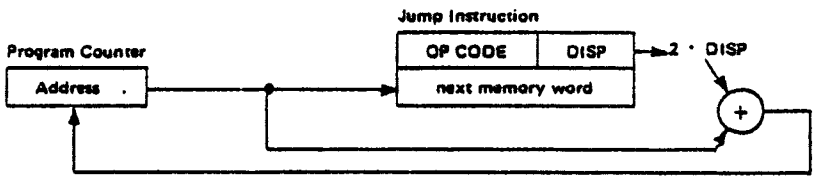
3.2.6 Immediate Addressing

The word following the instruction contains the operand.



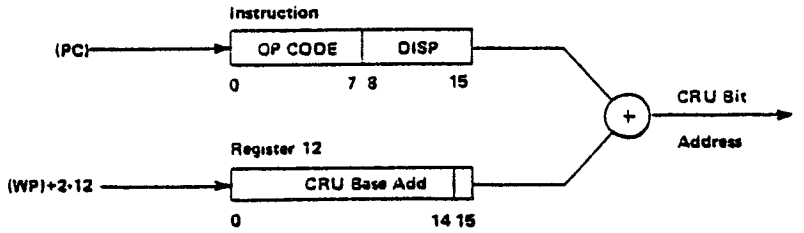
3.2.7 Program Counter Relative Addressing

The eight-bit signed displacement in the right byte (bits 8 through 15) of the instruction is multiplied by 2 and added to the updated contents of the program counter. The result is placed in the PC.



3.2.8 CRU Relative Addressing

The eight-bit signed displacement in the right byte of the instruction is added to the CRU base address (bits 0 through 14 of workspace register 12). The result is the CRU address of the selected CRU bit.



3.3 Definition of Terminology

The terminology used in describing the instructions of the 9995 SE is defined in Table X 6.

3.4 Status Register Manipulation

Various 9995 SE machine instructions affect the status register. Figure 5 shows the status register bit assignments. Table 1 lists the instructions and their effect on the status register.

3.5 Instructions

3.5.1 Dual Operand Instructions with Multiple Addressing for Source and Destination Operand

General Format:

0 1 2 3	4 5	6 7 8 9	10 11	12 13 14 15
OP CODE	B	TD	D	TS	S

If B = 1, the operands are bytes and the operand addresses are byte addresses. If B = 0, the operands are words and the LSB of the operand address is ignored.

The addressing mode for each operand is determined by the T-field of that operand.

T_S or T_D	S or D	ADDRESSING MODE	NOTES
00	0,1...15	Workspace register	1
01	0,1...15	Workspace register indirect	
10	0	Symbolic	4
10	1,2...15	Indexed	2,4
11	0,1...15	Workspace register indirect auto increment	3

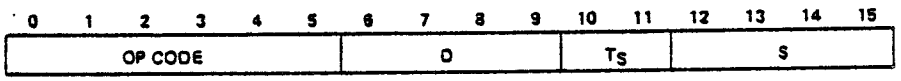
NOTES:

1. When a workspace register is the operand of a byte instruction (bit 3 = 1), the left byte (bits 0 through 7) is the operand and the right byte (bits 8 through 15) is unchanged.
2. Workspace register 0 may not be used for indexing.
3. The workspace register is incremented by 1 for byte instructions (bit 3 = 1) and is incremented by 2 for word instructions (bit 3 = 0).
4. When TS = TD = 10, two words are required in addition to the instruction word. The first word is the source operand base address and the second word is the destination operand base address.

MNEMONIC	OP CODE 0 1 2	B 3	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
A	1 0 1	0	Add	Yes	0-4	(SA)+(DA) → (DA)
AB	1 0 1	1	Add bytes	Yes	0-5	(SA)+(DA) → (DA)
C	1 0 0	0	Compare	No	0-2	Compare (SA) to (DA) and set appropriate status bits
CB	1 0 0	1	Compare bytes	No	0-2,5	Compare (SA) to (DA) and set appropriate status bits
S	0 1 1	0	Subtract	Yes	0-4	(DA) − (SA) → (DA)
SB	0 1 1	1	Subtract bytes	Yes	0-5	(DA) − (SA) → (DA)
SOC	1 1 1	0	Set ones corresponding	Yes	0-2	(DA) OR (SA) → (DA)
SOCB	1 1 1	1	Set ones corresponding bytes	Yes	0-2,5	(DA) OR (SA) → (DA)
SZC	0 1 0	0	Set zeroes corresponding	Yes	0-2	(DA) AND ($\overline{SA}$) → (DA)
SZCB	0 1 0	1	Set zeroes corresponding bytes	Yes	0-2,5	(DA) AND ($\overline{SA}$) → (DA)
MOV	1 1 0	0	Move	Yes	0-2	(SA) → (DA)
MOVB	1 1 0	1	Move bytes	Yes	0-2,5	(SA) → (DA)

3.5.2 Dual Operand Instructions with Multiple Addressing Modes for the Source Operand and Workspace Register Addressing for the Destination

General format:



The addressing mode for the source operand is determined by the TS field.

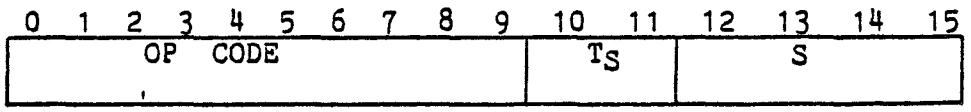
T_S	S	ADDRESSING MODE	NOTES
00	0, 1, . . . 15	Workspace register	
01	0, 1, . . . 15	Workspace register indirect	
10	0	Symbolic	
10	1, 2, . . . 15	Indexed	1
11	0, 1, . . . 15	Workspace register indirect auto increment	2

NOTES: 1. Workspace register 0 may not be used for indexing.
2. The workspace register is incremented by 2.

MNEMONIC	OP CODE 0 1 2 3 4 5	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
COC	0 0 1 0 0 0	Compare ones corresponding	No	2	Test (D) to determine if 1's are in each bit position where 1's are in (SA). If so, set ST2.
CZC	0 0 1 0 0 1	Compare zeros corresponding	No	2	Test (D) to determine if 0's are in each bit position where 1's are in (SA). If so, set ST2.
XOR	0 0 1 0 1 0	Exclusive OR	Yes	0-2	(D) ⊕ (SA) → (D)
MPY	0 0 1 1 1 0	Multiply	No		Multiply unsigned (D) by unsigned (SA) and place unsigned 32-bit product in D (most significant) and D+1 (least significant). If WR15 is D, the next word in memory after WR15 will be used for the least significant half of the product.
DIV	0 0 1 1 1 1	Divide	No		If unsigned (SA) is less than or equal to unsigned (D), perform no operation and set ST4. Otherwise, divide unsigned (D) and (D+1) by unsigned (SA). Quotient → (D), remainder → (D+1). If D = 15, the next word in memory after WR 15 will be used for the remainder.

3.5.3 Signed Multiply and Divide Instructions

General Format:



The addressing mode for the source operand is determined by the TS field.

T_S	S	ADDRESSING MODE	NOTES
00	0,1 ... 15	Workspace register	1
01	0,1 ... 15	Workspace register indirect	1
10	0	Symbolic	1
10	1,2 ... 15	Indexed	1,2
11	0,1 ... 15	Workspace register indirect auto increment	1,3

NOTES: 1. Workspace registers 0 and 1 contain operands used in the signed multiply and divide operations.

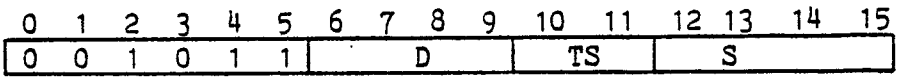
2. Workspace register 0 may not be used for indexing.

3. The workspace register is incremented by 2.

MNEMONIC	OP CODE 0123456789	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
MPYS	0000000111	Signed Multiply	Yes	0-2	Multiply signed two's complement integer in WR0 by signed two's complement integer (SA) and place signed 32-bit product in WR0 (most significant) and WR1 least significant).
DIVS	0000000110	Signed Divide	Yes	0-2,4	If the quotient cannot be expressed as a signed 16 bit quantity (8000(hex) is a valid negative number), set ST4. Otherwise, divide signed, two's complement integer in WR0 and WR1 by the signed two's complement integer (SA) and place the signed quotient in WR0 and the signed remainder in WR1. The sign of the quotient is determined by algebraic rules. The sign of the remainder is the same as the sign of the dividend and |REMAINDER| < |DIVISOR|

3.5.4 Extended Operation (XOP) Instruction

General Format:



The TS and S fields provide multiple mode addressing capability for the source operand. When the XOP is executed, the following transfers occur:

$(40_{16} + 4D) \longrightarrow (WP)$

$(42_{16} + 4D) \longrightarrow (PC)$

$SA \longrightarrow$ (new WR11)

(old WP) $\longrightarrow$ (new WR13)

(old PC) $\longrightarrow$ (new WR14)

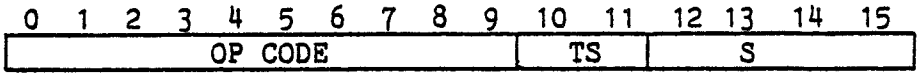
(old ST) $\longrightarrow$ (new WR15)

After these transfers have been made, ST6 is set to one, and ST7, ST8, ST9, ST10 (Overflow Interrupt Enable), and ST11 are all set to zero.

The 9995 SE does not service interrupt trap requests (except for the Reset and NMI Requests) at the end of the XOP instruction.

3.5.5 Single Operand Instructions

General Format:



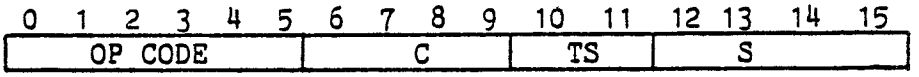
The TS and S fields provide multiple mode addressing capability for the source operand.

MNEMONIC	OP CODE 0123456789	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
B	0000010001	Branch	No	---	SA→(PC)
BL	0000011010	Branch and link	No	---	(PC)→(WR11); SA—(PC)
BLWP	0000010000	Branch and load workspace pointer	No	---	(SA)→(WP); (SA+2)—(PC); (old WP)→(new WR13); (old PC)→(new WR14); (old ST)→(new WR15); The 9995 SE does not service interrupt trap requests (except for the Reset and NMI Requests) at the end of the BLWP instruction.
CLR	0000010011	Clear Operand	No	---	0→(SA)
SETO	0000011100	Set to ones	No	---	$FFFF_{16}$→(SA)
INV	0000010101	Invert	Yes	0-2	$(\overline{SA})$→(SA)
NEG	0000010100	Negate	Yes	0-4	-(SA)→(SA)
ABS	0000011101	Absolute value*	No	0-4	|(SA)|→(SA)
SWPB	0000011011	Swap bytes	No	---	(SA), bits 0 thru 7→(SA) bits 8 thru 15; (SA), bits 8 thru 15→(SA), bits 0 thru 7.
INC	0000010110	Increment	Yes	0-4	(SA)+1→(SA)
INCT	0000010111	Increment by two	Yes	0-4	(SA)+2→(SA)
DEC	0000011000	Decrement	Yes	0-4	(SA)-1→(SA)
DECT	0000011001	Decrement by two	Yes	0-4	(SA)-2→(SA)
X**	0000010010	Execute	No	---	Execute the instruction at SA.

*Operand is compared to zero for status bit.
**If additional memory words for the execute instruction are required to define the operands of the instruction located at SA, these words wiil be accessed from PC and the PC will be updated accordingly. The instruction acquisition signal (IAQ) will not be true when the TMS 9995 accesses the instruction at SA. Status bits are affected in the normal manner for the instruction executed.

3.5.6 CRU Multiple-Bit Instructions

General Format:



The C field specifies the number of bits to be transferred. If C = 0, 16 bits will be transferred. The CRU base register (WR12, bits 0 through 14) defines the starting CRU bit address. The bits are transferred serially and the CRU address is incremented with each bit transfer, although the contents of WR12 is not affected. TS and S provide multiple mode addressing capability for the source operand. If eight or fewer bits are transferred (C = 1 through 8), the source address is a byte address. If nine or more bits are transferred (C = 0, 9 through 15), the source address is a word address. If the source is addressed in the workspace register indirect auto increment mode, the workspace register is incremented by one if C = 1 through 8, and is incremented by two otherwise. If the source is addressed in the register mode, bits 8-15 are unchanged if the transfer is eight bits or less.

MNEMONIC	OP CODE 0 1 2 3 4 5	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
LDCR	0 0 1 1 0 0	Load communication register	Yes	0-2,5*	Beginning with LSB of (SA), transfer the specified number of bits from (SA) to the CRU.
STCR	0 0 1 1 0 1	Store communication register	Yes	0-2,5*	Beginning with LSB of (SA), transfer the specified number of bits from the CRU to (SA). Load unfilled bit positions with 0.

*ST5 is affected only if $1 \leq C \leq 8$.

3.5.7 CRU Single-Bit Instructions

General Format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE								SIGNED DISPLACEMENT							

The signed displacement is added to the contents of WR12 (bits 0-14) to form the address of the CRU bit to be selected.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7	MEANING	STATUS BITS AFFECTED	DESCRIPTION
SBO	0 0 0 1 1 1 0 1	Set bit to one	---	Set the selected output bit to 1.
SBZ	0 0 0 1 1 1 1 0	Set bit to zero	---	Set the selected output bit to 0.
TB	0 0 0 1 1 1 1 1	Test bit	2	If the selected CRU input bit=1, set ST2; if the selected CRU input = 0, set ST2 = 0.

3.5.8 Jump Instructions

General Format:

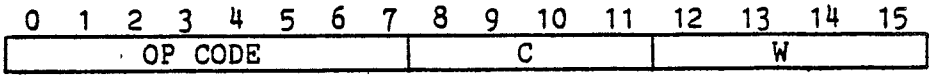
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE								SIGNED DISPLACEMENT							

Jump instructions cause the PC to be loaded with the value selected by PC relative addressing if the bits of ST are at specified values. Otherwise, no operation occurs and the next instruction is executed since the PC points to the next instruction. The signed displacement field is a word count to be added to PC. Thus, the jump instruction has a range of -128 to 127 words from memory-word address following the jump instruction.
No ST bits are affected by jump instructions.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7	MEANING	ST CONDITION TO LOAD PC
JEQ	0 0 0 1 0 0 1 1	Jump equal	ST2 = 1
JGT	0 0 0 1 0 1 0 1	Jump greater than	ST1 = 1
JH	0 0 0 1 1 0 1 1	Jump high	ST0 = 1 and ST2 = 0
JHE	0 0 0 1 0 1 0 0	Jump high or equal	ST0 = 1 or ST2 = 1
JL	0 0 0 1 1 0 1 0	Jump low	ST0 = 0 and ST2 = 0
JLE	0 0 0 1 0 0 1 0	Jump low or equal	ST0 = 0 or ST2 = 1
JLT	0 0 0 1 0 0 0 1	Jump less than	ST1 = 0 and ST2 = 0
JMP	0 0 0 1 0 0 0 0	Jump unconditional	Unconditional
JNC	0 0 0 1 0 1 1 1	Jump no carry	ST3 = 0
JNE	0 0 0 1 0 1 1 0	Jump not equal	ST2 = 0
JNO	0 0 0 1 1 0 0 1	Jump no overflow	ST4 = 0
JOC	0 0 0 1 1 0 0 0	Jump on carry	ST3 = 1
JOP	0 0 0 1 1 1 0 0	Jump odd parity	ST5 = 1

3.5.9 Shift Instructions

General Format:

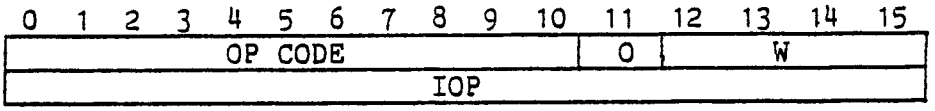


If C = 0, bits 12 through 15 of WR0 contain the shift count. If C = 0 and bits 12 through 15 of WR0 = 0, the shift count is 16.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
SLA	0 0 0 0 1 0 1 0	Shift left arithmetic	Yes	0-4	Shift (W) left. Fill vacated bit positions with 0.
SRA	0 0 0 0 1 0 0 0	Shift right arithmetic	Yes	0-3	Shift (W) right. Fill vacated bit positions with original MSB of (W).
SRC	0 0 0 0 1 0 1 1	Shift right circular	Yes	0-3	Shift (W) right. Shift previous LSB into MSB.
SRL	0 0 0 0 1 0 0 1	Shift right logical	Yes	0-3	Shift (W) right. Fill vacated bit positions with 0's.

3.5.10 Immediate Register Instructions

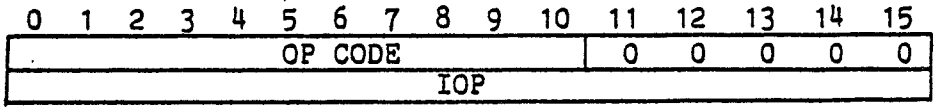
General Format:



MNEMONIC	OP CODE 012345678910	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
AI	00000010001	Add immediate	Yes	0-4	(W)+IOP→(W)
ANDI	00000010010	AND immediate	Yes	0-2	(W) AND IOP→(W)
CI	00000010100	Compare immediate	Yes	0-2	Compare (W) to IOP and set appropriate status bits.
LI	00000010000	Load immediate	Yes	0-2	IOP→(W)
ORI	00000010011	OR immediate	Yes	0-2	(W) OR IOP→(W)

3.5.11 Internal Register Load Immediate Instructions

General Format:



MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MEANING	DESCRIPTION
LWPI	0 0 0 0 0 0 1 0 1 1 1	Load workspace pointer immediate	IOP→(W), no ST bits affected.
LIMI	0 0 0 0 0 0 1 1 0 0 0	Load interrupt mask	IOP, bits 12 thru 15→ ST12 thru ST15.

3.5.10 Internal Register Load and Store Instructions

General Format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE												W			

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10 11	MEANING	STATUS BITS AFFECTED	DESCRIPTION
STST	0 0 0 0 0 0 1 0 1 1 0 0	Store status Register	---	(ST)→(W)
LST	0 0 0 0 0 0 0 0 1 0 0 0	Load status Register	0-15	(W)→(ST)
STWP	0 0 0 0 0 0 1 0 1 0 1 0	Store work-space pointer	---	(WP)→(W)
LWP	0 0 0 0 0 0 0 0 1 0 0 1	Load work-space pointer	---	(W)→(WP)

3.5.13 Return Workspace Pointer (RTWP) Instruction

General Format:

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

The RTWP instruction causes the following transfers to occur:

(WR15)→(ST)

(WR14)→(PC)

(WR13)→(WP)

3.5.14 External Instructions

General Format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE											0	0	0	0	0

External instructions cause three data lines (D0 through D2) to be set to the levels described below, and the ~~WE /CRUCLK~~ CRUCLK line to be pulsed, allowing external control functions to be initiated.

MNEMONIC	OP CODE 012345678910	MEANING	STATUS BITS AFFECTED	DESCRIPTION	DATA BUS D0	D1	D2
IDLE	00000011010	Idle	---	Suspend 9995 SE instruction execution until an unmasked interrupt level request occurs.	L	H	L
RSET	00000011011	Reset	12-15	Set ST12-ST15 to zero	L	H	H
CKOF	00000011110	User defined	---	---	H	H	L
CKON	00000011101	User defined	---	---	H	L	H
LREX	00000011111	User defined	---	---	H	H	H

3.5.15 MID Interrupt Opcodes

The instruction opcodes that will cause an MID interrupt request (see Section ~~2.3.2.1~~ 2.3.2.2) are (hex numbers) :

0000-007F
00A0-017F
0210-021F
0230-023F
0250-025F
0270-027F
0290-029F
02B0-02BF
02D0-02DF
02E1-02FF
0301-033F
0341-035F
0361-037F
0381-039F
03A1-03BF
03C1-03DF
03E1-03FF
0780-07FF
0C00-0FFF

3.6 Instruction Execution

3.6.1 Microinstruction Cycle

Each 9995 SE instruction is executed by a sequence of machine states (microinstructions) with the length of each sequence depending upon the specific instruction being executed. Each microinstruction is completed in one CLKOUT cycle unless Wait states are added to a memory or CRU cycle. (Also, ~~each external memory space access of a word and~~ each external CRU cycle requires at least two CLKOUT cycles but will be accomplished with a single microinstruction)

3.6.2 Execution Sequence

The 9995 SE incorporates an instruction prefetch scheme which ~~minimizes, and in some cases~~ eliminates the time required to fetch the instruction from memory. Without the prefetch, a typical instruction execution sequence is as follows:-

1. Fetch instruction
2. Decode instruction
3. Fetch source operand, if needed
4. Fetch destination operand, if needed
5. Process the operands
6. Store the results, if required

The 9995 SE makes use of the fact that during Step 5 the memory interface is not required; therefore, the fetch of the next instruction can be ~~started, and in some cases even~~ completed in this time. This instruction is then decoded during the state (s) that is (are) required to store the results of the previous instruction, which creates even more execution overlap. Figure 25 illustrates the case of maximum efficiency (instruction opcodes and operands are located in the internal RAM). Note that it effectively takes only four machine states to perform all six steps.

It should be noted that the instruction prefetch scheme employed by the 9995 SE can cause self-modifying software to execute incorrectly. Incorrect execution will result when an instruction is supposed to generate the opcode of the very next instruction to be executed (the 9995 SE will begin the fetch of the opcode of the next instruction before the currently executing instruction will store the results of its execution).

3.6.3 Instruction Execution Times

Instruction execution times for the 9995 SE are a function of:

1) Machine state time, tc2
2) Addressing mode used where operands can be fetched via multiple addressing modes.
3) Number of Wait states introduced, as appropriate.

Table 7 lists the number of machine states and memory accesses required to execute each 9995 SE instruction. For instructions with multiple addressing modes for either or both operands, the main table of table 7 lists machine states and associated memory accesses with all operands addressed in the workspace register mode. To determine the total number of machine states and memory accesses required for other operand addressing modes, the appropriate values from Table A are added to the base amounts for that instruction.

The total execution time for an instruction is:

$$T = tc2(C + W*M)$$

where:

- T = total instruction execution time
- tc2 = CLKOUT cycle time
- C = number of states for instruction execution plus operand address derivation
- W = number of Wait states introduced per memory cycle.
- M = number of memory accesses for instruction execution plus operand address derivation.

As an example, the instruction MOV in a system with tc2 = 0.333 µS, no Wait states, and both operands are addressed in the workspace register mode:

$$T = tc2(C + W*M) = (0.333)(3 + 0*3)\mu S = 1.0\mu S$$

If two wait states per memory access were required:

$$T = (0.333)(3 + 2*3) = 3.0\mu S$$

If the source operand was addressed in the symbolic mode and two wait states were required:

$$C = 3+1 = 4 \; ; \; M = 3+1 = 4$$

$$T = (0.333)(4 + 2*4) = 4.0\mu S$$

4.0 Pin Description

Table 3 defines the 9995 SE pin assignments and describes the function of each pin. Figure 26 illustrates the 9995 SE pin assignment information.

5.0 ELECTRICAL SPECIFICATIONS

5.1 ~~TMS9995~~ Maximum Ratings

The absolute maximum ratings of the 9995 SE are given in Table ~~X~~ 8.

5.2 ~~TMS9995~~ Recommended Operating Conditions

The recommended operating conditions for the 9995 SE are given in Table ~~X~~ 9.

5.3 ~~TMS9995~~ Electrical Characteristics

5.3.1 General

The electrical characteristics of the 9995 SE for the recommended free-air operating temperature range are given in Table ~~X~~ 10.

The switching characteristics of the 9995 SE for the 11 recommended operating conditions are listed in Table ~~X~~, and the switching parameters of this table are defined in Figures 27 through 32. The test load circuit used for loading the 9995 SE outputs for measurement of the switching characteristics is shown in Figure 33. Switching times are measured at the 10% and 90% points of the voltage waveform, as defined in Figure 34.

5.3.2 Clock Characteristics

The 9995 SE has an internal oscillator/clock generator that is controlled by an external crystal. The user may also disable the oscillator and directly inject a frequency source into the ~~XTAL1/CLKIN~~ XTAL2/CLKIN input.

5.3.2.1 Internal Oscillator

The internal oscillator is enabled by connecting a crystal across XTAL2/CLKIN and XTAL1. The frequency of ~~CLKOUT~~ (CLKOUT frequency is the machine state frequency) is one-half the crystal fundamental frequency, fx. The crystal should be a fundamental series-resonant type. Figure 35 details the connection of the external crystal to activate the internal oscillator. 1/4

5.3.2.2 External Oscillator

An external oscillator can be connected to the 9995 SE using the XTAL2/CLKIN pin as detailed in Figure 36. The frequency of CLKOUT (CLKOUT frequency is the machine state frequency) is 11 one-half that of the external oscillator. Table ~~X~~ and Figure 27 define the timing requirements of CLKIN. XTAL2/CLKIN can be driven with standard TTL voltage levels. 1/4

6.0 MECHANICAL SPECIFICATIONS

The physical dimensions of the 9995 ~~SE 40-pin~~ 64-pin dual-in-line package are specified in Figure 37.

