The Engineering Staff of
TEXAS INSTRUMENTS INCORPORATED
Semiconductor Group

SBP 9900 A SYSTEMS DEVELOPMENT MANUAL

AUGUST 1978

TEXAS INSTRUMENTS
INCORPORATED

The Engineering Staff of
TEXAS INSTRUMENTS INCORPORATED
Semiconductor Group

SBP 9900 A SYSTEMS DEVELOPMENT MANUAL

AUGUST 1978

PRELIMINARY RELEASE

(Review Copy)

TEXAS INSTRUMENTS
INCORPORATED

PREFACE

This *SBP 9900A System Development Manual* is intended to guide the design engineer in the specification, design, and implementation of an SBP 9900A based microprocessor system. This manual describes the SBP 9900A's general system level operation; and contains interfacing details, suggested hardware configurations, power source considerations, and electrical requirements. The reader is assumed to be familiar with the SBP 9900A instruction set and architecture as described in: 1) section three of the *Bipolar Microcomputer Components Data Book*, [as supplemented by the information contained in Appendix A of this manual] and 2) the *Model 990 Computer, Assembly Language Programmer's Guide*. Also, the information contained in the *990 Computer Family System Handbook* will be useful as a supplement to the systems description provided herein. These documents are available from your Texas Instruments sales representative.

IMPORTANT NOTICES

Texas Instruments reserves the right to make changes at any time in order to improve design and to supply the best product possible.

TI cannot assume any responsibility for any circuits shown or represent that they are free from patent infringement.

TEXAS INSTRUMENTS
INCORPORATED
Semiconductor Group
Post Office Box 1443 Houston, Texas 77001

TABLE OF CONTENTS

Paragraph **Title** **Page**

SECTION I. INTRODUCTION

SECTION II. SBP 9900A ARCHITECTURE

SECTION III. MEMORY

TABLE OF CONTENTS (Continued)

Paragraph Title Page

SECTION IV. INTERRUPTS

SECTION V. INPUT/OUTPUT

SECTION VI. AUXILIARY SYSTEM FUNCTIONS

TABLE OF CONTENTS (Continued)

SECTION VII. POWER SOURCE CONSIDERATIONS

SECTION VIII. ELECTRICAL REQUIREMENTS

APPENDIXES

LIST OF ILLUSTRATIONS

Figure Title Page

LIST OF ILLUSTRATIONS (Continued)

LIST OF TABLES

SECTION I
INTRODUCTION

1.1 GENERAL DESCRIPTION

The SBP 9900A microprocessor, architecturally and functionally shown in Figure 1-1, 1-2, and 1-3, is a ruggedized monolithic parallel 16-bit Central Processing Unit (CPU) fabricated with oxide separated Integrated Injection Logic (I^2L) technology. The SBP 9900A combines the properties of I^2L technology with an advanced memory-to-memory achitecture, the flexibility of word/byte/bit data handling, and the versatility of a full 16-bit minicomputer instruction set, to extend the end application reach of Texas Instruments 9900 series microprocessor family into those applications requiring highly reliable, stable performance under severe operating conditions including nuclear radiation environments. I^2L technology enables the SBP 9900A to operate over a very wide ambient temperature range from a single d-c current source. Static logic is used throughout with TTL compatible I/O. This allows use with standard logic/memory devices and thereby eliminates the need for specialized clock and interface functions. SBP 9900A software is directly compatible with other 9900 series microprocessor family members and is upward compatible with Texas Instruments 990 series minicomputer family.

1.2 KEY FEATURES

- Parallel 16-Bit Word Length
- Full Minicomputer Instruction Set Includes Multiply and Divide
- Directly Addresses Up to 65,536 Bytes/32,768 Words of Memory
- Advanced Memory-To-Memory Architecture
- Multiple 16-Word Register Files (Work Spaces) Reside in Memory
- Separate I/O, Memory, and Interrupt Bus Structures
- 16-Prioritized Hardware Interrupts
- 16 Software Interrupts (XOPS)
- Programmed, and DMA, I/O Capability
- 64-Pin Package
- Software Compatible with TI 9900 Microprocessor/990 Minicomputer Family

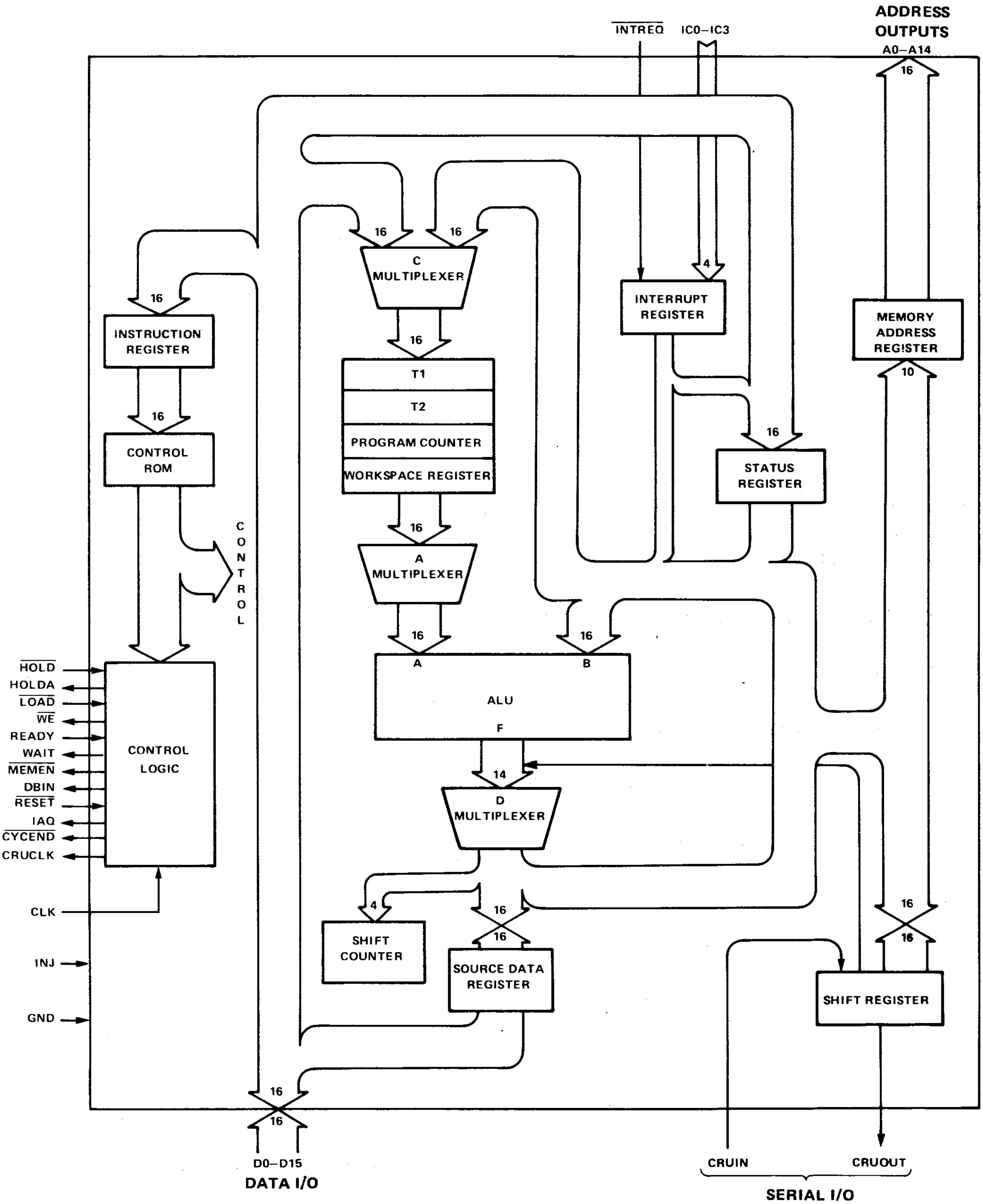


Figure 1-1. SBP 9900A Architecture

- I²L Technology:
  - DC to 3 MHz Guaranteed Clock Frequency Range
  - Single d-c Power Supply
  - Fully Static Operation
  - Single Phase, Edge-Triggering Clock
  - Directly TTL Compatible I/O (Including Clock)
  - Wide Ambient Temperature Operation:
    SBP 9900 ACJ: 0°C to +70°C
    SBP 9900 AEJ: −40°C to +85°C
    SBP 9900 ANJ: −55°C to +125°C
    SBP 9900 AMJ: −55°C to +125°C (High Reliability Processing).

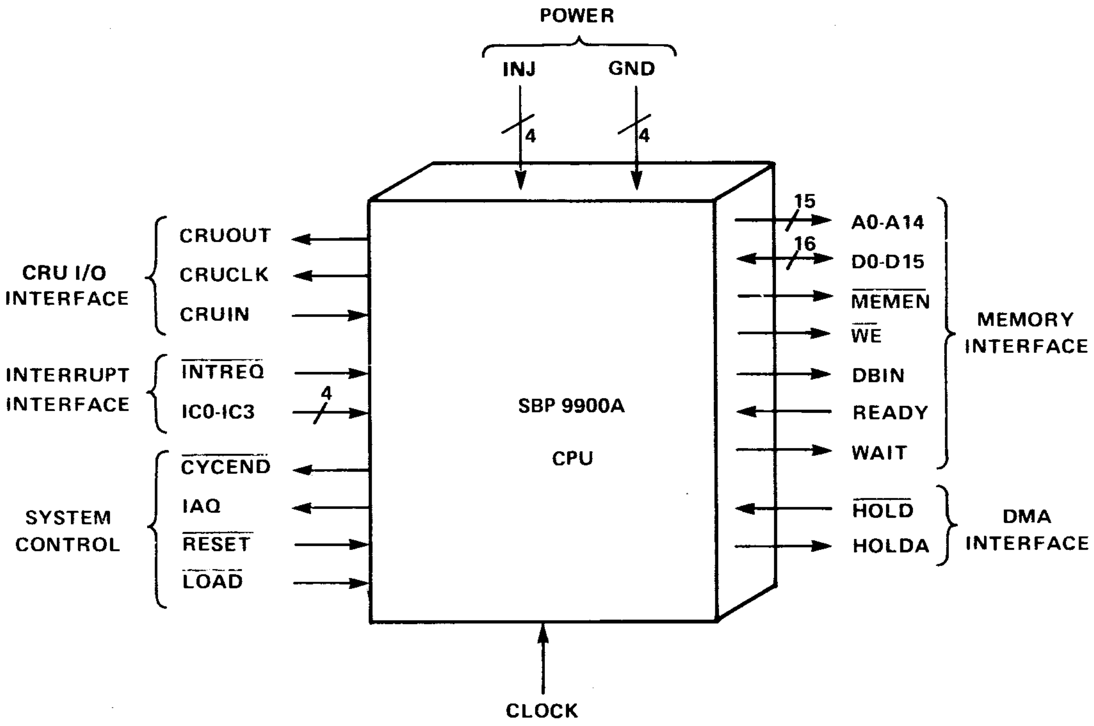


Figure 1-2. SBP 9900A Functional Description

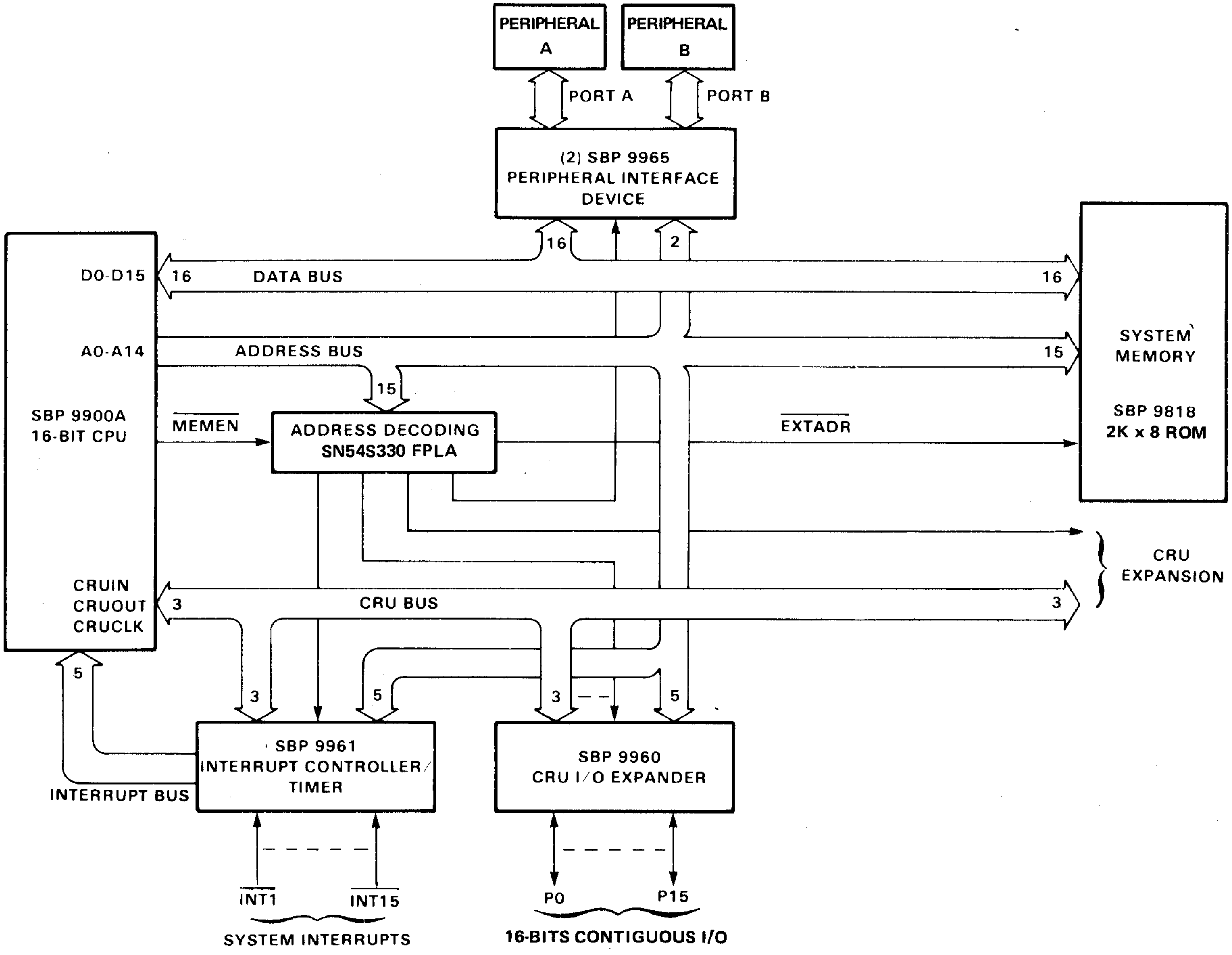


Figure 1-3. SBP 9900A System

SECTION II
SBP 9900A ARCHITECTURE

The SBP 9900A employs an advanced memory-to-memory architecture, as shown in Figure 2-1, which replaces the concept of a hardware register file internal to the CPU, with the concept of multiple register files external to the CPU. Each individual external register file is referred to as a ''workspace'' and is implemented as 16 contiguous system memory words. Each word may function as a general register holding either data, an address, an index value, or accumulated results. Therefore, the SBP 9900A commands the flexibility of multiple 16-word general-register workspaces limited in number only by the depth of the system memory itself.

2.1 REGISTERS

In support of an *active* SBP 9900A program environment, 19 registers (shown in Figure 2-2) which are accessible to the user, include those 16 workspace-register memory words which constitute the active workspace plus three hardware registers internal to the SBP 9900A: the workspace-pointer (WP) register, the program-counter (PC) register, and the status (ST) register.

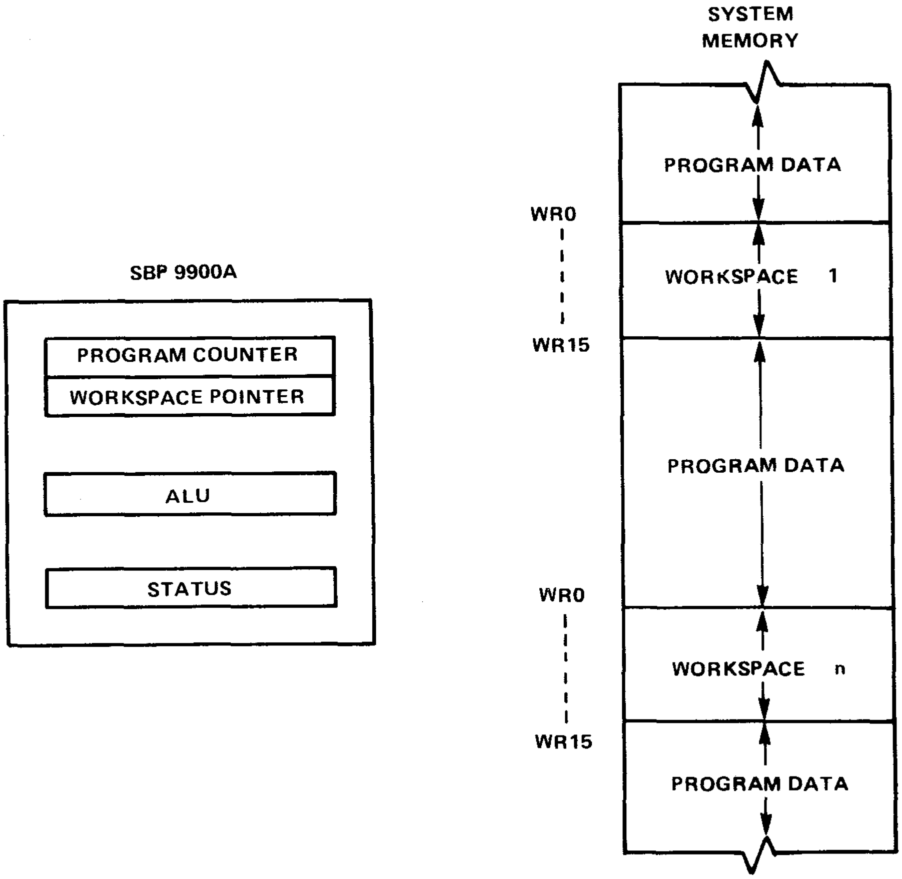


Figure 2-1. SBP 9900A Memory-to-Memory Architecture

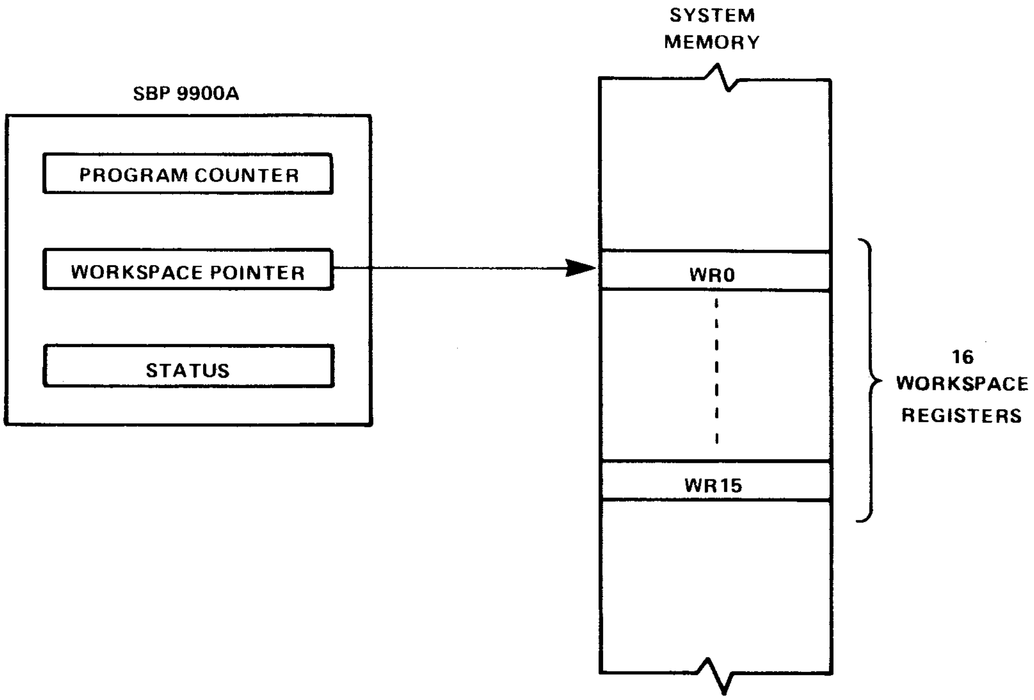


Figure 2-2. Registers Supporting Immediate SBP 9900A Program

2.1.1 WORKSPACE-POINTER (WP) REGISTER. The workspace-pointer (WP) register is a 16-bit register that defines an *active* workspace by specifying the memory address (shown in Table 2-1) of the first of 16 contiguous memory word locations which together form the workspace. The SBP 9900A automatically calculates the memory address of a specific workspace-register (WR), within the active workspace, by adding the WP value to the specified WR number weighted by a factor of two:

$$MA = WP \text{ plus } 2R \text{ where } R = 0 \text{ through } F_{16}.$$

When either a different or extended set of workspace-registers is required, the program simply activates a new workspace by loading the WP with the initial memory word address of that new workspace. Through proper selection of the new WP value, the new workspace may either overlap* or be exclusive from the old workspace. This highly efficient procedure for switching from one workspace to another results in a significant reduction in system overhead when compared to internal-register-file based microprocessors which must perform "n" memory-write operations to save the contents of "n" internal file registers; then perform "n" memory-read operations to redefine the contents of "n" internal file registers.

*When overlapping workspaces, caution must be exercised such that when utilizing those registers common to each workspace, the individual functional definitions of each register (see 2.1.2) do not conflict.

Table 2-1. SBP 9900A Workspace Pointer and Workspace Register

			REGISTER USE		
	MEMORY ADDRESS	REGISTER			
WORKSPACE POINTER	WP + 00	0	DATA OR ADDRESSES		OPTIONAL SHIFT COUN
	WP + 02	1		INDEX CAPABILITY	
	WP + 04	2			
	WP + 06	3			
	WP + 08	4			
	WP + 0A	5			
	WP + 0C	6			
	WP + 0E	7			
	WP + 10	8			
	WP + 12	9			
	WP + 14	10			
	WP + 16	11			– BL RETURN ADDRES
	WP + 18	12			– CRU BASE ADDRESS
	WP + 1A	13			– SAVED WP
	WP + 1C	14			– SAVED PC
	WP + 1E	15			– SAVED ST

2.1.2 WORKSPACE-REGISTERS. A workspace is comprised of 16 contiguous 16-bit memory words (shown in Table 2-1) which function as general workspace-registers to store temporary data, addresses, or accumulated results. Additionally, bits 12 through 15 of WRO may specify the number of bit positions an operand is to be shifted by the instructions SLA, SRA, SRC, and SRL. WR1 through WR15 may each be used as an index register to specify a bias from the contents of memory word location PC plus 2 in selection of an operand located anywhere in general memory. WR11 will store the return PC value when the Branch and Link (BL) instruction is executed. Bits 3 through 14 of WR12 contain the 12-bit CRU base address used in conjunction with the Communications Register Unit (CRU) instructions SBO, SBZ, TB, LDCR, and STCR. WR13 through WR15 of a *new* workspace, activated under program control through execution of either the Branch and Load Workspace Pointer (BLWP) instruction or an Extended Operation (XOP) instruction, or activated under hardware control in response to either an interrupt or the LOAD/RESET signal commands, will store the *old* WP, PC, and ST values for preservation of return linkage.

2.1.3 PROGRAM-COUNTER (PC) REGISTER. The program-counter (PC) register is a 16-bit register similar in function to that of other microprocessors. It contains the memory address of the next instruction to be executed. As each instruction is in turn executed, the PC is either incremented by 2 to the next consecutive memory word address or altered directly by the executing instruction.

2.1.4 STATUS REGISTER (ST). The status register (ST) is a 16-bit register similar in function to that of other microprocessors. It is composed of flag-bits which indicate the results of the most recent arithmetic or Boolean operation performed. Additionally, bits ST12 through ST15 contain a 4-bit interrupt mask which defines the lowest priority interrupt that will be recognized. This interrupt mask may be altered under program control through execution of the Load Interrupt Mask Immediate (LIMI) instruction, or automatically altered under hardware control in response to acceptance of an interrupt.

2.2 MEMORY-TO-MEMORY OPERATIONS

The instructions of the SBP 9900A are not limited to workspace-registers as operand sources or result destinations. Any of the general-source/general-destination instructions permit operations to be performed on any general memory location by any other general memory location. In other words, a single two-address instruction has the capability to:

a. fetch an operand from any general memory location,

b. fetch a second operand from another general memory location,

c. combine the operands, and

d. store the results in that general memory location from which the second operand was fetched.

For example, a single Add (A) instruction can add any memory word to any other memory word within the 32K word memory space. To accomplish this same operation, the architectures of other microprocessors require execution of a series of instructions through which memory contents are moved to internal registers, added, and then returned to memory. Consequently, the memory-to-memory architecture of the SBP 9900A supports lower memory cost plus lower system development cost by:

a. minimizing program storage requirements through reduction of the number of instructions required to accomplish execution of a given task,

b. reducing program complexity,

c. reducing program documentation, and

d. reducing program debug time.

2.3 BUS STRUCTURES.

The SBP 9900A employs separate memory, CRU, and interrupt bus structures as shown in Figure 2-3. Each bus is optimized for its individual function.

2.3.1 MEMORY BUS. The memory bus handles uniform width memory words which are transferred in parallel between memory or memory-mapped peripheral I/O devices (i.e., SBP 9965). The memory reference instructions operate on parallel words or bytes, with additional masking instructions available to isolate individual bits. The SBP 9900A maintains separate address outputs, control I/O, and data I/O to optimize communication with standard memories; the SBP 9900A employs $\overline{\text{HOLD}}$ and HOLDA handshaking signals, similar in function to other microprocessors, to aid system implementation of DMA capability. The non-multiplexed memory bus structure of the SBP 9900A simplifies board layout and lowers system cost by precluding the need for external bus demultiplexing devices. A complete discussion of memory interface to the SBP 9900A is contained in Section 3.0.

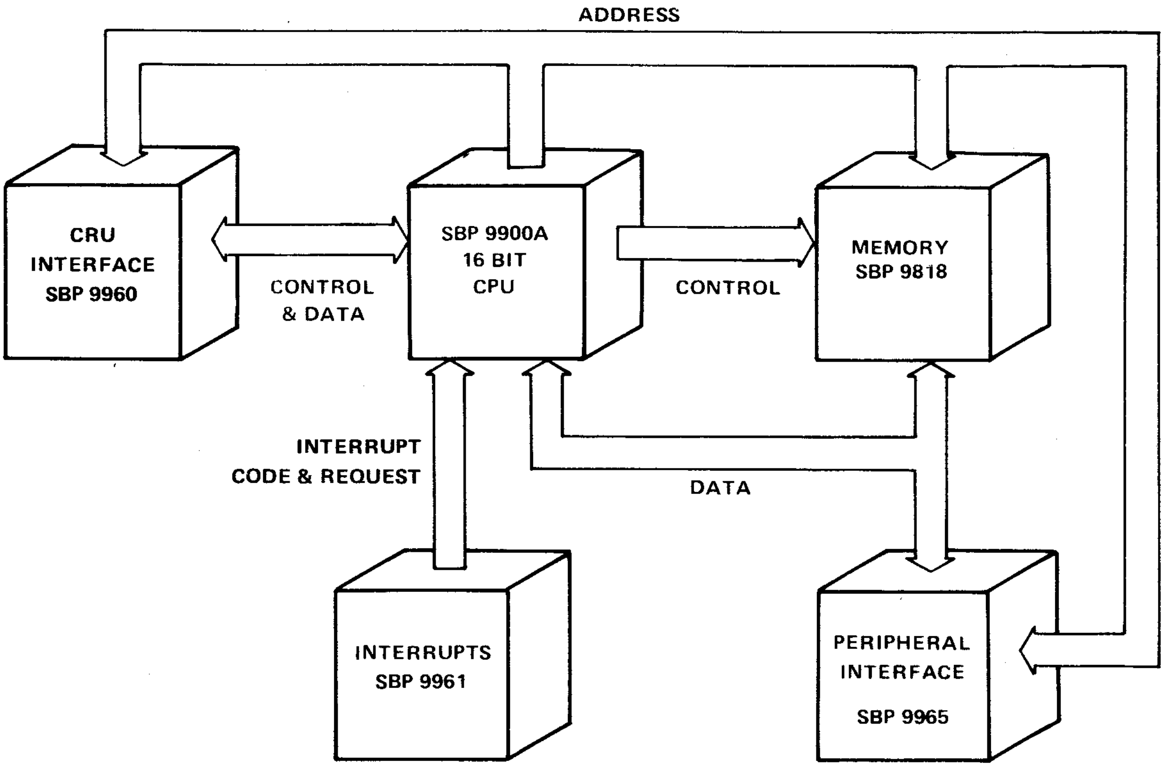


Figure 2-3. SBP 9900A System Bus Structure

2.3.2 CRU BUS. The CRU bus handles variable bit-length data each bit of which is addressed in parallel and transferred serially between the SBP 9900A and Communications Register Unit (CRU) interface devices such as the SBP 9960 and SBP 9961. Those interface devices in turn communicate with auxillary systems devices ranging from bit-oriented sensors and actuators to word/byte/n-bit-field oriented peripherals. Therefore, the Communications Register Unit (CRU) is a cost-effective flexible means by which the SBP 9900A may be interfaced to peripheral I/O devices independent of the memory data bus. As a result, the necessity to reserve a memory address for each I/O device is eliminated. For I/O device interfacing, the SBP 9900A supports CRU expansion up to 4096 individually addressable input bits and up to 4096 individually addressable output bits. A single CRU instruction can operate on an individual CRU bit or on variable CRU bit-fields ranging anywhere from one to 16 bits in length. A complete discussion of the operation, implementation, and flexibility of the CRU is contained in Section 5.0.

2.3.3 INTERRUPT BUS. The SBP 9900A handles multiple prioritized interrupts with internal generation of associated trap vectors. The interrupt bus provides the SBP 9900A with an interrupt request plus a 4-bit code identifying the priority level of the interrupt requesting service. A complete discussion of SBP 9900A interrupt processing is contained in Section 4.0.

2.4 CONTEXT SWITCHING

An operation referred to as a ''context switch'' is automatically performed by the SBP 9900A in response to:

a. recognition of an interrupt,

b. assertion of a $\overline{\text{LOAD}}$ or $\overline{\text{RESET}}$ signal command, or

c. execution of an XOP or BLWP instruction.

The purpose of a context switch is to expeditiously perform a hardware branch from the present (old) program environment to a new program environment with preservation of return linkage to the original (old) program environment. The hardware effect of a context switch is to initiate new WP and PC values, yet preserve the original (old) WP, PC, and ST values for return linkage. Dedicated memory locations described in Section 3.1, contain the new WP and PC values associated with:

a. each individual interrupt priority level,

b. the $\overline{\text{LOAD}}$ or $\overline{\text{RESET}}$ signal command, or

c. each individual XOP instruction.

The new WP and PC values associated with the BLWP instruction are contained in consecutive memory word locations defined through the active addressing mode of the BLWP instruction under execution. The sequence of hardware events which are accomplished during a context switch are:

a. the new WP value is fetched from memory and loaded into the SBP 9900A's temporary T_1 register,

b. the present (old) ST value is preserved in WR15 of the new workspace defined by T_1,

c. the present (old) PC value is preserved in WR14 of the new workspace defined by T_1,

d. the present (old) WP value is preserved in WR13 of the new workspace defined by T_1,

e. the contents of T_1 (the new WP value) is transferred into the SBP 9900A's WP register,

f. the new PC value is fetched from memory and loaded into the SBP 9900A's PC register,

g. the first instruction of the new program environment is fetched from that location addressed by the PC (new PC value) register, and

h. instruction execution begins in the new program environment.

When the new program executes a Return Workspace Pointer (RTWP) instruction, the original (old) WP, PC, and ST values (preserved in WR13, 14, and 15 respectively of the new workspace) are automatically restored to the SBP 9900A thereby reactivating the old workspace and the original (old) program environment. Consequently, the workspace-oriented memory-to-memory architecture of the SBP 9900A in conjunction with its automatic context switching flexibility, is the basic hardware mechanism which allows the SBP 9900A to expeditiously handle unlimited nesting of interrupts and/or subroutines.

2.5 MACHINE CYCLES

Each operation performed by the SBP 9900A consists of a sequence of machine cycles. In each machine cycle the SBP 9900A performs a data transfer with memory or the CRU, and/or an internal arithmetic or Boolean operation.

2.5.1 ALU MACHINE CYCLES. Each ALU machine cycle is two clock periods in length. In an ALU cycle, the ALU performs an arithmetic or Boolean operation on two operands contained internally. No external data transfers occur during an ALU cycle.

2.5.2 MEMORY READ MACHINE CYCLES. The function of the memory read cycle is to transfer a word of data contained in memory to the SBP 9900A. An ALU operation can be performed during a memory read cycle. Memory read cycles are a minimum of two clock periods in length. If wait states are inserted to allow access to slow memories, the length of the memory read cycle is extended one clock period for each wait state.

IAQ is active (high) during any CPU initiated instruction acquisition memory read cycle. IAQ is held active for the duration of the entire memory read cycle.

2.5.3 MEMORY WRITE MACHINE CYCLES. The memory write cycle is identical to the memory read cycle, except that data is written to rather than read from memory.

2.5.4 CRU OUTPUT MACHINE CYCLES. Each CRU output machine cycle is two clock periods in length. In addition to outputting a bit of CRU data, an ALU operation may also be performed.

2.5.5 CRU INPUT MACHINE CYCLES. The CRU input machine cycle is identical to the CRU output cycle, except that one bit of CRU data is input rather than output.

2.5.6 INSTRUCTION EXECUTION EXAMPLES. Examples of how sequences of machine cycles are used to execute instructions are shown in Figures 2-4 to 2-7. For additional information on instruction machine cycles refer to Appendix B. Note that the first machine cycle of each instruction is always an instruction fetch memory read cycle; the second is always an ALU cycle.

SBO

CYCLE	TYPE	FUNCTION
1	Memory Read	Instruction Fetch
2	ALU	Decode Op Code
3	ALU	Calculate Address of WR12
4	Memory Read	Fetch (WR12)
5	ALU	Calculate CRU Address
6	CRU Output	Output Bit, Increment PC

Figure 2-4. **SBO Instruction Machine Cycles**

STCR RO,5

CYCLE	TYPE	FUNCTION
1	Memory Read	Instruction Fetch
2	ALU	Decode Op Code
3	Memory Read	Fetch (WRO)
4	ALU	Cal^ulate Address of WR12
5	Memory Read	Fetch (WR12)
6	ALU	Set Up
7	ALU	Set Up
8-12	CRU Input	Transfer 5 Bits
13	ALU	Set Up
14	ALU	Set Up
15-17	ALU	Zero Filling
18	ALU	Parity Generation
19	ALU	Load WRO Address in MA
20	ALU	Byte Positioning
21	Memory Write	Store Data in WRO, Increment PC

Figure 2-5. **STCR Instruction Machine Cycles**

RTWP

CYCLE	TYPE	FUNCTION
1	Memory Read	Instruction Fetch
2	ALU	Decode Opcode
3	ALU	Calculate Address of WR15
4	Memory Read	Restore Status from WR15
5	Memory Read	Restore PC from WR14
6	Memory Read	Restore WP from WR13
7	ALU	Load PC into MA

Figure 2-6. RTWP Instruction Machine Cycles

A *R1,R2

CYCLE	TYPE	FUNCTION
1	Memory Read	Instruction Fetch
2	ALU	Decode Opcode
3	Memory Read	Fetch (WR1)
4	ALU	Set Up
5	Memory Read	Fetch ((WR1))
6	ALU	Set Up
7	Memory Read	Fetch (WR2)
8	ALU	Addition
9	Memory Write	Store Result in WR2, Increment PC

Figure 2-7. 'A' Instruction Machine Cycles

2.6 MACHINE CYCLE MIN/MAX LIMITS; MAXIMUM LATENCY TO ENTER THE HOLD STATE

Machine cycle minimum/maximum limits, useful information for system design, are listed in Table 2-2. The SBP 9900A can enter the hold state only during an ALU, CRU input, CRU output, or combination CRU/ALU machine cycle. Consequently, the maximum number of back to-back memory cycles determines the maximum machine cycle latency for the SBP 9900A to enter the hold state. Table 2-2 shows the maximum number of back-to-back memory cycles to be three consecutive memory read cycles. As a result, the maximum latency for the SBP 9900A to enter the hold state is seven clocks: MAXIMUM HOLD LATENCY = setup (1 clock) plus three consecutive memory read cycles (6 clocks assuming no wait states).

TABLE 2-2. MACHINE CYCLE MINIMUM/MAXIMUM LIMITS

	MINIMUM	MAXIMUM
Consecutive memory read cycles	1	3
Consecutive memory write cycles	1	1
Consecutive ALU cycles	1	51
Consecutive CRU cycles	1	16
Frequency of consecutive memory cycle to non-memory cycle events	5 pairs (64 machine cycles during DIV)	

2.7 CYCLE END "CYCEND" SIGNAL

When active, the CYCEND signal of the SBP 9900A indicates that a new machine cycle will be initiated on the next positive-going transition of the clock. Since $\overline{\text{MEMEN}}$ remains active-low for as many as three back-to-back memory read cycles, CYCEND may be used in conjunction with $\overline{\text{MEMEN}}$ to allow external logic to identify the boundaries of each individual memory read cycle.

SECTION III
MEMORY

The SBP 9900A, as discussed in Section 2.1, utilizes system memory for implementation of multiple workspace-register-files and storage of context switch transfer vectors in addition to storage of data and program instructions. The full parallel memory bus of the SBP 9900A facilitates interface to standard types of semiconductor memory devices. For nonvolatile program/data storage, Texas Instruments offers a wide selection of factory-masked ROMs, field-programmable PROMs, and erasable EPROMs.

3.1 MEMORY ORGANIZATION

The SBP 9900A forms a 16-bit address word, internally, which describes a 64K x 8-bit address space. The 15 most-significant bits of the address word are passed to external memory to describe a 32K x 16-bit address space. Consequently, all parallel data transfers between the SBP 9900A and memory are formatted as 16-bit words. The least-significant bit of the 16-bit address word is retained internal to the SBP 9900A to distinguish between the even and odd 8-bit halves (bytes) of the 16-bit memory word.

Although the above partitioning of address space allows the SBP 9900A to process with equal effectiveness both 16-bit words and 8-bit bytes, the total number of directly addressable 16-bit words is bounded by an address reach of 32K. This address reach may be indirectly expanded through utilization of address bank-mapping techniques. For example, two 32K word memory banks, shown in Figure 3-1, can be activated exclusive from one another through program control of a CRU bit. In other words the SBP 9900A, under program control, executes either a Set Bit To One (SBO) or Set Bit To Zero (SBZ) CRU instruction to activate the desired memory bank. As a result, memory depth is indirectly expanded, for purpose of this example, from 32K x 16-bits to 64K x 16 bits.

In addition to standard storage of data and program object code, the memory space of an SBP 9900A system is also used for implementation of multiple workspace-register-files plus storage of context switch transfer vectors (new WP and PC values). Although workspace-register-files can be located anywhere in the available memory space, the context switch transfer vectors are restricted to dedicated memory locations as shown in the general SBP 9900A memory map of Figure 3-2.

3.1.1 $\overline{\text{RESET}}$ TRANSFER VECTORS. Activation of the $\overline{\text{RESET}}$ signal command causes the SBP 9900A to initiate a level-zero interrupt sequence. As a result, the SBP 9900A traps to memory locations 0000_{16} and 0002_{16} to fetch the stored transfer vectors (new WP and PC values respectively) which define the program environment for the associated $\overline{\text{RESET}}$ service routine.

3.1.2 HARDWARE INTERRUPT TRANSFER VECTORS. Memory locations 0004_{16} through $003E_{16}$ contain the hardware interrupt vectors (new WP and PC values) which define the program environment for the corresponding service routine associated with each hardware interrupt level (1 through 15). If a particular hardware service interrupt level is not used in a given system, then the corresponding memory words (2) which normally contain the associated transfer vectors can be used for general program/data storage.

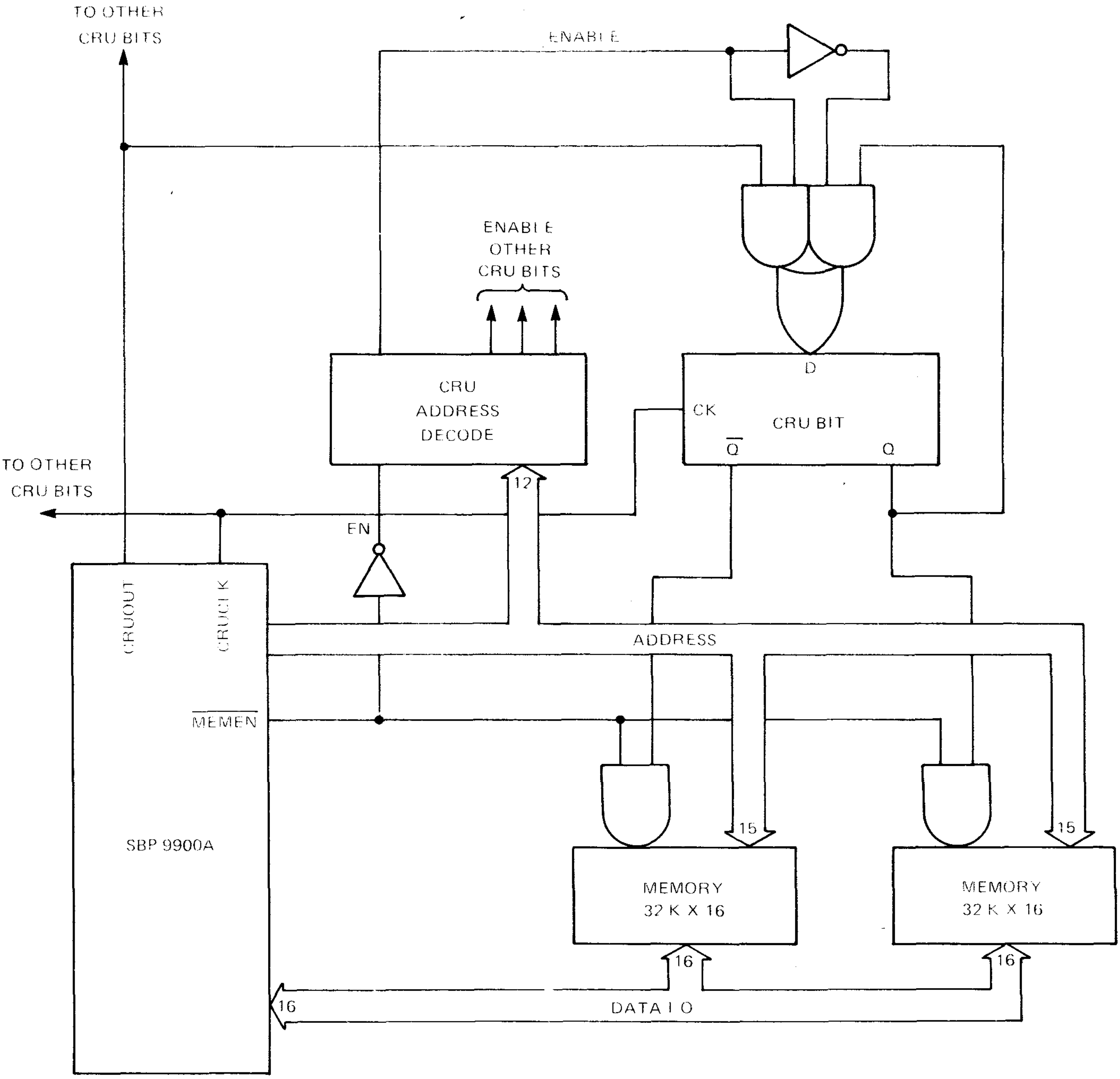


Figure 3-1. Memory Bank Mapping Via CRU Bit

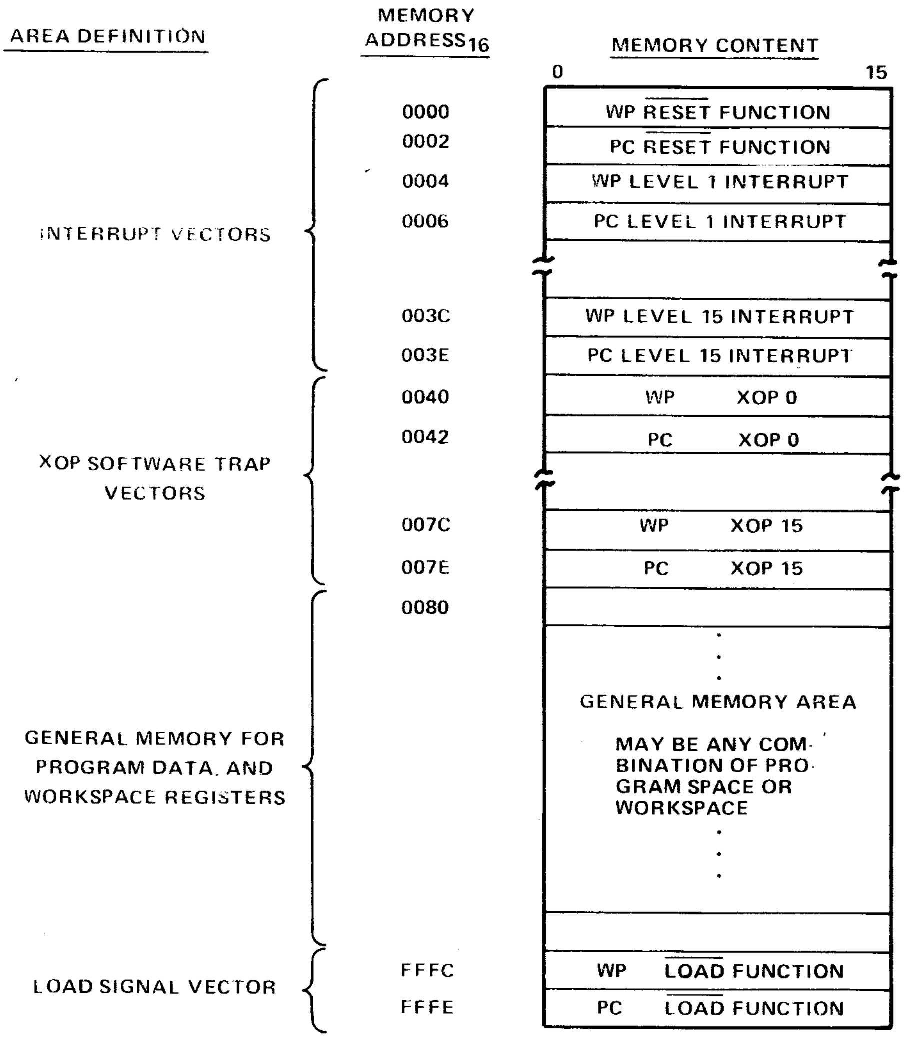

AREA DEFINITION
MEMORY ADDRESS$_{16}$
MEMORY CONTENT
0
15
INTERRUPT VECTORS
0000
WP $\overline{\text{RESET}}$ FUNCTION
0002
PC $\overline{\text{RESET}}$ FUNCTION
0004
WP LEVEL 1 INTERRUPT
0006
PC LEVEL 1 INTERRUPT
003C
WP LEVEL 15 INTERRUPT
003E
PC LEVEL 15 INTERRUPT
XOP SOFTWARE TRAP VECTORS
0040
WP XOP 0
0042
PC XOP 0
007C
WP XOP 15
007E
PC XOP 15
GENERAL MEMORY FOR PROGRAM DATA, AND WORKSPACE REGISTERS
0080
GENERAL MEMORY AREA
MAY BE ANY COMBINATION OF PROGRAM SPACE OR WORKSPACE
LOAD SIGNAL VECTOR
FFFC
WP $\overline{\text{LOAD}}$ FUNCTION
FFFE
PC $\overline{\text{LOAD}}$ FUNCTION

Figure 3-2. SBP 9900A General Memory Map

3.1.3 XOP INSTRUCTION: SOFTWARE INTERRUPT TRANSFER VECTORS. Memory locations 0040_{16} through $007E_{16}$ contain the XOP instruction software interrupt transfer vectors (new WP and PC values) which define the program environment for the corresponding subroutine associated with each XOP instruction (0 through 15). If a particular XOP instruction is not used in a given system, then the corresponding memory words (2) which normally contain the associated transfer vectors can be used for general program/data storage.

3.1.4 $\overline{\text{LOAD}}$ TRANSFER VECTORS. Activation of the $\overline{\text{LOAD}}$ signal command causes the SBP 9900A to fetch the stored transfer vectors (new WP and PC values respectively) from memory locations $FFFC_{16}$ and $FFFE_{16}$ which define the program environment for the associated $\overline{\text{LOAD}}$ service routine.

3.1.5 TRANSFER VECTOR STORAGE REQUIREMENTS. The transfer vectors can be stored in either ROM or RAM. ROM-based vectors are fixed and therefore may not be altered under program control. The $\overline{\text{RESET}}$ transfer vectors should reside in non-volatile memory to ensure proper system start-up. The $\overline{\text{RESET}}$ service routine should initialize any transfer vector which is stored in RAM. The program can then manipulate the RAM-based vectors to alter workspace assignments or service routine entry points as necessary.

3.2 MEMORY CONTROL

During read or write memory cycles, the activity of the address bus and data bus is specified by three control signals: memory enable ($\overline{\text{MEMEN}}$), data bus in (DBIN), and write enable ($\overline{\text{WE}}$). $\overline{\text{MEMEN}}$ becomes active (low) with the beginning of any memory cycle and remains active for the duration of that memory cycle. In the case of back-to-back memory read cycles, $\overline{\text{MEMEN}}$ becomes active with the beginning of the first read cycle and remains active until the completion of the last read cycle. DBIN becomes active (high) with the beginning of a memory read cycle and remains active for the duration of that read cycle. In the case of back-to-back memory read cycles, DBIN becomes active with the beginning of the first read cycle and remains active until the completion of the last read cycle. An active DBIN indicates that the SBP 9900A has disabled its data bus output buffers so memory read data can be received. $\overline{\text{WE}}$ becomes active (low) with the negative-going edge of the first clock of a memory write cycle; $\overline{\text{WE}}$ is deactivated with the negative-going edge of the last clock of a memory write cycle. The timing of $\overline{\text{WE}}$ is compatible with the read/write (R/$\overline{\text{W}}$) control signal associated with many standard RAMs.

3.2.1 MEMORY READ CYCLE. Figure 3-3 shows the timing for three back-to-back memory read machine cycles with no wait states. At the beginning of the first read cycle, $\overline{\text{MEMEN}}$ and DBIN become active with a valid address applied on the address lines. $\overline{\text{WE}}$ remains inactive (high) during all memory read operations. If no wait states are required, the READY line must be high prior to the second-going clock of each individual read cycle.At the end of the last memory read cycle, $\overline{\text{MEMEN}}$ and DBIN become inactive. At that time, though the address may change, the data bus remains in the input mode until terminated by the next negative-going clock. Note that $\overline{\text{CYCEND}}$ may be used during back-to-back read cycles, to distinguish the beginning of each individual read cycle.

3.2.2 MEMORY WRITE CYCLE. Figure 3-4 shows the timing for a memory write machine cycle with no wait states. At the start of a memory write cycle, $\overline{\text{MEMEN}}$ becomes active with a valid address applied to the address bus. $\overline{\text{WE}}$ becomes active (low) with the first negative-going clock after $\overline{\text{MEMEN}}$ becomes active; $\overline{\text{WE}}$ becomes inactive with the negative-going clock preceding the end of the memory cycle. DBIN remains inactive throughout the memory write cycle. At the end of the memory cycle, $\overline{\text{MEMEN}}$ becomes inactive. As in the memory read cycle, if no wait states are required, READY must be high before the second positive-going clock of the write cycle. $\overline{\text{CYCEND}}$ may be used by an external device to indicate that the CPU has completed the current memory cycle.

3.2.3 READ/WRITE CONTROL WITH DBIN. In some memory systems, it may be desirable to have READ and WRITE control signals active during the full memory cycle. Figure 3-5 shows how the READ and WRITE signals can be generated. Any cycle in which $\overline{\text{MEMEN}}$ = 0 ''AND'' DBIN = 0 is a memory write cycle.

3.2.4 SLOW MEMORY. Timing compatibility between slow memories and the SBP 9900A can be achieved through invocation of SBP 9900A wait states. If both minimized system power dissipation and maximized system throughput are coexistent designal goals, a compromise system solution can be achieved by selecting slow memories for their general low power dissipation characteristics, and by invoking SBP 9900A wait states to match SBP 9900A memory reference timing to the requirements of the memory. This allows the SBP 9900A to maximize system throughput by operating at a higher speed when performing non-memory operations. To support invocation of SBP 9900A wait states, both READY and WAIT signals (similar in function to other microprocessors) are used to synchronize the speed of the SBP 9900A with that of slow memories.

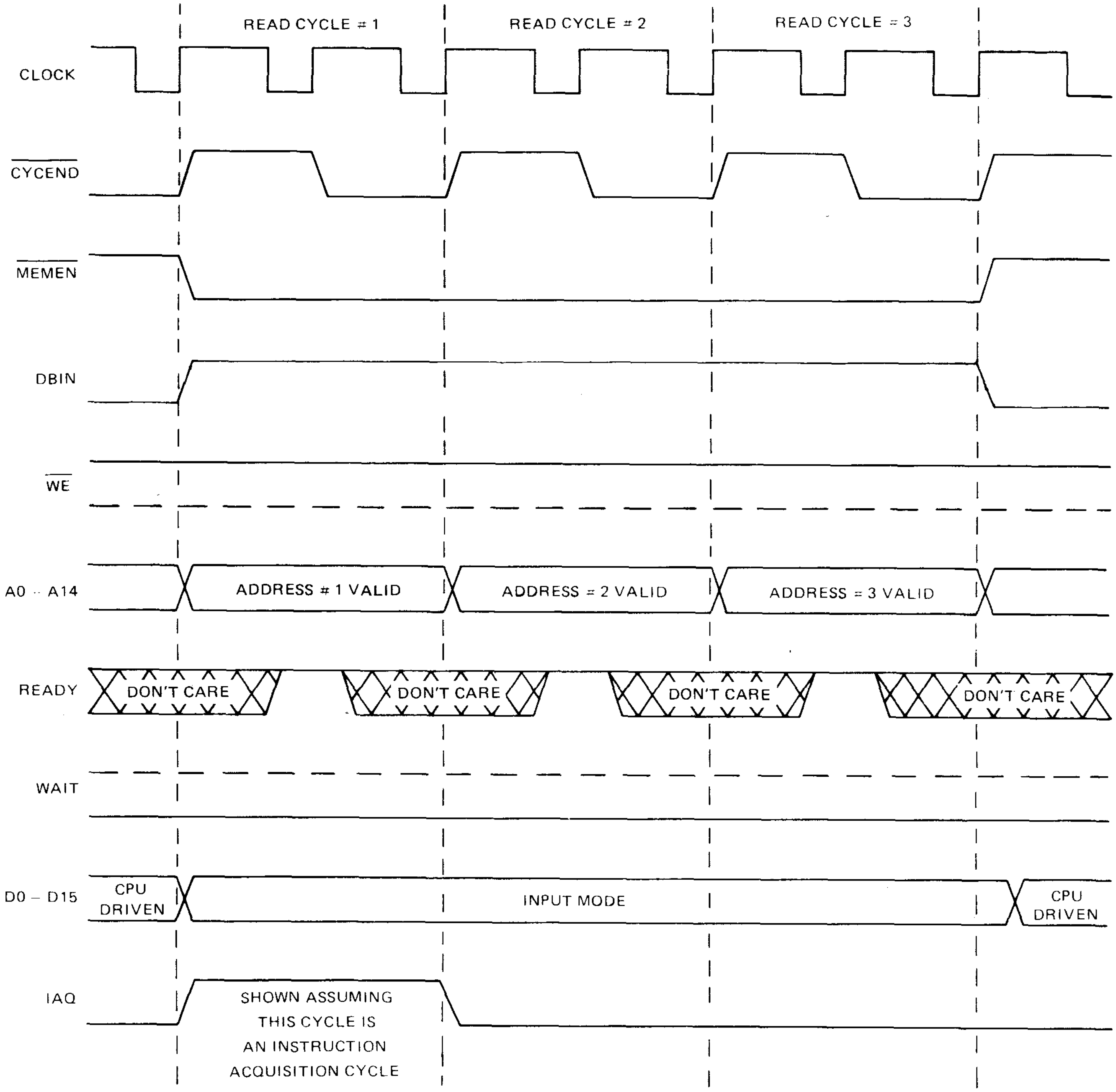


Figure 3-3. Timing for Three Back-to-Back Memory Read Cycles

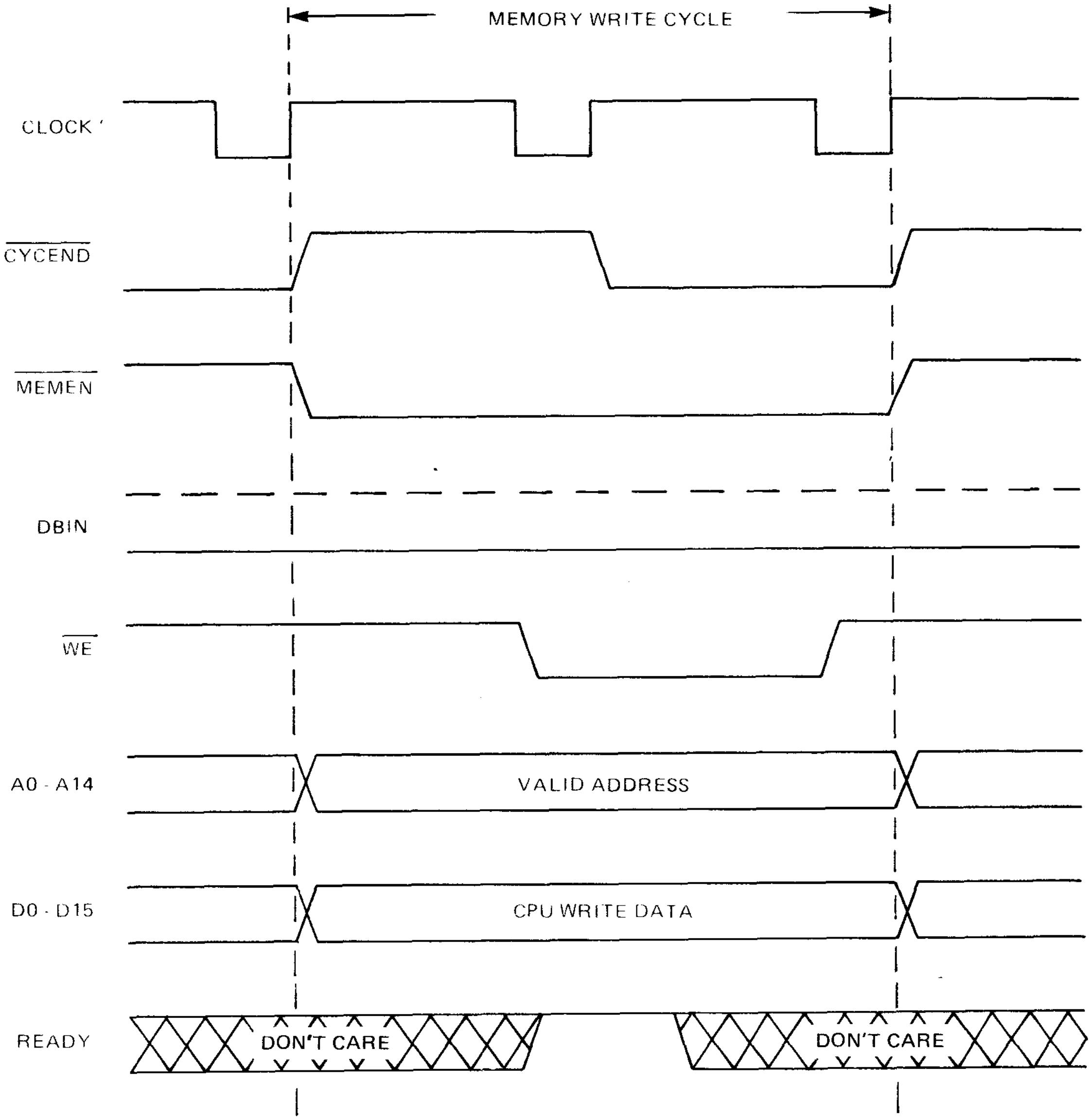


Figure 3-4. Memory Write Cycle Timing

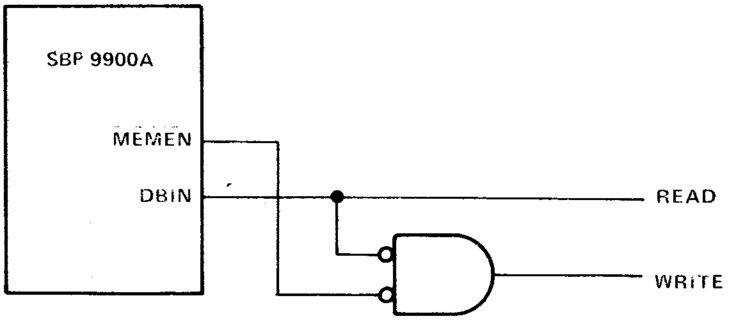


Figure 3-5. READ/WRITE Generation

The wait state timing for memory read and memory write cycles is shown in Figures 3-6 and 3-7 respectively. During both memory read and memory write cycles, READY is sampled on the second positive-going clock of each memory cycle. If READY is not active when sampled, the SBP 9900A enters the wait state, holds the address and data buses at their present values, and activates (↑ high) the WAIT status signal. When an activated (high) READY signal is sampled by a positive-going clock, the SBP 9900A exits the wait state, deactivates (↓ low) the WAIT status signal, and resumes normal completion of the memory cycle. Note that $\overline{\text{CYCEND}}$ comprehends wait states and remains a valid indication of when the present machine cycle terminates with the simultaneous initialization of the next machine cycle.

If one wait state is required to synchronize the speed of the SBP 9900A with all segments of a given memory system, then the WAIT output may be connected directly to the READY input. Note that for invocation of a single wait state, the WAIT output satisfies all of the timing requirements for the READY input. Figure 3-8 shows the connection of the WAIT output to the READY input for invocation of one wait state whenever a slow speed memory segment is addressed. For example, if memory addresses 8000_{16} through $FFFE_{16}$ select a memory segment implemented with slow speed devices, the external address decode logic may generate a slow memory identification signal (SLOMEM = A0) to enable the external wait-state-request logic. Figure 3-9 shows how two consecutive wait states may be invoked to synchronize the speed of the SBP 9900A with a slow speed memory segment. In this case, the propagation of the active WAIT output to the READY input is delayed through a D-type flip-flop by one clock period.

3.2.5 MEMORY ACCESS-TIME CALCULATION. The maximum allowable memory device access time (t_{ACC}) for an SBP 9900A compatible memory system can be calculated through the formula:

$$t_{ACC} = (2 + n)\, t_{CY} - t_{PLH} - t_{D} - t_{su}$$

where n equals the number of invoked wait states and t_{CY} equals the clock period of the SBP 9900A. For maximum design flexibility, t_{CY} may range anywhere from its rated guaranteed minimum (333 ns) to infinity (clock completely stopped at either logic-level high or low). The memory control signals and address bus, in reference to the memory-read cycle timing shown in Figure 3-6, become stable t_{PLH} nanoseconds after the positive-going clock which initiates the cycle. The propagation delay (t_D) effects of address decoding, memory buffering, and control signal gating, is system dependent and can range anywhere from zero nanoseconds to whatever value is indicated. Memory-read data must be stable at the SBP 9900A's data bus inputs t_{su} nanoseconds prior to the positive-going clock which terminates the memory cycle.

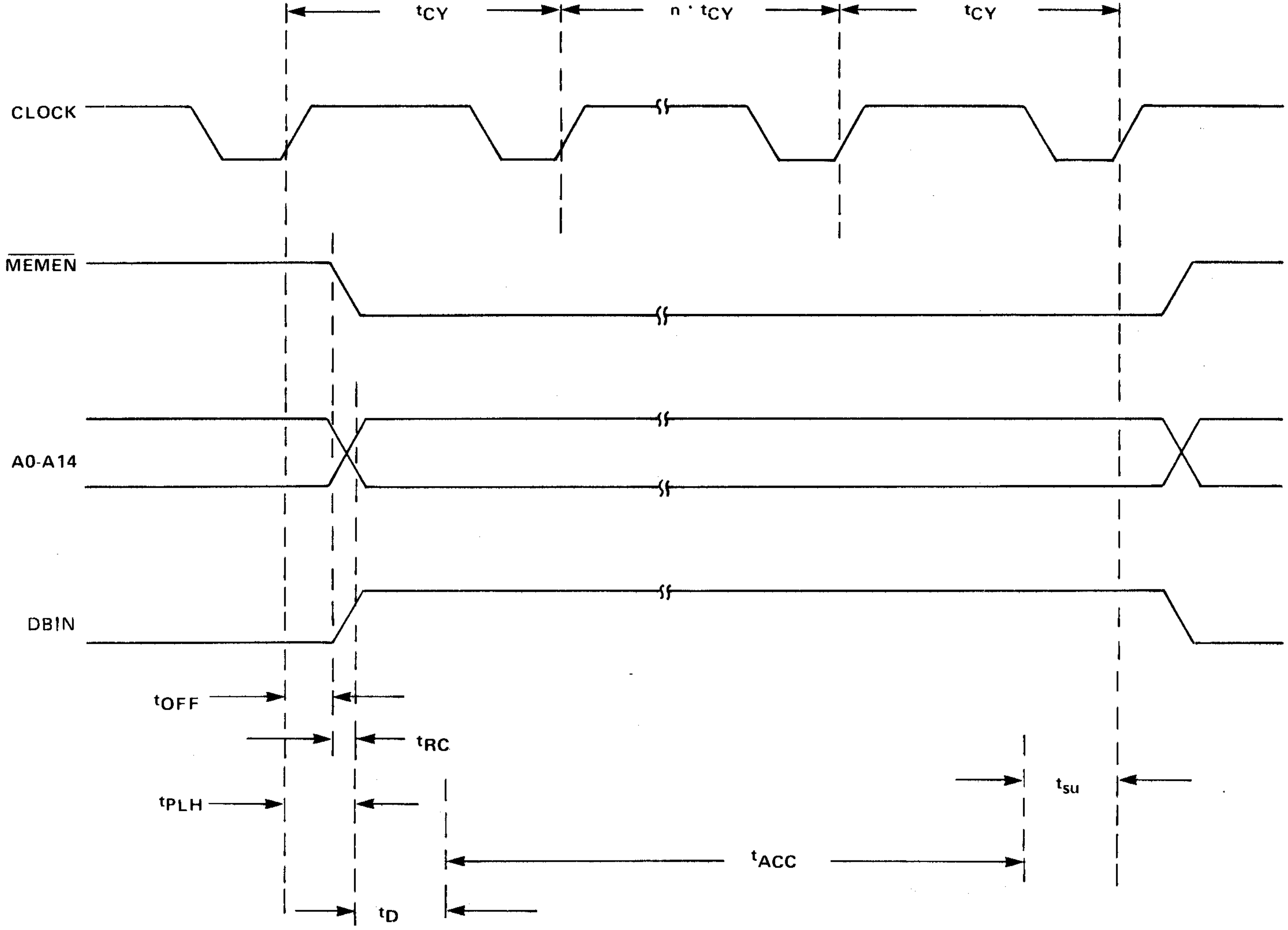


Figure 3-6. Memory-Read Cycle Timing

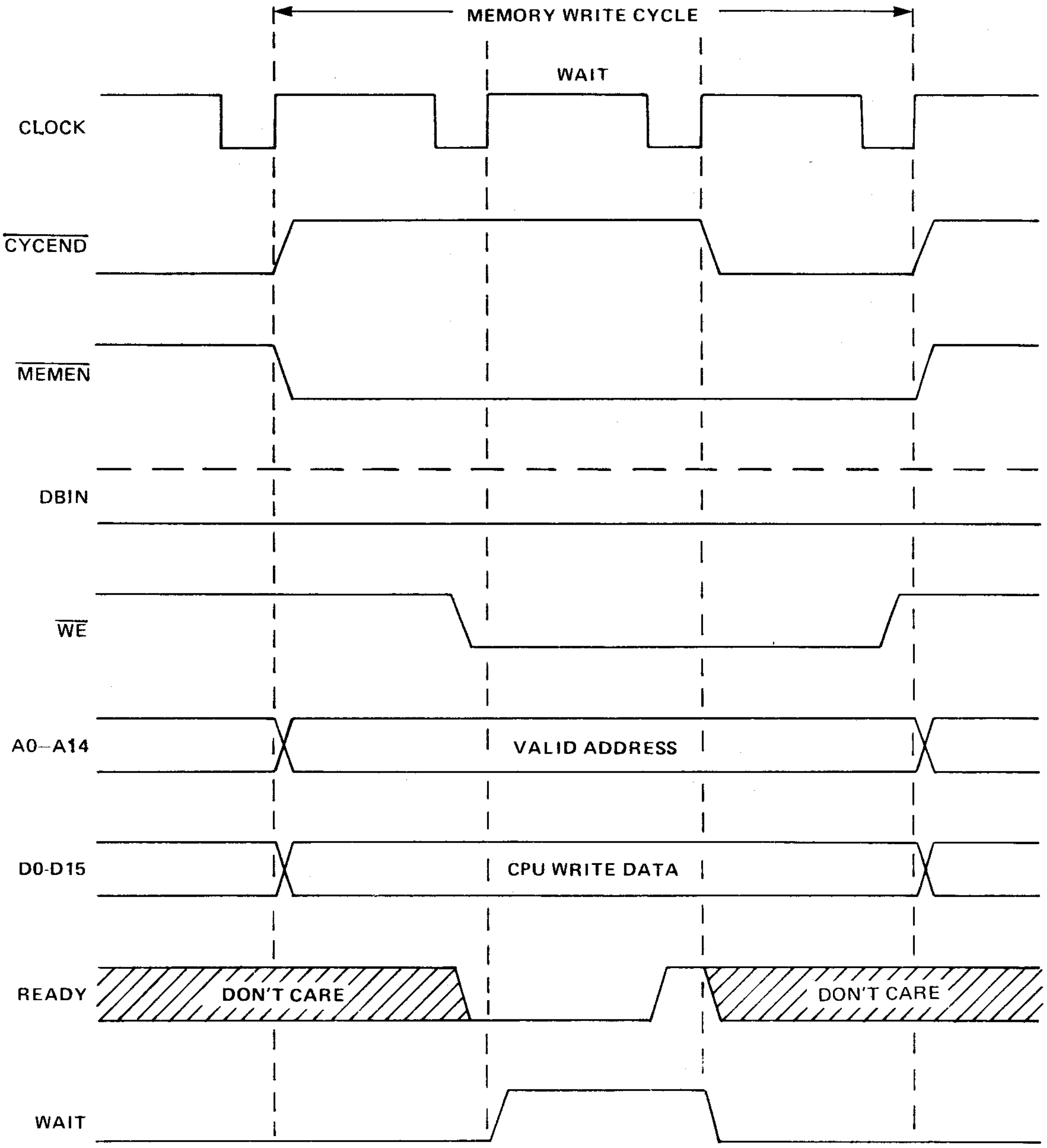


Figure 3-7. Write Memory Cycle Timing with One Wait State

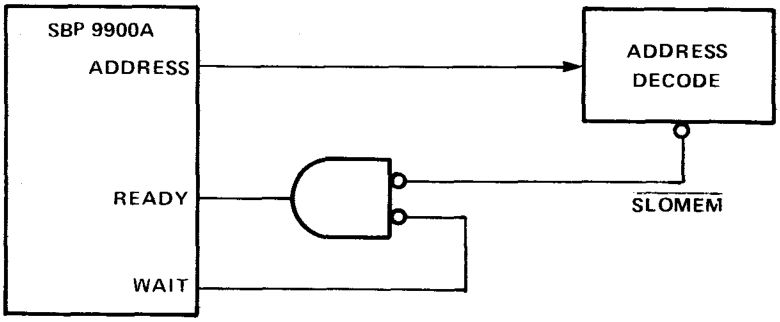


Figure 3-8. Single Wait State for Slow Memory

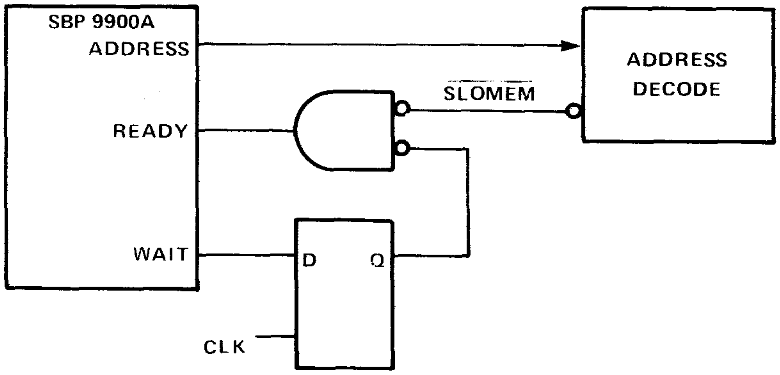


Figure 3-9. Double Wait States for Slow Memory

Time t_{PLH}, as defined above, is composed of two worst case additive components: the turn-off time t_{OFF} associated with a particular SBP 9900A open-collector output transistor, plus the associated bus turn-on time t_{RC}. Time, t_{RC}, is a function of the open-collector pull-up resistor (R_L) charging the bus load capacitance (C_L) to achieve a switching threshold of 1.5 volts. Time, t_{RC}, can be calculated from the formula:

$$f(R_L\,C_L) = 1.5 \text{ volts} = V_{final}\,[1 - e^{-(t_{R_LC_L}/R_LC_L)}]$$

$$1.5 = 5\,[1 - e^{-(t_{R_LC_L}/R_LC_L)}]$$

$$\frac{1.5-5}{5} = (-0.7) = -\,e^{-(t_{R_LC_L}/R_LC_L)}$$

$$(0.7) = e^{-(t_{R_LC_L}/R_LC_L)}$$

$$0.7 = \frac{1}{e+(t_{R_LC_L}/R_LC_L)}$$

$$0.7\,e^{\,t_{R_LC_L}/R_LC_L} = 1$$

$$e^{\,t_{R_LC_L}} = \frac{1}{0.7} = 1.429$$

$$\text{Ln}\,[e^{t_{R_LC_L}/R_LC_L}] = \text{In}\,(1.429) = 0.357$$

$$\frac{t_{R_LC_L}}{R_LC_L} = 0.357$$

$$t_{R_LC_L} = 0.357\,R_LC_L$$

Since time t_{PLH} for each SBP 9900A output has been characterized under circuit load conditions where $R_L = 300\ \Omega$ and $C_L = 150$ pF, the relationship where $t_{R_LC_L} = 0.357\ R_LC_L$ can be used to recalculate t_{PLH} where actual, in system, SBP 9900A load circuit parameters depart from the characterization model. For example, the maximum time t_{PLH} for the SBP 9900A output DBIN is 160 ns with the load circuit parameters $R_L = 300\ \Omega$ and $C_L = 150$ pF. If, however, actual in system circumstances show $R_L = 300\ \Omega$ and $C_L = 25$ pF, then time t_{PLH} for DBIN can be recalculated as follows:

$$t_{PLH} = t_{OFF} + t_{R_LC_L}$$

$$t_{PLH} = 160 \text{ ns when } R_L = 300\ \Omega \text{ and } C_L = 150 \text{ pF}$$

$$t_{OFF} = 160 \text{ ns} - t_{R_LC_L}$$

$$t_{R_LC_L} \times 0.357\,R_LC_L$$

$$= 0.357\,(300)\,(150 \cdot 10^{-12})$$

$$= 16065 \cdot 10^{-12}$$

$$t_{R_LC_L} = 16.065 \text{ ns}$$

therefore: $$t_{OFF} = 160 - 16.065$$

$$t_{OFF} = 143.935$$

With $R_L = 300\ \Omega$ and $C_L = 25$ pf,

$$t_{PLH} = 143.935 + 0.357\,(300)\,(25 \cdot 10^{-12})$$

$$= 143.935 + 2.678$$

and t_{PLH} becomes 147 ns.