TABLE 1. STATUS REGISTER BIT DEFINITIONS
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST0	LOGICAL GREATER THAN	C, CB	If MSB(SA)=1 and MSB(DA)=0, or If MSB(SA)=MSB(DA) and MSB of [(DA)-(SA)]=1.
		CI	If MSB(W)=1 and MSB of IOP=0, or if MSB(W)=MSB of IOP and MSB of [(IOP-(W)]=1.
		ABS, LDCR	If (SA) ≠ 0
		RTWP	If bit (0) of WR15 is 1
		LST	If bit (0) of selected WR is 1
		A, AB, AI, ANDI, DEC, DECT, LI, MOV, MOVB, NEG, ORI, S, SB, DIVS, MPYS, INC INCT, INV, SLA, SOC, SOCB, SRA, SRC, SRL, STCR, SZC, SZCB, XOR	If result ≠ 0
		Reset Interrupt	Unconditionally sets status bit to 0
		All other instructions and interrupts	Do not affect the status bit (see note 2)
ST1	ARITHMETIC GREATER THAN	C, CB	If MSB(SA)=0 and MSB(DA)=1, or If MSB(SA)=MSB(DA) and MSB of [(DA)-(SA)]=1.
		CI	If MSB(W)=0 and MSB of IOP=1, or if MSB(W)=MSB of IOP and MSB of [(IOP-(W)]=1
		ABS, LDCR	If MSB(SA)=0 and (SA) ≠0
		RTWP	If bit (1) of WR15 is 1
		LST	If bit (1) of selected WR is 1

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D.)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST1 (CONTD)	OVERFLOW	A, AB, AI, ANDI, DEC, DECT, LI, MOV, MOVB, NEG, ORI, S, SB, DIVS, MPYS, INC, INCT, INV, SLA, SOC, SOCB, SRA, SRC, SRL, STCR, SZC, SZCB, XOR	If MSB of result =0 and result $\neq$ 0
		Reset Interrupt	Unconditionally sets status bit to 0
		All other instructions and interrupts	Do not affect the status bit (See Note 2)
ST2	EQUAL	C,CB	If (SA) = (DA)
		CI	If (W)=IOP
		COC	If (SA) and ($\overline{DA}$) =0
		CZC	If (SA) and (DA) =0
		TB	If CRUIN=1 for addressed CRU bit
		ABS, LDCR	If (SA)=0
		RTWP	If bit (2) of WR15 is 1
		LST	If bit (2) of selected WR is 1
		A, AB, AI, ANDI, DEC, DECT, LI, MOV, MOVB, NEG, ORI, S, SB, DIVS, MPYS, INC, INCT, INV, SLA, SOC, SOCB, SRA, SRC, SRL, STCR, SZC, SZCB, XOR	If result =0
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D.)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST3	CARRY	A,AB,ABS,AI,DEC, DECT,INC,INCT, NEG,S,SB	If CARRY OUT =1
		SLA,SRA,SRL,SRC	If last bit shifted out =1
		RTWP	If bit (3) of WR15 is 1
		LST	If bit (3) of selected WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)
ST4	OVER-FLOW	A, AB	If MSB(SA)=MSB(DA)and MSB of result ≠ MSB(DA)
		AI	If MSB(W)=MSB of IOP and MSB of result ≠ MSB(W)
		S, SB	If MSB(SA)≠MSB(DA) and MSB of result ≠MSB(DA)
		DEC,DECT	If MSB(SA)=1 and MSB of result =0
		INC,INCT	If MSB(SA)=0 and MSB of result =0
		SLA	If MSB changes during shift
		DIV	If MSB(SA)=0 and MSB(DA)=1, or if MSB(SA)=MSB(DA) and MSB of [(DA)-(SA)] =0
		DIVS	If the quotient cannot be expressed as a signed 16 bit quantity (8000_{16} is a valid negative number)
		ABS,NEG	If (SA)=8000_{16}
		RTWP	If bit (4) of WR15 is 1
		LST	If bit (4) of selected WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D.)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST4 (CONTD)		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)
ST5	ODD PARITY	CB, MOVB	If (SA) has odd number of 1's
		LDCR	If $1 \leq C \leq 8$ and (SA) has odd number of 1's. If C=0 or $9 \leq C \leq 15$, does not affect the status bit.
		STCR	If $1 \leq C \leq 8$ and the stored bits have an odd number of 1's. If C=0 or $9 \leq C \leq 15$, does not affect the status bit.
		AB,SB,SOCB,SZCB	If result has odd number of 1's.
		RTWP	If bit (5) of WR15 is 1
		LST	If bit (5) of seleted WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)
ST6	XOP	XOP	If XOP instruction is executed
		RTWP	If bit (6) of WR15 is 1
		LST	If bit (6) of selected WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST7 ST8 ST9 and ST11	Unused Bits	RTWP	If corresponding bit of WR15 is 1
		LST	If corresponding bit of selected WR is 1.
		XOP, Any Interrupt	Unconditionally sets each of these status bits to 0
		All other Instructions	Do not affect these status bits (See Note 2)
ST10	ARITHMETIC OVERFLOW ENABLE	RTWP	If bit (10) of WR is 1
		LST	If bit (10) of selected WR is 1
		XOP, Any Interrupt	Unconditionally sets status bit to 0
		All other instructions	Do not affect the status bit (See Note 2)
ST12 ST13 ST14 and ST15	INTERRUPT MASK	LIMI	If corresponding bit of IOP is 1
		RTWP	If corresponding bit of WR15 is 1
		LST	If corresponding bit of selected WR is 1.
		RST, Reset and NMI Interrupts	Unconditionally sets each of these status bits to 0
		All other Interrupts	If ST12-ST15=0, no change If ST12=ST15≠0, set to one Less than level of the interrupt trap taken
		All other Instructions	Do not affect these status bits (See Note 2)

NOTES:

1. See Table 7 for Definition of Terminology Used.
2. The X instruction itself does not affect any status bit; the instruction executed by the X instruction sets status bits as defined for that instruction.

TABLE 2. DEDICATED WORKSPACE REGISTERS

REGISTER NO.	CONTENTS	USED DURING
0	Shift count (optional)	Shift instructions (SLA, SRA, SRC, and SLC)
	Multiplicand and MSW of result	Signed Multiply
	MSW of dividend and quotient	Signed Divide
1	LSW of result	Signed Multiply
	LSW of dividend and remainder	Signed Divide
11	Return Address	Branch and Link Instruction (BL)
	Effective Address	Extended Operation (XOP)
12	CRU Base Address	CRU instructions (SBO, SBZ, TB, LDCR, and STCR)
13	Saved WP register	Context switching (BLWP, RTWP, XOP, interrupts)
14	Saved PC register	Context switching (BLWP, RTWP, XOP, interrupts)
15	Saved ST register	Context switching (BLWP, RTWP, XOP, interrupts)

TABLE 3. 9995 SE PIN DESCRIPTION

SIGNAL	PIN	I/O	DESCRIPTION
			POWER SUPPLIES
Vcc	59		Supply voltage (+5V nom)
Vss	19,44		Ground reference
			CLOCKS
XTAL2/CLKIN ~~XTAL1/CLKIN~~	25	IN	Crystal input pin for internal oscillator. Also input pin for external oscillator.
XTAL1 ~~XTAL2~~	24	IN	Crystal input pin for internal oscillator.
CLKOUT1	26	OUT	Clock output signal. The frequency of CLKOUT1 is one half the oscillator input (external oscillator) or crystal (internal oscillator) frequency.
CLKOUT2	27	OUT	Clock output signal. The frequency of CLKOUT2 is one half the oscillator input (external oscillator) or crystal (internal oscillator) frequency. CLKOUT2 is one quarter cycle delayed from CLKOUT1

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			ADDRESS BUS
A0	7	OUT	Address Bus. A0 is the most significant bit of the 16 bit memory address bus and the 15 bit CRU address bus. A14 is the 2nd least significant bit of the 16 bit memory address bus and the LSB of the 15 bit CRU address bus. The address bus assumes the high impedance state when the 9995 SE is in the Hold state.
A1	8	OUT	
A2	9	OUT	
A3	10	OUT	
A4	11	OUT	
A5	12	OUT	
A6	13	OUT	
A7	14	OUT	
A8	15	OUT	
A9	16	OUT	
A10	17	OUT	
A11	18	OUT	
A12	20	OUT	
A13	21	OUT	
A14	22	OUT	
A15/CRUOUT	23	OUT	Address bit 15/CRU output data. A15/CRUOUT is the LSB of the 16 bit memory address bus and the output data line for CRU output instructions. A15/CRUOUT assumes the high impedance state when the 9995 SE is in the Hold state.
			DATA BUS
D0	43	I/O	Data bus. During memory cycles (MEMEN- active) D0 (the MSB) through D15 (the LSB) are used to transfer data to/from the external memory system. During non-memory cycles (MEMEN- inactive) D0, D1 and D2 are used to indicate whether the 9995 SE is performing a CRU cycle or an External instruction. The data bus assumes the high impedance state when the 9995 SE is in the Hold state.
D1	42	I/O	
D2	41	I/O	
D3	40	I/O	
D4	39	I/O	
D5	38	I/O	
D6	37	I/O	
D7	36	I/O	
D8	35	I/O	
D9	34	I/O	
D10	33	I/O	
D11	32	I/O	
D12	31	I/O	
D13	30	I/O	
D14	29	I/O	
D15	28	I/O	

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			CRU
CRUIN	54	IN	CRU Input data. During CRU cycles, CRUIN is the input data line for CRU input data.
CRUCLK	53	OUT	CRU clock. When active (high) CRUCLK indicates that CRU data out is available on A15/CRUOUT (when MEMEN- = 1 and D0 = D1 = D2 = 0) or that an external interface should decode External instructions (when MEMEN- = 1 and D0, D1, and D2 are not all equal to 0).
			CONTROL
MEMEN-	1	OUT	Memory enable. When active (low) MEMEN- indicates that DBIN- and the address and data buses are being used for a memory cycle. When inactive (high) MEMEN- indicates that DBIN- and the address and data buses are being used for a CRU cycle, or are indicating that the 9995 SE is performing an External instruction. MEMEN- does not assume the high impedance state when the 9995 SE is in the Hold state.
DBIN-	62	OUT	Data bus in. During memory read cycles, DBIN- is active (low) to indicate that the 9995 SE has disabled its data bus output buffers to allow external memory to enable 3-state drivers that output data onto the data bus. During CRU input cycles, DBIN- is also active to indicate that the CRU cycle is an input cycle. DBIN- assumes the high impedance state when the 9995 SE is in the Hold state.
WE-	64	OUT	Write enable. When active (low) WE- indicates that memory write data is available on the data bus. WE- assumes the high impedance state when the 9995 SE is in the Hold state.

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			CONTROL (Cont'd)
R/W-	57	OUT	Read/write indication. When high, R/W- indicates that the 9995 SE is performing a memory read cycle (if MEMEN-=0) or a CRU input cycle (if MEMEN-=1). When low, R/W- indicates that the 9995 SE is performing a memory write cycle (if MEMEN-=0) or a CRU output cycle (if MEMEN-=1). R/W- does not assume the high impedance state when the 9995 SE is in the Hold state.
READY	5	IN	Ready. When active (high), READY indicates that the present memory or CRU cycle is ready to be completed. When not ready is indicated, a Wait state (defined as extension of the present cycle by one CLKOUT cycle) is entered. At the end of each Wait state READY is examined to determine if another Wait state is to be generated or if the cycle is to be completed.
HOLD-	63	IN	Hold state request. When active (low), HOLD- indicates to the 9995 SE that an external controller desires to use the address and date buses. Upon sensing a Hold request, the 9995 SE will enter a Hold state (defined as suspension of instruction execution) after it has completed its present cycle. ~~(see Section 2.3.1.1.3 for details of entry into a Hold state).~~ At the beginning of the Hold state, the 9995 SE places DBIN-, WE-, and the address and data buses in the high impedance state, and then responds by asserting HOLDA. When HOLD- is removed, the 9995 SE returns to normal operation.
HOLDA	60	OUT	Hold acknowledge. When active (high), HOLDA indicates that it is in the Hold state.
IAQ	61	OUT	Instruction acquisition. When active (high), IAQ indicates that the memory read cycle in progress is that of reading an

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			CONTROL (Cont'd)
ST8-	58	OUT	Status Bit 8. ST8- is the inverted contents of Status Register bit ST8 (MAP0/MAP1) only during memory cycles (MEMEN-=0). During all other cycles (MEMEN-=1) ST8- is held high.
WORD/BYTE-	56	OUT	Word/Byte. When low, WORD/BYTE- indicates that the memory cycle in progress is that of accessing a single byte on the 16-bit data bus. WORD/BYTE- is held high at all other times.

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			INTERRUPTS
RESET-	4	IN	Reset. When active (low) RESET- causes the 9995 SE to enter a RESET state ~~[illegible]~~ and inhibit its control signals. When RESET- is released, the 9995 SE initiates a level zero interrupt sequence that acquires WP and PC from memory word addresses 0000 and 0002, and begins execution using this vector. RESET- will terminate an Idle state. RESET- is a Schmitt-trigger input.
NMI-	2	IN	Non-maskable Interrupt. When active (low), NMI- causes the 9995 SE to execute a non-maskable interrupt sequence with the trap vector (WP and PC) in memory word addresses FFFC and FFFE. NMI- will terminate an Idle state. NMI- is recognized only once for each high-to-low transition. (NMI- must be taken inactive before it will be recognized again).
INTREQ-	50	IN	Interrupt request. When active (low), INTREQ- indicates that an external interrupt is requested and the code of its level is present on IC0 through IC3.
IC0 (MSB)	49	IN	Interrupt code. IC0 through IC3 are sampled when INTREQ- is active. When IC0 through IC3 are LLLH, the highest priority maskable external interrupt is being requested, and when HHHH, the lowest priority maskable external interrupt is being requested.
IC1	48	IN	
IC2	47	IN	
IC3 (LSB)	46	IN	

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			SPECIAL SIGNALS
9995 SE/9990–	45	IN	9995 SE memory control / 9990 memory control. This signal, which is permanently connected to either Vcc or Ground in a system, tells the 9995 SE device which memory map to use with its internal memory control logic.
DESTWRITE	55	OUT	Destination Write. When high, DESTWRITE indicates that a write cycle to the destination address of the instruction being completed will occur after the current fetch of the next instruction is completed.
FREEZE–	3	IN	Freeze. When active (low) the 9995 SE is conditioned for the immediately following RESET–. The conditioning consists of ensuring that during the following RESET– the 9995 SE inhibits (keeps at inactive logic level) WE–, DBIN–, MEMEN– and CRUCLK, and the 9995 SE will not change the contents of its PC, WP, or ST registers.
IACK	51	OUT	Interrupt Acknowledge. IACK is active (high) when the 9995 SE is reading the WP of the vector during an interrupt (any interrupt, including RESET, NMI, and MID) context switch.
MID	52	OUT	MID Interrupt. MID is active (high) when the 9995 SE is reading the WP of the vector during an MID interrupt context switch.

TABLE 4. INTERRUPT LEVEL DATA

PRIORITY LEVEL	VECTOR LOCATION (Memory Address, In Hex	MASK VALUES TO ENABLE ACCEPTING THE INTERRUPT (ST12 thru ST15)	VALUE MASK SET TO UPON TAKING THE INTERRUPT (ST12 thru ST15)	SOURCE AND ASSIGNMENT
0 (Highest Priority)	0000	0000 thru 1111 (See Note 1)	0000	External: Reset (RESET- Signal)
MID	0008 (See Note 2)	0000 thru 1111 (See Note 1)	0001 (See Note 2)	Internal: MID
NMI	FFFC	0000 thru 1111 (See Note 1)	0000	External: (NMI- Signal)
1	0004	0001 thru 1111	0000	External: IC0-IC3 = 0001
2	0008 (See Note 2)	0010 thru 1111 (See Note 3)	0001 (See Note 2)	Int. and/or Ext.: Arithmetic Overflow, IC0-IC3 = 0010
3	000C	0011 thru 1111	0010	External: IC0-IC3 = 0011
4	0010	0100 thru 1111	0011	External: IC0-IC3 = 0100
5	0014	0101 thru 1111	0100	External: IC0-IC3 = 0101
6	0018	0110 thru 1111	0101	External: IC0-IC3 = 0110
7	001C	0111 thru 1111	0110	External: IC0-IC3 = 0111
8	0020	1000 thru 1111	0111	External: IC0-IC3 = 1000
9	0024	1001 thru 1111	1000	External: IC0-IC3 = 1001
10	0028	1010 thru 1111	1001	External: IC0-IC3 = 1010
11	002C	1011 thru 1111	1010	External: IC0-IC3 = 1011
12	0030	1100 thru 1111	1011	External: IC0-IC3 = 1100
13	0034	1101 thru 1111	1100	External: IC0-IC3 = 1101
14	0038	1110 and 1111	1101	External: IC0-IC3 = 1110
15 (Lowest Priority)	003C	1111	1110	External: IC0-IC3 = 1111

NOTES:-

1. Level 0, MID, and NMI cannot be disabled with the Interrupt Mask.
2. MID and Level 2 use the same trap vector and change the Interrupt Mask to the same value.
3. Generation of a Level 2 request by an Arithmetic Overflow condition (ST4 set to 1) is also enabled/disabled by bit ST10 of the Status Register

TABLE 5. 9995 SE EXTERNAL INSTRUCTION CODES

INSTRUCTION	CODE DURING CYCLE		
	D0	D1	D2
CRU: SBO,SBZ,TB, LDCR or STCR	0	0	0
IDLE	0	1	0
RSET	0	1	1
CKON	1	0	1
CKOF	1	1	0
LREX	1	1	1

TABLE 6. DEFINITION OF TERMINOLOGY

TERM	DEFINITIONS
B	Byte Indicator (1 = byte; 0 = word)
C	Bit Count
D	Destination address register
DA	Destination address
IOP	Immediate operand
LSB(n)	Least significant (right most) bit of (n)
MSB(n)	Most significant (left most) bit of (n)
N	Don't care
PC	Program Counter
Result	Result of operation performed by instruction
S	Source address register
SA	Source address
ST	Status register
STn	Bit n of status register
TD	Destination address modifier
TS	Source address modifier
W	Workspace register
WRn	Workspace register n
(n)	Contents of n
a→b	a is transferred to b
|n|	Absolute value of n

TABLE 6. DEFINITION OF TERMINOLOGY (CONT'D)

TERMS	DEFINITIONS
+	Arithmetic addition
-	Arithmetic subtraction
AND	Logical AND
OR	Logical OR
$\oplus$	Logical exclusive OR
$\bar{n}$	Logical complement of n
.	Arithmetic multiplication

TABLE 7. INSTRUCTION EXECUTION TIMES

INSTRUCTIONS	MACHINE STATES C	MEMORY ACCESS M	OPERAND ADDRESS DERIVATION *	
			SOURCE	DEST
A	4	4	A	A
AB	4	4	A	A
ABS	3	3	A	-
AI	4	4	-	-
ANDI	4	4	-	-
B	3	1	A	-
BL	5	2	A	-
BLWP	11	6	A	-
C	4	3	A	A
CB	4	3	A	A
CI	4	3	-	-
CKON	7	1	-	-
CKOF	7	1	-	-
CLR	3	2	A	-
COC	4	3	A	-
CZC	4	3	A	-
DEC	3	3	A	-
DECT	3	3	A	-
DIV (ST4 is set)	10	4	A	-
DIV (ST4 is reset)**	28	6	A	-
DIVS (ST4 is set)	10	4	A	-
DIVS (ST4 is reset)**	33	6	A	-
IDLE	7+2N	1	-	-
INC	3	3	A	-
INCT	3	3	A	-
INV	3	3	A	-
JUMP (PC is changed)	3	1	-	-
(PC is not changed)	3	1	-	-
LDCR (C = 0)	41	3	A	-
(1 ≤ C ≤ 15)	9+2C	3	A	-
LI	3	3	-	-
LIMI	5	2	-	-
LREX	7	1	-	-
LST	5	2	-	-
LWP	3	2	-	-
LWPI	3	2	-	-
MOV	3	3	A	A
MOVB	3	4	A	A
MPY	23	5	A	-
MPYS	25	5	A	-
NEG	3	3	A	-
ORI	4	4	-	-
RSET	7	1	-	-
RTWP	6	4	-	-

TABLE 7. INSTRUCTION EXECUTION TIMES (CONT'D.)

INSTRUCTIONS	MACHINE STATES C	MEMORY ACCESS M	OPERAND ADDRESS DERIVATION * SOURCE	DEST
S	4	4	A	A
SB	4	4	A	A
SBO	8	2	-	-
SBZ	8	2	-	-
SETO	3	2	-	-
SHIFT (C ≠ 0)	5+C	3	-	-
(C = 0, Bits 12-15 of WR0 = 0)	23	4	-	-
(C = 0, Bits 12-15 of WR0 = N ≠ 0)	7+N	4	-	-
SOC	4	4	A	A
SOCB	4	4	A	A
STCR (C = 0)	43	4	A	-
(1 ≤ C ≤ 8)	19+C	4	A	-
(9 ≤ C ≤ 15)	27+C	4	A	-
STST	3	2	-	-
STWP	3	2	-	-
SWPB	13	3	A	-
SZC	4	4	A	A
SZCB	4	4	A	A
TB	8	2	-	-
X***	2	1	A	-
XOP	15	7	A	-
XOR	4	4	A	-
Interrupt Context Switch (For any Interrupt, including Reset, NMI, MID, and Overflow)	14	6	-	-

*The letter A refers to the table that follows.
**Execution time is dependent upon the partial quotient after each clock cycle during execution.
***Execution time is added to the execution time of the source address.

TABLE 7. INSTRUCTION EXECUTION TIMES (CONT'D)

OPERAND ADDRESS DERIVATION - TABLE A

ADDRESSING MODE	MACHINE STATES C	MEMORY ACCESSES M
WR (T_S or T_D = 00)	0	0
WR indirect (T_S or T_D = 01)	1	1
WR indirect auto-increment (T_S or T_D = 11)	3	2
Symbolic (T_S or T_D = 10, S or D = 0)	1	1
Indexed (T_S or T_D = 10, S or D $\neq$ 0)	3	2

TABLE 8. 9995 SE MAXIMUM RATINGS

ABSOLUTE MAXIMUM RATINGS OVER OPERATING FREE AIR ~~TEMPERATURE~~ TEMPERATURE RANGE (UNLESS OTHERWISE NOTED)*

Supply voltage, Vcc (see Note 1).....7V

Input voltage.....-0.3 to 20V

Off-state output voltage.....-0.3 to 7V

Continuous power dissipation.....1.0W

Operating free-air temperature range.....0° to 70°C

Storage temperature range.....-55° to +150°C

*Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stree rating only and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute maximum rated conditions for extended periods may affect device reliability.

NOTE 1: Voltage values are with respect to network ground terminal, Vss.

TABLE 9. 9995 SE RECOMMENDED OPERATING CONDITIONS

PARAMETER	MIN	NOM	MAX	UNITS
Supply voltage, Vcc	4.5	5.0	5.5	V
Supply voltage, Vss		0.0		V
High-level input voltage, V_{IH} (All inputs)	2.0			V
Low-level input voltage, V_{IL} (All inputs)			0.8	V
High-level output current, I_{OH} (All outputs)			100	uA
Low-level output current, I_{OL} (All outputs)			2.0	mA
Operating free-air temperature, T_A	0		70	°C

TABLE 10. 9995 SE ELECTRICAL CHARACTERISTICS
(OVER RECOMMENDED FREE AIR TEMPERATURE RANGE)

PARAMETER	TEST CONDITIONS*		MIN	TYP**	MAX	UNIT
V_{OH}, High Level Output Voltage	Vcc=MIN, I_{OH}=MAX		2.4	3.4		V
V_{OL}, Low Level Output Voltage	Vcc=MIN, I_{OL}=MAX			0.24	0.4	V
I_{OFF}, Off-state (High Impedance State) Output Current	Vcc=MAX	Vo = 2.4 V			20	uA
		Vo = 0.4 V			-20	
I_I, Input Current	V_I = Vss to Vcc				±50	uA
I_{OS} Short Circuit Output Current***	Vcc = MAX					mA
Icc, Supply Current	Vcc = MAX					
C_I, Input Capacitance (All Inputs Except Data Bus)	f = 1 MHz, All Other Pins at 0 V			15		pf
C_{DB}, Data Bus Capacitance				25		pf
C_O Output Capacitance (Except Data Bus)				10		pf

*For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

**All typical values are at Vcc = 5 V, T_A = ~~20~~ 25°C.

***Not more than one output should be shorted at a time, and duration of the short circuit should not exceed one second.

TABLE 11. 9995 SE SWITCHING CHARACTERISTICS (CONT'D)
(OVER RECOMMENDED OPERATING CONDITIONS RANGE,
UNLESS OTHERWISE NOTED)

SYMBOL	PARAMETER(1)	MIN	NOM(2)	MAX	UNIT
tSU3	CRUIN data set-up time	100			ns
tH3	CRUIN data hold time	0			ns
tSU4	READY set-up time, CRU cycles	200			ns
tSU5	HOLD- set-up time	100			ns
tSU6	RESET-, NMI- set-up time	120			ns
tACC	Memory access time (read cycle)	3/4tC2-130 (3)			ns
tD12	Delay from CLKOUT1 to CLKOUT2	$\frac{1}{2}$tC2 - 10	$\frac{1}{2}$tC2	$\frac{1}{2}$tC2 + 10	ns
tH4	Address, WORD/BYTE-, R/W-, STB- Hold time after WE-	$\frac{1}{4}$tC2 - 20			ns
tH5	Data Hold time	$\frac{1}{4}$tC2 - 40			ns
tWL4	WE- pulse width	$\frac{1}{2}$tC2 - 40 (4)			ns
tH6	Address, WORD/BYTE-, R/W-, STB-, CRUOUT Hold time after CRUCLK	$\frac{1}{4}$tC2 - 20			ns
tD13	Delay from CLKOUT1 to CRUCLK asserted	0		40	ns
tD14	Delay from CLKOUT1 to CRUCLK asserted	0		40	ns
tR4	CRUCLK rise time		20	40	ns
tF3	CRUCLK fall time		20	40	ns
tSU7	INTREQ-, IC0-IC3 set-up time	100			ns
tH7	INTREQ-, IC0-IC3 hold time	0			ns
~~[illegible]~~	~~INTMEM/[illegible] delay time~~	~~[illegible]~~			~~ns~~
~~tH8~~	~~INTMEM/[illegible]~~	~~[illegible]~~			~~[illegible]~~
tD15	Delay from CLKOUT1 to IACK, MID asserted	$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+40	ns
tD16	Delay from CLKOUT1 to IACK, MID asserted	$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+40	ns
tD17	Delay from CLKOUT1 to DESTWRITE asserted	$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+40	ns
tD18	Delay from CLKOUT1 to DESTWRITE released	$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+40	ns

TABLE 11. 9995 SE SWITCHING CHARACTERISTICS

(OVER RECOMMENDED OPERATING CONDITIONS RANGE, UNLESS OTHERWISE NOTED)

SYMBOL	PARAMETER (1)		MIN	NOM(2)	MAX	UNIT
tC1	CLKIN cycle time		160	167	1000	ns
tR1	CLKIN rise time			4	15	ns
tF1	CLKIN fall time			4	15	ns
tWH1	CLKIN high level pulse width		80	$\frac{1}{2}$tC1-tR1	.60 tC1	ns
tWL1	CLKIN low-level pulse width		80	$\frac{1}{2}$tC1-tF1	.60 tC1	ns
tD1	Delay from CLKIN to CLKOUT1			150		ns
tC2	CLKOUT cycle time	External Clock Source		2tC1		ns
		Internal Oscillator (XTAL1 freq. =fx)	$\frac{1.96}{fx}$	$2/f_x$	$\frac{2.04}{fx}$	ns
tR2	CLKOUT rise time			20	35	ns
tF2	CLKOUT fall time			15	20	ns
tWH2	CLKOUT high-level pulse width		.4tC2	$\frac{1}{2}$tC2-tR2	.6 tC2	ns
tWL2	CLKOUT low-level pulse width		.4tC2	$\frac{1}{2}$tC2-tF2	.6 tC2	ns
tD2	Delay from CLKOUT1 to valid address, WORD/BYTE-, R/W-, STG-		$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+95	ns
tD3	Delay from CLKOUT to MEMEN- asserted		$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+40	ns
tD4	Delay from CLKOUT1 to MEMEN- released		$\frac{1}{4}$tC2		$\frac{1}{4}$tC2+50	ns
tSU1	READY set-up time, Memory cycles		100			ns
tH1	READY hold time		0			ns
tDZ	~~Data bus float delay~~ Float Delay				$\frac{1}{4}$tC2+60	ns
tSU2	Input data set-up time		65			ns
tH2	Input data hold time		0			ns
tD5	Delay from CLKOUT1 to DBIN- asserted		0		40	ns
tD6	Delay from CLKOUT1 to DBIN- released		0		50	ns
tD7	Delay from CLKOUT1 to IAQ, HOLDA asserted		0		40	ns
tD8	Delay from CLKOUT1 to IAQ, HOLDA released		0		50	ns
tD9	Delay from CLKOUT1 to valid write data		0		40	ns
tD10	Delay from CLKOUT1 to WE- asserted		0		40	ns
tD11	Delay from CLKOUT1 to WE- released		0		50	ns
tR3	WE- rise time			25	50	ns

TABLE II. 9995 SE SWITCHING CHARACTERISTICS (CONT'D)
(OVER RECOMMENDED OPERATING CONDITIONS RANGE, UNLESS OTHERWISE NOTED)

NOTES:-

1. Additional Test Conditions, All Outputs:
   I_{OL} = MAX, I_{OH} = MAX, C_L = 100pf
   (In figure 34 C_L = 100pf, R1=2,4K, R2=24K)
2. All Nominal Values at TA=25°C, Vcc=NOM
3. Memory access time (read cycle) MIN parameter is determined as follows:- (No Wait States)
   $t_{ACC,MIN} = t_{C2,nom} - t_{D2,max} - t_{SU2,MIN} - t_{F2,MAX}$
   $= 3/4\,t_{C2} - 130$ ns
   Therefore, for t_{C2}=333 ns,
   $t_{ACC,MIN}$=120 ns.
4. WE- pulse width MIN parameter is determined as follows:- (No Wait States)
   $t_{WL4,MIN} = \frac{1}{2}t_{C2,NOM} + t_{D11,MAX} - t_{R3,MAX} - t_{D10,MAX}$
   $= \frac{1}{2}t_{C2} - 40$
   Therefore, for t_{C2} = 333 ns,
   $t_{WL4,MIN}$ = 127 ns

5.0 ELECTRICAL SPECIFICATIONS

5.1 ~~TMS9900~~ Maximum Ratings

The absolute maximum ratings of the 9995 SE are given in Table ~~X~~ 8.

5.2 ~~TMS9900~~ Recommended Operating Conditions

The recommended operating conditions for the 9995 SE are given in Table ~~X~~ 9.

5.3 ~~TMS9900~~ Electrical Characteristics

5.3.1 General

The electrical characteristics of the 9995 SE for the recommended free-air operating temperature range are given in Table ~~X~~ 10.

The switching characteristics of the 9995 SE for the 11 recommended operating conditions are listed in Table ~~X~~, and the switching parameters of this table are defined in Figures 27 through 32. The test load circuit used for loading the 9995 SE outputs for measurement of the switching characteristics is shown in Figure 33. Switching times are measured at the 10% and 90% points of the voltage waveform, as defined in Figure 34.

5.3.2 Clock Characteristics

The 9995 SE has an internal oscillator/clock generator that is controlled by an external crystal. The user may also disable the oscillator and directly inject a frequency source into the ~~XTAL1/CLKIN~~ XTAL2/CLKIN input.

5.3.2.1 Internal Oscillator

The internal oscillator is enabled by connecting a crystal across XTAL2/CLKIN and XTAL1. The frequency of CLKOUT (CLKOUT frequency is the machine state frequency) is one-half the crystal fundamental frequency, fx. The crystal should be a fundamental series-resonant type. Figure 35 details the connection of the external crystal to activate the internal oscillator.

5.3.2.2 External Oscillator

An external oscillator can be connected to the 9995 SE using the XTAL2/CLKIN pin as detailed in Figure 36. The frequency of CLKOUT (CLKOUT frequency is the machine state frequency) is 11 one-half that of the external oscillator. Table ~~X~~ and Figure 27 define the timing requirements of CLKIN. XTAL2/CLKIN can be driven with standard TTL voltage levels.

6.0 MECHANICAL SPECIFICATIONS

The physical dimensions of the 9995 SE ~~40-pin~~ 64-pin dual-in-line package are specified in Figure 37.

TABLE 1. STATUS REGISTER BIT DEFINITIONS
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST0	LOGICAL GREATER THAN	C, CB	If MSB(SA)=1 and MSB(DA)=0, or If MSB(SA)=MSB(DA) and MSB of [(DA)-(SA)]=1.
		CI	If MSB(W)=1 and MSB of IOP=0, or if MSB(W)=MSB of IOP and MSB of [(IOP-(W)]=1.
		ABS, LDCR	If (SA) ≠ 0
		RTWP	If bit (0) of WR15 is 1
		LST	If bit (0) of selected WR is 1
		A, AB, AI, ANDI, DEC, DECT, LI, MOV, MOVB, NEG, ORI, S, SB, DIVS, MPYS, INC INCT, INV, SLA, SOC, SOCB, SRA, SRC, SRL, STCR, SZC, SZCB, XOR	If result ≠ 0
		Reset Interrupt	Unconditionally sets status bit to 0
		All other instructions and interrupts	Do not affect the status bit (see note 2)
ST1	ARITHMETIC GREATER THAN	C, CB	If MSB(SA)=0 and MSB(DA)=1, or If MSB(SA)=MSB(DA) and MSB of [(DA)-(SA)]=1.
		CI	If MSB(W)=0 and MSB of IOP=1, or if MSB(W)=MSB of IOP and MSB of [(IOP-(W)]=1
		ABS, LDCR	If MSB(SA)=0 and (SA) ≠0
		RTWP	If bit (1) of WR15 is 1
		LST	If bit (1) of selected WR is 1

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D.)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED, ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST1 (CONTD)	OVERFLOW	A, AB, AI, ANDI, DEC, DECT, LI, MOV, MOVB, NEG, ORI, S, SB, DIVS, MPYS, INC, INCT, INV, SLA, SOC, SOCB, SRA, SRC, SRL, STCR, SZC, SZCB, XOR	If MSB of result =0 and result $\neq$ 0
		Reset Interrupt	Unconditionally sets status bit to 0
		All other instructions and interrupts	Do not affect the status bit (See Note 2)
ST2	EQUAL	C,CB	If (SA) = (DA)
		CI	If (W)=IOP
		COC	If (SA) and ($\overline{DA}$) =0
		CZC	If (SA) and (DA) =0
		TB	If CRUIN=1 for addressed CRU bit
		ABS, LDCR	If (SA)=0
		RTWP	If bit (2) of WR15 is 1
		LST	If bit (2) of selected WR is 1
		A, AB, AI, ANDI, DEC, DECT, LI, MOV, MOVB, NEG, ORI, S, SB, DIVS, MPYS, INC, INCT, INV, SLA, SOC, SOCB, SRA, SRC, SRL, STCR, SZC, SZCB, XOR	If result =0
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and	Do not affect the status bit (See Note 2)

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D.)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST3	CARRY	A,AB,ABS,AI,DEC, DECT,INC,INCT, NEG,S,SB	If CARRY OUT =1
		SLA,SRA,SRL,SRC	If last bit shifted out =1
		RTWP	If bit (3) of WR15 is 1
		LST	If bit (3) of selected WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)
ST4	OVER-FLOW	A, AB	If MSB(SA)=MSB(DA)and MSB of result ≠ MSB(DA)
		AI	If MSB(W)=MSB of IOP and MSB of result ≠ MSB(W)
		S, SB	If MSB(SA)≠MSB(DA) and MSB of result ≠MSB(DA)
		DEC,DECT	If MSB(SA)=1 and MSB of result =0
		INC,INCT	If MSB(SA)=0 and MSB of result =0
		SLA	If MSB changes during shift
		DIV	If MSB(SA)=0 and MSB(DA)=1, or if MSB(SA)=MSB(DA) and MSB of [(DA)-(SA)] =0
		DIVS	If the quotient cannot be expressed as a signed 16 bit quantity (8000_{16} is a valid negative number)
		ABS,NEG	If (SA)=8000_{16}
		RTWP	If bit (4) of WR15 is 1
		LST	If bit (4) of selected WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D.)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST4 (CONTD)		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)
ST5	ODD PARITY	CB, MOVB	If (SA) has odd number of 1's
		LDCR	If $1 \leq C \leq 8$ and (SA) has odd number of 1's. If C=0 or $9 \leq C \leq 15$, does not affect the status bit.
		STCR	If $1 \leq C \leq 8$ and the stored bits have an odd number of 1's. If C=0 or $9 \leq C \leq 15$, does not affect the status bit.
		AB,SB,SOCB,SZCB	If result has odd number of 1's.
		RTWP	If bit (5) of WR15 is 1
		LST	If bit (5) of seleted WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)
ST6	XOP	XOP	If XOP instruction is executed
		RTWP	If bit (6) of WR15 is 1
		LST	If bit (6) of selected WR is 1
		Reset Interrupt	Unconditionally sets status bit to 0
		All other Instructions and Interrupts	Do not affect the status bit (See Note 2)

TABLE 1. STATUS REGISTER BIT DEFINITIONS (CONT'D)
(SEE NOTE 1)

BIT	NAME	INSTRUCTION AND/OR INTERRUPT	CONDITION TO SET BIT TO 1, OTHERWISE SET TO 0 FOR INSTRUCTION LISTED. ALSO, THE EFFECT OF OTHER INSTRUCTIONS AND INTERRUPTS
ST7 ST8 ST9 and ST11	Unused Bits	RTWP	If corresponding bit of WR15 is 1
		LST	If corresponding bit of selected WR is 1.
		XOP, Any Interrupt	Unconditionally sets each of these status bits to 0
		All other Instructions	Do not affect these status bits (See Note 2)
ST10	ARITHMETIC OVERFLOW ENABLE	RTWP	If bit (10) of WR is 1
		LST	If bit (10) of selected WR is 1
		XOP, Any Interrupt	Unconditionally sets status bit to 0
		All other instructions	Do not affect the status bit (See Note 2)
ST12 ST13 ST14 and ST15	INTERRUPT MASK	LIMI	If corresponding bit of IOP is 1
		RTWP	If corresponding bit of WR15 is 1
		LST	If corresponding bit of selected WR is 1.
		RST, Reset and NMI Interrupts	Unconditionally sets each of these status bits to 0
		All other Interrupts	If ST12-ST15=0, no change If ST12=ST15≠0, set to one Less than level of the interrupt trap taken
		All other Instructions	Do not affect these status bits (See Note 2)

NOTES:

1. See Table 7 for Definition of Terminology Used.
2. The X instruction itself does not affect any status bit; the instruction executed by the X instruction sets status bits as defined for that instruction.

TABLE 2. DEDICATED WORKSPACE REGISTERS

REGISTER NO.	CONTENTS	USED DURING
0	Shift count (optional)	Shift instructions (SLA, SRA, SRC, and SLC)
	Multiplicand and MSW of result	Signed Multiply
	MSW of dividend and quotient	Signed Divide
1	LSW of result	Signed Multiply
	LSW of dividend and remainder	Signed Divide
11	Return Address	Branch and Link Instruction (BL)
	Effective Address	Extended Operation (XOP)
12	CRU Base Address	CRU instructions (SBO, SBZ, TB, LDCR, and STCR)
13	Saved WP register	Context switching (BLWP, RTWP, XOP, interrupts)
14	Saved PC register	Context switching (BLWP, RTWP, XOP, interrupts)
15	Saved ST register	Context switching (BLWP, RTWP, XOP, interrupts)

TABLE 3. 9995 SE PIN DESCRIPTION

SIGNAL	PIN	I/O	DESCRIPTION
			POWER SUPPLIES
Vcc	59		Supply voltage (+5V nom)
Vss	19,44		Ground reference
			CLOCKS
XTAL2/CLKIN ~~XTAL1/CLKIN~~	25	IN	Crystal input pin for internal oscillator. Also input pin for external oscillator.
XTAL1 ~~XTAL2~~	24	IN	Crystal input pin for internal oscillator.
CLKOUT1	26	OUT	Clock output signal. The frequency of CLKOUT1 is one half the oscillator input (external oscillator) or crystal (internal oscillator) frequency.
CLKOUT2	27	OUT	Clock output signal. The frequency of CLKOUT2 is one half the oscillator input (external oscillator) or crystal (internal oscillator) frequency. CLKOUT2 is one quarter cycle delayed from CLKOUT1

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			ADDRESS BUS
A0	7	OUT	Address Bus. A0 is the most significant bit of the 16 bit memory address bus and the 15 bit CRU address bus. A14 is the 2nd least significant bit of the 16 bit memory address bus and the LSB of the 15 bit CRU address bus. The address bus assumes the high impedance state when the 9995 SE is in the Hold state.
A1	8	OUT	
A2	9	OUT	
A3	10	OUT	
A4	11	OUT	
A5	12	OUT	
A6	13	OUT	
A7	14	OUT	
A8	15	OUT	
A9	16	OUT	
A10	17	OUT	
A11	18	OUT	
A12	20	OUT	
A13	21	OUT	
A14	22	OUT	
A15/CRUOUT	23	OUT	Address bit 15/CRU output data. A15/CRUOUT is the LSB of the 16 bit memory address bus and the output data line for CRU output instructions. A15/CRUOUT assumes the high impedance state when the 9995 SE is in the Hold state.
			DATA BUS
D0	43	I/O	Data bus. During memory cycles (MEMEN- active) D0 (the MSB) through D15 (the LSB) are used to transfer data to/from the external memory system. During non-memory cycles (MEMEN- inactive) D0, D1 and D2 are used to indicate whether the 9995 SE is performing a CRU cycle or an External instruction. The data bus assumes the high impedance state when the 9995 SE is in the Hold state.
D1	42	I/O	
D2	41	I/O	
D3	40	I/O	
D4	39	I/O	
D5	38	I/O	
D6	37	I/O	
D7	36	I/O	
D8	35	I/O	
D9	34	I/O	
D10	33	I/O	
D11	32	I/O	
D12	31	I/O	
D13	30	I/O	
D14	29	I/O	
D15	28	I/O	

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			CRU
CRUIN	54	IN	CRU Input data. During CRU cycles, CRUIN is the input data line for CRU input data.
CRUCLK	53	OUT	CRU clock. When active (high) CRUCLK indicates that CRU data out is available on A15/CRUOUT (when MEMEN- = 1 and D0 = D1 = D2 = 0) or that an external interface should decode External instructions (when MEMEN- = 1 and D0, D1, and D2 are not all equal to 0).
			CONTROL
MEMEN-	1	OUT	Memory enable. When active (low) MEMEN- indicates that DBIN- and the address and data buses are being used for a memory cycle. When inactive (high) MEMEN- indicates that DBIN- and the address and data buses are being used for a CRU cycle, or are indicating that the 9995 SE is performing an External instruction. MEMEN- does not assume the high impedance state when the 9995 SE is in the Hold state.
DBIN-	62	OUT	Data bus in. During memory read cycles, DBIN- is active (low) to indicate that the 9995 SE has disabled its data bus output buffers to allow external memory to enable 3-state drivers that output data onto the data bus. During CRU input cycles, DBIN- is also active to indicate that the CRU cycle is an input cycle. DBIN- assumes the high impedance state when the 9995 SE is in the Hold state.
WE-	64	OUT	Write enable. When active (low) WE- indicates that memory write data is available on the data bus. WE- assumes the high impedance state when the 9995 SE is in the Hold state.

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			CONTROL (Cont'd)
R/W-	57	OUT	Read/write indication. When high, R/W- indicates that the 9995 SE is performing a memory read cycle (if MEMEN-=0) or a CRU input cycle (if MEMEN-=1). When low, R/W- indicates that the 9995 SE is performing a memory write cycle (if MEMEN-=0) or a CRU output cycle (if MEMEN-=1). R/W- does not assume the high impedance state when the 9995 SE is in the Hold state.
READY	5	IN	Ready. When active (high), READY indicates that the present ~~external~~ memory or CRU cycle is ready to be completed. When not ready is indicated, a Wait state (defined as extension of the present cycle by one CLKOUT1 cycle) is entered. At the end of each Wait state READY is examined to determine if another Wait state is to be generated or if the cycle is to be completed.
HOLD-	63	IN	Hold state request. When active (low), HOLD- indicates to the 9995 SE that an external controller desires to use the address and date buses. Upon sensing a Hold request, the 9995 SE will enter a Hold state (defined as suspension of instruction execution) after it has completed its present cycle. ~~(see Section 2.3.1.1.3 for details of entry into a Hold state).~~ At the beginning of the Hold state, the 9995 SE places DBIN-, WE-, and the address and data buses in the high impedance state, and then responds by asserting HOLDA. When HOLD- is removed, the 9995 SE returns to normal operation.
HOLDA	60	OUT	Hold acknowledge. When active (high), HOLDA indicates that it is in the Hold state.
IAQ	61	OUT	Instruction acquisition. When active (high), IAQ indicates that the memory read cycle in progress is that of reading an instruction opcode.

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			CONTROL (Cont'd)
ST8-	58	OUT	Status Bit 8. ST8- is the inverted contents of Status Register bit ST8 (MAP0/MAP1) only during memory cycles (MEMEN-=0). During all other cycles (MEMEN-=1) ST8- is held high.
WORD/BYTE-	56	OUT	Word/Byte. When low, WORD/BYTE- indicates that the memory cycle in progress is that of accessing a single byte on the 16-bit data bus. WORD/BYTE- is held high at all other times.

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			INTERRUPTS
RESET-	4	IN	Reset. When active (low) RESET- causes the 9995 SE to enter a RESET state ~~(see Section 2.3 Table 2-1)~~ and inhibit its control signals. When RESET- is released, the 9995 SE initiates a level zero interrupt sequence that acquires WP and PC from memory word addresses 0000 and 0002, and begins execution using this vector. RESET- will terminate an Idle state. RESET- is a Schmitt-trigger input.
NMI-	2	IN	Non-maskable Interrupt. When active (low), NMI- causes the 9995 SE to execute a non-maskable interrupt sequence with the trap vector (WP and PC) in memory word addresses FFFC and FFFE. NMI- will terminate an Idle state. NMI- is recognized only once for each high-to-low transition. (NMI- must be taken inactive before it will be recognized again).
INTREQ-	50	IN	Interrupt request. When active (low), INTREQ- indicates that an external interrupt is requested and the code of its level is present on IC0 through IC3.
IC0 (MSB) IC1 IC2 IC3 (LSB)	49 48 47 46	IN IN IN IN	Interrupt code. IC0 through IC3 are sampled when INTREQ- is active. When IC0 through IC3 are LLLH, the highest priority maskable external interrupt is being requested, and when HHHH, the lowest priority maskable external interrupt is being requested.

TABLE 3. 9995 SE PIN DESCRIPTION (CONT'D)

SIGNAL	PIN	I/O	DESCRIPTION
			SPECIAL SIGNALS
9995 SE/9990–	45	IN	9995 SE memory control/9990 memory control. This signal, which is permanently connected to either Vcc or Ground in a system, tells the 9995 SE device which memory map to use with its internal memory control logic.
DESTWRITE	55	OUT	Destination Write. When high, DESTWRITE indicates that a write cycle to the destination address of the instruction being completed will occur after the current fetch of the next instruction is completed.
FREEZE–	3	IN	Freeze. When active (low) the 9995 SE is conditioned for the immediately following RESET–. The conditioning consists of ensuring that during the following RESET– the 9995 SE inhibits (keeps at inactive logic level) WE–, DBIN–, MEMEN– and CRUCLK, and the 9995 SE will not change the contents of its PC, WP, or ST registers.
IACK	51	OUT	Interrupt Acknowledge. IACK is active (high) when the 9995 SE is reading the WP of the vector during an interrupt (any interrupt, including RESET, NMI, and MID) context switch.
MID	52	OUT	MID Interrupt. MID is active (high) when the 9995 SE is reading the WP of the vector during an MID interrupt context switch.

TABLE 4. INTERRUPT LEVEL DATA

PRIORITY LEVEL	VECTOR LOCATION (Memory Address, In Hex	MASK VALUES TO ENABLE ACCEPTING THE INTERRUPT (ST12 thru ST15)	VALUE MASK SET TO UPON TAKING THE INTERRUPT (ST12 thru ST15)	SOURCE AND ASSIGNMENT
0 (Highest Priority)	0000	0000 thru 1111 (See Note 1)	0000	External: Reset (RESET- Signal)
MID	0008 (See Note 2)	0000 thru 1111 (See Note 1)	0001 (See Note 2)	Internal: MID
NMI	FFFC	0000 thru 1111 (See Note 1)	0000	External: (NMI- Signal)
1	0004	0001 thru 1111	0000	External: IC0-IC3 = 0001
2	0008 (See Note 2)	0010 thru 1111 (See Note 3)	0001 (See Note 2)	Int. and/or Ext.: Arithmetic Overflow, IC0-IC3 = 0010
3	000C	0011 thru 1111	0010	External: IC0-IC3 = 0011
4	0010	0100 thru 1111	0011	External: IC0-IC3 = 0100
5	0014	0101 thru 1111	0100	External: IC0-IC3 = 0101
6	0018	0110 thru 1111	0101	External: IC0-IC3 = 0110
7	001C	0111 thru 1111	0110	External: IC0-IC3 = 0111
8	0020	1000 thru 1111	0111	External: IC0-IC3 = 1000
9	0024	1001 thru 1111	1000	External: IC0-IC3 = 1001
10	0028	1010 thru 1111	1001	External: IC0-IC3 = 1010
11	002C	1011 thru 1111	1010	External: IC0-IC3 = 1011
12	0030	1100 thru 1111	1011	External: IC0-IC3 = 1100
13	0034	1101 thru 1111	1100	External: IC0-IC3 = 1101
14	0038	1110 and 1111	1101	External: IC0-IC3 = 1110
15 (Lowest Priority)	003C	1111	1110	External: IC0-IC3 = 1111

NOTES:

1. Level 0, MID, and NMI cannot be disabled with the Interrupt Mask.
2. MID and Level 2 use the same trap vector and change the Interrupt Mask to the same value.
3. Generation of a Level 2 request by an Arithmetic Overflow condition (ST4 set to 1) is also enabled/disabled by bit ST10 of the Status Register.