Therefore by adjusting the load capacitance (C_L) from 150 pf to 25 pf, time t_{PLH} for DBIN improves from 160 ns to 147 ns.

Table 3-1 and Figure 3-10 show how memory device access time t_{ACC} varies as a function of R_L and C_L for an unbuffered SBP 9900A memory system.

TABLE 3-1. MAXIMUM AVAILABLE MEMORY ACCESS TIME

when:

n = 0 wait states
t_{CY} = 333 ns
t_{SY} = 65 ns
t_{PLH} (Address Bus) = 130 ns
t_D = 41 ns

R_L (Ω)	C_L (pf)	$t_{R_LC_L}$ (ns)	Address Bus t_{PLH} (ns)	Max. t_{ACC} (ns)
300	25	2.7	116.5	444
300	50	5.4	119.2	441
300	100	10.8	124.6	436
300	150	16.2	130	431
1K	150	54.0	167.8	393
3K	150	162.0	275.8	285
5K	150	270.0	383.8	177
7K	150	378.0	491.8	69
8K	150	432.0	545.8	15
10K	150	540.0	653.8	(−93)
10K	100	360.0	473.8	87
10K	50	180.0	293.8	267
10K	25	90	203.8	357

A more complex, buffered memory system is shown in Figure 3-11. Memory device access time t_{ACC} for that system can be calculated as follows:

Assumptions

a. The worstcase path for calculating t_{ACC} is through the address decoder (SN74S138) and associated buffer (SN74S241),

b. SN74S138 and SN74S241 combined $t_D = 70$ ns,

c. $R_L = 300\ \Omega$; $C_L = 50$ pf; $C_M = 150$ pf,

d. $t_{su} = 65$ ns,

e. $t_{CY} = 333$ ns, and

f. N = 0 wait states.

Calculations

(a.) t_{PLH} for A0→A2 is 130 ns for $R_L = 300\ \Omega$ and $C_L = 150$ pF.

(b.) Recalculating t_{PLH} for $R_L = 300\ \Omega$ and $C_L = 50$ pF:

$t_{PLH} = t_{OFF} + t_{R_LC_L}$,

$t_{OFF} = 130 - 0.357\,(300)\,(150 \cdot 10^{-12})$,

$t_{OFF} = 113.9$,

recalculated $t_{PLH} = 113.9 + 0.357\,(300)\,(50 \cdot 10^{-12})$, and

recalculated $t_{PLH} = 119$ ns.

c. $t_{ACC} = 2\,t_{CY} - t_{PLH} - t_D - t_{su}$

$= 2\,(333.3) - 119 - 70 - 65$

$= 667 - 254$

$t_{ACC} = 413$ ns.

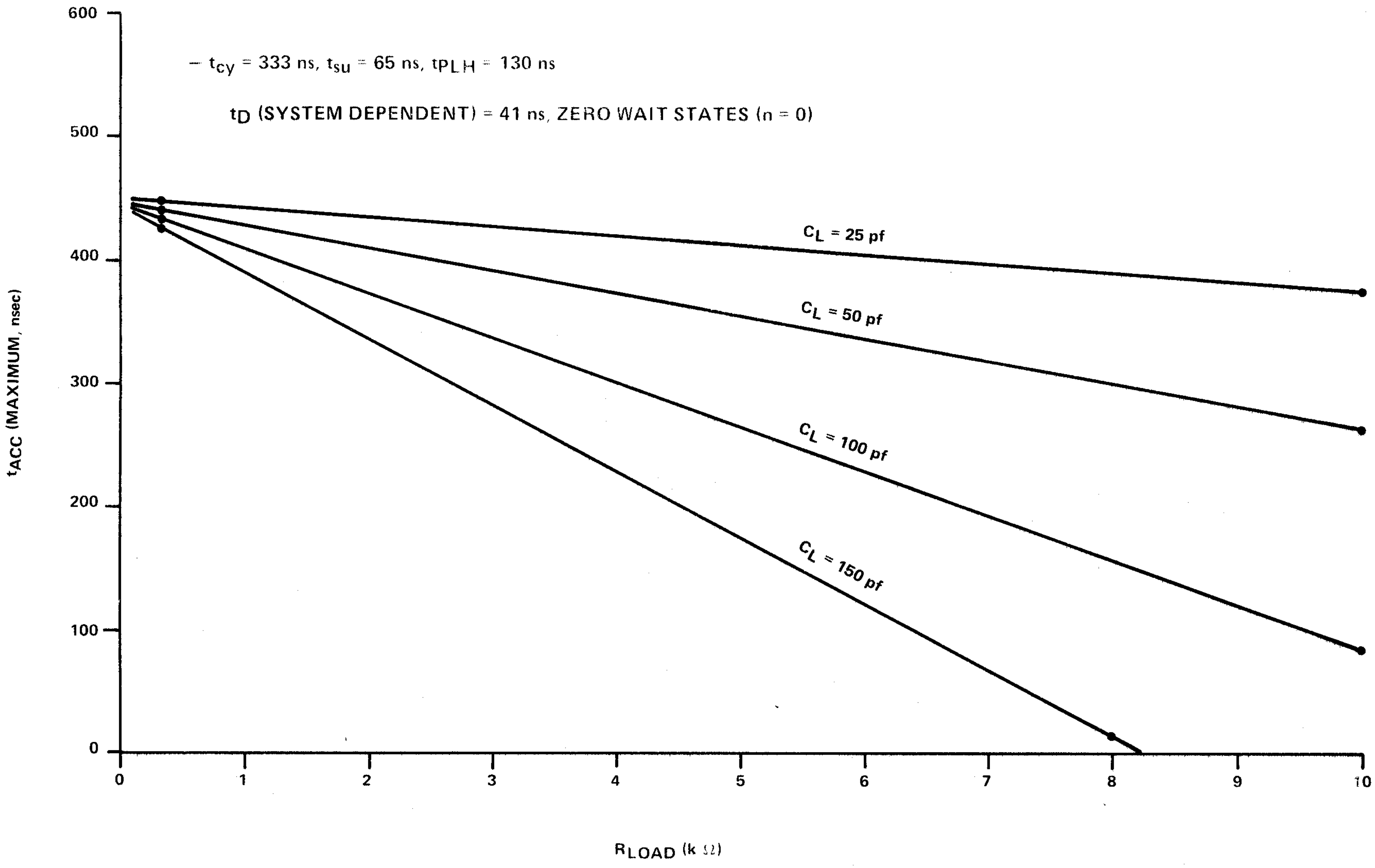


Figure 3-10. Maximum Memory Access Time vs R_L, C_L

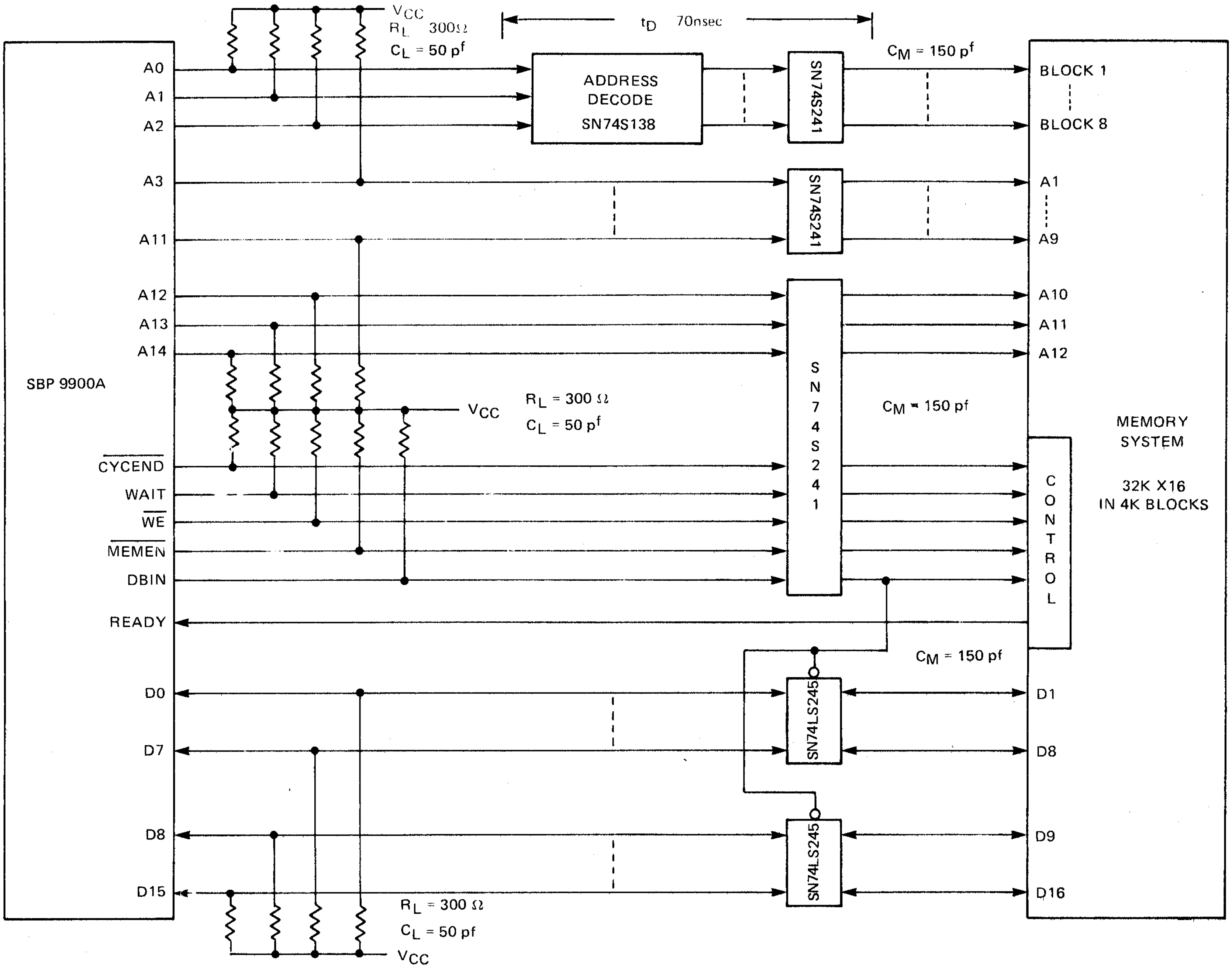


Figure 3-11. SBP 9900A Buffered Memory System

Note that, in Figure 3-11, the buffered form of DBIN is used to select the direction control on the SN74LS245 bidirectional data bus buffers. This is because the SBP 9900A's output transistors sink current from two sources: 1) R_L, and 2) the device(s) being asserted. With a total sink current capability of 20 mA associated with each SBP 9900A output, and $R_L = 300\ \Omega$ accounting for 17 mA, only 3 mA remain which can be supplied by the asserted device(s) without violating the 9900A's output current-sink specification.

$$I_{OL} = 20\text{ mA} = I_{RL} + I_{device}$$

$$I_{RL} = \frac{V_{CC} - V_{OL}}{R_L} = \frac{5.5 - 0.4}{300} = 17\text{ mA}$$

$$I_{device} = 20 - 17 = 3\text{ mA}$$

Therefore, using DBIN directly to drive the SN74S241 plus both SN74LS245s would require the SBP 9900A's output DBIN transistor to sink 23 mA [(I_R = 17) plus (3 I_{device} = 3(2) = 6)] in violation of its specified capability. Further note that the buffered SBP 9900A memory system of Figure 3-11 offers a convenient, 3-state means through which the SBP 9900A can be isolated from the memory system for purposes of effecting either block, clock-stopped, or transparent DMA as described in Section 3.6.

3.2.6 WAIT STATE CALCULATION. For a memory system with given t_{ACC} characteristics, synchronization to the SBP 9900A may be accomplished by invoking SBP 9900A wait states. The number of invoked wait states (n) can be calculated through the formula:

$$n = \frac{t_{ACC} + t_{PLH} + t_D + t_{su} - 2\,t_{CY}}{t_{CY}}$$

where t_{PLH}, t_D, t_{su}, and t_{CY} are as defined in Section 3.2.5 and shown in Figure 3-6. This approach can result in a non-integer value for "n". In that case, "n" must either be rounded to the next higher integer (i.e., n = 1.337 → n = 2) or the parameters t_{PLH}, t_D, and t_{CY} must be adjusted such that the calculated value of "n" becomes an integer. Time t_{CY} can be adjusted by varying the clock frequency of the SBP 9900A. Figure 3-12 shows memory access time (t_{ACC}) for n = 0, 1, 2, 3 memory wait states.

3.3 STATIC MEMORY

Static RAMs, PROMs, and ROMs are easily interfaced to the SBP 9900A.

3.3.1 STATIC MIXED MEMORY. The static 512-word by 16-bit memory system shown in Figure 3-13 is composed of one 256-word bank of SN54S287 PROMs and one 256-word bank of SN74S207 RAMs. When MEMEN is active (low), A0 is used to enable either the PROM memory bank or the RAM memory bank. If A0 is low, the PROMs are selected; if A0 is high the RAMs are selected. A7 through A14 select 1 of 256 words within the enabled PROM or RAM memory bank. Since the full addressing reach of the SBP 9900A is not required to support this particular memory system, A1 through A6 are not used.

When the RAM memory bank is enabled, DBIN is used to select the I/O mode of the SN74S207s. If DBIN is high, the RAMs bidirectional I/O is selected to the output (read) mode; if DBIN is low, the RAMs I/O is selected to the input (write) mode. When DBIN is low, WE is allowed to access the SN74S207s write control.

The I^2L SBP 9818 ROM and TMS 40L45 memory system of Figure 3-14 also uses DBIN to distinguish between READ/WRITE RAM operations.

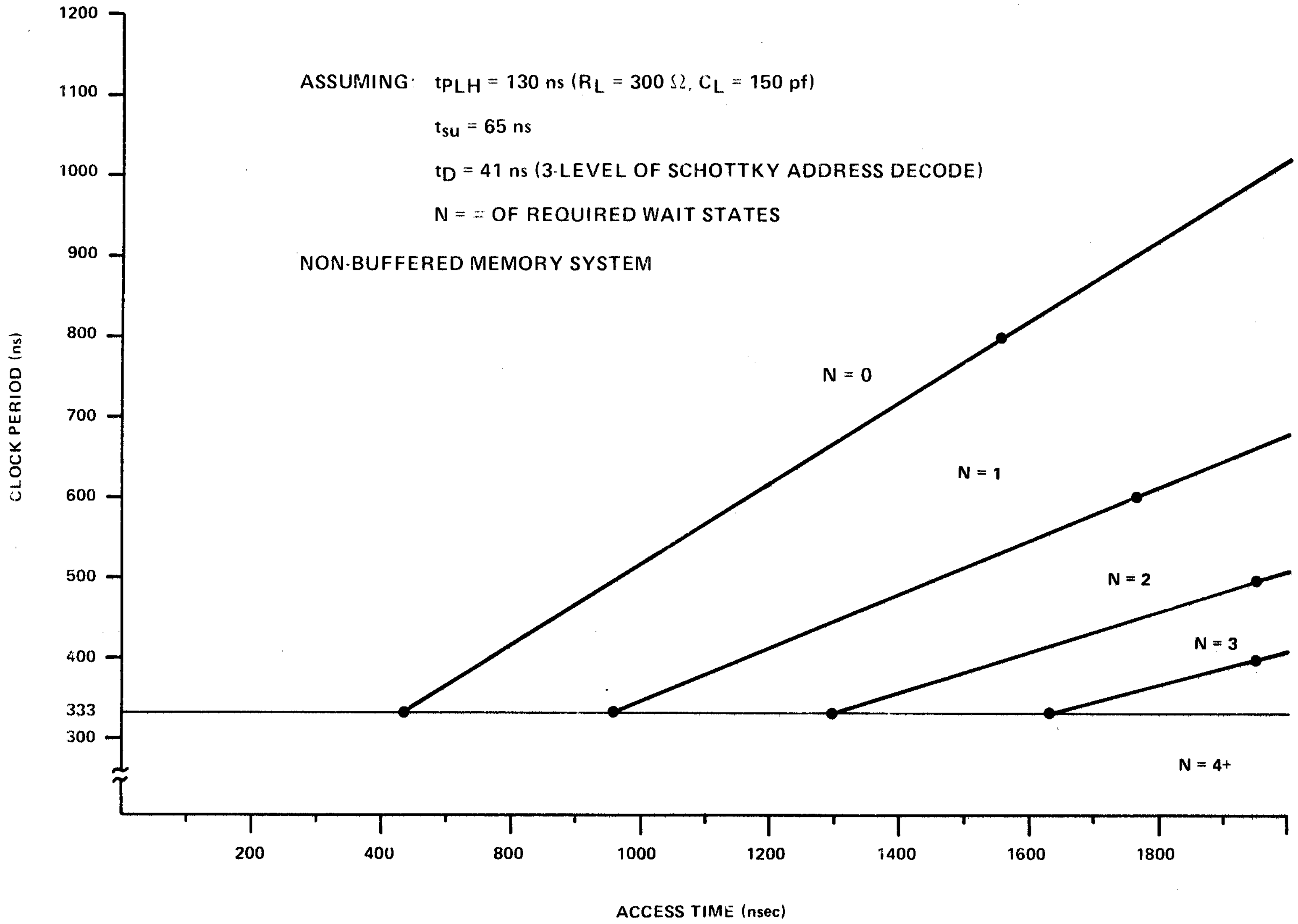


Figure 3-12. Memory Wait Time for Slow Memory

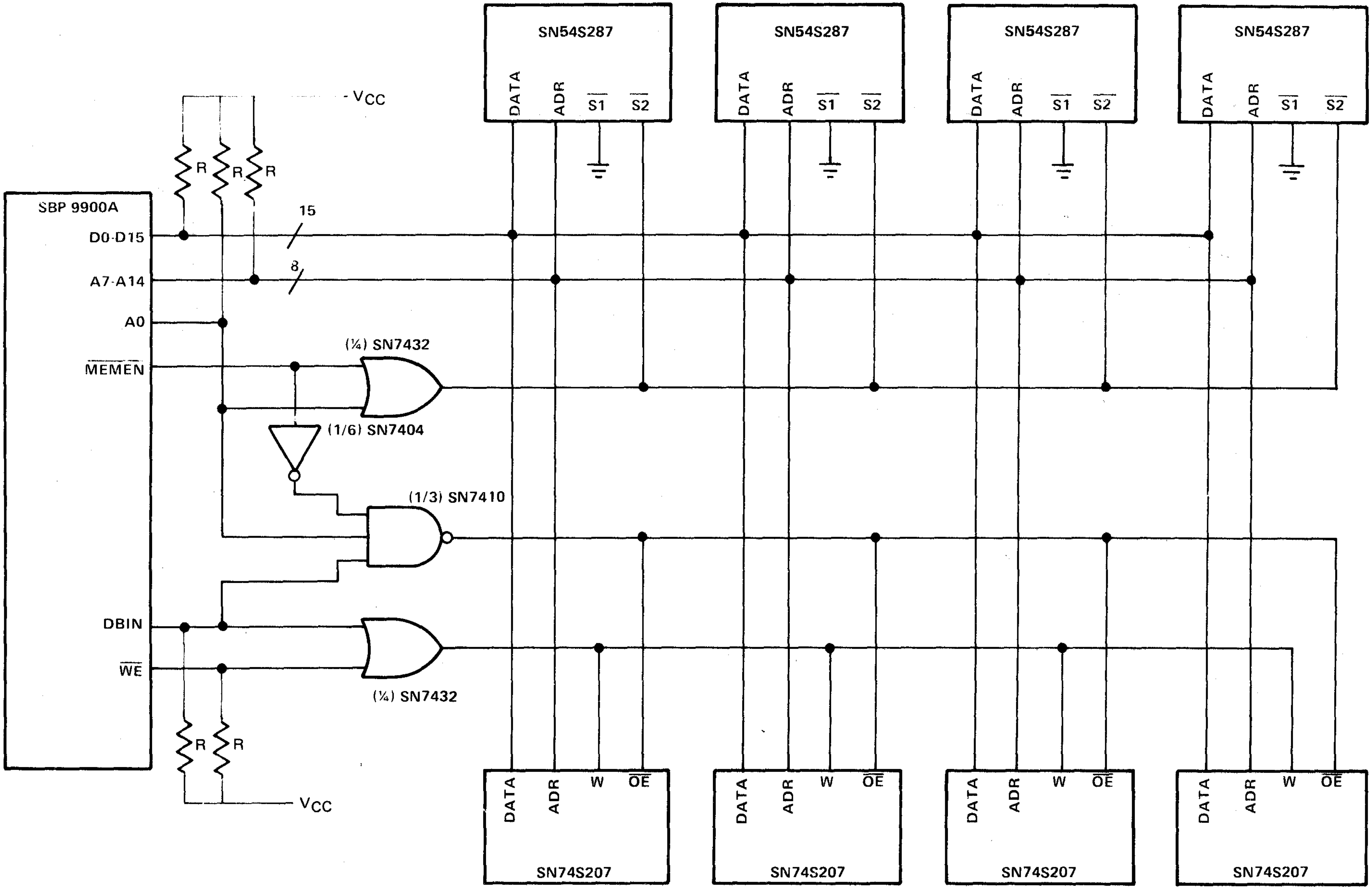


Figure 3-13. Static Mixed Memory

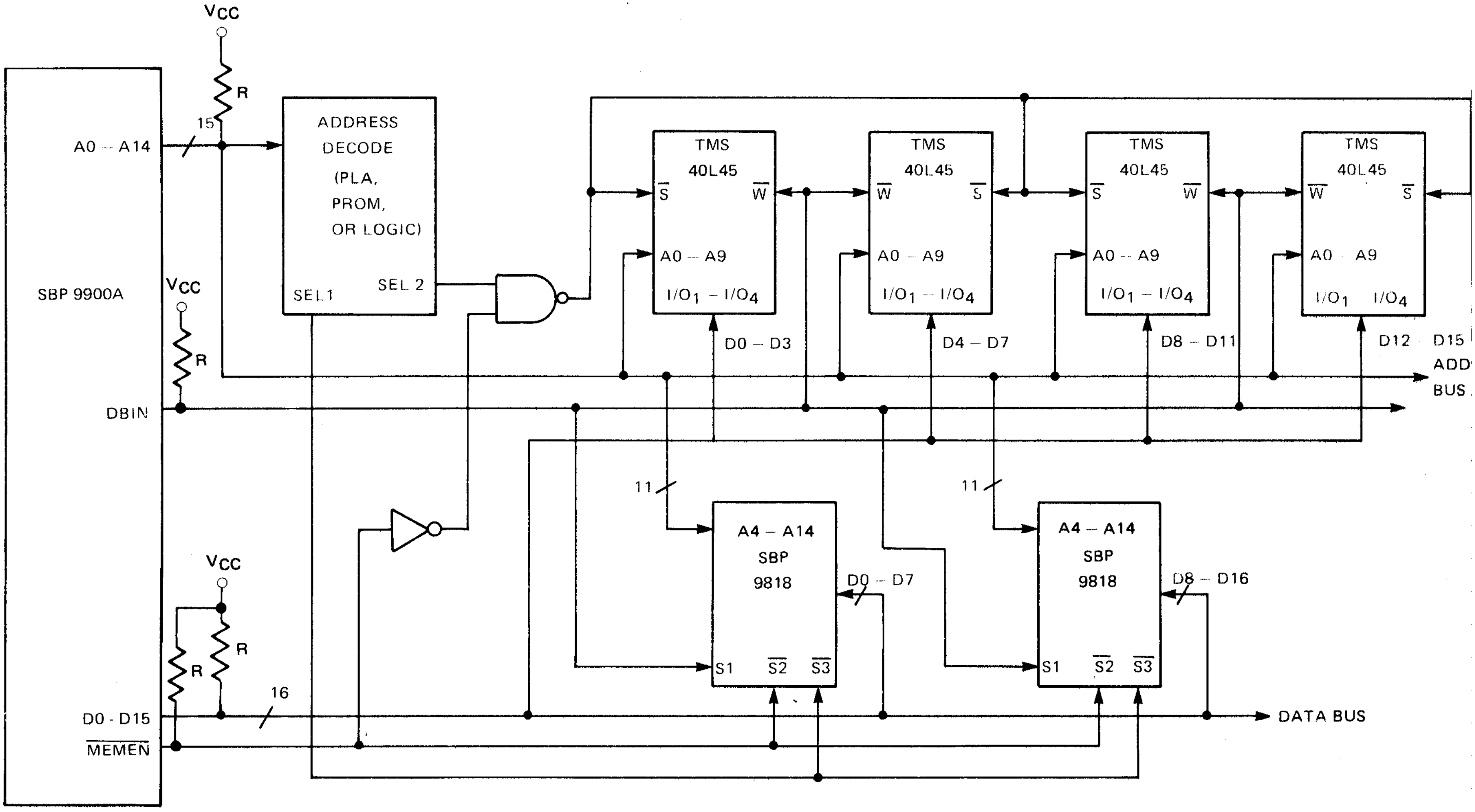


Figure 3-14. Memory System Using SBP 9818 I²L ROM and TMS 40L45 RAM

3.3.2 STATIC MEMORY TIMING AND LOADING. The memory timing of the SBP 9900A is compatible with the requirements of most standard static memory devices. Most bipolar static memory devices are fast enough to completely eliminate the need to invoke wait states. In that case, the READY input of the SBP 9900A may be permanently forced to the logic-level high polarity.

To minimize bus loading, the outputs of most memory devices can be placed, when not enabled, in the high impedance state. Each output of the SBP 9900A is capable of sinking a cumulative current of 20 mA. For logic-level low, this cumulative sink current is composed of that current sourced by the enabled memory device plus that current sourced by the open-collector pull-up resistor (R_L) that is user-wired to each SBP 9900A output. Through proper choice of the open-collector pull-up resistors value (R_L), the SBP 9900A can be tailored to the input-voltage/current requirements of most 5-volt memory devices. There is a tradeoff however, between pull-up resistance (R_L) versus memory bus rise time, power dissipation, and sourcing ability. As a general rule, the smaller the value of R_L, the shorter the memory bus rise time, the higher the power dissipation, and the fewer memory inputs that can be sourced. The larger the value of R_L, the longer the memory bus rise time, the less power dissipation, and the more memory inputs that can be sourced. Consideration must also be given to minimum and maximum input voltages for the high or low level inputs to a particular memory device. A guide to selection of appropriate pull-up resistor values is given in Section 8.2. Due to the large capacitances and increased sourcing demands imposed by some memory systems, it may become desirable to use buffers external to the SBP 9900A to decrease propagation delays and provide the sourcing capability needed. Texas Instruments manufactures a variety of buffer circuits compatible with the SBP 9900A. The SN54LS245 noninverting, bidirectional octal buffer with 3-state outputs is an example.

3.4 OTHER MEMORIES

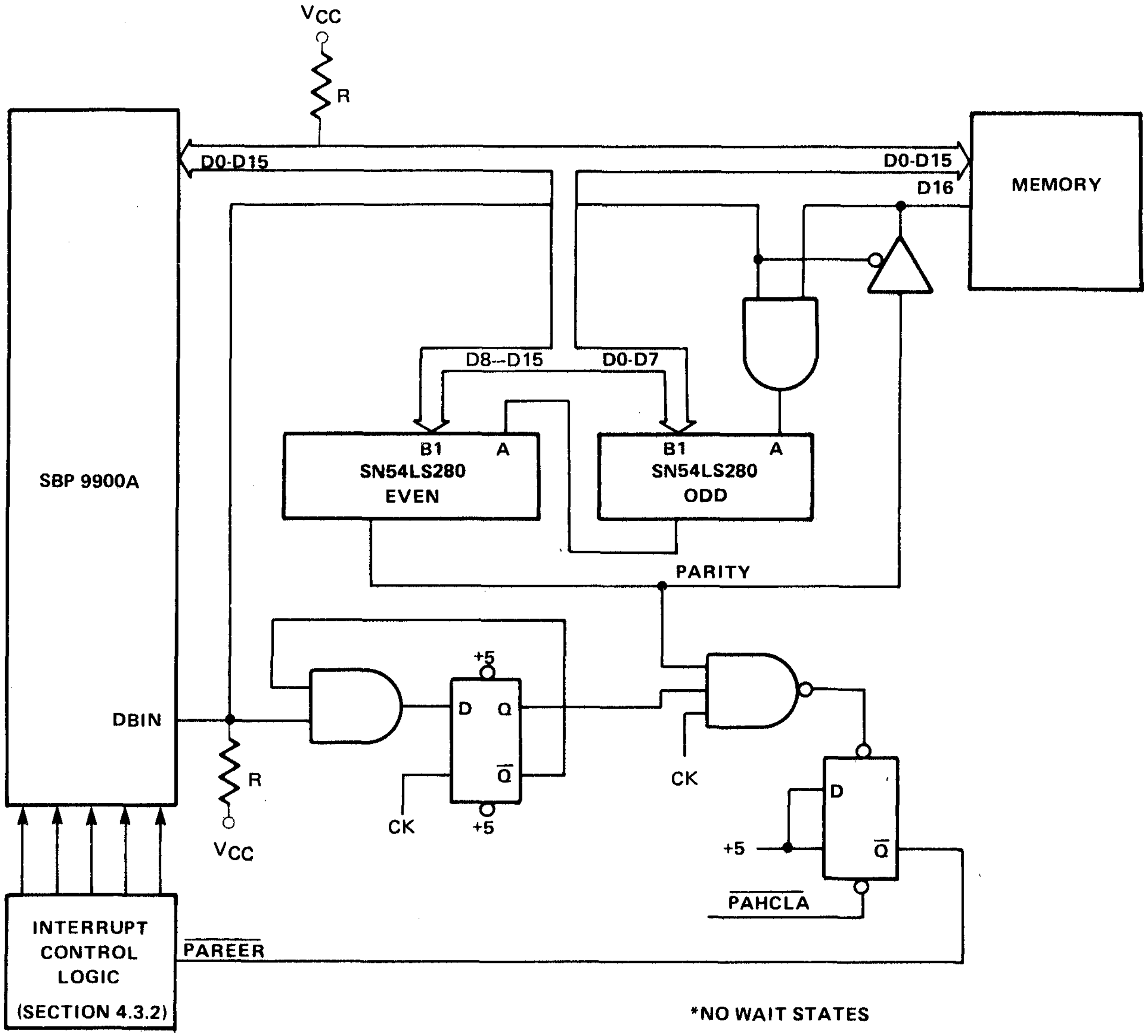
Because of its static I^2L implementation, the SBP 9900A is recommended for use with static memory systems. However, the SBP 9900A is also completely capable of use with dynamic memory systems. A complete discussion of the use of dynamic memories with the SBP 9900A is contained in Appendix G.

3.5 MEMORY PARITY

Parity or other error detection/correction schemes are often used to minimize the effects of memory errors. Error detection schemes such as parity are used to indicate the presence of bad data, while error correction schemes correct single or multiple errors.

The SN54LS280 parity generator/checker can be used to implement memory parity in an SBP 9900A system. The system in Figure 3-15 uses two SN54LS280 circuits to generate and check odd memory parity. During memory write cycles, the generated parity bit is output to bit D16 of the memory. During memory read cycles, the parity is checked and an interrupt (PARERR) is generated if the parity is even.

It should be noted that a faulty memory word will already have been used by the SBP 9900A as an op code, address, or data before the interrupt is generated. For example, an error in bit 8 of the CLR op code will cause the SBP 9900A to branch unconditionally. Consequently when the interrupt is serviced, there will be no linkage to the portion of the program at which the error occurs. A diagnostic routine can often isolate such errors by scanning the memory and checking parity under program control. An error in the diagnostic itself can be extremely difficult to isolate. However an external address register, such as the SN54S299, clocked by IAQ can be used to retain program linkage under the above circumstances. When a parity error is detected, the SN54S299 is frozen, thereby retaining the address of the instruction during which the parity error occurred.



*NO WAIT STATES

Figure 3-15. Memory Parity Generator Checker

3.6 DIRECT MEMORY ACCESS

The SBP 9900A controls CRU-based I/O transfers between the memory and peripheral devices. Data must pass through the SBP 9900A during these program driven I/O transfers; the SBP 9900A may need to be synchronized with the I/O device by interrupts or status bit polling.

Some I/O devices, such as disk units, transfer large amounts of data to or from memory. Program driven CRU-based I/O can require relatively long response times, high program overhead, or complex programming techniques. Consequently, direct memory access (DMA) is used to permit the I/O device to transfer data to or from memory without CPU intervention. DMA can result in improved I/O response time and system throughput, especially for block data transfers. The DMA control circuitry is somewhat more expensive and complex than the economical CRU I/O circuitry and should be used only when required.

SBP 9900A based DMA can take place in several operational modes: the block mode utilizing the $\overline{\text{HOLD}}$/HOLDA facilities of the SBP 9900A, the block mode utilizing the SBP 9900A's static clock-stopping capability, and the transparent mode utilizing those conditions under which the SBP 9900A will not require communication with the memory system. So called "cycle stealing" DMA modes are discussed herein as a subset of the block DMA modes.

3.6.1 $\overline{\text{HOLD}}$/HOLDA BLOCK DMA. SBP 9900A based block DMA may utilize the $\overline{\text{HOLD}}$ and HOLDA facilities. An external DMA controller, as shown in Figure 3-16, activates (low) the $\overline{\text{HOLD}}$ input of the SBP 9900A whenever an I/O device requires direct communication with the memory system. At the beginning of the next

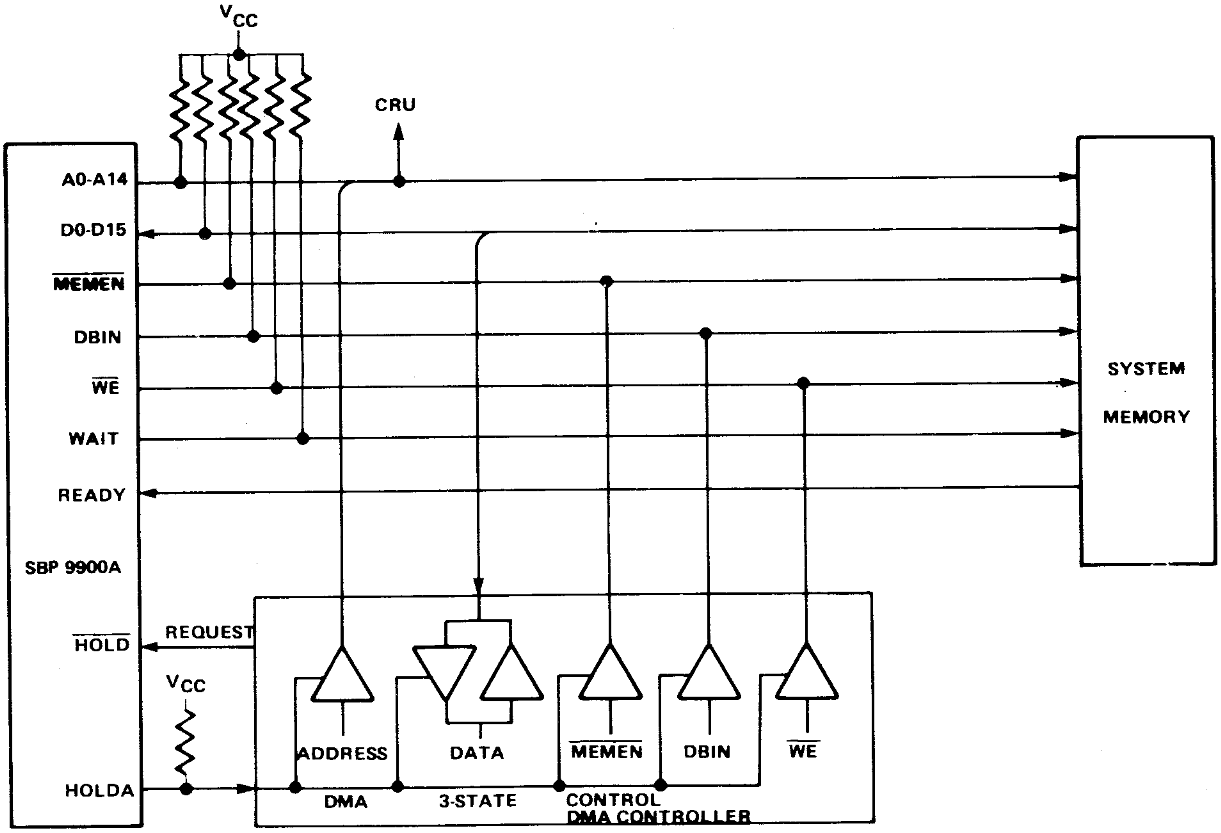


Figure 3-16. $\overline{\text{HOLD}}$/HOLDA Block DMA

non-memory cycle, the SBP 9900A enters the hold state and acknowledges the HOLD directive (request) by activating (↑ high) HOLDA. The maximum latency time from the assertion of HOLD to the activation of HOLDA is the sum of one clock cycle plus three back-to-back memory cycles. Therefore a system with no invoked wait states has a maximum HOLD to HOLDA latency of seven clock periods. Activation of HOLDA indicates that the address bus, data bus, DBIN signal, CYCEND signal, WE signal, and MEMEN signal have been pulled to the logic-level high polarity. Consequently, the DMA controller has the "go-ahead" to generate the proper memory address and appropriately timed control signal so that the I/O device may communicate directly with the memory system free from possible SBP 9900A interference. The I/O device can use the memory system for one data transfer (cycle stealing DMA) or multiple data transfers (block DMA). When the memory communication requirements of the I/O device have been satisfied, the DMA controller releases HOLD and normal SBP 9900A operations resume. The SBP 9900A's HOLD and HOLDA timing is shown in Figure 3-17.

3.6.2 CLOCK-STOPPED BLOCK DMA. SBP 9900A based block DMA can use the static clock-stopping capability of the SBP 9900A. An external DMA controller, as in the buffered SBP 9900A DMA system shown in Figure 3-18, can gain access to the memory system by 1) isolating the SBP 9900A from the memory bus via the 3-state buffers and 2) stopping the SBP 9900A's clock at the logic-level low polarity.* The REQ signal from the DMA controller is synchronized with the SBP 9900A to halt the SBP 9900A's clock after the completion of the current machine cycle (ALU or Memory Cycle). Concurrently with halting the SBP 9900A's clock, DACK is

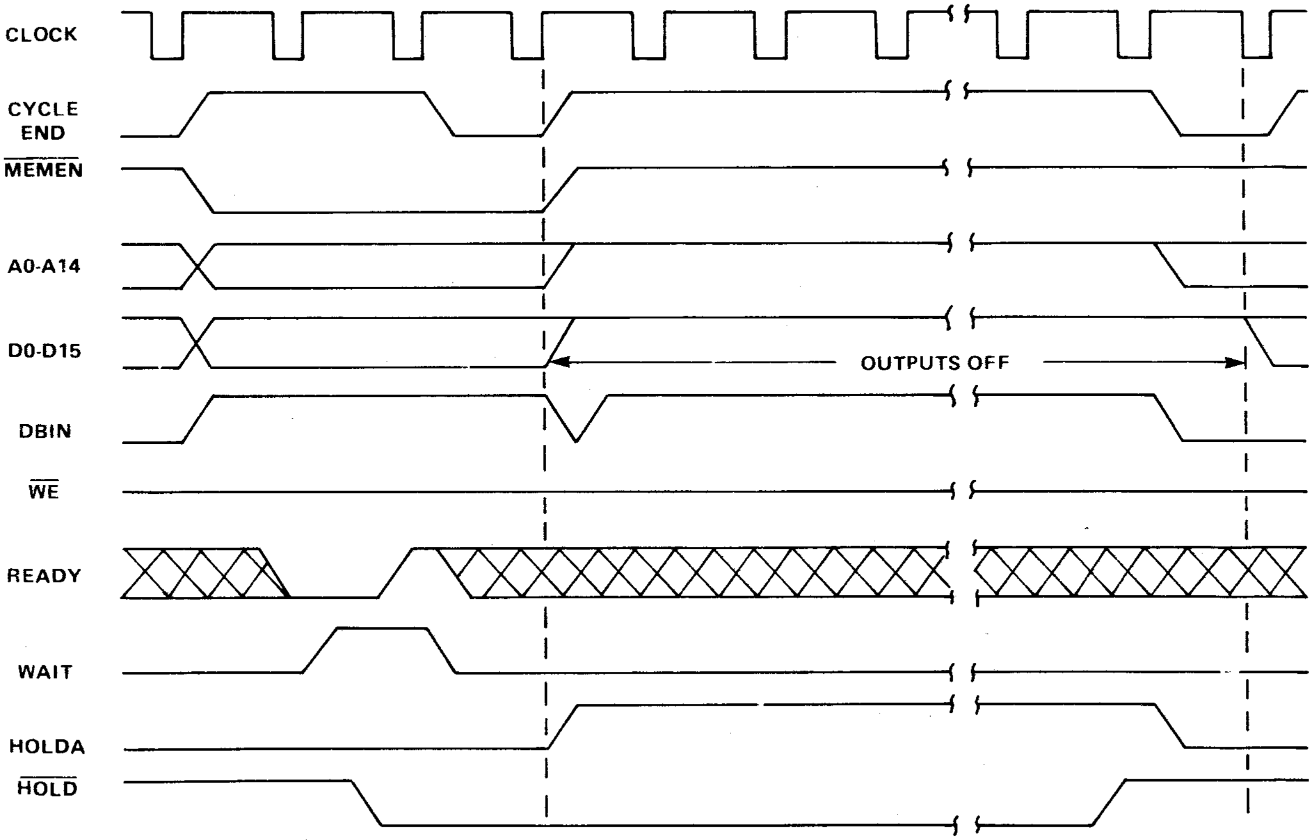
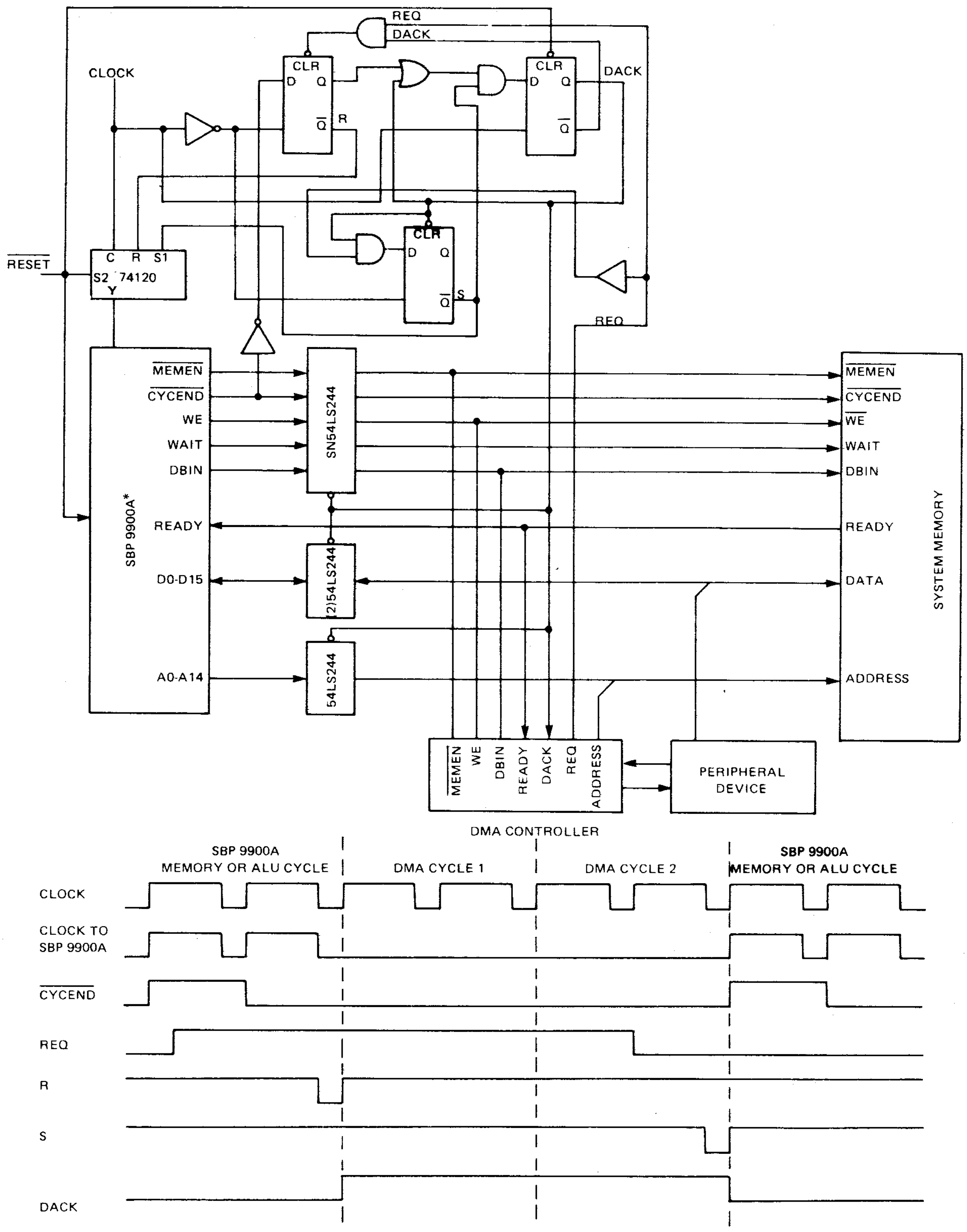


Figure 3-17. SBP 9900A Hold Timing

*The SBP 9900A's static clock may be indefinitely stopped at either the logic-level high or low polarity. It is a function of this example only which requires the SBP 9900A's clock to be stopped while at the logic-level low polarity.



*All outputs must be supplied with an open-collector pull-up resistor.

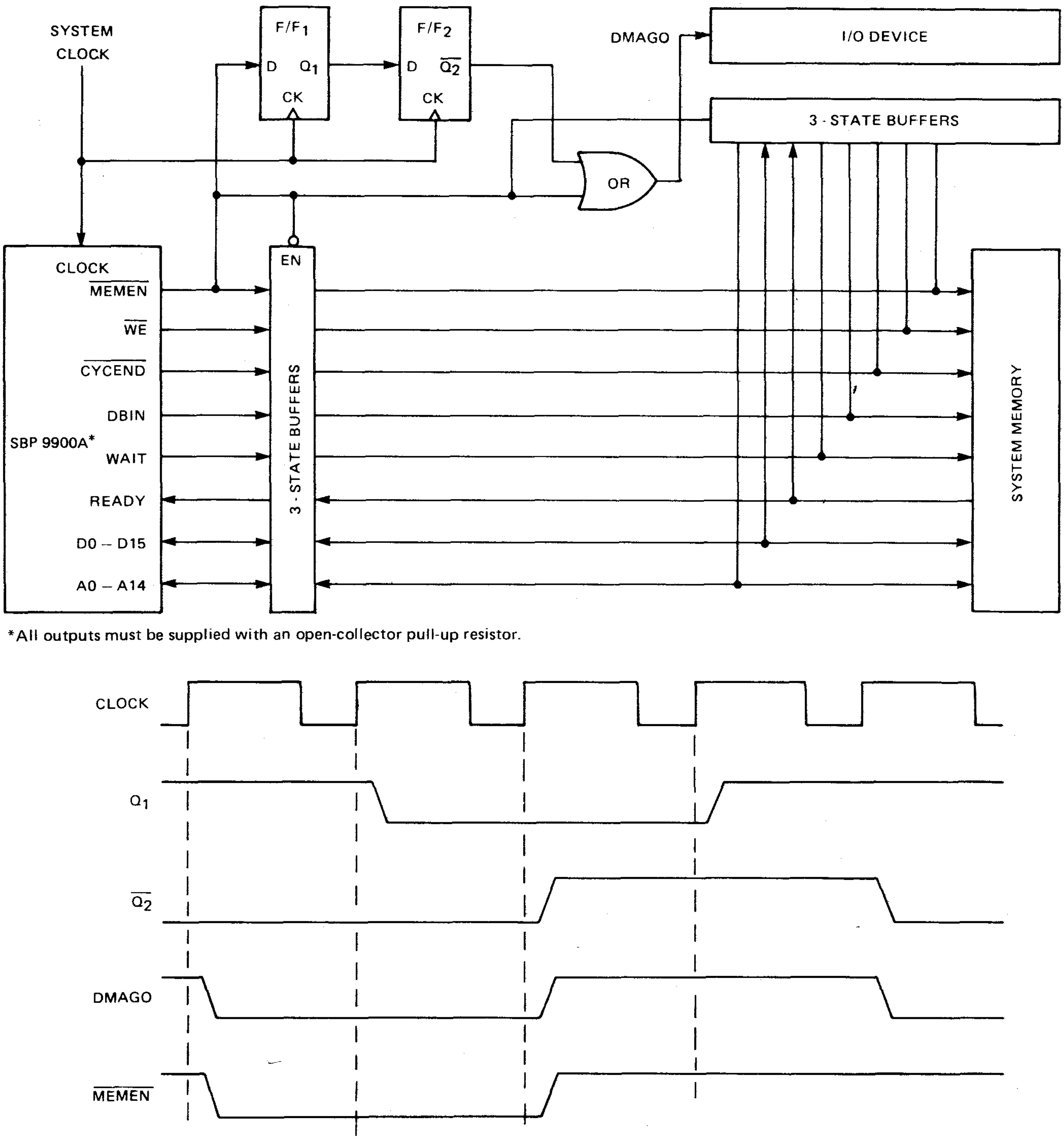
Figure 3-18. Clock-Stopped Block DMA

asserted and the memory bus is 3-stated allowing the DMA Controller to assert the memory bus. When the DMA Controller is in the final clock cycle of the last transfer, REQ is released to indicate to auxillary devices that DMA is completed and that the SBP 9900A may regain memory bus control. The clock control logic then reinitiates the SBP 9900A's clock. The maximum latency between REQ and DACK occurs when REQ just misses the high-to-low transition of the system clock to trigger the R signal which stops the SBP 9900A's clock. Assuming no memory wait states, this maximum latency amounts to two clock periods plus the low time of the clock. In a 3 MHz system this latency equals 777 ns, a 3X improvement over the HOLD/HOLDA handshake method of obtaining DMA as described in Section 3.5.1.

3.6.3 TRANSPARENT DMA. Since a minimium of one ALU cycle always directly follows deactivation (↑ high) of $\overline{\text{MEMEN}}$, $\overline{\text{MEMEN}}$ may be used to define the two clock-period time window DMAGO (as shown in Figure 3-19) through which an external I/O device may communicate with the memory system free from SBP 9900A intervention. When DMAGO becomes active (↑ high), the device has two SBP 9900A clock periods to generate the proper memory address, with appropriately timed signals, and complete the memory transaction. *Note* that since the SBP 9900A's I/O lines are active during ALU cycles, the SBP 9900A must be isolated from the memory system via a 3-state buffer.

3.7 MEMORY LAYOUT. It is generally advantageous to layout memory devices as two dimensional arrays. First, positioning the devices in an orderly fashion simplifies identification of a particular memory element when troubleshooting. Second, and most important, positioning of memory arrays simplifies board layout, shortens interconnections, and generally allows a more compact and efficient utilization of board space. Crosstalk between adjacent lines in memory arrays is minimized by running address and data lines parallel to each other, and by running chip enable signals perpendicular to the address lines.

Memory devices generally require substantially greater supply currents when addressed than other times. Therefore it is important that all power and ground paths be as wide as possible. Furthermore, in order to suppress spikes in supply voltages, it is advisable to decouple supply voltages with capacitors as close as possible to the pins of the memory devices.



*All outputs must be supplied with an open-collector pull-up resistor.

Figure 3-19. Transparent DMA

SECTION IV
INTERRUPTS

The SBP 9900A provides 15 maskable interrupt levels in addition to the RESET and LOAD signal-commands. The SBP 9900A has a priority ranking system to resolve conflicts between simultaneous interrupts, and a level mask to disable lower priority interrupts. Once an interrupt is recognized, the SBP 9900A performs a vectored context switch to the interrupt service routine. RESET and LOAD are initialized by external input signals and should not be confused with the RSET and LREX instructions which are described in Section 5.

4.1 RESET.

The RESET signal-command is normally used to initialize the SBP 9900A following power-up. When active (low), the RESET command inhibits WE and CRUCLK, allows the memory bus and control signals to be pulled to the logic-level high state, and clears both the interrupt mask and the status bit field of the status register. When the RESET command is released, the SBP 9900A fetches the restart vector from memory locations 0000 (WP) and 0002 (PC), stores the old WP, PC, and ST into the new workspace, resets all status bits to zero and starts execution at the new PC. The RESET command must be held active for a minimum of three clock periods. The RESET machine cycle sequence is shown in Figure 4-1.

CYCLE	TYPE	FUNCTION
*	*	Loop While Reset is active
1	ALU	Set Up
2	ALU	Set Up
3	Memory	Fetch New WP, Move Status to T Reg, Clear Status
4	ALU	Set Up
5	Memory	Store Status
6	ALU	Set Up
7	Memory	Store PC
8	ALU	Sct Up
9	Memory	Store WP
10	ALU	Set Up
11	Memory	Fetch New PC
12	ALU	Set Up MAR for Next Instruction

Figure 4-1. RESET Machine Cycles

A convenient method of generating the RESET signal-command is to use the SN54LS121 Schmitt-triggered monostable multivibrator connected as shown in Figure 4-2. Timing resistor and capacitor values are chosen to allow an output pulse of at least 3 clock periods duration. For 3 MHz operation, values of $R_T = 10$ KΩ and $C_T = 200$ pF yield a pulse width of 1.5 μs.

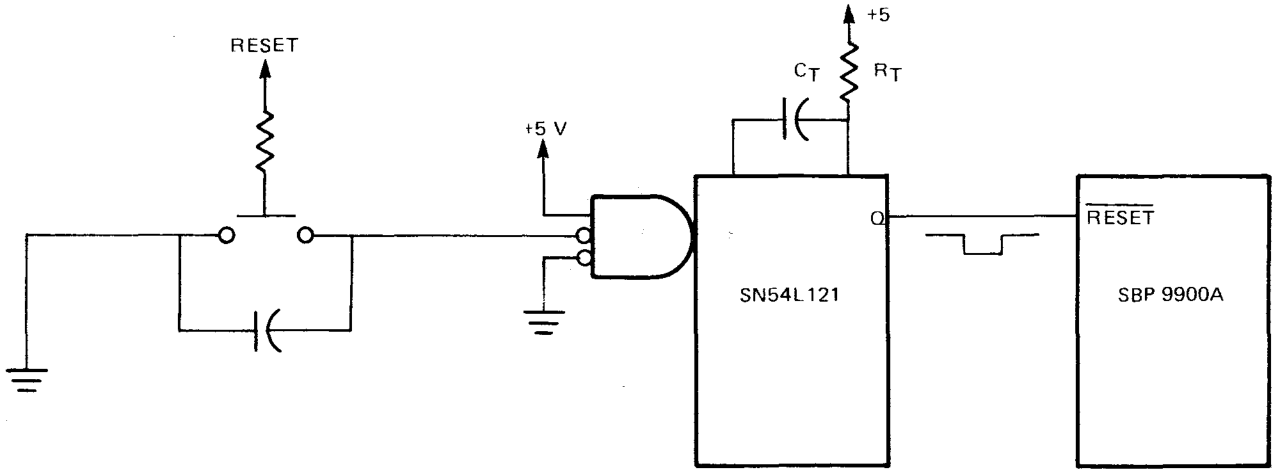


Figure 4-2. $\overline{\text{RESET}}$ Generation

4.2 $\overline{\text{LOAD}}$.

The $\overline{\text{LOAD}}$ signal-command is normally used in conjunction with a restart ROM loader or front panel functions. When active (low), the $\overline{\text{LOAD}}$ command causes the SBP 9900A to perform a non-maskable interrupt. The $\overline{\text{LOAD}}$ command can be used to terminate a CPU idle state.

The $\overline{\text{LOAD}}$ command should be active for one instruction period. Since there is no standard SBP 9900A instruction period, IAQ should be used to determine instruction boundaries as shown in Figure 4-3. If the $\overline{\text{LOAD}}$ command is active during the time that the $\overline{\text{RESET}}$ command is released, the SBP 9900A will perform the $\overline{\text{LOAD}}$ function immediately after the $\overline{\text{RESET}}$ function is completed. The CPU performs the $\overline{\text{LOAD}}$ function by fetching the $\overline{\text{LOAD}}$ vectors from addresses FFFC$_{16}$ (WP) and FFFE$_{16}$ (PC), storing the old WP, PC, and ST in the new workspace, and starting the $\overline{\text{LOAD}}$ service routine at the new PC, as shown in Figure 4-4.

As an example, the $\overline{\text{LOAD}}$ signal can be used to initiate a bootstrap ROM loader. When the $\overline{\text{LOAD}}$ signal is activated, the CPU enters the service routine, transfers a program from peripheral storage to RAM, and then transfers control to the loaded program.

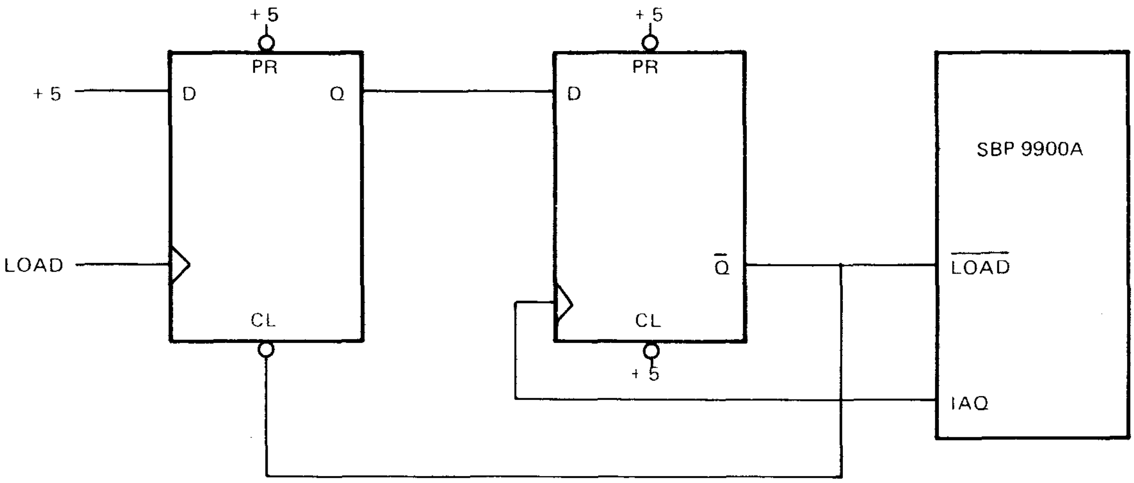


Figure 4-3. $\overline{\text{LOAD}}$ Generation

CYCLE	TYPE	FUNCTION
1	ALU	Set Up
2	Memory Read	Fetch New WP
3	ALU	Set Up
4	Memory Write	Store Status
5	ALU	Set Up
6	Memory Write	Store PC
7	ALU	Set Up
8	Memory Write	Store WP
9	ALU	Set Up
10	Memory Read	Fetch New PC
11	ALU	Set Up MAR for Next Instruction

Figure 4-4. $\overline{\text{LOAD}}$ Machine Cycle Sequence

4.3 MASKABLE INTERRUPTS.

The SBP 9900A has 16 hardware interrupt levels with the lower 15 priority levels used for maskable interrupts. The maskable interrupts are prioritized and have transfer vectors similar to those associated with the $\overline{\text{RESET}}$ and $\overline{\text{LOAD}}$ signal-commands.

4.3.1 INTERRUPT SERVICE. In general, a pending interrupt with an unmasked priority level is sensed by the SBP 9900A during the final machine cycle of the executing instruction. However, interrupt sensing is inhibited during the execution of a Branch and Load Workspace Pointer (BLWP) instruction; interrupt sensing is inhibited during the execution of the first instruction of a $\overline{\text{RESET}}$, $\overline{\text{LOAD}}$, XOP, or interrupt service routine. In either case, the one instruction delay in interrupt sensing permits a Load Interrupt Mask Immediate (LIMI) instruction to be executed thereby locking out higher priority maskable interrupts before an interrupt context switch can occur.

The pending interrupt request should remain active until recognized by the SBP 9900A during the service routine. The interrupt request should then be cleared under program control. The CRU bit manipulation instructions can be used to clear the interrupt requests.

The interrupt context switch causes the interrupt vectors to be fetched, the old WP, PC, and ST to be saved in the new workspace, and the new WP and PC to be loaded. Bits 12-15 of ST are loaded with a value of one less than the level of the interrupt being serviced. The old WP, PC, and ST are stored in the new workspace registers 13, 14, and 15. When the return instruction is executed, the old WP, PC, and ST are restored to the SBP 9900A. Since the ST contains the interrupt mask, the old interrupt level is also restored. Consequently, all interrupt service routines should be terminated with the Return Workspace Pointer (RTWP) instruction in order to restore the SBP 9900A to its ''before interrupt'' state.

The linkage between two interrupt service routines is shown in Figure 4-5 and the interrupt machine cycle sequence is shown in Figure 4-6.

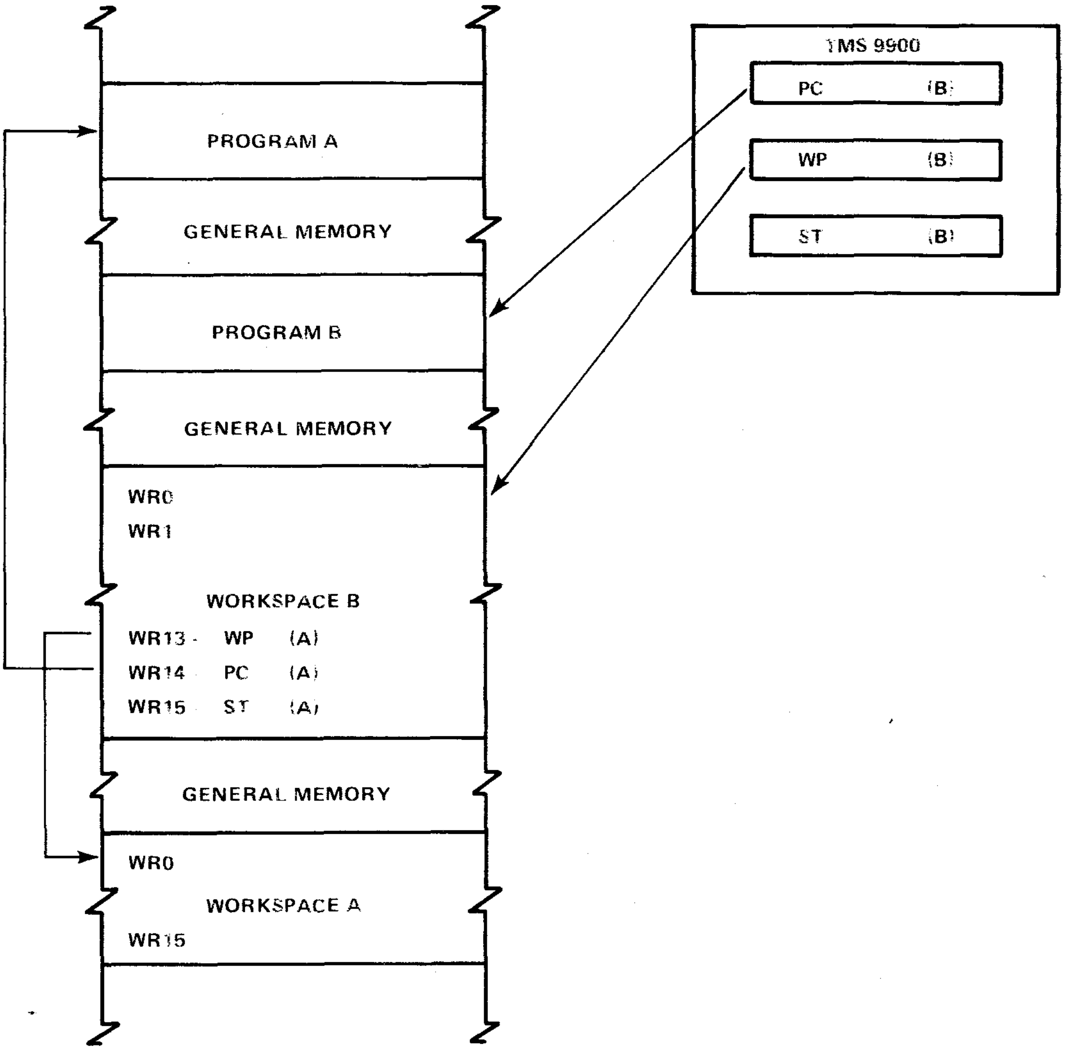


Figure 4-5. Interrupt Linkage

CYCLE	TYPE	FUNCTION
1	ALU	Set Up
2	Memory Read	Fetch New WP
3	ALU	Set Up
4	Memory Write	Store Status
5	ALU	Set Up
6	Memory Write	Store PC
7	ALU	Set Up
8	Memory Write	Store WP
9	ALU	Set Up
10	Memory Read	Fetch New PC
11	ALU	Set Up MAR for Next Instruction

Figure 4-6. Interrupt Processing Machine Cycle Sequence

4.3.2 INTERRUPT SIGNALS. The SBP 9900A has five inputs used for maskable interrupts, the $\overline{\text{INTREQ}}$ signal is active (low) when a maskable interrupt is pending. If $\overline{\text{INTREQ}}$ is active at the end of the instruction cycle, the SBP 9900A compares the priority code on IC0 through IC3 to the interrupt mask (ST12-ST15). If the interrupt code of the pending interrupt is equal to or less than the current interrupt mask, the SBP 9900A fetches the appropriate interrupt vectors from memory, otherwise the interrupt request is ignored. The interrupt priority codes are shown in Table 4-1. Note that the level 0 interrupt code should not be used for external interrupts since level 0 is reserved for $\overline{\text{RESET}}$.

Table 4-1. Interrupt Priority Codes

Interrupt Level		Vector Location (Memory Address In Hex)	Device Assignment	Interrupt Mask Values To Enable Respective Interrupts (ST12 thru ST15)	Interrupt Codes IC0 thru IC3
(Highest priority)	0	00	Reset	0 through F*	0000
	1	04	External device	1 through F	0001
	2	08		2 through F	0010
	3	0C		3 through F	0011
	4	10		4 through F	0100
	5	14		5 through F	0101
	6	18		6 through F	0110
	7	1C		7 through F	0111
	8	20		8 through F	1000
	9	24		9 through F	1001
	10	28		A through F	1010
	11	2C		B through F	1011
	12	30		C through F	1100
	13	34		D through F	1101
	14	38		E and F	1110
(Lowest priority)	15	3C	External device	F only	1111

*Level 0 can not be disabled.

Figure 4-7 shows the use of the interrupt control portion of the SBP 9961 for masking, prioritization, and generation of the interrupt code for 15 individual interrupt input lines. Note that the D-type synchronizers prevent transitions of IC0-IC3 while code is being read. A single interrupt system with an arbitrarily chosen level seven code is shown in Figure 4-8. The single input does not need to be synchronized since the hardwired interrupt code is always stable.

4.3.3 INTERRUPT MASKING. The SBP 9900A uses a four-bit field in the status register, ST12 through ST15, to determine the current interrupt priority level. The interrupt mask is automatically loaded with a value of one less than the level of the maskable interrupt being serviced. The interrupt mask is also affected by the Load Interrupt Mask Immediate instruction (LIMI).

Since the interrupt mask is compared to the external interrupt code before an interrupt is recognized, an interrupt service routine will not be halted due to another interrupt of lower or equal priority unless a LIMI instruction is used to alter the interrupt mask. The LIMI instruction can be used to alter the interrupt mask level in order to disable intervening interrupt levels. At the end of the service routine, a return (RTWP) restores the interrupt mask the value of which was active before the current interrupt occurred.

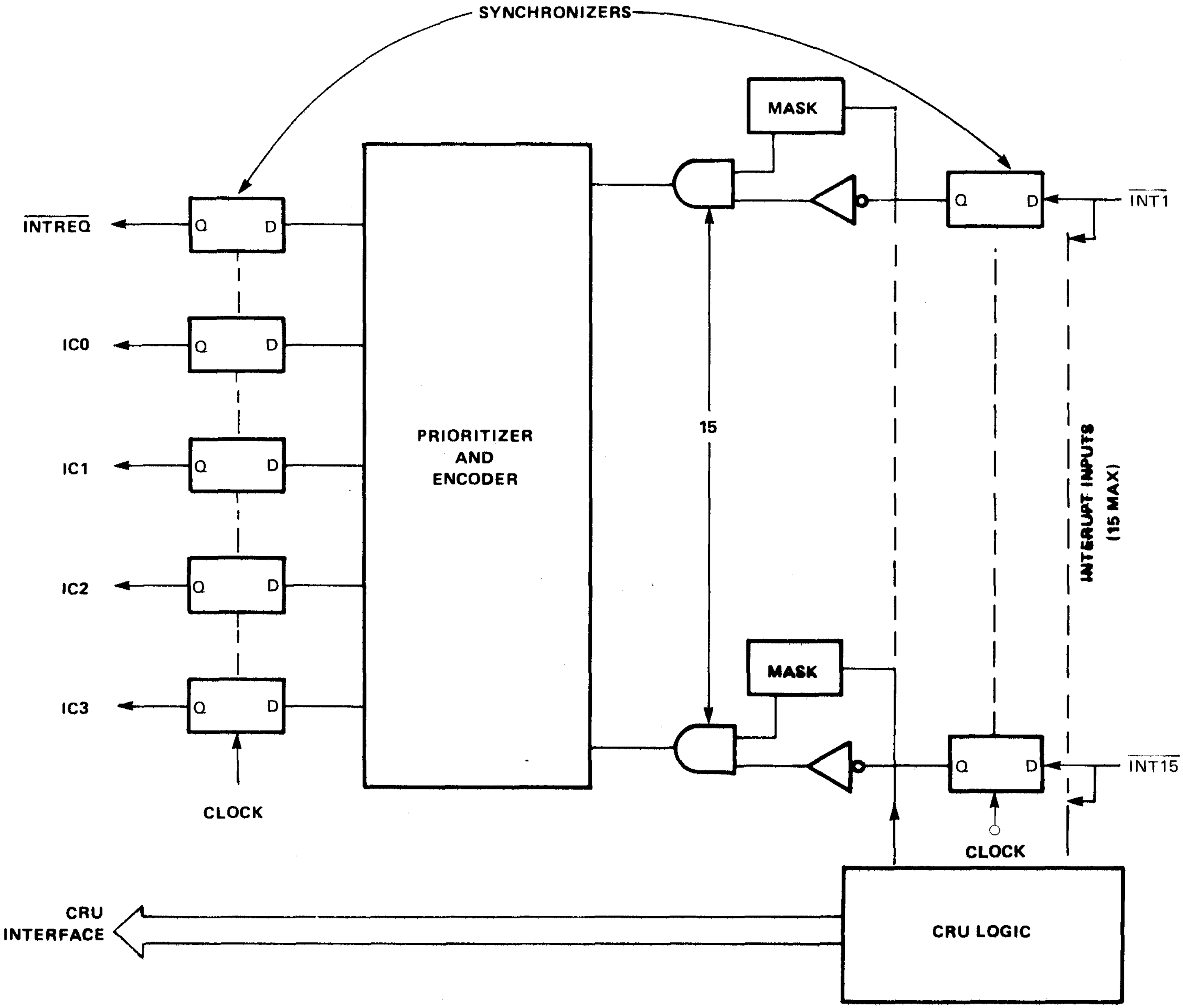


Figure 4-7. Interrupt Control Logic

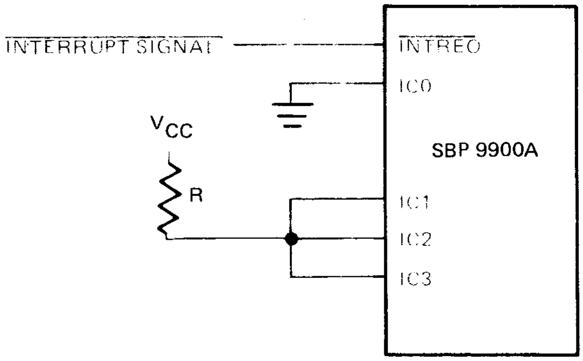


Figure 4-8. Single Interrupt System

Note that the SBP 9900A actually generates the interrupt vector addressing using IC0-IC3 several clock cycles after it has sampled INTREQ and compared the interrupt code to the interrupt mask in the status register. Therefore, interrupt sources which have individual masking capability can cause erroneous operation if a command to the device to mask the interrupt occurs at a time when the interrupt is active but after the SBP 9900A has sampled INTREQ and before the vector address has been generated using IC0-IC3.