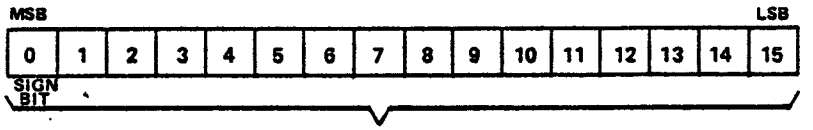
TABLE 5: 9995 SE EXTERNAL INSTRUCTION CODES

INSTRUCTION	CODE DURING CYCLE		
	D0	D1	D2
CRU: SBO,SBZ,TB, LDCR or STCR	0	0	0
IDLE	0	1	0
RSET	0	1	1
CKON	1	0	1
CKOF	1	1	0
LREX	1	1	1

TABLE 6. DEFINITION OF TERMINOLOGY

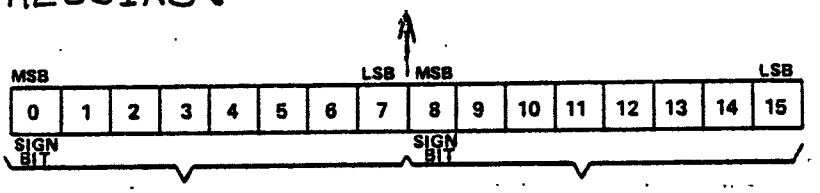
TERM	DEFINITIONS
B	Byte Indicator (1 = byte; 0 = word)
C	Bit Count
D	Destination address register
DA	Destination address
IOP	Immediate operand
LSB(n)	Least significant (right most) bit of (n)
MSB(n)	Most significant (left most) bit of (n)
N	Don't care
PC	Program Counter
Result	Result of operation performed by instruction
S	Source address register
SA	Source address
ST	Status register
STn	Bit n of status register
TD	Destination address modifier
TS	Source address modifier
W	Workspace register
WRn	Workspace register n
(n)	Contents of n
a→b	a is transferred to b
|n|	Absolute value of n

WORD ADDRESSING:



Even 16-bit Byte Address

BYTE ADDRESSING:



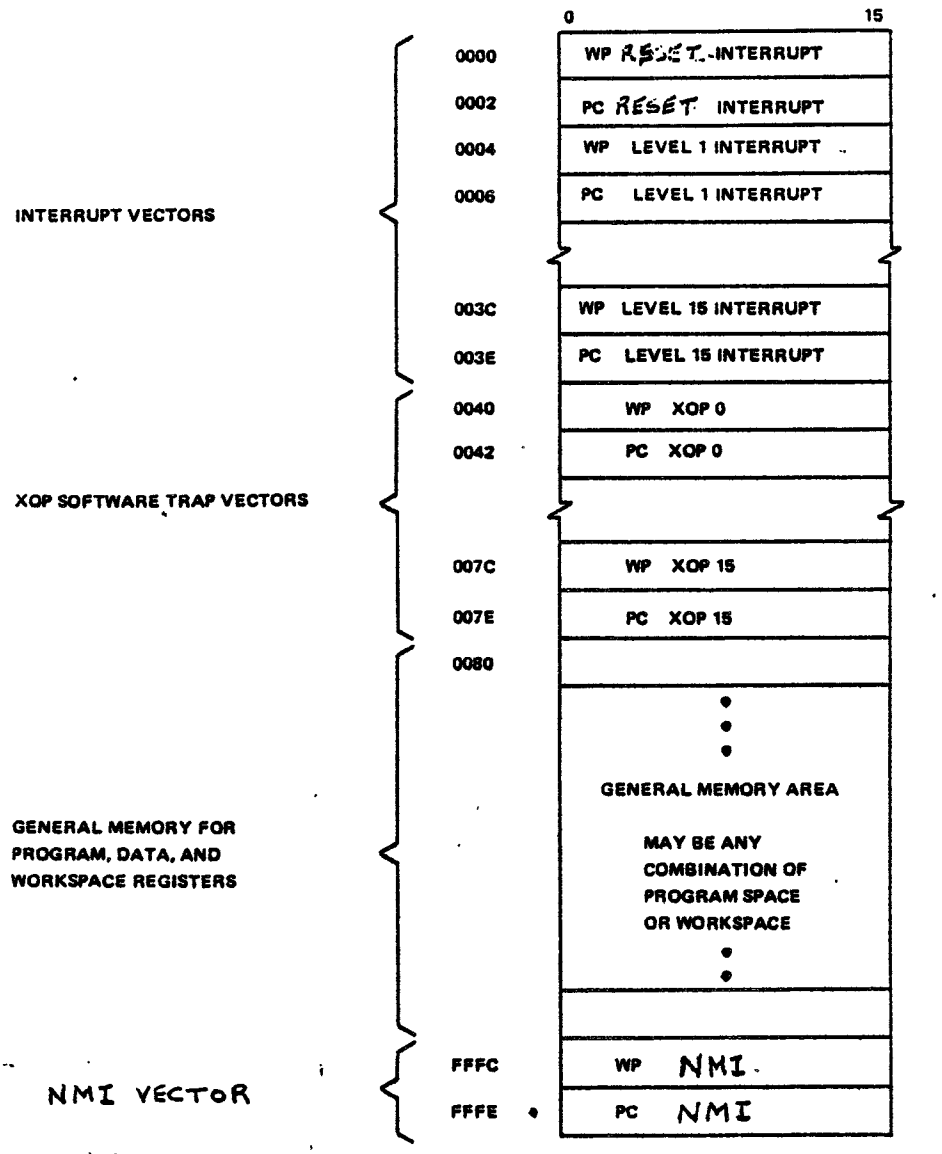
Even 16-bit Byte Address

Odd 16-bit Byte Address

Odd Address = Even Address + 1

WORD AND BYTE FORMATS

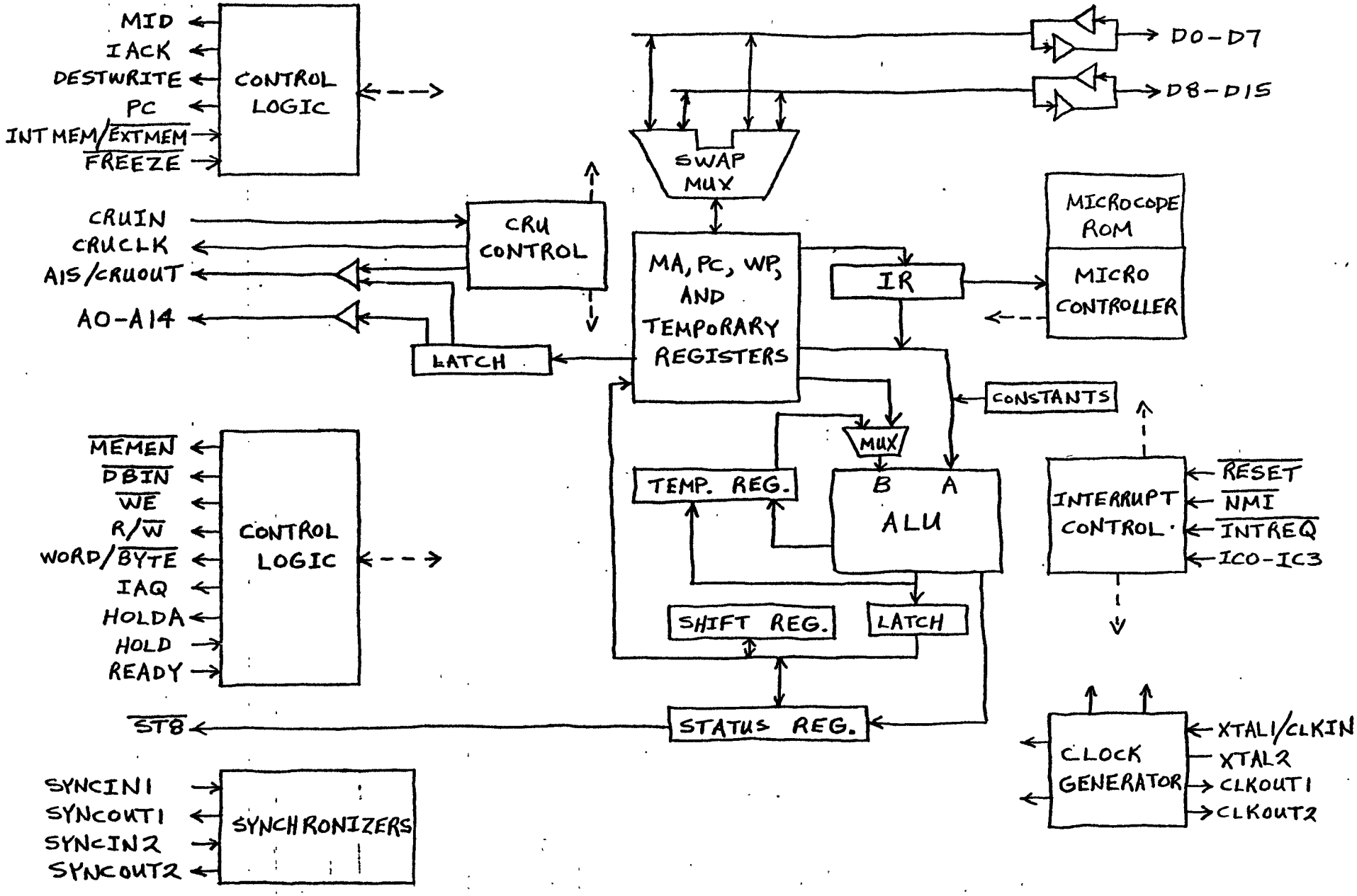
FIGURE 1.



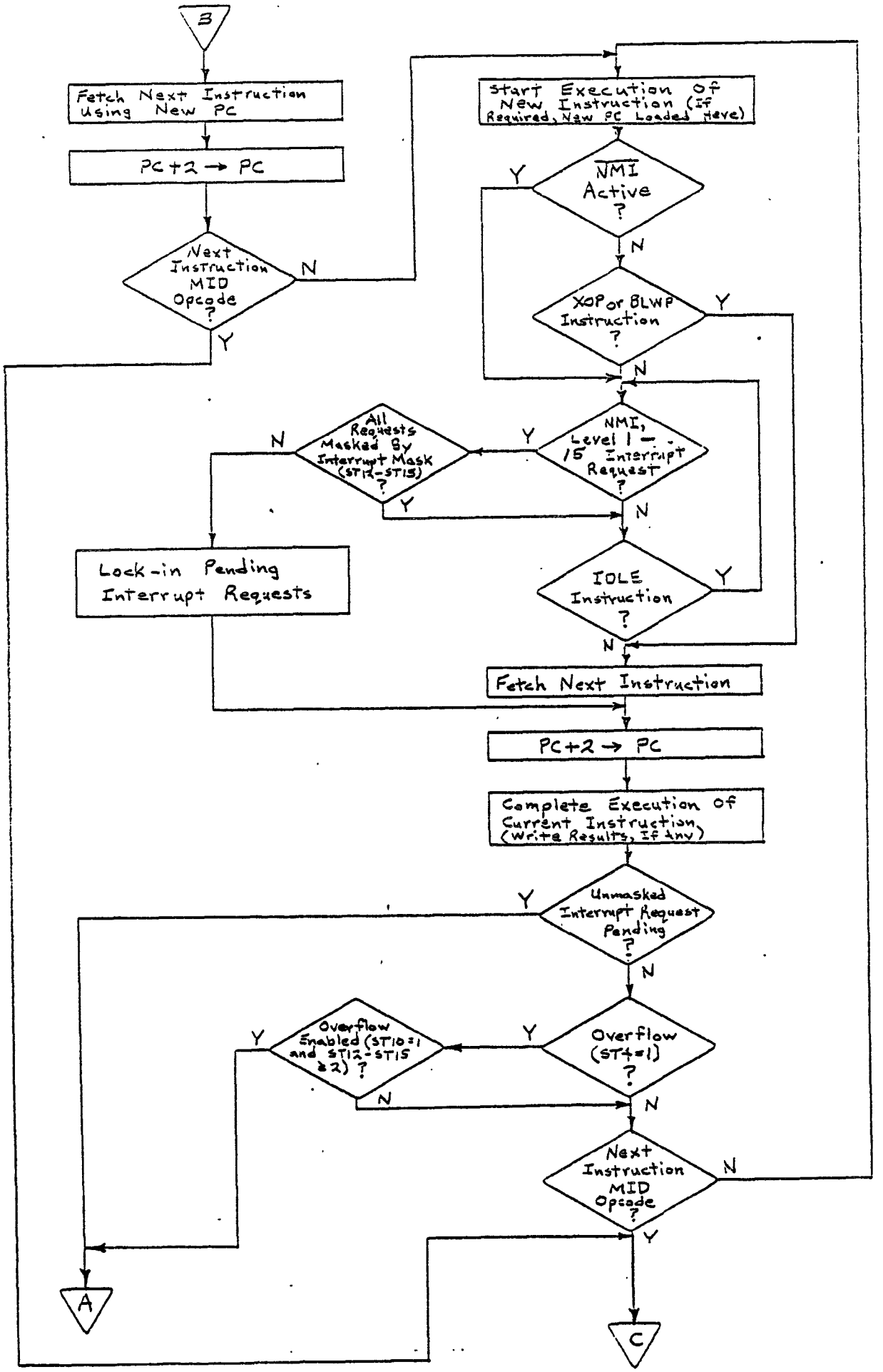
Note: Addresses Are Word Addresses In Hex

9995 SE MEMORY MAP

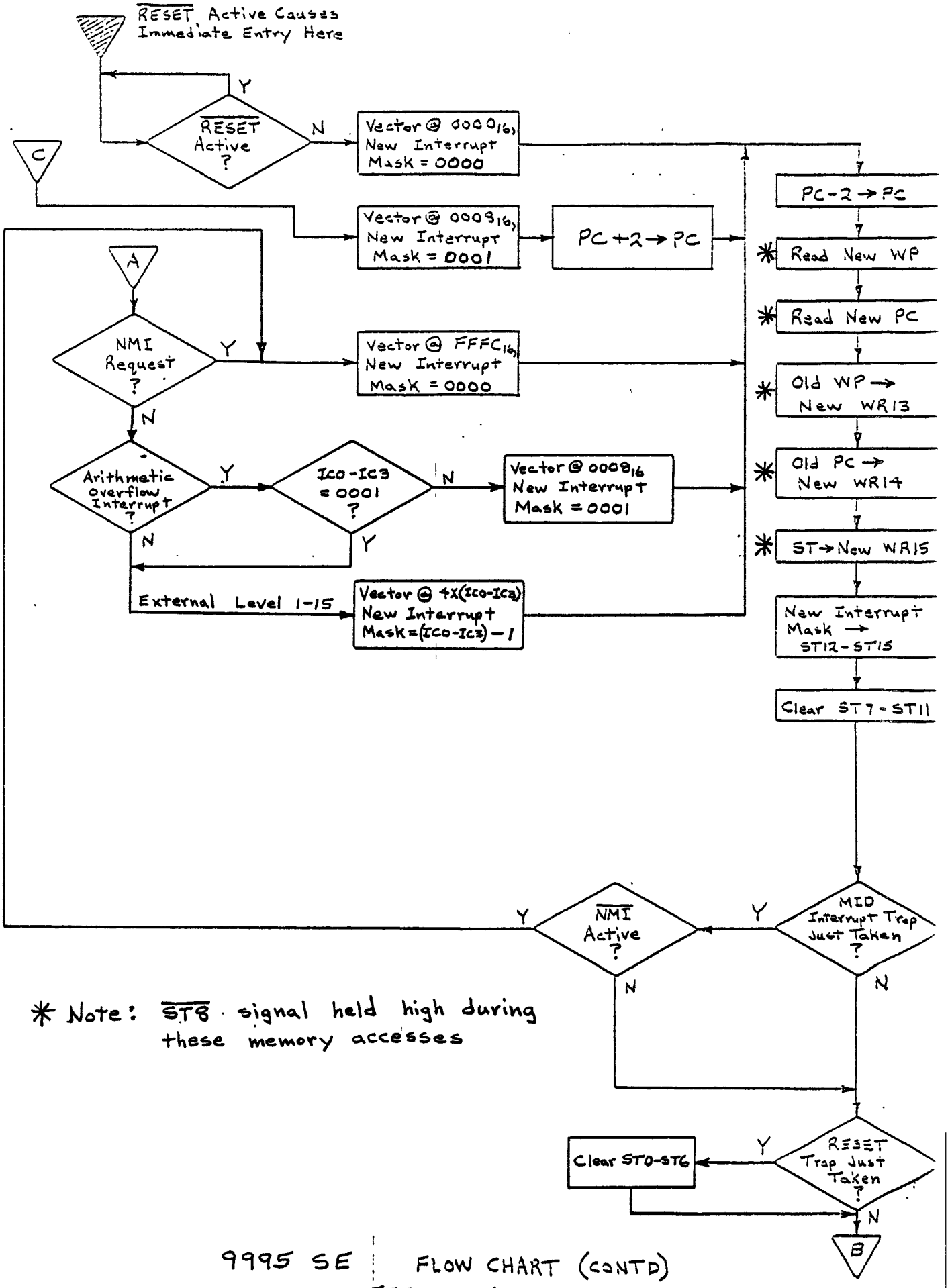
FIGURE 2.



9995 SE BLOCK DIAGRAM
FIGURE 3.



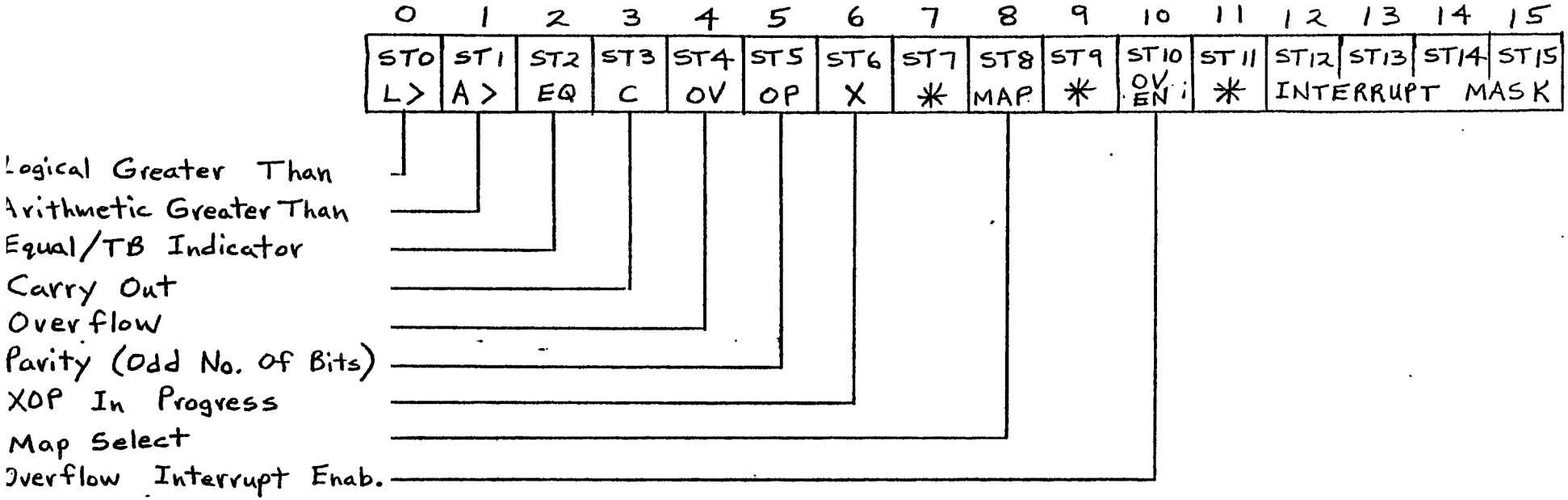
9995 SE FLOW CHART



* Note: $\overline{ST8}$ signal held high during these memory accesses

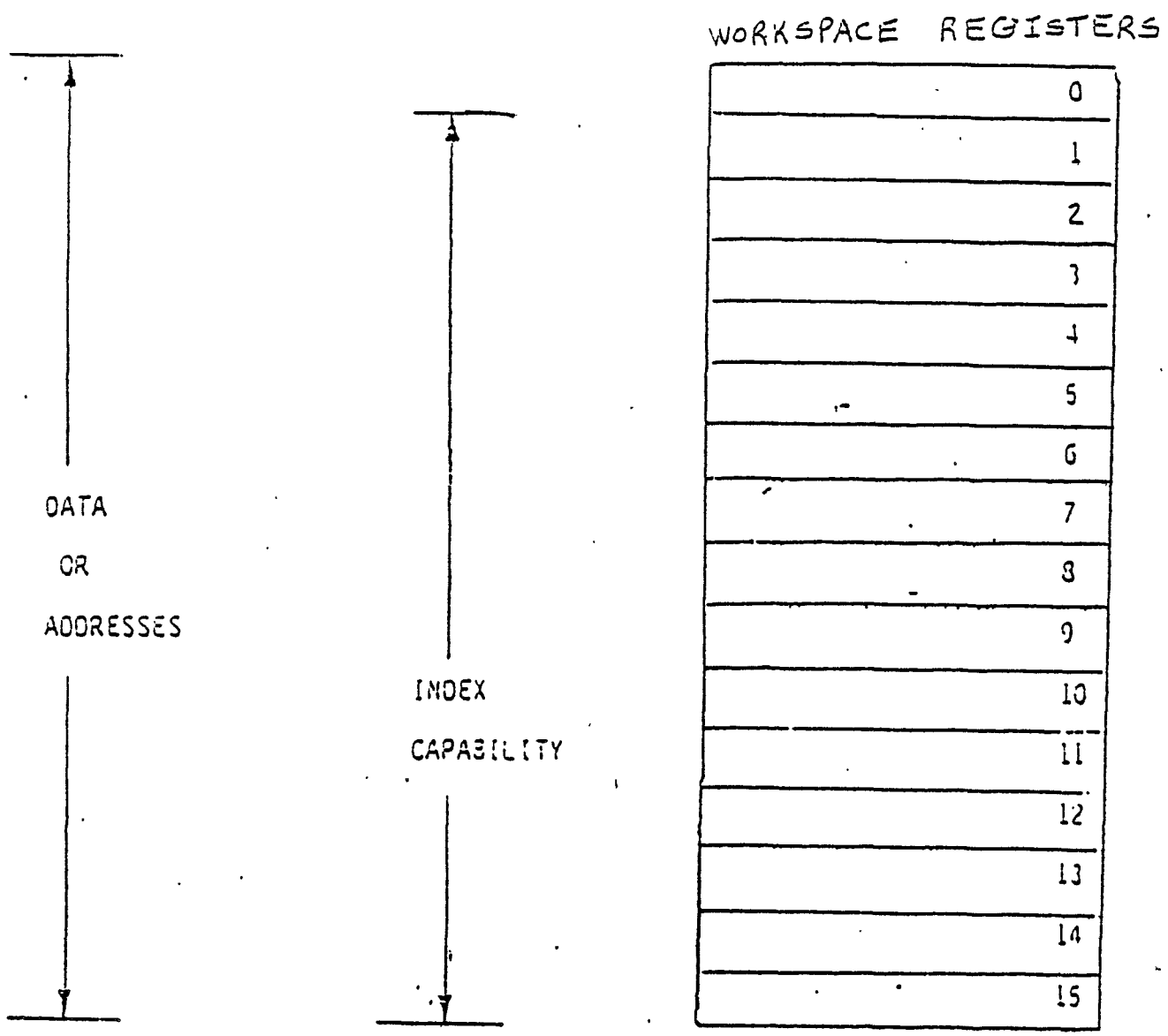
9995 SE FLOW CHART (CONTD)

FIGURE 4.



* Note: ST7, ..ST9, and ST11 are not used in the 9995 SE but still physically exist in the register. These bits could therefore be used as flag bits, but software transportability should be kept in mind when doing so as these bits are defined in other 9900 microprocessor family and 990 minicomputer family products.

STATUS REGISTER BIT ASSIGNMENTS
FIGURE 5.



NOTE: The WP register contains the address of workspace register zero.

FIGURE. 6 WORKSPACE REGISTERS USABLE AS INDEX REGISTERS

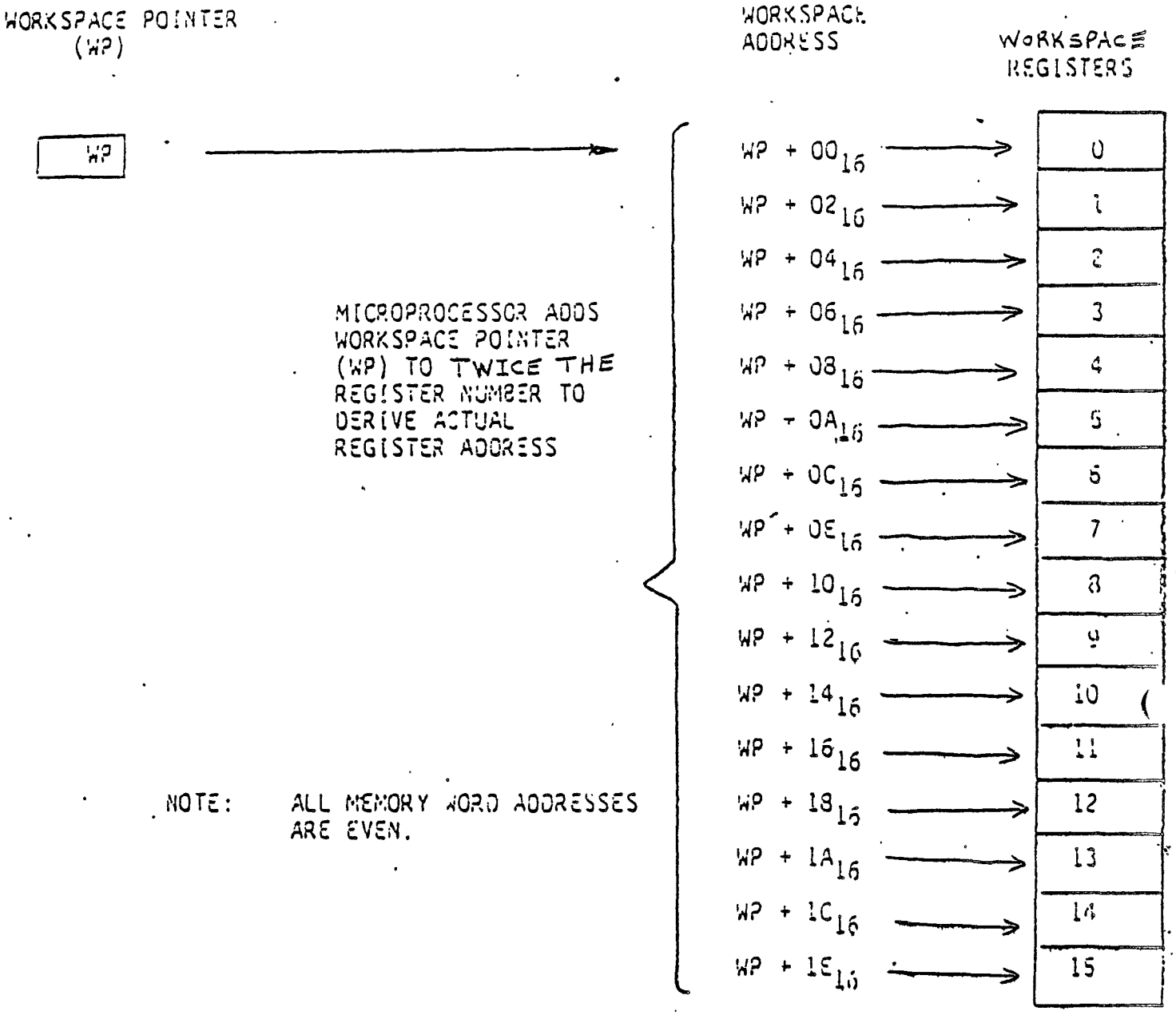


FIGURE 7 WORKSPACE POINTER AND REGISTERS

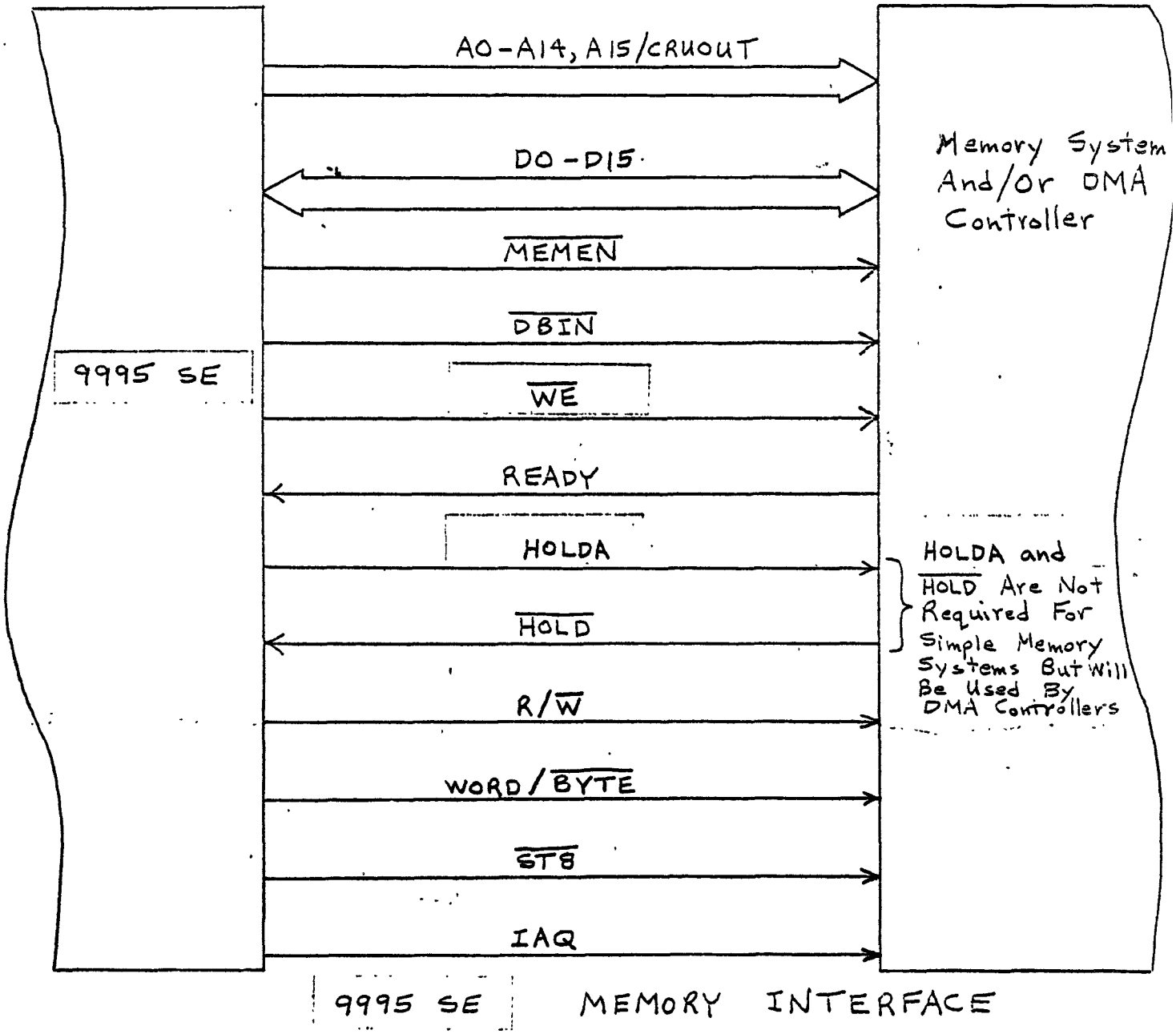
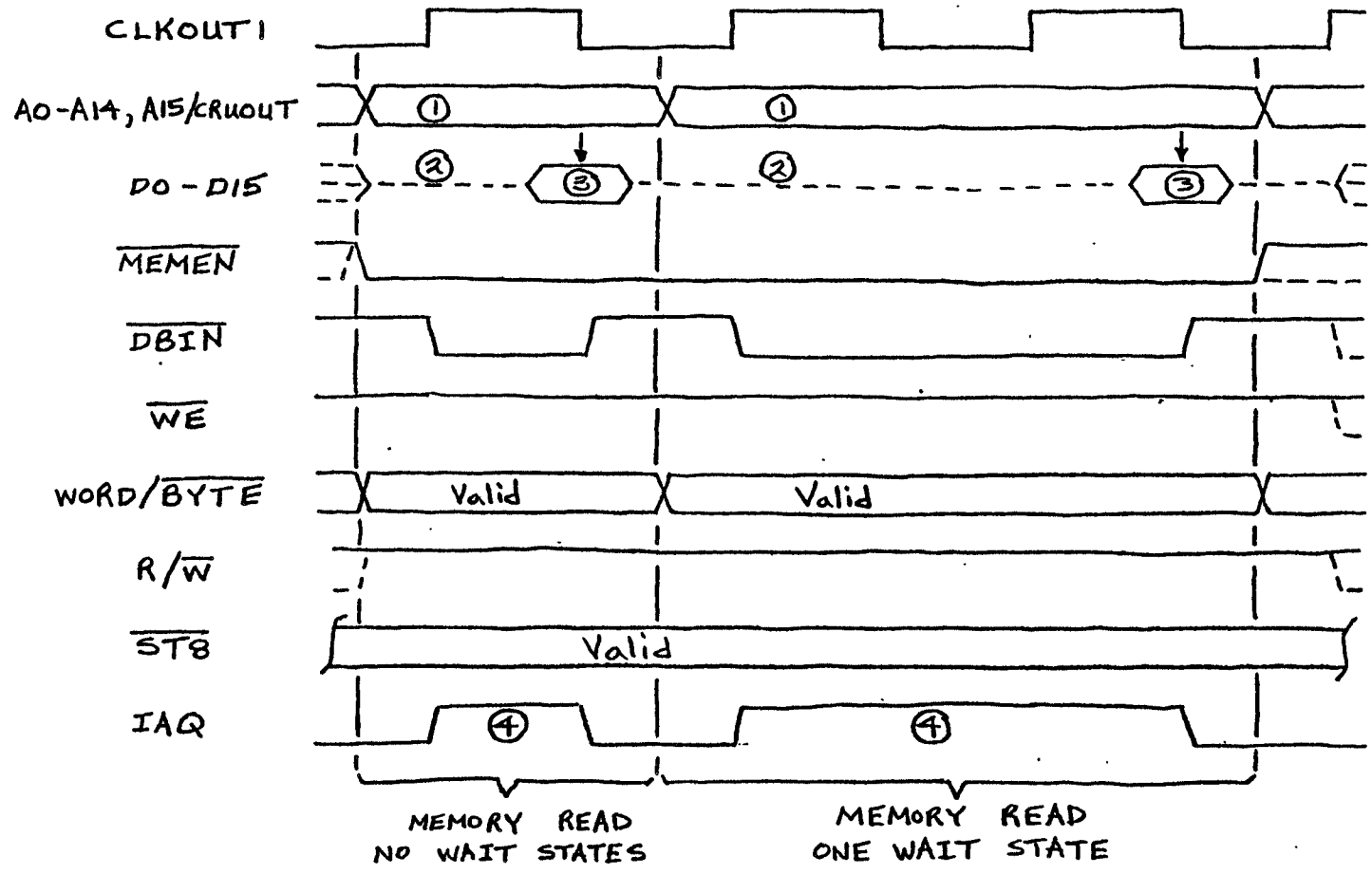

A0-A14, A15/CRUOUT
D0-D15
MEMEN
DBIN
WE
READY
HOLDA
HOLD
R/W
WORD/BYTE
ST8
IAQ
9995 SE
Memory System And/Or DMA Controller
HOLDA and HOLD Are Not Required For Simple Memory Systems But Will Be Used By DMA Controllers
9995 SE MEMORY INTERFACE

FIGURE 8.

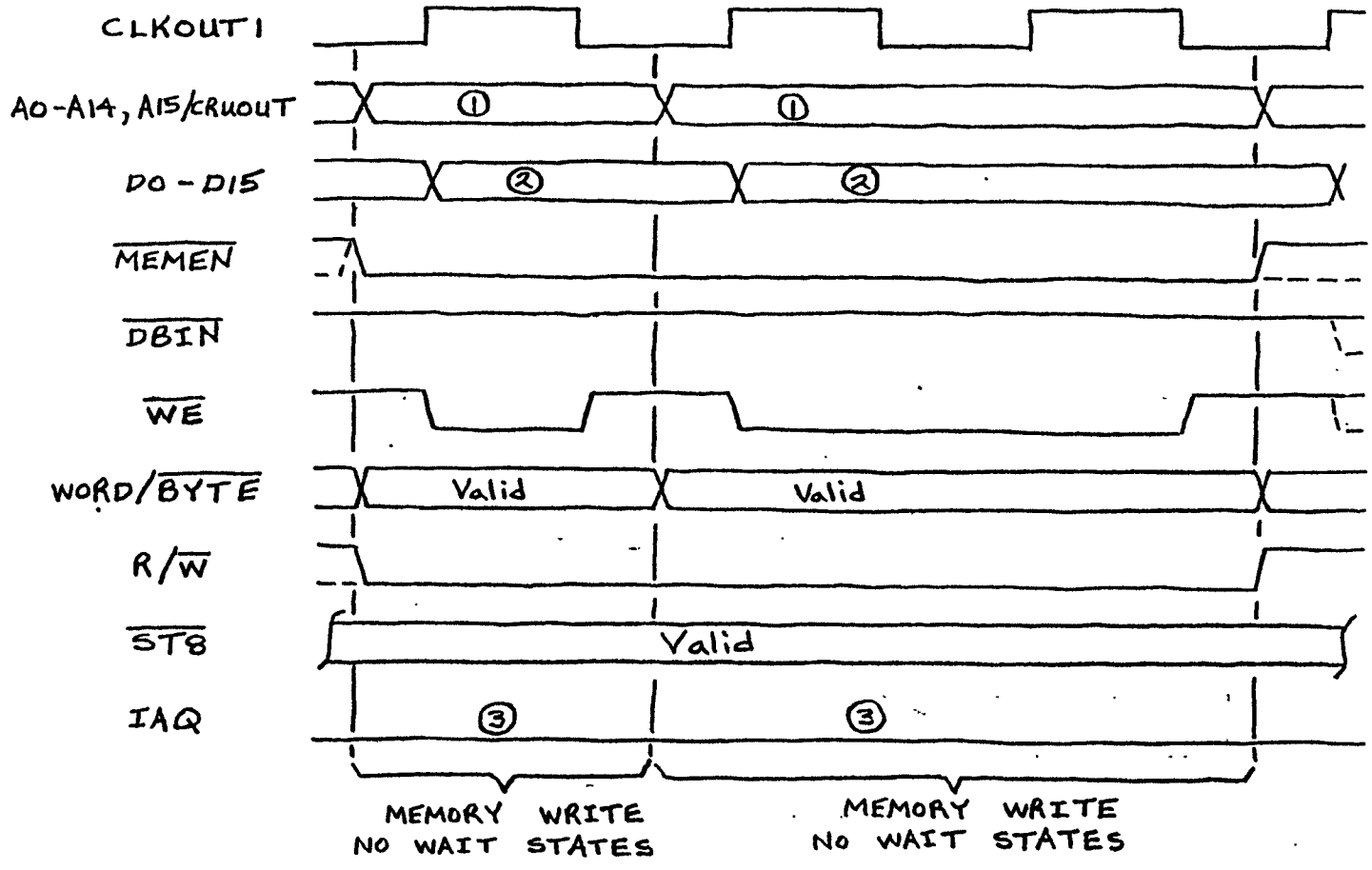


Notes:

1. Valid Address
2. In Input Mode (Drivers @ High-Z)
3. Memory Read Data Must Be Valid At CLKOUT1 Edge Indicated
4. IAQ Will Only Be Asserted During Memory Read Cycles If An Instruction Opcode Is Being Read

9995 SE MEMORY READ CYCLE

FIGURE 9.

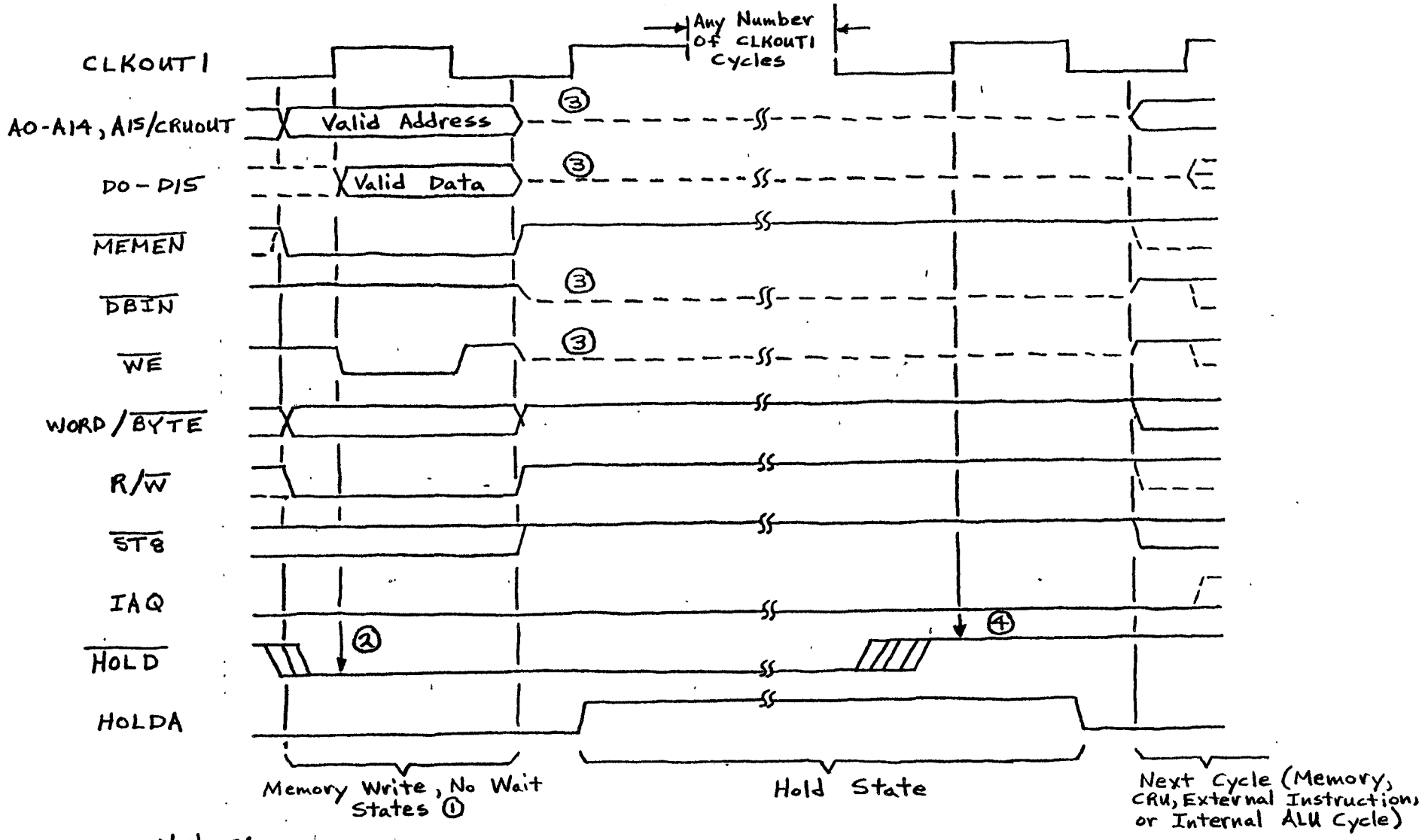


Notes:

① Valid Address
② Valid Memory Write Data
③ IAQ Will Never Be Asserted During A Memory Write Cycle

9995 SE MEMORY WRITE CYCLE

FIGURE 10.



Notes:

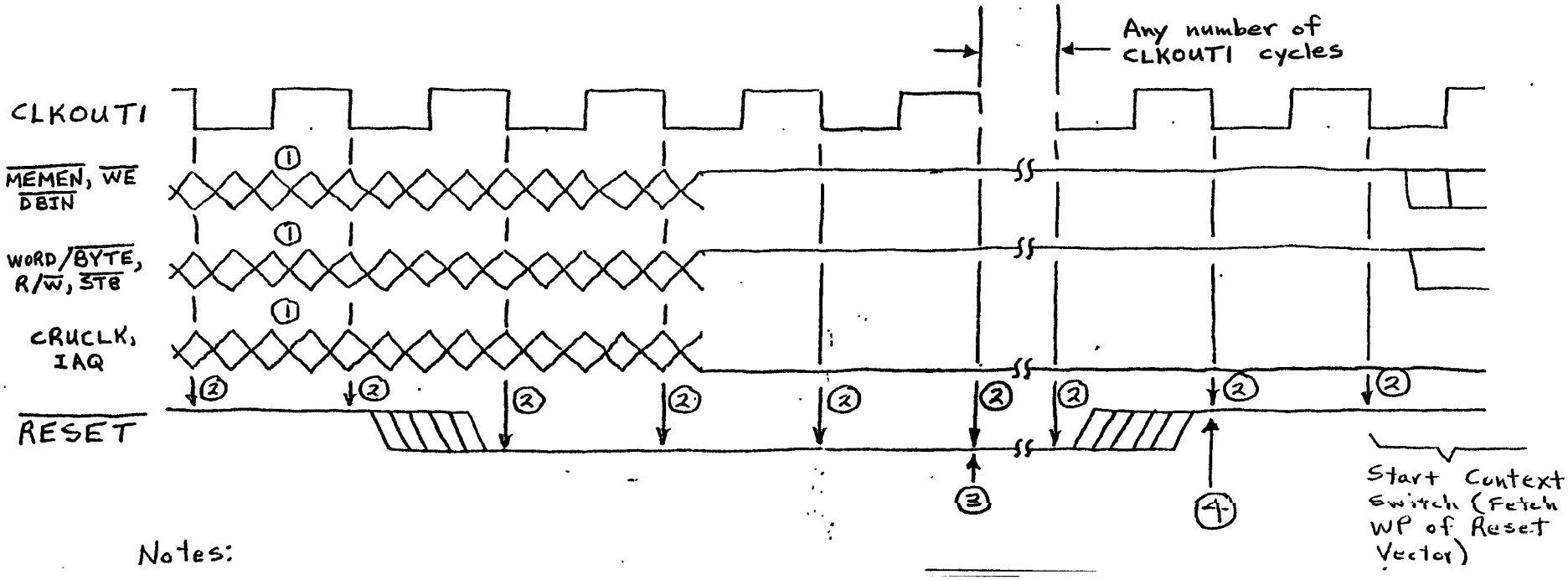
① Cycle before the Hold State could have been Memory (With any number of Wait State), CRU, External Instruction, or Internal ALU

② $\overline{\text{HOLD}}$ must be valid at last low-to-high transition of a cycle for next low-to-high CLKOUT1 transition to begin a Hold State

③ In High-impedance mode (output drivers)

④ Next cycle will begin after first low-to-high CLKOUT1 transition at which $\overline{\text{HOLD}}$ is high

FIGURE 11.



Notes:

① Don't care XXXX indicates that any type of 9995 SE cycle can be taking place

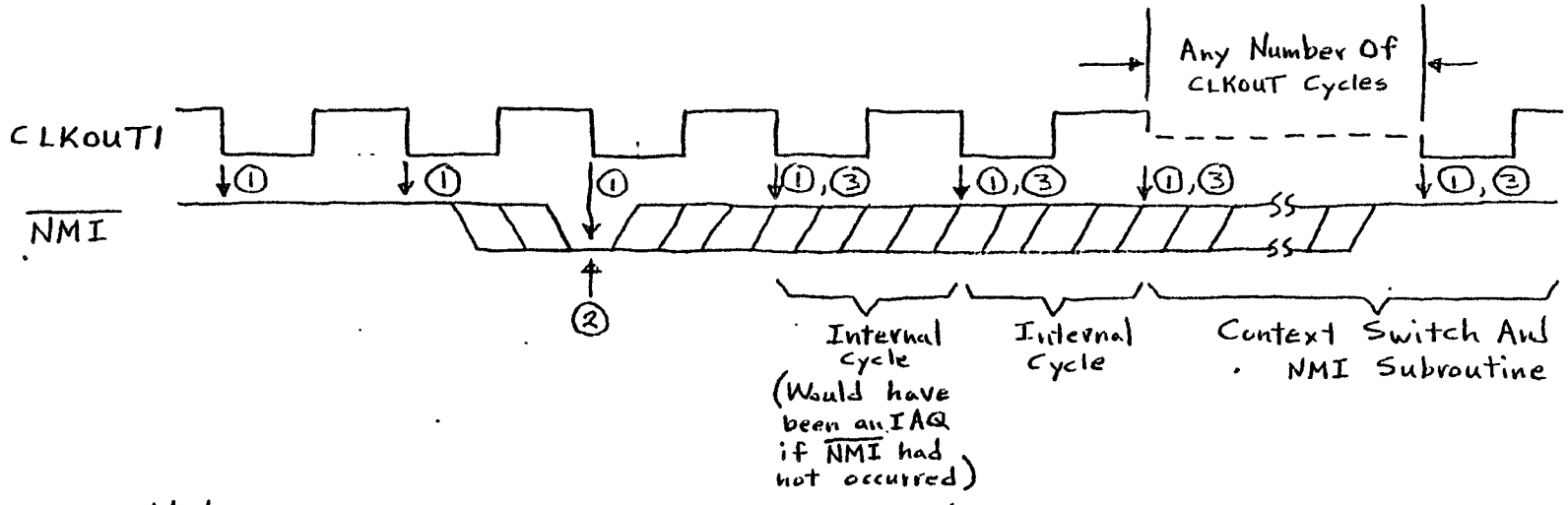
② $\overline{\text{RESET}}$ is sampled at every high-to-low CLKOUT1 transition

③ $\overline{\text{RESET}}$ is required to be active (low) for a minimum of four samples to initiate the sequence. The context switch would begin two CLKOUT1 cycles after ③ if $\overline{\text{RESET}}$ were inactive (high) at the next CLKOUT1 falling edge after ③

④ The context switch using the Reset trap vector begins one CLKOUT1 cycle after $\overline{\text{RESET}}$ is sampled as having returned to the inactive (high) level.