The individual interrupt masking operation can be easily allowed if the masking instruction is placed in a short subroutine which masks all interrupts with a LIMI 0 instruction before individually masking the interrupt at the device.

4.3.4 INTERRUPT PROCESSING EXAMPLE. The routine in Figure 4-9 shows the use of the LIMI instruction as a privileged or non-interruptable instruction. The "level 5" routine sets a CRU bit and then loops until a corresponding CRU bit is true.

```
Level 5     LIMI     0        Disable Maskable INTREQs
            SB0      ACK      Set CRU Output Bit
Loop        TB       RDY      Test CRU Input Bit
            JNE      LOOP     Loop Until Input True
            RTWP              Return
```

Figure 4-9. LIMI Instruction

The first instruction in the routine is completed before a higher priority interrupt can be recognized. The LIMI instruction however, raises the SBP 9900A priority level to 0 in order to disable all other maskable interrupts. Consequently, the "level 5" routine will run to completion unless a RESET signal-command or a LOAD signal-command is generated. At the end of the routine, the RTWP instruction restores the CPU to its state before the level 5 interrupt occurred.

SECTION V
INPUT/OUTPUT

The SBP 9900A has three I/O modes: direct memory access (DMA), memory mapped, and Communications Register Unit (CRU). This multi-mode capability enables the designer to optimize an SBP 9900A I/O system to match a specific application. One or all modes can be used, as shown in Figure 5-1.

5.1 DIRECT MEMORY ACCESS

DMA is used for high speed block data transfer when CPU intervention is undesirable or not required. The DMA control circuitry can be relatively complex and expensive when compared to other I/O methods. A complete discussion of SBP 9900A based DMA is contained in Section 3.6

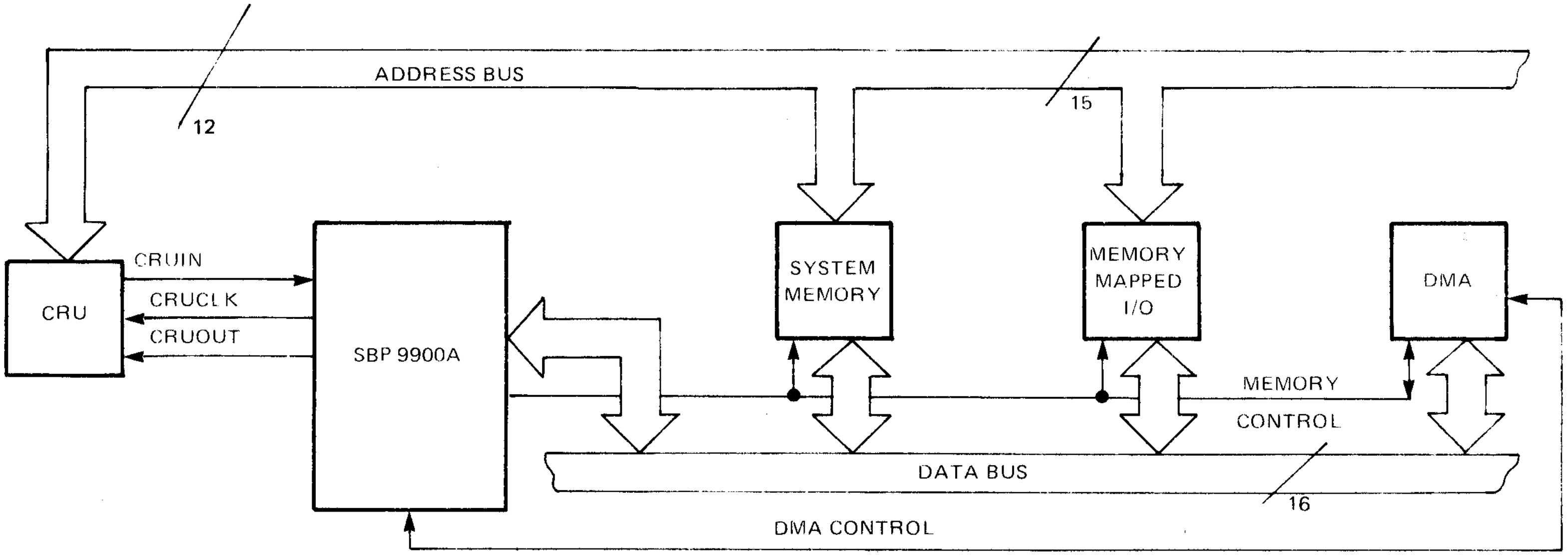
5.2 MEMORY MAPPED I/O

Memory mapped I/O permits peripheral interface devices to be addressed as memory locations. Since parallel data transfer is accomplished exactly like that of RAM memory, the peripheral interface device must ''look'' like a RAM memory location to the SBP 9900A. *Caution* should be exercised when using the MOV or MOVB instructions to load peripheral devices since these instructions perform a memory read cycle on the destination address before the write cycle is initiated on that destination address. Also, the interface logic at the peripheral device must assert the READY signal during the read memory cycle to prevent the SBP 9900A from entering an endless wait state. Figure 5-2 shows the SBP 9965 memory mapped peripheral interface with eight latched outputs and eight latched inputs. The SN74S330 Field Programmable Logic Array provides the address decode from address bits A3 through A14.

5.3 COMMUNICATIONS REGISTER UNIT (CRU)

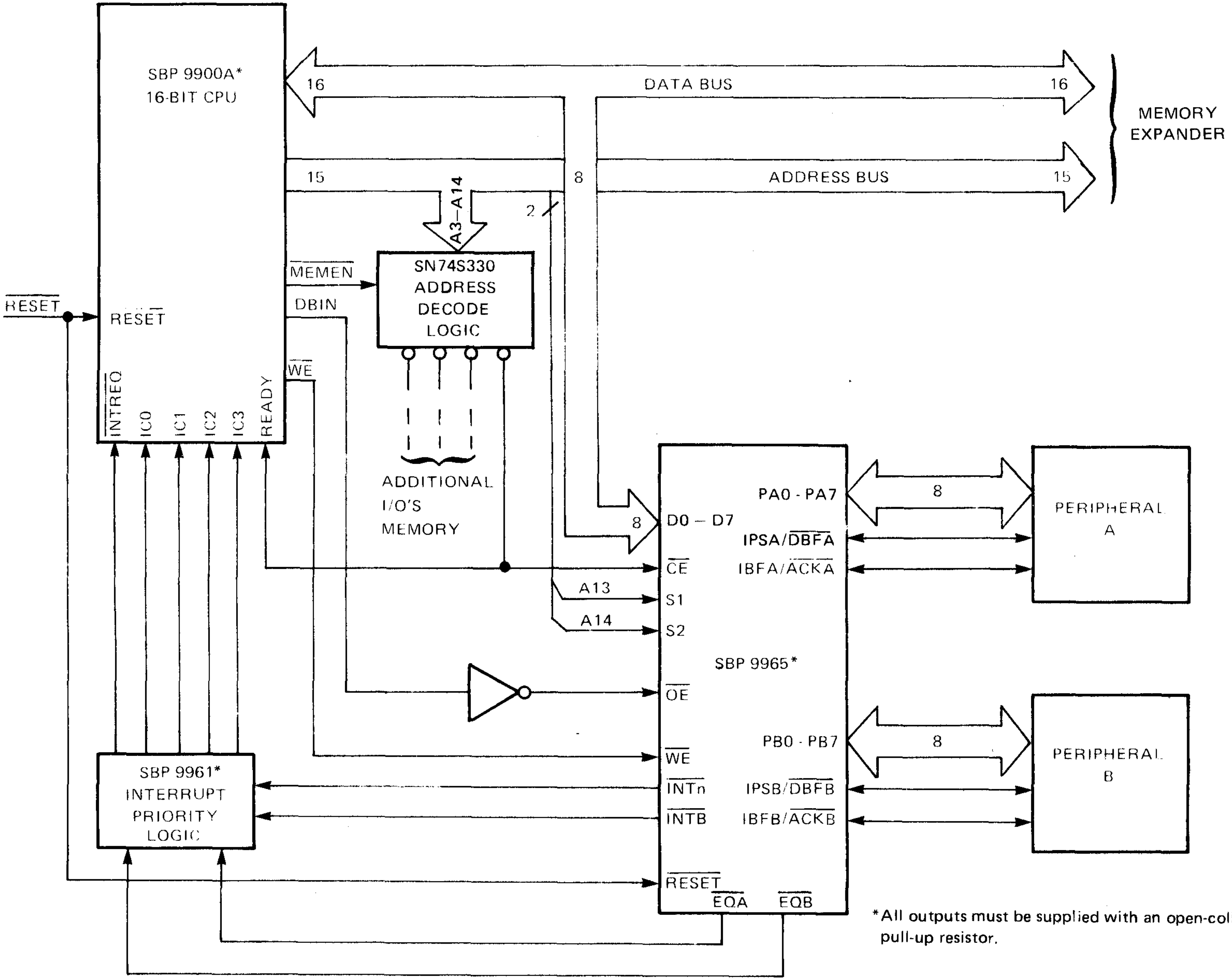
The SBP 9900A Communications Register Unit is a versatile, dedicated, program-driven, parallel-addressed/serial-data I/O interface. Up to 4096 input and 4096 output peripheral bits may be transferred in bit fields of one to sixteen. CRU interface circuitry requires fewer signals than a parallel memory interface; CRU I/O capabilities can be easily expanded independent of the memory system. In many applications, CRU I/O is superior to memory mapped I/O because of the powerful bit manipulation capability of the CRU, flexible field lengths, and simple bus structure. The CRU bit manipulation instructions eliminate the need for the bit masking instructions required by memory mapped I/O. The CRU multiple-bit instructions allow the use of I/O fields not identical to the memory word size thereby permitting optimum use of the I/O interface. Therefore, the CRU minimizes the size and complexity of the I/O control programs, while increasing system throughput.

The CRU does not utilize the data bus. The data bus is used only for system memory and memory mapped peripheral interfacing. This can reduce the complexity of printed circuit board layouts for most systems. The space saving 28-pin SBP 9960 CRU I/O device is possible as a result of the serial CRU bus which eliminates the need for several pins dedicated to a parallel data bus with multiple control lines. System costs are lower because of simplified circuit layouts, increased density, and lower component costs.



- COMMUNICATIONS REGISTER UNIT   CRU
- MEMORY MAPPED I/O
- DIRECT MEMORY ACCESS - DMA

Figure 5-1. SBP 9900A I/O Capability



*All outputs must be supplied with an open-collector pull-up resistor.

Figure 5-2. SBP 9965 Memory Mapped I/O Interface

5.3.1 CRU INTERFACE. The interface between the SBP 9900A and CRU-based peripheral devices consists of A0-A14, CRUIN, CRUOUT, and CRUCLK as shown in Figure 5-1. A0-A2 indicate whether data is to be transferred or if one of five external control instructions (IDLE, RSET, CKOF, CKON, LREX) is being executed. A3-A14 contain the address of the selected bit for data transfer. Therefore, up to 2^{12} or 4,096 bits of input and 4,096 bits of output may be individually addressed. CRU operations and memory data transfers both use A0-A14; however, these operations are performed independently with $\overline{\text{MEMEN}}$ inactive during CRU operations. Therefore no conflict arises.

5.3.2 CRU MACHINE CYCLES. Each CRU operation consists of one or more CRU output or CRU input machine cycles. Each CRU cycle is two clock periods in length. As shown in Table 5-1, five instructions (LDCR, STCR, SBO, SBZ, TB) transfer data to or from the SBP 9900A by CRU machine cycles, and five External Control instructions (IDLE, RSET, CKOF, CKON, LREX) generate control signals through CRU output machine cycles.

Table 5-1. Instructions Generating CRU Cycles

Instruction	Number of CRU Cycles	Type of CRU Cycles	A0—A2	Data Transfer
LDCR	1-16	Output	0 0 0	Yes
STCR	1-16	Input	0 0 0	Yes
SRO	1	Output	0 0 0	Yes
SBZ	1	Output	0 0 0	Yes
TB	1	Input	0 0 0	Yes
IDLE	1	Output	0 1 0	No
RSET	1	Output	0 1 1	No
CKOF	1	Output	1 0 1	No
CKON	1	Output	1 1 0	No
LREX	1	Output	1 1 1	No

5.3.2.1 CRU Output Machine Cycles. Figure 5-3 shows the timing for CRU output machine cycles. Address (A0-A14) and data (CRUOUT) are output during the first clock period of the CRU cycle. During the second clock period, the SBP 9900A outputs a pulse on CRUCLK indicating to the peripheral CRU device the presence of a data bit. Therefore, CRUCLK can be used as a strobe since address and data are stable during the pulse. Referring again to Table 5-1, it is important to note that output data transfers occur only when A0-A2 = 000, otherwise no data transfer occurs and an External Control instruction is being executed. In that case A0-A2 should be decoded to determine the nature of the External Control instruction. The generation of control strobes for external instructions and a data transfer strobe (OUTCLK) is shown in Figure 5-4. If none of the External Control instructions are used, A0-A2 need not be decoded for data transfer, since they will always equal 000.

5.3.2.2 CRU Input Machine Cycles. The timing for CRU input machine cycles is shown in Figure 5-5. The address is output during the first clock period of 'he CRU cycle. The CRUIN data input is sampled during the low-to-high transition of the clock which terminates the CRU cycle. Thus, CRU input is accomplished by simply multiplexing the addressed bit onto the SBP 9900A CRUIN input. A0-A2 will always be 000 and may be ignored. CRU input machine cycles cannot be differentiated from ALU cycles by external logic, therefore no operations (such as clearing interrupts) other than CRU input should be performed during CRU input machine cycles.

5.3.3 CRU DATA TRANSFER. When a CRU data transfer is performed, the CRU base address is defined by bits 3-14 of the current WR12. Bits 0-2 and bit 15 of WR12 are ignored for CRU address derivation.

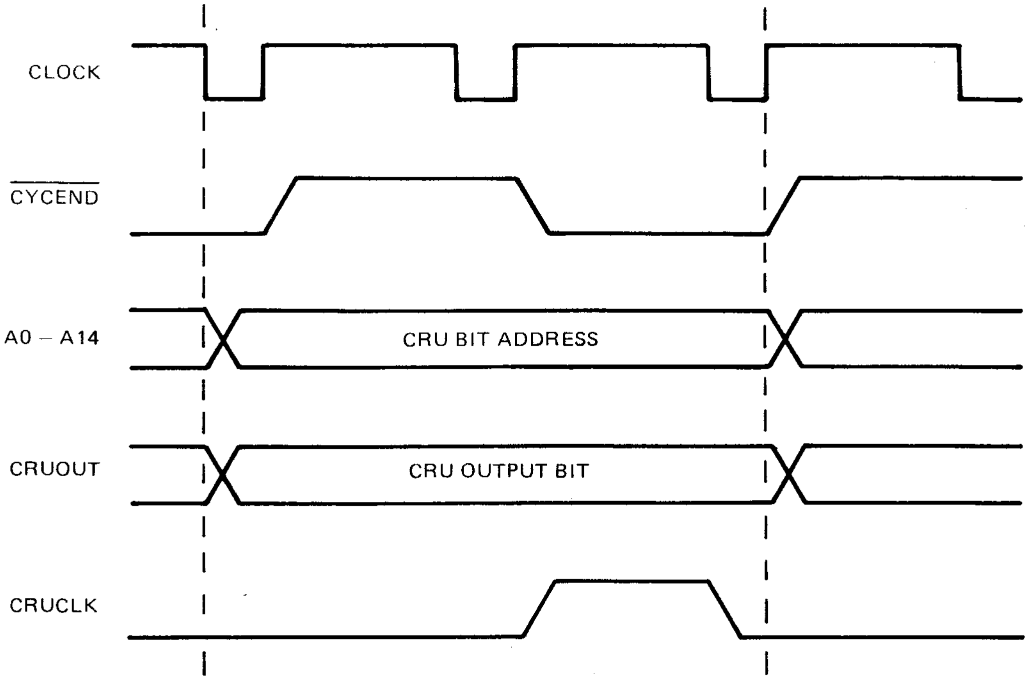


Figure 5-3. CRU Output Machine Cycle Timing

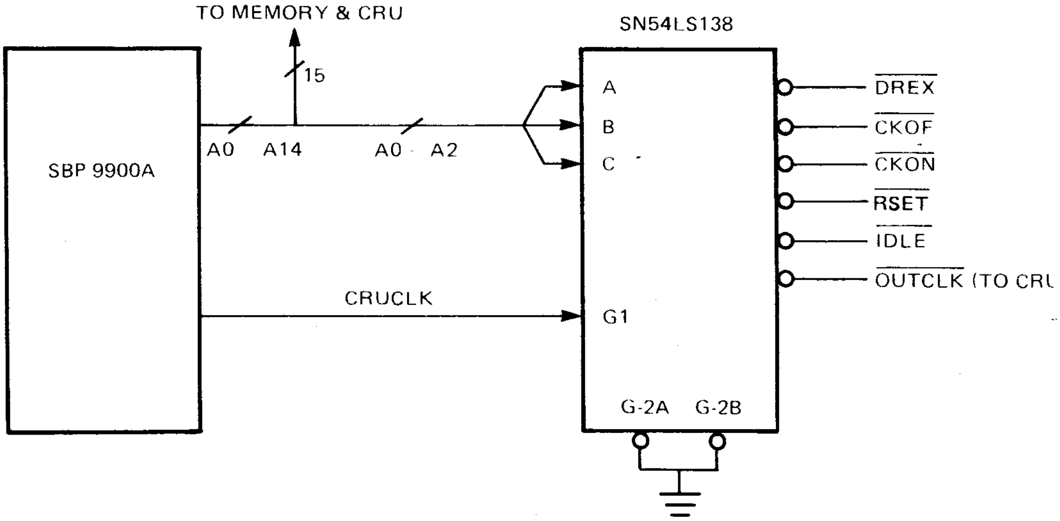


Figure 5-4. CRU Control Strobe Generation

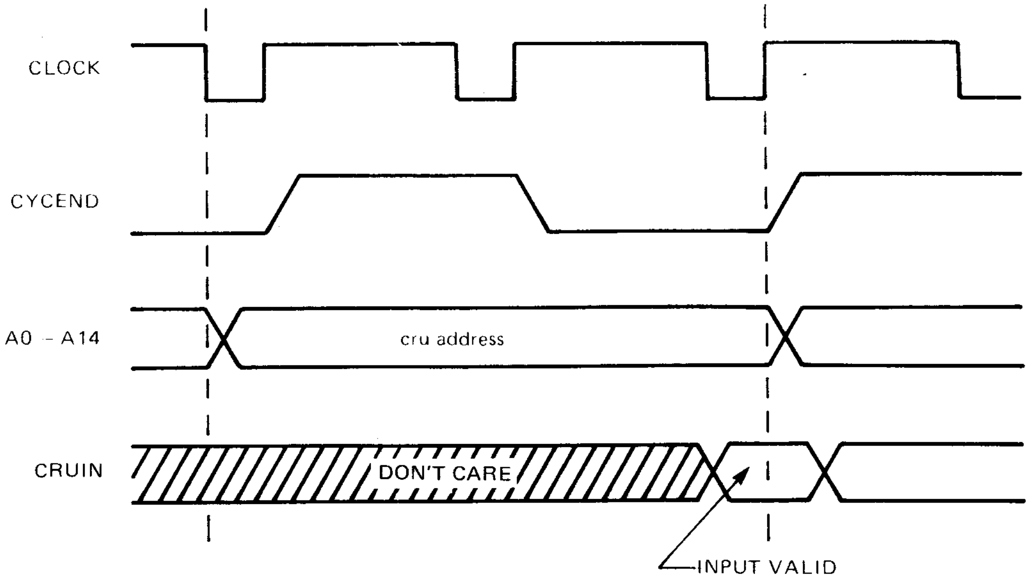


Figure 5-5. CRU Input Machine Cycle Timing

5.3.3.1 Single Bit Instructions. For single bit CRU instructions (SBO, SBZ, TB), the address of the CRU bit to or from which data is transferred is determined as shown in Figure 5-6. Bits 8-15 of the instruction contain a signed displacement. This signed displacement is added to the CRU base address (bits 3-14 of WR12). The result of the addition is output on A3-A14 during the CRU output or CRU input machine cycle.

For example, assume the instruction "SBO 9" is executed when WR12 contains a value of 1040_{16}. The machine code for "SBO 9" is $1D09_{16}$ and the signed displacement is 0009_{16}. The CRU base address is 0820_{16} (bits 0-2 and bit 15 of WR12 are ignored). Thus, the effective CRU address is $(0820)_{16}$ + $(0009)_{16}$ = $(0829)_{16}$. This address is output on A0-A14 during the CRU output machine cycle.

As a second example, assume that the instruction "TB-32" is executed when WR12 = 0100_{16}. The effective CRU address is 80_{16} (CRU Base) + $FFF0_{16}$ (binary two's complement of displacement 32_{10}) = 60_{16}. Thus, the "TB-32" instruction in this example causes the value of the CRU input bit at address 60_{16} to be transferred to bit two of the status register. This value may then be operated on by the JEQ or JNE instructions, which cause the PC to be changed depending on the value of ST2.

5.3.3.2 LDCR Instruction. The LDCR instruction may transfer from 1 to 16 bits of data from memory to the CRU periphery. The output of each bit is performed by a CRU output machine cycle. Thus, the number of CRU output machine cycles performed by a LDCR instruction is equal to the number of bits to be transferred.

As an example, assume that the instruction "LDCR 600,10" is executed and that WR12 =0800_{16} with the memory word at address 600 containing the bit pattern shown in Figure 5-7. In the first CRU output machine cycle, A0-A14 = 400_{16} and CRUOUT = "a". With each successive machine cycle, the address is incremented by one and the next least significant bit of the operand is output on CRUOUT until output of ten bits has been accomplished. It is important to note that the CRU base address is unaltered by the LDCR instruction even though the effective address is incremented as each successive bit is output.

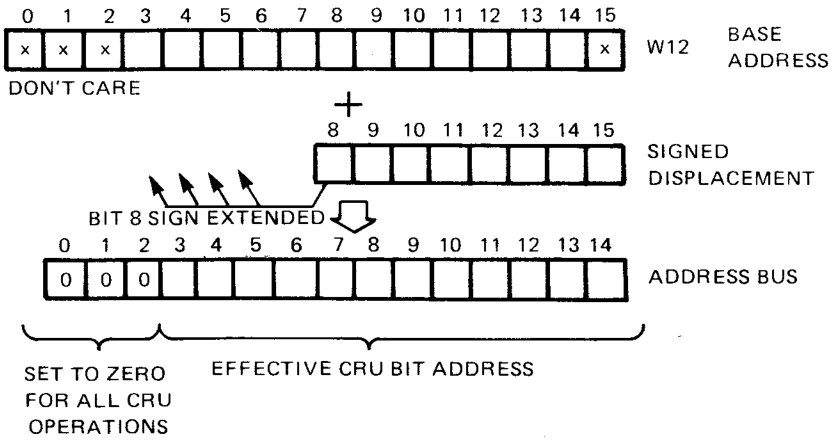


Figure 5-6. SBP 9900A Single-Bit CRU Address Development

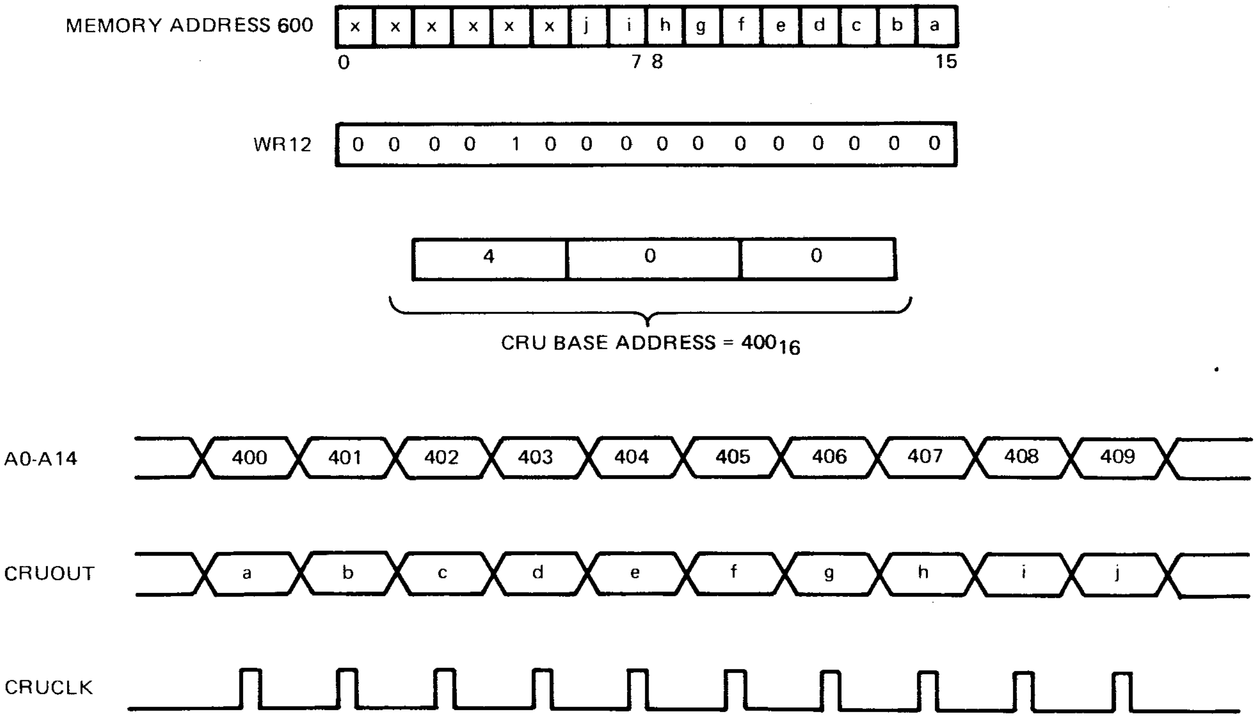


Figure 5-7. Multiple-Bit CRU Output

As a more specific example of the above, assume that address location 600 contains the value 02F8$_{16}$. The output would be as shown in Figure 5-8.

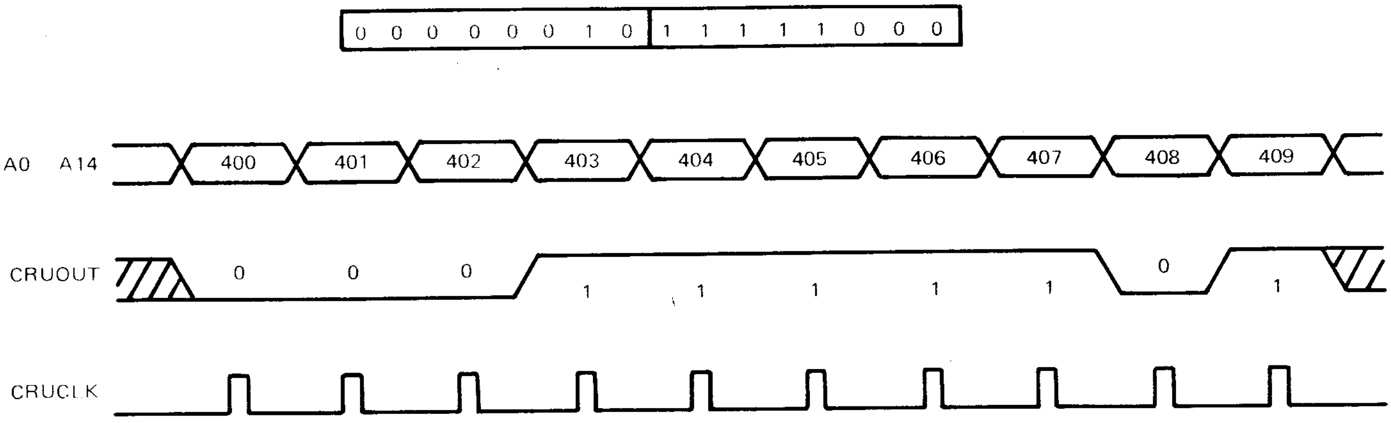


Figure 5-8. Output of O2F816

5.3.3.3 STCR Instruction. The STCR instruction causes from 1 to 16 bits of CRU data to be transferred from the CRU periphery into memory. Each bit is input by the CRU input machine cycle.

Consider the example shown in Figure 5-9. The SBP 9960's CRU interface logic multiplexes input signals "M" through "T" onto the CRUIN line for addresses 200$_{16}$ — 207$_{16}$. If WR12 = 400$_{16}$ when the instruction "STCR : 602,6" is executed, the operation is performed as shown in Figure 5-10. At the end of the instruction, the six LSB's of memory byte 602 are loaded with "M" through "R". The upper bits of the operand are forced to zero. The generation of SEL200 may be simplified if the total CRU address space is not used.

5.3.4 CRU INTERFACE LOGIC. The CRU based I/O interface is easily implemented using the SBP 9960 and/or SBP 9961. These I/O circuits can be cascaded with the addition of simple address decoding logic.

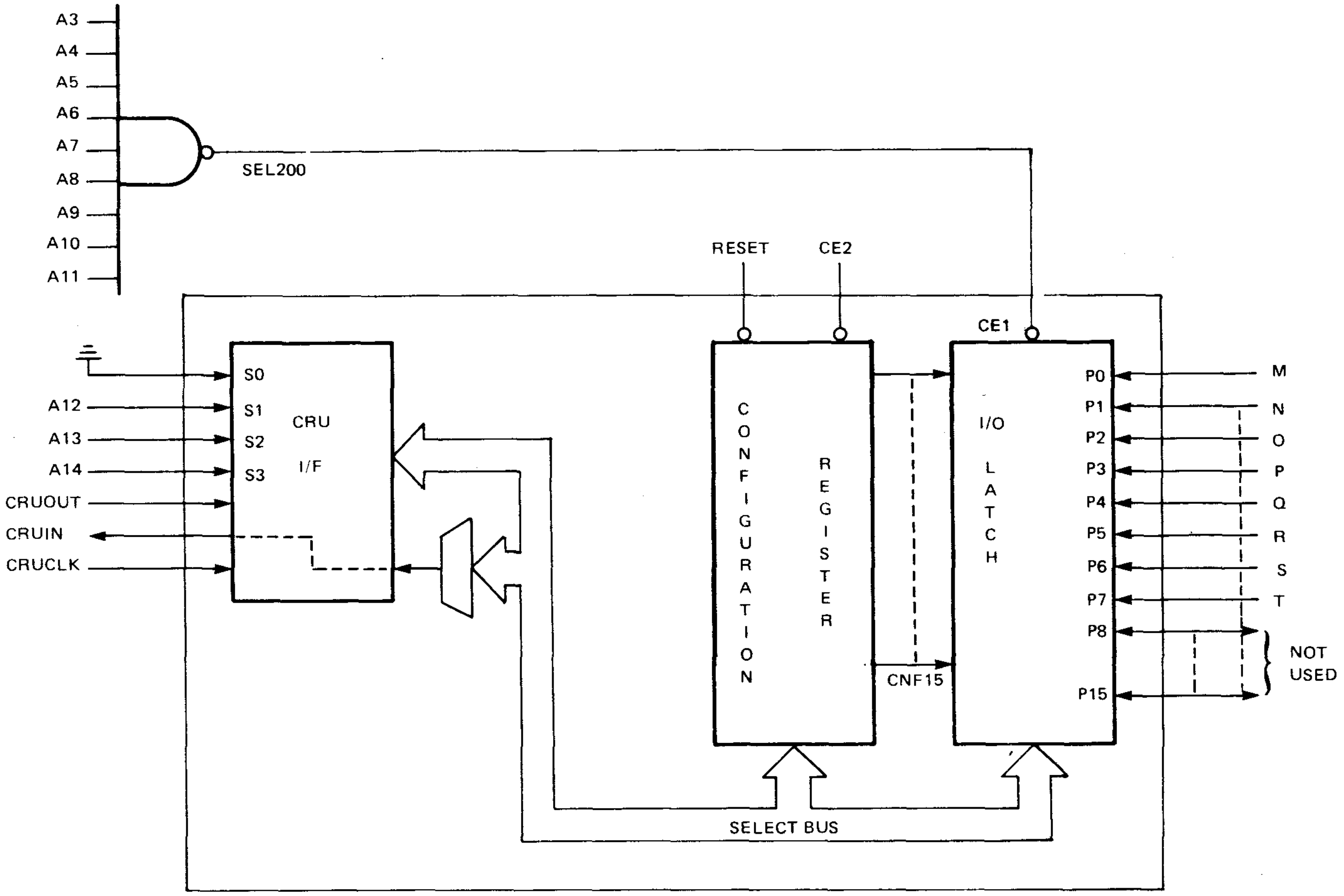


Figure 5-9. CRU Input Example

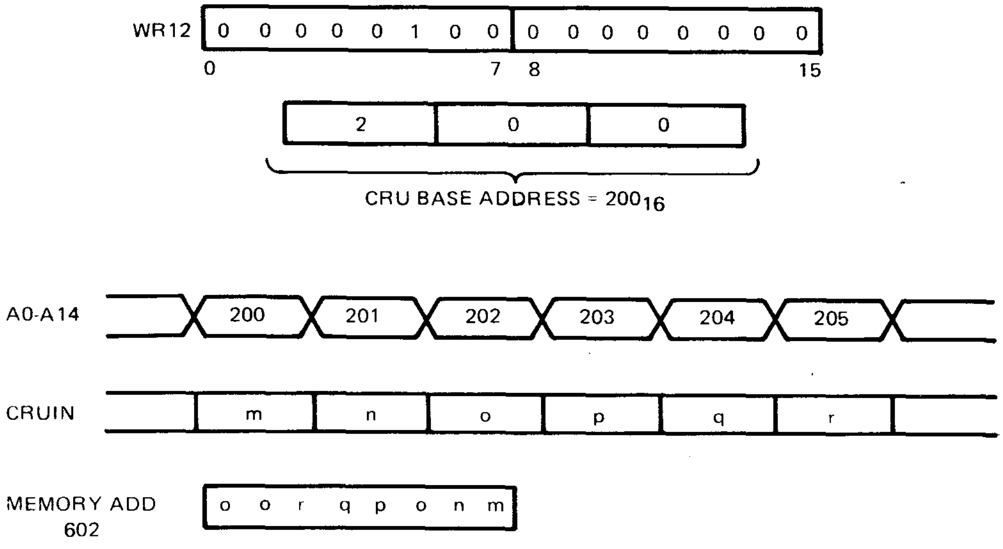


Figure 5-10. Multiple-Bit CRU Input

5.4 I2L CRU-BASED PERIPHERAL DEVICES

5.4.1 SBP 9960 PROGRAMMABLE CRU I/O EXPANDER

5.4.1.1 Introduction.

5.4.1.1.1 General Description. The SBP 9960 Programmable CRU I/O Expander is a ruggedized monolithic software-configurable input/output device fabricated with oxide separated Integrated Injection Logic (I^2L) technology. The SBP 9960 provides a flexible and efficient Communications Register Unit (CRU) based interface between the SBP/TMS 9900 series Family of Microprocessors and auxiliary systems functions ranging from bit-oriented sensors and actuators to byte/word/n-bit-field oriented peripherals.

Under software control, each of the SBP 9960s sixteen single-bit I/O ports may be individually configured to either the input or output mode. I^2L technology enables the SBP 9960s static logic, and TTL compatible I/O, to operate over a wide ambient temperature range from a single d-c power source with output current sink capability up to 40 mA. When the SBP 9960 is used in conjunction with the SBP 9961 I^2L Interrupt-Controller/Timer, the SBP 9960/ SBP 9961 pair form an I^2L systems alternate to the N-channel MOS TMS 9901 Programmable Systems Interface device while maintaining strict compatibility with existing software handlers developed in support of the TMS 9901.

5.4.1.1.2 Key Features.

- SBP/TMS 9900 Series Microprocessor Family Peripheral
- 16 Individual, Single-Bit, Software Compatible I/O Ports
- 20/40 mA Current Sinking Outputs
- 28-Pin Package
- Software Compatible with TMS 9901 when used in Conjunction with SBP 9961
- TTL Compatible I/O
- Wide Ambient Temperature Operation
  - — SBP 9960CJ: 0°C to +70°C
  - — SBP 9960EJ: −40°C to +85°C
  - — SBP 9960 MJ: −55°C to +125°C
  - — SBP 9960 NJ: −55°C to +125°C (with high-reliability processing)

- I^{2}L Technology
  - — Constant Current Power Source
  - — Fully Static Operation
  - — Single Phase Edge-Triggering Clock
  - — Wide Temperature Stability

5.4.1.2 Functional Description.

5.4.1.2.1 SBP 9960/CRU Interface. The SBP 9960 communicates with the CPU through the Communications Register Unit (CRU) interface as shown in Figures 5-11 and 5-13. The SBP 9960s CRU interface consists of:

a) five CRU address select lines (S0-S4)

b) a single chip enable ($\overline{CE}$)

c) a 9960 to CPU serial data-bit line (CRUIN)

d) a CPU to 9960 serial data-bit line (CRUOUT)

e) a CPU to 9960 serial data-bit clock (CRUCLK)

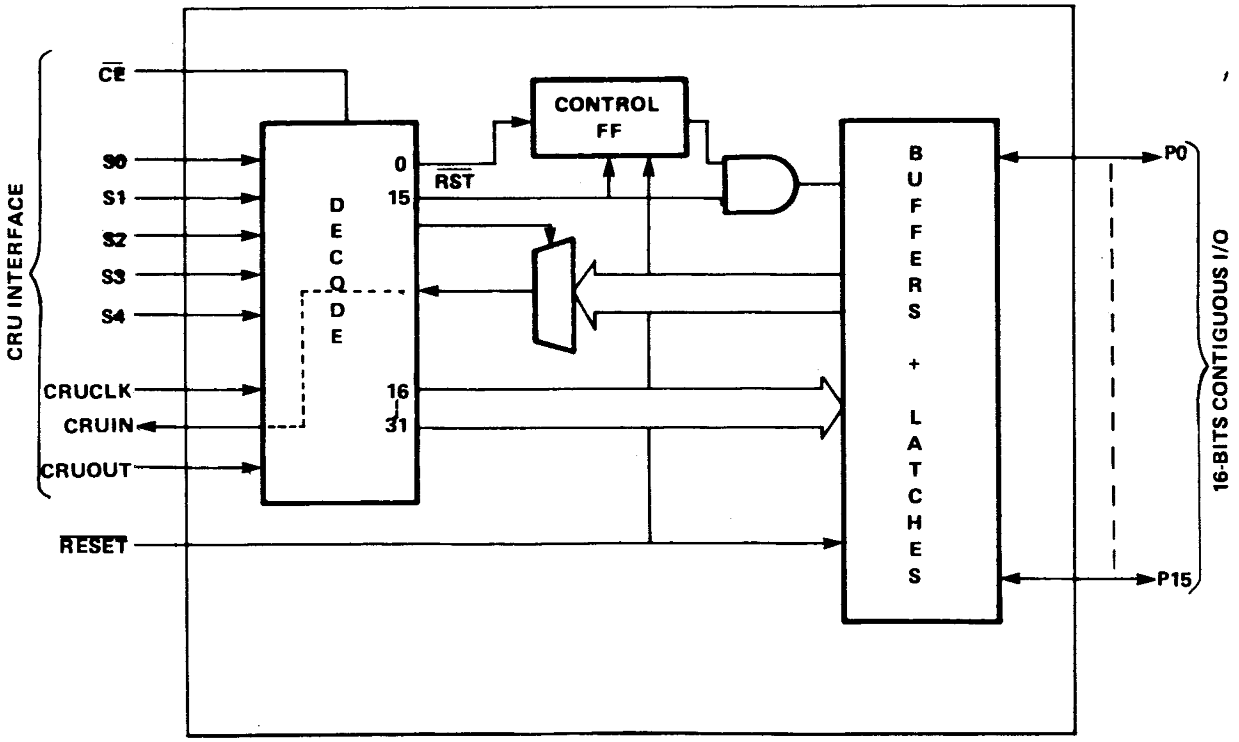


Figure 5-11. CRU Interface Circuitry

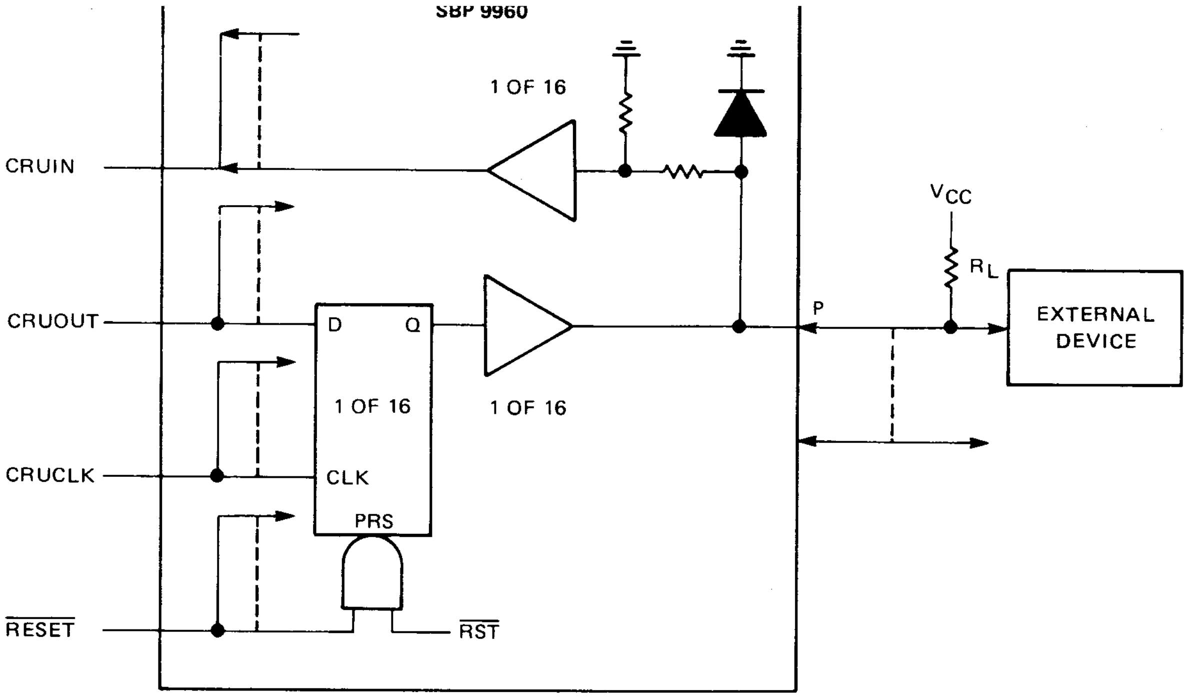


Figure 5-12. 1 of 16 Single Bit I/O Ports

When $\overline{CE}$ is activated (logic-level low), S0-S4 select a specific single-bit I/O port as indicated in Table 5-2. In the case of an SBP 9960 write operation, the datum is transferred from the CPU to the SBP 9960 via the CRUOUT line. The CRUOUT datum is strobed into the selected single-bit port by CRUCLK. In the case of a SBP 9960 read operation, the selected single-bit port is sampled by the CPU via the CRUIN line.

5.4.1.2.2 ***CRU Bit Assignments.*** Table 5-2 describes the SBP 9960s CRU bit assignments. Note that CRU bits 1-14 have been reserved for the SBP 9961 thereby insuring software compatibility between the SBP 9960/SBP 9961 pair and the TMS 9901.

5.4.1.2.3 ***Input/Output.*** One of sixteen SBP 9960, single-bit, combination open-collector-output/resistor-divider-input I/O ports is conversationally represented in Figure 5-12. As a direct result of the open-collector output structure, the direction of data flow through the port is determined by the stored logic-level of the associated output-register bit in combination with the data flow direction of the external device serviced by the port. When the output-register bit (Q) is at logic-level high, the corresponding package pin (P) is essentially floating and therefore free to be externally pulled to either the high or low logic level. In other words, when Q is at logic-level high, the ports data flow direction can be either inward, where an external device pulls P to the high or low logic-level; or the data flow direction can be outward, where an external resistor (R_L) both pulls P to logic-level high and sources current drive into the inputs of external devices. When Q is at logic-level low, the ports unconditional data flow direction is outward, where P has the capacity to sink 20/40* MAz of current from external devices. Q can be read to logic-level low through CPU execution of a SET BIT TO ZERO (SBZ) instruction; Q can be set to logic-level high through: 1) a hardware initiated reset ($\overline{\text{RESET}}$), 2) a software initiated reset ($\overline{\text{RST}}$: CRU BIT 15) preceded by setting the control bit (CRU BIT 0) to a logic-level high, or 3) CPU execution of a SET BIT TO ONE (SBO) instruction. Note that both $\overline{\text{RESET}}$ and $\overline{\text{RST}}$ affect all sixteen single-bit I/O ports while CPU execution of either an SBO or SBZ instruction can be targeted at an individual single-bit port independent of uninvolved ports. Once the data flow direction has been established for each single-bit port, CPU communication with the external devices driven or sensed by each individual port is affected through execution of the CRU instructions: LDCR, STCR, SBO, SBZ, and TB.

*Outputs Po, P1, P2, and P3 have extended current sink capability to 40 mA.

Table 5-2. SBP 9960 CRU Bit Assignments

CRU BIT	S0	S1	S2	S3	S4	CRU READ DATA	CRU WRITE DATA
0	0	0	0	0	0	Control Bit	Control Bit
1-14						Note 1	Note 1
15	0	1	1	1	1	"1"	No Operation/ $\overline{\text{RST}}$ [2]
16	1	0	0	0	0	P0 Input[3]	P0 Output[4] (5)
17	1	0	0	0	1	P1 ↑	P1 ↑ (5)
18	1	0	0	1	0	P2	P2 (5)
19	1	0	0	1	1	P3	P3 (5)
20	1	0	1	0	0	P4	P4
21	1	0	1	0	1	P5	P5
22	1	0	1	1	0	P6	P6
23	1	0	1	1	1	P7	P7
24	1	1	0	0	0	P8	P8
25	1	1	0	0	1	P9	P9
26	1	1	0	1	0	P10	P10
27	1	1	0	1	1	P11	P11
28	1	1	1	0	0	P12	P12
29	1	1	1	0	1	P13	P13
30	1	1	1	1	0	P14 ↓	P14 ↓
31	1	1	1	1	1	P15 Input[3]	P15 Output[4]

NOTES: (1) Bits 1-14 reserved for SBP 9961 Interval Timer/Interrupt Controller

(2) Writing a zero to bit 15 while CONTROL = 1 executes a software reset of the I/O ports.

(3) Data present on the port will be read without affecting the data.

(4) Writing data to the port will both program the port to the output mode and output the data.

(5) These outputs are provided with extended sink-current capability to 40 mA.

5.4.1.2.4 System Operation. During a typical power-up sequence of an SBP 9960-based system, $\overline{\text{RESET}}$ should be activated (logic-level low) to force the SBP 9960 to the state where each of the sixteen individual single-bit I/O ports is in the input mode. System software should then configure each single-bit port as required. If a given port must be reconfigured from the input to output mode after power-up, the associated output-register bit must be set to logic-level high through CPU execution of an SBO instruction.

*5.4.1.2.5 SBP 9960/SBP 9961 Emulation of the TMS 9901.*Figure 5-13 shows the system configuration of an SBP 9960 functioning in conjunction with an SBP 9961 in emulation of a TMS 9901. Note the common connection of: a) the individual chip enables, and b) the CRU interface lines. For a complete description of the SBP 9961 and the TMS 9901 refer respectiely to the *SBP 9961 Interrupt-Controller/Timer Data Manual* and the *TMS 9901 Programmable Systems Interface Data Manual.*

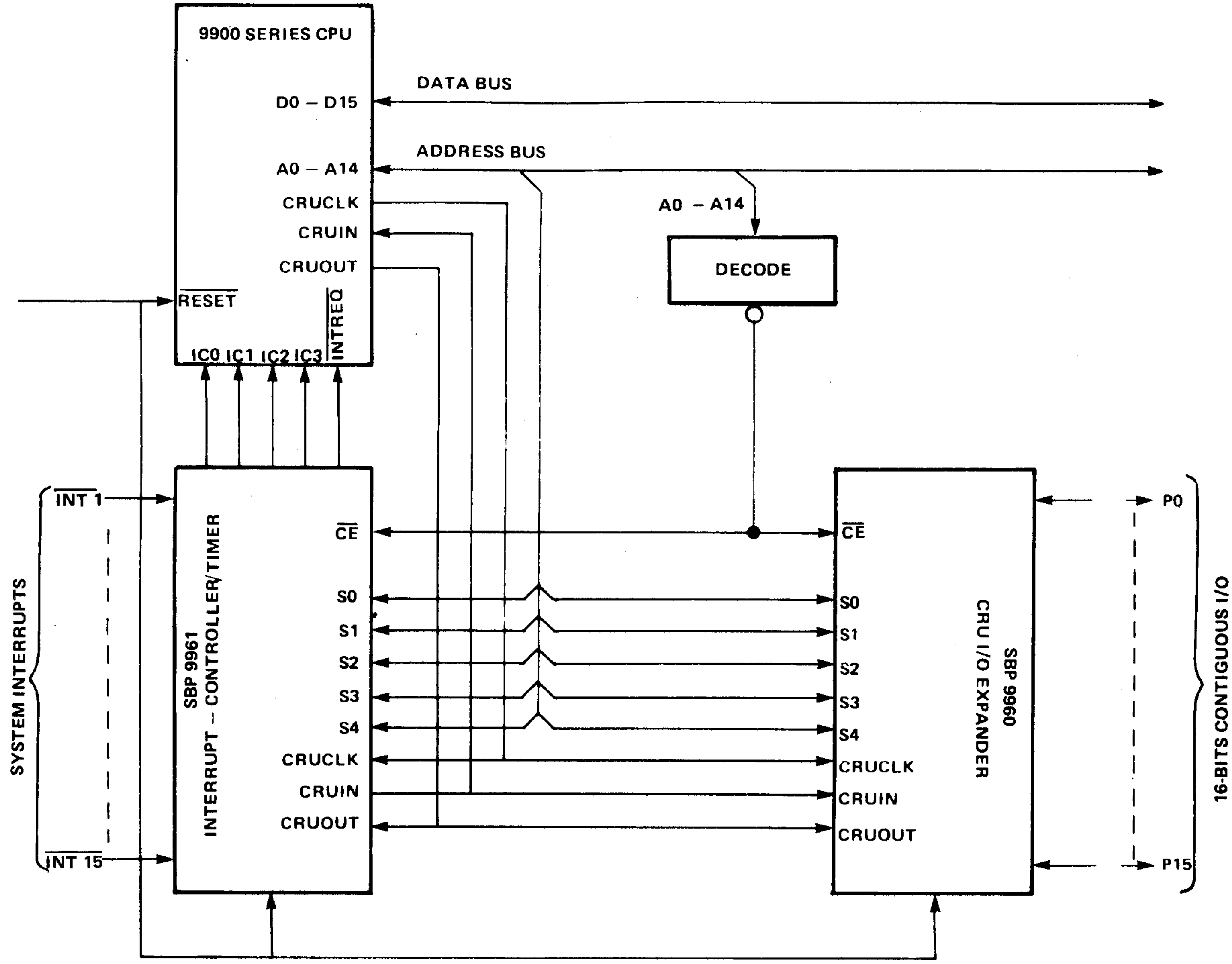
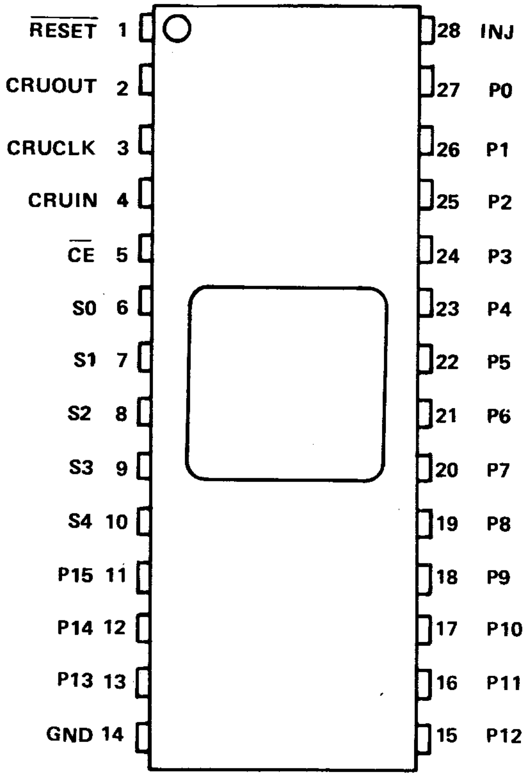


Figure 5-13. SBP 9960/SBP 9961 System Configuration

5.4.1.2.6 SBP 9960 Pin Assignments and Functions

SIGNATURE	PIN	I/O	DESCRIPTION
S0	6	IN	ADDRESS SELECT LINES. The data bit being accessed by the CRU interface is specified by the 5-bit code appearing on S0-S4.
S1	7	IN	
S2	8	IN	
S3	9	IN	
S4	10	IN	
CRUIN	2	IN	CRU DATA IN (to CPU). Data specified by S0-S4 is transmitted to the CPU by CRUINI. When $\overline{\text{CE}}$ is not active, CRUIN is pulled to logic-level high.
CRUOUT	2	IN	CRU DATA OUT (from CPU). When $\overline{\text{CE}}$ is active data present on the CRUOUT input will be strobed by CRUCLK and written into the CRU bit specified by S0-S4.
CRUCLK	3	IN	CRU CLOCK (from CPU). CRUCLK specifies that alid data is present on the CRUOUT line.
$\overline{\text{RESET}}$	1	IN	POWER-UP RESET. When active (low), $\overline{\text{RESET}}$ forces all I/O's (P0-P15) to input mode.
$\overline{\text{CE}}$	5	IN	CHIP ENABLE. When active (low), data may be bidirectionally transferred between the SBP 9960 and the CPU.
INJ	28		Supply Current
GND	14		Ground Reference
P0	27	I/O	I/O pins
P1	26	I/O	
P2	25	I/O	
P3	24	I/O	
P4	23	I/O	
P5	22	I/O	
P6	21	I/O	
P7	20	I/O	
P8	19	I/O	
P9	18	I/O	
P10	17	I/O	
P11	16	I/O	
P12	15	I/O	
P13	15	I/O	
P14	12	I/O	
P15	11	I/O	



5.4.1.3 Electrical Specifications

5.4.1.3.1 Recommended Operating Conditions, Unless Otherwise Noted I_{CC} = 70 mA

PARAMETER		MIN	NOM	MAX	UNIT
Supply current, I_{CC}		63	70	77	mA
High-level output voltage, V_{OH}				5.5	V
Low-level output current, I_{OL}				20†	mA
Operating free-air temperature, T_A	SBP 9960MJ, SBP 9960NJ	–55		125	°C
	SBP 9960EJ	–40		85	
	SBP 9960CJ	0		70	

†40 mA on extended drive outputs P0, P1, P2, and P3

5.4.1.3.2 Electrical Characteristics (Over Recommended Operating Free-Air Temperature Range, Unless Otherwise Noted)

PARAMETER		TEST CONDITIONS†	MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage		2			V
V_{IL}	Low-level input voltage				0.8	V
V_{IK}	Input clamp voltage	I_{CC} = MIN, I_I = –12 mA			–1.5	V
I_{OH}	High-level output current	I_{CC} = 70 mA, V_{IH} = 2 V, V_{IL} = 0.8 V, V_{OH} = 5.5 V			400	μA
V_{OL}	Low-level output voltage	I_{CC} = 70 mA, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OL} = 20 mA (40 mA§)			0.5	V
I_I	Input current	I_{CC} = 70 mA, V_I = 2.4 V		225		μA

†For conditions shown as MAX, use the appropriate value specified under recommended operating conditions.

‡All typical values are at I_{CC} = 70 mA, T_A = 25°C.

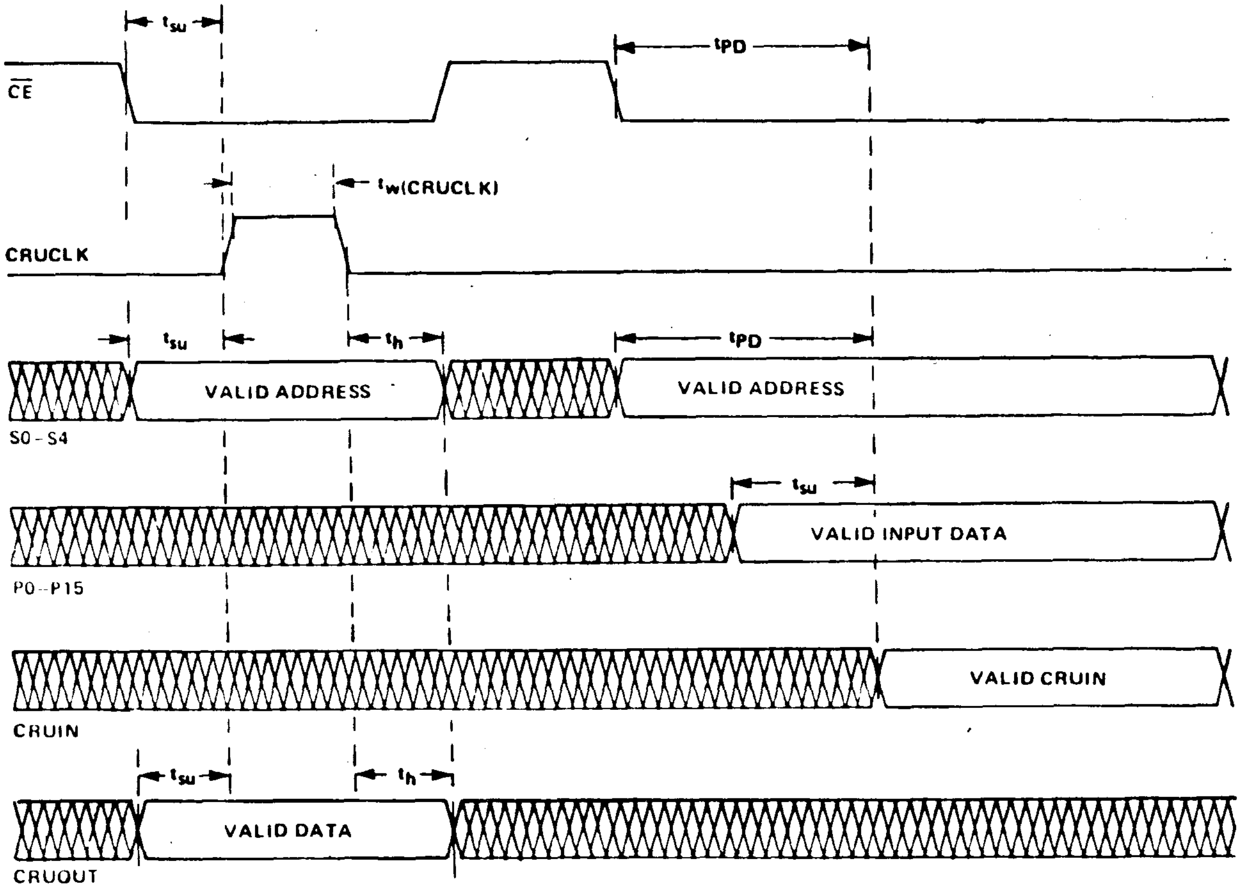
§Extended drive outputs only.

5.4.1.3.3 Timing Requirements Over Full Range of Operating Conditions

PARAMETER		MIN	NOM	MAX	UNIT
t_{su}	Setup time for S0-S4, $\overline{CE}$, or CRUOUT before CRUCLK		200		ns
t_{su}	Setup time, input before valid CRUIN		200		ns
$t_{w(CRUCLK)}$	CRU clock pulse width		100		ns
t_h	Hold time for Address or Data		0		ns

5.4.1.3.4 Switching Characteristics over Full Range of Recommended Operating Conditions

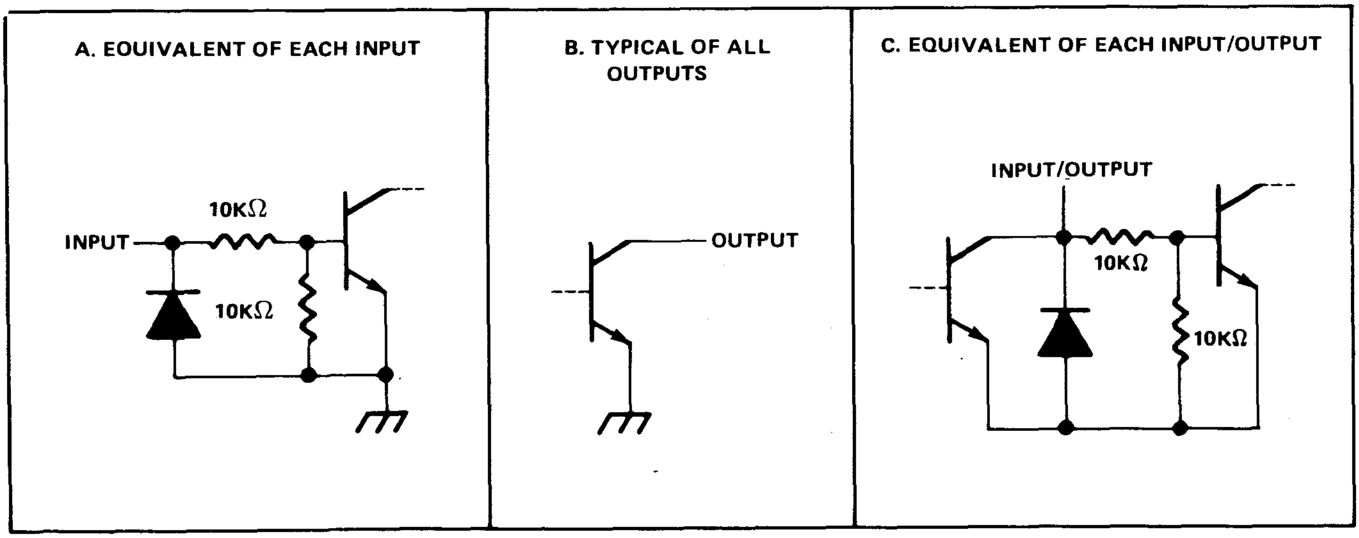
PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PD}	Propagation delay, S0-S4 or $\overline{CE}$ to valid CRUIN	C_L = 100 pF, R_L = 300 Ω		300		ns



NOTE 1: ALL TIMING MEASUREMENTS ARE FROM 10% and 90% POINTS

SWITCHING CHARACTERISTICS

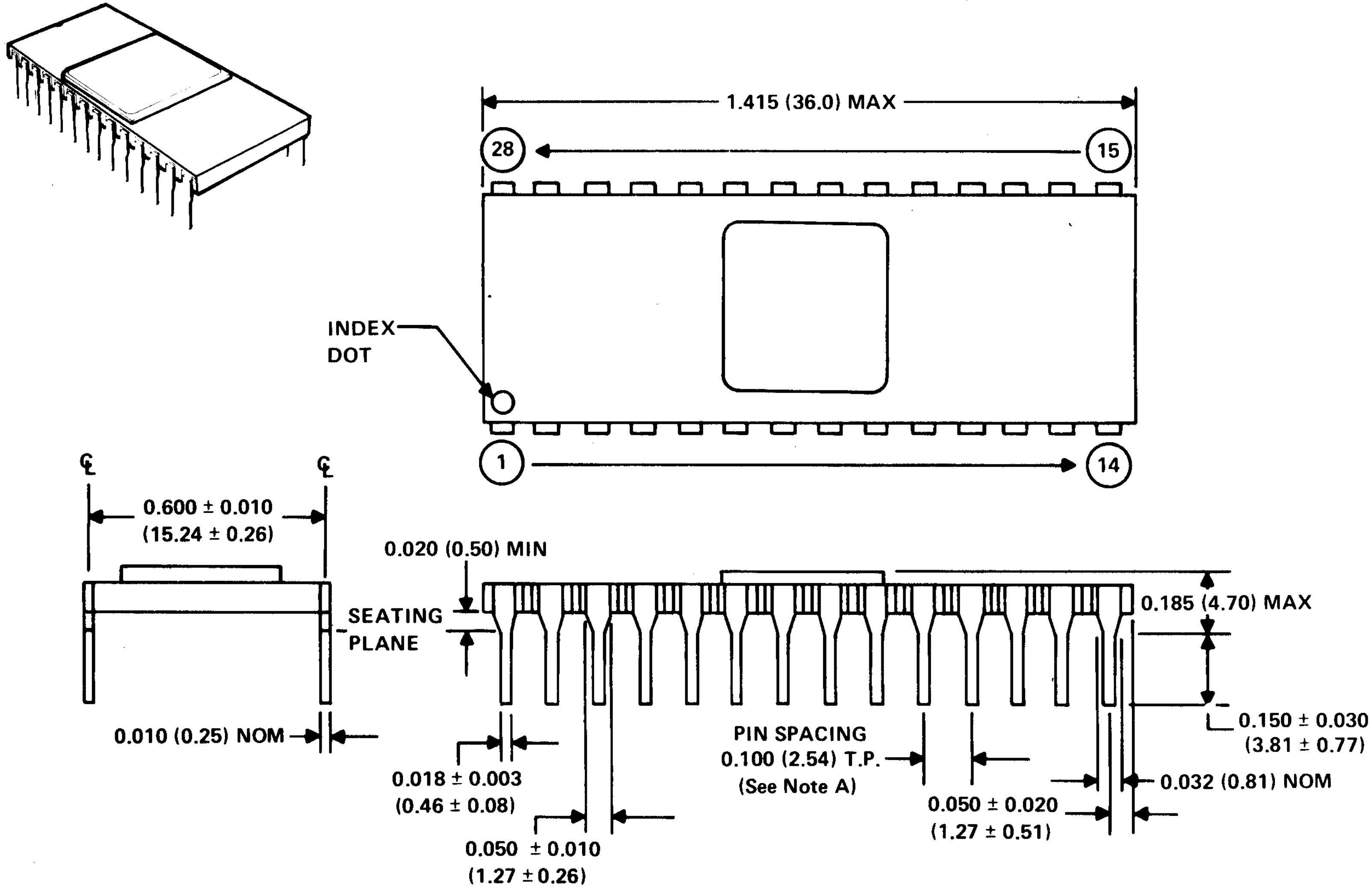
5.4.1.4 Input, Output, and Input/Output Structures


A. EOUIVALENT OF EACH INPUT

B. TYPICAL OF ALL OUTPUTS

C. EQUIVALENT OF EACH INPUT/OUTPUT

5.4.1.5 Mechanical Data — 28-Pin Ceramic Dual-In-Line Package



NOTES: a. Each pin centerline is located within 0.010 inch (0.26 millimeters) of its true longitudinal position.

b. All linear dimensions are shown in inches (and parenthetically in millimeters for reference only). Inch dimensions govern.

5.4.2 SBP 9961 INTERRUPT-CONTROLLER/TIMER.

5.4.2.1 Introduction.

5.4.2.1.1 General Description. The SBP 9961 Interrupt-Controller/Timer is a ruggedized, monolithic, Communications Register Unit (CRU) programmable, multifunction systems support device fabricated with oxide separated Integrated Injection Logic (I^2L) technology. The SBP 9961 provides the SBP/TMS 9900 series Family of Microprocessors with a flexible independently clocked interval/event timer plus maskable prioritized interrupt encoding capability. I^2L technology enables the SBP 9961s static logic, and TTL compatible I/O, to operate over a very wide ambient temperature range from a single d-c power source. When the SBP 9961 is used in conjunction with the I^2L SBP 9960 CRU I/O Expander, the SBP 9961/SBP 9960 pair form an I^2L systems alternate to the N-channel MOS TMS 9901 Programmable Systems Interface device while maintaining strict compatibility with existing software handlers developed in support of the TMS 9901.

5.4.2.1.2 Key Features.

- SBP/TMS 9900 Microprocessor Family Peripheral
- 15 Dedicated, Maskable, Prioritized, Encoded Interrupts
- 20 mA Current Sinking Outputs
- 40-Pin Package
- Independently Clocked 14-Bit Interval/Event Timer
- Software Compatible with TMS 9901 when used in conjunction with SBP 9960
- TTL Compatible I/O
- Wide Ambient Temperature Operation
  - SBP 9961CJ: 0°C to +70°C
  - SBP 9961EJ: −40°C to +85°C
  - SBP 9961MJ: −55°C to +125°C
  - SBP 9961NJ: −55°C to +125°C (with high-reliability processing)
- I^2L Technology
  - Constant Current Power Source
  - Fully Static Operation
  - Single Phase Edge-Triggering Clock
  - Wide Temperature Stability

5.4.2.2 Functional Description

5.4.2.2.1 SBP 9961/CPU Interface. The SBP 9961 communicates with the CPU through the Communications Register Unit (CRU) interface as shown in Figures 5-14 and 5-17. The SBP 9961s CRU interface consists of: a) five CRU address select lines (SO-S4), b) a single chip enable ($\overline{\text{CE}}$), c) a 9961 to CPU serial data-bit line (CRUIN), d) a CPU to 9961 serial data-bit line (CRUOUT), and e) a CPU to 9961 serial data-bit clock (CRUCLK). When $\overline{\text{CE}}$ is activated (logic-level low), S0-S4 selects a specific CRU-bit function as indicated in Table 5-3. In the case of a SBP 9961 write operation, the datum is transferred from the CPU to the SBP 9961 via the CRUOUT line. The CRUOUT datum is strobed into the selected 9961 CRU-bit function by CRUCLK. In the case of a SBP 9961 read operation, the selected CRU-bit function is sampled by the CPU via the CRUIN line.

Table 5-3. CRU Bit Assignments

CRU BIT	S0	S1	S2	S3	S4	CRU READ DATA	CRU WRITE DATA
0	0(4)	0	0	0	0	Control Bit	Control Bit(1)
1	0	0	0	0	1	$\overline{\text{INT}}$1/TIM1(2)	Mask1/TIM1(1)
2	0	0	0	1	0	$\overline{\text{INT}}$2/TIM2	Mask2/TIM2
3	0	0	0	1	1	$\overline{\text{INT}}$3/TIM3	Mask3/TIM3
4	0	0	1	0	0	$\overline{\text{INT}}$4/TIM4	Mask4/TIM4
5	0	0	1	0	1	$\overline{\text{INT}}$5/TIM5	Mask5/TIM5
6	0	0	1	1	0	$\overline{\text{INT}}$6/TIM6	Mask6/TIM6
7	0	0	1	1	1	$\overline{\text{INT}}$7/TIM7	Mask7/TIM7
8	0	1	0	0	0	$\overline{\text{INT}}$8/TIM8	Mask8/TIM8
9	0	1	0	0	1	$\overline{\text{INT}}$9/TIM9	Mask9/TIM9
10	0	1	0	1	0	$\overline{\text{INT}}$10/TIM10	Mask10/TIM10
11	0	1	0	1	1	$\overline{\text{INT}}$11/TIM11	Mask11/TIM11
12	0	1	1	0	0	$\overline{\text{INT}}$12/TIM12	Mask12/TIM12
13	0	1	1	0	1	$\overline{\text{INT}}$13/TIM13	Mask13/TIM13
14	0	1	1	1	1	$\overline{\text{INT}}$14/TIM14	Mask14/TIM14
15	0	1	1	1	1	$\overline{\text{INT}}$15/$\overline{\text{INTREQ}}$	Mask 15

NOTES

(1) 0 = Interrupt Mode; 1 = TIMCK Mode.

(2) Data present on $\overline{\text{INT}}$ input (or timer value) will be read regardless of mask value.

(3) While in the Interrupt Mode (Control Bit = 0), writing a "1" into a mask will enable interrupt, "0" will disable.

(4) When the SBP 9961/SBP 9960 pair are used in emulation of the TMS 9901, S0 is used to distinguish between activation of the 9961 (S0 = 0) vs the 9960 (S0 = 1).

5.4.2.2.2 Interrupt Control. A block diagram of the SBP 9961s interrupt control section is shown in Figure 5-15. The interrupt inputs are sampled on the positive-going edge of CLOCK and are ANDed with their respective mask bits. If an interrupt input is active (low) and enabled (MASK = 1), the signal is passed through the priority encoder where the highest priority signal is encoded into a 4-bit binary word as shown in Table 5-4. This word, along with an interrupt request, is then output to the CPU on the positive-going edge of the next CLOCK.

The output signals will remain valid until either the corresponding interrupt input is removed, the interrupt is disabled (MASK = 0), or a higher priority enabled interrupt becomes active. When the highest priority enabled interrupt is removed, the code corresponding to the next highest priority enabled interrupt is output. If no enabled interrupt is active, $\overline{\text{INTREQ}}$ will be pulled to logic-level high with IC0-IC3 retaining the last asserted interrupt code. $\overline{\text{RESET}}$ (power-up reset) will force the interrupt code IC0-IC3 to (0,0,0,0) with $\overline{\text{INTREQ}}$ pulled high, and will reset all mask bits low (interrupts disabled). Individual interrupts can be subsequently enabled (disabled) by programming the appropriate mask bits. Unused interrupt inputs may be used as data inputs by disabling the interrupt (MASK = 0).