9995 SE RESET SIGNAL TIMING RELATIONSHIPS

FIGURE 12



Notes:

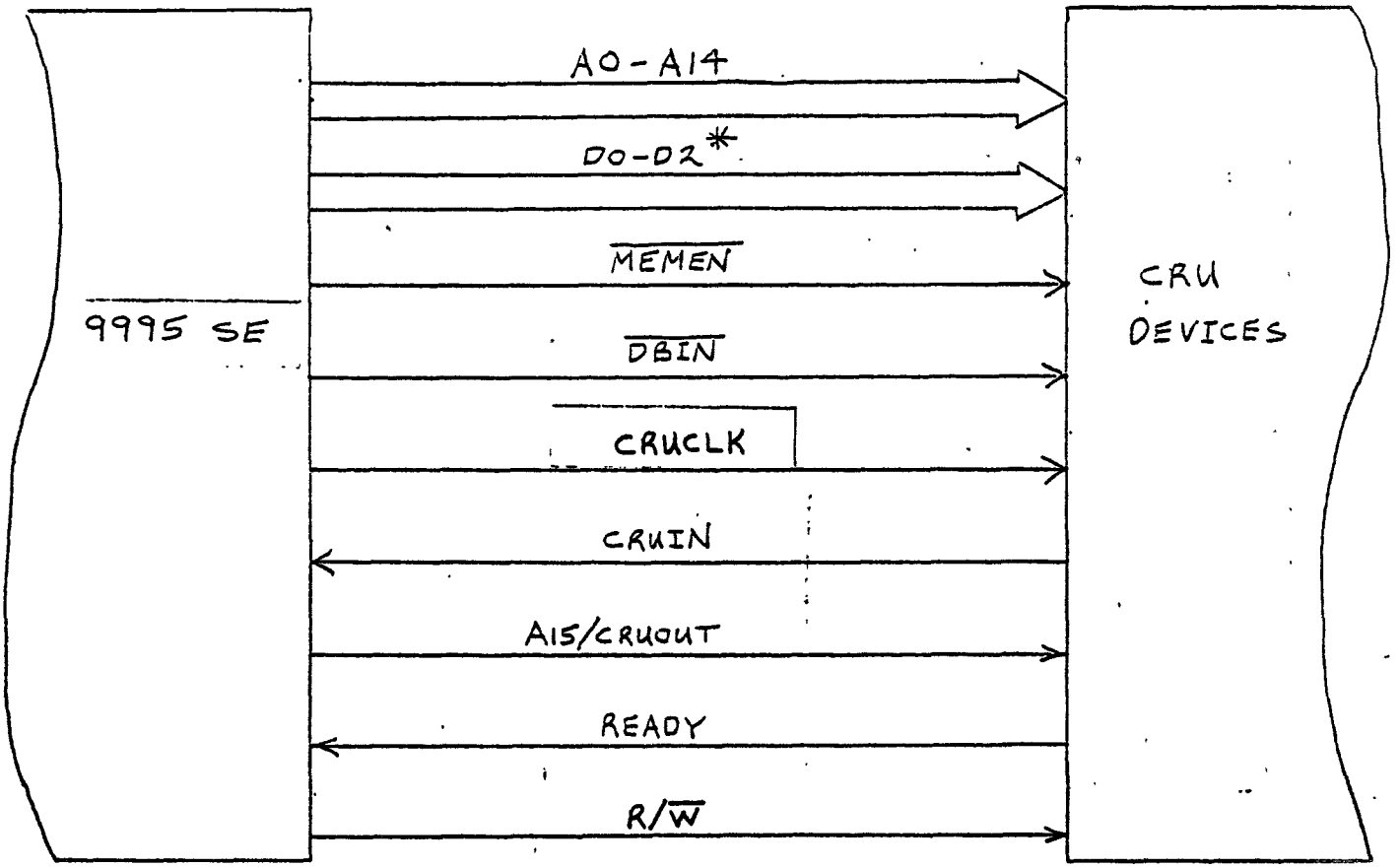
① $\overline{NMI}$ is sampled at every high-to-low CLKOUT1 transition

② To be recognized, $\overline{NMI}$ must be active (low) at the end of an instruction. Since instructions are variable in length, the minimum active time for $\overline{NMI}$ is variable according to the instruction being executed. Shown by ② is the last possible time that $\overline{NMI}$ must be recognized at or by before execution of the next instruction will commence. The NMI context switch begins three CLKOUT1 cycles after execution of the current instruction is complete.

③ After an NMI context switch sequence has been initiated, $\overline{NMI}$ can remain active (low) indefinitely without causing consecutive NMI trap requests. To enable another NMI trap request, $\overline{NMI}$ must be taken inactive (high) and be sampled at least once at the inactive level.

9995 SE | NMI SIGNAL TIMING RELATIONSHIPS

FIGURE 13

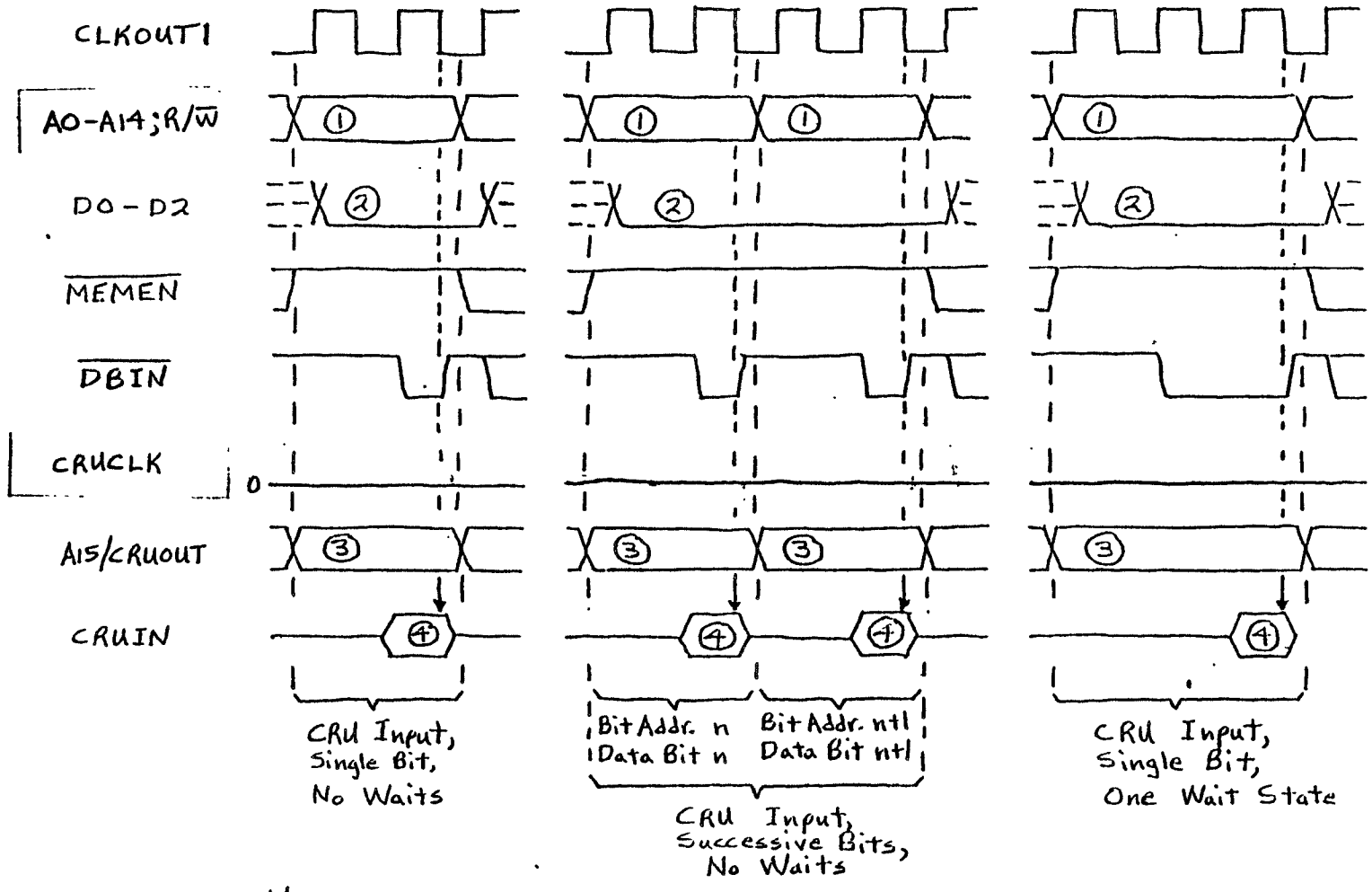


* Note:

D0-D2 are used to distinguish between CRU and external instruction cycles. If external instructions are not used in a system, D0-D2 are not necessary in the CRU interface.

9995 SE CRU INTERFACE

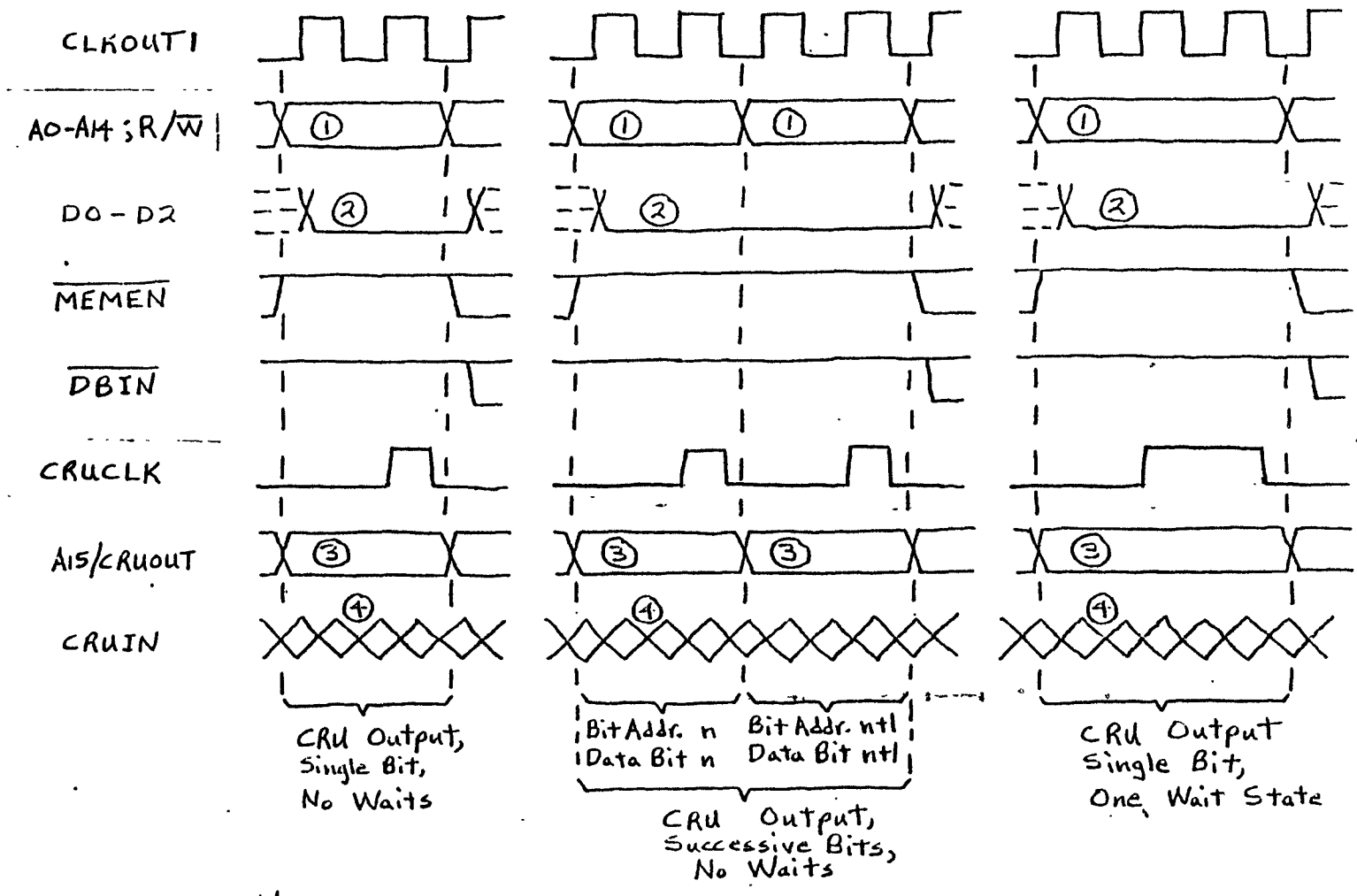
FIGURE 14



Notes:

① Valid Address; R/W = 1
② D0-D2 Each Output Logic Zero
③ Non-specific Output Bit
④ CRU Input Bit Must Be Valid On CRUIN At CLKOUT1 Edge Indicated

9995 SE. CRU INPUT CYCLE
FIGURE 15



Notes:

1. Valid Address; R/$\overline{W}$=0
2. D0-D2 Each Output Logic Zero
3. Valid CRU Output Bit For Address Being Output
4. Don't Care

9995 SE | CRU OUTPUT CYCLE

FIGURE 16.

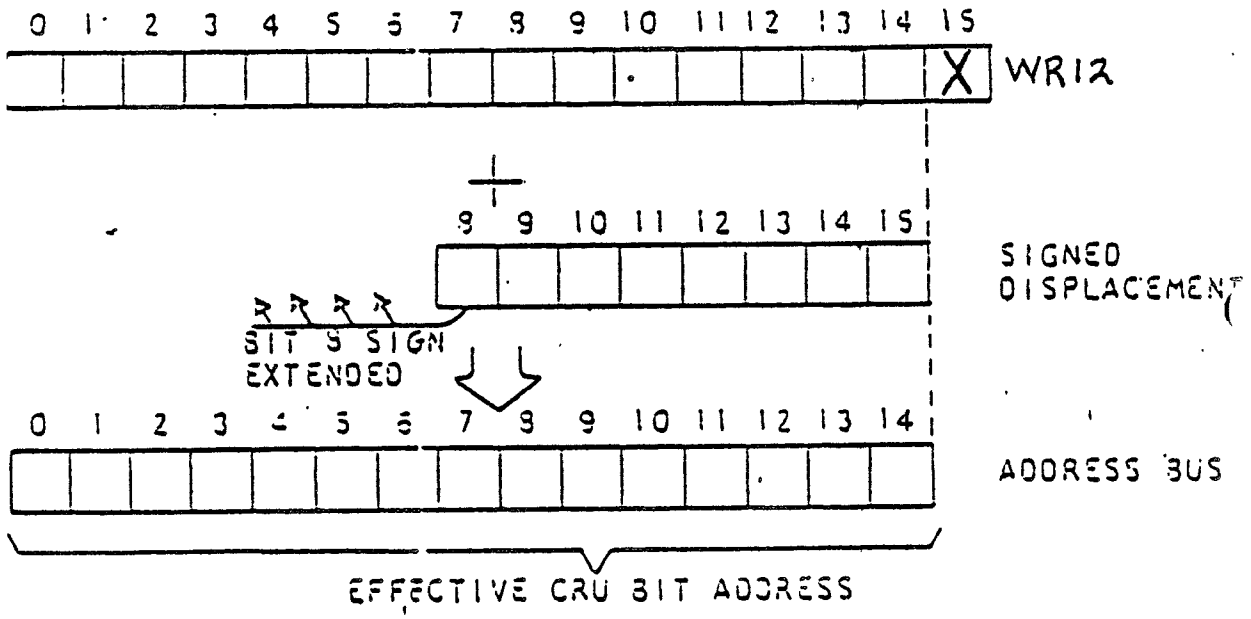


FIGURE 17. SINGLE BIT CRU ADDRESS DEVELOPMENT

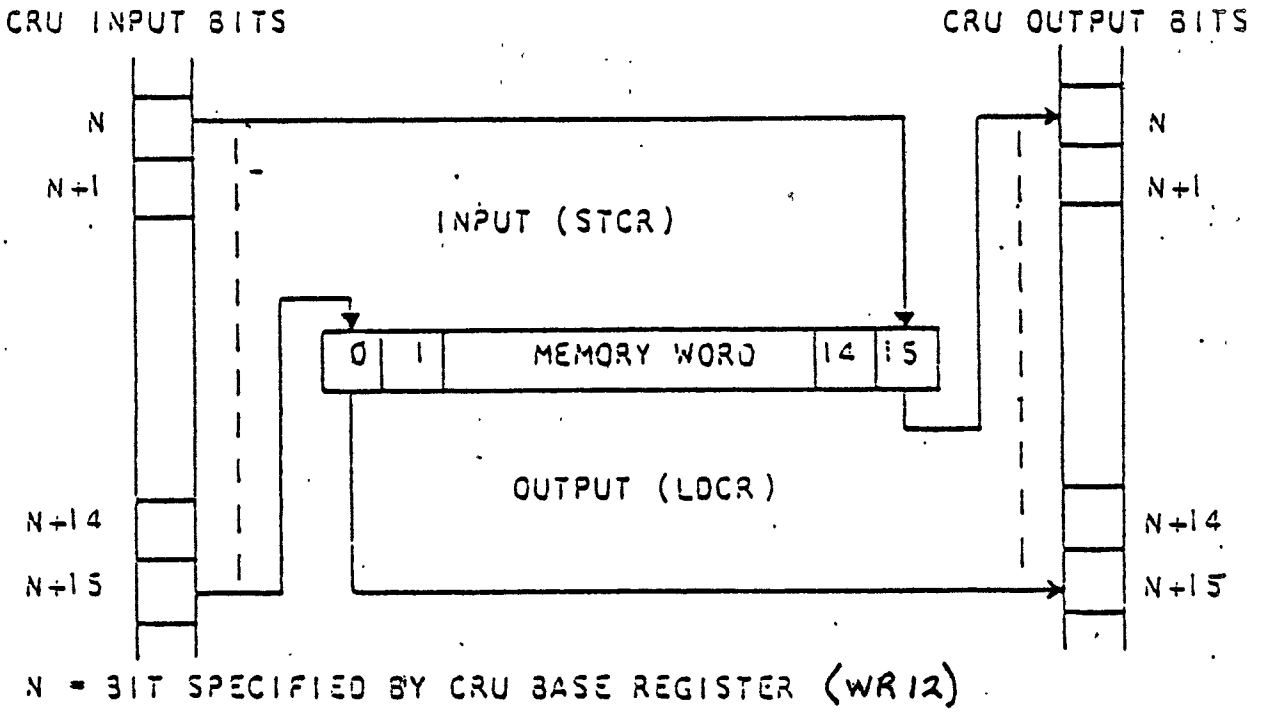
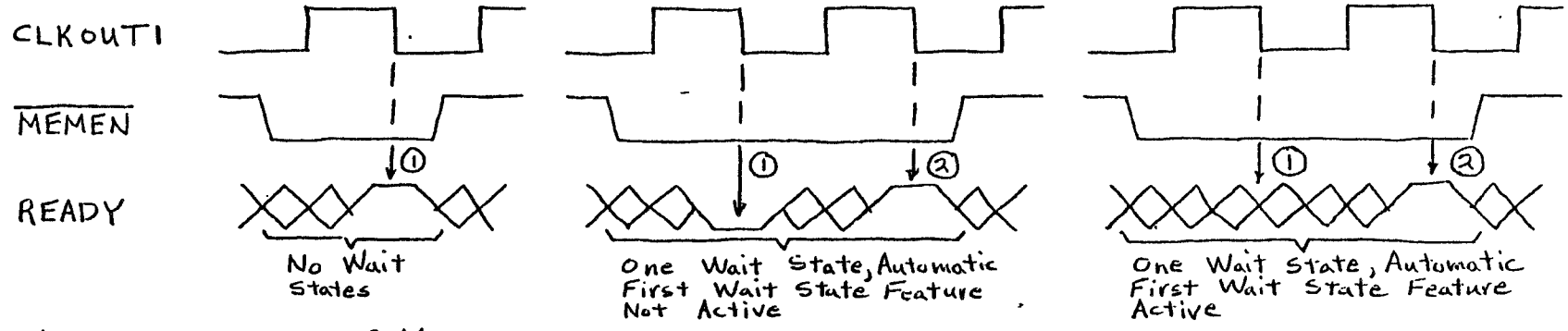
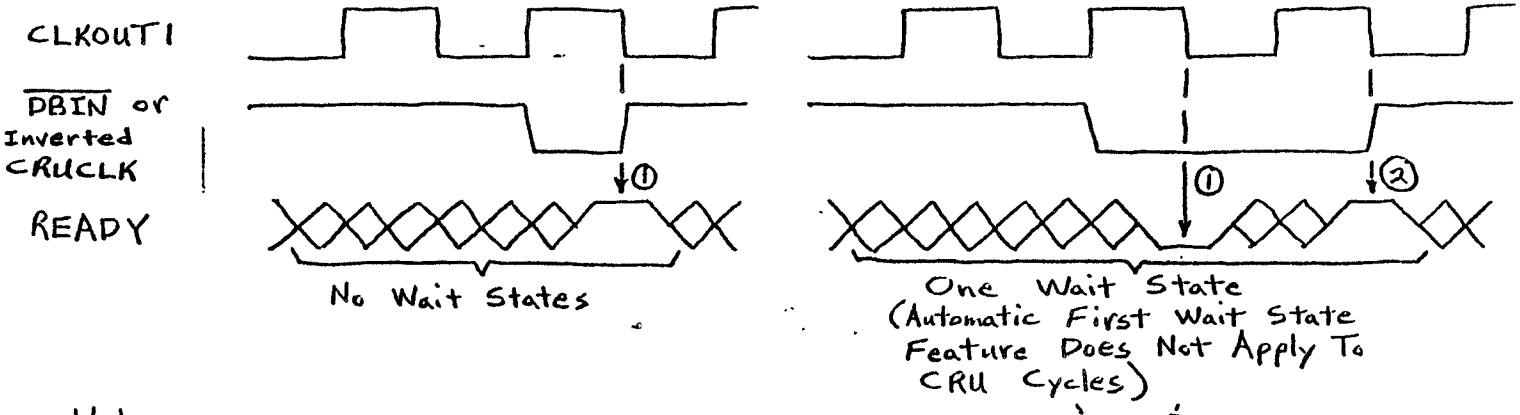


FIGURE 18. LDCR/STCR DATA TRANSFERS

MEMORY CYCLES:



CRU CYCLES AND EXTERNAL INSTRUCTION CYCLES:



Notes:

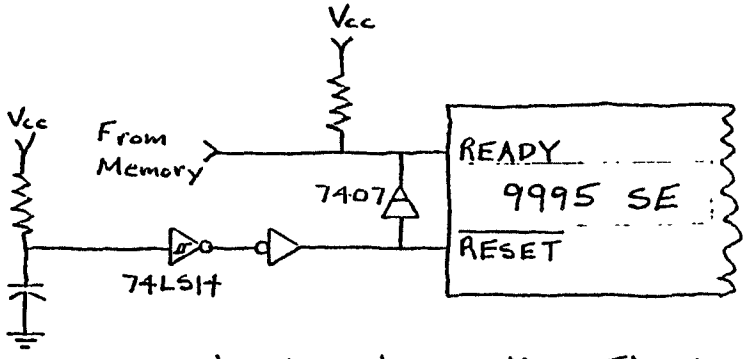
① First Sample Time Of READY In Cycle

② Second Sample Time Of READY In Cycle. Additional Wait States Can Be Generated By Keeping READY Low At This And Subsequent Sample Times.

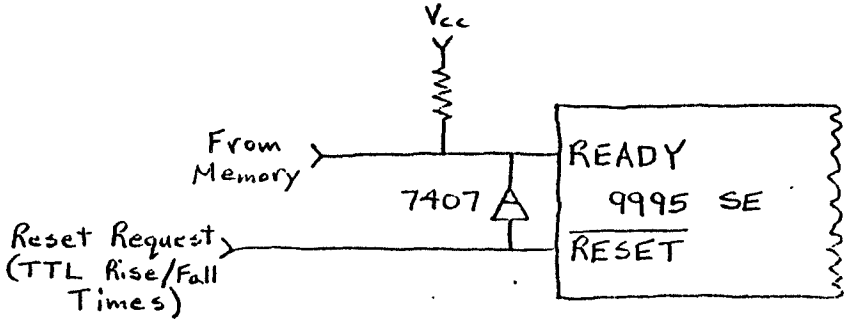
XXXXX Denotes Don't Care

WAIT STATE GENERATION FOR MEMORY, CRU, AND EXTERNAL INSTRUCTION CYCLES

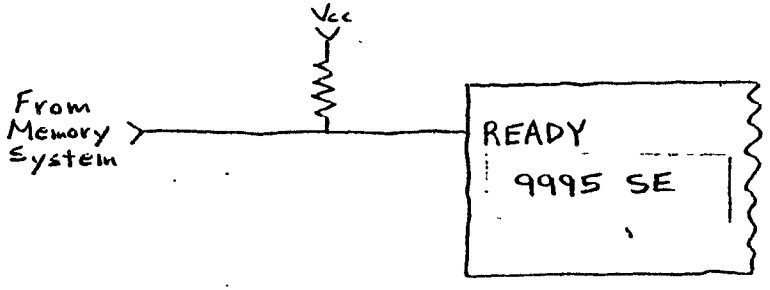
FIGURE 9.



(a) Inhibiting Automatic First Wait State, R-C Power-up Reset



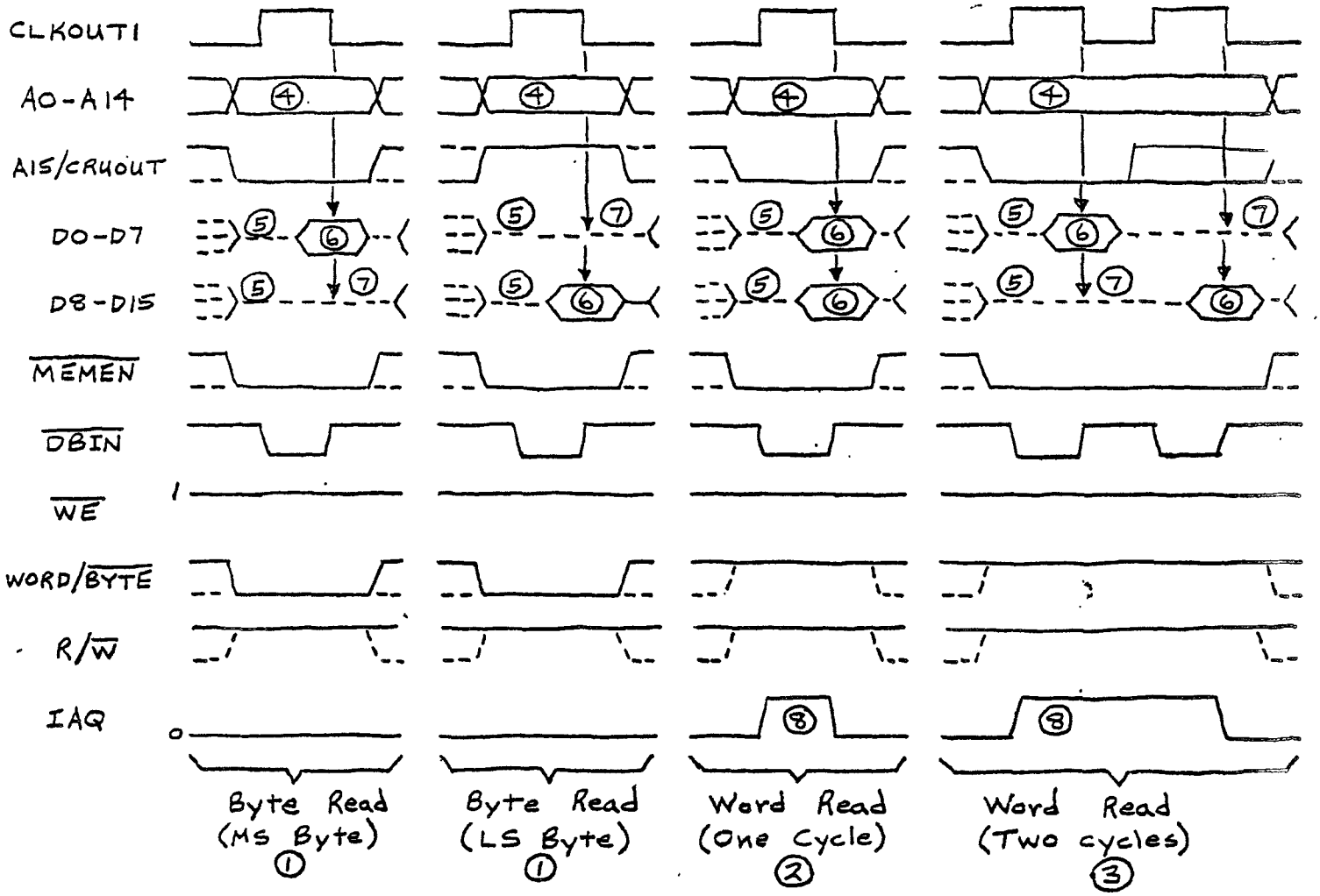
(b) Inhibiting Automatic First Wait State, TTL-Speed Reset Request



(c) Invoking Automatic First Wait State (Reset Can Be R-C Power-up Or Otherwise)

EXTERNAL CIRCUITRY FOR INVOKING/INHIBITING AUTOMATIC FIRST WAIT STATE GENERATION FEATURE

FIGURE 20.



Notes:

① 9995 SE/9990- is either connected to Vcc or Ground

② 9995 SE/9990- is connected to Ground or 9995 SE/9990- is connected to Vcc and Address of word being accessed is in the on-chip memory address space of the TMS 9995

③ 9995 SE/9990- is connected to Vcc and Address of word being accessed is in the off-chip memory address space of the TMS 9995

④ Valid Address

⑤ In input mode (Drivers @ High-Z)

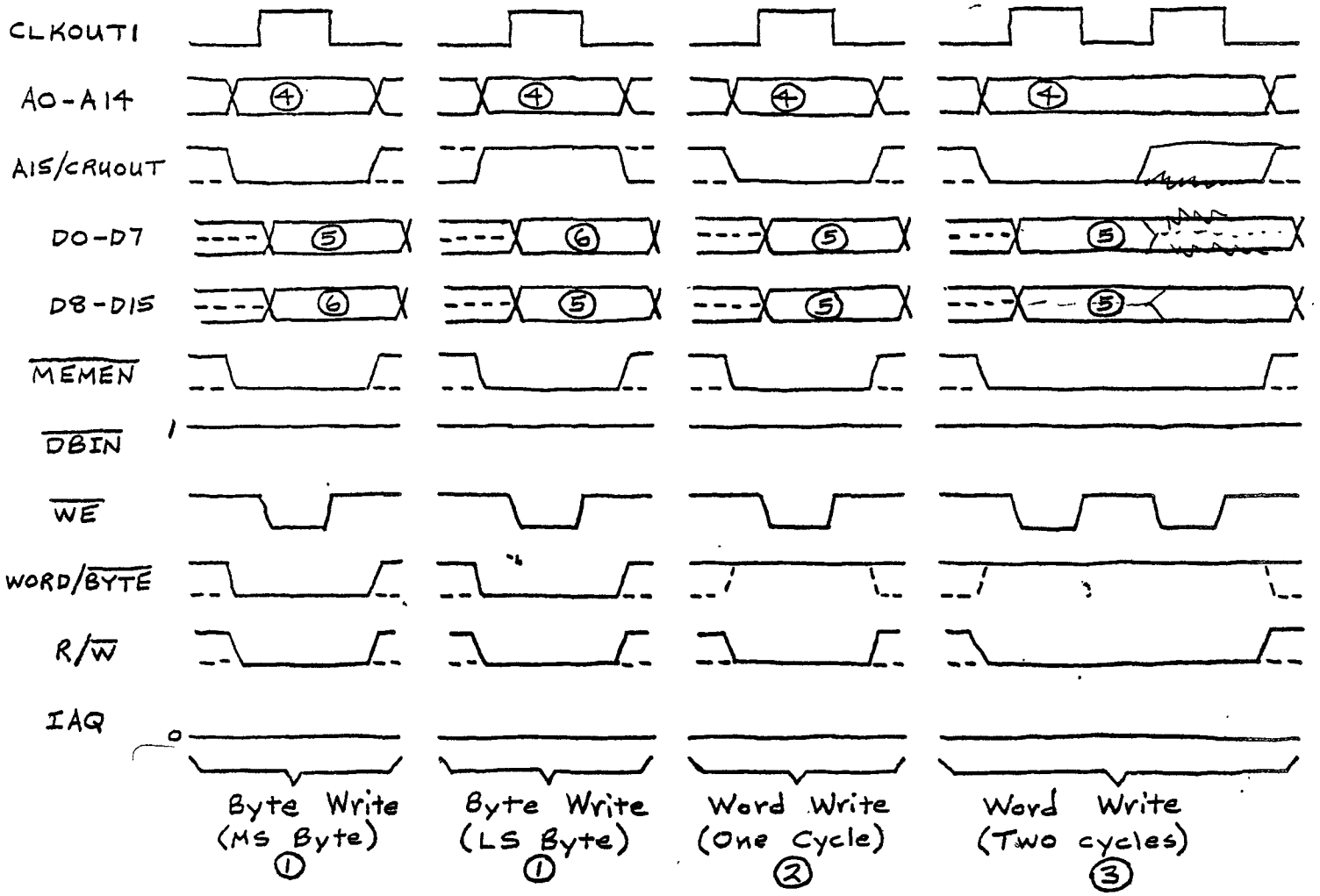
⑥ Addressed Byte of memory read data must be valid at CLKOUT1 edge indicated

⑦ This half of data bus is don't care inputs at CLKOUT1 edge indicated

⑧ IAQ will only be asserted if an instruction opcode is being read

9995 SE/9990- SIGNAL EFFECT ON MEMORY READ CYCLES

FIGURE 21.



Notes:

① 9995 SE/9990- is either connected to Vcc or Ground

② 9995 SE/9990- is connected to Ground or 9995 SE/9990- is connected to Vcc and Address of word being accessed is in the on-chip memory address space of the TMS 9995

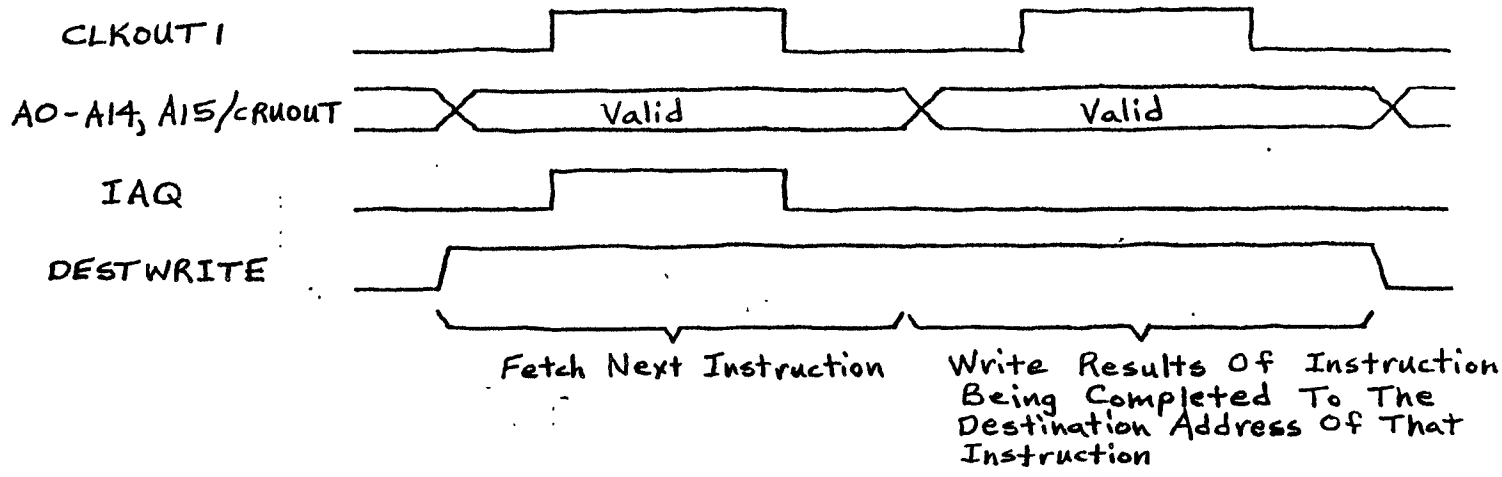
③ 9995 SE/9990- is connected to Vcc and Address of word being accessed is in the off-chip memory address space of the TMS 9995

④ Valid Address

⑤ Valid write data for addressed byte of memory address space

⑥ Non-specific data

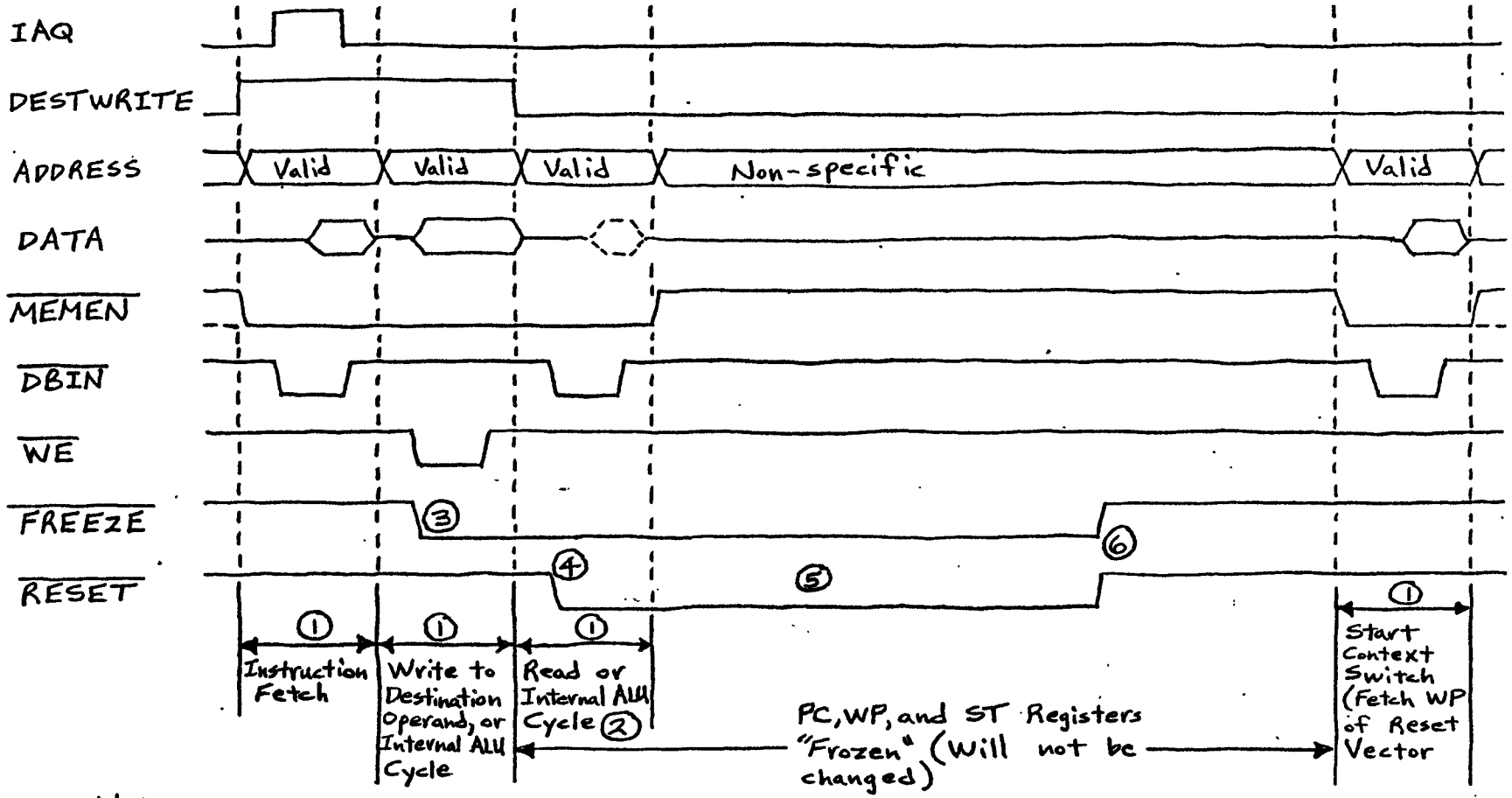
9995 SE/9990- SIGNAL EFFECT ON MEMORY WRITE CYCLES
FIGURE 22.



Note: Though shown as one CLKOUT1 cycle memory accesses, these accesses can each have one or more CLKOUT1 cycles and DESTWRITE will be high for as long as the two accesses require.

DESTWRITE SIGNAL CHARACTERISTICS

FIGURE 23.



Notes:

1. Memory accesses shown as full word accessed in one clock cycle, but could be two bytes requiring many clock cycles
2. The read cycle or Hold acknowledge that may occur following a valid FREEZE- must be inhibited by external emulator hardware
3. FREEZE- must be triggered low for every [(DESTWRITE=1)·(IAQ=0)·(CLKOUT1 ↑)]. If FREEZE- is taken high within 128 ns of the CLKOUT1 ↑ that triggered it low, the "freeze" will not occur. If FREEZE- is kept low and RESET- is asserted, "freeze" will occur as shown.
4. RESET- for a "freeze" is triggered low by [(DESTWRITE=0)·(FREEZE-=0)·(CLKOUT1 ↑)].
5. RESET- must be low for a minimum of four samples (RESET- and FREEZE- are sampled at every high-to-low CLKOUT1 transition). The "freeze" can be extended beyond the minimum shown by keeping both FREEZE- and RESET- low.
6. FREEZE- is triggered high by the same CLKOUT1 rising edge that triggers RESET- high.

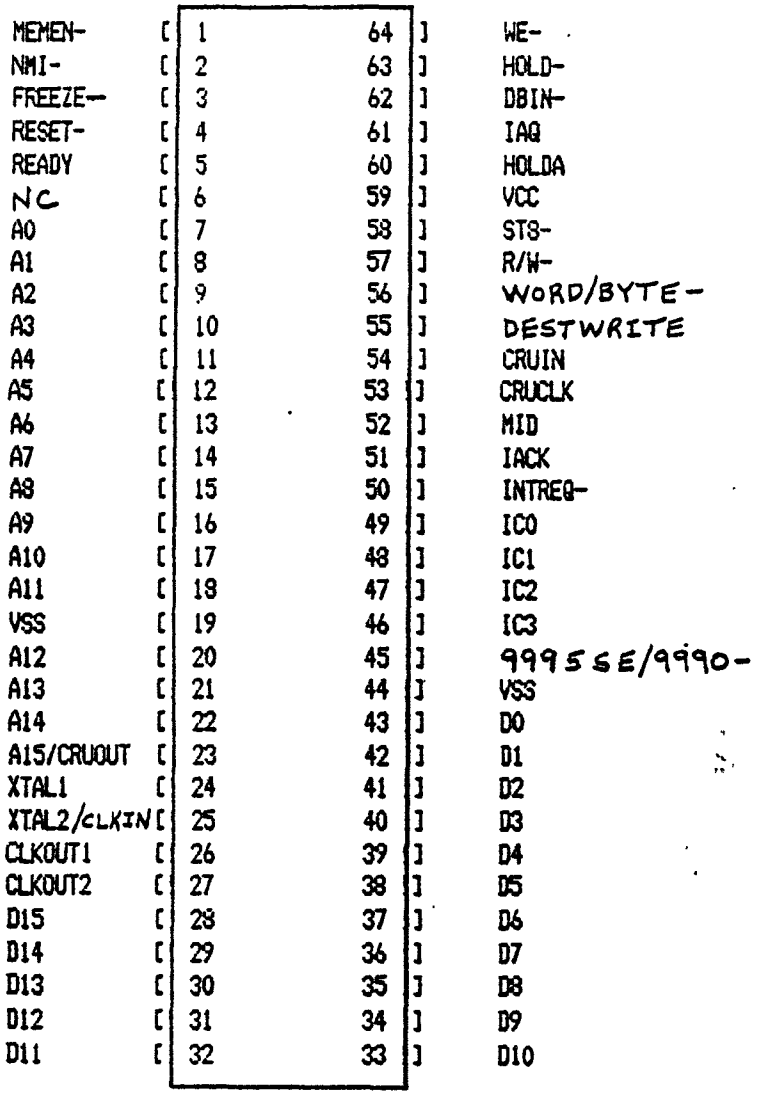
FREEZE SIGNAL CHARACTERISTICS
FIGURE 24.

Add: Register-to-Register

STEP	STATE COUNT	MEMORY CYCLE	INTERNAL FUNCTIO
1		FETCH INSTRUCTION	process previous opera
2	1	write results	DECODE INSTRUCTI
3	2	FETCH SOURCE	
4	3	FETCH DESTINATION	
5	4	fetch next instruction	ADD
6		WRITE RESULTS	decode instruction

EXECUTION SEQUENCE EXAMPLE

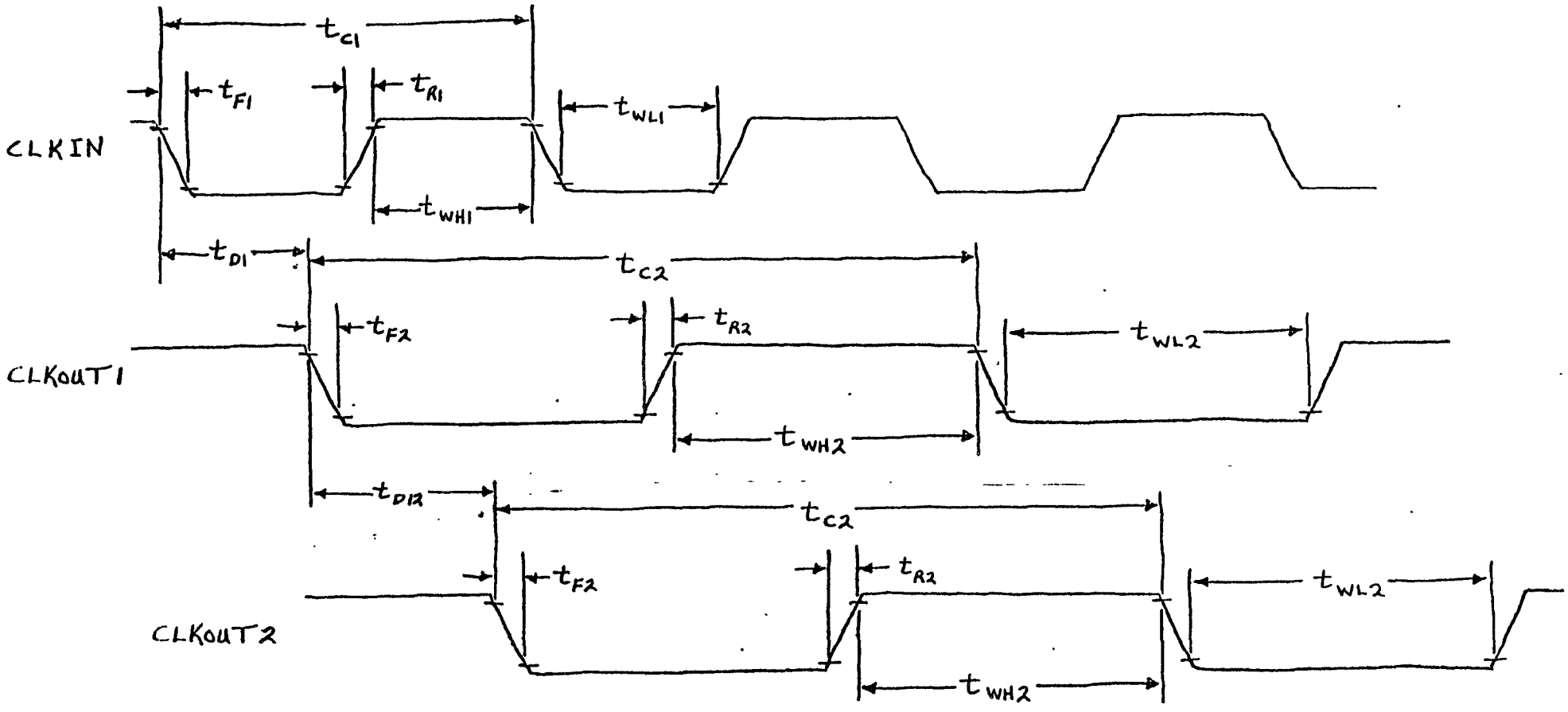
FIGURE 25.