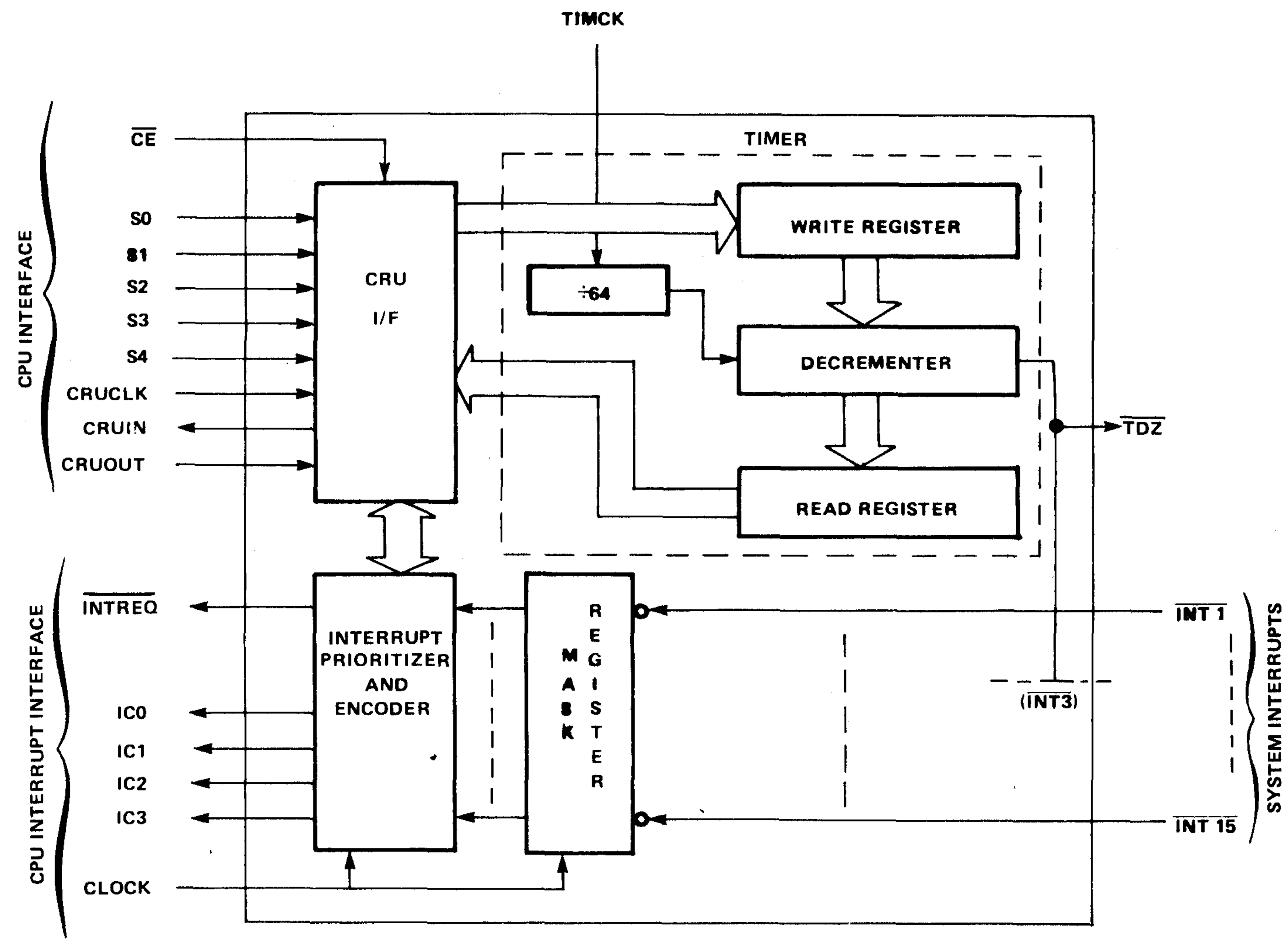


Figure 5-14. SBP 9961 Block Diagram

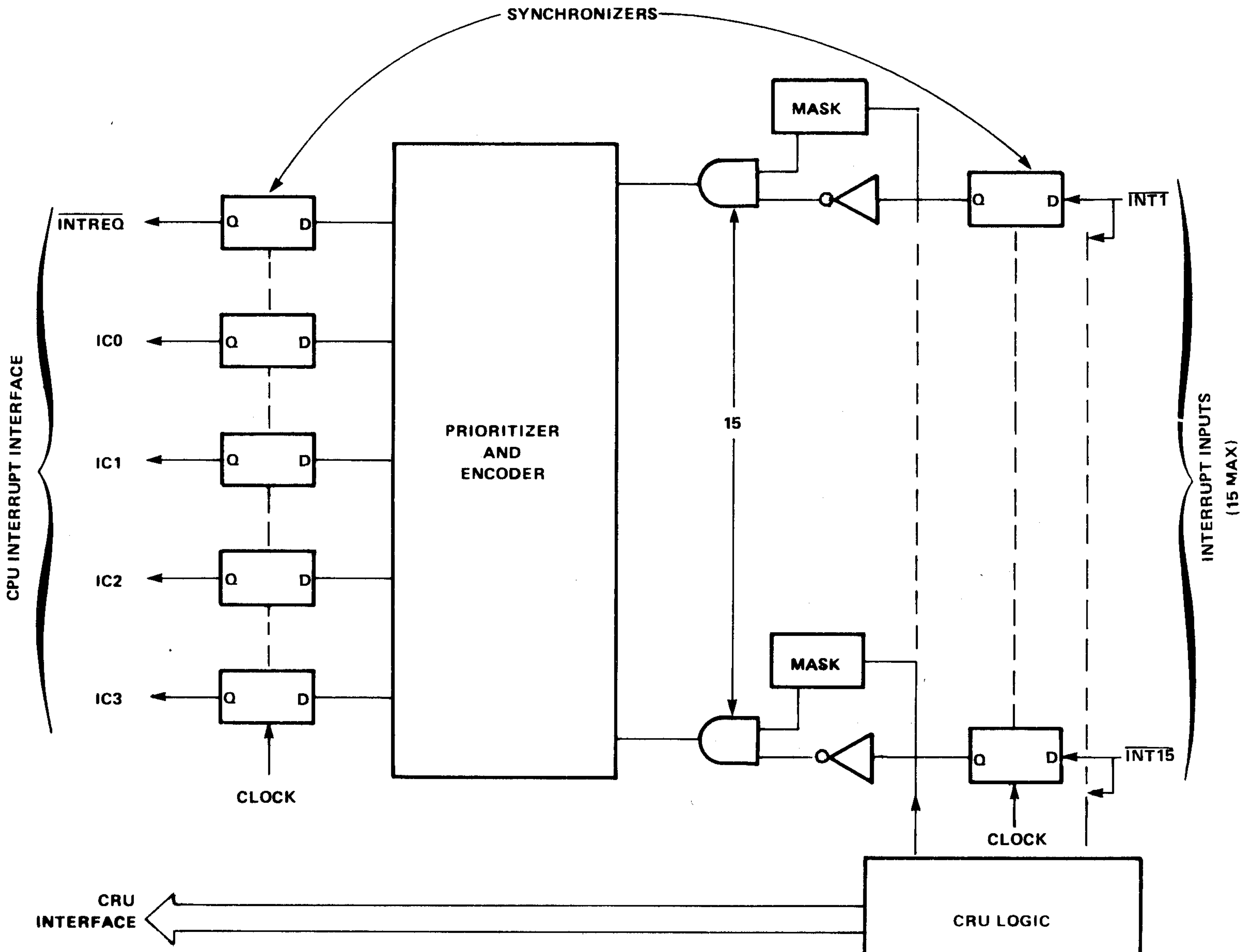


Figure 5-15. Interrupt Control Logic

Table 5-4. Interrupt Code Generation

INTERRUPT/STATE	PRIORITY	I_{C0}	I_{C1}	I_{C2}	I_{C3}	INTREQ
$\overline{\text{INT}}$1	1 (HIGHEST)	0	0	0	1	0
$\overline{\text{INT}}$2	2	0	0	1	0	0
$\overline{\text{INT}}$3/TIMER	3	0	0	1	1	0
$\overline{\text{INT}}$4	4	0	1	0	0	0
$\overline{\text{INT}}$5	5	0	1	0	1	0
$\overline{\text{INT}}$6	6	0	1	1	0	0
$\overline{\text{INT}}$7	7	0	1	1	1	0
$\overline{\text{INT}}$8	8	1	0	0	0	0
$\overline{\text{INT}}$9	9	1	0	0	1	0
$\overline{\text{INT}}$10	10	1	0	1	0	0
$\overline{\text{INT}}$11	11	1	0	1	1	0
$\overline{\text{INT}}$12	12	1	1	0	0	0
$\overline{\text{INT}}$13	13	1	1	0	1	0
$\overline{\text{INT}}$14	14	1	1	1	0	0
$\overline{\text{INT}}$15	15 (LOWEST)	1	1	1	1	0
NO INTERRUPT	—	Note 1	Note 1	Note 1	Note 1	1

NOTE 1: IC0-IC3 hold the level code of the previous interrupt.

5.4.2.2.3 Interval Timer. The SBP 9961's interval/event timer, shown in Figure 5-16, has the following operational features:

a) Independent clock input TIMCK

b) Programmable 14-bit decrementer

c) Timer-reaches-zero issues level-3 interrupt (unless masked)

d) Timer-reaches-zero issues output pulse TDZ

e) Able to read the current decremented value and therefore function as an event timer

f) Able to determine the SBP 9961s operating mode and value of $\overline{\text{INTREQ}}$.

The SBP 9961 has an independent timer clock input, TIMCK which allows the user to define an interval timer clock frequency other than that ofthe CPU. This, however, does not preclude the user option of connecting TIMCK to the CLOCK input and therefore running the interval timer at the CPUs clock frequency. The typical operating range of TIMCK is 0-5 MHz.

The timer's CRU control bits are shown in Table 5-3. The SBP 9961 is placed into the timer-access mode by writing a logic-level high to the control bit located at CRU address zero. CRU bits 1-14 are then used to initiate the write-register with the desired start count. Writing a non-zero value to the write-register a) enables the decrementer, b) programs the third priority interrupt ($\overline{\text{INT3}}$) as the timer interrupt, and c) disables the influence of external interrupts on the $\overline{\text{INT3}}$ input pin. A single LDCR instruction can be used to accomplish the above initialization operation. After the write-register has been initialized with the desired start count, the timer begins decrementing toward zero. Upon reaching zero, the timer issues the level-3 interrupt, outputs the timer-zero pulse $\overline{\text{TDZ}}$, and restarts itself with the write-register value. Since the timer interrupt is latched, clearing that interrupt is accomplished by writing either a logic-level low or high to the respective interrupt mask bit at CRU address three. The CRUCLK that accompanies that write operation is the stimulus which resets the timers interrupt latch. However, in order to retain the current mask value, the appropriate SBZ or SBO CRU-write instruction must be executed, unless the mask value itself is to be changed. At any point in the timer's decrement sequence, a timer restart can be accomplished by either reinitiating the entire write-register with an LDCR instruction, or by writing to any individual write-register bit with an SBZ or SBO instruction.

If the control bit is at logic-level low, the timer's read-register is updated with the current decrementer value after each decrement operation (once every 64 TIMCK clocks); if the control bit is at logic-level high (timer-access mode), the read-register retains its current value thereby ensuring that the read-register is not changed in the event a CRU read operation is executing during a decrement operation. Consequently, the current value of the timer's decrementer can be interrogated by 1) placing the SBP 9961 into the timer-access mode, and 2) performing a CRU-read operation on the timer's read-register through execution of an STCR instruction. The timer, then, can function as an event timer by reading the elapsed time between software events as shown in Table 5-5. Note that when accessing the timer, all interrupts should be disabled. The timer is disabled by either $\overline{\text{RESET}}$ (power-up reset) or by writing all zeroes to the write-register.

5.4.2.2.4 SBP 9961 Status. The SBP 9961s status can be determined by reading the value of the control bit located at CRU address zero. If the control bit has a logic-level low value, then the interrupt masks may be changed and data on the interrupt inputs may be read. However, access to the interval timer is inhibitéd. If the control bit has a logic-level high value, then the timer may be initiated, restarted, or read. Also, reading CRU address fifteen gives the status of $\overline{\text{INTREQ}}$ where logic-level low indicates activation.

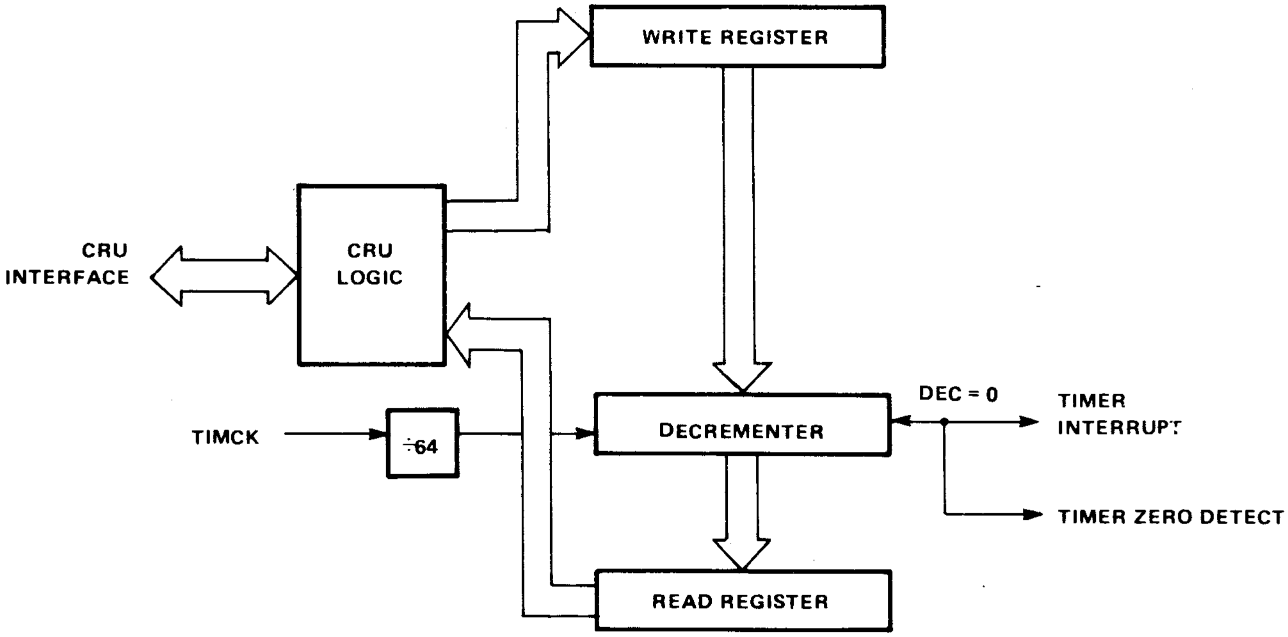


Figure 5-16. Interval/Event Timer

Table 5-5. Software Examples

ASSUMPTIONS:

- — Total of 6 interrupts are used
- — $\overline{\text{RESET}}$ has been applied
- — System uses timer at maximum interval

```
             LI       R12,CRUBAS       Setup CRU Base Address to point to 9961
             LDCR     ;X,0             Program Timer with maximum interval
SYSTEM       LDCR     ;Y,7             Re-enter interrupt mode and enable top 6 interrupts
SETUP        (X)——————► FFFF
             (Y)——————► 7FXX
             .
             .
             .
             .
             .
             BLWP     CLKVCT           Save Interrupt Mask
             .
             .
             .
CLKPC        LIMI     0                Disable Interrupts
             LI       R12,CRUBASE+1    Set up CRU base
             SBO      -1               Set 9961 into timer-access mode
             STCR     R4,14            Store read register into R4
             .
             .                         Process Timer Value
             .
             .
             SBZ      -1               Re-enter Interrupt Mode (i.e., Exit Time-Access Mode)
             RTWP                      Restore Interrupt Mask
             .
             .
             .
CLKUCT       DATA     CLKWP,CLKPC
```

5.4.2.2.5 System Operation. During power-up, $\overline{\text{RESET}}$ should be activated (low) to force the SBP 9961 into a known state. $\overline{\text{RESET}}$ will disable all interrupts, disable the timer, and force IC0-IC3 to (0,0,0,0) with $\overline{\text{INTREQ}}$ pulled high. System software should then enable the proper interrupts and program the timer (if used). See Table 5-5 for an example. After initial power up, the SBP 9961 is accessed only as needed to service the timer and enable or disable interrupts.

Figure 5-17 shows the SBP 9961 system configuraiion. Figure 5-18 shows the use of a SBP 9961 with a SBP 9960 CRU I/O expander in emulation of the TMS 9901. (See *TMS 9901 Systems Interface Data Manual.*)

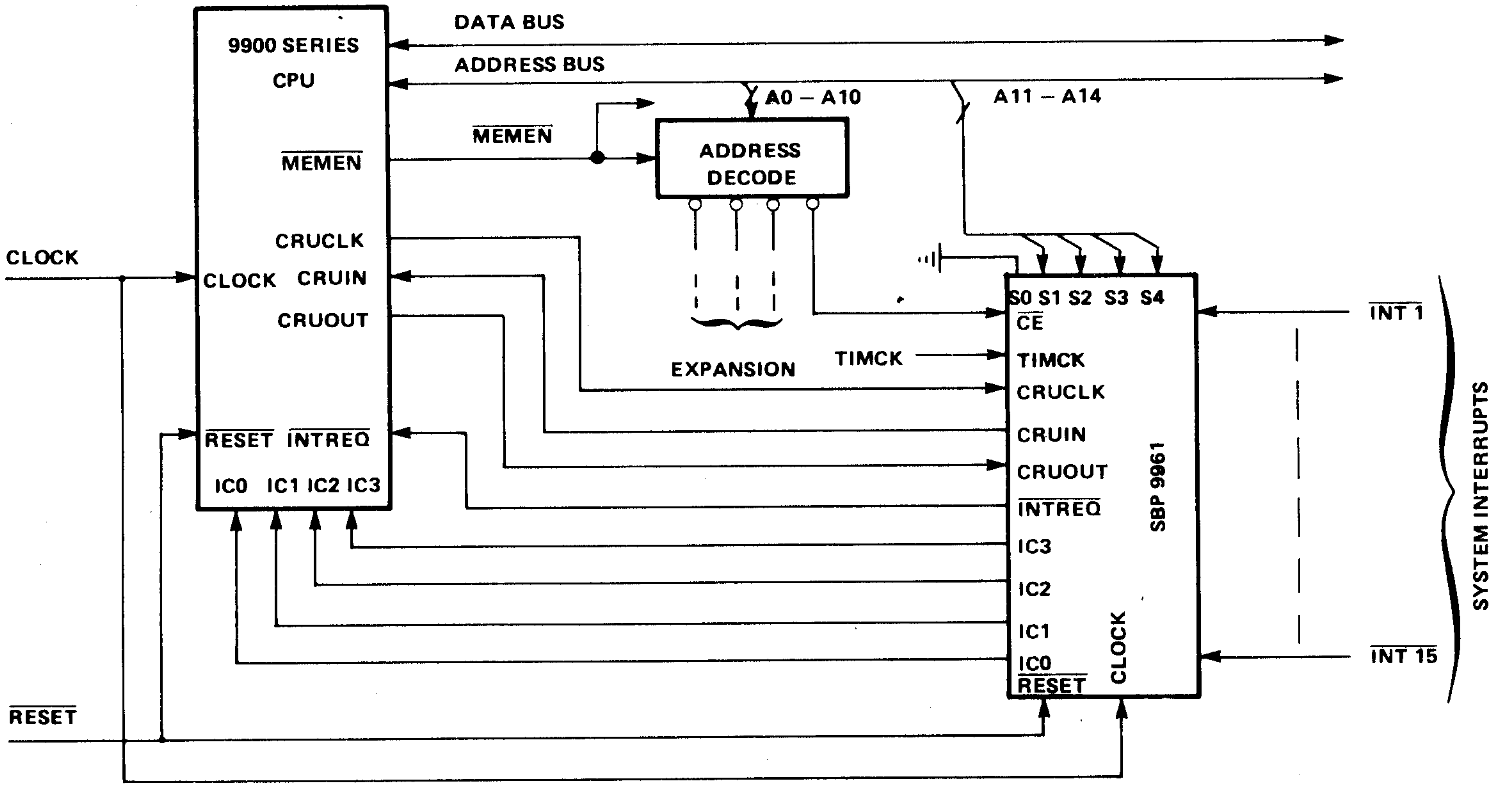


Figure 5-17. SBP 9961 System Configuration

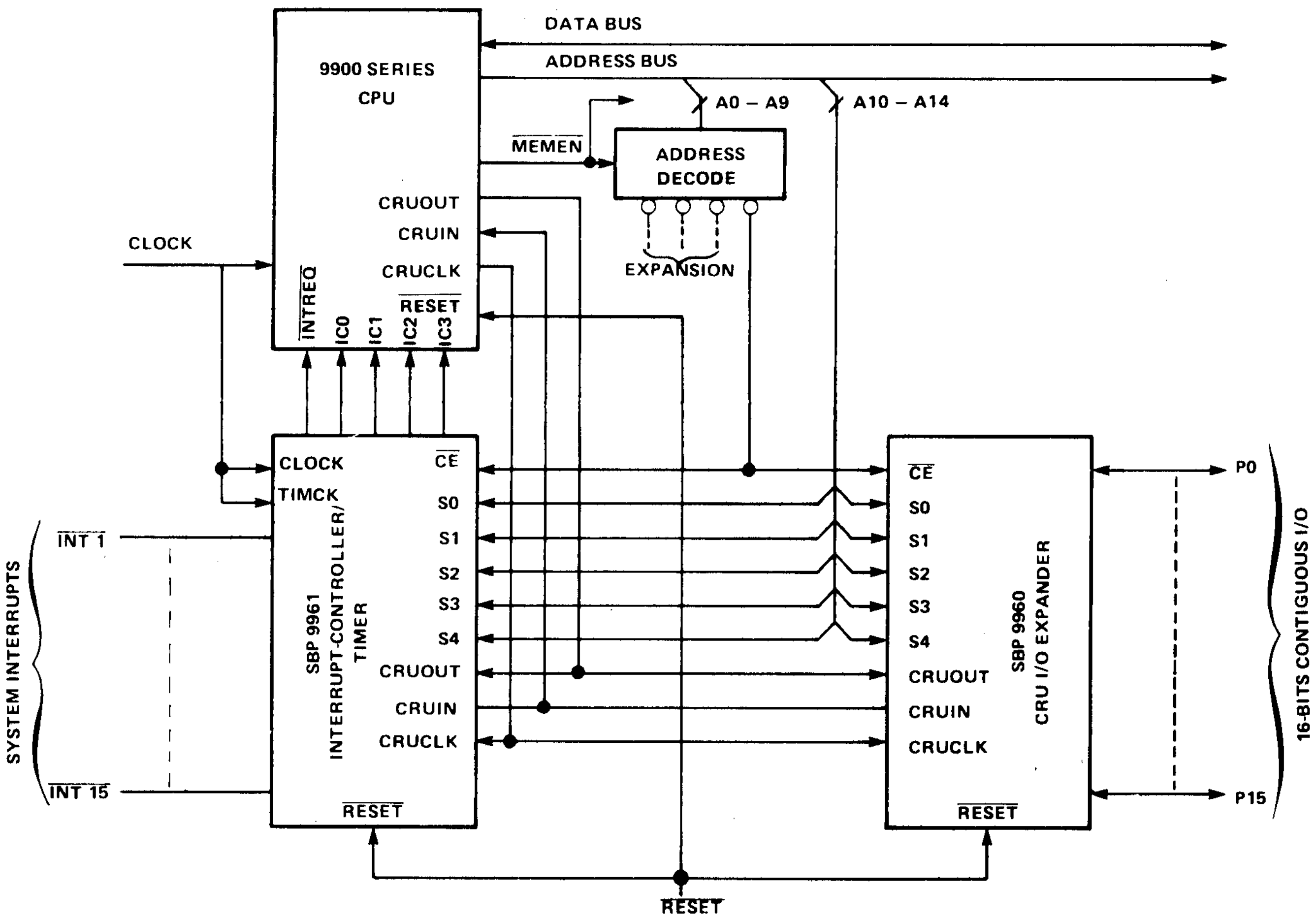
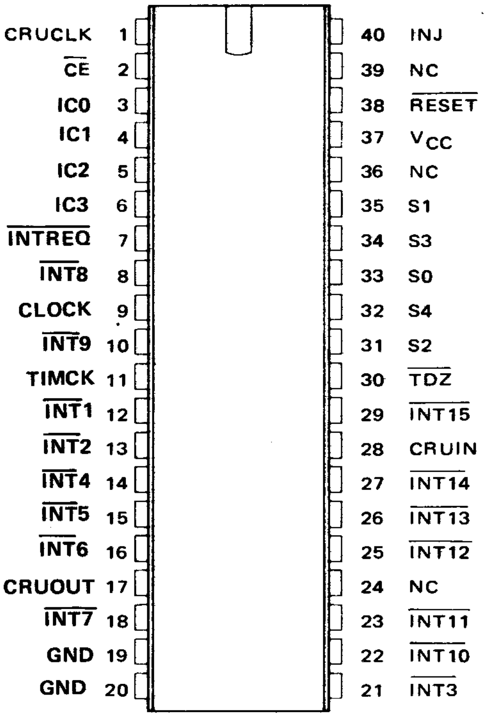


Figure 5-18. SBP 9961 Configuration with SBP 9960

5.4.2.2.6 SBP 9961 Pin Assignments and Functions.

SIGNATURE	PIN	I/O	DESCRIPTION
S0	33	IN	ADDRESS SELECT LINES. The data bit being accessed by the CRU interface is specified by the 4-bit code appearing on S1-S4. S0 is used as the high order select line when the SBP 9961 is used with the SBP 9960 in emulation of the TMS 9901. Otherwise, tie S0 to logic-level low.
S1	35	IN	
S2	31	IN	
S3	34	IN	
S4	32	IN	
CRUIN	28	OUT	CRU DATA IN (to CPU). Data specified by S0-S4 is transmitted to the CPU by CRUIN. When $\overline{\text{CE}}$ is not active, CRUIN is logic-level high.
CRUOUT	17	IN	CRU DATA OUT (from CPU). When $\overline{\text{CE}}$ is active, data present on the CRUOUT input will be sampled during CRUCLK and written into the CRU bit specified by S0-S4.
CRUCLK	1	IN	CRU CLOCK (from CPU). CRUCLK specifies that valid data is present on the CRUOUT line.
$\overline{\text{RESET}}$	38	IN	POWER-UP RESET. When active (low), $\overline{\text{RESET}}$ forces all interrupt masks to "0", and disables the clock.
$\overline{\text{CE}}$	2	IN	CHIP ENABLE. When active (low), data transfers may occur between the CPU and the SBP 9961.
TIMCK	11	IN	TIMER CLOCK IN. External clock used for the timer decrementer. May be externally tied to the CLOCK input pin.
$\overline{\text{TDZ}}$	30	OUT	TIMER DECREMENTER EQUALS ZERO. Low active pulse įndicating that the timers decrementer contains a value of zero (all logic-level lows).
IC0	3	OUT	INTERRUPT CODE LINES (to CPU). IC0 (MSB) through IC3 output the binary code corresponding to the highest priority enabled interrupt most recently asserted.
IC1	4	OUT	
IC2	5	OUT	
IC3	6	OUT	
$\overline{\text{INTREQ}}$	7	OUT	INTERRUPT REQUEST (to CPU). When active (low) $\overline{\text{INTREQ}}$ indicates to the CPU that an enabled interrupt has been asserted, prioritized, and encoded.
CLOCK	9	IN	CPU SYSTEM CLOCK. Used by the SBP 9961 to synchronize the interrupt interface ($\overline{\text{INTREQ}}$, IC0-IC3) to the CPU.
INJ	40		Supply Current
GND	19,20		Ground Reference
V_{CC}	37		Common voltage return/reference for all I/O pull-up resistors.
$\overline{\text{INT}}$1	12	IN	INTERRUPT INPUTS. When active (low), the signal is ANDed with its corresponding mask bit and if enabled sent to the interrupt control section. $\overline{\text{INT}}$1 has highest priority.
$\overline{\text{INT}}$2	13	IN	
$\overline{\text{INT}}$3	21	IN	
$\overline{\text{INT}}$4	14	IN	
$\overline{\text{INT}}$5	15	IN	
$\overline{\text{INT}}$6	16	IN	
$\overline{\text{INT}}$7	18	IN	
$\overline{\text{INT}}$8	8	IN	
$\overline{\text{INT}}$9	10	IN	
$\overline{\text{INT}}$10	22	IN	
$\overline{\text{INT}}$11	23	IN	
$\overline{\text{INT}}$12	25	IN	
$\overline{\text{INT}}$13	26	IN	
$\overline{\text{INT}}$14	27	IN	
$\overline{\text{INT}}$15	29	IN	



5.4.2.3 Electrical Specifications.

5.4.2.3.1 Recommended Operating Conditions, Unless Otherwise Noted I_{CC} ‡ 130 mA

		MIN	NOM	MAX	UNIT
Supply current, I_{CC}		115	130	145	mA
High-level output voltage, V_{OH}				5.5	V
Low-level output current, I_{OL}				20	mA
Operating free-air temperature, T_A	SBP 9961MJ, SBP 9961NJ	−55		125	°C
	SBP 9961EJ	−40		85	
	SBP 9961CJ	0		70	

5.4.2.3.2 Electrical Characteristics (Over Recommended Operating Free-Air Temperature Range, Unless Otherwise Noted)

PARAMETER		TEST CONDITIONS†		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage			2			V
V_{IL}	Low-level input voltage					0.8	V
V_{IK}	Input clamp voltage	I_{CC} = MIN,	I_I = −12 mA			−1.5	V
I_{OH}	High-level output current	I_{CC} = 130 mA, V_{IL} = 0.8 V,	V_{IH} = 2 V, V_{OH} = 5.5 V			−400	μA
V_{OL}	Low-level output voltage	I_{CC} = 130 mA, V_{IL} = 0.8 V,	V_{IH} = 2 V, I_{OL} = 20 mA			0.5	V
I_I	Input current	I_{CC} = 130 mA,	V_I = 2.4 V		180		μA

†For conditions shown as MAX, use the appropriate value specified under recommended operating conditions

‡All typical values are at I_{CC} = 130 mA, T_A = 25°C.

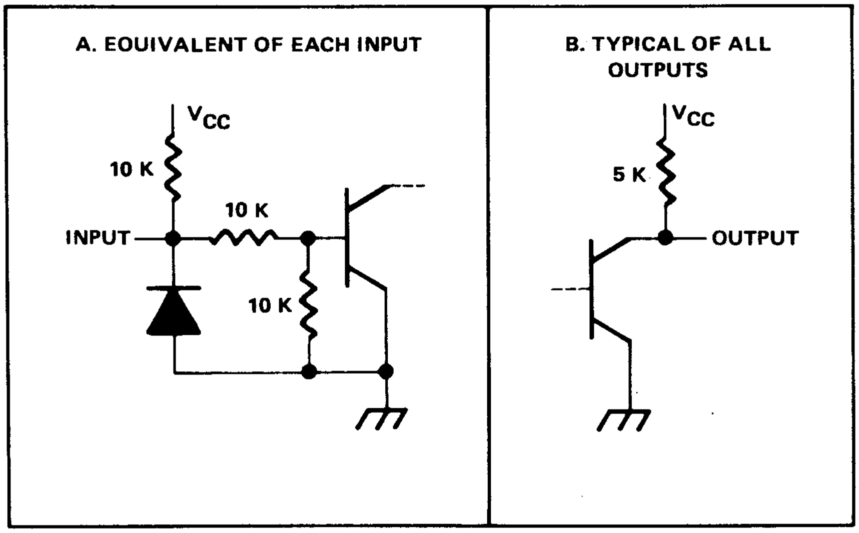
5.4.2.3.3 *Timing Requirements over Full Range of Operating Conditions*

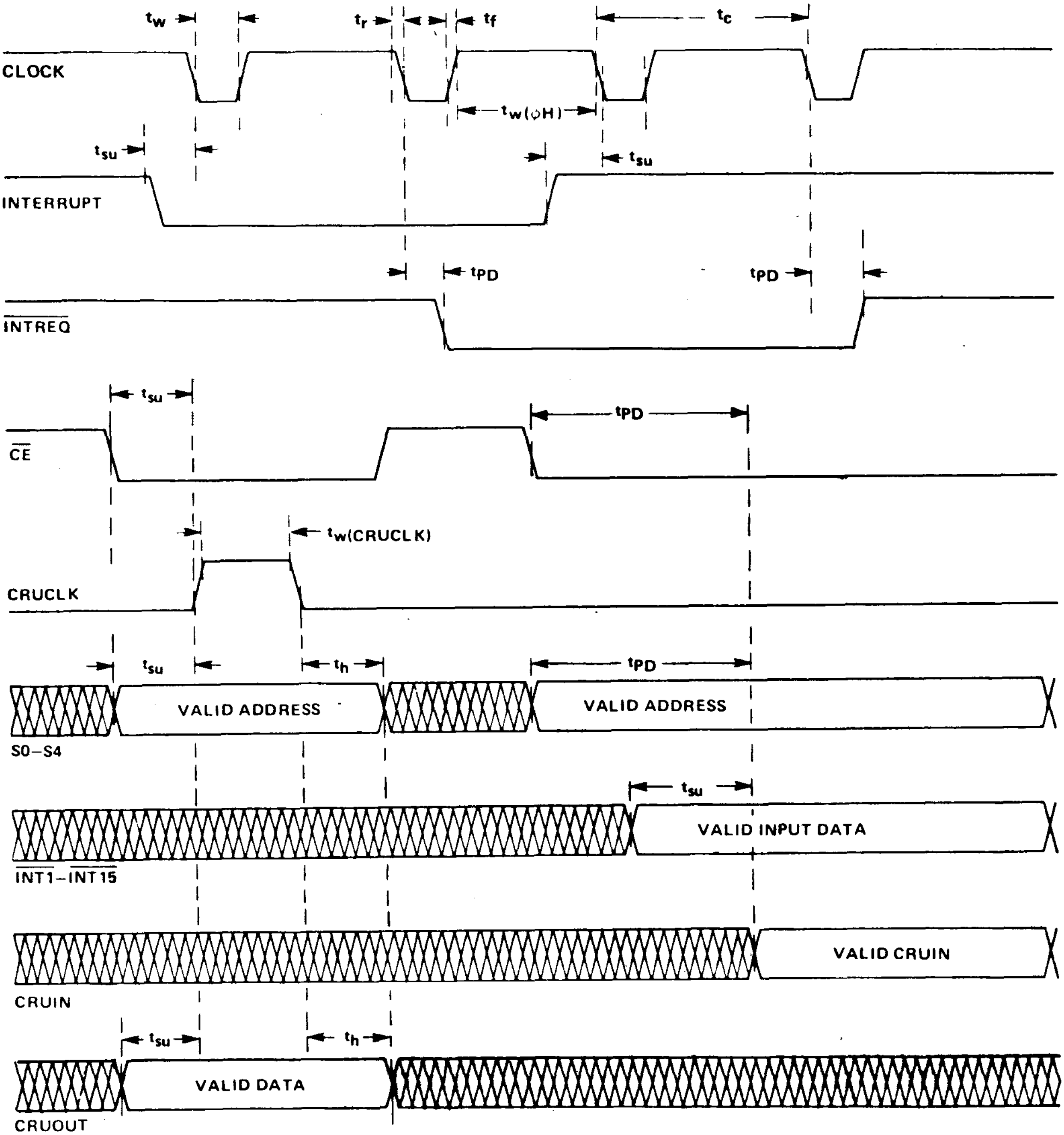
PARAMETER		MIN	NOM	MAX	UNIT
t_c	Clock cycle time	333			ns
t_r	Clock rise time		10	20	ns
t_f	Clock fall time		10	20	ns
t_{wL}	Clock pulse low width	111			ns
t_{wH}	Clock pulse high width	222			ns
t_{su}	Setup time for S0-S4, $\overline{\text{CE}}$, or CRUOUT before CRUCLK		200		ns
t_{su}	Setup time, input before valid CRUIN		200		ns
t_{su}	Setup time, interrupt before clock high		60		ns
$t_{w(CRUCLK)}$	CRU clock pulse width		100		ns
t_h	Address hold time		80		ns
t_{TC}	TIMCK cycle time		200		ns

5.4.2.3.4 *Switching Characteristics over Full Range of Recommended Operating Conditions*

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PD}	Propagation delay, ↑ CLOCK to valid $\overline{\text{INTREQ}}$, IC0-IC3	C_L = 25 pF, R_L = 5K Ω		150		ns
t_{PD}	Propagation delay, S0-S4 or $\overline{\text{CE}}$ to valid CRUIN	C_L = 25 pF, R_L = 5K Ω		330		ns

5.4.2.4 Input, Output, and Input/Output Structures



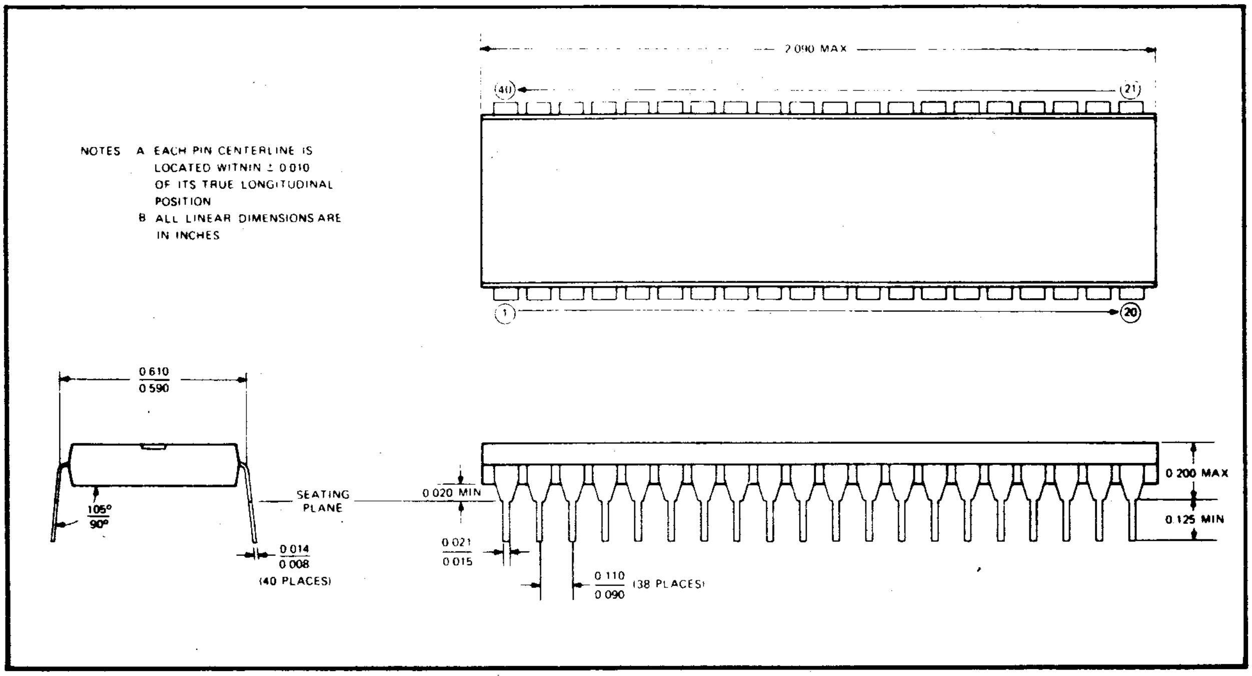


NOTE 1: ALL TIMING MEASUREMENTS ARE FROM 10% and 90% POINTS

SWITCHING CHARACTERISTICS

5.4.2.5 Mechanical Data

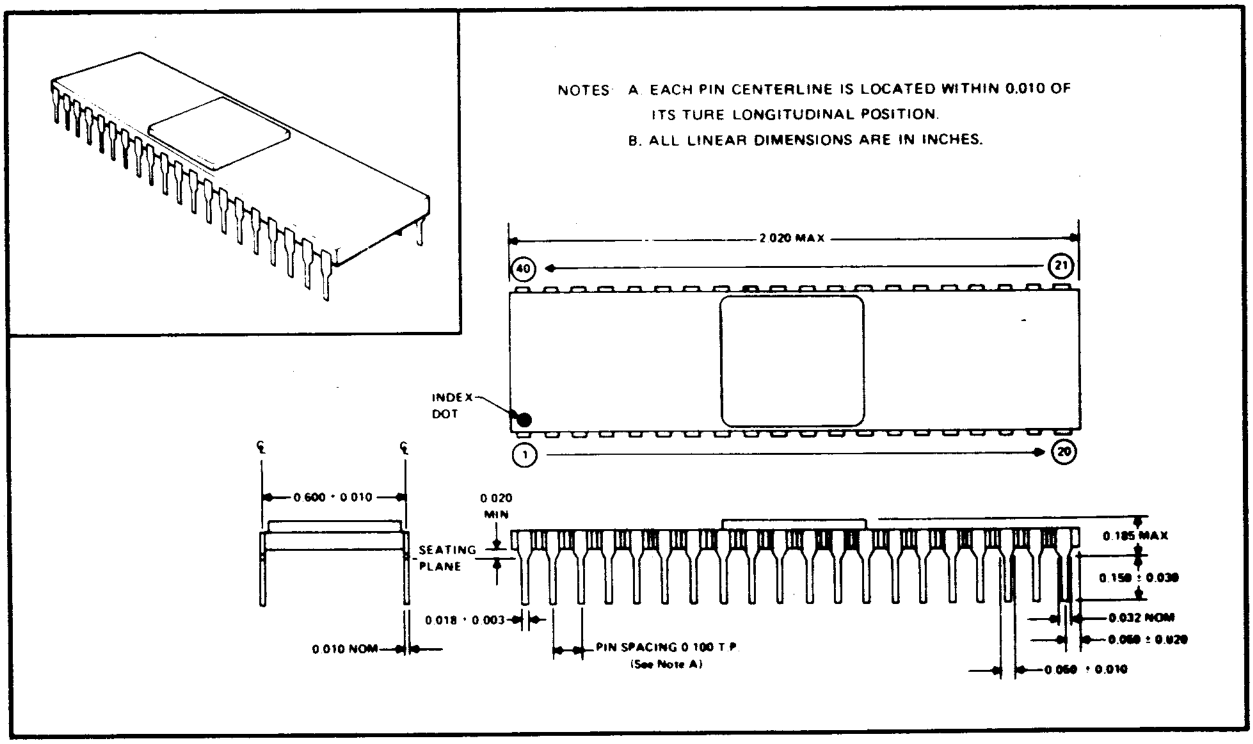
5.4.2.5.1 40-Pin Plastic Dual-In-Line Package



NOTES A EACH PIN CENTERLINE IS LOCATED WITNIN ± 0.010 OF ITS TRUE LONGITUDINAL POSITION

B ALL LINEAR DIMENSIONS ARE IN INCHES

5.4.2.5.2 40-Pin Ceramic Dual-In-Line Package



NOTES A EACH PIN CENTERLINE IS LOCATED WITHIN 0.010 OF ITS TURE LONGITUDINAL POSITION.

B. ALL LINEAR DIMENSIONS ARE IN INCHES.

5.5 SOFTWARE — UART

Some systems may not require the full capabilities offered by a hardware UART. For example, a small system can be interfaced to the 733 ASR Terminal for initial program loading only. If transmission is half-duplex and the program has no significant tasks to perform between input characters, a software-UART function can be programmed to serially input and output bits with software format conversion.The system in Figure 19 uses a software-UART program to minimize the system component count at the expense of added software.

The software-UART consists of a receive routine and a transmit routine which are called by XOP or BLWP instructions. The routines share a common workspace and a common delay subroutine. The seven-bit, plus parity, transmit and receive, characters are passed through register 0 of the calling program. Since the software-UART routines must run to completion, the Interrupt Mask is cleared to disable all maskable interrupts. Registers 4, 5, 8, and 10 are used for temporary results. Register 6 and 7 contain pointers to the half-bit and full-bit subroutines and register 9 contains a bit mask. Registers 6, 7, 9, and 12 are initialized by the system restart routine. The delay routines introduce software determined delays equal to half-bit and full-bit times for a 300 bps data terminal. The receive routine rejects input characters with bad parity, false start bits, or missing stop bits. The software-UART routines are shown in Figure 5-20.

5.6 I^{2}L MICROPROCESSOR SYSTEM

An I^{2}L microprocessor system is shown in Figure 5-21.

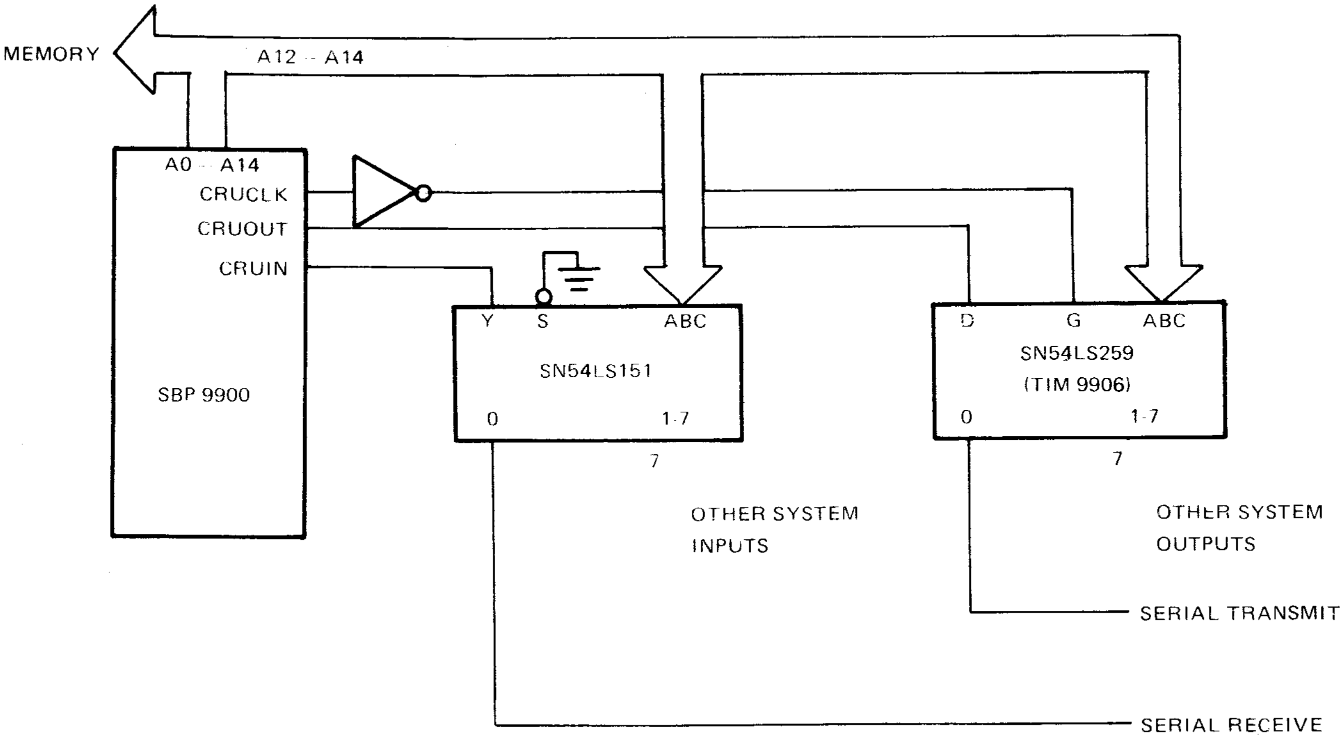
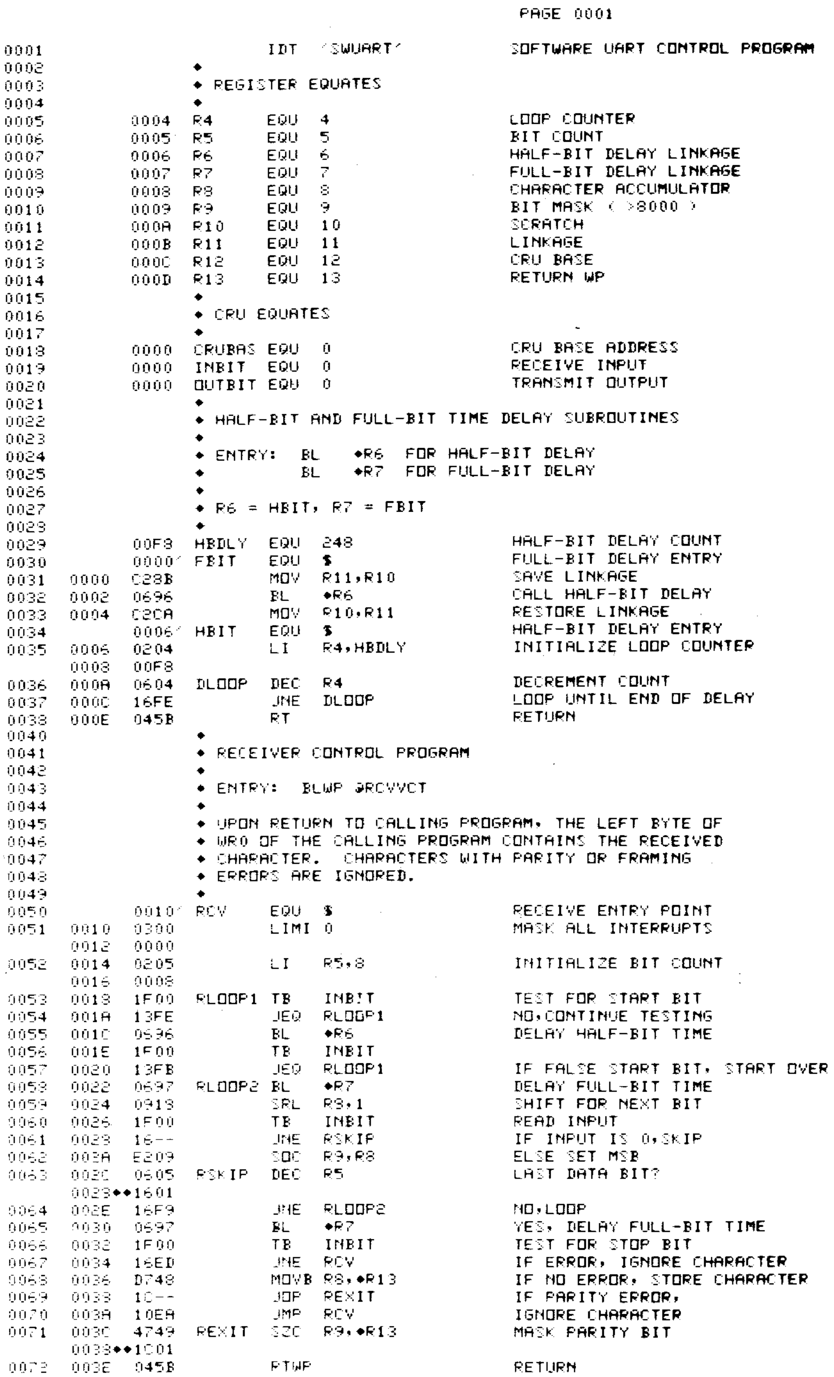


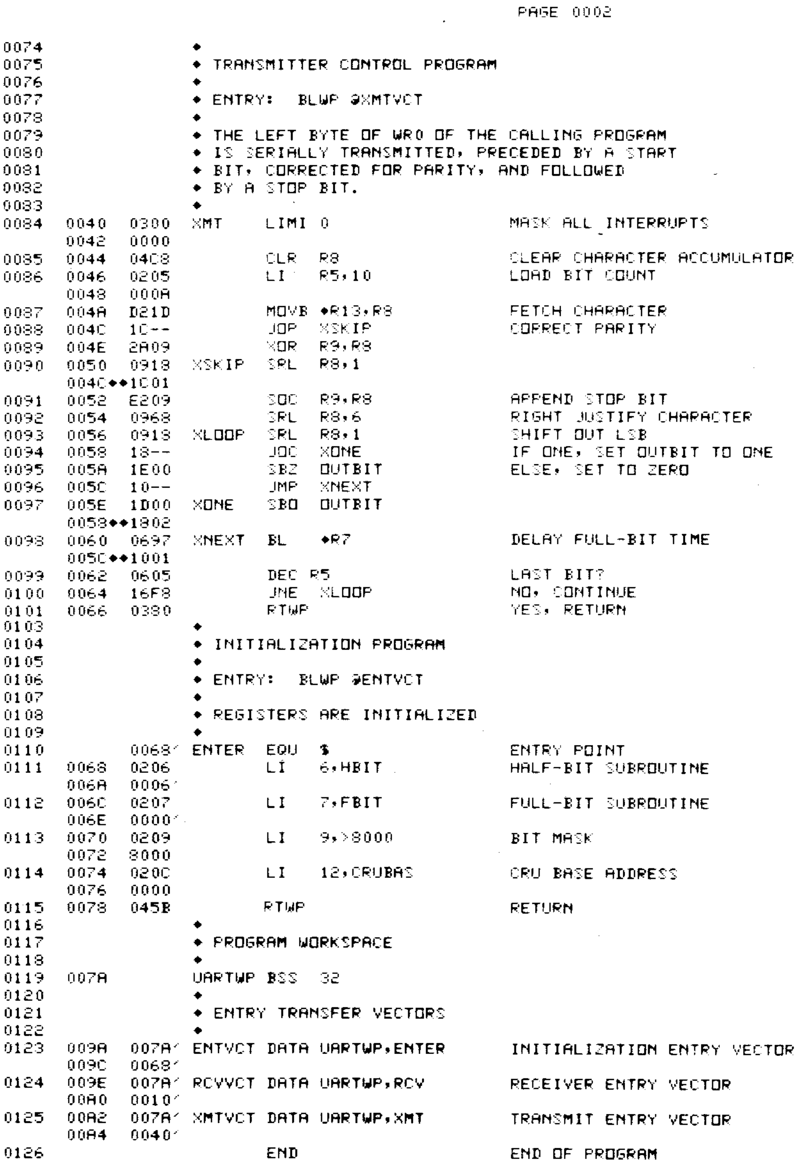
Figure 5-19. Software UART CRU Interface



```
                                                          PAGE 0001

0001                  IDT   'SWUART'             SOFTWARE UART CONTROL PROGRAM
0002              ◆
0003              ◆ REGISTER EQUATES
0004              ◆
0005         0004 R4     EQU  4                  LOOP COUNTER
0006         0005 R5     EQU  5                  BIT COUNT
0007         0006 R6     EQU  6                  HALF-BIT DELAY LINKAGE
0008         0007 R7     EQU  7                  FULL-BIT DELAY LINKAGE
0009         0008 R8     EQU  8                  CHARACTER ACCUMULATOR
0010         0009 R9     EQU  9                  BIT MASK ( >8000 )
0011         000A R10    EQU  10                 SCRATCH
0012         000B R11    EQU  11                 LINKAGE
0013         000C R12    EQU  12                 CRU BASE
0014         000D R13    EQU  13                 RETURN WP
0015              ◆
0016              ◆ CRU EQUATES
0017              ◆
0018         0000 CRUBAS EQU  0                  CRU BASE ADDRESS
0019         0000 INBIT  EQU  0                  RECEIVE INPUT
0020         0000 OUTBIT EQU  0                  TRANSMIT OUTPUT
0021              ◆
0022              ◆ HALF-BIT AND FULL-BIT TIME DELAY SUBROUTINES
0023              ◆
0024              ◆ ENTRY:  BL   ◆R6  FOR HALF-BIT DELAY
0025              ◆         BL   ◆R7  FOR FULL-BIT DELAY
0026              ◆
0027              ◆ R6 = HBIT, R7 = FBIT
0028              ◆
0029         00F8 HBDLY  EQU  248                HALF-BIT DELAY COUNT
0030         0000'FBIT   EQU  $                  FULL-BIT DELAY ENTRY
0031    0000 C28B        MOV  R11,R10            SAVE LINKAGE
0032    0002 0696        BL   ◆R6                CALL HALF-BIT DELAY
0033    0004 C2CA        MOV  R10,R11            RESTORE LINKAGE
0034         0006'HBIT   EQU  $                  HALF-BIT DELAY ENTRY
0035    0006 0204        LI   R4,HBDLY           INITIALIZE LOOP COUNTER
        0008 00F8
0036    000A 0604 DLOOP  DEC  R4                 DECREMENT COUNT
0037    000C 16FE        JNE  DLOOP              LOOP UNTIL END OF DELAY
0038    000E 045B        RT                      RETURN
0040              ◆
0041              ◆ RECEIVER CONTROL PROGRAM
0042              ◆
0043              ◆ ENTRY:  BLWP @RCVVCT
0044              ◆
0045              ◆ UPON RETURN TO CALLING PROGRAM, THE LEFT BYTE OF
0046              ◆ WR0 OF THE CALLING PROGRAM CONTAINS THE RECEIVED
0047              ◆ CHARACTER.  CHARACTERS WITH PARITY OR FRAMING
0048              ◆ ERRORS ARE IGNORED.
0049              ◆
0050         0010'RCV    EQU  $                  RECEIVE ENTRY POINT
0051    0010 0300        LIMI 0                  MASK ALL INTERRUPTS
        0012 0000
0052    0014 0205        LI   R5,8               INITIALIZE BIT COUNT
        0016 0008
0053    0018 1F00 RLOOP1 TB   INBIT              TEST FOR START BIT
0054    001A 13FE        JEQ  RLOOP1             NO,CONTINUE TESTING
0055    001C 0696        BL   ◆R6                DELAY HALF-BIT TIME
0056    001E 1F00        TB   INBIT
0057    0020 13FB        JEQ  RLOOP1             IF FALSE START BIT, START OVER
0058    0022 0697 RLOOP2 BL   ◆R7                DELAY FULL-BIT TIME
0059    0024 0918        SRL  R8,1               SHIFT FOR NEXT BIT
0060    0026 1F00        TB   INBIT              READ INPUT
0061    0028 16--        JNE  RSKIP              IF INPUT IS 0,SKIP
0062    002A E209        SOC  R9,R8              ELSE SET MSB
0063    002C 0605 RSKIP  DEC  R5                 LAST DATA BIT?
        0028◆◆1601
0064    002E 16F9        JNE  RLOOP2             NO,LOOP
0065    0030 0697        BL   ◆R7                YES, DELAY FULL-BIT TIME
0066    0032 1F00        TB   INBIT              TEST FOR STOP BIT
0067    0034 16ED        JNE  RCV                IF ERROR, IGNORE CHARACTER
0068    0036 D748        MOVB R8,◆R13            IF NO ERROR, STORE CHARACTER
0069    0038 1C--        JOP  REXIT              IF PARITY ERROR,
0070    003A 10EA        JMP  RCV                IGNORE CHARACTER
0071    003C 4749 REXIT  SZC  R9,◆R13            MASK PARITY BIT
        0038◆◆1C01
0072    003E 045B        RTWP                    RETURN
```

Figure 5-20. Software UART Control Program (Page 1 of 2)



```
                                                                  PAGE 0002
0074                 *
0075                 * TRANSMITTER CONTROL PROGRAM
0076                 *
0077                 * ENTRY:   BLWP @XMTVCT
0078                 *
0079                 * THE LEFT BYTE OF WR0 OF THE CALLING PROGRAM
0080                 * IS SERIALLY TRANSMITTED, PRECEDED BY A START
0081                 * BIT, CORRECTED FOR PARITY, AND FOLLOWED
0082                 * BY A STOP BIT.
0083                 *
0084   0040   0300   XMT    LIMI 0                 MASK ALL INTERRUPTS
       0042   0000
0085   0044   04C8          CLR  R8                CLEAR CHARACTER ACCUMULATOR
0086   0046   0205          LI   R5,10             LOAD BIT COUNT
       0048   000A
0087   004A   D21D          MOVB *R13,R8           FETCH CHARACTER
0088   004C   1C--          JOP  XSKIP             CORRECT PARITY
0089   004E   2A09          XOR  R9,R8
0090   0050   0918   XSKIP  SRL  R8,1
       004C**1C01
0091   0052   E209          SOC  R9,R8             APPEND STOP BIT
0092   0054   0968          SRL  R8,6              RIGHT JUSTIFY CHARACTER
0093   0056   0918   XLOOP  SRL  R8,1              SHIFT OUT LSB
0094   0058   18--          JOC  XONE              IF ONE, SET OUTBIT TO ONE
0095   005A   1E00          SBZ  OUTBIT            ELSE, SET TO ZERO
0096   005C   10--          JMP  XNEXT
0097   005E   1D00   XONE   SBO  OUTBIT
       0058**1802
0098   0060   0697   XNEXT  BL   *R7               DELAY FULL-BIT TIME
       005C**1001
0099   0062   0605          DEC R5                 LAST BIT?
0100   0064   16F8          JNE  XLOOP             NO, CONTINUE
0101   0066   0380          RTWP                   YES, RETURN
0103                 *
0104                 * INITIALIZATION PROGRAM
0105                 *
0106                 * ENTRY:   BLWP @ENTVCT
0107                 *
0108                 * REGISTERS ARE INITIALIZED
0109                 *
0110          0068'  ENTER  EQU  $                 ENTRY POINT
0111   0068   0206          LI   6,HBIT            HALF-BIT SUBROUTINE
       006A   0006'
0112   006C   0207          LI   7,FBIT            FULL-BIT SUBROUTINE
       006E   0000'
0113   0070   0209          LI   9,>8000           BIT MASK
       0072   8000
0114   0074   020C          LI   12,CRUBAS         CRU BASE ADDRESS
       0076   0000
0115   0078   045B          RTWP                   RETURN
0116                 *
0117                 * PROGRAM WORKSPACE
0118                 *
0119   007A          UARTWP BSS  32
0120                 *
0121                 * ENTRY TRANSFER VECTORS
0122                 *
0123   009A   007A'  ENTVCT DATA UARTWP,ENTER      INITIALIZATION ENTRY VECTOR
       009C   0068'
0124   009E   007A'  RCVVCT DATA UARTWP,RCV        RECEIVER ENTRY VECTOR
       00A0   0010'
0125   00A2   007A'  XMTVCT DATA UARTWP,XMT        TRANSMIT ENTRY VECTOR
       00A4   0040'
0126                        END                    END OF PROGRAM
```

Figure 5-20. Software UART Control Program (Page 2 of 2)

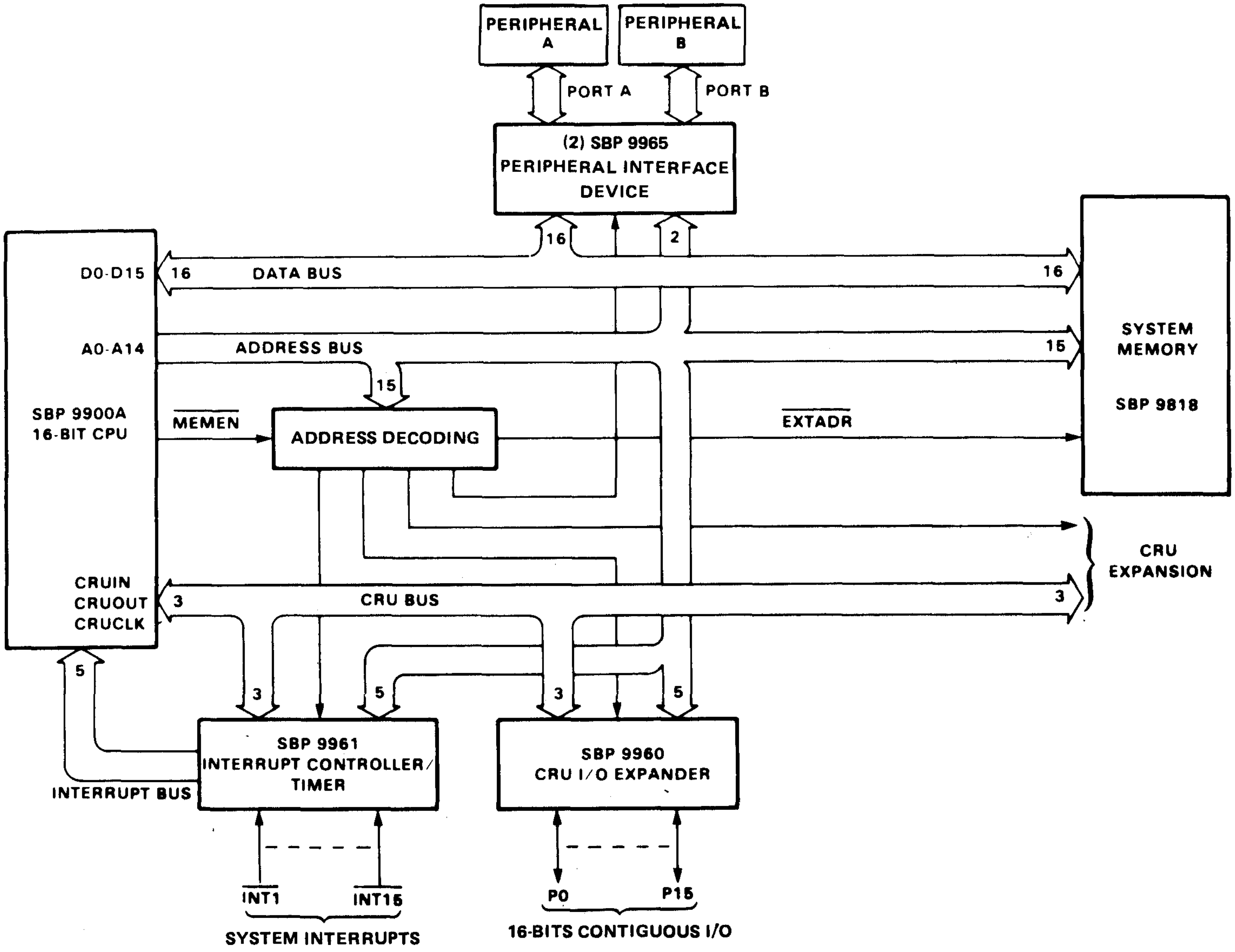


Figure 5-21. I^{2}L Microprocessor System

SECTION VI
AUXILIARY SYSTEM FUNCTIONS

This section describes the circuitry and application of several functions which may be useful in an SBP 9900A system.

6.1 UNUSED OP CODES

The unused op codes for the SBP 9900A are shown in Table 6-1. An instruction which consists of any of these codes will cause the SBP 9900A to perform no operation other than a) updating the PC to point to the next sequential even word address, and/or b) processing $\overline{\text{RESET}}$, $\overline{\text{LOAD}}$, and interrupts, in the normal manner. Since future upward compatible members of the SBP 9900A microprocessor family will use these codes for presently unimplemented instructions, it is not recommended that any of these codes be used to perform the function of a no-op. The SBP 9900A assembler has defined the pseudo-instruction "NOP" to generate a code (1000_{16}) equivalent to a "JMP $+2" instruction.

6.1.1 UNUSED OP CODE DETECTION. The occurrence of unused op codes may be detected by the circuitry shown in Figure 6-1. The signal ILOPCD is set high at the end of the memory cycle which fetches the unused op code. On the first clock cycle of each instruction fetch, the IAQTI signal is set high. Then, when memory data is read (on the next clock cycle in which the SBP 9900A is not in a WAIT state) the IAQTI signal is reset. If $\overline{\text{ILODEC}} = 0$, the illegal op code flag ILOPCD is set. Note that the SN74S330 Field Programmable Logic Array is used to decode the illegal op codes. If WAIT states do not occur, the SN54LS32 gate can be deleted.

Table 6-1. Unused Op Code

HEXADECIMAL	BINARY			
	0 1 2 3	4 5 6 7	8 9 10 11	12 13 14 15
000—01FF	0 0 0 0	0 0 0 X	X X X X	X X X X
0320—033F	0 0 0 0	0 0 1 1	0 0 1 X	X X X X
0780—07FF	0 0 0 0	0 1 1 1	1 X X X	X X X X
0000—0FFF	0 0 0 0	1 1 X X	X X X X	X X X X

6.1.2 UNUSED OP CODE PROCESSING. The occurrence of an unused op code is an error condition which can be caused by device failure or errors in software. Therefore, the error flag (ILOPCD) generated should be used as a high-priority (low level) interrupt causing the SBP 9900A to suspend operation before any further instructions are executed. The flag should be cleared by the CLILOP signal, perhaps via the CRU, only after the cause of the error is determined.

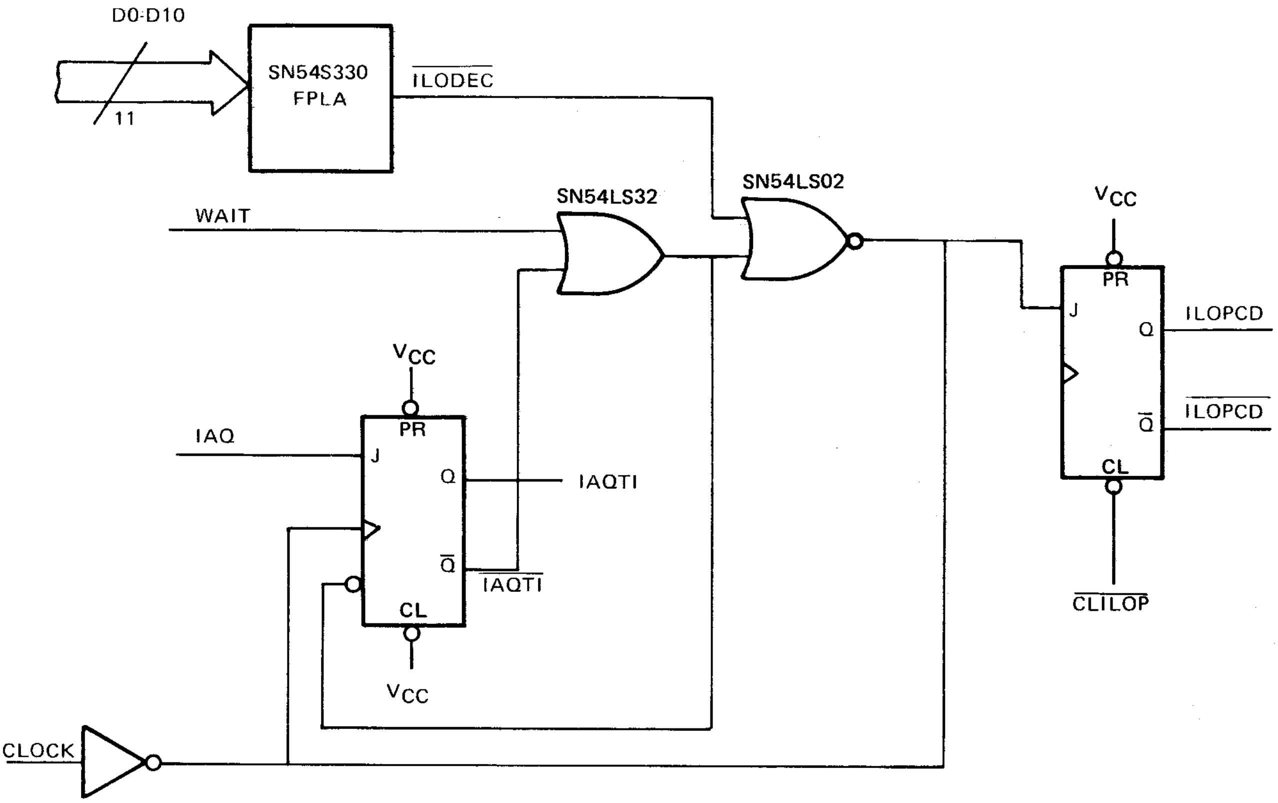


Figure 6-1. Illegal Op Code Detector Circuitry

6.2 SOFTWARE FRONT PANEL

A useful feature during hardware and software development is a means for controlling the operation of the SBP 9900A at a more basic level than is required during normal system operation. For instance, it is desirable to start and stop execution of a program segment, to execute single instructions, to inspect and to modify memory and internal registers, and to load programs into RAM. These features are typically associated with the front panel of a conventional minicomputer and are normally implemented in hardware. Similar capabilities can be provided for any SBP 9900A system very economically using a combination of hardware and software.

Figure 6-2 shows the hardware required to implement a software-controlled front panel for a SBP 9900A system. In this configuration, the $\overline{\text{LOAD}}$ function, including the $\overline{\text{LOAD}}$ vectors at addresses FFFC_{16}—FFFE_{16}, is dedicated to the front panel. Some portion of the memory space must be allocated for the control program and temporary storage. Circuitry to control generation of the $\overline{\text{LOAD}}$ input to the SBP 9900A requires approximately five devices. Finally, some means of communication between the user and the system must be provided. In this example, a front panel with switches and LED display was selected because of the simplicity of the interface logic and the overall cost-effectiveness of the design.

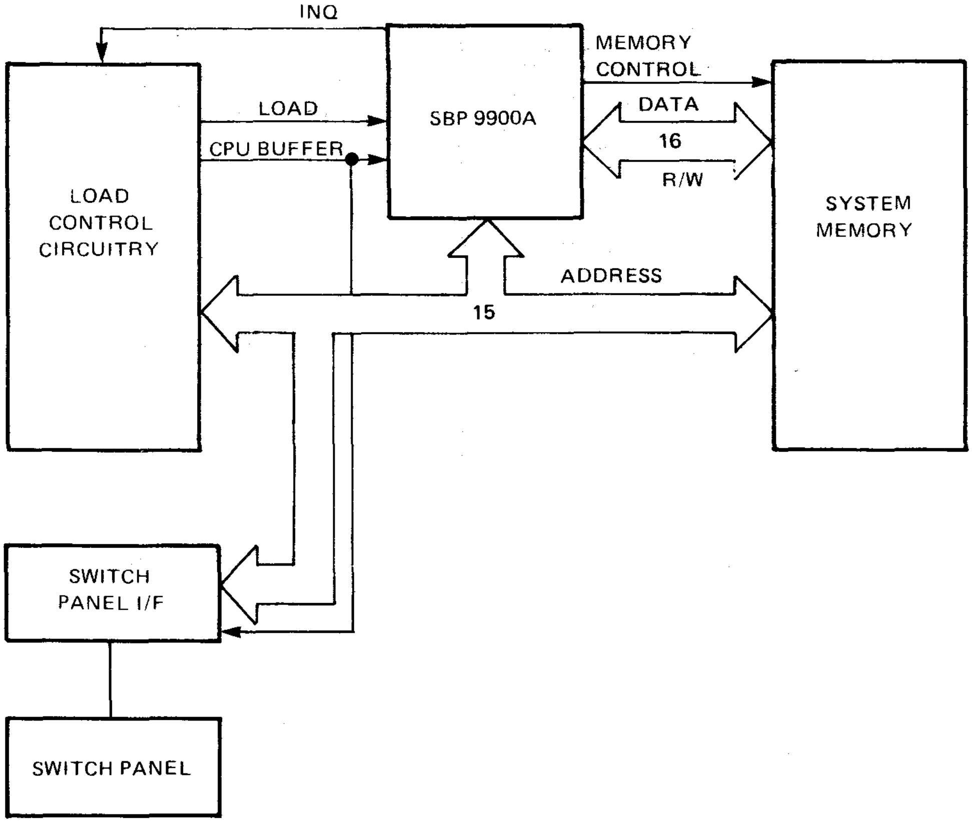


Figure 6-2. System Configuration for Software Front Panel

6.2.1 MEMORY REQUIREMENTS. As shown in Figure 6-3, the $\overline{\text{LOAD}}$ vectors at memory addresses $FFFC_{16}$ and $FFFE_{16}$ are required. It is normally convenient to use the contiguous memory area for the control program, since both will be contained in ROM or PROM. The size of the control program is a direct function of the number and sophistication of utilities provided. Control program RAM requirements are minimal and can be satisfied with system RAM or implemented separately as desired.

6.2.2 DESCRIPTION OF OPERATION. When power is applied, the system "comes up" operating in one of two modes: 1) FRONT PANEL, and 2) RUN. In the Front Panel Mode (FPMODE = 1), the SBP 9900A executes instructions under the supervision of the front panel control program. In this mode, the user can communicate with the system via the switch panel. Commands are entered through the panel and transmitted to the system, where the control program interprets the command and performs the desired operation. Commands can cause memory contents to be changed or displayed, particular programs to be executed or other control operations, such as setting up breakpoints or single instruction execution. The Front Panel Mode is entered through the $\overline{\text{LOAD}}$ function. Therefore the previous context is saved and can be restored when returning to the Run Mode (FPMODE = 0). In the Run Mode, the SBP 9900A executes instructions at any location in memory, uncontrolled by the user until the $\overline{\text{LOAD}}$ function again becomes active.

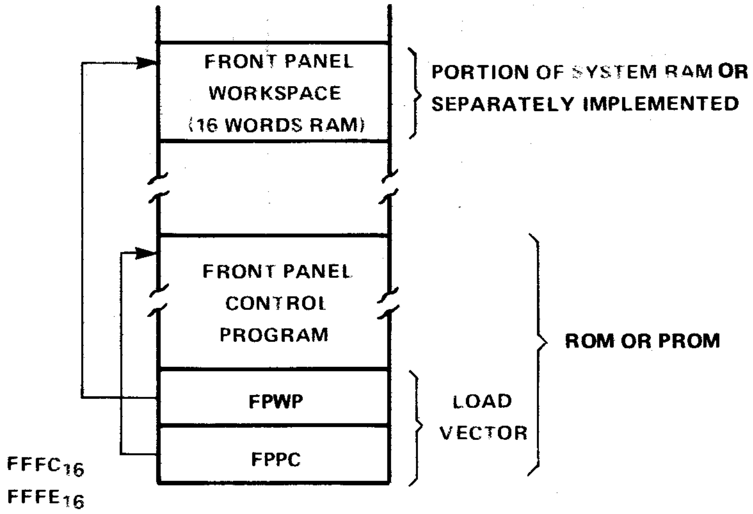


Figure 6-3. Software Front Panel Memory Requirements

6.2.2.1 Entry Into Front Panel Mode. Figure 6-4 shows the front panel control circuitry used to generate the $\overline{\text{LOAD}}$ signal to the SBP 9900A. Initially, the power-on $\overline{\text{RESET}}$ signal to the system causes FPMODE to be reset. When the user desires to gain control, he actuates a pushbutton causing $\overline{\text{HALT}}$ to be active. During the next instruction fetch by the SBP 9900A (IAQ = 1), FPMODE is set and the system enters the Front Panel Mode. If the SBP 9900A is currently executing an IDLE instruction and waiting for an unmasked interrupt to occur, FPMODE is set by the IDLE output of the SN54LS138. The $\overline{\text{LOAD}}$ signal remains active until the next instruction is fetched. The $\overline{\text{LOAD}}$ signal is therefore active for one instruction. This ensures that the $\overline{\text{LOAD}}$ function is executed, since this input is tested by the SBP 9900A at the end of each instruction cycle. It is necessary that the $\overline{\text{LOAD}}$ signal be active for no more than one instruction cycle, otherwise, the $\overline{\text{LOAD}}$ operation is repeated. Each time the $\overline{\text{LOAD}}$ function is performed, the internal registers of the SBP 9900A are saved in registers WR13-15 of the Front Panel Workspace. Thus, if the $\overline{\text{LOAD}}$ operation is repeated, the saved internal register contents would be destroyed when the internal registers are again stored. After the $\overline{\text{LOAD}}$ context switch is performed, the processor begins execution of the Front Panel Control Program.

6.2.2.2 Single Instruction Execution. Part of the circuitry in Figure 6-4 allows the Front Panel Control Program to execute single instructions in general memory and then to return to Front Panel Control. The CKOF instruction is used to initiate this operation, causing the $\overline{\text{CKOF}}$ signal to be pulsed. $\overline{\text{CKOF}}$ is gated with FPMODE and $\overline{\text{LOAD}}$ to ensure that no instructions outside the Control Program may initiate this operation. The gated CKOF signal sets SIEQ. The timing for the $\overline{\text{LOAD}}$ signal generation is shown in Figure 6-5. The instruction sequence to perform single instruction execution (SIE) is:

```
.
.
.
CKOF        SET SIEQ

RTWP        RELOAD INTERNAL REGISTERS
.
.
.
```

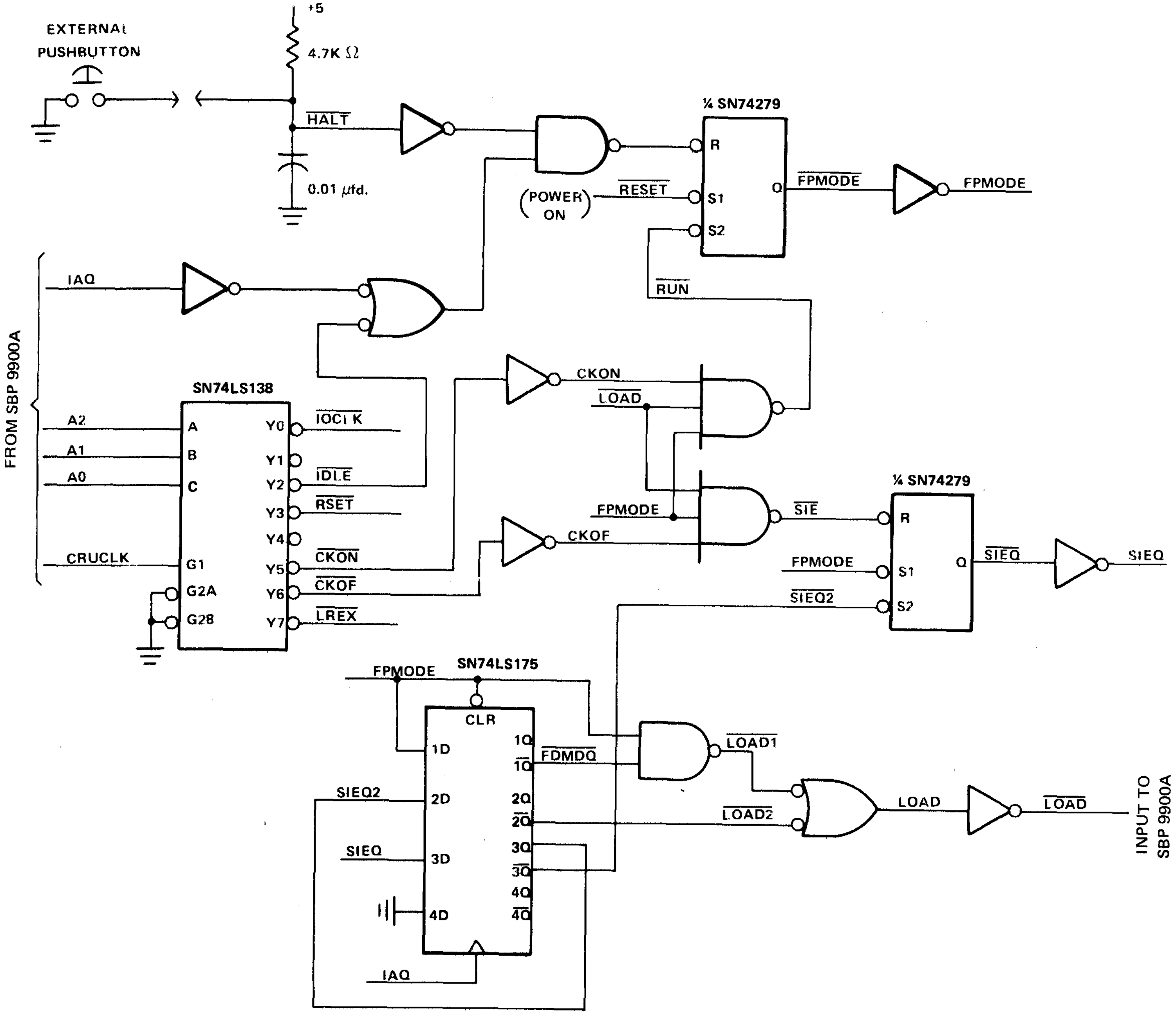


Figure 6-4. Front Panel Control Circuitry

The circuitry in Figure 6-4 causes the $\overline{\text{LOAD}}$ signal to be generated after delaying one instruction; therefore, the $\overline{\text{LOAD}}$ signal is active during the first instruction after RTWP, and control returns to the Front Panel Control Program. This SIE function can also be used to implement breakpoints and constant updates of program operation. For example, a breakpoint can be accomplished by repetitively performing SIE and comparing the saved PC with the desired value.

6.2.2.3 Return to Run Mode. With the circuitry in Figure 6-4, the SBP 9900A returns to the Run Mode by executing the following instruction sequence:

```
.
.
.
CKON          RESET FPMODE

RTWP          RELOAD INTERNAL REGISTER
.
.
.
```

The CKON pulse, gated with FPMODE and $\overline{\text{LOAD}}$, resets FPMODE and, after the RTWP instruction is executed, the SBP 9900A resumes Run Mode operation and continues until the HALT pushbutton is again pressed.

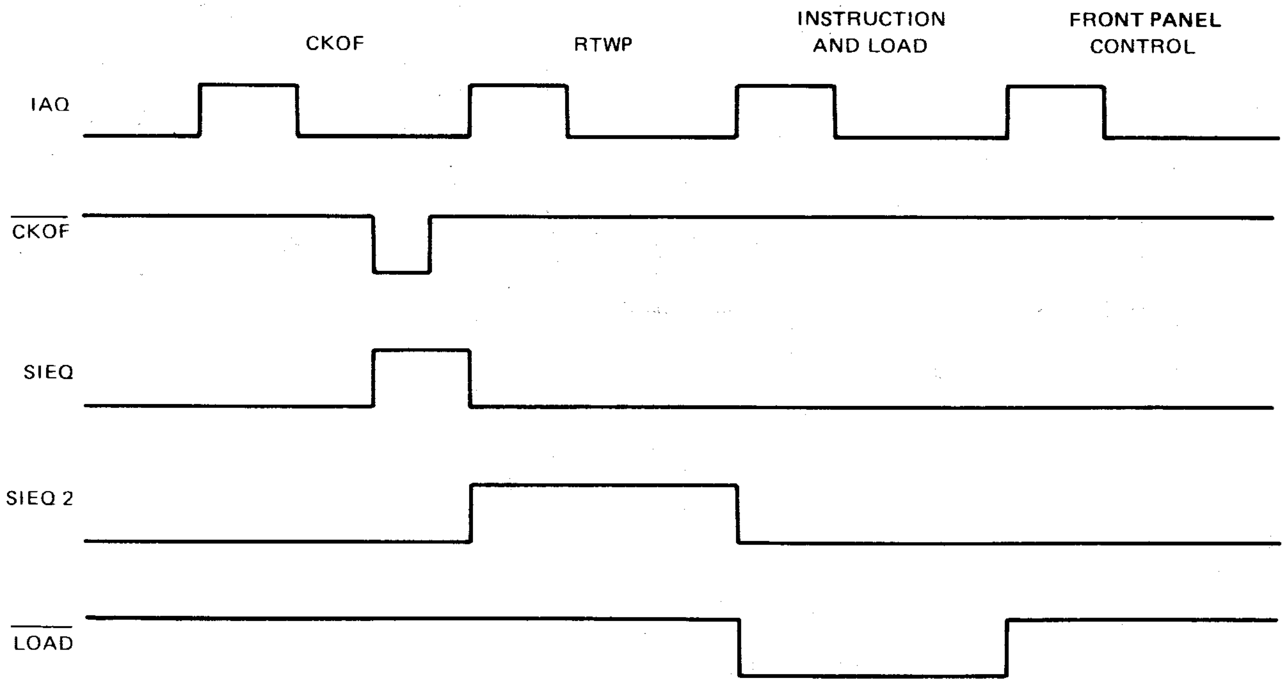


Figure 6-5. Single Instruction Execution Timing

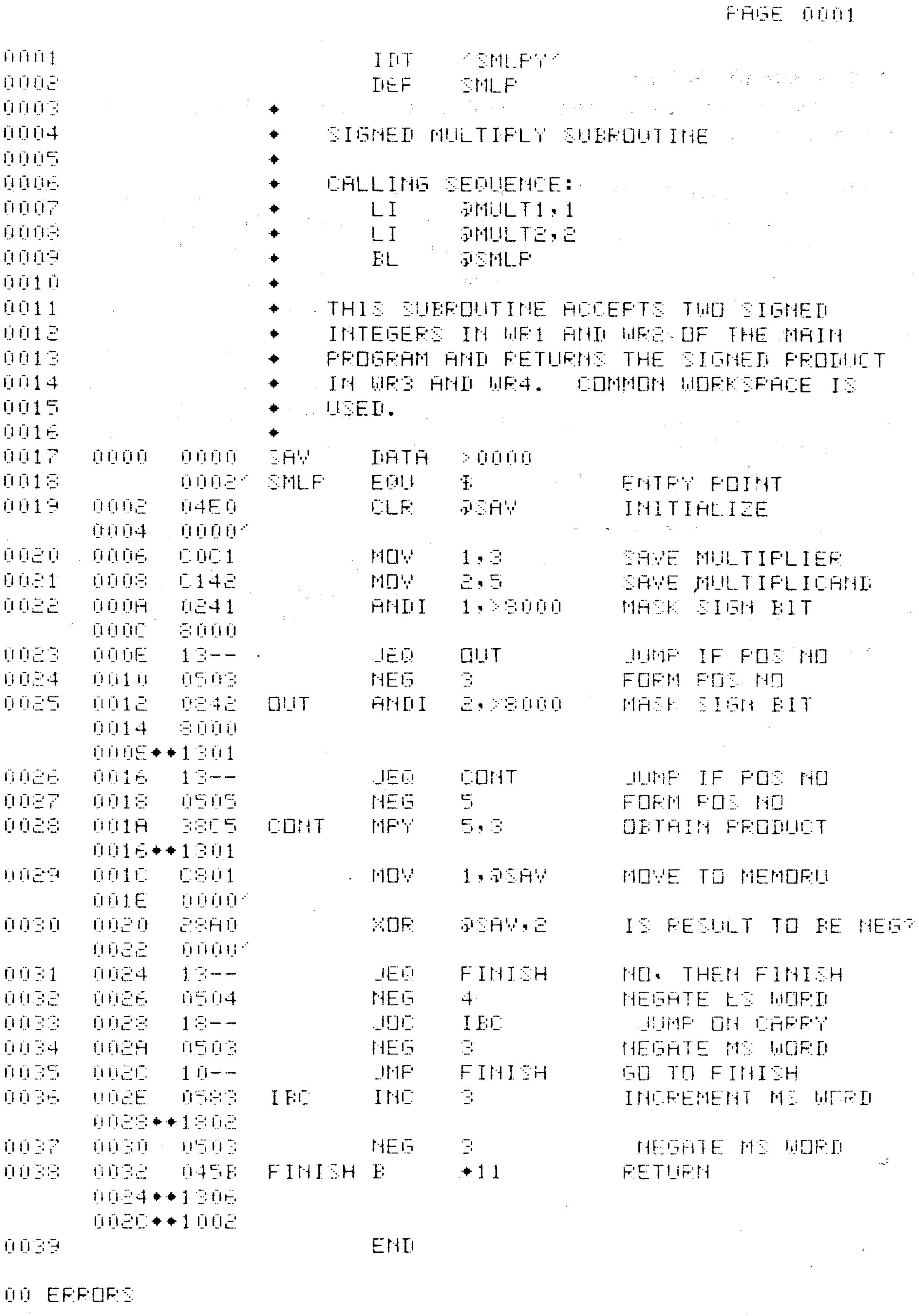
6.3 SOFTWARE EXAMPLES

The following software examples are written in 9900 Assembly Language. It is assumed that the reader is familiar with the ''Assembly Language Programmer's Guide''.

6.3.1 SIGNED MULTIPLY ROUTINE. Figure 6-6 lists the assembled source code for the signed multiply routine. This program accepts two signed integers (in WR1 and WR2 of the calling program) and sets flags 1 and 2 in accordance with the sign bit of the multiplier and multiplicand. If a number is negative, the two's complement is taken to form the positive number. After the two unsigned integers are multiplied, the sign bits are exclusive-ORed together to determine the sign of the result. If the executive-OR is equal to one, the product is negated and passed to WR3 and WR4 of the calling programs workspace.

6.3.2 SIGNED DIVIDE ROUTINE. Figure 6-7 lists the assembled source code for the signed divide routine. This program is similar to the signed multiply routine in that the exclusive-OR of the sign bits determines the sign of the quotient. If the divisor is less than or equal to the unsigned dividend, ST4 is set indicating an overflow. In this case the remainder is set equal to −1 and is passed to the calling program.

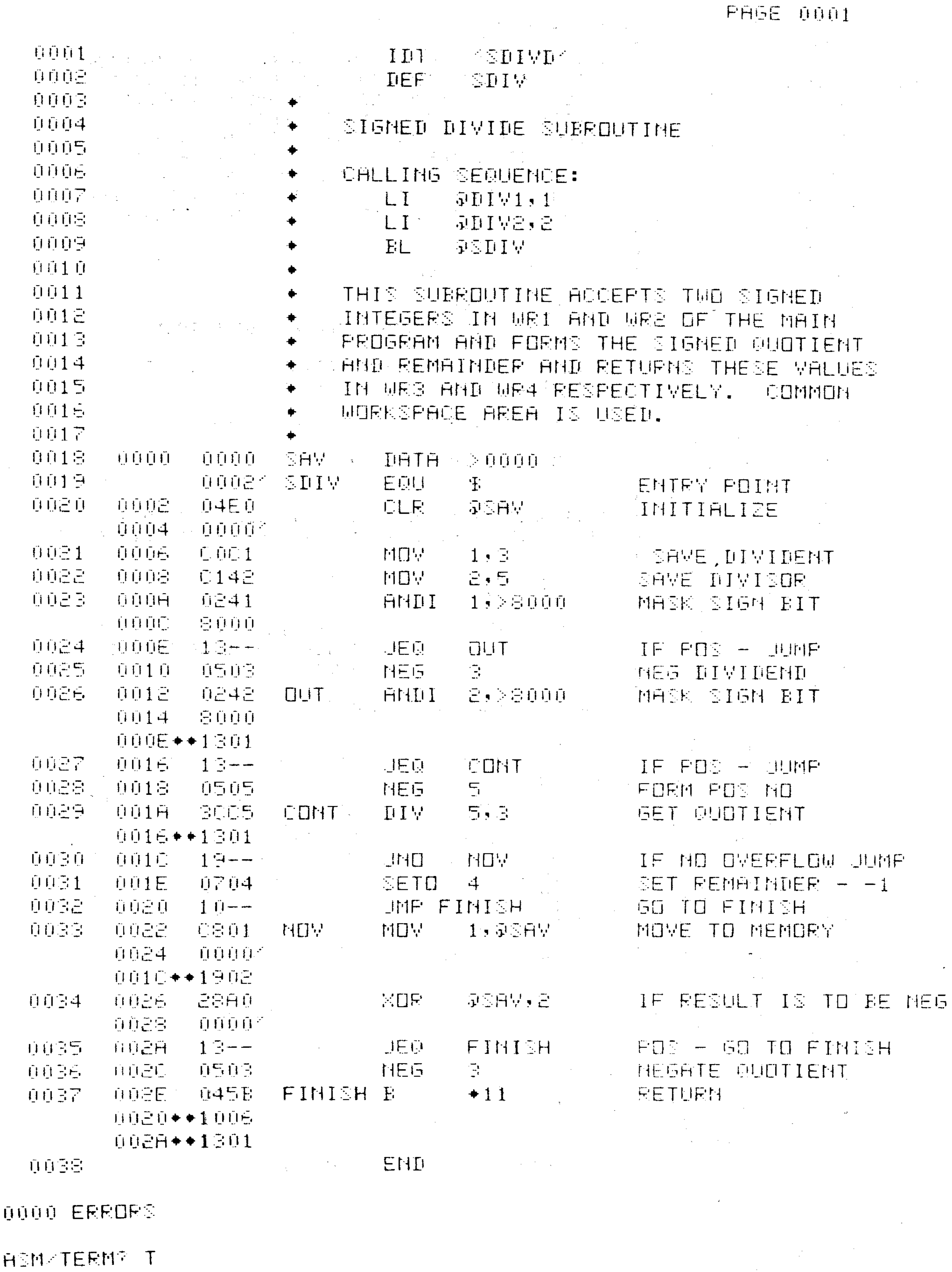
6.3.3 INTEGER SQUARE ROOT ROUTINE. Figure 6-8 lists the assembled source code for an integer square root routine. This subroutine computes the square root using the ''sum of the odd digits'' method. This method adds the odd digits until the sum is greater than or equal to the number for which the square root is desired. The number of odd digits added is the integer square root. If the sum is greater than the number, then minus 1 must be added to the integer square root.



```
                                                                    PAGE 0001
0001                              IDT    'SMLPY'
0002                              DEF    SMLP
0003                    *
0004                    *   SIGNED MULTIPLY SUBROUTINE
0005                    *
0006                    *   CALLING SEQUENCE:
0007                    *       LI     @MULT1,1
0008                    *       LI     @MULT2,2
0009                    *       BL     @SMLP
0010                    *
0011                    *   THIS SUBROUTINE ACCEPTS TWO SIGNED
0012                    *   INTEGERS IN WR1 AND WR2 OF THE MAIN
0013                    *   PROGRAM AND RETURNS THE SIGNED PRODUCT
0014                    *   IN WR3 AND WR4.  COMMON WORKSPACE IS
0015                    *   USED.
0016                    *
0017   0000   0000   SAV      DATA   >0000
0018          0002'  SMLP     EQU    $           ENTRY POINT
0019   0002   04E0            CLR    @SAV        INITIALIZE
       0004   0000'
0020   0006   C0C1            MOV    1,3         SAVE MULTIPLIER
0021   0008   C142            MOV    2,5         SAVE MULTIPLICAND
0022   000A   0241            ANDI   1,>8000     MASK SIGN BIT
       000C   8000
0023   000E   13--            JEQ    OUT         JUMP IF POS NO
0024   0010   0503            NEG    3           FORM POS NO
0025   0012   0242   OUT      ANDI   2,>8000     MASK SIGN BIT
       0014   8000
       000E**1301
0026   0016   13--            JEQ    CONT        JUMP IF POS NO
0027   0018   0505            NEG    5           FORM POS NO
0028   001A   38C5   CONT     MPY    5,3         OBTAIN PRODUCT
       0016**1301
0029   001C   C801            MOV    1,@SAV      MOVE TO MEMORU
       001E   0000'
0030   0020   28A0            XOR    @SAV,2      IS RESULT TO BE NEG?
       0022   0000'
0031   0024   13--            JEQ    FINISH      NO, THEN FINISH
0032   0026   0504            NEG    4           NEGATE LS WORD
0033   0028   18--            JOC    IBC          JUMP ON CARRY
0034   002A   0503            NEG    3           NEGATE MS WORD
0035   002C   10--            JMP    FINISH      GO TO FINISH
0036   002E   0583   IBC      INC    3           INCREMENT MS WORD
       0028**1802
0037   0030   0503            NEG    3            NEGATE MS WORD
0038   0032   045B   FINISH   B      *11         RETURN
       0024**1306
       002C**1002
0039                              END

00 ERRORS
```

Figure 6-6. Signed Multiply Routine

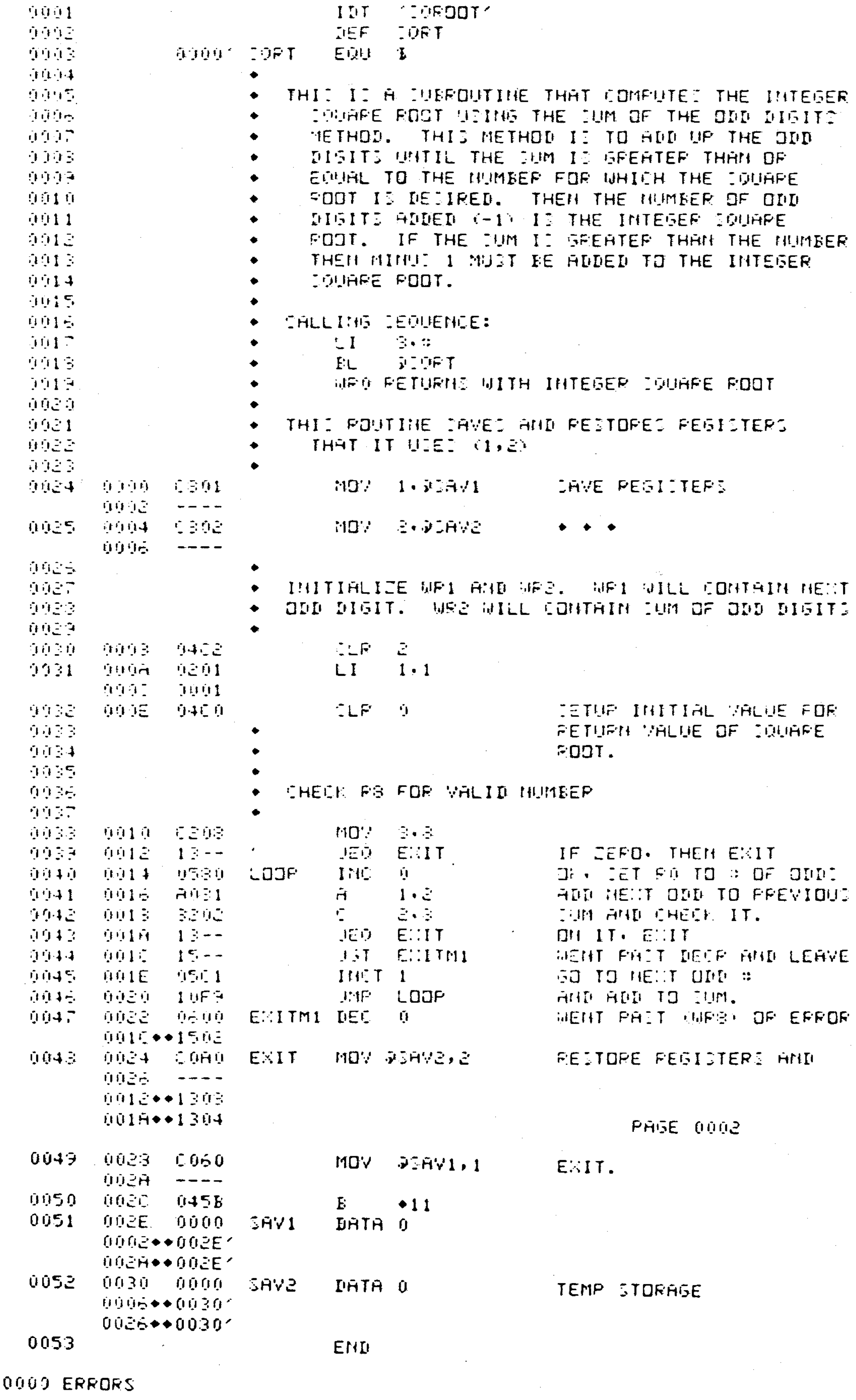


```
                                                                    PAGE 0001
0001                                IDT    'SDIVD'
0002                                DEF    SDIV
0003                         *
0004                         *    SIGNED DIVIDE SUBROUTINE
0005                         *
0006                         *    CALLING SEQUENCE:
0007                         *       LI    @DIV1,1
0008                         *       LI    @DIV2,2
0009                         *       BL    @SDIV
0010                         *
0011                         *    THIS SUBROUTINE ACCEPTS TWO SIGNED
0012                         *    INTEGERS IN WR1 AND WR2 OF THE MAIN
0013                         *    PROGRAM AND FORMS THE SIGNED QUOTIENT
0014                         *    AND REMAINDER AND RETURNS THESE VALUES
0015                         *    IN WR3 AND WR4 RESPECTIVELY.  COMMON
0016                         *    WORKSPACE AREA IS USED.
0017                         *
0018  0000   0000   SAV     DATA   >0000
0019         0002'  SDIV    EQU    $                 ENTRY POINT
0020  0002   04E0           CLR    @SAV              INITIALIZE
      0004   0000'
0021  0006   C0C1           MOV    1,3               SAVE,DIVIDENT
0022  0008   C142           MOV    2,5               SAVE DIVISOR
0023  000A   0241           ANDI   1,>8000           MASK SIGN BIT
      000C   8000
0024  000E   13--           JEQ    OUT               IF POS - JUMP
0025  0010   0503           NEG    3                 NEG DIVIDEND
0026  0012   0242   OUT     ANDI   2,>8000           MASK SIGN BIT
      0014   8000
      000E**1301
0027  0016   13--           JEQ    CONT              IF POS - JUMP
0028  0018   0505           NEG    5                 FORM POS NO
0029  001A   3CC5   CONT    DIV    5,3               GET QUOTIENT
      0016**1301
0030  001C   19--           JNO    NOV               IF NO OVERFLOW JUMP
0031  001E   0704           SETO   4                 SET REMAINDER - -1
0032  0020   10--           JMP FINISH               GO TO FINISH
0033  0022   C801   NOV     MOV    1,@SAV            MOVE TO MEMORY
      0024   0000'
      001C**1902
0034  0026   28A0           XOR    @SAV,2            IF RESULT IS TO BE NEG
      0028   0000'
0035  002A   13--           JEQ    FINISH            POS - GO TO FINISH
0036  002C   0503           NEG    3                 NEGATE QUOTIENT
0037  002E   045B   FINISH  B      *11               RETURN
      0020**1006
      002A**1301
0038                        END

0000 ERRORS

ASM/TERM? T
```

Figure 6-7. Signed Divide Routine



```
                                IDT   'SQROOT'
                                DEF   SQRT
             0000'  SQRT        EQU   $
                    *
                    *  THIS IS A SUBROUTINE THAT COMPUTES THE INTEGER
                    *     SQUARE ROOT USING THE SUM OF THE ODD DIGITS
                    *     METHOD.  THIS METHOD IS TO ADD UP THE ODD
                    *     DIGITS UNTIL THE SUM IS GREATER THAN OR
                    *     EQUAL TO THE NUMBER FOR WHICH THE SQUARE
                    *     ROOT IS DESIRED.  THEN THE NUMBER OF ODD
                    *     DIGITS ADDED (-1) IS THE INTEGER SQUARE
                    *     ROOT.  IF THE SUM IS GREATER THAN THE NUMBER
                    *     THEN MINUS 1 MUST BE ADDED TO THE INTEGER
                    *     SQUARE ROOT.
                    *
                    *  CALLING SEQUENCE:
                    *        LI    8,#
                    *        BL    @SQRT
                    *        WR0 RETURNS WITH INTEGER SQUARE ROOT
                    *
                    *  THIS ROUTINE SAVES AND RESTORES REGISTERS
                    *     THAT IT USES (1,2)
                    *
  0000   C801                   MOV   1,@SAV1         SAVE REGISTERS
  0002   ----
  0004   C802                   MOV   2,@SAV2         * * *
  0006   ----
                    *
                    *  INITIALIZE WR1 AND WR2.  WR1 WILL CONTAIN NEXT
                    *  ODD DIGIT.  WR2 WILL CONTAIN SUM OF ODD DIGITS
                    *
  0008   04C2                   CLR   2
  000A   0201                   LI    1,1
  000C   0001
  000E   04C0                   CLR   0               SETUP INITIAL VALUE FOR
                    *                                 RETURN VALUE OF SQUARE
                    *                                 ROOT.
                    *
                    *  CHECK R8 FOR VALID NUMBER
                    *
  0010   C208                   MOV   8,8
  0012   13--                   JEQ   EXIT            IF ZERO, THEN EXIT
  0014   0580       LOOP        INC   0               OK, SET R0 TO # OF ODDS
  0016   A081                   A     1,2             ADD NEXT ODD TO PREVIOUS
  0018   8202                   C     2,8             SUM AND CHECK IT.
  001A   13--                   JEQ   EXIT            ON IT, EXIT
  001C   15--                   JGT   EXITM1          WENT PAST DECR AND LEAVE
  001E   05C1                   INCT  1               GO TO NEXT ODD #
  0020   10F9                   JMP   LOOP            AND ADD TO SUM.
  0022   0600       EXITM1      DEC   0               WENT PAST (WR8) OR ERROR
  001C**1503
  0024   C0A0       EXIT        MOV @SAV2,2           RESTORE REGISTERS AND
  0026   ----
  0012**1303
  001A**1304
                                                            PAGE 0002

  0028   C060                   MOV   @SAV1,1         EXIT.
  002A   ----
  002C   045B                   B     *11
  002E   0000       SAV1        DATA  0
  0002**002E'
  002A**002E'
  0030   0000       SAV2        DATA  0               TEMP STORAGE
  0006**0030'
  0026**0030'
                                END

0000 ERRORS

ASM/TERM? T
```

Figure 6-8. Integer Square Root Routine

SECTION VII
POWER SOURCE CONSIDERATIONS

7.1 I^2L OPERATION

The lowest common denominator I^2L logic function, the inverter gate, is comprised of a multiple collector NPN switching transistor with an associated continuously "on" PNP injection-current-source transistor as shown in Figure 7-1. The multiple collector configuration of the NPN switching transistor provides the signal isolation through which generation of higher level logic functions is accomplished. For example, the positive NOR logic function is easily generated by wire-ORing selected collectors from several NPN transistors as shown in Figure 7-2. From a different point of view,the positive NAND logic function is generated by considering dot-AND logic to be performed at the base of the inverting NPN transistor as shown in Figure 7-3. In either case, the monolithic implementation of I^2L merges (i.e. Merged Transistor Logic) the active regions of the associated PNP-NPN transistor pair such that each NPN transistor is considered equivalent to one gate regardless of input fan-in.

7.1.1 I^2L CURRENT UTILIZATION. Supply current (dc) of an externally adjusted magnitude enters the SBP 9900A through four user-shorted INJ terminals. The supply current is balanced internally in the SBP 9900A through computer specified integrated resistors, and is distributed to the emitters of the various PNP injection-current-source transistors. The collector of each continuously "on" PNP transistor then sources injection current which governs the switching action of each associated NPN transistor (gate) through controlled injection current steering. Simply stated, if the injection current is steered into the base of the NPN transistor-gate, the gate turns "on" forcing the outputs (collectors) to logic-level low. If the injection current is steered away from the base of the NPN transistor-gate, the gate turns "off" allowing the outputs to be pulled to logic-level high. In either case, the magnitude of the injection current remains constant. As a direct result, I^2L is considered to be non-noise-producing.

7.1.2 CURRENT STEERING CONTROL. Current steering is controlled by adjusting the base-to-emitter NPN transistor-gate input voltage V_{BE}. A logic-level low input voltage of less than one V_{BE} (750 mV) pulls the injection current away from the base (input) of the NPN transistor-gate through the "on" (low) output of the driving NPN transistor-gate as shown in Figure 7-4A. Deprived of its base drive, the transistor-gate turns "off" allowing the outputs (collectors) to be pulled high to the clamp voltage levels (750 mV) at the input of the next transistor-gates. A logic-level high input voltage of greater than or equal to one V_{BE} (750 mV) is achieved by default when the driving transistor-gate is "off" as shown in Figure 7-4B. In this case, the injection current, deprived of a low impedance path of less than one V_{BE} potential, is steered into the base of the transistor-gate. With its base driven with injection current, the transistor-gate turns "on" forcing the outputs (collectors) to logic-level low-typically one V_{SAT} (50 mV) above substrate ground.

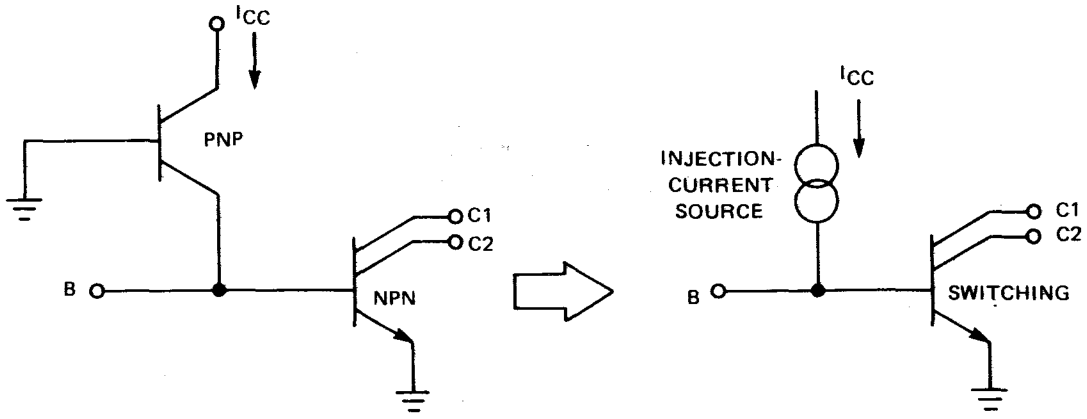


Figure 7-1. Basic I^2L Converter

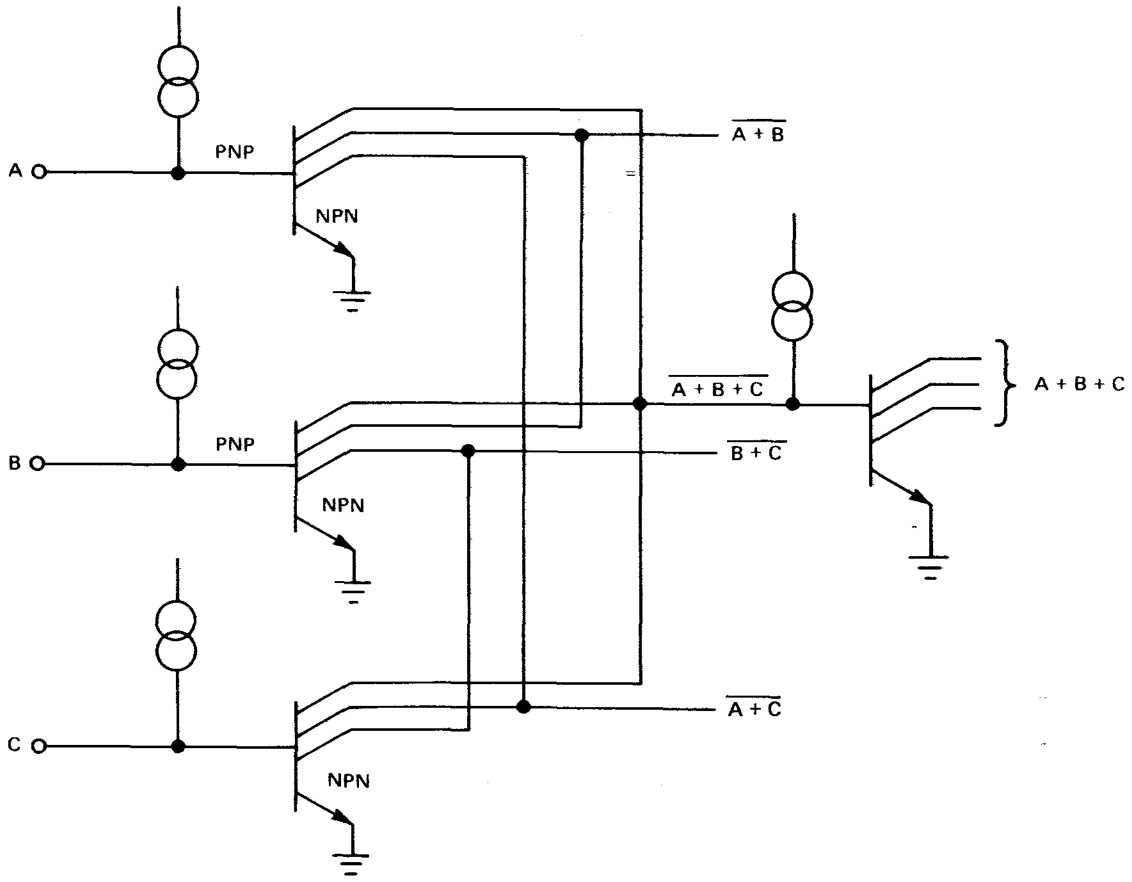


Figure 7-2. I^2L Positive NOR Functions

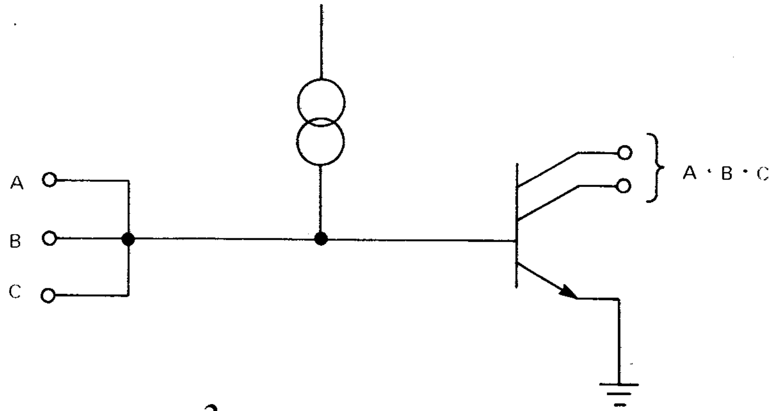
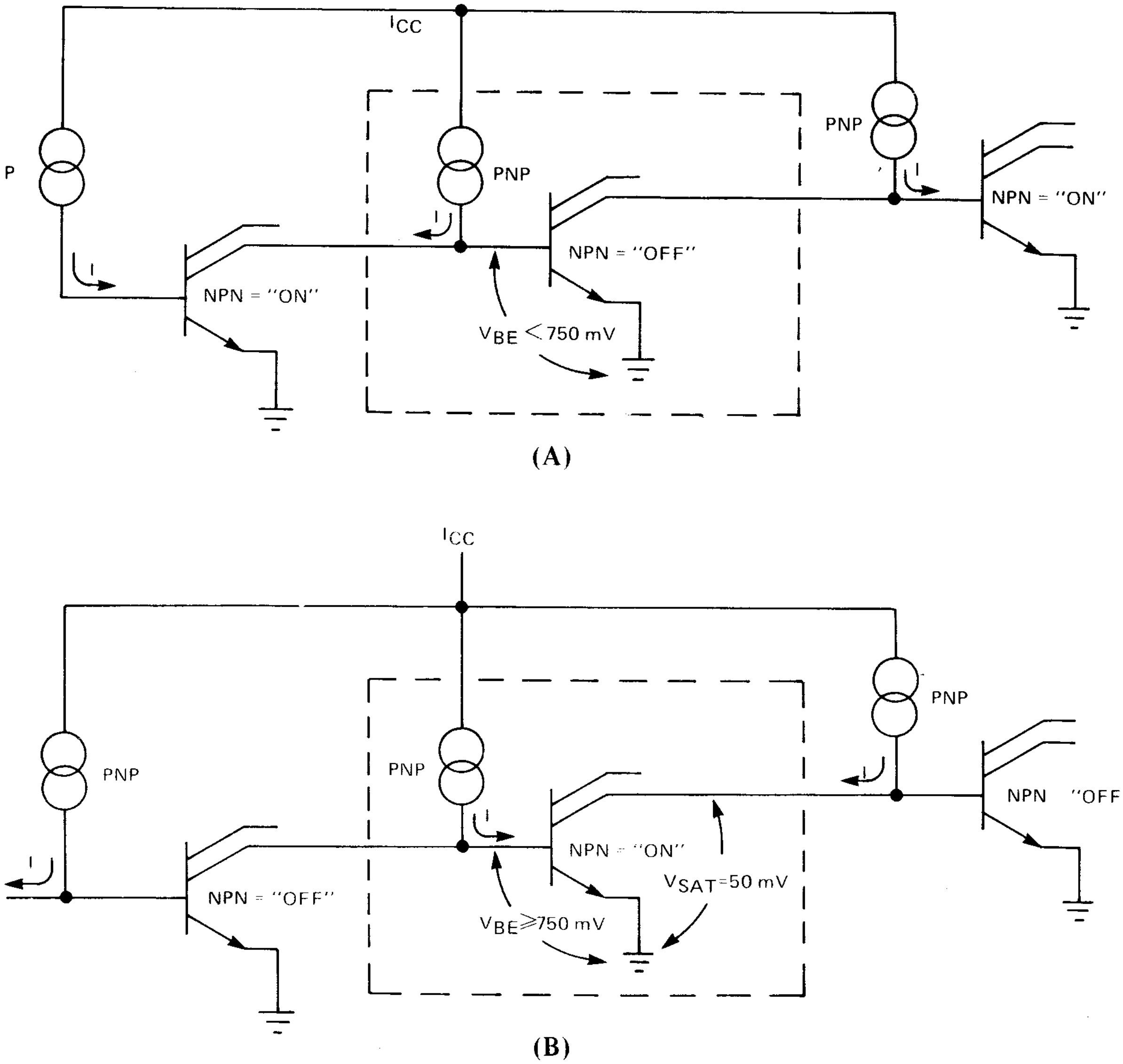


Figure 7-3. I^{2}L Positive NAND Function



(A)

(B)

Figure 7-4. Current Sterring Control

7.2 NODE VOLTAGE (V_{INJ}) VARIATIONS, SUPPLY CURRENT (INJ) TERMINALS

NOTE

The discussion contained within this section is based on calculated data and is therefore presented only to acquaint the reader with the basic mechanisms which affect the node voltage of the SBP 9900As supply current terminals. Actual data will be used in Section 7.3 to develop an SBP 9900A current supply.

The node voltage of the SBP 9900As user-shorted supply current (INJ) terminals varies with: a) ambient temperature b) applied current c) normal fluctuations in the semiconductor fabrication process. When placed across a curve tracer, the SBP 9900As INJ terminals demonstrate voltage-current characteristics, as shown in Figure 7-5,which resemble those of a collection of silicon PN diodes connected in parallel. Note that at $T_A = 25°C$, the piece-wise linear portion of Figure 7-5 intersects the "voltage" axis at 800 mV. This value of 800 mV increases/decreases, respectively, with decreasing/increasing ambient temperature at a linear rate of approximatey 0.71 mV/°C. Therefore, at $T_A = 125°C$, the calculated value of intersection becomes 729 mV where:

$$125°C - 25°C = |100°C|$$

$$100°C \text{ x } 0.71 \text{ mV/°C} = 71 \text{ mV}$$

and

$$800 \text{ mV} - 71 \text{ mV} = 729 \text{ mV}.$$

At $T_A = -55°C$, the calculated value of intersection becomes 857 mV where

$$-55°C - 25°C = |80°C|,$$

$$80°C \times 0.71 \text{ mV/°C} = 57 \text{ mV},$$

and

$$800 \text{ mV} + 57 \text{ mV} = 857 \text{ mV}.$$

With a cumulative dc supply current magnitude of 500 mA entering the SBP 9900A's INJ terminals, the INJ terminal voltage is typically 1.175 volts at $T_A = 25°C$. As a result, the resistive component (r) of the SBP 9900A's INJ terminal impedance can be calculated to be 0.75 Ω where

$$1.175 \text{ V} - 0.8 \text{ V} = 0.374 \text{ V}$$

and

$$0.375 \text{ V} \div 0.5 \text{ a} = 0.75 \,\Omega.$$

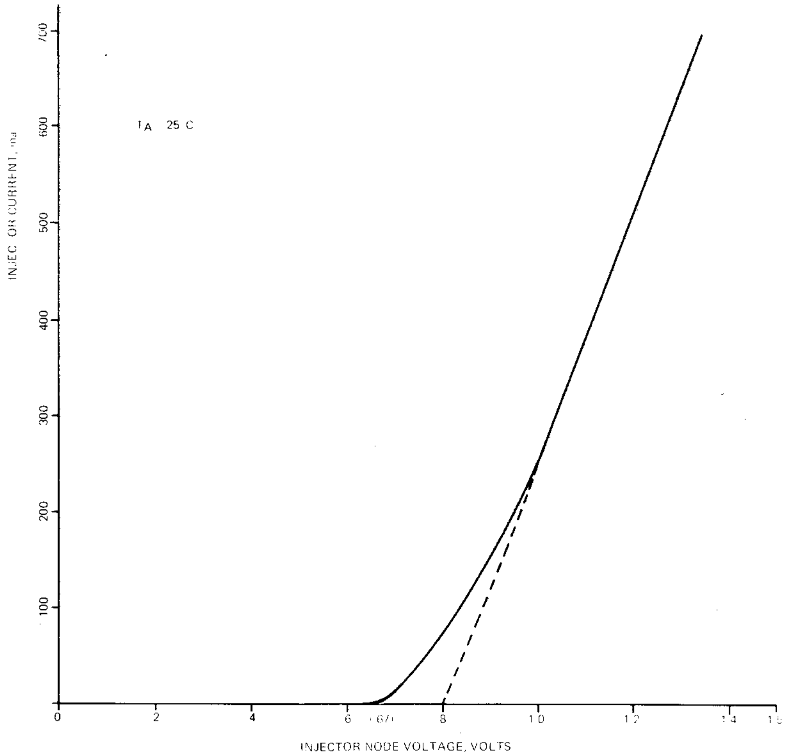


Figure 7-5. SBP 9900A Resembles a Silicon Switching Diode

Due to normal fluctuations in most semiconductor fabrication processes, "r" can vary as much as ±30%. This variance, assumed to be independent of temperature, contains "r" to the range $0.577\ \Omega \leq r \leq 0.975\ \Omega$ where

$$f\,(r = +\,30\%) = 1.3r = 1.3\,(.75) = 0.975\ \Omega$$

and

$$f\,(r = -\,30\%) = (1-3)r = .7\,(.75) = .525\ \Omega$$

Superimposing the calculated extremes of "r" upon the calculated, temperature related, piece-wise linear, diode characteristics of Figure 7-6 suggest the range within which the SBP 9900A's INJ-terminal node voltage *could* vary depending on the cumulative effects of: a) ambient temperature, and b) normal fluctuations in the semiconductor fabrication process. At a supply current magnitude of 500 mA, V_{INJ} is therefore contained within the range $.9915\ V \leq V_{INJ} \leq 1.345\ V$ where

$$f\,(T_A = -55°C, r = -30\%) = .9915\ V$$

and

$$f\,(T_A = +125°C, r = +30\%) = 1.345\ V$$

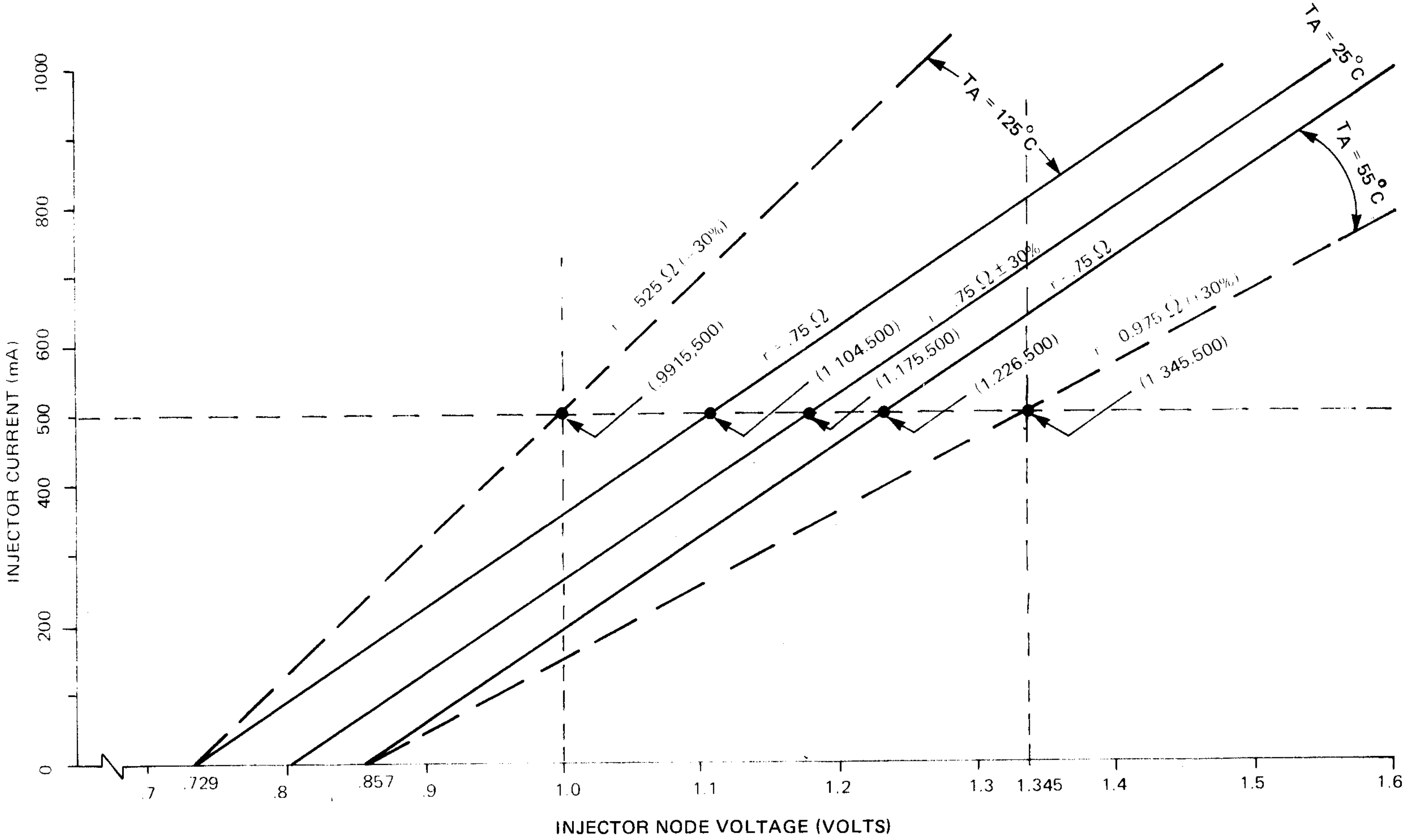


Figure 7-6. Effects of Temperature and Process Fluctuations

7.3 SBP 9900A RESISTOR CURRENT SUPPLY

A passive resistor used in conjunction with a dc voltage source can be used to configure an SBP 9900A current supply as shown in Figure 7-7. The dc voltage source can be any value capable of forcing the desired current (I_{CC}) through the resistor (R). For a desired supply current, the value of R can be calculated from the Ohm's Law formula:

$$R = \frac{V - V_{INJ}}{I_{CC}}$$

For example, a TTL voltage source can be used, as shown in Figure 7-8, with the following assumptions:

a. $V_{TTL} = 5\ V \pm 4\%$ or $4.8\ V \leq V_{TTL} \leq 5.2\ V$,

b. $I_{CC} = 500\ mA \pm 10\%$ or $450\ mA \leq I_{CC} \leq 550\ mA$,

c. R has a tolerance of ± 1% and

d. $1.055\ V \pm V_{INJ} \pm 1.285\ V$ where

$$1.055 = f\,(T_A = 125°C, r = r - 30\%)$$

$$1.285 = f\,(T_A = -55°C, r = r + 30\%)$$

Note that the values for V_{INJ} are based on actual SBP 9900A, temperature extreme, characterization data with a 100 mV delta of additional confidence. To source a desired supply current (I_{CC}) in the range of 500 mA ± 10%, the extreme limits of V_{TTL} and V_{INJ} must be considered such that the minimum voltage drop across R is developed. Therefore the calculation for R becomes:

$$R = \frac{V_{TTL} - V_{INJ}}{I_{CC}}$$

$$R = \frac{4.8 - 1.285}{0.452} = 7.7\ \Omega.$$

Once R is fixed at 7.7 Ω ±, the effect that the maximum voltage drop across R has on the value of I_{CC} must also be considered as follows:

$$I_{CC} = \frac{V_{TTL} - V_{INJ}}{R}$$

$$I_{CC} = \frac{5.2 - 1.055}{7.7} = 544\ mA.$$

When considering the effects of a ± 1% tolerance on R, supply current is contained within the range $451.97\ mA \leq I_{CC} \leq 543.75\ mA$ under the varying conditions of: a) supply voltage, b) ambient temperature, and c) SBP 9900A impedance. This range of supply current is completely compatible with sustaining the SBP 9900A's clock at a rate of dc through 3 MHz over ambient temperatures ranging from −55°C through +125°C.

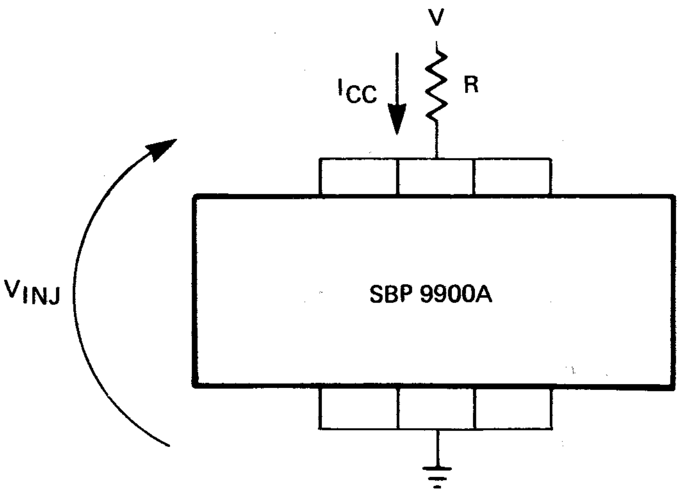


Figure 7-7. SBP 9900A Current Supply

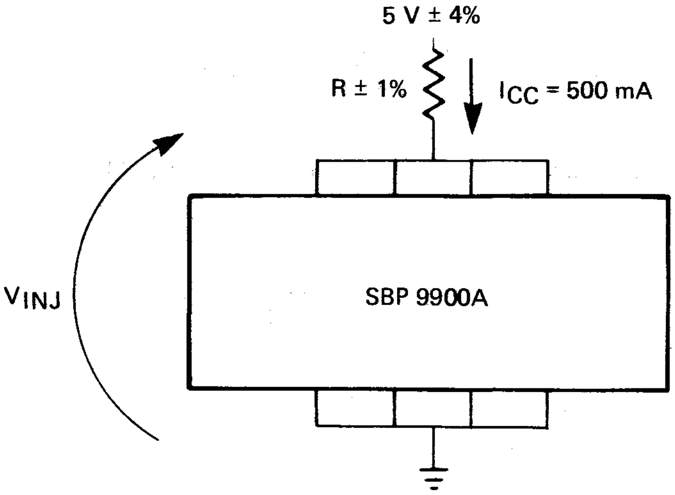


Figure 7-8. TTL Voltage Source

7.3.1 TEMPERATURE EFFECTS ON THE RESISTANCE OF "R". Ambient temperature effects the resistance of R in a manner dependent upon the composition of R. Therefore proper selection of the composition of R may be used to advantage when R is utilized in the configuration of an SBP 9900A current supply. For example, a wirewound resistor with a wire composition of 77% copper and 23% nickle offers a resistive, positive, temperature coefficient (TC) of 180 ± 30 parts per million (PPM) per degree centigrade over ambient temperatures ranging from −65°C through +150°C. Consequently, the positive temperature coefficient of this wirewound resistor can be used to somewhat offset the SBP 9900A's negative temperature coefficient.

Any resistor of whatever composition, has both an ''off'' and ''on'' ohmic value. The resistor ''off'' value, measured by forcing a trickle current through the resistor, is usually the specified ohmic value used for purposes of procurement. The resistor ''on'' value is that value obtained when application specified current is passed through the resistor. For example, a wirewound 7.5 Ω ± 1% resistor passing a current of 500 mA at T_A = 25°C experiences a temperature rise of 75°C. This temperature rise of 75°C must be considered with respect to the given temperature coefficient (TC) in determining the value of R_{ON}.

7.3.2 CALCULATIONS FOR R_{ON}. Assumptions:

a. R_{OFF} = 7.5 Ω ± 1% @ 25°C,

b. TC = +180 ± 30 PPM/°C,

c. R_{ON} Temperature Rise = 75°C and

d. I_{CC} = 500 mA a T_A = 25°C

TC = +180 − 30 PPM/°C	TC = +180 + 30 PPM/°C
TC = +150 PPM/°C = 0.00015/°C	TC = 210 PPM/°C = 0.00021/°C
When T_A = 25°C, Δ = 0.00015/°C × 75°C = 0.01125	When T_A = 25°C, Δ = 0.00021/°C × 75°C = 0.01575
R_{ON} = (1 + Δ) R_{OFF} = (1.01125) 7.5 R_{ON} = 7.584 Ω a T_A = 25°C	R_{ON} = (1 + Δ) R_{OFF} = (1.01575) 7.5 R_{ON} = 7.618 Ω a T_A = 25°C
When T_A = 125°C R_{ON} = 1.015 (7.584) = 7.698 Ω	When T_A = 125°C R_{ON} = 1.015 (7.618) = 7.732 Ω
When T_A = −55°C, R_{ON} = 0.988 (7.584) = 7.493 Ω	When T_A = −55°C, R_{ON} = 0.988 (7.618) = 7.527 Ω

The effects which the values of R_{ON} have on the SBP 9900A's supply current can be calculated under the following assumptions:

a. $V_{TTL} = 5\text{ V} \pm 5\%$ or $4.75\text{ V} \le V_{TTL} \le 5.25\text{ V}$,

b. $I_{CC} = 500\text{ mA} \pm 10\%$ or $450\text{ mA} \le I_{CC} \le 550\text{ mA}$,

c. R_{ON} has a tolerance of ± 1%,

d. R is a wirewound resistor with a 77% copper and 23% nickle composition and

e. $1.055\text{ V} \le V_{INJ} \le 1.285\text{ V}$ where

$$1.055 = f(T_A = 125°\text{C}, r = r - 30\%)$$

$$1.285 = f(T_A = -55°\text{C}, r = r + 30\%)$$

Again note that the values for V_{INJ} are based on actual SBP 9900A, temperature extreme characterization data with a 100 mV delta of additional confidence. To source a desired supply current (I_{CC}) in the range of 500 mA ± 10%, the extreme limits of V_{TTL} and V_{INJ} must again be considered with respect to developing the minimum and maximum voltage drops across R. Since the maximum value of V_{INJ} occurs at $T_A = -55°\text{C}$, the maximum value for R at $T_A = -55°\text{C}$ plus the minimum value of V_{TTL} must be considered in calculating the lower extreme of I_{CC} as follows:

$$I_{CC} = \frac{V_{TTL} - V_{INJ}}{R + 1\%}$$

$$= \frac{4.75 - 1.285}{7.527 + 1\%} = \frac{4.195}{7.622}$$

$$I_{CC} = 456\text{ mA}$$

Since the minimum value of V_{INJ} occurs at $T_A = 125°\text{C}$, the minimum value of R at $T_A = 125°\text{C}$ plus the maximum value of V_{TTL} must be considered in calculating the upper extreme of I_{CC} as follows:

$$I_{CC} = \frac{V_{TTL} - V_{INJ}}{R - 1\%}$$

$$= \frac{5.25 - 1.055}{7.698 - 1\%} = \frac{4.195}{7.622}$$

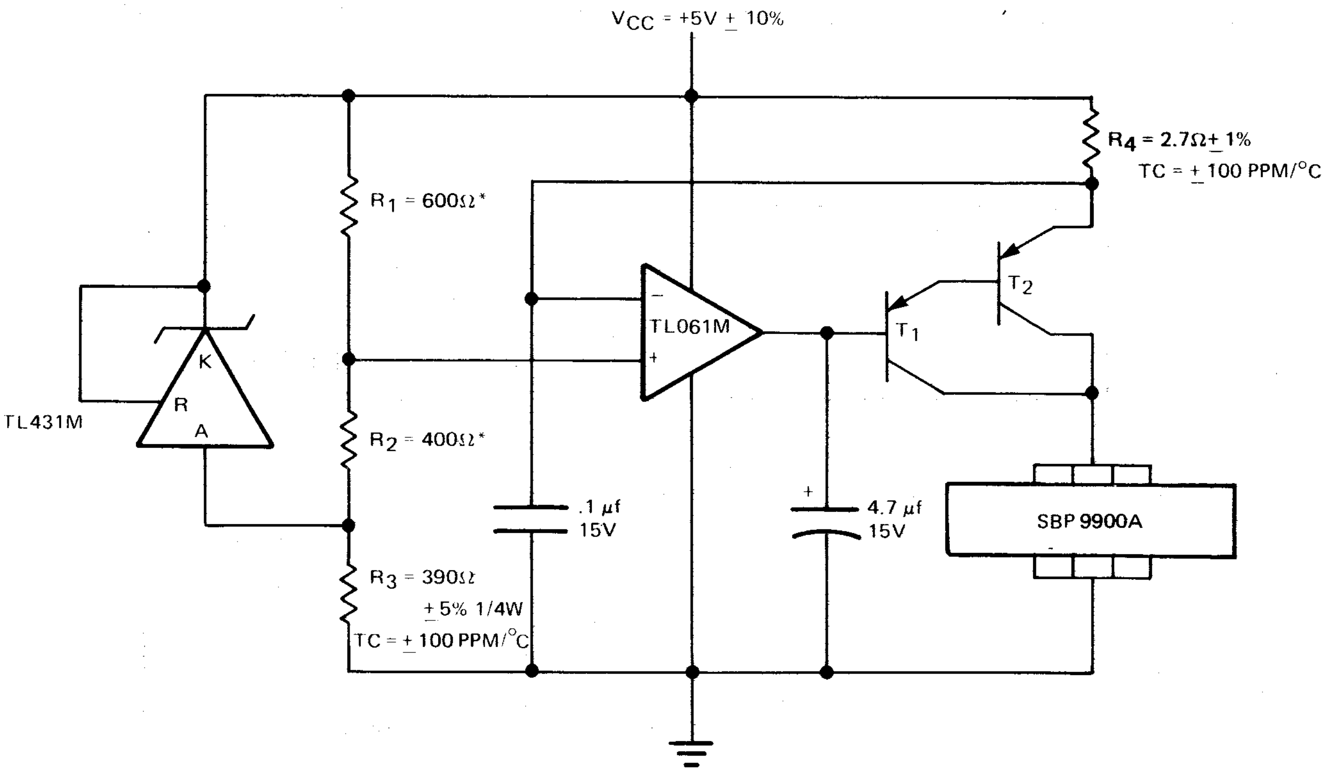
$$I_{CC} = 500\text{ mA}$$

Therefore, the SBP 9900A's supply current is contained within the range $456\text{ mA} \le I_{CC} \le 500\text{ mA}$ over ambient temperatures ranging from −55°C through 125°C. Note that the positive temperature coefficient of the wirewound resistor somewhat offsets the negative temperature coefficient of the SBP 9900A and thereby allows the tolerance of V_{TTL} to loosen from ±4% to ±5%.

7.3.4 RESISTOR POWER DISSIPATION AT ELEVATED TEMPERATURES. Since the capability of a resistor to dissipate power degrades at elevated temperatures, it is recommended that a ten watt resistor be used when configuring a SBP 9900A current source for application over ambient temperatures ranging from −55°C through +125°C.

7.4 TLO61M FEEDBACK REGULATED CURRENT SUPPLY

Figure 7-9 shows a TLO61M operational amplifier based, well-regulated, temperature compensating, current source for the SBP 9900A. The TL431M adjustable shunt regulator is used as a stable voltage reference of approximately 2.5 volts. This voltage is divided by R_1 and R_2 to supply about 1.5 volts (referenced to V_{CC}) to the non-inverting input of the TL061M operational amplifier. R_1, R_2 and the TL431M are biased with current set by R_3. The current through R_4 consists of both the collector-load, and base-drive currents of the Darlington-connected PNP transistor pair. This current is sensed as the voltage drop across R_4 and fed back to the inverting input of the TL061M. Therefore, if the TL061Ms inputs differ, the TL061Ms output will compensate for that difference by sinking as necessary, from the Darlington-pair more or less base-drive, until the TL061Ms inputs match. Due to the large gains involved, the capacitors are added to preclude oscillation. Table 7-1 summarizes data which was obtained at T_A = 25°C. While varying the simulated SBP 9900A node voltage between 0.851 and 1.666 volts, and allowing V_{CC} to fluctuate between 4.5 and 5.5 volts, an SBP 9900A supply current magnitude of 507 mA was regulated to within ±0.5 mA. Preliminary results of testing over the −55°C to 125°C temperature range indicate regulation of 507 mA ± 2%.



*R_1, R_2 are metal film resistors,
1/10 W, ± .1%, TC = ± 30 PPM/°C

T_1 – Low power silicon (2N2905, 2N3486, 2N4403, etc.)
T_2 – Medium power silicon (2N3792, 2N5005, 2N5153, etc.)

Figure 7-9. Feedback Regulated Power Supply

Table 7-1. $T_A = 25°C$

V_{CC} (V)	V_N (V)	I_{INJ} (mA)
4.50	1.665	507.0
5.00	1.665	507.2
5.50	1.666	507.4
4.50	1.088	507.2
5.00	1.088	507.3
5.50	1.089	507.5
4.50	0.851	507.0
5.00	0.851	507.2
5.50	0.851	507.4

7.5 LM117 FEEDBACK REGULATED CURRENT SUPPLY

Figure 7-10 shows an LM117 adjustable regulator based current source for the SBP 9900A. The LM117s V_{IN} input can be sourced by any dc voltage supply, regulated or ***unregulated***, ranging in value from a lower limit of 4.5 volts to an upper limit of 35 volts. The LM117 maintains equality between its 1.25 volt internal reference and the voltage differential between V_{OUT} and V_{ADJ}. As a result, the LM117 will adjust V_{OUT} such that the voltage differential across R is maintained at 1.25 volts in compensation for the SBP 9900As temperature/process variant node voltage.

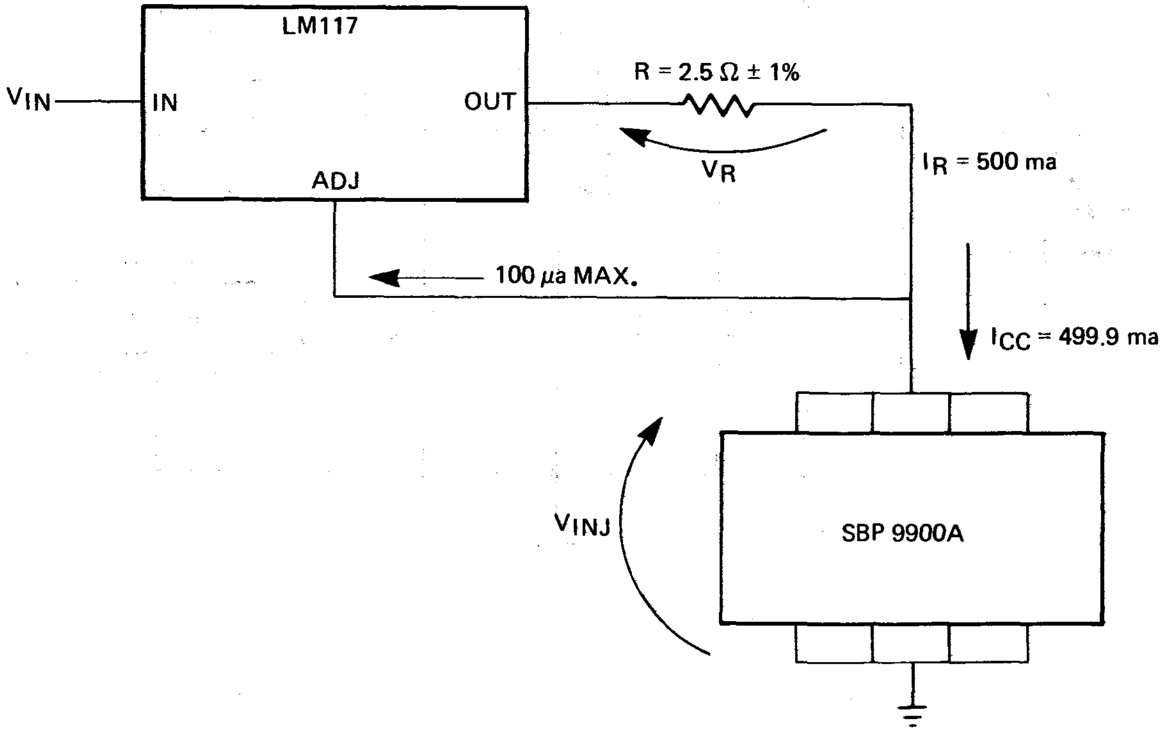


Figure 7-10. Regulated Current Source

The value of R required to supply 500 mA of current to the SBP 9900A is calculated as follows:

$$\begin{aligned} R &= \frac{V_R}{I_R} \\ &= \frac{1.25}{0.5} \\ R &= 2.5\,\Omega \pm 1\%\ (10\text{ W, TC} = \pm\ 100\text{ PPM/°C}) \end{aligned}$$

Since the V_{ADJ} input of the LM117 sinks a maximum of 100 μA of current:

$$\begin{aligned} I_{CC} &= I_R - 100\ \mu\text{a} \\ &= 500\text{ mA} - 0.1\text{ mA} \\ I_{CC} &= 499.9\text{ mA} \end{aligned}$$

The above SBP 9900A current supply was laboratory tested with temperature varied from −55°C to 125°C and V_{IN} varied from 4.5 to 6.0 volts. The observed value of I_{CC} was maintained at $I_{CC} = 499.9$ mA ± 4%.