NC - No Internal Connection

9995 SE PIN ASSIGNMENTS

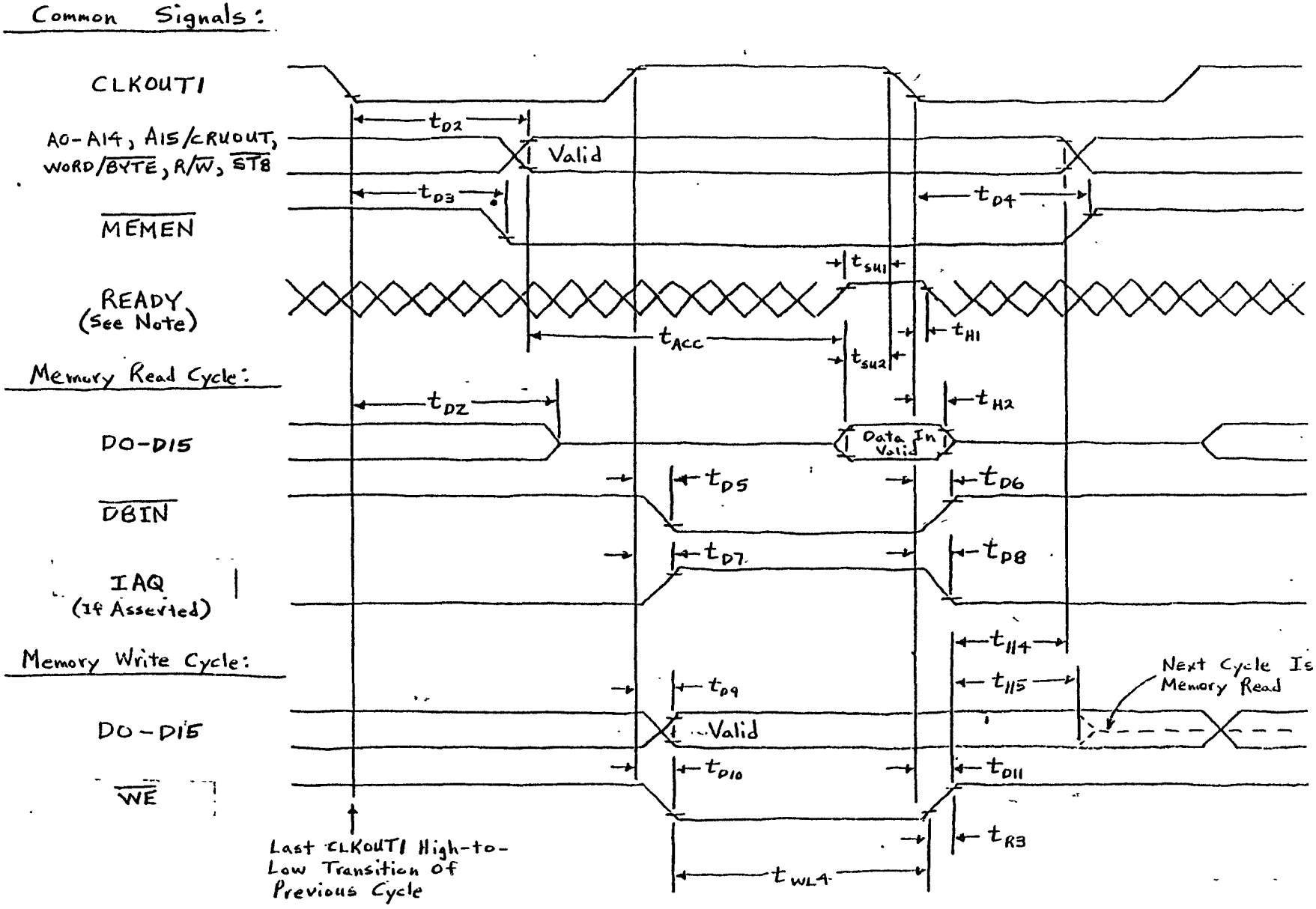
FIGURE 26.



Note: t_{C1}, t_{F1}, t_{R1}, t_{WH1}, t_{WL1}, and t_{D1} Become Undefined Parameters When A Crystal Is Connected Between XTAL1/CLKIN And XTAL2, And The Internal Oscillator Is Consequently Enabled

9995 SE CLOCK TIMING

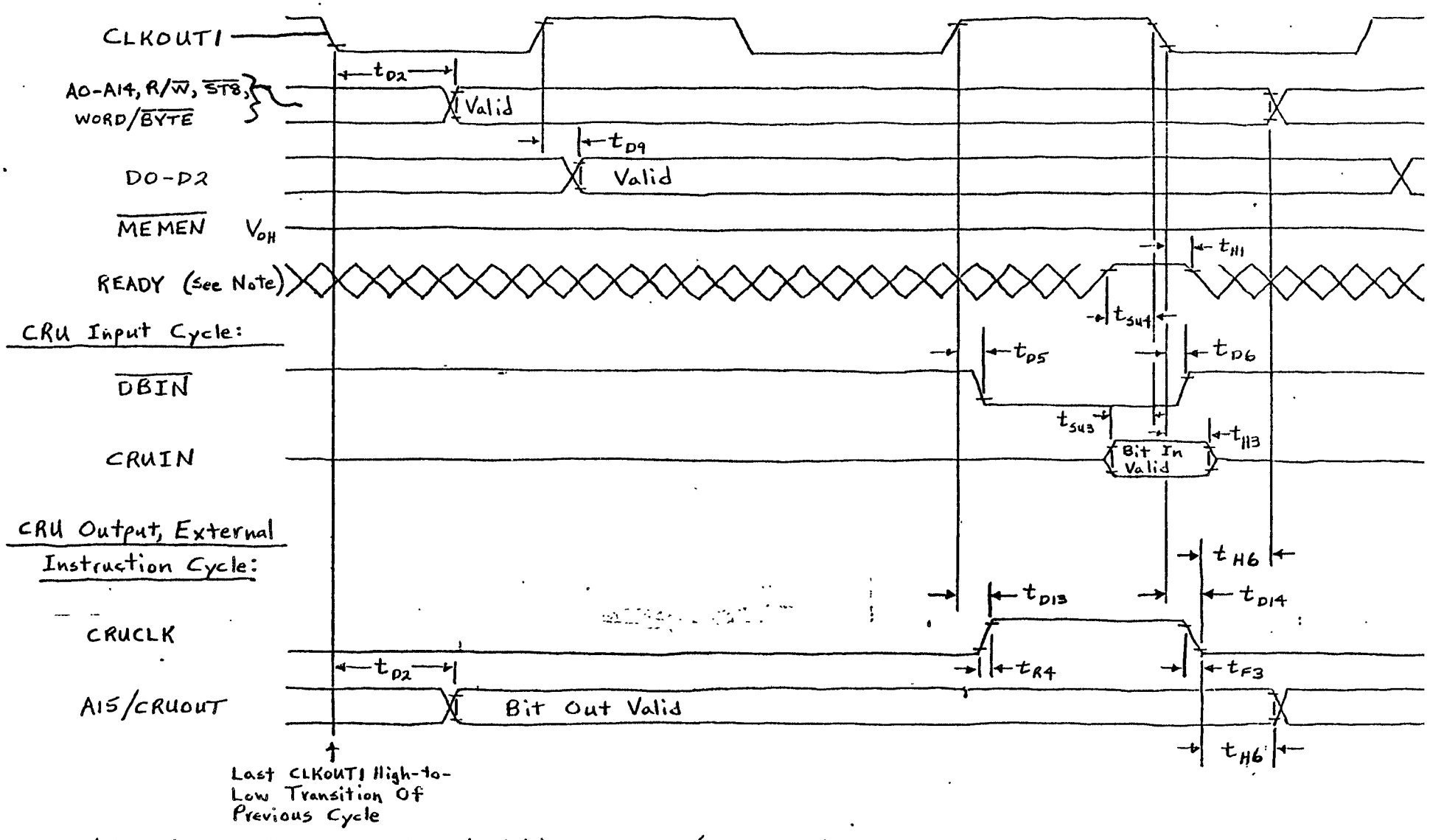
FIGURE 27.



Note: Cycle Shown Is For No Wait States (With Wait States, CLKOUT Cycles Get Added, But The Switching Parameters Do Not Change)

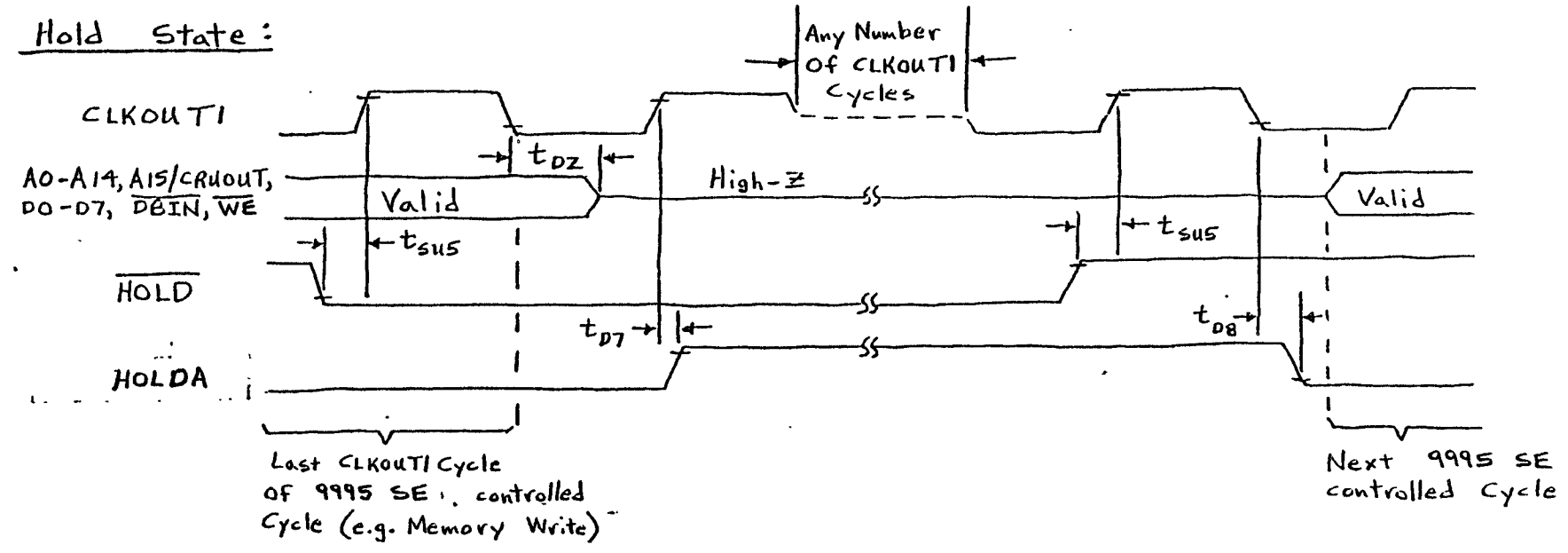
9995 SE

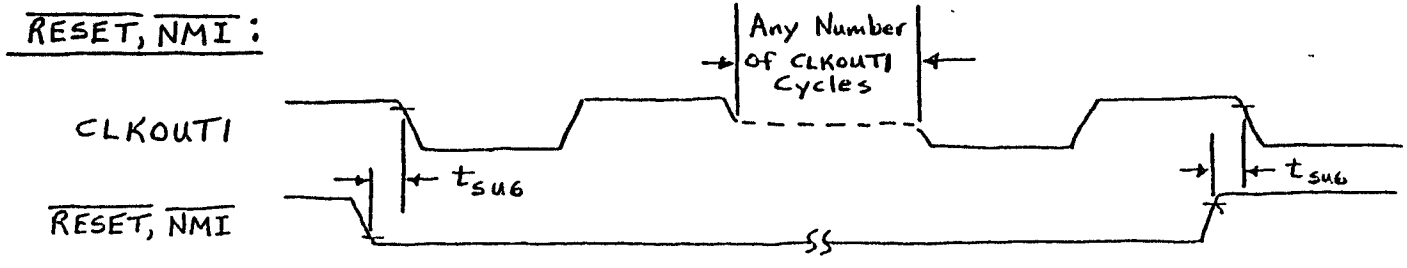
MEMORY INTERFACE TIMING
FIGURE 28.



Note: Cycle Shown Is For No Wait States (With Wait States, CLKOUT Cycles Get Added, But The Switching Parameters Do Not Change). Also, READY Is A Don't Care Input During External Instruction Cycles (Wait States Cannot Be Inserted Into External Instruction Cycles).

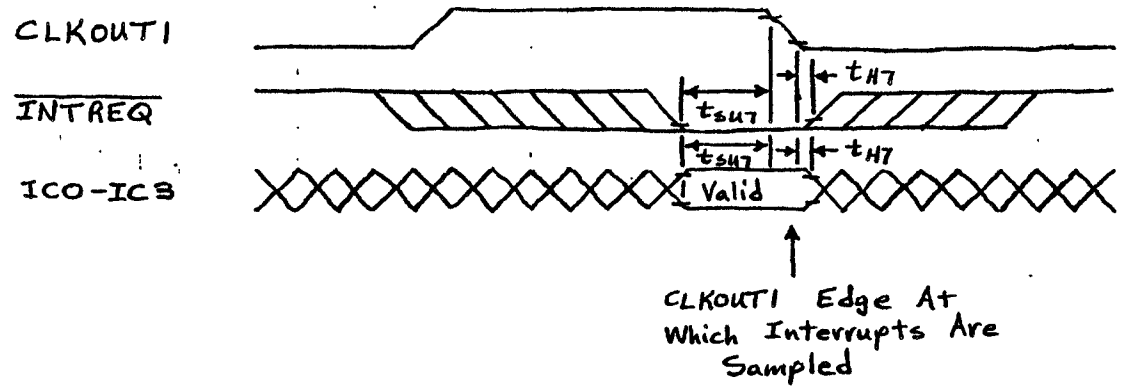
9995 SE. CRU, EXTERNAL INSTRUCTION TIMING
FIGURE 29





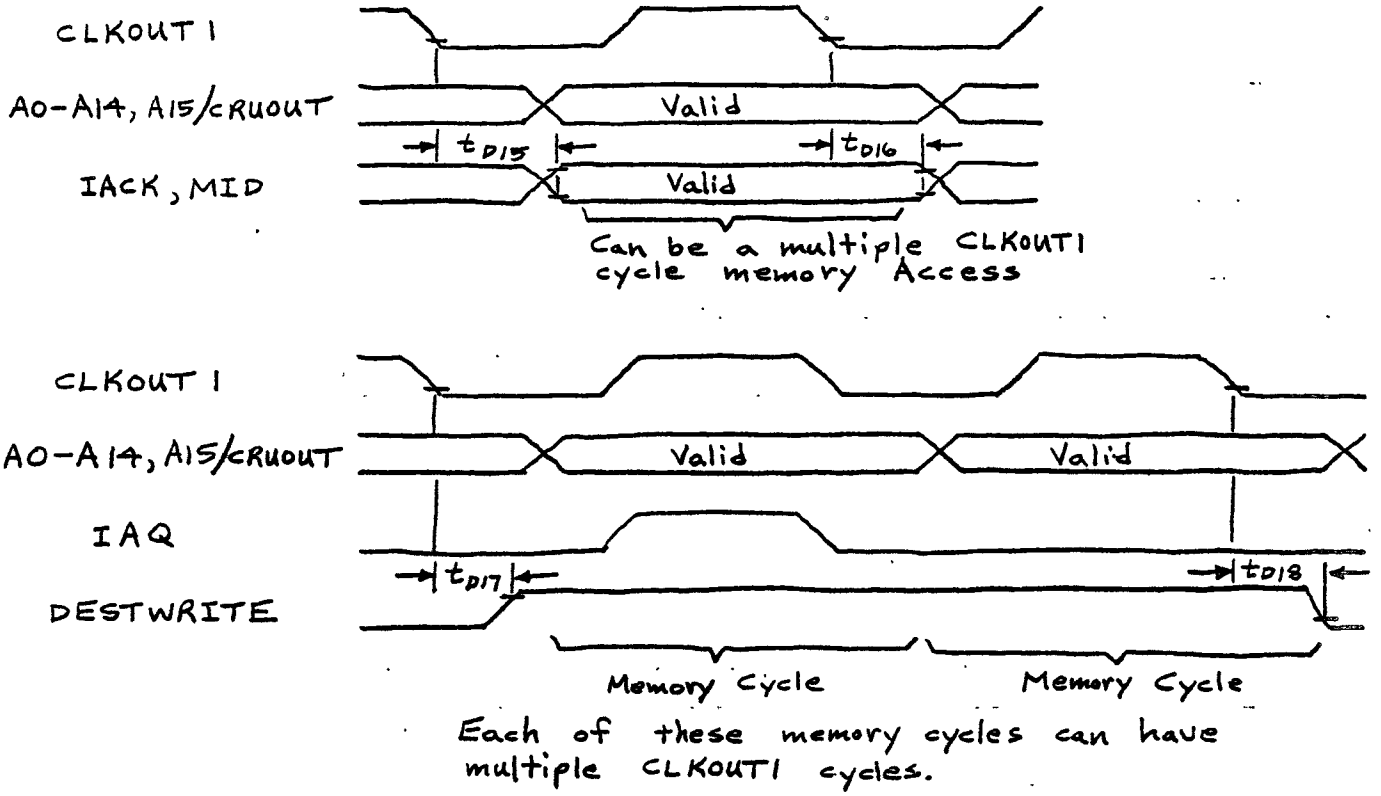
| 9995 SE HOLD, RESET, AND NMI TIMING

FIGURE 30.



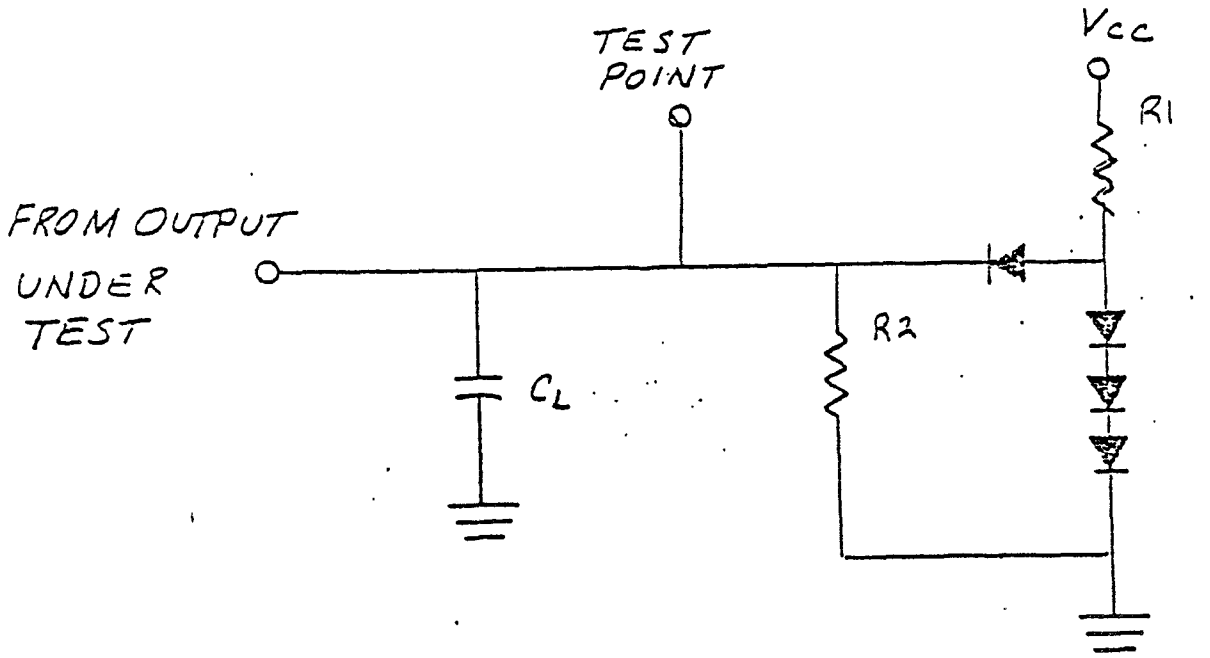
9995 SE INTREQ-, IC0-IC3 TIMING

FIGURE 31.



TIMING OF SPECIAL SIGNALS

FIGURE 32.

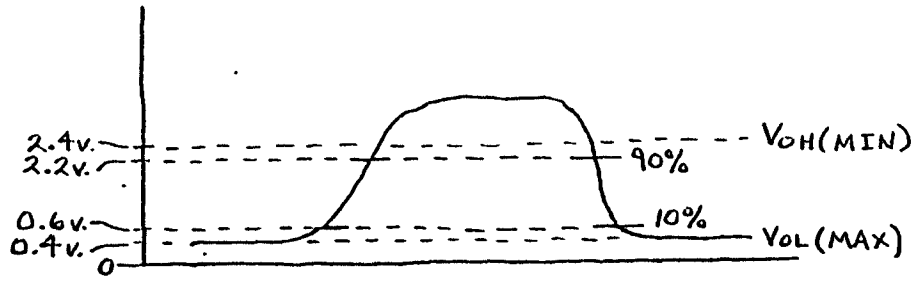


Notes: See Table II For Values Of C_L, R1, and R2.
All diodes are 1N916 or 1N3064

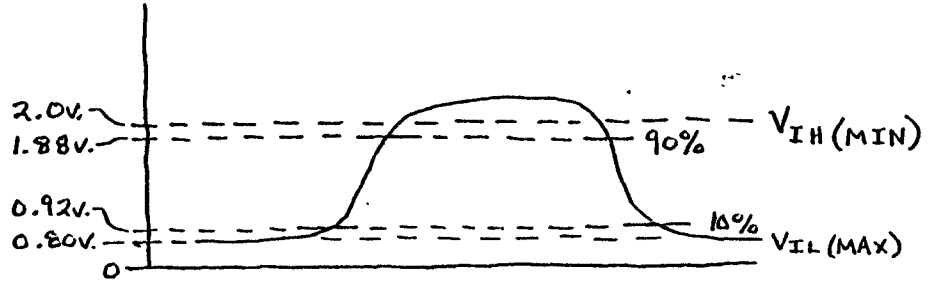
SWITCHING CHARACTERISTICS TEST LOAD CIRCUIT

FIGURE 33.

9995 SE Outputs:

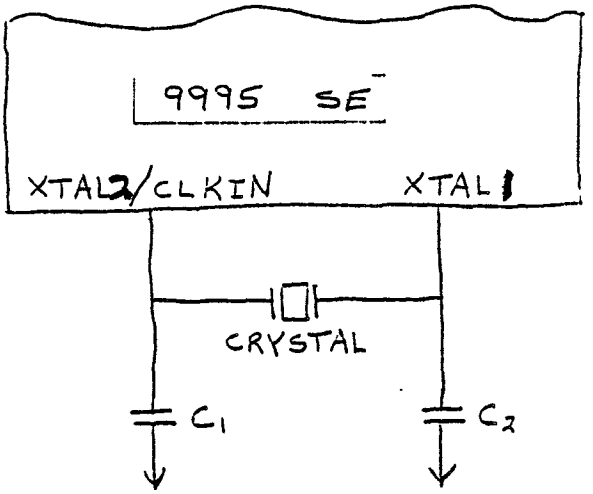


9995 SE Inputs:



MEASUREMENT POINTS FOR SWITCHING CHARACTERISTICS

FIGURE 34.

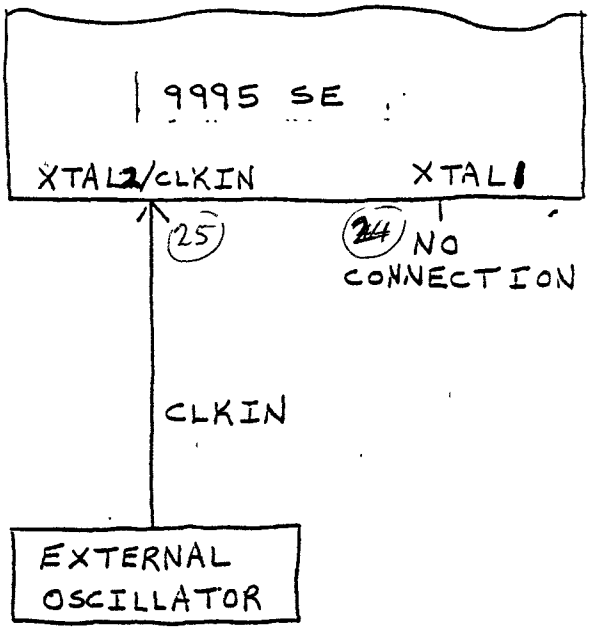


C_1 and C_2 represent the total capacitance on these pins including strays and parasitics.

PARAMETER	TEST CONDITIONS	MIN	NOM	MAX	UNI
Crystal Frequency, f_X	0°C - 70°C		6		MHz
C_1, C_2	0°C - 70°C	10	15	25	pF

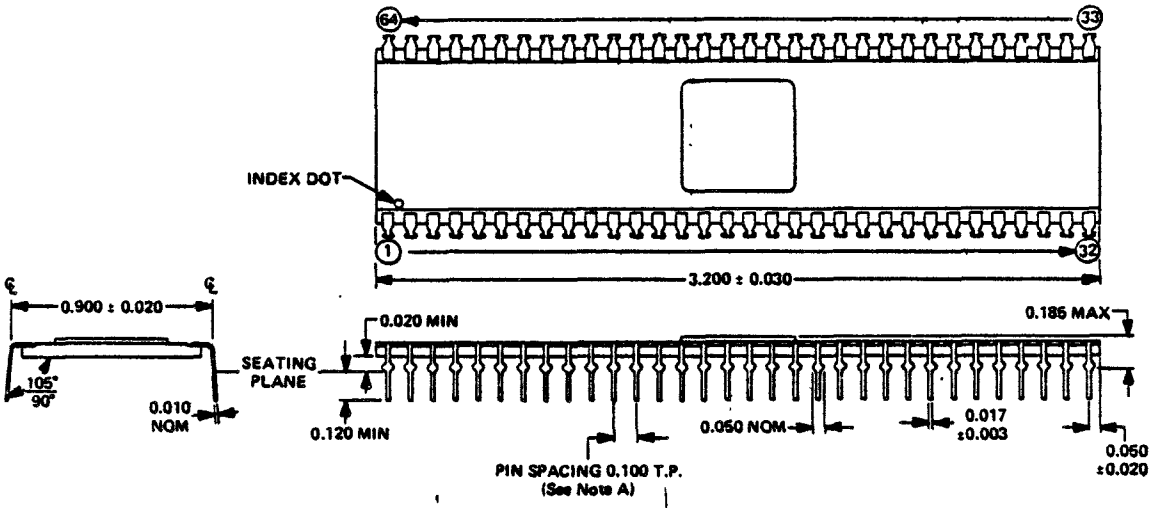
INTERNAL OSCILLATOR

FIGURE 35.



EXTERNAL OSCILLATOR

FIGURE 36.



NOTE A. Each pin centerline is located within 0.010 of its true longitudinal position.

9995 SE MECHANICAL DATA

FIGURE 37.