Note that the power dissipated by the LM117 is approximately:

$$\begin{aligned} P &= (V_{IN} - V_{OUT})\, I_R \\ &= [V_{IN} - (V_R + V_{INJ})]\, I_R \\ P &= [V_{IN} - (\approx 1.25 + 1.175)]\,(0.5) \end{aligned}$$

If, for example, V_{IN} is sourced by an ***unregulated*** 5-volt (minimum) primary voltage supply, then:

$$\begin{aligned} P_{MIN} &= [5 - (\approx 1.25 + 1.175)]\,(0.5) \\ &= [5 - 2.425]\,(0.5) \\ P_{MIN} &= 1.29\text{ watts} \end{aligned}$$

As V_{IN} increased, the LM117s power dissipation is also increased. Consequently, it is recommended that the LM117 be mounted on a heat sink.

SECTION VIII

ELECTRICAL REQUIREMENTS

This section reviews the SBP 9900A electrical requirements including the system clock generation and interface signal characteristics. SBP 9900A specifications can be found in Appendix A.

8.1 SBP 9900A CLOCK GENERATION

The SBP 9900A requires a single phase, 67% duty cycle, TTL compatible, edge-triggering clock. Because the SBP 9900A is a static device, it is not necessary to impose minimum clock frequency requirements. Consequently, the clock frequency can vary from dc through 3 MHz. This allows interface designs which inhibit the clock to the SBP 9900A to perform such functions as clock-stopped DMA, and single step microinstruction execution for system debugging. Clock timing is shown in Figure 8-1.

8.1.1 SBP 9900A CLOCK GENERATION. Figure 8-2 shows a SBP 9900A clock generator using two edge-triggered J-K flip-flops that sequence through three states to produce the recommended 67% duty cycle clock. The crystal controlled oscillator must operate at 3X the desired SBP 9900A clock frequency. The additional CKA and CKA outpus can be used to synchronize external logic to the SBP 9900A.

An alternate method of generating the 3X oscillator frequency is shown in Figure 8-3. This method incorporates the SN54LS320 voltage controlled oscillator.

8.2 SBP 9900A SIGNAL INTERFACING

The SBP 9900A input/output characteristics were selected for compatibility with most 5-volt logic/memory families. The input/output schematics of the SBP 9900A are shown in Figure 8-4. The input circuit used on the SBP 9900A is basically an RTL configuration which has been modified for TTL compatibility as shown in Figure 8-4A. An input-clamping diode is incorporated to limit negative excursions (ringing) when the SBP 9900A is on the receiving end of a transmission line; an input switching threshold of nominally +1.5 volts has been specified for improved noise immunity. This threshold is achieved via two resistors which function as a voltage divider to increase the one V_{BE} threshold of the I^2L input transistor to +1.5 volts.

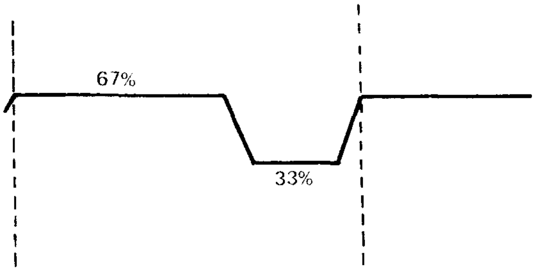


Figure 8-1. SBP 9900A Clock Timing

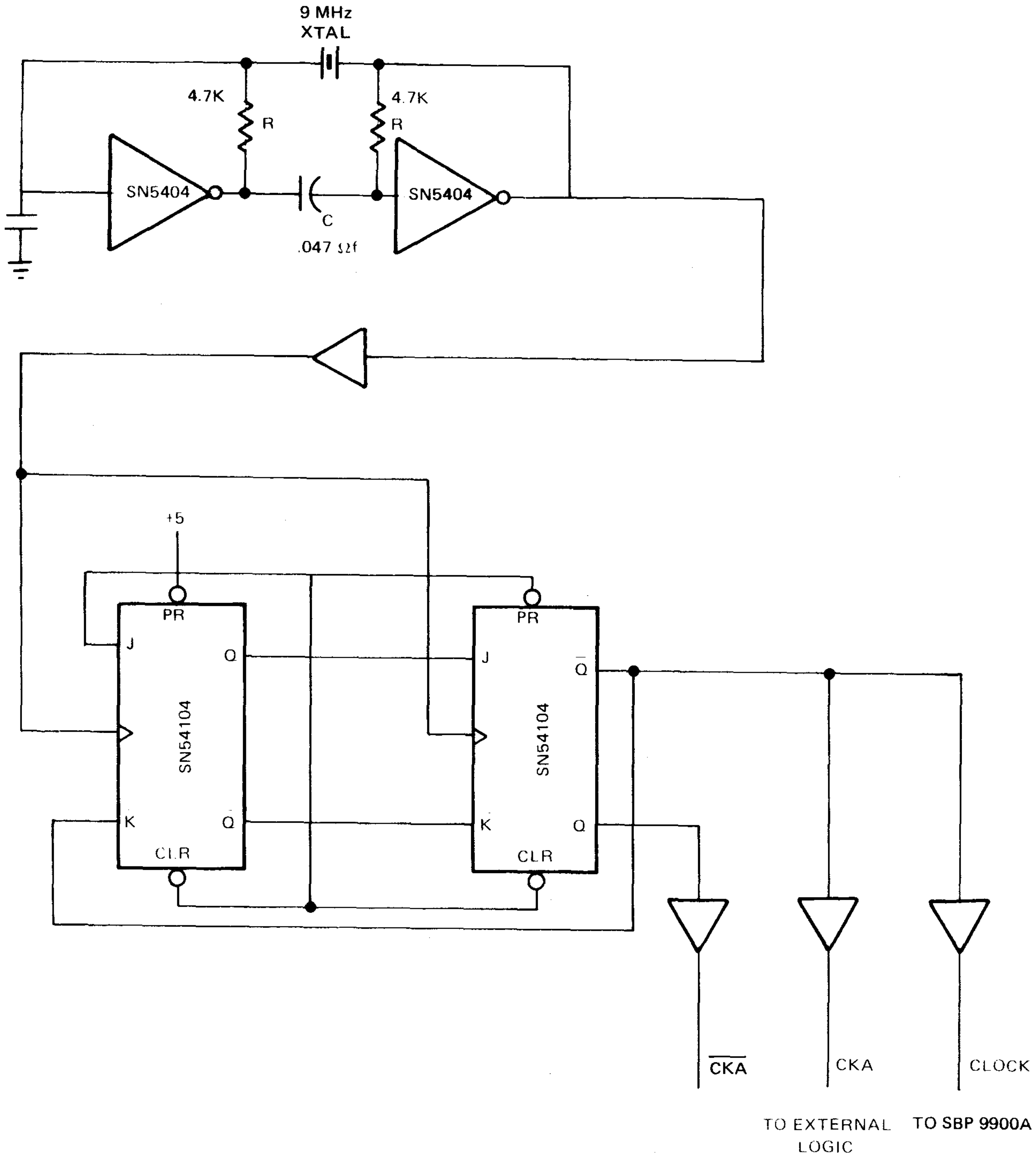


Figure 8-2. SBP 9900A Clock Generator

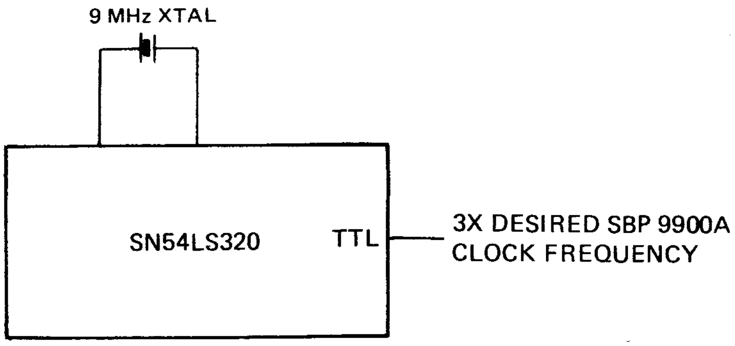
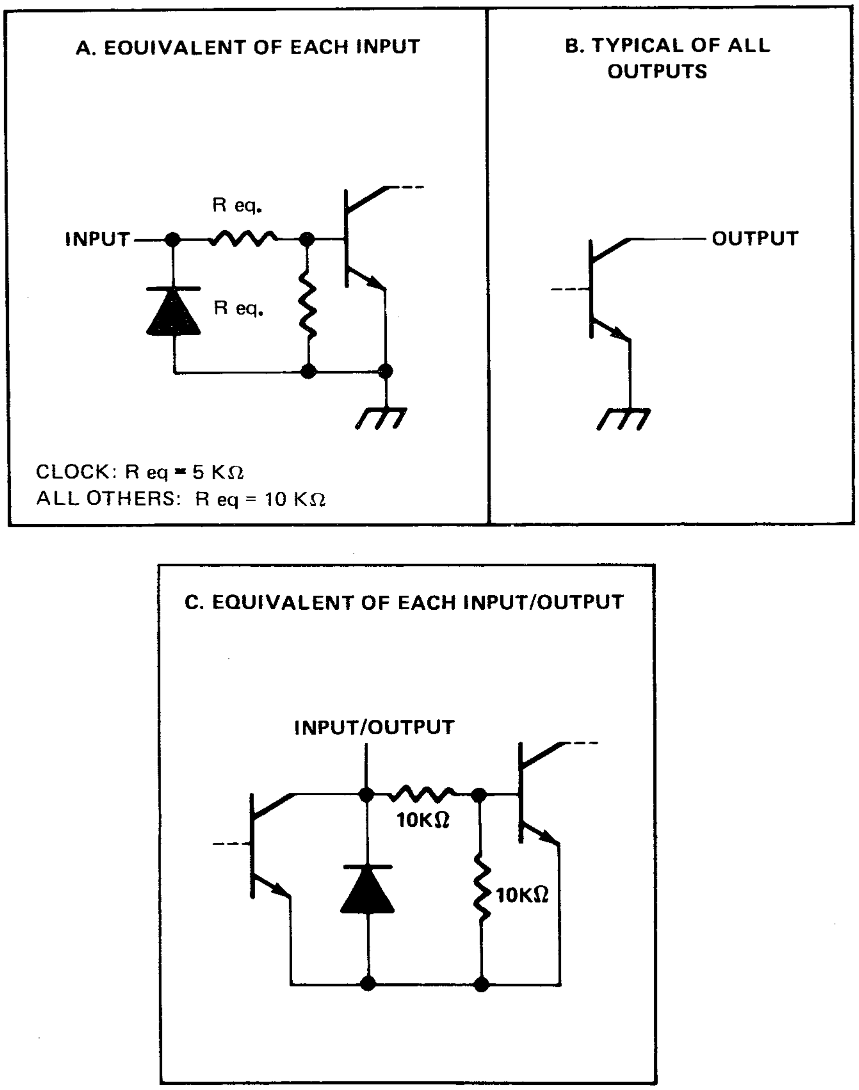


Figure 8-3. Voltage Controlled Oscillator


A. EQUIVALENT OF EACH INPUT

CLOCK: R eq = 5 KΩ
ALL OTHERS: R eq = 10 KΩ

B. TYPICAL OF ALL OUTPUTS

C. EQUIVALENT OF EACH INPUT/OUTPUT

Figure 8-4. Schematics of Equivalent Inputs, Outputs, Inputs/Outputs

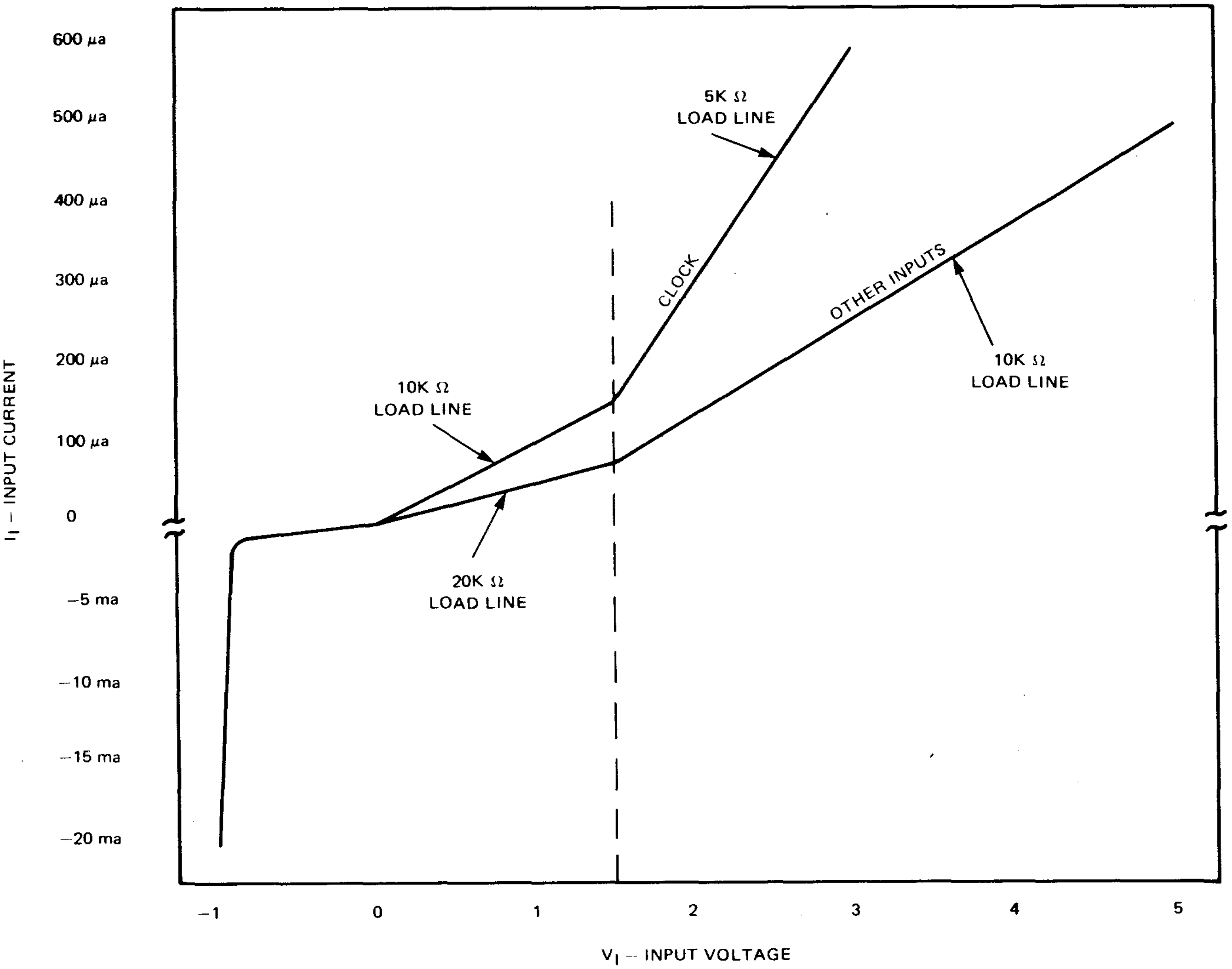


Figure 8-5. Typical Input Characteristics

The input circuit characteristics for input current versus input voltage are shown in Figure 8-5. The 10K and 20K ohm load lines and threshold knee at +1.5 volts provide a high-impedance characteristic to reduce input loading and improve the low-logic level input noise immunity over some standard TTL inputs. Full compatibility is maintained with virtually all 5 volt logic/memory families.

The output circuit selected for the SBP 9900A is an injected open-collector transistor shown in Figure 8-4B. The output circuit characteristic for logic level low output voltage (V_{OL}) versus logic-level low output current (I_{OL}) is shown in Figure 8-6. At rated injector current, the SBP 9900A output circuit offers a low-level output voltage of typically 220 mV with I_{OL} = 20 ma.

The output circuit characteristics for 1) logic-level high output voltage (V_{OH}) and current (I_{OH}), 2) rise times, and 3) next stage input noise immunity, are a function of the load circuit being sourced. The load circuit can be either:

a. the direct input, if no source current is required, of a five-volt logic family function, or for greater noise immunity and improved rise times,

b. the direct input of a five-volt logic family function in conjunction with a discrete pull-up resistor.

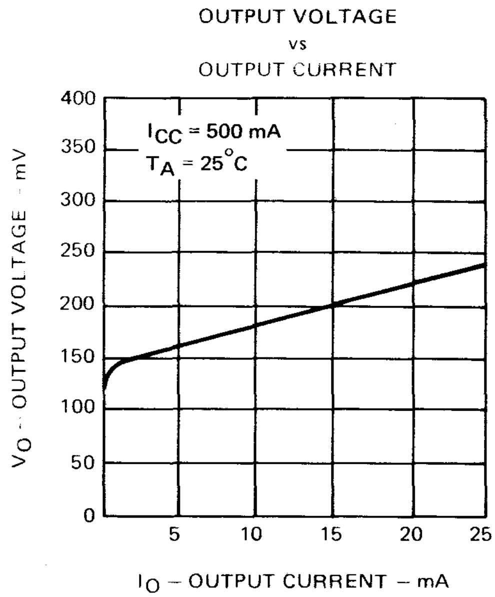


Figure 8-6. Typical Output Characteristics

8.2.1 INPUT INTERFACING. The SBP 9900A inputs can be driven directly from most 5-volt logic families. Five-volt functions which have internal pull-ups require no external components normally, and open-collector functions can be employed with selected external pull-up resistors. Calculations for determining the resistive value of the pull-up are as shown in Table 8-1.

Table 8-1. Interfacing Driven Inputs

	CLOCK	ALL OTHER INPUTS
Driven by open-collector TTL, CMOS	$R_P = \frac{V_{CC} - 3.3}{0.0005\ (N)}$	$R_P = \frac{V_{CC} - 3.3}{0.00025\ (N)}$
Driven by MOS, CMOS: Low-Threshold	$R_B = \frac{V_{CC} - 2.4}{I_{OH} - 0.0005\ (N)}$	$R_B = \frac{V_{CC} - 2.4}{I_{OH} - 0.00025\ (N)}$
High-Threshold	Not Recommended	
Driven by TTL with 5-V active pull-up	Drive Directly	

Just as with most 5-volt devices, inputs of the SBP 9900A which are hardwired to a high-logic level low-impedance source such as V_{CC} should incorporate a current-limiting resistor. To provide transient protection, any number of inputs can utilize a common pull-up based on the formula shown in Table 8-2.

Table 8-2. Terminating Unused Inputs

	ALL INPUTS
Hardwire to V_{IH}	$RP = \frac{V_{CC} - 3.3}{0.0025\ (N)}$
Hardwire to V_{IL}	Strap Directly to GND (0 V)

Inputs requiring a hardwired low-logic level can be connected directly to ground.

8.2.2 OUTPUT INTERFACING. The open-collector I^2L outputs, when supplied with a proper load resistor R_L, can be used to drive virtually any 5-volt logic/memory inputs directly. Total fan-out from the SBP 9900As output is based on calculations which determine, as shown in Figure 8-7, the minimum and maximum values of R_L which are possible. A minimum resistor value must be determined which will ensure that current through this resistor and sink current from the loads will not exceed the SBP 9900A's low-level output current rating. A maximum resistor value must be determined which will ensure that sufficient current is available to satisfy both the driven loads and the high-level output current requirements yet the voltage drop of which is insufficient to pull the high-level output voltage below 2.4 volts.

8.2.3 OUTPUT LOAD RESISTOR GUIDE. Table 8-3 provides both minimum and maximum R_L values for one, five, and nine or ten loads of the most popular 5 V logic families. The calculations are based on the following guidelines:

V_{source} = 5 V
V_{OH} = 2.4 V (satisfies virtually all 5 V logic)
V_{OL} = 0.4 V (based on max noise margin provided by SBP 9900A)
I_{OH} = 400 μA
I_{OL} as specified (20 mA, 10 mA)

And unit loads of:	I_{IL} =	I_{IH} =
54LS/74LS	0.36 mA	10 μA
54/74	1.6 mA	40 μA
54S/74S	2 mA	50 μA
N-MOS	10 μA	10 μA
C-MOS	10 pA	10 pA

HIGH-LEVEL (OFF-STATE) CIRCUIT CALCULATIONS

The allowable voltage drop across the load resistor (V_{RL}) is the difference between the pull-up source and the V_{OH} level required at the load:

$$V_{RL} = V_{source} - V_{OH}\text{ Min}$$

The total current through the load resistor (I_{RL}) is the sum of the load current and the high-level output current (I_{OH}):

$$I_{RL} = \text{Load Current (into the load inputs)} + I_{OH}$$

where: $I_{OH} = 400\ \mu A$ Max.

Therefore, calculations for the maximum value of R_L would be:

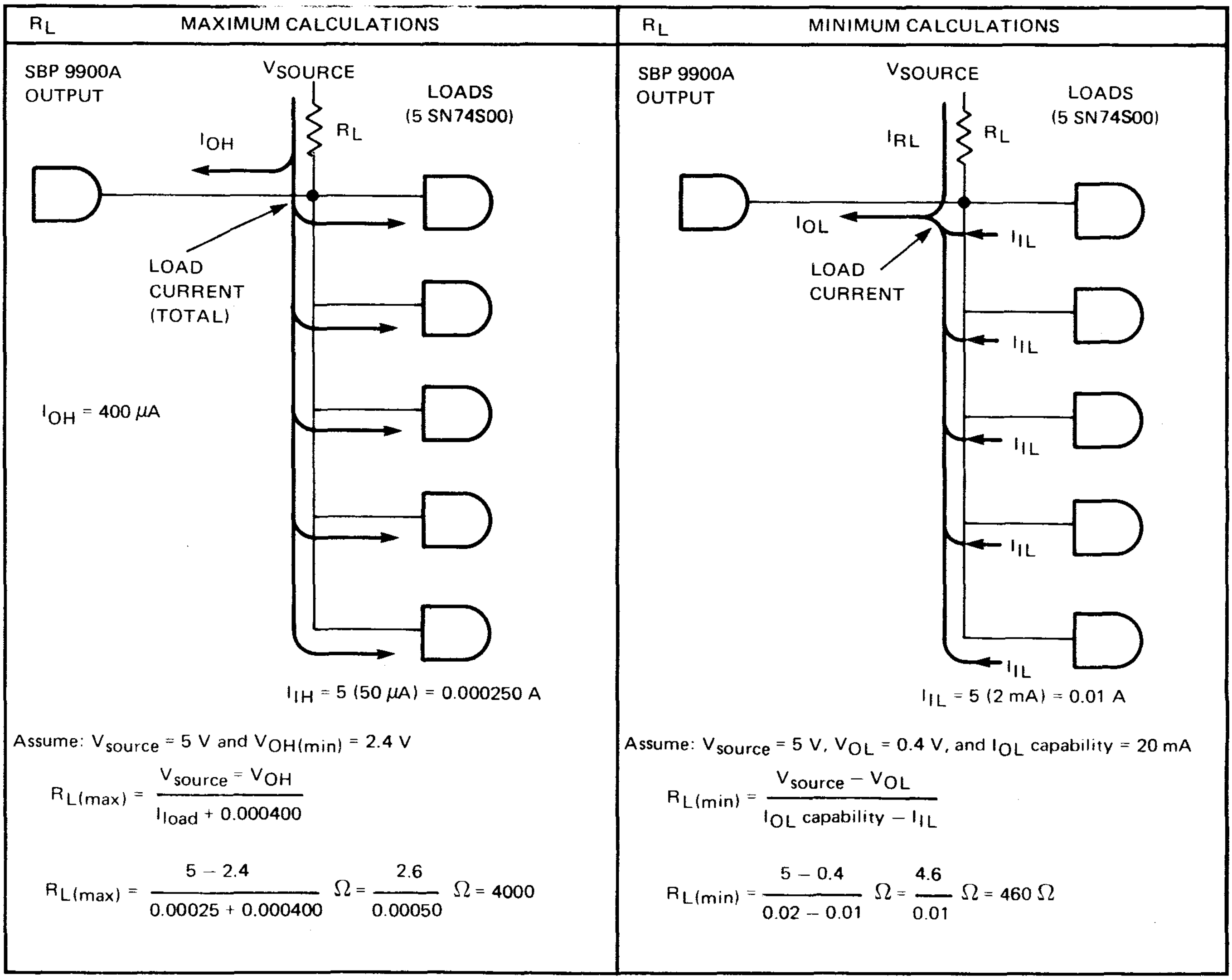
$$R_{L(max)}\text{ in ohms} = \frac{V_{source} - V_{OH}\text{ Min}}{\text{Amperes of Load Current} + 0.000400}$$

LOW-LEVEL (ON-STATE) CIRCUIT CALCULATIONS

The maximum current through the load resistor when the output is on, plus the amount of current from the low-level input load, must be limited to the I_{OL} capability of the output.

Therefore, the equation is:

$$R_{L(min)} = \frac{V_{source} - V_{OL}\text{ Max}}{I_{OL}\text{ capability} - I_{load}}$$



Assume: V_{source} = 5 V and $V_{OH(min)}$ = 2.4 V

$$R_{L(max)} = \frac{V_{source} = V_{OH}}{I_{load} + 0.000400}$$

$$R_{L(max)} = \frac{5 - 2.4}{0.00025 + 0.000400}\ \Omega = \frac{2.6}{0.00050}\ \Omega = 4000$$

Assume: V_{source} = 5 V, V_{OL} = 0.4 V, and I_{OL} capability = 20 mA

$$R_{L(min)} = \frac{V_{source} - V_{OL}}{I_{OL}\text{ capability} - I_{IL}}$$

$$R_{L(min)} = \frac{5 - 0.4}{0.02 - 0.01}\ \Omega = \frac{4.6}{0.01}\ \Omega = 460\ \Omega$$

Figure 8-7. Output Load Resistor Calculations

Table 8-3. Output Load Resistor Values

SBP 9900 OUTPUT TYPE	DRIVING 1 LOAD		DRIVING 5 LOADS		DRIVING 10 LOADS		TYPE OF LOGIC
	R_L (MIN)	R_L (MAX)	R_L (MIN)	R_L (MAX)	R_L (MIN)	R_L (MAX)	
20 mA SINK OUTPUTS	234 Ω	6341 Ω	252 Ω	5778 Ω	280 Ω	5200 Ω	54LS/74LS
	250 Ω	5909 Ω	383 Ω	4333 Ω	1150 Ω	3250 Ω	54/74
	256 Ω	5777 Ω	460 Ω	4000 Ω	2300 Ω	2889 Ω	54S/74S
	230 Ω	6341 Ω	230 Ω	5778 Ω	231 Ω	5200 Ω	MOS
	230 Ω	6500 Ω	230 Ω	6500 Ω	231 Ω	6498 Ω	C-MOS

Specific designs can be tailored for minimum power or maximum performance by making the individual calculations as described previously.

APPENDIX A
SBP 9900A
SPECIFICATIONS

A.1 INTRODUCTION

The SBP 9900A parallel 16-bit CPU microprocessor incorporates I^2L technological improvements to enhance system throughput rates by 50% beyond that of the SBP 9900. Designed specifically to retain functional and electrical interchangeability with its predecessor, the ruggedized SBP 9900A also features from 29% to 48% reduction in propagation delay times to further simplify system implementation.

This static bipolar microprocesor operates from a single-phase clock, over frequencies from DC to 3 megahertz with TTL compatible I/O, permitting direct use with standard logic and memory devices to reduce or eliminate the need for special clock and interface circuits. The architecture, see Figure A-1, is based on an advanced memory-to-memory concept which places register files in main memory, meaning that the number of general-purpose registers is limited only by the size of the program memory, and that context switching or interrupts can be handled efficiently. Software compatibility with other 9900 microprocessor family members provides a common body of hardware/software within Texas Instruments 990 minicomputer family.

SBP 9900A Ordering Information

TYPE NUMBER	CLOCK FREQUENCY	INJECTOR CURRENT	OPERATING TEMPERATURE
SBP 9900 AMJ	DC to 3 MHz	450 to 550 mA	−55°C to 125°C
SBP 9900 ANJ†	DC to 3 MHz	450 to 550 mA	−55°C to 125°C
SBP 9900 AEJ	DC to 3 MHz	450 to 550 mA	−40°C to 85°C
SBP 9900ACJ	DC to 3 MHz	500 to 620 mA	0°C to 70°C

†Includes additional high reliability assurance processing with variables data and lot traceability.

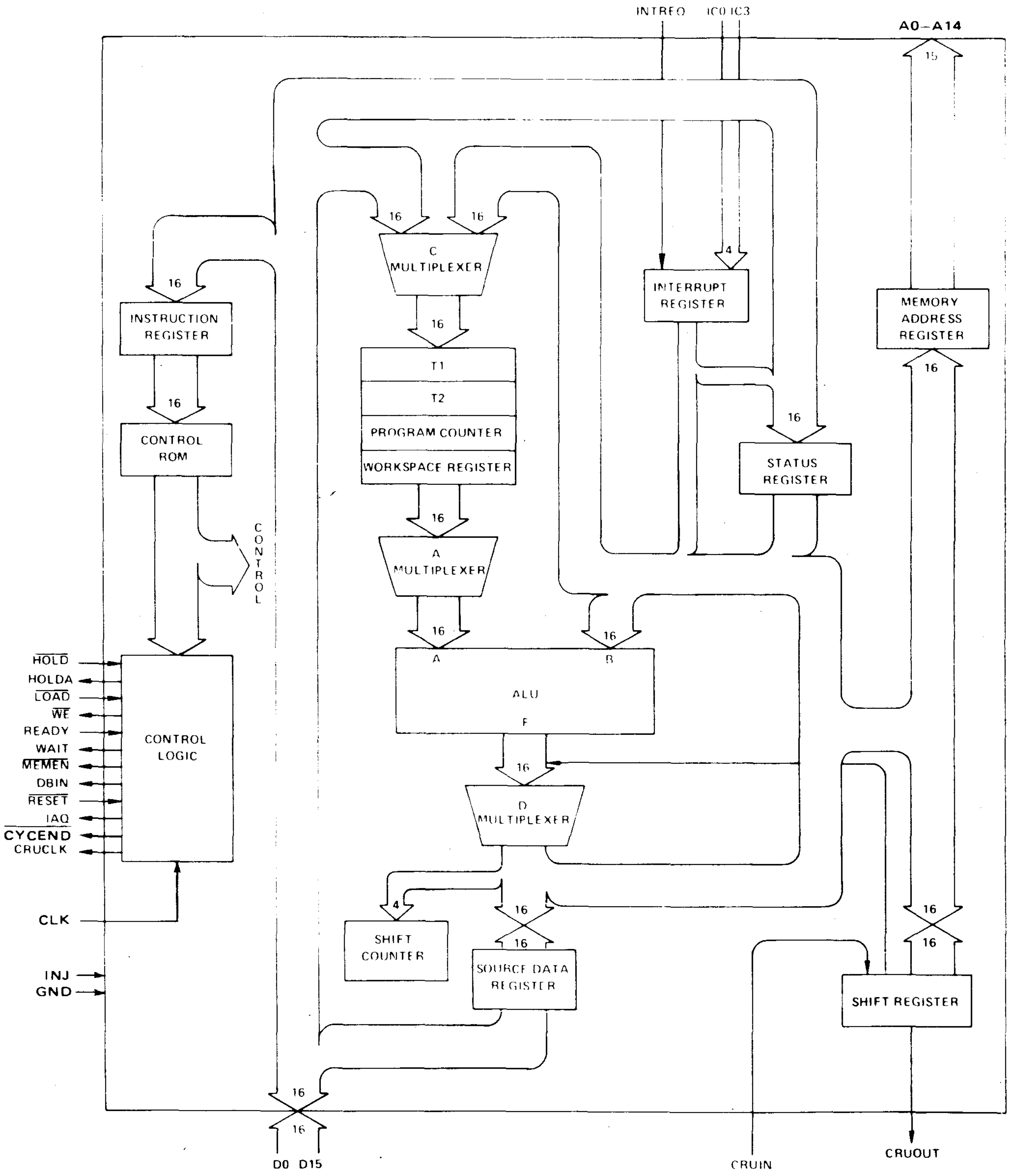
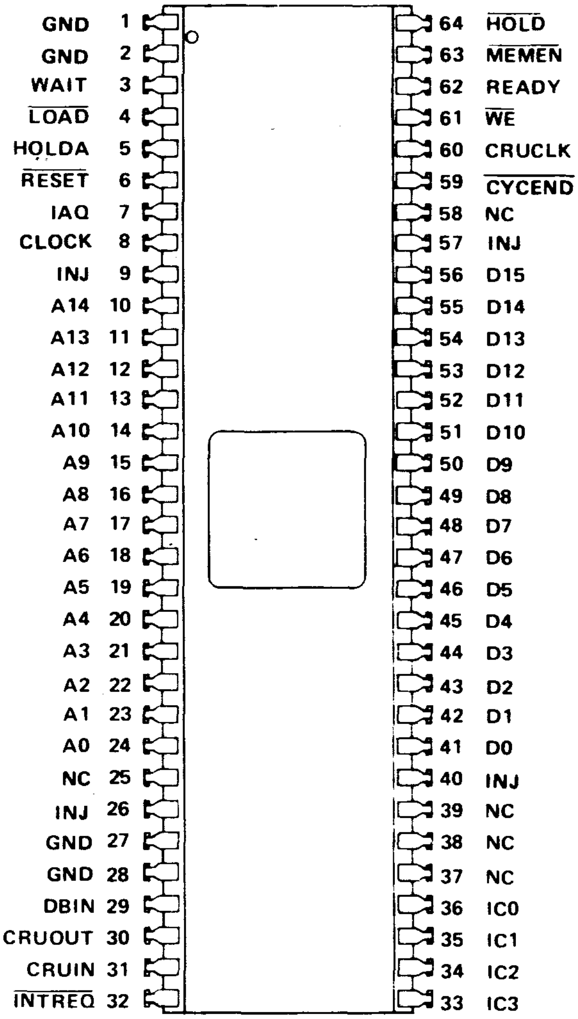


Figure A-1. SBP 9900A Architecture

A.2 KEY FEATURES

- Parallel 16-Bit Word Length
- Full Minicomputer Instruction Set Includes Multiply and Divide
- Directly Addresses Up to 65,536 Bytes/32,768 Words of Memory
- Advanced Memory-To-Memory Architecture
- Multiple 16-Word Register Files (Work Spaces) Reside in Memory
- Separate I/O, Memory and Interrupt Bus Structures
- 16 Prioritized Hardware Interrupts
- 16 Software Interrupts (XOPs)
- Programmed, and DMA, I/O Capability
- 16-Bit Parallel I/O Port; or Serial I/O Via Communications-Register-Unit (CRU)
- 64-Pin Package
- Software Compatible with TI 9900 Microprocessor/990 Minicomputer Family
- I2L Technology:
  - — 3 MHz Clock Rate at 588 mW Across Operating Temperature Range
  - — Single d-c Power Supply
  - — Fully Static Operation
  - — Single Phase Clock
  - — Directly TTL Compatible I/O (Including Clock)

PIN ASSIGNMENTS



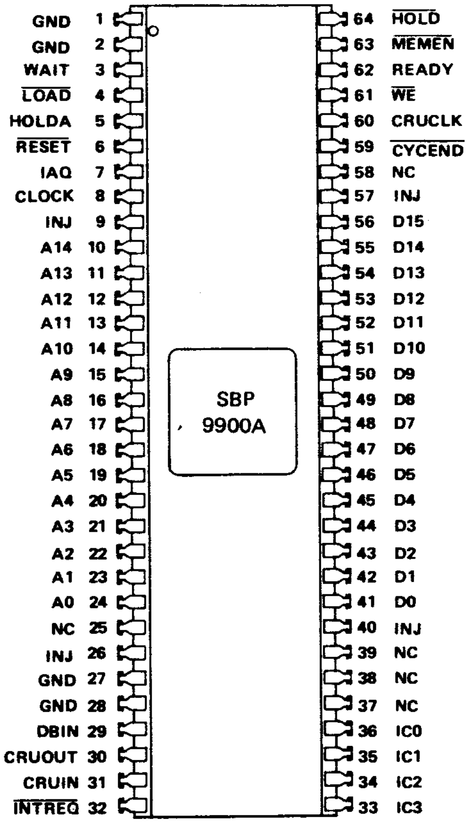
NC–No internal connection

A.3 SBP 9900A PIN DESCRIPTION

Table A-3 describes the function of each SBP 9900A pin, and Figure A-11 shows their assigned locations.

TABLE A-3
SBP 9900A PIN ASSIGNMENTS AND FUNCTIONS

SIGNATURE	PIN	I/O	DESCRIPTION
			ADDRESS BUS
A0 (MSB)	24	OUT	A0 (MSB) through A14 (LSB) comprise the address bus. This open-collector bus provides the memory-address vector to the external-memory system when $\overline{\text{MEMEN}}$ is active, and I/O-bit addresses to the I/O system when $\overline{\text{MEMEN}}$ is inactive. When HOLDA is active, the address bus is pulled to the logic level HIGH state by the individual pull-up resistors
A14 (LSB)	10	OUT	tied to each respective open-collector output.
			DATA BUS
D0 (MSB)	41	I/O	D0 (MSB) through D15 (LSB) comprise the bidirectional open-collector data bus. This bus transfers memory data to (when writing) and from (when reading) the external memory system when $\overline{\text{MEMEN}}$ is active. When HOLDA is active, the data bus is pulled to the logic level HIGH state by
D15 (LSB)	56	I/O	the individual pull-up resistors tied to each respective open-collector output.
			POWER SUPPLY
INJ	9		Injector-Supply-Current
INJ	26		Injector-Supply-Current
INJ	40		Injector-Supply-Current
INJ	57		Injector-Supply-Current
GND	1		Ground Reference
GND	2		Ground Reference
GND	27		Ground Reference
GND	28		Ground Reference
			CLOCK
CLOCK	8	IN	CLOCK
			BUS CONTROL
DBIN	29	OUT	DATA BUS IN. When active (pulled to logic level HIGH), DBIN indicates that the SBP 9900A has disabled its output buffers to allow the memory to place memory-read data on the data bus during $\overline{\text{MEMEN}}$. DBIN remains at logic level LOW in all other cases except when HOLDA is active (pulled to logic level HIGH).
$\overline{\text{MEMEN}}$	63	OUT	MEMORY ENABLE. When active (logic level LOW), $\overline{\text{MEMEN}}$ indicates that the address bus contains a memory address.
$\overline{\text{WE}}$	61	OUT	WRITE ENABLE. When active (logic level LOW), $\overline{\text{WE}}$ indicates that the SBP 9900A data bus is outputting data to be written into memory.
CRUCLK	60	OUT	COMMUNICATIONS-REGISTER-UNIT (CRU) CLOCK. When active (pulled to logic level HIGH), CRUCLK indicates to the external interface logic the presence of output data on CRUOUT, or the presence of an encoded external instruction on A0 through A2.
CRUIN	31	IN	CRU DATA IN. CRUIN, normally driven by 3-state or open-collector devices, receives input data from the external interface logic. When the SBP 9900A executes a STCR or TB instruction, it samples CRUIN for the level of the CRU input bit specified by the address bus (A3 through A14).



NC—No internal connection

TABLE A-3
SBP 9900A PIN ASSIGNMENTS AND FUNCTIONS (Continued)

SIGNATURE	PIN	I/O	DESCRIPTION
CRUOUT	30	OUT	CRU DATA OUT. CRUOUT outputs serial data when the SBP 9900A executes a LDCR, SBZ, SBO instruction. The data on CRUOUT should be sampled by the external interface logic when CRUCLK goes active (pulled to logic level HIGH).
			INTERRUPT CONTROL
$\overline{\text{INTREQ}}$	32	IN	INTERRUPT REQUEST. When active (logic level LOW), $\overline{\text{INTREQ}}$ indicates that an external interrupt is requesting service. If $\overline{\text{INTREQ}}$ is active, the SBP 9900A loads the data on the interrupt-code input-lines IC0 through IC3 into the internal interrupt-code storage register. The code is then compared to the interrupt mask bits of the status register. If equal or higher priority than the enabled interrupt level (interrupt code equal or less than status register bits 12 through 15), the SBP 9900A initiates the interrupt sequence. If the comparison fails, the SBP 9900A ignores the interrupt request. In that case, $\overline{\text{INTREQ}}$ should be held active. The SBP 9900A will continue to sample IC0 through IC3 until the program enables a sufficiently low interrupt-level to accept the requesting interrupt.
IC0 (MSB) IC3 (LSB)	36 33	IN IN	INTERRUPT CODES. IC0 (MSB) through IC3 (LSB), receiving an interrupt identity code, are sampled by the SBP 9900A when $\overline{\text{INTREQ}}$ is active (logic level LOW). When IC0 through IC3 are LLLH, the highest priority ***external*** interrupt is requesting service; when HHHH, the lowest priority external interrupt is requesting service.
			MEMORY CONTROL
HOLD	64	IN	When active (logic level LOW), $\overline{\text{HOLD}}$ indicates to the SBP 9900A that an external controller (e.g., DMA device) desires to use both the address bus and data bus to transfer data to or from memory. In response, the SBP 9900A enters the hold state after completion of its present memory cycle. The SBP 9900A then allows its address bus, data bus, $\overline{\text{WE}}$, $\overline{\text{MEMEN}}$, DBIN, and HOLDA facilities to be pulled to the logic level HIGH state. When HOLD is deactivated, the SBP 9900A returns to normal operation from the point at which it was stopped.
HOLDA	5	OUT	HOLD ACKNOWLEDGE. When active (pulled to logic level HIGH), HOLDA indicates that the SBP 9900A is in the hold state and that its address bus, WE, MEMEN, and DBIN facilities are pulled to the logic level HIGH state.
READY	62	IN	When active (logic level HIGH), READY indicates that the memory will be ready to read or write during the next clock cycle. When not-ready is indicated a memory operation, the SBP 9900A enters a wait state and suspends internal operation until the memory systems activate READY.
WAIT	3	OUT	When active (pulled to logic level HIGH), WAIT indicates that the SBP 9900A has entered a wait state in response to a not-ready condition from memory.
			TIMING AND CONTROL
IAQ	7	OUT	INSTRUCTION ACQUISITION. IAQ is active (pulled to logic level HIGH) during any SBP 9900A initiated instruction acquisition memory cycle. Consequently, IAQ may be used to facilitate detection of illegal op codes.
$\overline{\text{CYCEND}}$	59	OUT	CYCLE END. When active (logic level LOW), $\overline{\text{CYCEND}}$ indicates the SBP 9900A will initiate a new machine cycle on the low-to-high transition of the next CLOCK.
$\overline{\text{LOAD}}$	4	IN	When active (logic level LOW), $\overline{\text{LOAD}}$ causes the SBP 9900A to execute a nonmaskable interrupt with memory addresses FFFC_{16} and FFFE_{16} containing the associated trap vectors (WP and PC). The load sequence is initiated after the instruction being executed is completed. $\overline{\text{LOAD}}$ will also terminate an idle state. If $\overline{\text{LOAD}}$ is active during the time $\overline{\text{RESET}}$ is active, the $\overline{\text{LOAD}}$ trap will occur after the $\overline{\text{RESET}}$ function is completed. $\overline{\text{LOAD}}$ should remain active for one instruction execution period (IAQ may be used to monitor instruction boundaries). $\overline{\text{LOAD}}$ may be used to implement cold-start ROM loaders. Additionally, front-panel routines may be implemented using CRU bits as front panel-interface signals, and software-control routines to direct the panel operations.
$\overline{\text{RESET}}$	6	IN	When active (logic level LOW), $\overline{\text{RESET}}$ causes the SBP 9900A to reset itself and inhibit $\overline{\text{WE}}$ and CRUCLK. When $\overline{\text{RESET}}$ is released, the SBP 9900A initiates a level-zero interrupt sequence acquiring the WP and PC trap vectors from memory locations 0000_{16} and 0002_{16}, sets all status register bits to logic level LOW, and then fetches the first instruction of the reset program environment. $\overline{\text{RESET}}$ must be held active for a minimum of three CLOCK cycles.

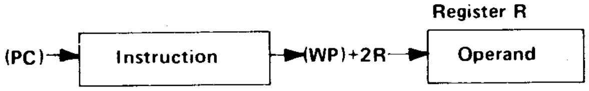
A.4 9900 INSTRUCTION SET

A.4.1 DEFINITION. Each 9900 instruction performs one of the following operations:

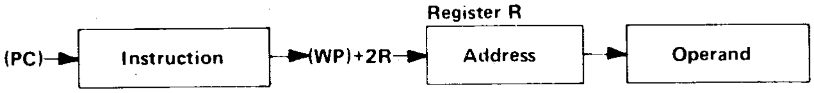
- Arithmetic, logical, comparison, or manipulation operations on data
- Loading or storage of internal registers (program counter, workspace pointer, or status)
- Data transfer between memory and external devices via the CRU
- Control functions.

A.4.2 ADDRESSING MODES The 9900 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.). The following figures graphically describe the derivation of the effective address for each addressing mode. The applicability of addressing modes to particular instructions is described in Section B, along with the description of the operations performed by the instruction. The symbols following the names of the addressing modes [R, *R, *R+,@LABEL, or TABLE (R)] are the general forms used by 9900 assemblers to select the addressing mode for register R.

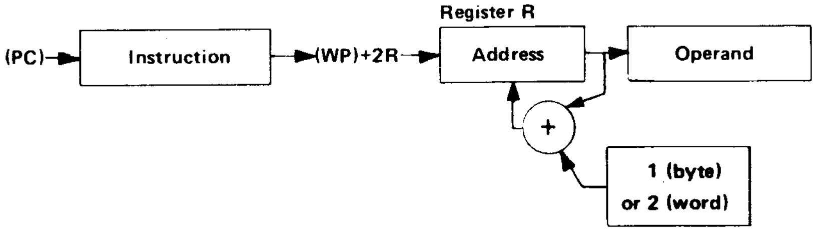
A.4.2.1 Workspace Register Addressing **R.** Workspace Register R contains the operand.



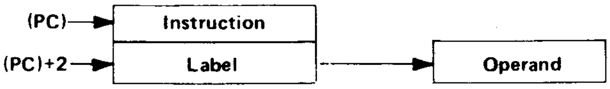
A.4.2.2 Workspace Register Indirect Addressing ***R.** Workspace Register R contains the address of the operand.



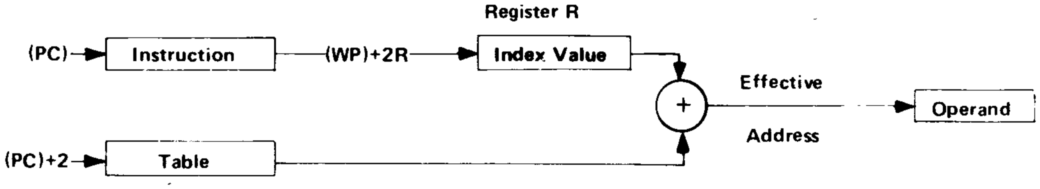
A.4.2.3 Workspace Register Indirect Auto Increment Addressing ***R+.** Workspace Register R contains the address of the operand. After acquiring the operand, the contents of workspace register R are incremented.



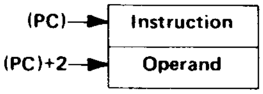
A.4.2.4 Symbolic (Direct) Addressing @ **LABEL.** The word following the instruction contains the address of the operand.



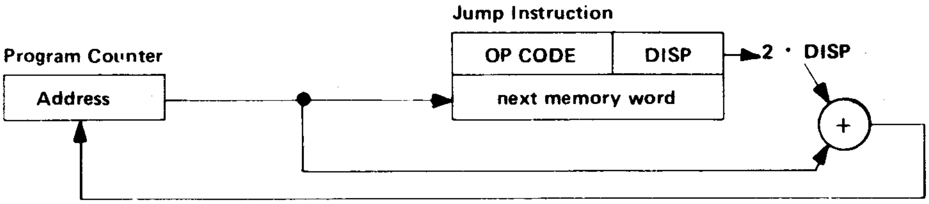
A.4.2.5 Indexed Addressing @ **Table (R)R.** The word following the instruction contains the base address. Workspace register R contains the index value. The sum of the base address and the index value results in the effective address of the operand.



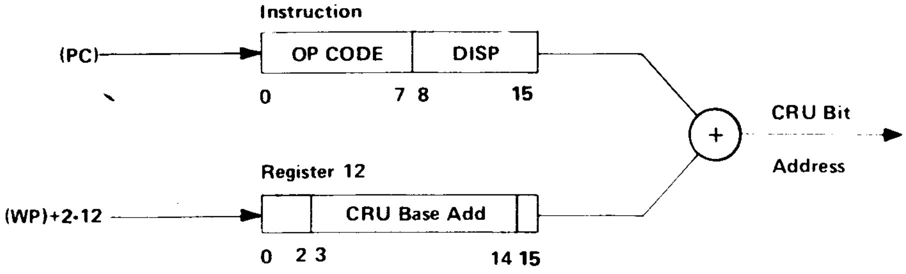
A.4.2.6 Immediate Addressing. The word following the instruction contains the operand.



A.4.2.7 Program Counter Relative Addressing. The 8-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.



A.4.2.8 CRU Relative Addressing. The 8-bit signed displacement in the right byte of the instruction is added to the CRU base address (bits 3 through 14 of workspace register 12). The result is the CRU address of the selected CRU bit.



A.5 TERMS AND DEFINITIONS

The terms used in describing the instructions of the 9900 are defined in Table A-4.

TABLE 4

TERM DEFINITIONS

TERM	DEFINITION
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
T_D	Destination address modifier
T_S	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
+	Arithmetic addition
−	Arithmetic subtraction
AND	Logical AND
OR	Logical OR
+	Logical exclusive OR
$\overline{n}$	Logical complement of n

A.6 STATUS REGISTER.

The status register contains the interrupt mask level and information pertaining to the instruction operation. Table A-5 explains the bit indications.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ST0	ST1	ST2	ST3	ST4	ST5	ST6	not used (=0)					ST12	ST13	ST14	ST15
L	A	=	C	O	P	X						Interrupt Mask			

TABLE A-5
STATUS REGISTER BIT DEFINITIONS

BIT	NAME	INSTRUCTION	CONDITION TO SET BIT TO 1
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	If (SA) = 0
		All others	If result = 0
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	If MSB(SA) = 0 and (SA) = 0
		All others	If MSB of result = 0 and result = 0
ST2	EQUAL	C,CB	If (SA) = (DA)
		CI	If (W) = IOP
		COC	If (SA) and (DA) = 0
		CZC	If (SA) and (DA) = 0
		TB	If CRUIN = 1
		ABS	If (SA) = 0
		All others	If result = 0
ST3	CARRY	A, AB, ABS, AI, DEC, DECT, INC, INCT, NEG, S, SB	If CARRY OUT = 1
		SLA, SRA, SRC, SRL	If last bit shifted out = 1
ST4	OVERFLOW	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 = 1
		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
		ABS, NEG	If (SA) = 8000_{16}
ST5	PARITY	CB, MOVB	If (SA) has odd number of 1's
		LDCR, STCR	If $1 \leq C \leq 8$ and (SA) has odd number of 1's
		AB, SB, SOCB, SZCB	If result has odd number of 1's
ST6	XOP	XOP	If XOP instruction is executed
ST12-ST15	INTERRUPT MASK	LIMI	If corresponding bit of IOP is 1
		RTWP	If corresponding bit of WR15 is 1

A.7 INSTRUCTIONS

A.7.1 DUAL OPERAND INSTRUCTIONS WITH MULTIPLE ADDRESSING MODES FOR SOURCE AND DESTINATION OPERAND.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
General format:	OP CODE			B	T_D		D				T_S		S			

If B = 1 the operands are bytes and the operand addresses are byte addresses. If B = 0 the operands are words and the operand addresses are word addresses.

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 $T_S = T_D$ = 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 (SA) → (DA)
SZCB	0 1 0	1	Set zeroes corresponding bytes	Yes	0-2,5	(DA) AND (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)

A.7.2 DUAL OPERAND INSTRUCTIONS WITH MULTIPLE ADDRESSING MODES FOR THE SOURCE OPERAND AND WORKSPACE REGISTER ADDRESSING FOR THE DESTINATION

<table>
<tr><td></td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td>General format:</td><td colspan="6">OP CODE</td><td colspan="4">D</td><td colspan="2">T_S</td><td colspan="4">S</td></tr>
</table>

The addressing mode for the source operand is determined by the T_S 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 zeroes 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	4	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.

A.7.3 EXTENDED OPERATION (XOP) INSTRUCTION.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
General format:	0	0	1	0	1	1	D				T_S		S			

The T_S and S fields provide multiple mode addressing capability for the source operand. When the XOP is executed ST6 is set and the following transfers occur:

$(40_{16} + 4D) \rightarrow (WP)$
$(42_{16} + 4D) \rightarrow (PC)$
$SA \rightarrow (\text{new } WR11)$
$(\text{old } WP) \rightarrow (\text{new } WR13)$
$(\text{old } PC) \rightarrow (\text{new } WR14)$
$(\text{old } ST) \rightarrow (\text{new } WR15)$

The 9900 does not test interrupt requests $\overline{\text{INTREQ}}$ upon completion of the XOP instruction.

A.7.4 SINGLE OPERAND INSTRUCTIONS

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
General format:	OP CODE										T_S		S			

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

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9	MEANING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
B	0 0 0 0 0 1 0 0 0 1	Branch	No	-	SA → (PC)
BL	0 0 0 0 0 1 1 0 1 0	Branch and link	No	-	(PC) → (WR 11); SA → (PC)

BLWP	0 0 0 0 0 1 0 0 0 0	Branch and load workspace pointer	No	-	(SA) → (WP); (SA+2) → (PC);
					(old WP) → (new WR 13);
					(old PC) → (new WR 14);
					(old ST) → (new WR 15);
					the interrupt input (INTREQ) is not BLWP instruction.

CLR	0 0 0 0 0 1 0 0 1 1	Clear operand	No	-	0 → (SA)
SETO	0 0 0 0 0 1 1 1 0 0	Set to ones	No	-	$FFFF_{16}$ → (SA)
INV	0 0 0 0 0 1 0 1 0 1	Invert	Yes	0-2	(SA) → (SA)
NEG	0 0 0 0 0 1 0 1 0 0	Negate	Yes	0-4	−(SA) → (SA)
ABS	0 0 0 0 0 1 1 1 0 1	Absolute value*	No	0-4	|(SA)| → (SA)
SWPB	0 0 0 0 0 1 1 0 1 1	Swap bytes	No	-	(SA), bits 0 thru 7 → (SA), bits 8 thru 15; (SA), bits 8 thru 15 → (SA), bits 0 thru 7.
INC	0 0 0 0 0 1 0 1 1 0	Increment	Yes	0-4	(SA) + 1 → (SA)
INCT	0 0 0 0 0 1 0 1 1 1	Increment by two	Yes	0-4	(SA) + 2 → (SA)
DEC	0 0 0 0 0 1 1 0 0 0	Decrement	Yes	0-4	(SA) − 1 → (SA)
DECT	0 0 0 0 0 1 1 0 0 1	Decrement by two	Yes	0-4	(SA) − 2 → (SA)
x†	0 0 0 0 0 1 0 0 1 0	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 will be accessed from PC and the PC will be updated accordingly. The instruction acquisition signal (IAQ) will not be true when the 9900 accesses the instruction at SA. Status bits are affected in the normal manner for the instruction executed.

A.7.5 CRU MULTIPLE-BIT INSTRUCTIONS.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
General format:	OP CODE						C				T_S		S			

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 3 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. T_S and S provide multiple mode addressing capability for the source operand. If 8 or fewer bits are transferred (C = 1 through 8), the source address is a byte address. If 9 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 1 if C = 1 through 8, and is incremented by 2 otherwise.

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 $1 \leq C \leq 8$.

A.7.6 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							

CRU relative addressing is used to address the selected CRU bit.

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 CRU output bit to 1.
SBZ	0 0 0 1 1 1 1 0	Set bit to zero	-	Set the selected CRU 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.

A.7.7 CRU JUMP INSTRUCTIONS.

General format:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
OP CODE								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 PC points to the next instruction. The 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 instruction.

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 no carry	ST3 = 1
JOP	0 0 0 1 1 1 0 0	Jump odd parity	ST5 = 1

A.7.8 SHIFT INSTRUCTION.

General format:

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

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.

A.7.9 IMMEDIATE REGISTER INSTRUCTIONS.

General format:

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td colspan="11">OP CODE</td><td>N</td><td colspan="4">W</td></tr>
<tr><td colspan="16">IOP</td></tr>
</table>

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MENAING	RESULT COMPARED TO 0	STATUS BITS AFFECTED	DESCRIPTION
AI	0 0 0 0 0 0 1 0 0 0 1	Add immediate	Yes	0-4	(W) + IOP → (W)
ANDI	0 0 0 0 0 0 1 0 0 1 0	AND immediate	Yes	0-2	(W) AND IOP → (W)
CI	0 0 0 0 0 0 1 0 1 0 0	Compare immediate	Yes	0-2	Compare (W) to IOP and set appropriate status bits
LI	0 0 0 0 0 0 1 0 0 0 0	Load immediate	Yes	0-2	IOP → (W)
ORI	0 0 0 0 0 0 1 0 0 1 1	OR immediate	Yes	0-2	(W) OR IOP → (W)

A.7.10 INTERNAL REGISTER LOAD IMMEDIATE INSTRUCTIONS.

General format:

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td colspan="11">OP CODE</td><td colspan="5">N</td></tr>
<tr><td colspan="16">IOP</td></tr>
</table>

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 → (WP), 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

A.7.11 INTERNAL REGISTER STORE INSTRUCTIONS.

General format:

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td colspan="11">OP CODE</td><td>N</td><td colspan="4">W</td></tr>
</table>

No ST bits are affected.

MNEMONIC	OP CODE 0 1 2 3 4 5 6 7 8 9 10	MEANING	DESCRIPTION
STST	0 0 0 0 0 0 1 0 1 1 0	Store status register	(ST) → (W)
STWP	0 0 0 0 0 0 1 0 1 0 1	Store workspace pointer	(WP) → (W)

A.7.12 RETURN WORKSPACE POINTER (RTWP) INSTRUCTION.

General format:

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td colspan="5">N</td></tr>
</table>

The RTWP instruction causes the following transfers to occur:

(WR15) → (ST)
(WR14) → (PC)
(WR13) → (WP)

A.7.13 EXTERNAL INSTRUCTIONS.

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

External instructions cause the three most-significant address lines (A0 through A2) to be set to the below-described levels and the CRUCLK line to be pulsed, allowing external control functions to be initiated.

MNEMONIC	OP CODE	MEANING	STATUS BITS AFFECTED	DESCRIPTION	ADDRESS BUS		
	0 1 2 3 4 5 6 7 8 9 10				A0	A1	A2
IDLE	0 0 0 0 0 0 1 1 0 1 0	Idle		Suspend SBP 9900A instruction execution until an interrupt, LOAD, or RESET occurs	L	H	L
RSET	0 0 0 0 0 0 1 1 0 1 1	Reset	12-15	0 → ST12 thru ST15	L	H	H
CKOF	0 0 0 0 0 0 1 1 1 1 0	User defined		---	H	H	L
CKON	0 0 0 0 0 0 1 1 1 0 1	User defined		---	H	L	H
LREX	0 0 0 0 0 0 1 1 1 1 1	User defined		---	H	H	H

A.8 MICROINSTRUCTION CYCLE

The SBP 9900A includes circuitry which will indicate the completion of a microinstruction cycle. Designated as the CYCEND function, it provides CPU status that can simplify system design. The CYCEND output will go to a low logic level as a result of the low-to-high transition of each clock pulse which initiates the last clock of a microinstruction.

A.9 SBP 9900A INSTRUCTION EXECUTION TIMES

Instruction execution times for the SBP 9900A are a function of:

1) Clock cycle time, t_c

2) Addressing mode used where operands have multiple addressing mode capability

3) Number of wait states required per memory access.

Table A-6 lists the number of clock cycles and memory accesses required to execute each SBP 9900A instruction. For instructions with multiple addressing modes for either or both operands, the table lists the number of clock cycles and memory accesses with all operands addressed in the workspace-register mode. To determine the additional number of clock cycles and memory accesses required for modified addressing, add the appropriate values from the referenced tables. The total instruction-execution time for an instruction is:

$$T = t_c\,(C + W \cdot M)$$

where:

T = total instruction execution time;
t_c = clock cycle time;
C = number of clock cycles for instruction execution plus address modification;
W = number of required wait states per memory access for instruction execution plus address modification;
M = number of memory accesses.

TABLE 6
INSTRUCTION EXECUTION TIMES

INSTRUCTION	CLOCK CYCLES C	MEMORY ACCESS M	ADDRESS MODIFICATION†	
			SOURCE	DEST
A	14	4	A	A
AB	14	4	B	B
ABS (MSB = 0)	12	2	A	-
(MSB = 1)	14	3	A	-
AI	14	4	-	-
ANDI	14	4	-	-
B	8	2	A	-
BL	12	3	A	-
BLWP	26	6	A	-
C	14	3	A	A
CB	14	3	B	B
CI	14	3	-	-
CKOF	12	1	-	-
CKON	12	1	-	-
CLR	10	3	A	-
COC	14	3	A	-
CZC	14	3	A	-
DEC	10	3	A	-
DECT	10	3	A	-
DIV (ST4 is set)	16	3	A	-
DIV (ST4 is reset)*	92-124	6	A	-
IDLE	12	1	-	-
INC	10	3	A	-
INCT	10	3	A	-
INV	10	3	A	-
Jump (PC is changed)	10	1	-	-
(PC is not changed)	8	1	-	-
LDCR (C = 0)	52	3	A	-
(1 ≤ C ≤ 8)	20+2C	3	B	-
(9 ≤ C ≤ 15)	20+2C	3	A	-
LI	12	3	-	-
LIMI	14	2	-	-
LREX	12	1	-	-
RESET function	26	5	-	-
LOAD function	22	5	-	-
Interrupt context switch	22	5	-	-

INSTRUCTION	CLOCK CYCLES C	MEMORY ACCESS M	ADDRESS MODIFICATION†	
			SOURCE	DEST
LWPI	10	2	-	-
MOV	14	4	A	A
MOVB	14	4	B	B
MPY	52	5	A	-
NEG	12	3	A	-
ORI	14	4	-	-
RSET	12	1	-	-
RTWP	14	4	-	-
S	14	4	A	A
SB	14	4	B	B
SBO	12	2	-	-
SBZ	12	2	-	-
SETO	10	3	A	-
Shift (C=0)	12+2C	3	-	-
(C=0, Bits 12-15 of WRO=0)	52	4	-	-
(C=0, Bits 12-15 of WRP=N=0)	20+2N	4	-	-
SOC	14	4	A	A
SOCB	14	4	B	B
STCR (C=0)	60	4	A	-
(1 ≤ C ≤ 7)	42	4	B	-
(C=8)	44	4	B	-
(9 ≤ C ≤ 15)	58	4	A	-
STST	8	2	-	-
STWP	8	2	-	-
SWPB	10	3	A	-
SZC	14	4	A	A
SZCB	14	4	B	B
TB	12	2	-	-
X**	8	2	A	-
XOP	36	8	A	-
XOR	14	4	A	-
Undefined op codes 0000-01FF, 0320-033F, 0C00-0FFF, 0780-07FF	6	1	-	-

*Execution time is dependent upon the partial quotient after each clock cycle during execution.

**Execution time is added to the execution time of the instruction located at the source address minus 4 clock cycles and 1 memory access time.

†The letters A and B refer to the respective tables that follow.

ADDRESS MODIFICATION - TABLE A

ADDRESSING MODE	CLOCK CYCLES C	MEMORY ACCESSES M
WR (T_S or T_D = 00)	0	0
WR indirect (T_S or T_D = 01)	4	1
WR indirect auto-increment (T_S or T_D = 11)	8	2
Symbolic (T_S or T_D = 10, S or D = 0)	8	1
Indexed (T_S or T_D = 10, S or D = 0)	8	2

ADDRESS MODIFICATION - TABLE B

ADDRESSING MODE	CLOCK CYCLES C	MEMORY ACCESSES M
WR (T_S or T_D = 00)	0	0
WR indirect (T_S or T_D = 01)	4	1
WR indirect auto-increment (T_S or T_D = 11)	6	2
Symbolic (T_S or T_D = 10, S or D = 0)	8	1
Indexed (T_S or T_D = 10, S or D = 0)	8	2

As an example, the instruction MOVB is used in a system with $t_C = 0.333\ \mu s$, and no wait states are required to access memory. Both operands are addressed in the workspace register mode:

$$T = t_C\,(C + W\cdot M) = 0.333\,(14 + 0\cdot 4)\ \mu s = 4.662\ \mu s$$

If two wait states per memory access were required, the execution time is:

$$T = 0.333\,(14 + 2\cdot 4)\ \mu s = 7.326\ \mu s$$

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

$$\begin{aligned} T &= t_C\,(C + W\cdot M) \\ &\quad C = 14 + 8 = 22 \\ &\quad M = 4 + 1 = 5 \\ T &= 0.333\,(22 + 2\cdot 5)\ \mu s = 10.659\ \mu s \end{aligned}$$

A.10 CLOCK FREQUENCY VS. TEMPERATURE

Stability of the operational frequency over the full temperature range of −55° C to 125° C,is shown in Figure A-6. The effects of temperature on clock frequency are nil above 25° C. Below 25° C the effects are typically less than −8%.

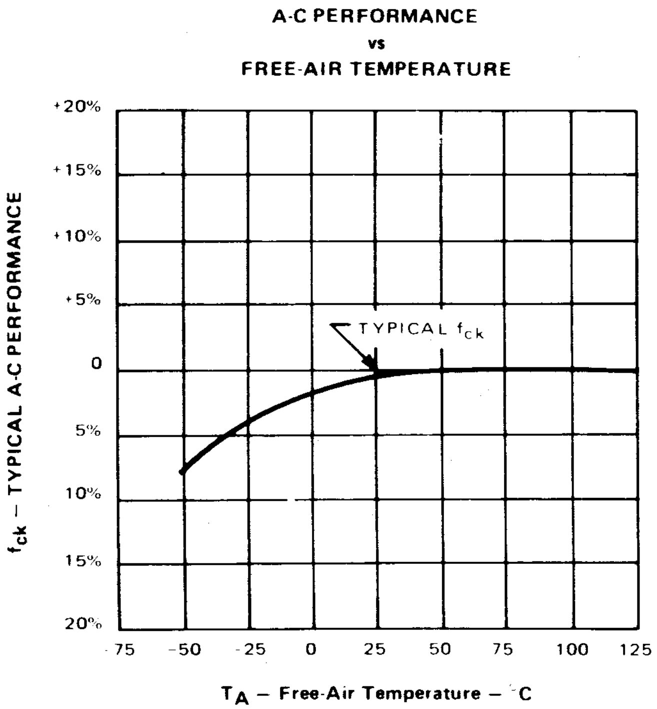


Figure A-2. SBP 9900A A-C Performance vs Temperature

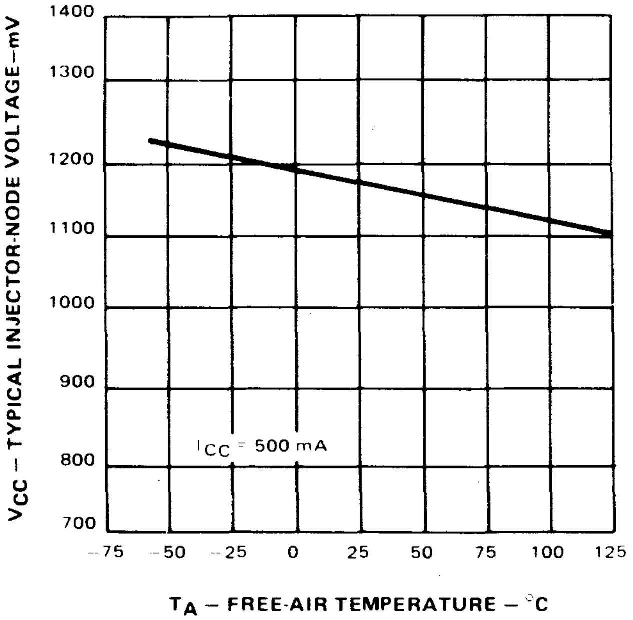


Figure A-3. SBP 9900A Injector Node Voltage vs Free-Air Temperature

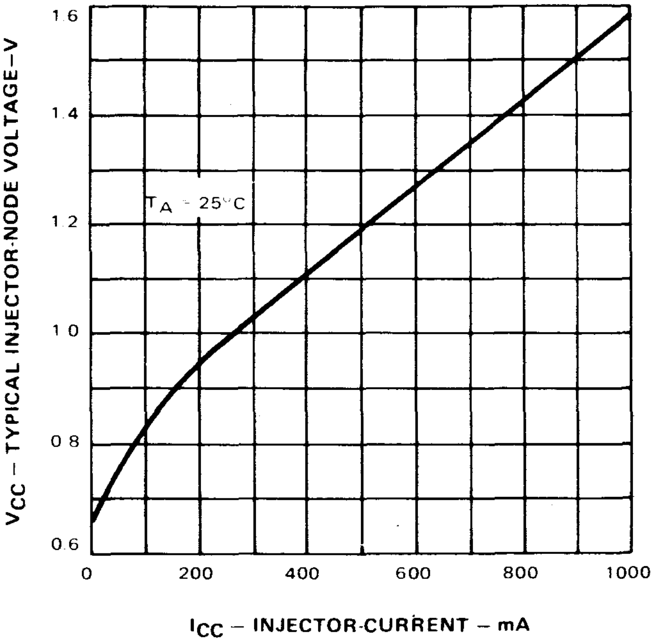
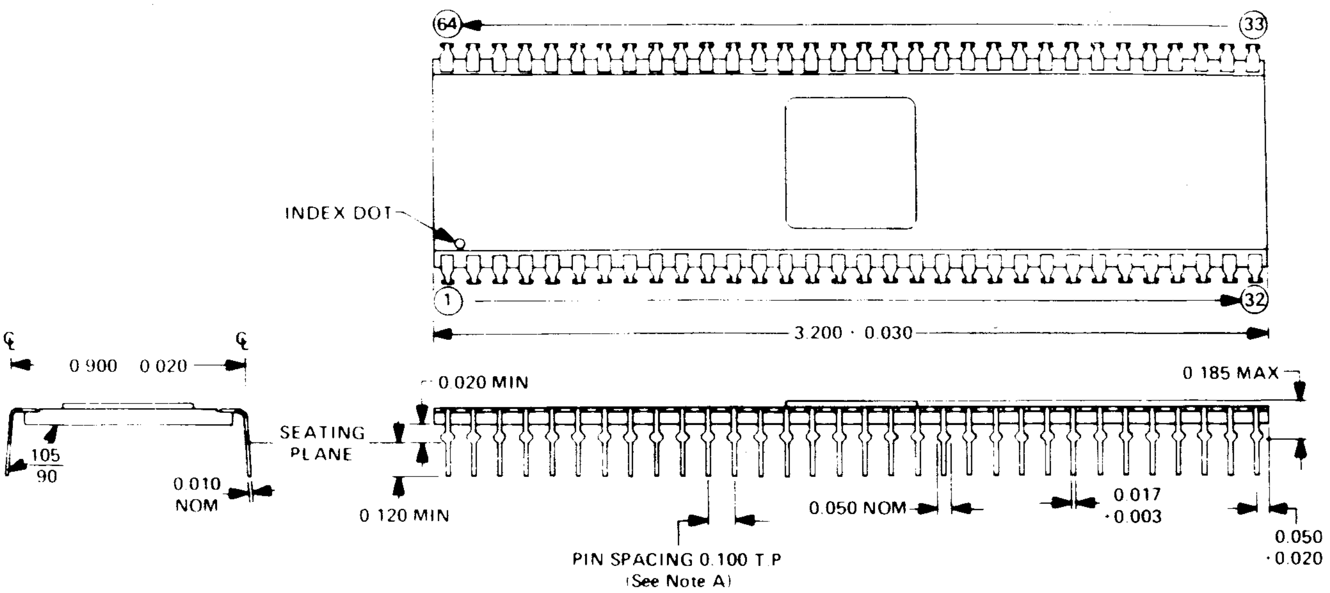


Figure A-4. SBP 9900A Injector Node Voltage vs Injector Current

A.11 MECHANICAL DATA



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

A.12 ELECTRICAL AND MECHANICAL SPECIFICATIONS

A.12.1 SBP 9900A RECOMMENDED OPERATING CONDITIONS, UNLESS OTHERWISE NOTED I_{CC} = 500 mA.

		MIN	NOM	MAX	UNIT
Supply current, I_{CC}	SBP 9900A MJ/NJ/EJ	450	500	550	mA
	SBP 9900ACJ	500	560	620	
High-level output voltage, V_{OH}				5.5	V
Low-level output current, I_{OL}				20	mA
Clock frequency, f_{clock}		DC		3	MHz
Width of clock pulse, t_w	High (67%)	222			ns
	Low (33%)	111			
Clock rise time, t_r				20	ns
Clock fall time, t_f				20	ns
Setup time, t_{su} (see Figure 24)	$\overline{\text{HOLD}}$	95†			ns
	READY	70†			
	D0 – D15	65†			
	CRUIN	50†			
	$\overline{\text{INTREQ}}$	25†			
	IC0 – IC3	25†			
Hold time, t_h (see Figure 24)	$\overline{\text{HOLD}}$	25†			ns
	READY	50†			
	D0 – D15	40†			
	CRUIN	40†			
	$\overline{\text{INTREQ}}$	55†			
	IC0 – IC3	55†			
Operating free-air temperature, T_A	SBP 9900AMJ, SBP 9900ANJ	−55		125	°C
	SBP 9900AEJ	−40		85	
	SBP 9900ACJ	0		70	

† Rising edge of clock pulse is reference.

A.12.2 SBP 9900A ELECTRICAL CHARACTERISTICS (OVER RECOMMENDED OPERATING FREE-AIR TEMPERATURE RANGE, UNLESS OTHERWISE NOTED).

PARAMETER			TEST CONDITIONS†		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage				2			V
V_{IL}	Low-level input voltage						0.8	V
V_{IK}	Input clamp voltage		I_{CC} = MIN,	$I_I = -12$ mA			−1.5	V
I_{OH}	High-level output current	I/O Pins	I_{CC} = 500 mA,	V_{IH} = 2 V,			350	µA
		Other outputs	V_{IL} = 0.8 V,	V_{OH} = 2.4 V			50	µA
		I/O Pins	I_{CC} = 500 mA,	V_{IH} = 2 V			1	mA
		Other outputs	V_{IL} = 0.8 V,	V_{OH} = 5.5 V			250	µA
V_{OL}	Low-level output voltage		I_{CC} = 500 mA, V_{IL} = 0.8 V,	V_{IH} = 2 V I_{OL} = 20 mA			0.4	V
I_I	Input current	Clock	I_{CC} = 500 mA,	V_I = 2.4 V		0.6	1	mA
		All other inputs				200	300	µA

†For conditions shown as MAX, use the appropriate value specified under recommended operating conditions.

‡All typical values are at I_{CC} = 500 mA, T_A = 25°C.

A.12.3 SBP 9900A SWITCHING CHARACTERISTICS, (I_{CC} = NOM, T_A = RECOMMENDED OPERATING FREE-AIR TEMPERATURE RANGE UNLESS OTHERWISE NOTED) SEE FIGURES A-4 AND A-5.

PARAMETER	FROM	TO	TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
t_{max}	MAXIMUM CLOCK FREQUENCY		C_L = 150 pF	3			MHz
t_{PLH} or t_{PHL}	CLOCK	ADDRESS BUS (A0 – A14)			90	130	ns
t_{PLH} or t_{PHL}	CLOCK	DATA BUS (D0 – D15)			105	145	ns
t_{PLH} or t_{PHL}	CLOCK	WRITE ENABLE ($\overline{WE}$)			100	155	ns
t_{PLH} or t_{PHL}	CLOCK	CYCLE END ($\overline{CYCEND}$)			90	135	ns
t_{PLH} or t_{PHL}	CLOCK	DATA BUS IN (DBIN)			115	160	ns
t_{PLH} or t_{PHL}	CLOCK	MEMORY ENABLE ($\overline{MEMEN}$)			90	130	ns
t_{PLH} or t_{PHL}	CLOCK	CRU CLOCK (CRUCK)			95	145	ns
t_{PLH} or t_{PHL}	CLOCK	CRU DATA OUT (CRUOUT)			110	175	ns
t_{PLH} or t_{PHL}	CLOCK	HOLD ACKNOWLEDGE (HLDA)			190	290	ns
t_{PLH} or t_{PHL}	CLOCK	WAIT			90	130	ns
t_{PLH} or t_{PHL}	CLOCK	INSTRUCTION ACQUISITION (IAQ)			90	130	ns

‡All typical values are at 25°C.

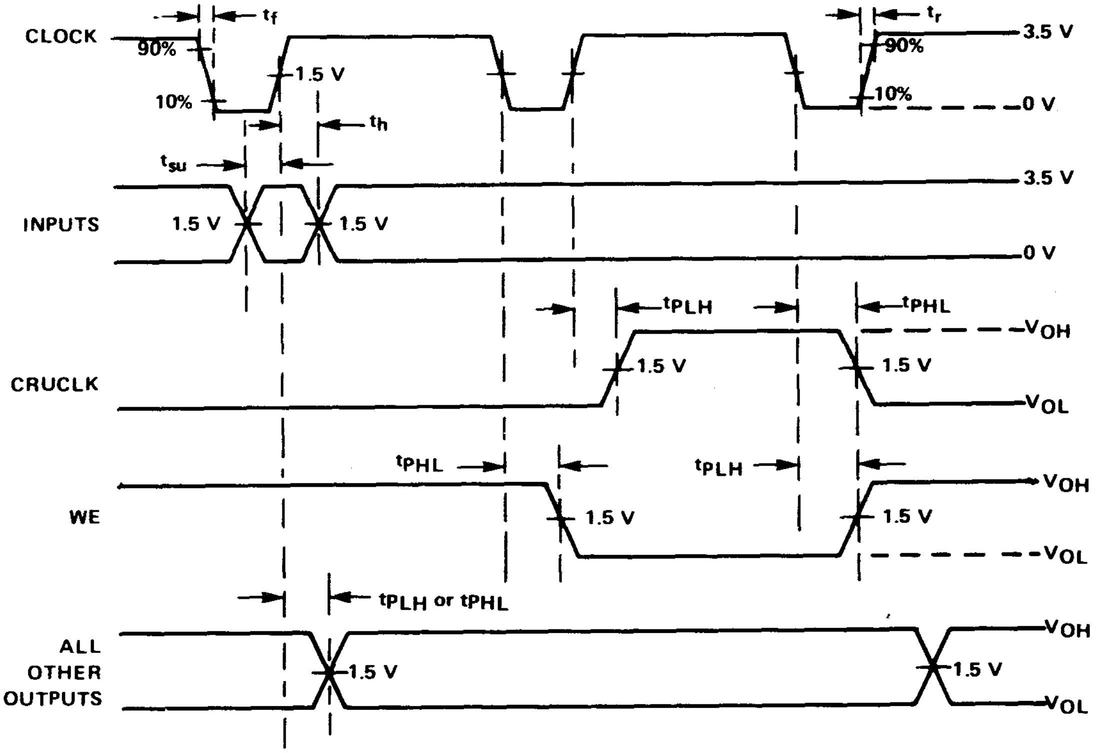


Figure A-5. Switching Times – Voltage Waveforms

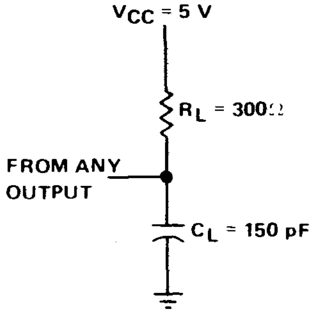


Figure A-6. Switching Times – Load Circuit

APPENDIX B

SBP 9900A MACHINE CYCLES

B.1 GENERAL DESCRIPTION OF MACHINE CYCLES

The SBP 9900A microprocessor executes a series of 2-clock machine cycles to perform an instruction or other operation. The SBP 9900As machine cycles, identical to those of the NMOS TMS 9900, are divided into three categories described in the following paragraphs.

B.1.1 ALU CYCLE. The ALU cycle performs an internal operation of the SBP 9900A. The memory interface control signal and CRU interface control signals are not affected by the execution of an ALU cycle. The ALU cycle takes two clock cycles to execute.

B.1.2 MEMORY CYCLE. The memory cycle primarily performs a data transfer between the SBP 9900A and an external memory device. Appropriate memory bus control signals are generated by the SBP 9900A as a result of memory cycle execution. The memory cycle takes 2+W (where W is the number of wait states) clock cycles to execute.

B.1.3 CRU CYCLE. The CRU cycle performs a bit transfer between the SBP 9900A and I/O devices. It takes two clock cycles per bit transfer to execute. The address of the CRU bit is set up during the first clock cycle. For an input operation the CRUIN line is sampled by the SBP 9900A during the second clock cycle. For an output operation the data bit is set up on the CRUOUT line at the same time the address is set up. The CRUCLK line is pulsed during the second clock cycle of the CRU output cycle. Refer to the SBP 9900A data manual for timing diagrams.

The SBP 9900A executes its operations under the control of a microprogrammed control ROM. Each microinstruction specifies a machine cycle. A microprogram specifies a sequence of machine cycles. The SBP 9900A executes a specific sequence of machine cycles for a specific operation. These sequences are detailed on the following pages. The information can be used by the systems designer to determine the bus contents and other interface behaviour at various instants during a specific SBP 9900A operation.

B.2 SBP 9900 MACHINE CYCLES SEQUENCES

Most SBP 9900A instruction executions consists of two parts: 1) data derivation and 2) operation execution. The data derivation sequence depends on the addressing mode for the data. Since the addressing modes are common to all instructions, the data derivation sequence is the same for same addressing modes irrespective of instruction. Therefore, the data derivation sequences are described first. These are then referred to in appropriate sequence during the instruction execution description.

B.3 TERMS AND DEFINITIONS

The following terms are used in describing the instructions of the SBP 9900A:

Term	Definition
B	Byte indicator (1 = byte, 0 = word)
C	Bit count
D	Destination address register
DA	Destination address
IOP	Immediate operand
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
SD	Source data register internal to the SBP 9900A microprocessor*
W	Workspace register
WRn	Workspace register n
(n)	Contents of n
Ns	Number of machine cycles to derive source operand
Nd	Number of machine cycles to derive destination operand
AB	Address Bus of the SBP 9900A
DB	Data Bus of the SBP 9900A
NC	No change from previous cycle

***NOTE**

The contents of the SD register remain latched at the last value written by the SBP 9900A unless changed by the ALU. Therefore, during all memory reads on ALU machine cycles the SD register and hence the data bus will contain the operand last written to the data bus by the SBP 9900A or the result of the last ALU cycle to have loaded the SD register.

B.4 DATA DERIVATION SEQUENCES

B.4.1 WORKSPACE REGISTER.

CYCLE	TYPE	DESCRIPTION

1	Memory Read	AB = Workspace register address
		DB = Operand

B.4.2 WORKSPACE REGISTER INDIRECT.

CYCLE	TYPE	DESCRIPTION
1	Memory Read	AB = Workspace register address
		DB = Workspace register contents
2	ALU	AB = NC
		DB = SD
3	Memory Read	AB = Workspace register content
		DB = Operand

B.4.3 WORKSPACE REGISTER INDIRECT AUTO-INCREMENT (BYTE OPERAND)

CYCLE	TYPE	DESCRIPTION
1	Memory Read	AB = Workspace register address
		DB = Workspace register contents
2	ALU	AB = NC
		DB = SD
3	Memory write	AB = Workspace register address
		DB = (WRn)+1
4	Memory Read	AB = Workspace register contents
		DB = Operand

B.4.4 WORKSPACE REGISTER INDRIECT AUTO-INCREMENT (WORD OPERAND)

CYCLE	TYPE	DESCRIPTION
1	Memory read	AB = Workspace register address
		DB = Workspace register contents
2	ALU	AB = NC
		DB = SD
3	ALU	AB = NC
		DB = SD
4	Memory write	AB = Workspace register address
		DB = (WRn)+2
5	Memory read	AB = Workspace register contents
		DB = Operand

B.4.5 SYMBOLIC

CYCLE	TYPE	DESCRIPTION
1	ALU	AB = NC
		DB = SD
2	ALU	AB = NC
		DB = SD
3	Memory read	AB = PC+2
		DB = Symbolic address
4	ALU	AB = NC
		DB = 0000_{16}
5	Memory read	AB = Symbolic address
		DB = Operand

B.4.6 INDEXED

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = Workspace register address
		DB = Workspace register contents

2	ALU	AB = NC
		DB = SD

3	Memory read	AB = PC+2
		DB = Symbolic address

4	ALU	AB = PC+2
		DB = Workspace register contents

5	Memory read	AB = Symbolic address+(WRn)
		DB = Operand

B.5 INSTRUCTION EXECUTION SEQUENCES

B.5.1 A, AB, C, CB, S, SB, SOC, SOCB, SZC, SZCB, MOV, MOVB, COC, CZC, XOR

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate sequence for source data addressing mode, from the data derivation sequences	

3+Ns	ALU	AB = NC
		DB = SD

Nd	Insert appropriate sequence for destination data addressing mode from the data derivation sequences	

4+Ns+Nd	ALU	AB = NC
		DB = SD

5+Ns+Nd	Memory write	AB = DA (Note 4)
		DB = Result

NOTES:

1) Since the memory operations of the SBP 9900A microprocessor family fetch or store 16-bit words; the source and the destination data fetched for byte operations are 16-bit words. The ALU operates on the specified bytes of these words and modifies the appropriate byte in the destination word. The adjacent byte in the destination word remains unaltered. At the completion of the instruction, the destination word, consisting of the modified byte and the adjacent unmodified byte, is stored in a single-memory write operation.

2) For MOVB instruction the destination data word (16 bits) is fetched. The specified byte in the destination word is replaced with the specified byte of the source-data word. The resultant destination word is then stored at the destination address.

3) For MOV instruction the destination data word (16 bits) is fetched although not used.

4) For C, CB, COC, CZC instructions cycle $5+N_s+N_d$ above is an ALU cycle with AB = DA and DB = SD.

B.5.2 MPY (MULTIPLY)

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data (multiplier) addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	Memory read	AB = Workspace register address
		DB = Workspace register contents

5+Ns	ALU	AB = NC
		DB =SD

6+Ns	ALU	AB = NC
		DB = Multiplier

7+Ns		Multiply the two operands

	16 ALU	AB = NC
		DB = MSH* of partial product

23+Ns	Memory write	AB = Workspace register address
		DB = MSH of the product

24+Ns	ALU	AB = DA+2
		DB = MSH of product

25+Ns	Memory write	AB = DA+2
		DB = LSH of the product

*MSH = Most Significant Half, LSH = Least Significant Half

B.5.3 DIV (DIVIDE)

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data (divisor) addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	Memory read	AB = Address of workspace register DB = Contents of workspace register

5+Ns	ALU	(Check overflow)
		AB = NC
		DB = Divisor

6+Ns	ALU	(Skip if overflow to next instruction fetch)
		AB = NC
		DB = SD

7+Ns	Memory read	AB = DA+2
		DB = Contents of DA+2

8+Ns	ALU	AB = NC
		DB = SD

9+Ns	ALU	AB = NC
		DB = SD

	Divide sequence consisting of Ni cycles where $32 \leq Ni \leq 48$. Ni is data dependent	DB = SD

10+Ns+Ni	ALU	AB = NC
		DB = SD

11+Ns+Ni	Memory write	AB = Workspace register address Quotient

12+Ns+Ni	ALU	AB = DA+2
		DB = Quotient

13+Ns+Ni	Memory write	AB = DA+2
		DB = Remainder

B.5.4 XOP

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	Instruction decode AB = NC DB = SD
Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	ALU	AB = NC
		DB = SA

5+Ns	ALU	AB = NC
		DB = SD

6+Ns	Memory read	AB = 40_{16} + (4 × D)
		DB = New workspace pointer

7+Ns	ALU	AB = NC
		DB = SA

8+Ns	Memory write	AB = Address of WR11
		DB = SA

9+Ns	ALU	AB = Address of WR15
		DB = SA

10+Ns	Memory write	AB = Address of workspace register 15 DB = Status register contents

11+Ns	ALU	AB = NC
		DB = PC+2

12+Ns	Memory write	AB = Address of workspace register 14 DB = PC+2

13+Ns	ALU	AB = Address of WR13
		DB = SD

14+Ns	Memory write	AB = Address of workspace register 13 DB = WP

15+Ns	ALU	AB = NC
		DB = SD

16+Ns	Memory read	AB = 42_{16} + (4 × D)
		DB = New PC

17+Ns	ALU	AB = NC
		DB = SD

B.5.5 CLR, SETO, INV, NEG, INC, INCT, DEC, DECT, SWPB

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU*	AB = NC
		DB = SD

4+Ns	Memory write	AB = Source data address
		DB = Modified source data

*NEG has one additional ALU cycle before the Memory Write Cycle.

NOTE

The operand is fetched for CLR and SETO although not used.

B.5.6 ABS

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	Test source data
		AB = NC
		DB = SD

4+Ns	ALU	Jump to 5+Ns if data positive
		AB = NC
		DB = SD

5+ns	ALU	Negate source
		AB = NC
		DB = SD

6+Ns	Memory write	AB = Source data address
		DB = Modified source data

B.5.7 X

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert the appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

NOTE

Add sequence for the instruction specified by the operand, less cycle 1 (Memory Read) and cycle 2 (ALU).

B.5.8 B

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

NOTE

The source data is fetched, although it is not used.

B.5.9 BL

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	ALU	AB = Address of WR11
		DB = SD

5+Ns	Memory write	B = Address of WR11
		DB = PC+2

NOTE

The source data is fetched although it is not used.

B.5.10 BLWP

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	ALU	AB = Address of WR15
		DB = NC

5+Ns	Memory write	AB = Address of workspace register 15 DB = Status register contents

6+Ns	ALU	AB = NC
		DB = PC+2

7+Ns	Memory write	AB = Address of workspace register 14 DB = PC+2
8+Ns	ALU	AB = Address of workspace register 13 DB = SD
9+Ns	Memory write	AB = Address of workspace register 13 DB = WP

10+Ns	ALU	AB = NC
		DB = SD

11+Ns	Memory read	AB = Address of new PC
		DB = New PC

12+Ns	ALU	AB = NC
		DB = SD

B.5.11 LDCR

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	ALU	AB = NC
		DB = SD

5+Ns	ALU	AB = Address of WR12
		DB = SD

6+Ns	ALU	AB = Address of WR12
		DB = SD

7+Ns	Memory read	AB = Address of WR12
		DB = Contents of WR12

8+Ns	ALU	AB = NC
		DB = SD

C	Enable CRUCLK. Shift next bit onto CRUOUT line. Increment CRU bit address on AB. Iterate this sequence C times, where C is number of bits to be transferred.	AB = Address + 2
		Increments C Times
		DB = SD

9+Ns+C	ALU	AB = NC
		DB = SD

B.5.12 STCR

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

Ns	Insert appropriate data derivation sequence according to the source data addressing mode	

3+Ns	ALU	AB = NC
		DB = SD

4+Ns	Memory read	AB = Address of WR12
		DB = Contents of WR12

5+Ns	ALU	AB = NC
		DB = SD

6+Ns	ALU	AB = NC
		DB = SD

C	Input selected CRU bit. Increment CRU bit address. Iterate this sequence C times where C is the number of CRU bits to be input.	AB = Address + 2
		C times
		DB = SD

7+Ns+C	ALU	AB = NC
		DB = SD

8+Ns+C	ALU	AB = NC
		DB = SD

C'	Right adjust (with zero fill) byte (if C < 8) or word (if 8 < C < 16).	AB = NC
		DB = SD

C'	= 1 if C = 8 or 16 = 8 −C if C < 8 = 16 −C if 8 <C < 16	

9+Ns+C+C'	ALU	AB = NC
		DB = SD

10+Ns+C+C'	ALU	AB = NC
		DB = SD

11+Ns+C+C'	ALU	AB = Source address
		DB = SD

12+Ns+C+C'	Memory write	AB = Source address
		DB = I/O data

NOTE

For STCR instruction the 16-bit word at the source address is fetched. If the number of CRU bits to be transferred is ≤ 8, the CRU data is right justified (with zero fill) in the specified byte of the source word and source data word thus modified is then stored back in memory. If the bits to be transferred is > 8 then the source data fetched is not used. The CRU data in this case is right justified in 16-bit word which is then stored at the source address.

B.5.13 SBZ, SBO

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = NC
		DB = SD

4	Memory read	AB = Address of WR12
		DB = Contents of WR12

5	ALU	AB = NC
		DB = SD

6	CRU	Set CRUOUT = 0 for SBZ = 1 for SBO Enable CRUCLK AB = CRU bit address DB = SD

B.5.14 TB

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = NC
		SB = SD

4	Memory read	AB = Address of WR12
		DB = Contents of WR12

5	ALU	AB = NC
		DB = SD

6	CRU	Set ST(2) = CRUIN
		AB = Address of CRU bit
		DB = SD

B.5.15 JEQ, JGT, JH, JHE, JL, JLE, JLT, JMP, JNC, JNE, JNO, JOC, JOP

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	Skip to cycle #5 if SBP 9900 status satisfies the specified jump condition
		AB = NC
		DB = SD

4	ALU	AB = NC
		DB = Displacement value

5	ALU	AB = NC

B.5.16 SRA, SLA, SRL, SRC

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	Memory read	AB = Address of the workspace register DB = Contents of the workspace register

4	ALU	Skip to cycle #9 if C = 0
		C = Shift count
		AB = NC
		DB = SD

5	ALU	AB = NC
		DB = SD

6	Memory read	AB = Address of WRO
		DB = Contents of WRO

7	ALU	AB = Source address
		DB = SD

8	ALU	AB = NC
		DB = SD

9		AB = NC
		DB = SD

C	Shift the contents of the specified workspace register in the specified direction by the specified number of bits. Set appropriate status bits.	
9+C	Memory write	AB = Address of the workspace register DB = Result

10+C	ALU	Increment PC
		AB = NC
		DB = SD

B.5.17 AI, ANDI, ORI

CYCLE	TYPE	DESCRIPTION
1	Memory read	AB = PC
		DB = Instruction
2	ALU	AB = NC
		DB = SD
3	ALU	AB = NC
		DB = SD
4	Memory read	AB = PC+2
		DB = Immediate operand
5	Memory read	AB = Address of workspace register
		DB = Contents of workspace register
		AB = NC
6	ALU	DB = SD
7	Memory write	AB = Address of workspace register
		DB = Result of instruction

B.5.18 CI

CYCLE	TYPE	DESCRIPTION
1	Memory read	AB = PC
		DB = Instruction
2	ALU	AB = NC
		DB = NC
3	Memory read	AB = Address of workspace register
		DB = Contents of workspace register
4	ALU	AB = NC
		DB = SD
5	Memory read	AB = PC+2
		DB = Immediate operand
6	ALU	AB = NC
		DB = SD
7	ALU	AB = NC
		DB = SD

B.5.19 LI

CYCLE	TYPE	DESCRIPTION
1	Memory read	AB = PC
		DB = Instruction
2	ALU	AB = NC
		DB =SD
3	ALU	AB = NC
		DB = SD

4	Memory read	AB = PC+2
		DB = Immediate operand

5	ALU	AB = Address of workspace register DB = SD
6	Memory write	AB = Address of workspace register DB = Immediate operand

B.5.20 LWPI

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = NC
		DB = SD

4	Memory read	AB = PC+2
		DB = Immediate operand

5	ALU	AB = NC
		DB = SD

B.5.21 LIMI

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = NC
		DB = SD

4	Memory read	AB = PC+2
		DB = Immediate data

5	ALU	AB = NC
		DB = SD

6	ALU	AB = NC
		DB = SD

7	ALU	AB = NC
		DB = SD

B.5.22 STWP, STST

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = Address of workspace register DB = SD
4	Memory write	AB = Address of the workspace register DB = SBP 9900A internal register contents (WP or ST)

B.5.23 CKON, CKOF, LREX, RSET

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = NC
		DB = SD

4	CRU	Enable CRUCLK
		AB = External instruction code
		DB = SD

5	ALU	AB = NC
		DB = SD

6	ALU	AB = NC
		DB = SD

B.5.24 IDLE

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SD

3	ALU	AB = NC
		DB = SD

4	CRU	Enable CRUCLK
		AB = Idle code
		DB = SD

5	ALU	AB = NC
		DB = SD

6*	ALU	AB = NC
		DB = NC

*SBP 9900A enters the IDLE loop by returning to cycle 3 and repeating cycles 3 through 6. LOAD and INTREQ are tested during cycle 6.

B.5.25 RTWP

CYCLE	TYPE	DESCRIPTION

1	Memory read	AB = PC
		DB = Instruction

2	ALU	AB = NC
		DB = SP

3	ALU	WP+30

4	Memory read	AB = WSR 15
		DB = STATUS old

5	Memory read	AB = WSR 14
		DB = PC old

6	Memory read	AB = WSR 13
		DB = WP old

7	ALU	AB = PC
		DB = SD

B.6 MACHINE-CYCLE SEQUENCES IN RESPONSE TO EXTERNAL STIMULI

B.6.1 RESET

CYCLE	TYPE	DESCRIPTION

1*	ALU	AB = NC
		DB = SD

2	ALU	AB =NC
		DB = SD

3	ALU	AB = 0
		DB = 0

4	Memory read	AB = 0
		DB = Workspace pointer

5	ALU	AB = NC
		DB = Status

6	Memory write	AB = Address of WR15
		DB = Contents of Status register

7	ALU	AB = NC
		DB = PC

8	Memory write	AB = Address of workspace register 14 DB = PC+2

9	ALU	AB = Address of WR13
		DB = SD

10	Memory write	AB = Address of workspace register 13 DB = WP

11	ALU	AB = NC
		DB = SD

12	Memory read	AB = 2
		DB = New PC

13	ALU	AB = NC
		DB = SD

B.6.2 LOAD

CYCLE	TYPE	DESCRIPTION

1**	ALU	AB = NC
		DB = SD

2	Memory read	AB = $FFFC_{16}$
		DB = Contents of $FFFC_{16}$

3	ALU	AB = NC
		DB = Status

4	Memory write	AB = Address WR15 DB = Contents of status register

5	ALU	AB = NC
		DB = PC

6	Memory write	AB = Address of WR14
		DB = PC+2

7	ALU	AB = Address of WR13
		DB = SD

8	Memory write	AB = Address of workspace register 13 DB = WP

9	ALU	AB = NC
		DB = SD

10	Memory read	AB = FFFE
		DB = New PC

11	ALU	AB = NC
		DB = SD

*Occurs immediately after RESET is released.
**Occurs immediately after last clock cycle of preceding instruction.

B.6.3 INTERRUPTS

CYCLE	TYPE	DESCRIPTION

1*	ALU	AB = NC
		DB = SD

2	Memory read	AB = Address of interrupt vector DB = WP (New)

3	ALU	AB =NC
		DB = Old Status

4	Memory write	AB = Address of WR15
		DB = Old Status

5	ALU	AB = NC
		DB = Old PC

6	Memory write	AB = Address of workspace register 14 DB = Old PC

7	ALU	AB = Address of WR13
		DB = Old PC

8	Memory write	AB = Address of workspace register 13 DB = Old WP

9	ALU	AB = NC
		DB = Old WP

10	Memory read	AB = Address of second word of interrupt vector DB = New PC

11	ALU	AB = NC
		DB = Old WP

*Occurs immediately after last clock cycle of preceding instruction.

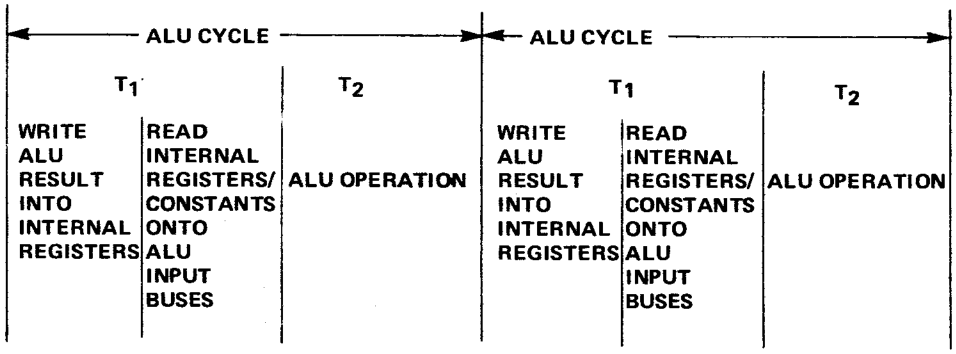

ALU CYCLE
ALU CYCLE
T1
T2
T1
T2
WRITE ALU RESULT INTO INTERNAL REGISTERS
READ INTERNAL REGISTERS/ CONSTANTS ONTO ALU INPUT BUSES
ALU OPERATION
WRITE ALU RESULT INTO INTERNAL REGISTERS
READ INTERNAL REGISTERS/ CONSTANTS ONTO ALU INPUT BUSES
ALU OPERATION

Figure B-1. ALU Cycle

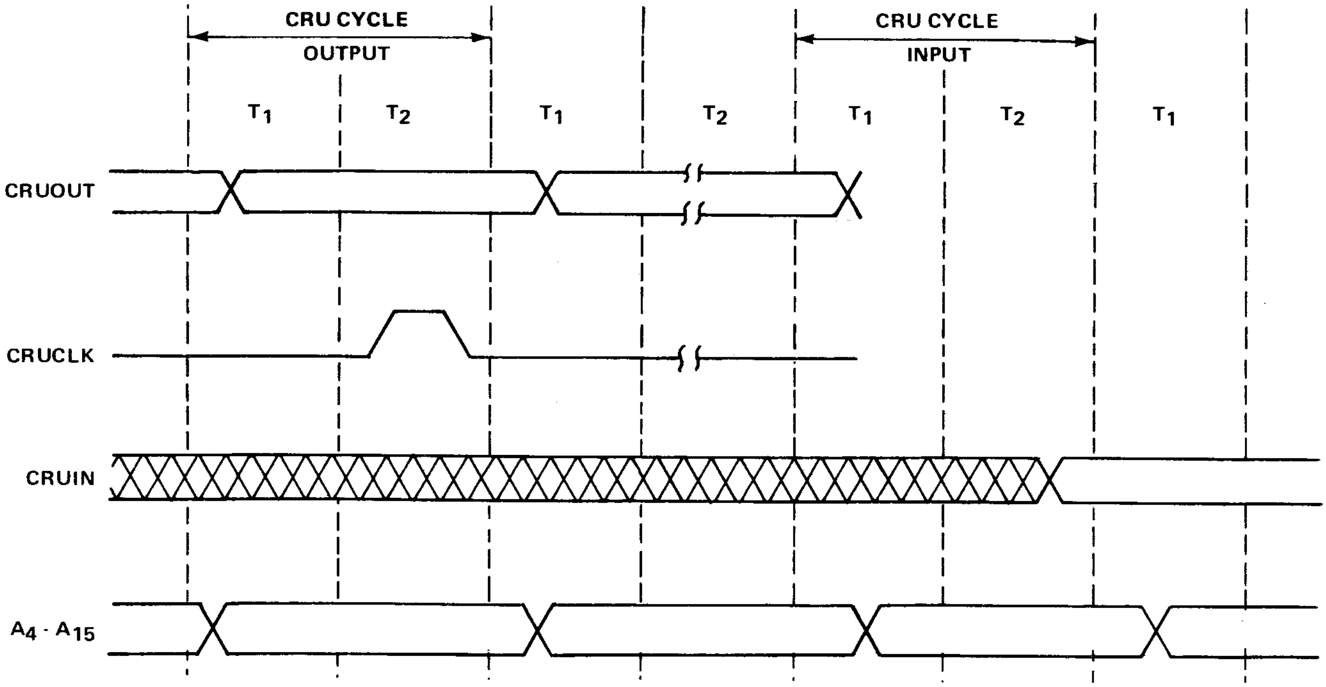


Figure B-2. CRU Cycle

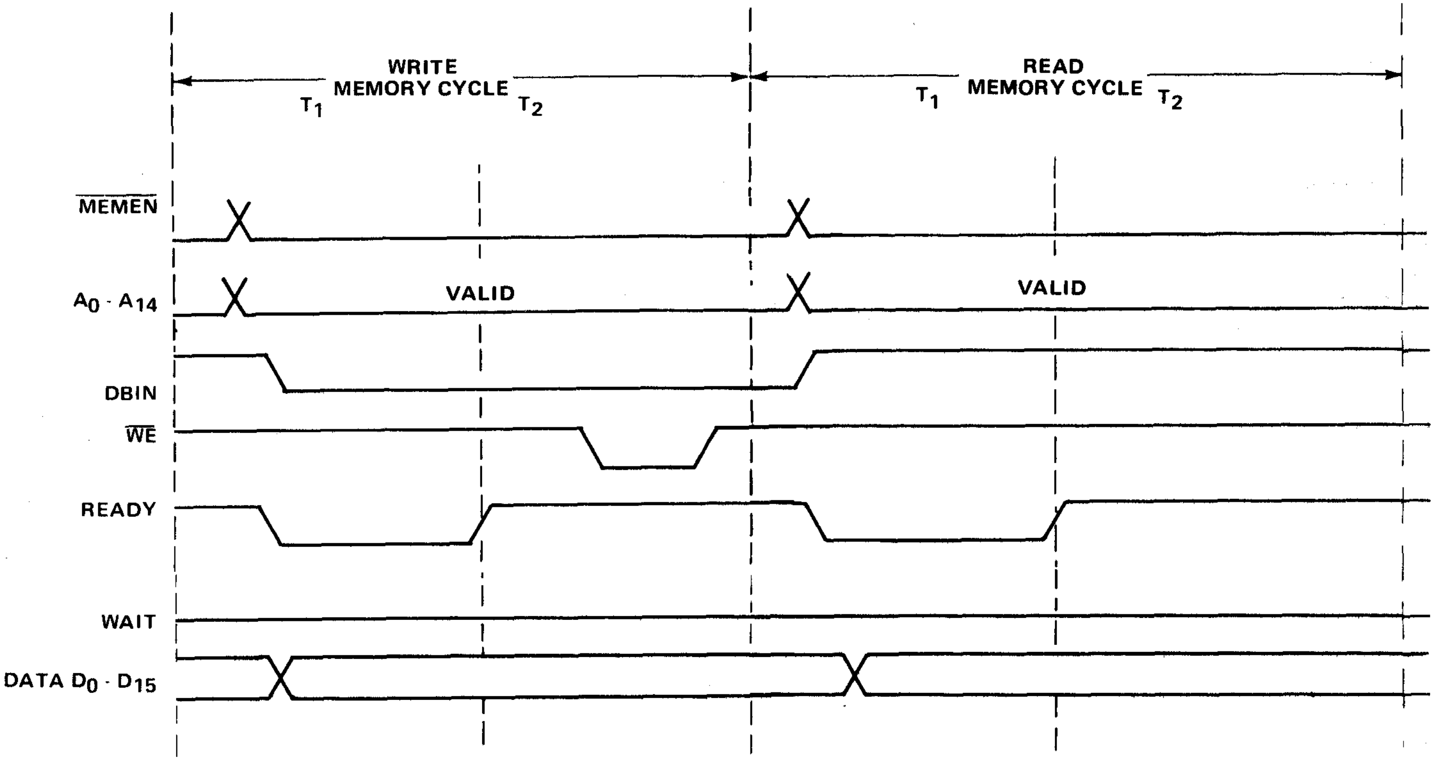


Figure B-3. SBP 9900A Memory Cycle (No Wait States)

APPENDIX C

SBP 9900A
HI-REL SPECIFICATION

MILITARY SPECIFICATION
MICROCIRCUITS, DIGITAL, MONOLITHIC SILICON
INTEGRATED INJECTION LOGIC (I^2L) PARALLEL 16-BIT MICROPROCESSOR

This specification is approved for use by all Departments and Agencies of the Department of Defense.

1. SCOPE

1.1 Scope. This specification covers the detail requirements for a monolithic silicon integrated injection logic (I^2L), parallel 16-bit microprocessor. Three product assurance classes and a choice of case outline/lead finish are provided for each type and are reflected in the complete part number.

1.1.1 Microcircuit group.

1.2 Part number. The part number shall be in accordance with MIL-M-38510.

M38510	460	01	B	TBD	C
Military Designator	Detail Specification	Device Type (1.2.1)	Device Class (1.2.2)	Case Outline (1.2.3)	Lead Finish (3.3)

1.2.1 Device type. The device type shall be as follows:

DEVICE TYPE	CIRCUIT
01	16-bit fixed instruction microprocessor

1.2.2 Device class. The device class shall be the product assurance level as defined in MIL-M-38510.

1.2.3 Case outline. The case outline shall be designated as follows:

OUTLINE LETTER	
To be determined	see figure 26 (0.800 x 3.20", 64-pin dual-in-line)

1.2.4 Absolute maximum ratings.

Supply Current, I_{CC}	550mA
Input Voltage, V_{IN} (all except CLK input)	+5.5V DC
V_{CLKIN}	+2.0V DC
Power Dissipation, P_D	750mW
Junction Temperature, T_J	175°C
Thermal Resistance, Junction to Case, θ_{JC}	25°C/watt
Lead Temperature (soldering 10 sec), L_T	300°C
Storage Temperature Range, T_{STG}	-65°C to +150°C

1.2.5 Recommended operating conditions, unless otherwise noted I_{CC}=500mA

		MIN	NOM	MAX	UNIT
Supply current, I_{CC}		450	500	550	mA
High-level output voltage, V_{OH}				5.5	V
Low-level output current, I_{OL}				20	mA
Clock frequency, f_{clock}		DC		3	MHz
Width of clock pulse, t_w	High (67%)	222			ns
	Low (33%)	111			
Clock rise time, t_r				20	ns
Clock fall time, t_f				20	ns
Setup time, t_{su} (see Figure 24)	$\overline{\text{HOLD}}$	95[†]			ns
	READY	70[†]			
	D0 – D15	65[†]			
	CRUIN	50[†]			
	$\overline{\text{INTREQ}}$	25[†]			
	IC0 – IC3	25[†]			
Hold time, t_h (see Figure 24)	$\overline{\text{HOLD}}$	25[†]			ns
	READY	50[†]			
	D0 – D15	40[†]			
	CRUIN	40[†]			
	$\overline{\text{INTREQ}}$	55[†]			
	IC0 – IC3	55[†]			
Operating free-air temperature, T_A		–55		125	°C

[†] Rising edge of clock pulse is reference.

TABLE I. Electrical performance characteristics

Performance characteristics. Performance characteristics shall be as specified herein, and apply over the full recommended ambient operating temperature range, unless otherwise specified.

ELECTRICAL CHARACTERISTICS (OVER RECOMMENDED OPERATING FREE-AIR TEMPERATURE RANGE, UNLESS OTHERWISE NOTED)

PARAMETER			TEST CONDITIONS[†]		MIN	MAX	UNIT
V_{IH}	High-level input voltage				2		V
V_{IL}	Low-level input voltage					0.8	V
V_{IK}	Input clamp voltage		I_{CC} = MIN,	I_I = –12 mA		–1.5	V
I_{OH}	High-level output current	I/O Pins	I_{CC} = 500 mA,	V_{IH} = 2 V,		300	μA
		Other outputs	V_{IL} = 0.8 V,	V_{OH} = 2.4 V		50	μA
		I/O Pins	I_{CC} = 500 mA,	V_{IH} = 2 V		1	mA
		Other outputs	V_{IL} = 0.8 V,	V_{OH} = 5.5 V		250	μA
V_{OL}	Low-level output voltage		I_{CC} = 500 mA, V_{IL} = 0.8 V,	V_{IH} = 2 V I_{OL} = 20 mA		0.4	V
I_I	Input current	Clock	I_{CC} = 500 mA,	V_I = 2.4 V		1000	μA
		All other inputs				300	

[†] For conditions shown as MAX, use the appropriate value specified under recommended operating conditions.

TABLE I (cont.)

SWITCHING CHARACTERISTICS (I_{CC} = 500mA)
(see figures 24 and 25)

PARAMETER	FROM	TO	TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
t_{max}	MAXIMUM CLOCK FREQUENCY		C_L = 150 pF	3			MHz
t_{PLH} or t_{PHL}	CLOCK	ADDRESS BUS (A0 – A14)				130	ns
t_{PLH} or t_{PHL}	CLOCK	DATA BUS (D0 – D15)				145	ns
t_{PLH} or t_{PHL}	CLOCK	WRITE ENABLE ($\overline{WE}$)				155	ns
t_{PLH} or t_{PHL}	CLOCK	CYCLE END ($\overline{CYCEND}$)				135	ns
t_{PLH} or t_{PHL}	CLOCK	DATA BUS IN (DBIN)				160	ns
t_{PLH} or t_{PHL}	CLOCK	MEMORY ENABLE ($\overline{MEMEN}$)				130	ns
t_{PLH} or t_{PHL}	CLOCK	CRU CLOCK (CRUCK)				145	ns
t_{PLH} or t_{PHL}	CLOCK	CRU DATA OUT (CRUOUT)				175	ns
t_{PLH} or t_{PHL}	CLOCK	HOLD ACKNOWLEDGE (HLDA)				290	ns
t_{PLH} or t_{PHL}	CLOCK	WAIT				130	ns
t_{PLH} or t_{PHL}	CLOCK	INSTRUCTION ACQUISITION (IAQ)				130	ns

2. APPLICABLE DOCUMENTS

2.1 Issues of documents. The following documents, of the issue in effect on date of invitation for bids or request for proposal, form a part of this specification to the extent specified herein.

SPECIFICATION

MILITARY

MIL-M-38510 - Microcircuits, General Specification for.

STANDARD

MILITARY

MIL-STD-883 - Test Methods and Procedures for Microelectronic

(Copies of specifications, standards, drawings, and publications required by contractors in connection with specific procurement functions should be obtained from the procuring activity or as directed by the contracting officer.)

3. REQUIREMENTS

3.1 Detail specifications. The individual item requirements shall be in accordance with MIL-M-38510, and as specified herein. In the event of conflict between MIL-M-38510 and this detail specification, this detail specification shall govern.

3.2 Design, construction, and physical dimensions. The design, construction, and physical dimensions shall be as specified in MIL-M-38510, and herein.

3.2.1 Terminal connections. Terminal connections (pin assignments) shall be as specified on figure 11.

3.2.2 Block diagram. The block diagram (architecture) shall be as specified on figure 1.

3.2.3 Memory map. The memory map shall be as specified on figure 4.

3.2.4 Functional description, terms, instructions and symbols. Functional description, terms, instructions, and symbols shall be as specified in paragraph 6.4.

3.2.5 Case outline. The case outline shall be in accordance with paragraph 1.2.3, see figure 26.

3.3 Lead material and finish. Lead material and finish shall be in accordance with MIL-M-38510.

3.4 Electrical performance characteristics. The electrical performance characteristics are as specified in table I, and apply over the full recommended ambient operating temperature range, unless otherwise specified.

3.5 Rebonding. Rebonding shall be per MIL-M-38510 and as specified herein.

3.6 Electrical test requirements. Electrical test requirements shall be as specified in table III for the applicable device type and device class. The subgroups of table III which constitute the minimum electrical test requirements for screening, qualification, and quality conformance by device class are specified in table II.

TABLE II. Electrical test requirements.

MIL-STD-883 TEST REQUIREMENT[1]	CLASS A DEVICES	CLASS B DEVICES	CLASS C DEVICES
Interim electrical parameters (pre burn-in) (method 5004)	1,7,9	1,7,9	None
Final electrical test parameters[2] (method 5004)	1,2,3,7,8 9,10,11	1,2,3,7,8 9,10,11	1,7,9
Group A test requirements (method 5005)	1,2,3,7,8 9,10,11	1,2,3,7,8 9,10,11	1,2,3,7,8, 9,10,11
Groups C and D endpoint electrical parameters (method 5005)	1,2,3,7,8 9,10,11	1,7,9	1,7,9
Additional electrical subgroups for Group C periodic inspections	None	2,3,8, 10,11	None

NOTES

1) Subgroup 4,5&6 may be substituted for subgroups 7,8,9,10&11.

2) PDA applies to subgroup 1,7,9 combined (see 4.3g) or subgroups 1&4.

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATEDARE OPEN)																				MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10-24	30	31	32	33-36	41-56	59	60	61	62	63	64		SUBGROUP T_A= 25°C		
		GND	INJ	WAIT	$\overline{LOAD}$	HOLDA	$\overline{RESET}$	IAQ	CLOCK	A14-A0	CRUOUT	CRUIN	$\overline{INTREQ}$	IC3-IC0	D0-D15	$\overline{CYCEND}$	CRUCLK	$\overline{WE}$	READY	$\overline{MEMEN}$	$\overline{HOLD}$		MIN	MAX	
V_{IC}	1	GND	450 ma		-12 ma																	4		-1.5	V
	2						-12 ma															6			
	3								-12 ma													8			
	4											-12 ma										31			
	5												-12 ma									32			
	6													-12 ma								33			
	7																					34			
	8																					35			
	9													↓								36			
	10				2.0V		A		A7				2.0V	0.8V	-12 ma				2.0V		2.0V	41			
	11																					42			
	12																					43			
	13																					44			
	14																					45			
	15																					46			
	16																					47			
	17																					48			
	18																					49			
	19																					50			
	20																					51			
	21																					52			
	22																					53			
	23																					54			
	24																					55			
	25				↓		↓		↓				↓		↓				↓		↓	56			
↓	26	↓	↓																-12 ma			62		↓	↓

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																				MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10-24	30	31	32	33-36	41-56	59	60	61	62	63	64		SUBGROUP T_A= 25°C		
		GND	INJ	WAIT	$\overline{LOAD}$	HOLDA	$\overline{RESET}$	IAQ	CLOCK	A14-A0	CRUOUT	CRUIN	$\overline{INTREQ}$	IC3-IC0	D0-D15	$\overline{CYCEND}$	CRUCLK	$\overline{WE}$	READY	$\overline{MEMEN}$	$\overline{HOLD}$		MIN	MAX	
V_{IC}	27	GND	450 ma																		-12 ma	64		-1.5	V
I_{IH}	28		500 ma		2.4v																	4		250	μa
	29						2.4v															6		250	
	30								2.4v													8		1000	
	31											2.4v										31		300	
	32												2.4v									32			
	33													2.4v								33			
	34																					34			
	35																					35			
	36													↓								36			
	37				2.0v		A		A7				2.0V	0.8V	2.4v				2.0V		2.0V	41			
	38																					42			
	39																					43			
	40																					44			
	41																					45			
	42																					46			
	43																					47			
	44																					48			
	45																					49			
	46																					50			
	47																					51			
	48																					52			
	49																					53			
	50																					54			
	51																					55			
↓	52	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	56		↓	↓

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																				MEASUREMENT TERMINAL	TEST LIMITS		
		1,2 27,29	9,26 40,57	3	4	5	6	7	8	10- 24	30	31	32	33- 36	41- 56	59	60	61	62	63	64				
		GND	INJ	WAIT	$\overline{\text{LOAD}}$	HOLDA	$\overline{\text{RESET}}$	IAQ	CLOCK	A14- A0	CRUOUT	CRUIN	$\overline{\text{INTREQ}}$	IC3- IC0	D0- D15	$\overline{\text{CYCEND}}$	CRUCLK	$\overline{\text{WE}}$	READY	$\overline{\text{MEMEN}}$	$\overline{\text{HOLD}}$		SUBGROUP T_A = 25°C		
																							MIN	MAX	
I_{IH}	53	GND	500 ma																2.4v			62		300	μa
↓	54																				2.4v	64		↓	↓
V_{OL}	55			20 ma	2.0v		A		A1				2.0v	0.8v					2.0V		2.0v	3		0.4	V
	56					20 ma			↓													5			
	57							20 ma	A3													7			
	58								A5	20 ma												10			
	59																					11			
	60																					12			
	61																					13			
	62																					14			
	63																					15			
	64																					16			
	65																					17			
	66																					18			
	67																					19			
	68																					20			
	69																					21			
	70																					22			
	71																					23			
	72																					24			
	73								A30		20ma.	0.8v										30			
	74								A5						20 ma							41			
	75																					42			
	76																					43			
	77																					44			
	78																					45			

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																				MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10-24	30	31	32	33-36	41-56	59	60	61	62	63	64		SUBGROUP T_A = 25°C		
		GND	INJ	WAIT	$\overline{\text{LOAD}}$	HOLDA	$\overline{\text{RESET}}$	IAQ	CLOCK	A14-A0	CRUOUT	CRUIN	$\overline{\text{INTREQ}}$	IC3-IC0	D0-D15	$\overline{\text{CYCEND}}$	CRUCLK	$\overline{\text{WE}}$	READY	$\overline{\text{MEMEN}}$	$\overline{\text{HOLD}}$		MIN	MAX	
V_{OL}	79	GND	500 ma		2.0V		A		A5				2.0V	0.8V	20 ma				2.0V		2.0V	46		0.4	V
↓	80	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	47		↓	↓
↓	81	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	48		↓	↓
↓	82	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	49		↓	↓
↓	83	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	50		↓	↓
↓	84	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	51		↓	↓
↓	85	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	52		↓	↓
↓	86	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	53		↓	↓
↓	87	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	54		↓	↓
↓	88	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	55		↓	↓
↓	89	↓	↓		↓		↓		↓				↓	↓	↓				↓		↓	56		↓	↓
↓	90	↓	↓		↓		↓		A2				↓	↓		20 ma			↓		↓	59		↓	↓
↓	91	↓	↓				0.8V										20 ma					60		↓	↓
↓	92	↓	↓		2.0V		A		A12				2.0V	0.8V				20 ma	2.0V		2.0V	61		↓	↓
↓	93	↓	↓		↓		↓		A7				↓	↓					↓	20 ma	↓	63		↓	↓
I_{OH}	94	↓	↓	5.5V	↓		↓		A8				↓	↓					0.8V		↓	3		250	μa
↓	95	↓	↓		↓	5.5V	↓		A1				↓	↓					2.0V		0.8V	5		↓	↓
↓	96	↓	↓		↓		↓	5.5V	A27				↓	↓					↓		2.0V	7		↓	↓
↓	97	↓	↓		↓		↓		↓	5.5V			↓	↓	5 ma				↓		↓	10		↓	↓
↓	98	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	11		↓	↓
↓	99	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	12		↓	↓
↓	100	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	13		↓	↓
↓	101	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	14		↓	↓
↓	102	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	15		↓	↓
↓	103	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	16		↓	↓
↓	104	↓	↓		↓		↓		↓	↓			↓	↓	↓				↓		↓	17		↓	↓

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																					MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10-24	30	31	32	33-36	41-56	59	60	61	62	63	64				
		GND	INJ	WAIT	$\overline{LOAD}$	HOLDA	$\overline{RESET}$	IAQ	CLOCK	A14-A0	CRUOUT	CRUIN	$\overline{INTREQ}$	IC3-IC0	D0-D15	$\overline{CYCEND}$	CRUCLK	$\overline{WE}$	READY	$\overline{MEMEN}$	$\overline{HOLD}$		SUBGROUP T_A=25°C		
																							MIN	MAX	
I_{OH}	105	GND	500 ma		2.0V		A		A27	5.5V			2.0V	0.8V	5 ma				2.0V		2.0V	18		250	ua
	106																					19			
	107																					20			
	108																					21			
	109																					22			
	110																					23			
	111						↓		↓	↓					↓							24			
	112						B		B46		5.5v				B							30		↓	↓
	113						A		A7						5.5V							41		1	mA
	114																					42			
	115																					43			
	116																					44			
	117																					45			
	118																					46			
	119																					47			
	120																					48			
	121																					49			
	122																					50			
	123																					51			
	124																					52			
	125																					53			
	126																					54			
	127																					55			
	128								↓						↓							56		↓	↓
	129						↓		A1							5.5V						59		250	UA
↓	130	↓	↓		↓		B		B46				↓	↓	B		5.5V		↓		↓	60		↓	↓

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																					MEASUREMENT TERMINAL	TEST LIMITS		
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10-24	29	31	32	33-36	41-56	59	60	61	62	63	64				
		GND	INJ	WAIT	$\overline{LOAD}$	HOLDA	$\overline{RESET}$	IAQ	CLOCK	A14-A0	DBIN	CRUIN	$\overline{INTREQ}$	IC3-IC0	D0-D15	$\overline{CYCEND}$	CRUCLK	$\overline{WE}$	READY	$\overline{MEMEN}$	$\overline{HOLD}$		SUBGROUP $T_A = 25°C$		
																							MIN	MAX	
V_{OL}	131	GND	500 ma		2.0v		A		A1		20ma		2.0v	0.8v					2.0v		2.0v	29		0.4	v
I_{OH}	132	GND	500 ma		2.0v		A		A7		5.5v		2.0v	0.8v					2.0v		2.0v	29		250	μa
I_{OH}	133	↓	↓		↓		↓		↓		2.4v		↓	↓					↓		↓	29		50	ua

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																				MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10-24	30	31	32	33-36	41-56	59	60	61	62	63	64		SUBGROUP T_A = 25°C		
		GND	INJ	WAIT	$\overline{\text{LOAD}}$	HOLDA	$\overline{\text{RESET}}$	IAQ	CLOCK	A14-A0	CRUOUT	CRUIN	$\overline{\text{INTREQ}}$	IC3-IC0	D0-D15	$\overline{\text{CYCEND}}$	CRUCLK	$\overline{\text{WE}}$	READY	$\overline{\text{MEMEN}}$	$\overline{\text{HOLD}}$		MIN	MAX	
I_{OH}	134	GND	500 ma				0.8V											5.5V				61		250	ma
I_{OH}	135	GND	500 ma		2.0V		A		A1				2.0V	0.8V					2.0V	5.5V	2.0V	63		↓	ma
I_{OH}	136	↓	↓				0.8V											2.4V				61		50	ua
I_{OH}	137				2.0		A		A1				2.0V	0.8V					2.0V	2.4V	2.0V	63		↓	ua

TABLE III. Group A Inspection DC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																				MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2 27,28	9,26 40,57	3	4	5	6	7	8	10- 24	30	31	32	33- 36	41- 56	59	60	61	62	63	64		SUBGROUP $T_A = 25°C$		
		GND	INJ	WAIT	LOAD	HOLDA	RESET	IAQ	CLOCK	A14- A0	CRUOUT	CRUIN	INTREQ	IC3- IC0	D0- D15	SYCEN	CRUCLK	WE	READY	MEMEN	HOLD		MIN	MAX	
I_{OH}	138	GND	500 ma		2.0V		A		A21	2.4V			2.0V	0.8V	5 ma				2.0V		2.0V	18		.50	ua
	139																					19			
	140																					20			
	141																					21			
	142																					22			
	143																					23			
	144																					24			
	145						B		B16		2.4V				B							30			
	146						A		A7						24V							41		350	ua
	147																					42			
	148																					43			
	149																					44			
	150																					45			
	151																					46			
	152																					47			
	153																					48			
	154																					49			
	155																					50			
	156																					51			
	157																					52			
	158																					53			
	159																					54			
	160																					55			
	161																					56			
	162								A1							2.4V						59		50	UA
	163						B		B46						B		24V					60			
Repeat Test 1 thru 163																						TA = 125°C			
Repeat Test 1 thru 163																						TA = -55°C			

TABLE IV. Group A Inspection AC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																					MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2,27,28	9,26,40,57	3	4	5	6	7	8	10-24	29	30	31	32	33-36	41-56	59	60	61	62	63	64				
		GND	INJ	WAIT	$\overline{\text{LOAD}}$	HOLDA	$\overline{\text{RESET}}$	IAQ	CLOCK	A14-A0	DBIN	CRUOUT	CRUIN	$\overline{\text{INTREQ}}$	IC3-IC0	D0-D15	$\overline{\text{CYCEND}}$	CRUCLK	$\overline{\text{WE}}$	READY	$\overline{\text{MEMEN}}$	$\overline{\text{HOLD}}$		SUBGROUP $T_A = 25°C$		
																								MIN	MAX	
t_{pLH}	1	GND	500 ma	D	3.0v		C		C8					3.0v	GND					D		3.0v	3		130	ns
	2					E			C9											3.0v		E	5		290	
	3							F	C27													3.0v	7		130	
	4								C11	F						G							10			
	5																						11			
	6																						12			
	7																						13			
	8																						14			
	9																						15			
	10																						16			
	11																						17			
	12																						18			
	13																						19			
	14																						20			
	15																						21			
	16																						22			
	17																						23			
	18																						24			
	19								C7		F												29		160	
	20								C41			F				G							30		175	
	21								C7							F							41		145	
	22																						42			
	23																						43			
	24																						44			
	25																						45			
	26																						46			

TABLE IV. Group A Inspection AC Parameters

SYMBOL	TEST No.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																					MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2,27,28	9,26,40,57	3	4	5	6	7	8	10-24	29	30	31	32	33-36	41-56	59	60	61	62	63	64				
		GND	INJ	WAIT	$\overline{LOAD}$	HOLDA	$\overline{RESET}$	IAQ	CLOCK	A14-A0	DBIN	CRUOUT	CRUIN	$\overline{INTREQ}$	IC3-IC0	D0-D15	$\overline{CYCEND}$	CRUCLK	$\overline{WE}$	READY	$\overline{MEMEN}$	$\overline{HOLD}$		SUBGROUP $T_A = 25°C$		
																								MIN	MAX	
t_{PLH}	27	GND	500 ma		3.0v		C		C7					3.0v	GND	F				3.0v		3.0v	47		145	NS
	28																						48			
	29																						49			
	30																						50			
	31																						51			
	32																						52			
	33																						53			
	34																						54			
	35																						55			
	36								↓							↓							56		↓	
	37								C3								F						59		135	
	38								C46							G		H					60		145	
	39								C13										I				61		155	
↓	40								C9											↓	F		63		130	
t_{PHL}	41			D					C9											D		↓	3		↓	
	42					E			C9											3.0v		E	5		290	
	43							F	C29													3.0v	7		130	
	44								C31	F						G							10			
	45																						11			
	46																						12			
	47																						13			
	48																						14			
	49																						15			
	50																						16			
	51																						17			
↓	52	↓	↓		↓		↓		↓	↓				↓	↓	↓				↓		↓	18		↓	↓

TABLE IV. Group A Inspection AC Parameters

SYMBOL	TEST No.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																						MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1, 2, 27, 28	9, 26, 40, 57	3	4	5	6	7	8	10-24	29	30	31	32	33-36	41-56	59	60	61	62	63	64					
		GND	INJ	WAIT	$\overline{\text{LOAD}}$	HOLDA	$\overline{\text{RESET}}$	IAQ	CLOCK	A14-A0	DBIN	CRUOUT	CRUIN	$\overline{\text{INTREQ}}$	IC3-IC0	D0-D15	$\overline{\text{CYCEND}}$	CRUCLK	$\overline{\text{WE}}$	READY	$\overline{\text{MEMEN}}$	$\overline{\text{HOLD}}$		SUBGROUP T_A=		
																								MIN	MAX	
t_{PHL}	53	GND	[illegible]		3.0V		C		C31	F				3.0V	GND	G				3.0V		3.0V	19		130	NS
	54																						20			
	55																						21			
	56																						22			
	57																						23			
	58								↓	↓						↓							24		↓	
	59								C9		F												29		160	
	60								C41			F				G							30		175	
	61								C38							G/J							41		145	
	62																						42			
	63																						43			
	64																						44			
	65																						45			
	66																						46			
	67																						47			
	68																						48			
	69																						49			
	70																						50			
	71																						51			
	72																						52			
	73																						53			
	74																						54			
	75																						55			
	76								↓							↓							56		↓	
	77								C2								F						59		135	
↓	78	↓	↓		↓		↓		C47					↓	↓	G		H		↓		↓	60		145	↓

TABLE IV. Group A Inspection AC Parameters

SYMBOL	TEST NO.	TERMINAL CONDITIONS (PINS NOT DESIGNATED ARE OPEN)																					MEASUREMENT TERMINAL	TEST LIMITS		UNITS
		1,2, 27,28	9,26, 40,57	3	4	5	6	7	8	10-24	29	30	31	32	33-36	41-56	59	60	61	62	63	64		SUBGROUP $T_A = 25°C$		
		GND	INJ	WAIT	$\overline{LOAD}$	HOLDA	$\overline{RESET}$	IAQ	CLOCK	A14-A0	DBIN	CRUOUT	CRUIN	$\overline{INTREQ}$	IC3-IC0	D0-D15	$\overline{CYCEND}$	CRUCLK	$\overline{WE}$	READY	$\overline{MEMEN}$	$\overline{HOLD}$		MIN	MAX	
t_{PHL} ↓	79 80	GND ↓	500 ma ↓		3.0v ↓		C ↓		C12 C7					3.0v ↓	GND ↓				I		F	3.0v ↓	61 63		155 130	NS ↓
Fmax	81	GND	500MA						SEE Note															3		MHZ
Repeat Test 1 Thru 81																								$TA = 125°C$		
Repeat Test 1 thru 81																								$TA = 125°C$		

NOTE - Using clock pulse and set-up and hold times specified on sheet 2, run all functional patterns specified on pages 73-166.

3.7 Marking. Marking shall be in accordance with MIL-M-38510 and 1.2. At the option of the manufacturer, the following marking may be omitted from the body of the microcircuit but shall be retained on the initial container.

a. Country of origin.

3.8 Microcircuit group assignment. The devices covered by this specification shall be in microcircuit group number (TBD) (see MIL-M-38510, appendix E).

4. PRODUCT ASSURANCE PROVISIONS

4.1 Sampling and inspection. Sampling and inspection procedures shall be in accordance with MIL-M-38510 and methods 5005 and 5007 of MIL-STD-883, except as modified herein.

4.2 Qualification inspection. Qualification inspection shall be in accordance with MIL-M-38510. Inspections to be performed shall be those specified herein for groups A, B, C, and D inspections (see 4.4.1, 4.4.2, and 4.4.3, except omit subgroup 1 of group B for class B and C devices). After qualification of one or more electrically and structurally similar types with a single lead finish, other lead finishes of the same case outline may be qualified by submitting a single type in the qualified case outline to the group B, subgroup 3 test and group D, subgroups 3 and 5 tests.

4.2.1 Combined LTPDs. During non-endpoint electrical parameter testing, LTPDs for subgroups shall be combined as follows:

SUBGROUPS	COMBINED LTPD
1,7,9 or 1&4	5
2,8,10 or 2&5	7
3,8,11 or 3&6	7

4.3 Screening. Screening shall be in accordance with method 5004 of MIL-STD-883 and as modified herein, and shall be conducted on all devices prior to qualification and quality conformance inspection.

a. The alternate screening option of paragraph 3.3 of MIL-STD-883, method 5004 is applicable. Description of the stress test is defined in Figure 31.

b. Constant acceleration shall be 5Kg maximum.

c. Burn-in test (method 1015 of MIL-STD-883).

1. Test condition D using the circuit shown on figure 27, or equivalent.

2. T_A = 125°C minimum.

d. Reverse bias burn-in and interim electrical test in accordance with 3.1.10 of method 5004 of MIL-STD-883 may be omitted.

e. Interim and final electrical test parameters shall be as specified in table II, except interim electrical parameters test prior to burn-in is optional at the discretion of the manufacturer.

f. Percent defective allowable (PDA) - specified as 5% of class A devices and 10% for class B devices based on failures from group A, subgroups 1 and 9 combined test after cooldown as final electrical test in accordance with method 5004 of MIL-STD-883, and with no intervening electrical measurements. If interim electrical parameter tests are performed prior to burn-in, failures resulting from pre burn-in screening may be excluded from the PDA.

If interim electrical parameter tests prior to burn-in are omitted, then all screening failures shall be included in the PDA. The verified failures of group A, subgroups 1 and 9, after burn-in divided by the total number of devices submitted for burn-in in that lot shall be used to determine the percent defective for that lot, and the lot shall be accepted or rejected based on the PDA for the applicable device class.

4.4 Quality conformance inspection. Quality conformance inspection shall be in accordance with MIL-M-38510. Inspections to be performed shall be those specified in method 5005 of MIL-STD-883 and herein for groups A, B, C, and D inspections (see 4.4.1, 4.4.2, and 4.4.3).

4.4.1 Group A inspection. Group A inspection shall be in accordance with table I of method 5005 of MIL-STD-883 and as follows:

a. Tests shall be as specified in table II.

b. Subgroups 4, 5, and 6 may be omitted.

c. LTPDs for subgroups shall be combined as follows:

SUBGROUPS	COMBINED LTPD
1,7,9 or 1&4	7
2,8,10 or 2&5	10
3,8,11 or 3&6	10

4.4.2 Group B inspection. Group B inspection shall be in accordance with table IIB of method 5005 of MIL-STD-883.

4.4.3 Group C and D inspections. Group C and D inspections shall be in accordance with tables III and IV of method 5005 of MIL-STD-883 and as follows:

a. Endpoint electrical parameters shall be as specified in table II.

b. Constant acceleration, subgroup 2 of group C and subgroup 4 of group D shall be condition A (5Kg).

c. Operating life test (method 1005 of MIL-STD-883) conditions:

1. Test condition D using the circuit shown on figure 27 or equivalent.
2. T_A = 125°C minimum.
3. Test duration - 1000 hours, except as permitted by appendix B of MIL-M-38510.
4. Change LTPD to 10.

d. For moisture resistance and salt atmosphere of subgroups 3 and 5 of group D, omit initial conditioning.

4.5 Methods of examination and test. Methods of examination and test shall be as specified in the appropriate table and as follows:

4.5.1 Voltage and current. All voltages given are referenced to the microcircuit ground terminal. Currents given are conventional current and positive when flowing into the referenced terminal.

4.5.2 Life test cooldown procedure. When devices are measured at 25°C following application of the operating life or burn-in test condition, they shall be cooled to 35°C prior to removal of the bias.

4.6 Inspection of preparation for delivery. Inspection of preparation for delivery shall be in accordance with MIL-M-38510, except that the rough-handling test shall not apply.

5. PREPARATION FOR DELIVERY

5.1 Preservation-packaging and packing. Microcircuits shall be prepared for delivery in accordance with MIL-M-38510.

6. NOTES

6.1 Notes. The notes specified in MIL-M-38510 are applicable to this specification.

6.2 Intended use. Microcircuits conforming to this specification are intended for original equipment design application and logistic support of existing equipment.

6.3 Ordering data. The contract or order should specify the following:

a. Complete part number (see 1.2).

b. Requirements for delivery of one copy of the data (see 4.6) pertinent to the device inspection lot to be supplied with each shipment by the device manufacturer, if applicable.

c. Requirement for certificate of compliance, if applicable.

d. Requirements for notification of change of product or process to procuring activity in addition to notification to qualifying activity, if applicable.

e. Requirements for packaging and packing (see 5.1). Appropriate level of MIL-M-55565 must be specified.

f. Requirements for failure analysis (including required test condition of method 5003), corrective action and reporting of results, if applicable.

g. Requirements for product assurance options.

h. Requirements for carriers, special lead lengths or lead forming, if applicable.

6.4 Instructions, terms, and definitions. The abbreviations, terms, definitions, and instructions used are defined in MIL-STD-1313, MIL-STD-1331, and herein.

6.4.1 Architecture. The architecture (block diagram) is shown in figure 1 and the CPU flow chart is shown in figure 2. The memory word is 16 bits long. Each word is also defined as 2 bytes of 8 bits. The instruction set allows both word and byte operands. Thus, all memory locations are on even address boundaries and byte instructions can address either the even or odd byte. The memory space is 65,536 bytes or 32,768 words. The word and byte formats are shown in figure 3.

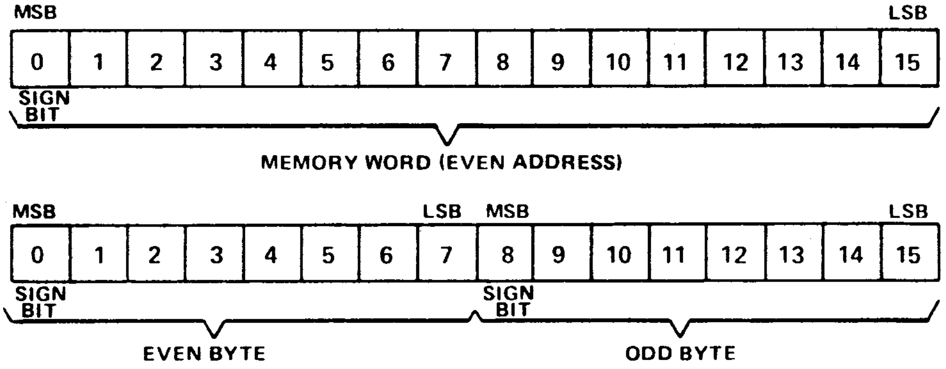


FIGURE 3 – WORD AND BYTE FORMATS

6.4.1.1 Registers and memory.

The device employs an advanced memory-to-memory architecture. Blocks of memory designated as workspace replace internal hardware registers with program-data registers. The device memory map is shown on figure 4. The first 32 words are used for interrupt trap vectors. The next contiguous block of 32 memory words is used by the extended operation (XOP) instruction for trap vectors. The last two memory words, $FFFC_{16}$ and $FFFE_{16}$, are used for the trap vector of the LOAD signal. The remaining memory is then available

NOTE

The data contained in this specification from Paragraph 6.4 through Figure 27A has been intentionally deleted from this Appendix. This data can be found both in the body of this manual and in Appendix A.

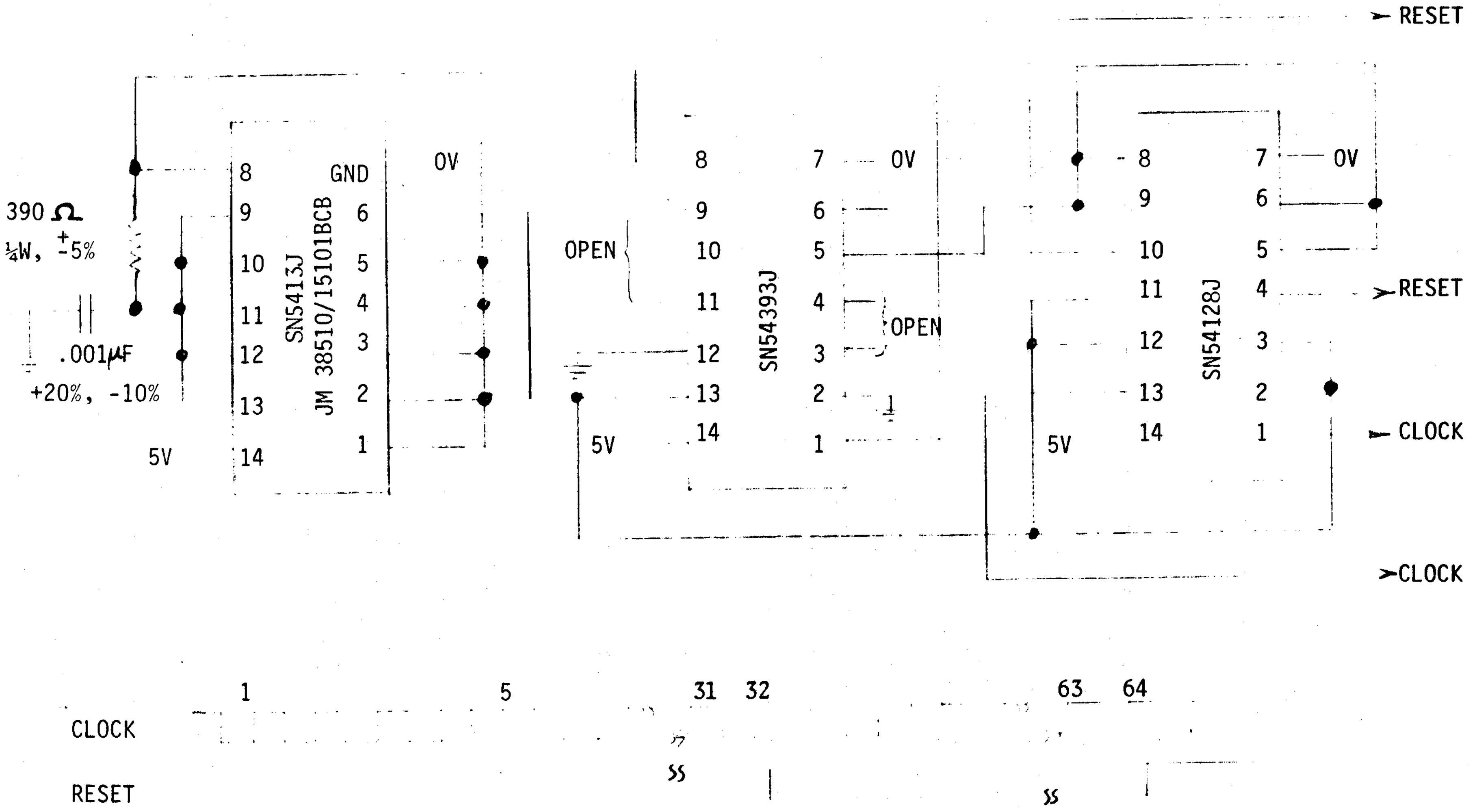
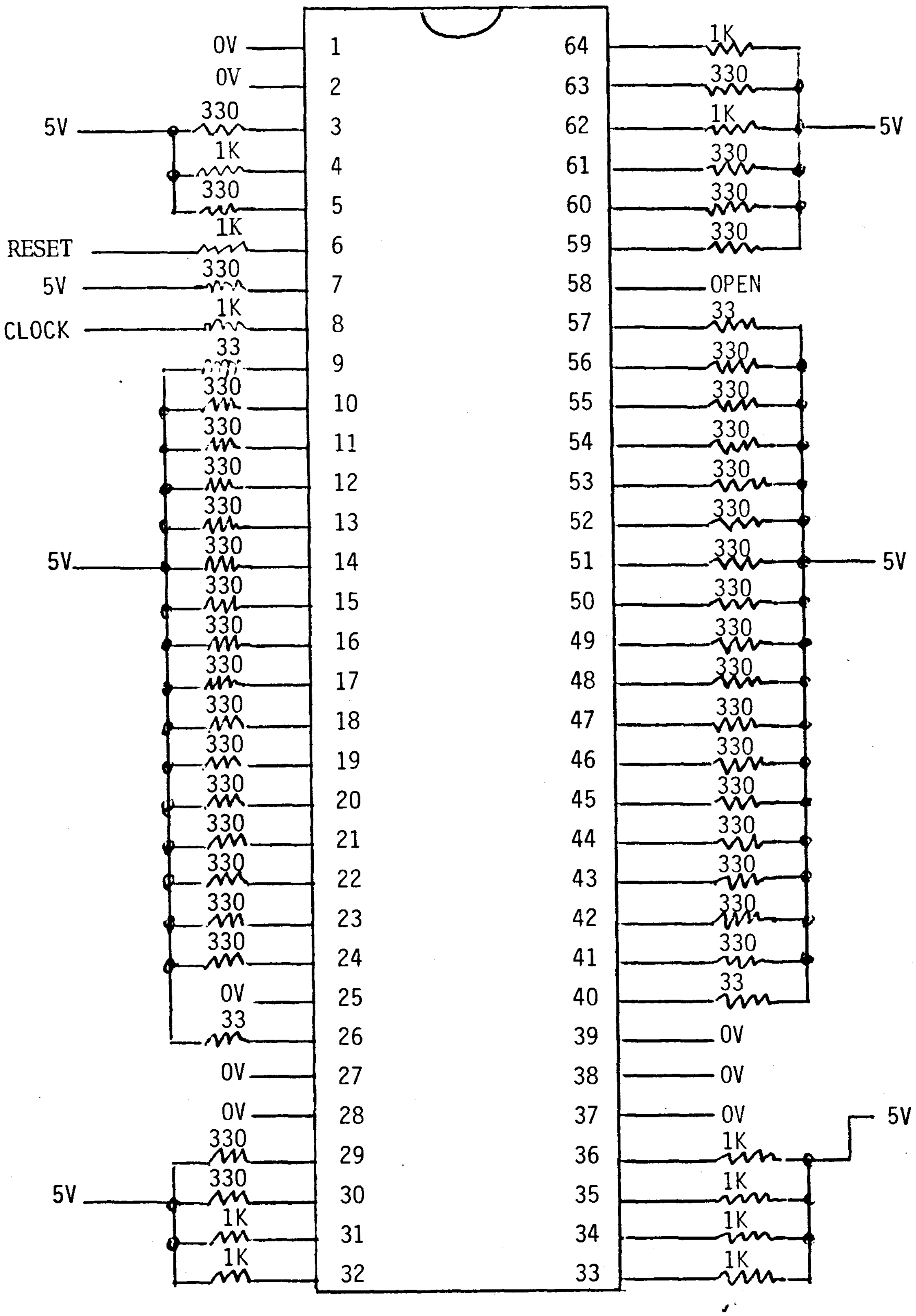


FIGURE 27A
BURN-IN AND OPERATING LIFE TEST CIRCUIT

FIGURE 27B
BURN-IN AND OPERATING LIFE TEST CIRCUIT



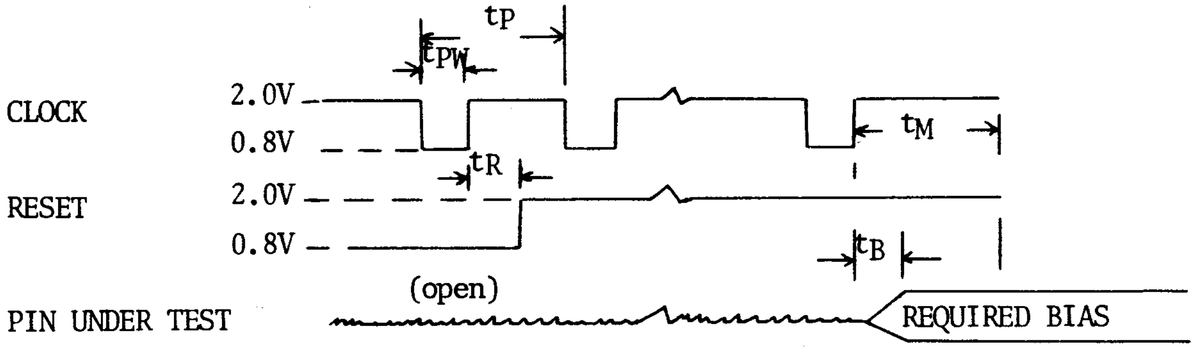
NOTES:

1. All resistor values are in ohms
2. Pins 9, 26, 40 and 57 are injector sources
3. The four 33 ohm resistors are metal film 1 watt ±1% tolerance, and all other resistors are ≥ 1/4 watt ±5%

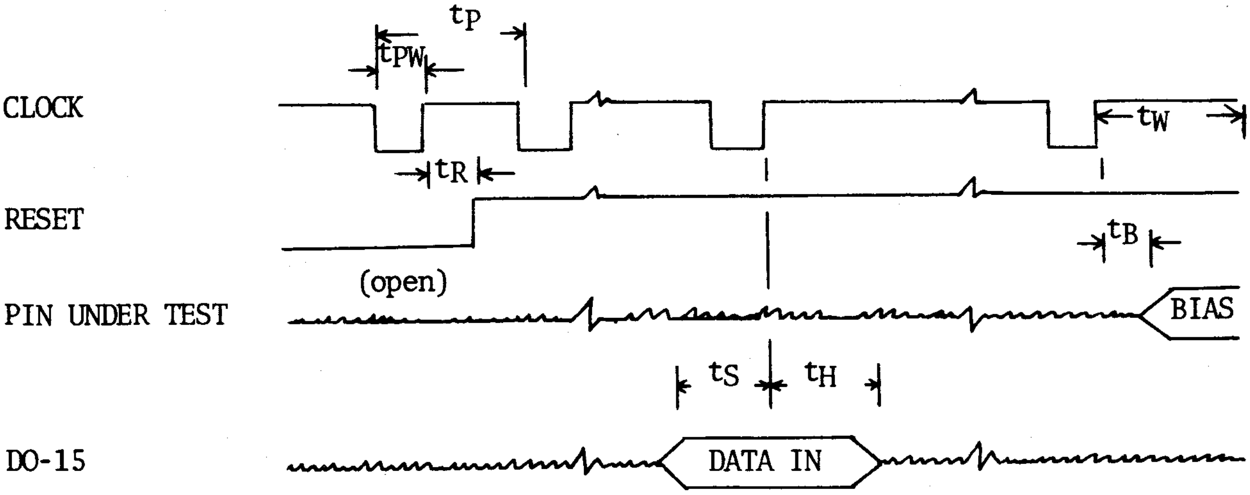
FIGURE 28

DC PARAMETRIC WAVEFORMS

AXX   XX = number of clock pulses



BXX   XX = number of clock pulses



DO -15 should have the following data in:

CLOCK PULSE	DATA IN (DO=MSD, D15=LSD)	DATA IN (1=2.0V, 0=0.8V)
9	FEE8	
25	FEA0	
29	3043	
33	6100	
43	FC20	

FIGURE 28 (cont.)

DC PARAMETRIC WAVEFORMS

DEFINITION OF t TERMS

t_{PW} = Clock Pulse Width = 333ns

t_P = Clock Period = 1μs

t_R = Reset Hold Time = 500ns

t_S = Data Setup Time = 500ns

t_H = Data Hold Time = 500ns

t_B = Bias Applied Time = 1ms

t_M = Measurement Time >500μs

```
   (Inputs)    (Device Pin #)

   DEFI  F1     (1=4, 2=6,

   (Outputs)   (Device Pin #)

   DEFO  F2     (1=3, 2=5,


  (Masked Outputs)  (Device Pin #)

  DEFM  F2M     (1=3, 2=5,


 (Tristate Outputs   (Device Pin #)

  DEFT  F3      (1=41, 2=42,

(Masked Tristate
      Outputs)   (Device Pin #)

  DEFM  F3M      (1=41, 2=42,
(Clock Inputs)   (Device Pin #)
  DEFC  F4       (1=8)
                                                                   Line
| A | |B|    C      D        E        F       G   |H  I              Number
PTT     X' 98C, 000000, FFFFFF, 0000, FFFF, 8; 0                  00000014

RPT   5X'  DOC, 000000, FFFFFF, 0000, FFFF, 8; 0                  00000014

PTT     X' CFC, 0A9022, 000000, 000F, 0000, 8; 8                  00000016
```

A and B   PTT = Pattern   RPT = Repeat Pattern "B" Number of Times.

```
 C     Input Conditions  98C = 1001 1000 1100
                         DOC = 1101 0000 1100
                         CFC = 1100 1111 1100

     Input 1, Device Pin 4             Input 10, Device Pin 64
```

FIGURE 29. DC Functional Patterns Definition and Matrix Key

D OUTPUT CONDITIONS 000000 = 0000 0000 0000 0000 0000 0000

0A9022 = 0000 1010 1001 0000 0010 0010

Output 1, Device Pin 3 Output 24, Device Pin 63

E OUTPUT MASKING FFFFFF = 1111 1111 1111 1111 1111 1111

000000 = 0000 0000 0000 0000 0000 0000

1 = Masked
0 = Unmasked

Output 1, Device Pin 3 Output 24, Device Pin 63

F TRISTATE OUTPUT CONDITIONS 0000 = 0000 0000 0000 0000

000F = 0000 0000 0000 1111

Output 1, Device Pin 41 Output 16, Device Pin 56

G TRISTATE MASKING FFFF = 1111 1111 1111 1111

0000 = 0000 0000 0000 0000

1 = Masked
0 = Unmasked

Output 1, Device Pin 41 Output 16, Device Pin 56

H CLOCK CONDITION 8 = 1000

1 = Clock
0 = No Clock

1st DEFC Group
(In 9900 there is only one DEFC Group)

I TRISTATE DRIVE CONDITIONS 8 = 1000
0 = 0000

1 = Drive
0 = No Drive

1st DEFT Group
(In 9900 there is only one DEFT Group)

Injector Current - 500mA

Input Voltage Levels

0 = 0.2 volts

1 = 3.0 volts

Output Voltage Levels

0 = 0.4 volts

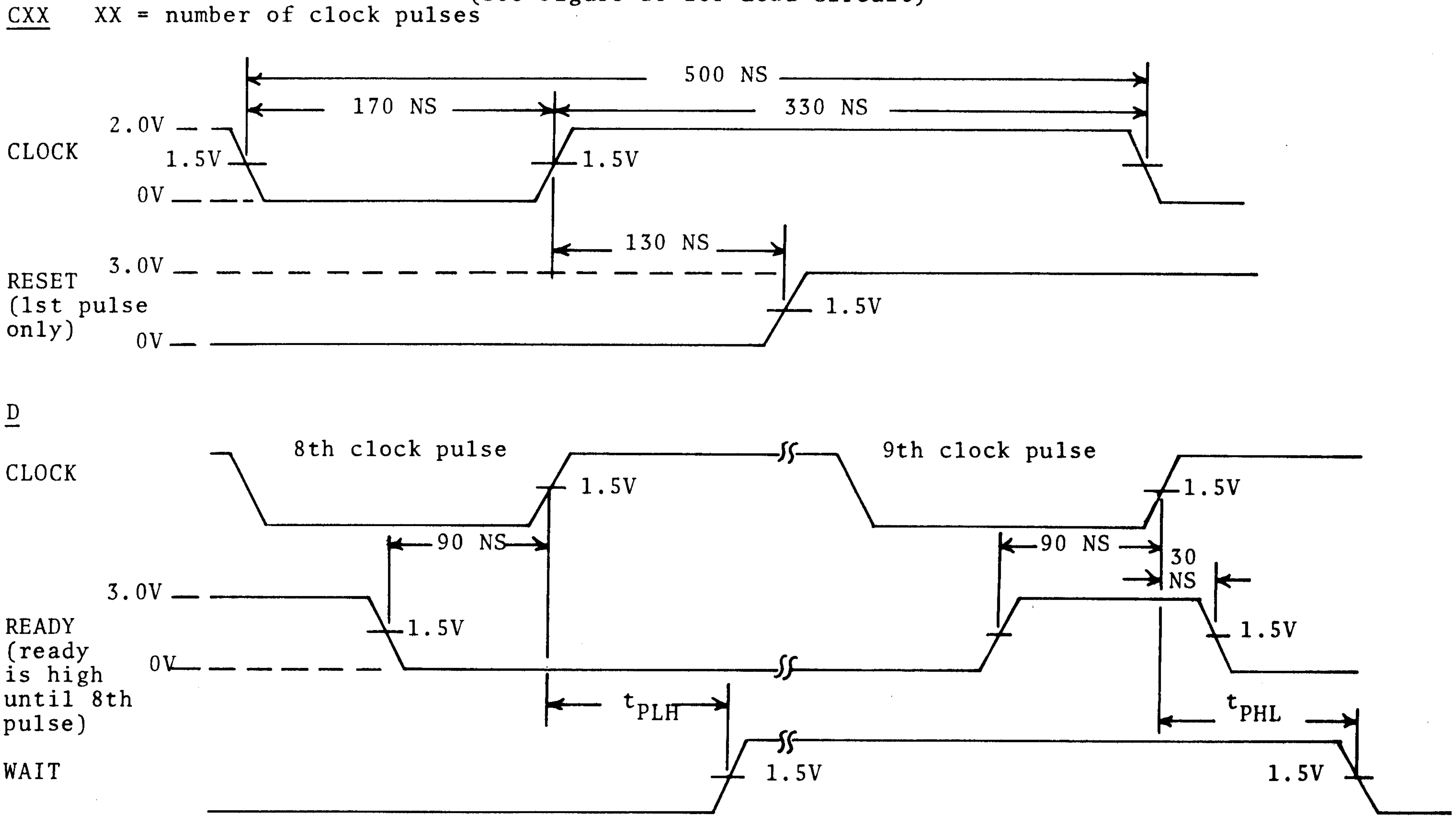
1 = 2.0 volts

FIGURE 29. (cont.)

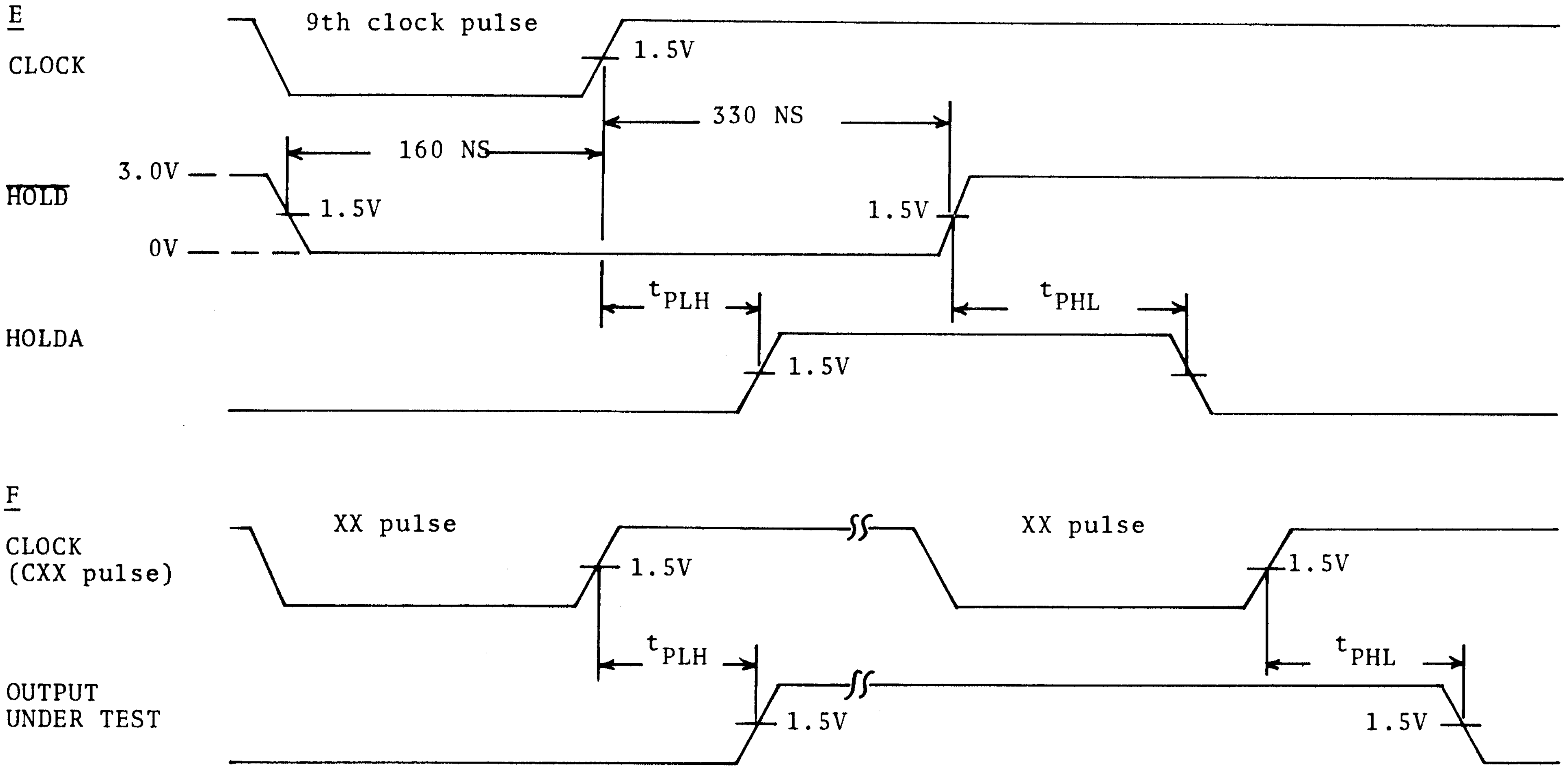
Figure 30 Page 1

AC PARAMETRIC WAVEFORMS

(See Figure 25 for Load Circuit)

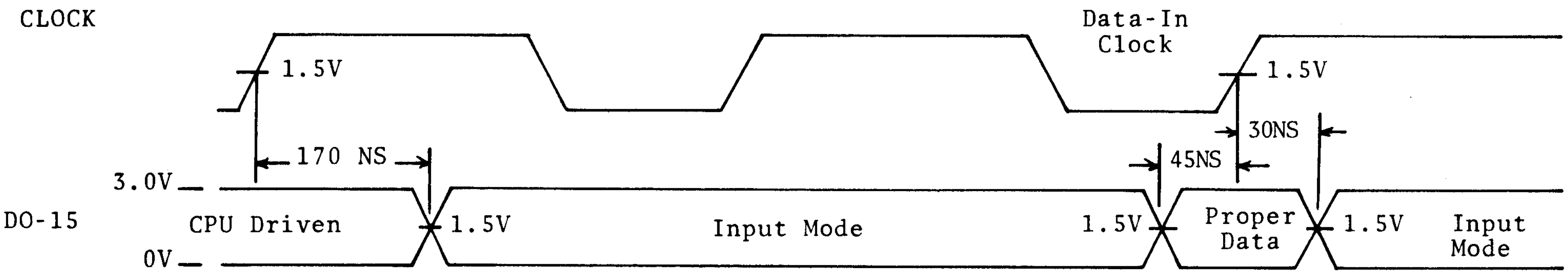


AC PARAMETRIC WAVEFORMS



AC PARAMETRIC WAVEFORMS

G (DO-D15 should not be driven except during data-in clock pulse period)



DATA-IN CLOCK PULSE	MEASUREMENT		PROPER DATA (DO=MSD, D15=LSD)
9	t_{PLH}	A0-14	FFE0
29	t_{PLH}	CRUOUT	1E7F
41	t_{PLH}	CRUOUT	1DFB
29	t_{PLH}	CRUCLK	3043
9	t_{PHL}	A0-14	0000
25	t_{PHL}	A0-14	FFFE
29	t_{PHL}	A0-14	A040
29	t_{PHL}	CRUOUT	1DFB
41	t_{PHL}	CRUOUT	1E7F
29	t_{PHL}	D0-15	A040
29	t_{PHL}	CRUCLK	3043

AC PARAMETRIC WAVEFORMS

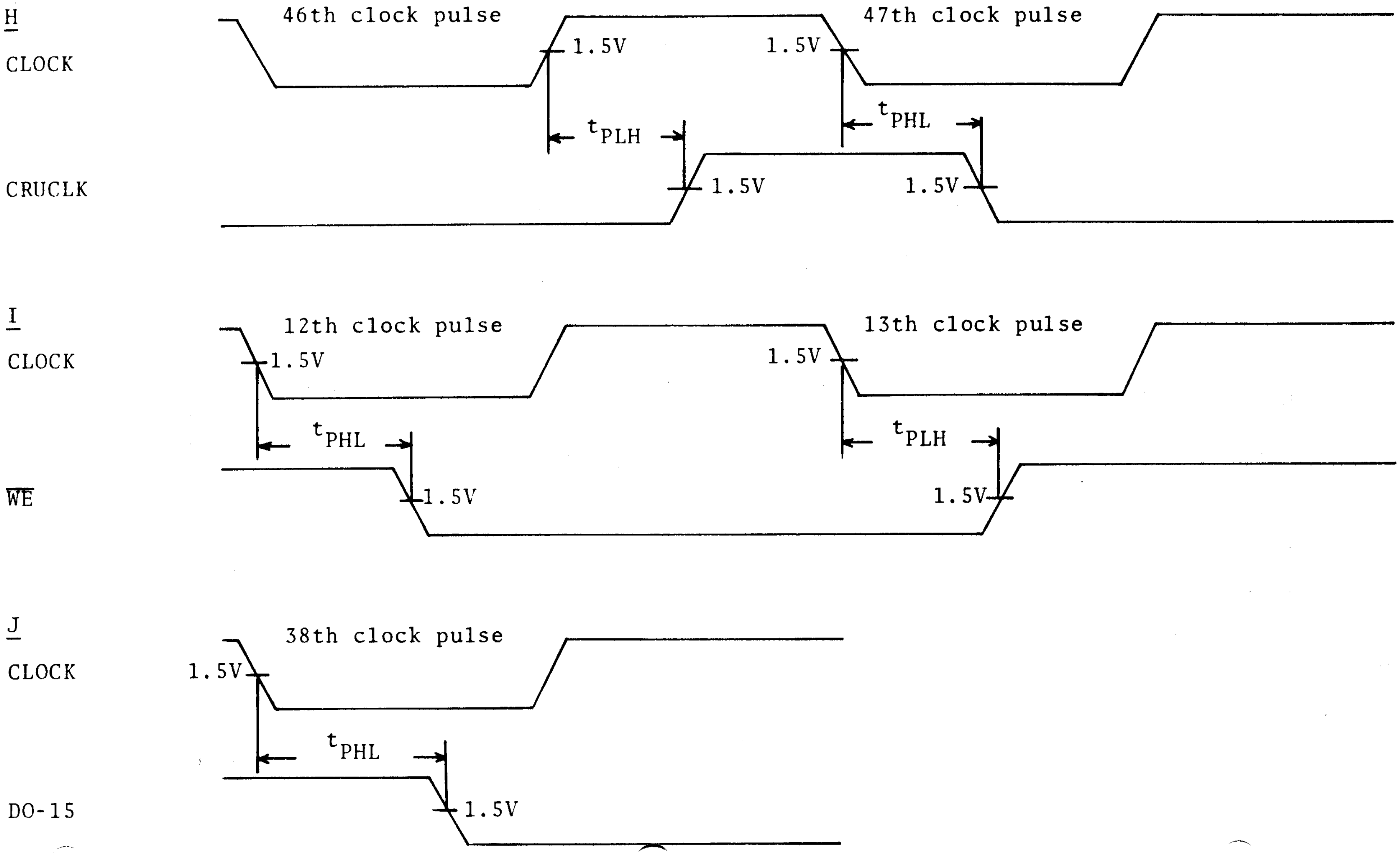


Figure 31

STRESS TESTS

PIN REQUIREMENTS

I_{INJ}	R_L	V_{IN}	V_L(DBUS)MIN	V_L(OUTPUTS)MIN
1.0 amp(1)	280Ω	3.6V(2)	5.86V(1)	5.72V(1)

APPLIED CONDITIONS

I_{INJ}	R_L	V_{IN}	V_L (DBUS)	V_L (OUTPUTS)
1.0 amp(1)	280Ω	3.6V(2)	6.0V(2)	6.0V(2)

Test patterns are applied to maximize number of outputs in Logic 1 state. Conditions held for a minimum of 20 MS. Test patterns are then applied to stress remaining outputs. Full functional and parametric tests are then performed.

(1) Values required for TI proposed 200% Current Stress.

(2) Values required to meet MIL-STD-883B 120% Voltage Stress.

TABLE V. FUNCTIONAL PATTERNS

PATTERN GROUP	NO. OF TESTS
T1B1	920
T1B2	936
T1B3	1004
T1B4	946
T2B1	624
T2B2	622
T2B3	141
T2B4	122
T2B5	356

APPENDIX D

PRELIMINARY RADIATION TEST RESULTS

NOTE

> Data presented in Sections D.1 through D.2.3 is representative of the original SBP 9900 advanced I^{2}L process. Data presented in Sections D.3 through Figure D-14 is representative of the SBP 9900A advanced oxide-separated I^{2}L process.

D.1 ADVANCED I^{2}L (SECOND GENERATION) GATES

D.1.1 TEST STRUCTURES. This section presents both total dose and neutron radiation effects on basic advanced I^{2}L cells typical of those used in the original SBP 9900. The units tested were from an early development lot. Test cells came from Texas Instruments X942 I^{2}L test chip. Figure D-1 shows the geometry tested. The multiple inverter ring counter used to measure average propagation delay is shown in Figure D-2. The geometrics used had four collectors (only one connected) and used the collector nearest the base contact. The circuit schematic for the multiple inverter is shown in Figure D-3.

D.1.2 GAMMA DOSE DEGRADATION (Co^{60}). The common-base current gain of the lateral PNP, under gamma dose degradation, is presented in Figure D-4. The degradation was very minimal even at 10^6 rad(Si). The effect of gamma dose on the inverse NPN was also very minimal and the data is presented in Figure D-5.

The propagation delay versus power dissipation would be predicted to be very nominal based on the above data. The data in Figure D-6 indicates only a very slight degradation even at 10^6 rad(Si). The gates operated over the whole range of injector current tested.

D.1.3 NEUTRON DEGRADATION. Data similar to the gamma test was taken over a wide range of currents. The neutron damage on the common-base current gain is shown in Figure D-7. The data, although not as impressive as the gamma dose results, is nevertheless greatly improved over the first generation I^{2}L. Figure D-8 shows the current gain of the NPN as a function of collector current and neutron level. The current gain drops below one at a collector current of 10 μA and 10^{14} n/cm^2. At 100 μA collector current, the current gain is above 2 even at 10^{14} n/cm^2.

The average propagation delay characteristics are shown in Figure D-9. The injector current level for the threshold of oscillation (I_{TH}) is indicated for each neutron level. The unit did not operate at 3 × 10^{14} n/cm^2.

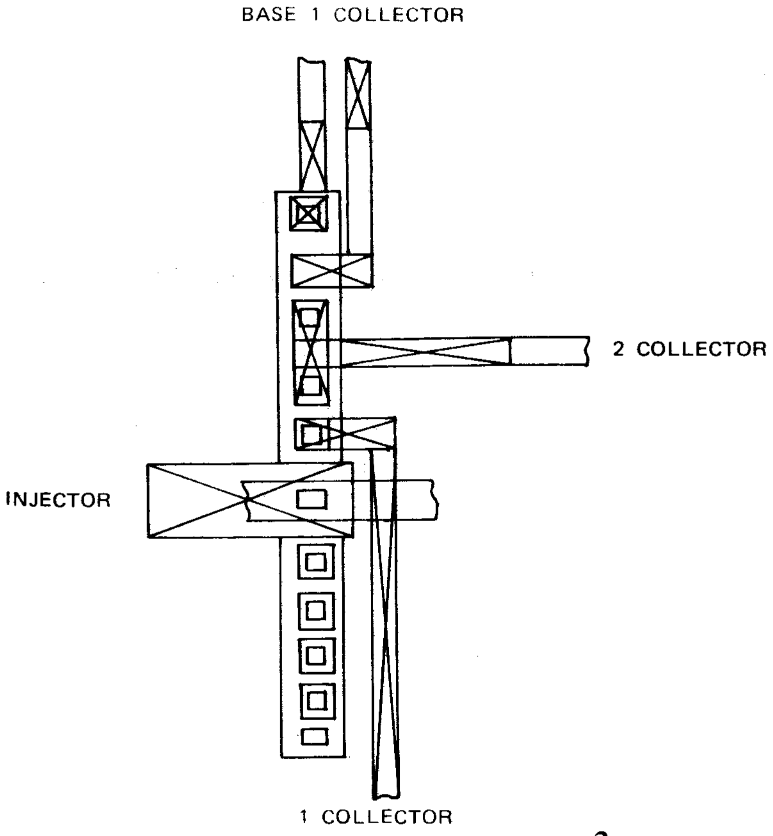


Figure D-1. Test Cell for Advanced I^2L

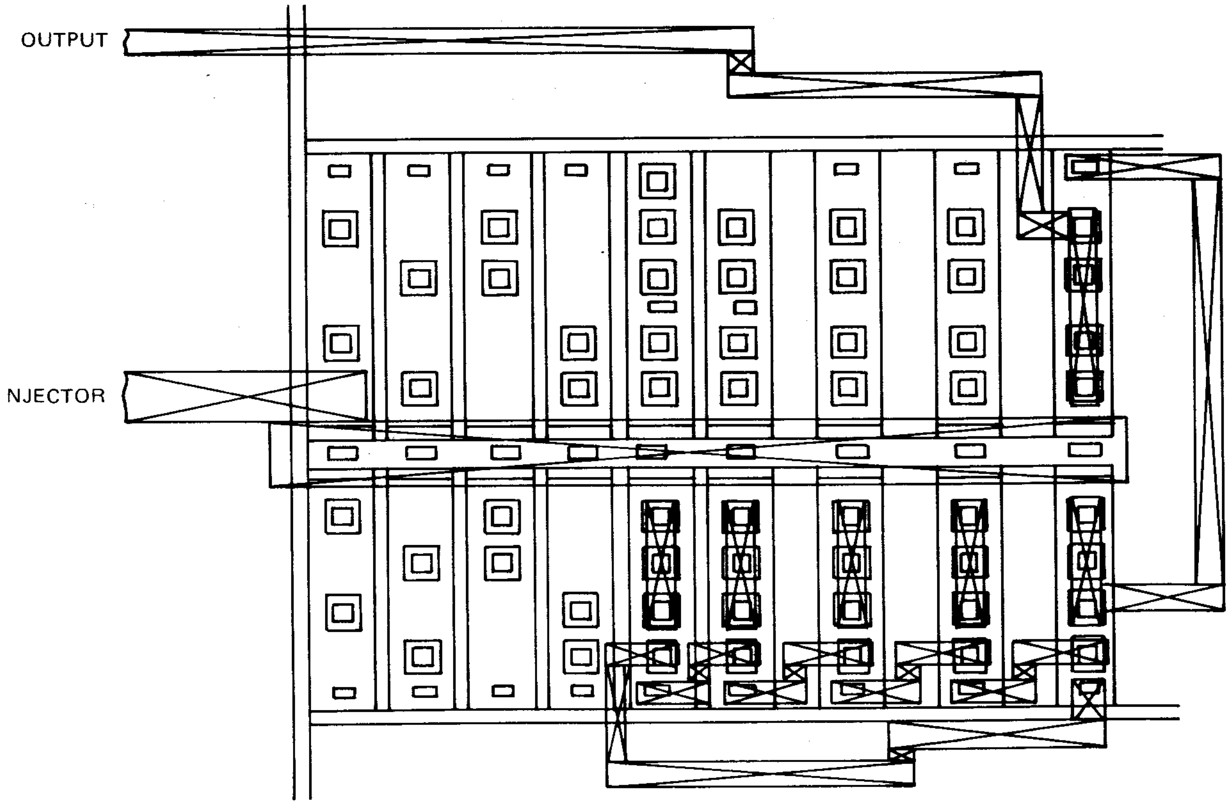


Figure D-2. Inverter Ring Counter

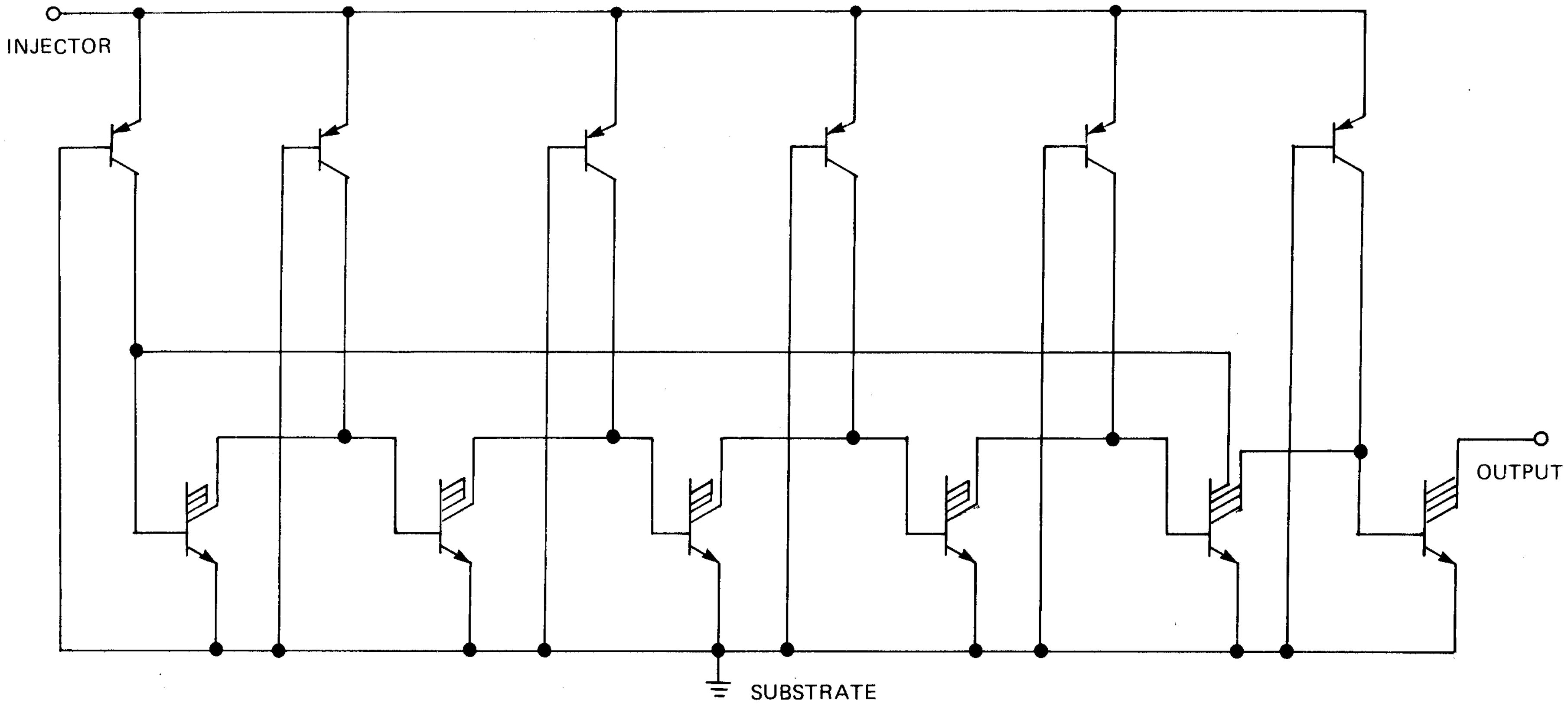


Figure D-3. Circuit Schematic for Multiple Inverter

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

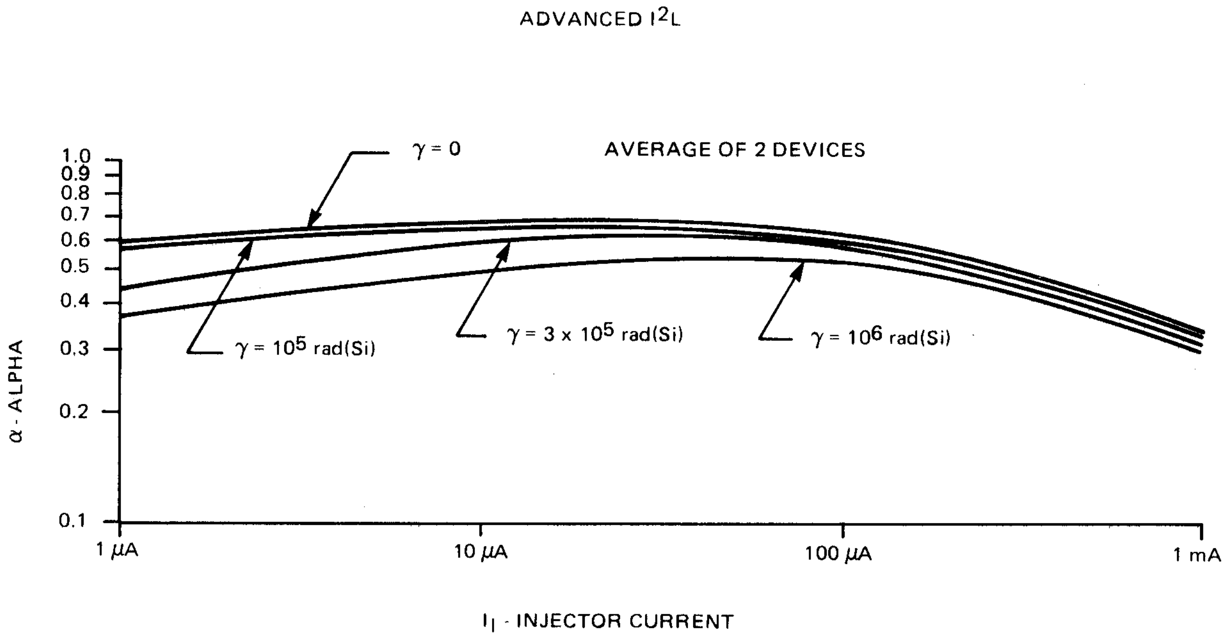


Figure D-4. Common-Base Current Gain

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

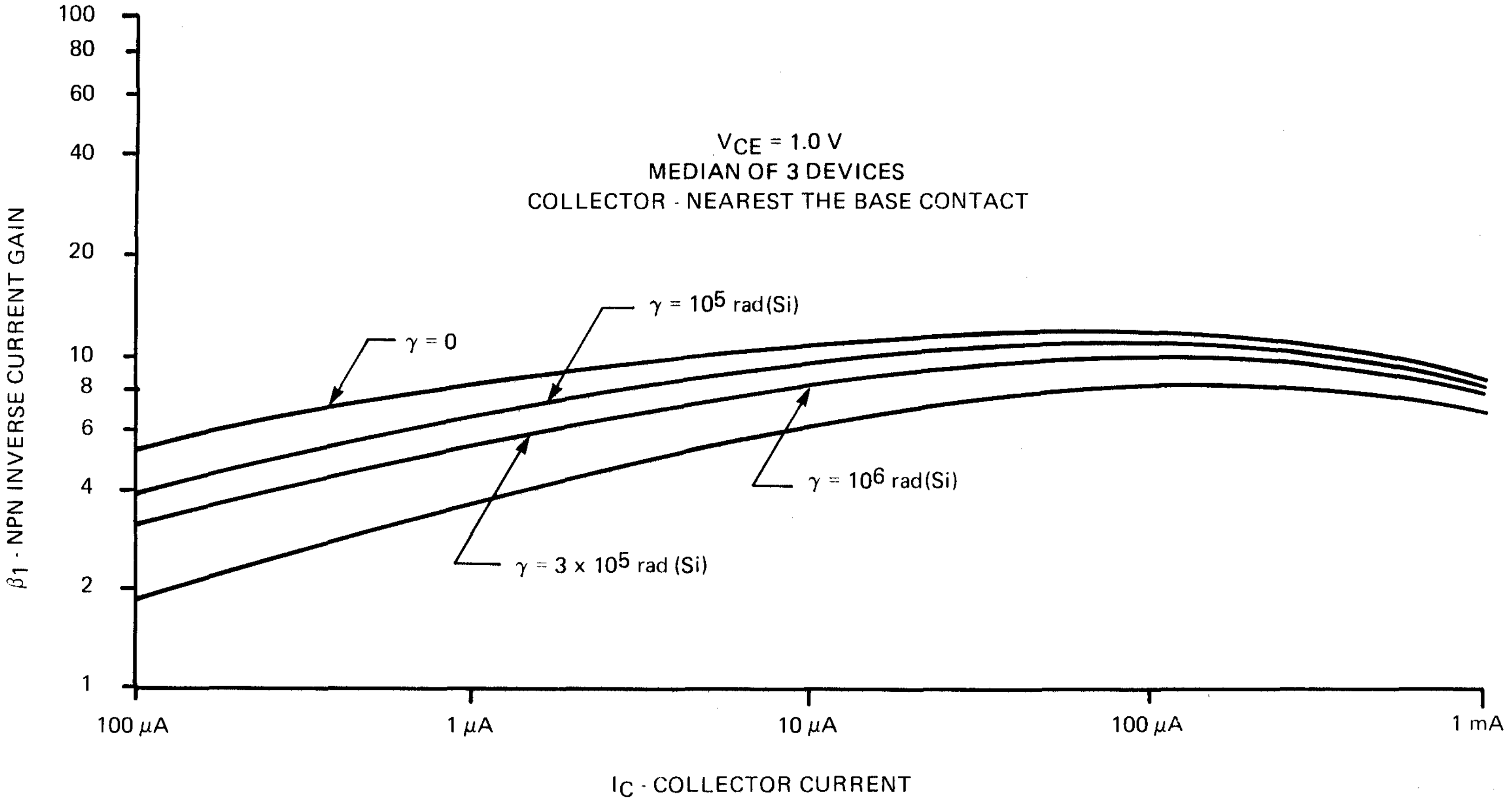


Figure D-5. Effect of Gamma Dose on the Inverse NPN

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

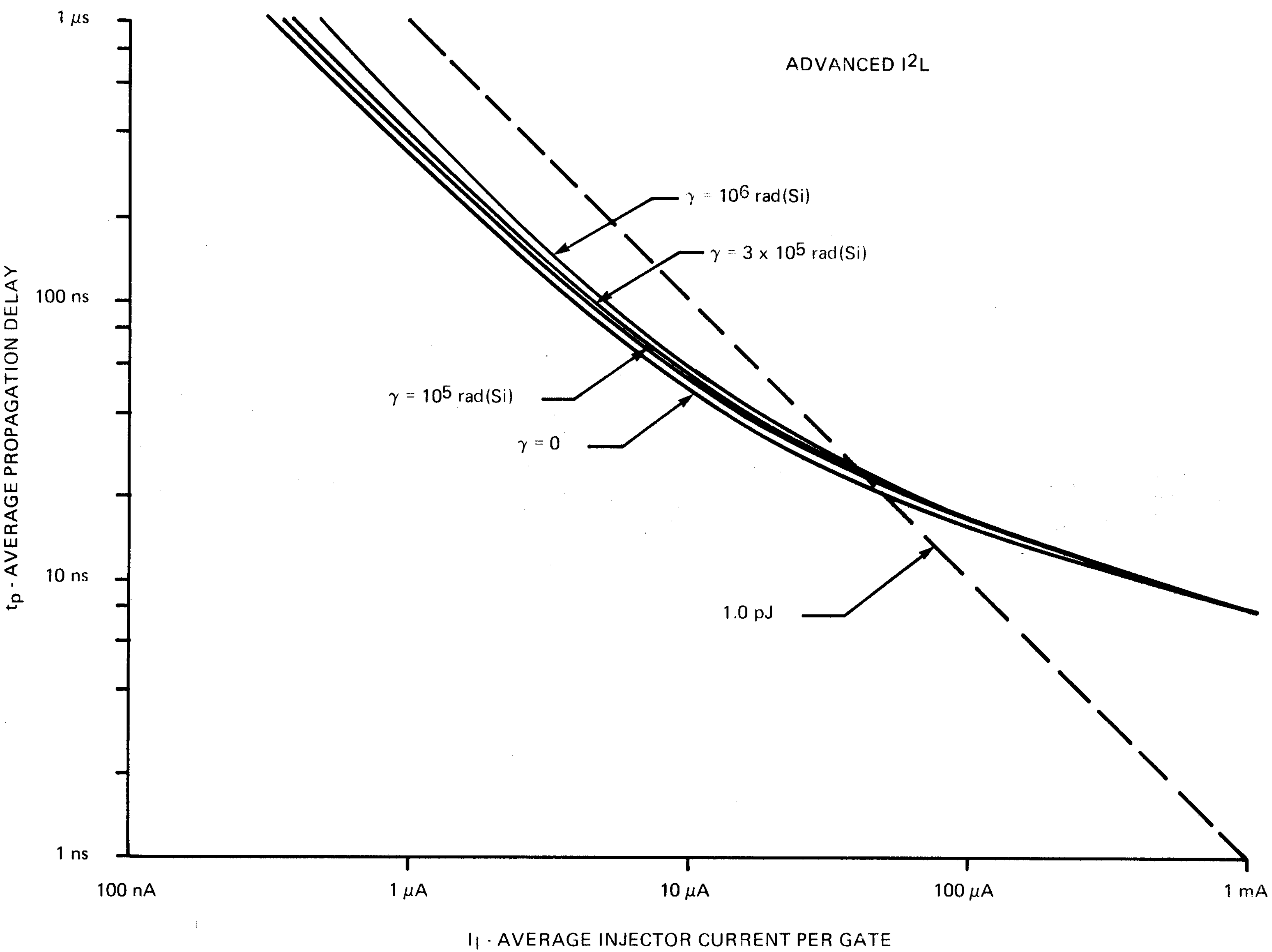


Figure D-6. Propagation Delay vs Power Dissipation

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

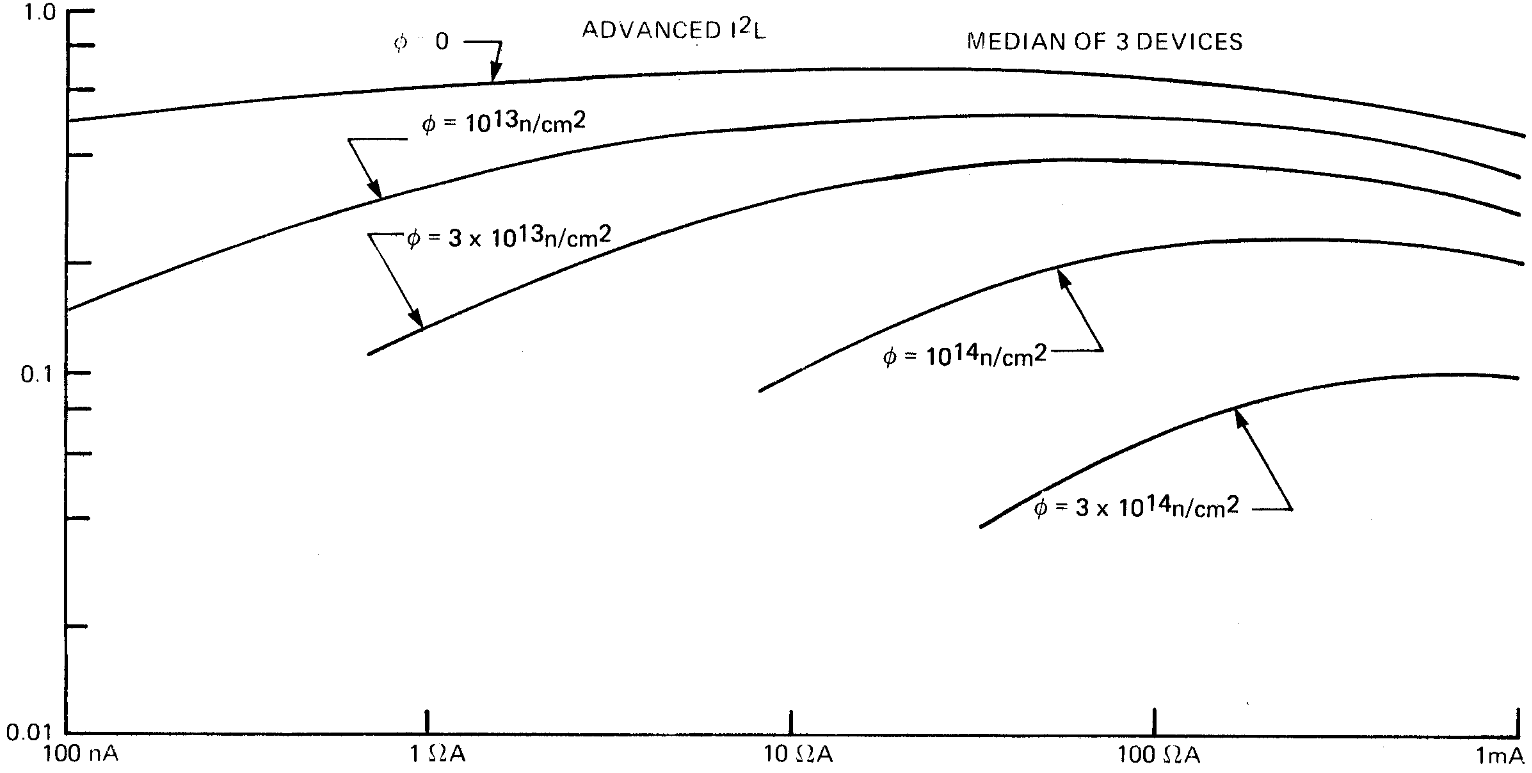


Figure D-7. Neutron Damage on the Common-Base Current Gain

DATA SOURCE: Naval Weapons Support Center – Crane, Indiana

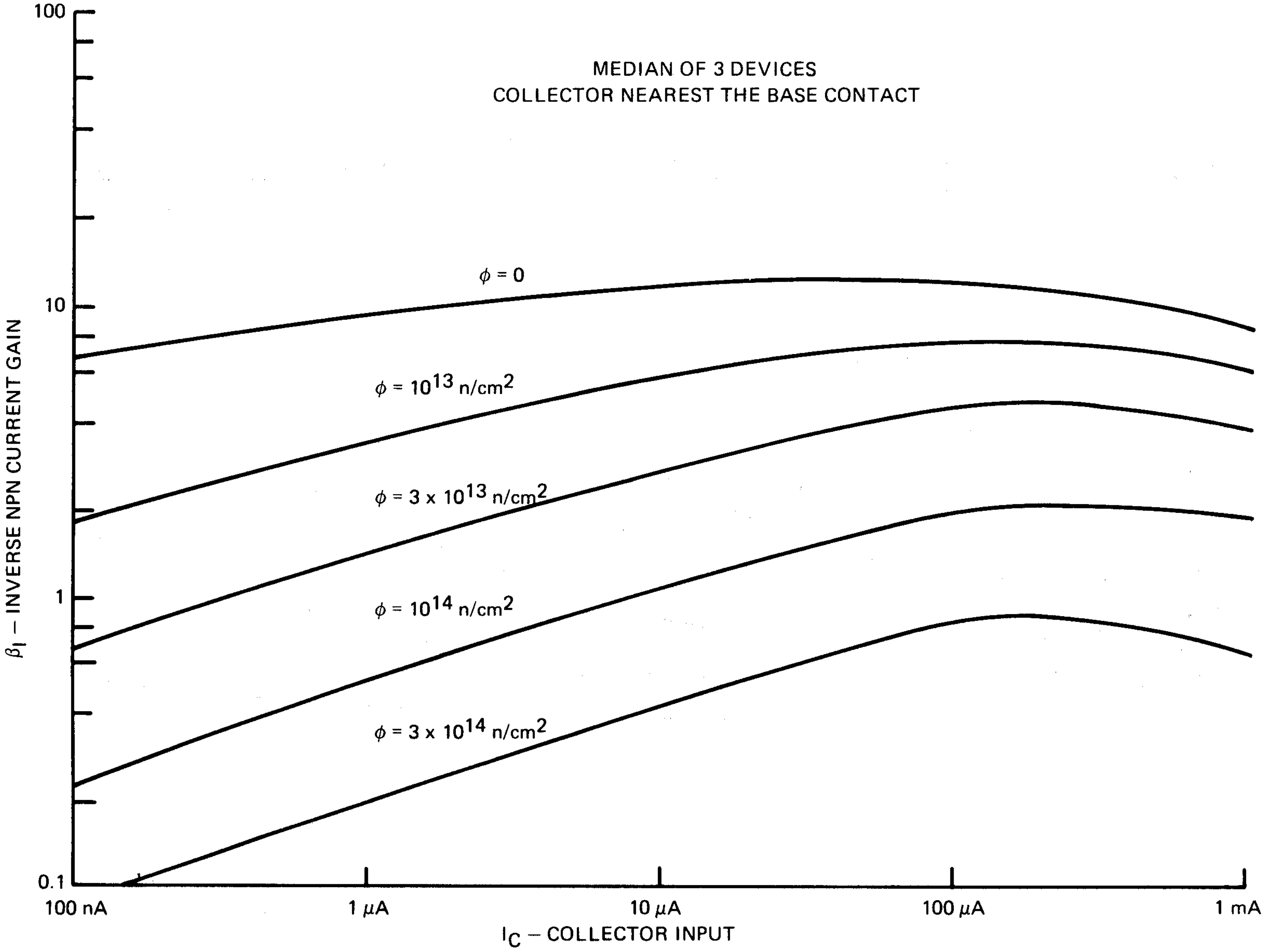


Figure D-8. Current Gain of the NPN

DATA SOURCE: Naval Weapons Support Center – Crane, Indiana

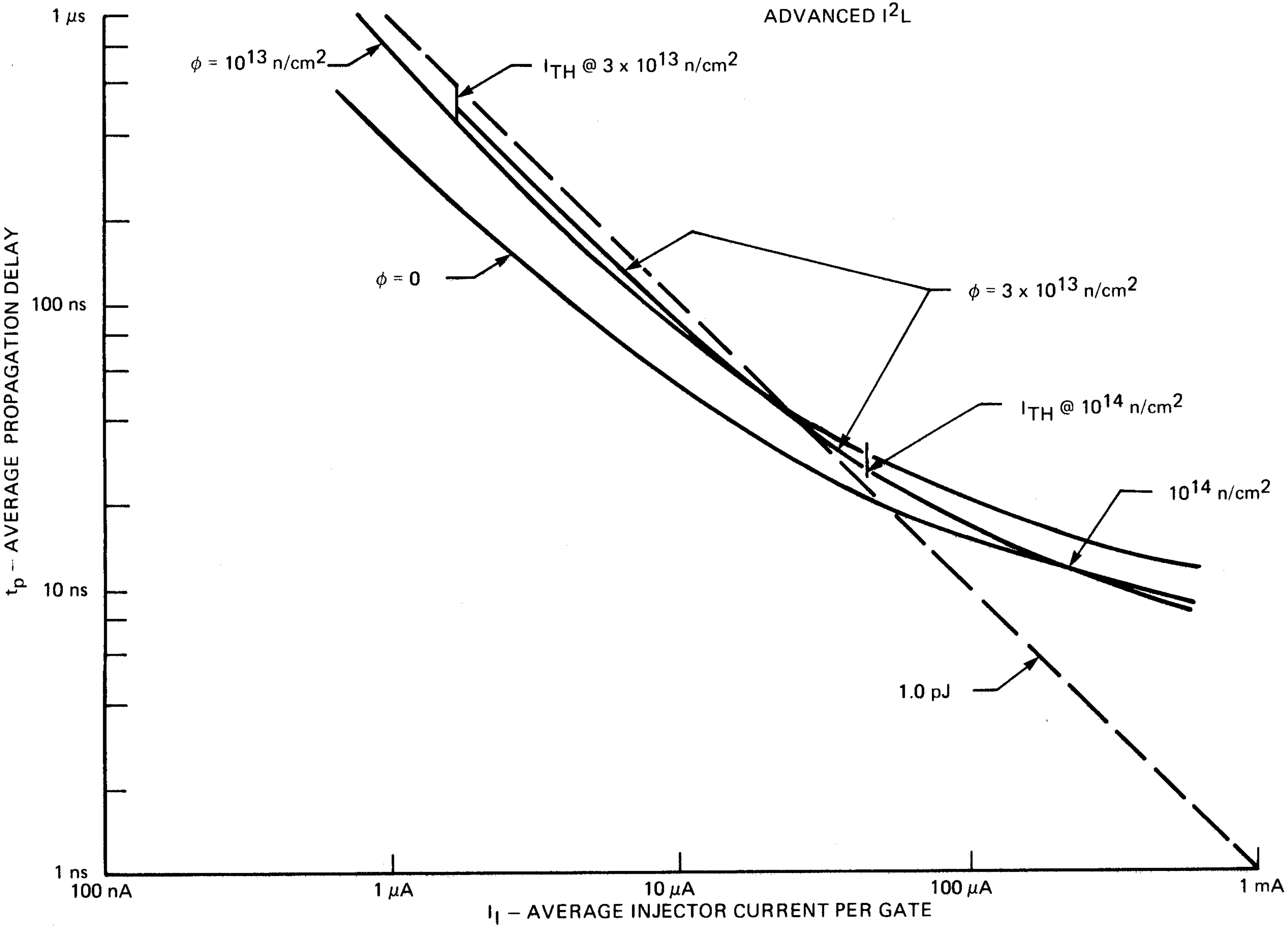


Figure D-9. Average Propagation Delay Characteristics

D.2 SBP 9900 TESTING

D.2.1 SBP 9900 TOTAL DOSE TOLERANCE

Units:	#1002, #1010, and #1011
Facility:	JPL Dynamitron 5/4/77
	2.5 MeV E-Beam
Outputs:	Set High, 1KΩ Pullup to 5V
Inputs:	#1002 Pulled High, 20KΩ to 5V
	#1010 Grounded
	#1011 Grounded

Total Injector Current - 10 mA during Irradiation

TOTAL DOSE	f_{MAX} AT I_C = 90 mA	f_{MAX} AT I_C = 520 mA
0	950 KHz	2.6 MHz
10^5	900	2.6
3×10^5	850	2.55
10^6	750	2.4

#1002 Fully Recovered after 35 Minutes - $f_{MAX} > 2$ MHz
#1010 Recovered after 48 Hours except for Decrement Instruction - $f_{MAX} > 2$ MHz
#1011 No Recovery Data Available

D.2.2 SBP 9900 UPSET LEVEL TOLERANCE

Units:	#1195 and #1197
Facility:	White Sands Linac 6/6/77
	20 MeV Electron Mode, PW = 40 nS, 100nS, 1μS

Dynamic Test at I_{INJ} = 50 mA and 500 mA/2 MHz

PULSE WIDTH	Y_{TH} AT 2 MHz #1197	I_{INJ} = 500 mA #1195
40 ns	4.4×10^8	7×10^8
100 ns	2.0×10^8	2.5×10^8
1000 ns	7.5×10^7	1.1×10^8

100 nS Pulse Width Tests at I_{INJ} = 50 mA had the Same Failure Threshold as Shown Above.

D.2.3 SBP 9900 NEUTRON TOLERANCE

Units:	#1189 and #1190
Facility:	Sandia Pulsed Reactor III 6/9/77 Units Passively Irradiated

Measured Parameters:

1) Max Freq. at Minimum Injector Current
2) Max Freq. at 100 mA, 200 mA and 500 mA Injector Current

UNIT #	n/cm^2 (1 MeV eq.)	IMIN/FMAX	100 mA FMAX	200 mA FMAX	300 mA FMAX	500 mA FMAX
1189	0	25/245K	890 KHz	2.1 MHz		2.82 MHz
1190	0	33/320K	875 KHz	1.6 MHz		2.7 MHz
1189	3.61×10^{12}	28/260K	845 K	1.53 M	2.76 M	
1190	3.61×10^{12}	35/335K	877 K	1.50 M		2.55 M
1189	1.66×10^{13}	58/470K	755K	1.38 M		2.62 M
1190	1.66×10^{13}	67/545K	780 K	1.4 M		2.70 M
1189	5.28×10^{13}	150/0.9M			1.55 MHz	2.20 M
1190		170/1.05M			1.65 MHz	2.25 M
1189	8.4×10^{13}	320/1.2M				1.82 MHz

D.3 ADVANCED OXIDE-SEPARATED I^2L (SECOND GENERATION) GATES

D.3.1 TEST STRUCTURES. This section presents both total dose and neutron radiation effects on basic advanced *oxide-separated* I^2L cells typical of those used in the SBP 9900A. the multiple inverter ring counter used to measure average propagation delay is shown in Figure D-10. The geometries used had five collectors (only one connected) and used the collector *farthest* from the base contact.

D.3.2 GAMMA DOSE DEGRADATION. The common base current gain of the lateral PNP, under gamma dose degradation, is presented in Figure D-11. The degradation was minimal even at 10^6 rads (Si). The effect of gamma dose on the inverse NPN was as shown in Figure D-12. Propagation delay versus power dissipation is as shown in Figure D-13.

D.3.3 NEUTRON DEGRADATION. Figure D-14 shows neutron degradation on the current gain of the NPN.

D.4 SPB 9900A TRANSIENT UPSET TESTING.

Laser sourced radiation of 40 nsec pulse width was applied to both the original SBP 9900 and the SBP 9900A. The SBP 9900A exhibited a $3.5 \cdot 10^9$ rads (Si)/sec tolerance versus $2.7 \cdot 10^9$ rads (Si)/sec. exhibited by the original SBP 9900.

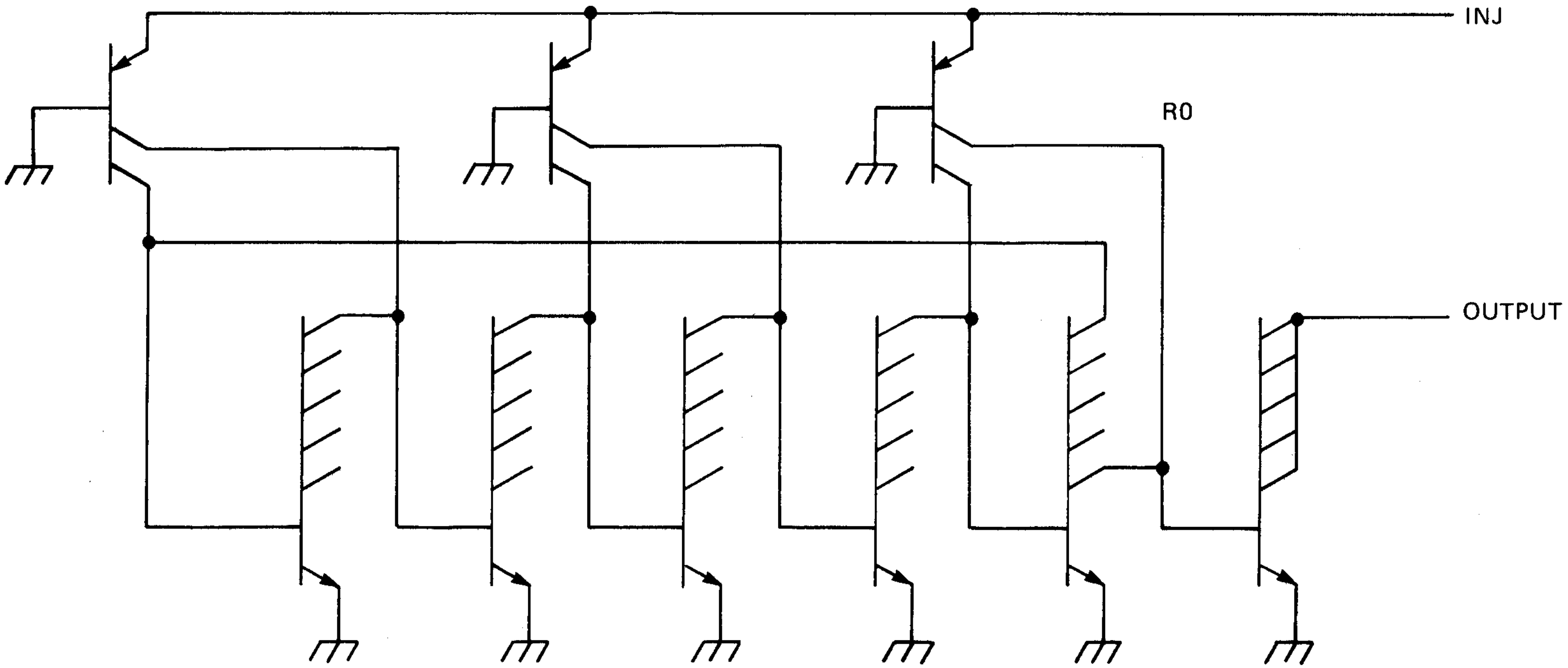


Figure D-10. Multiple Inverter Ring Counter

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

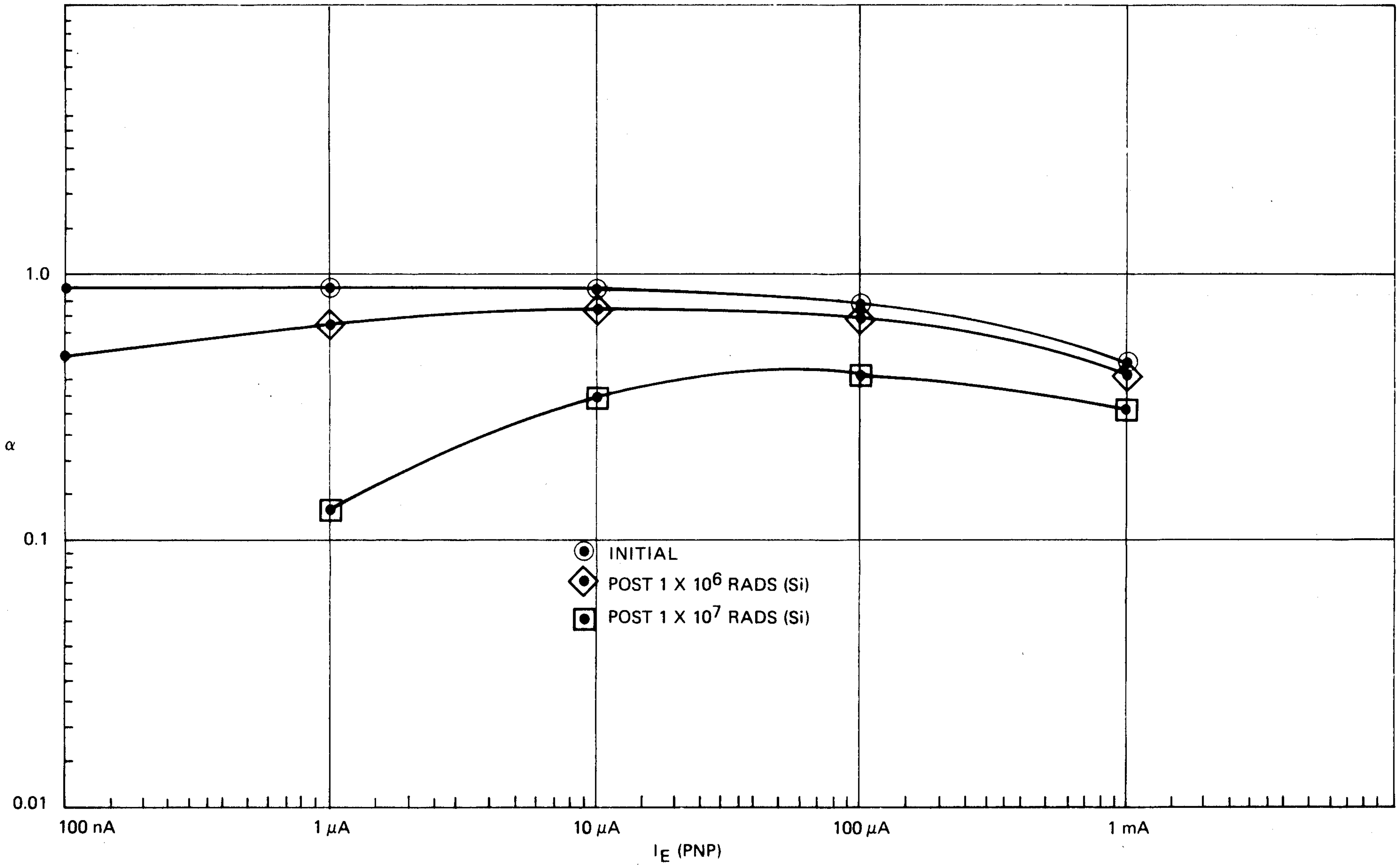


Figure D-11. α (PNP) vs I_E

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

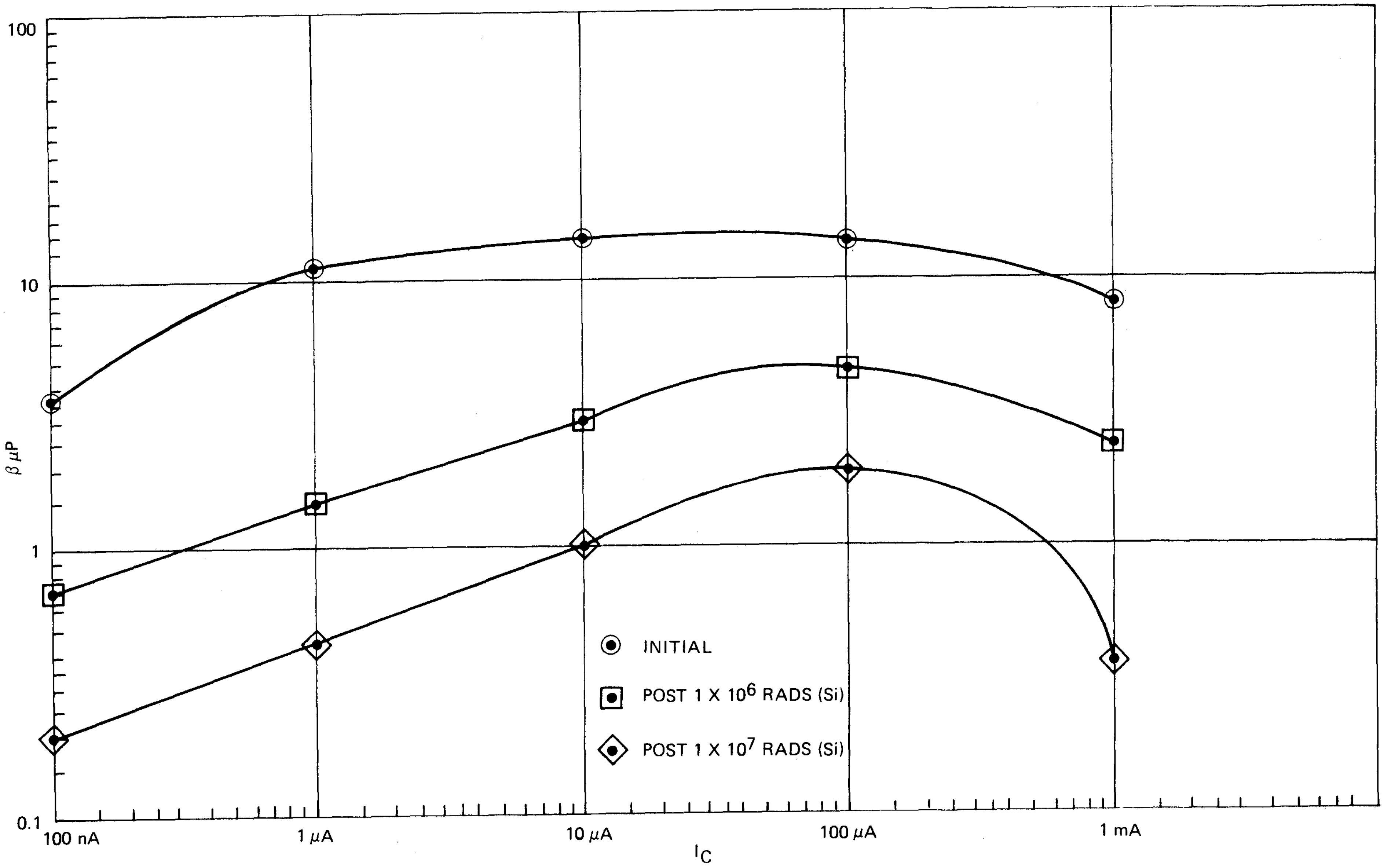


Figure D-12. $\beta\ \mu P$ vs I_C (For Collector)

DATA SOURCE: Naval Weapons Support Center — Crane, Indiana

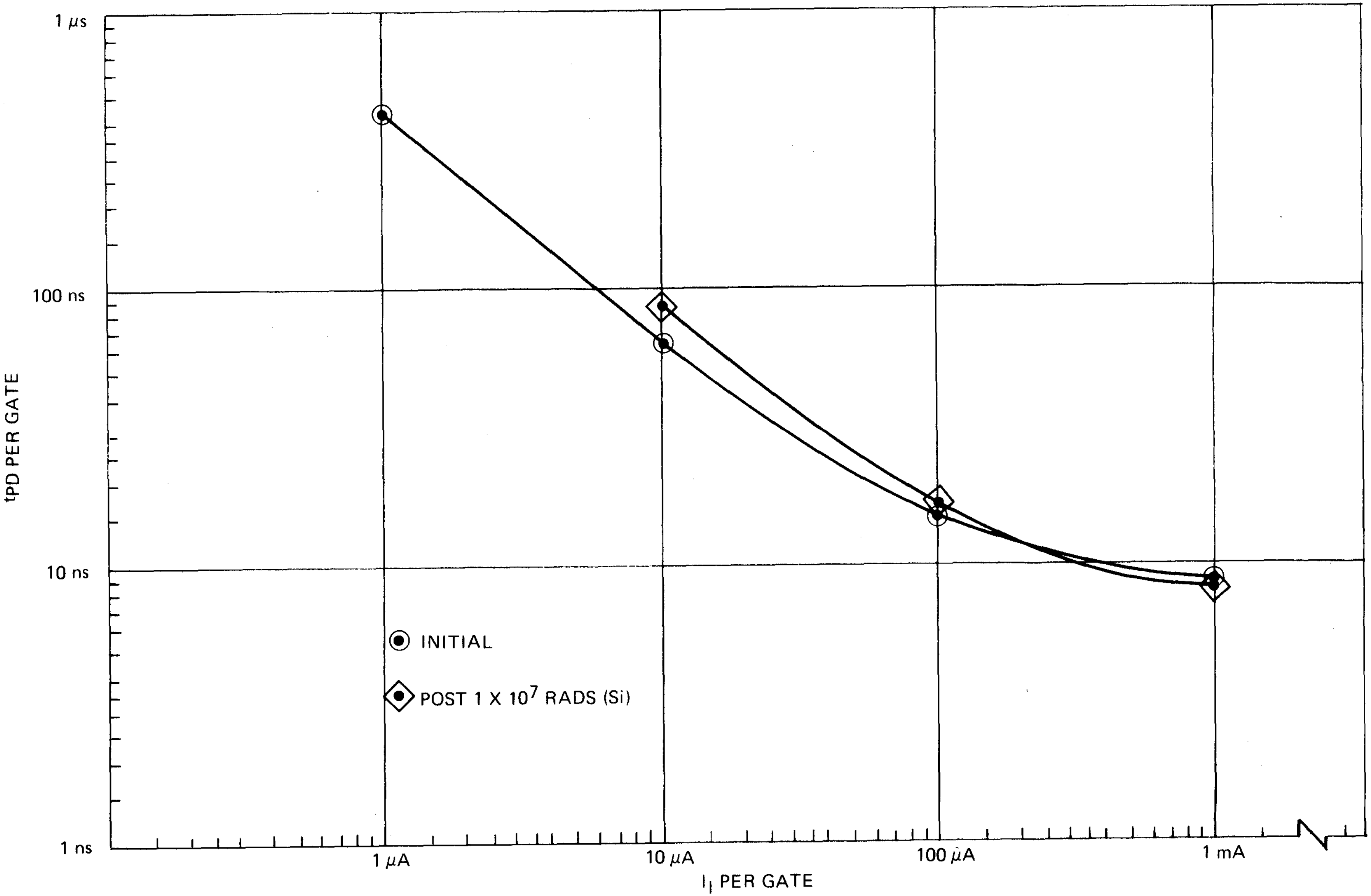


Figure D-13. Average t_{PD}/Gate vs I_I/Gate

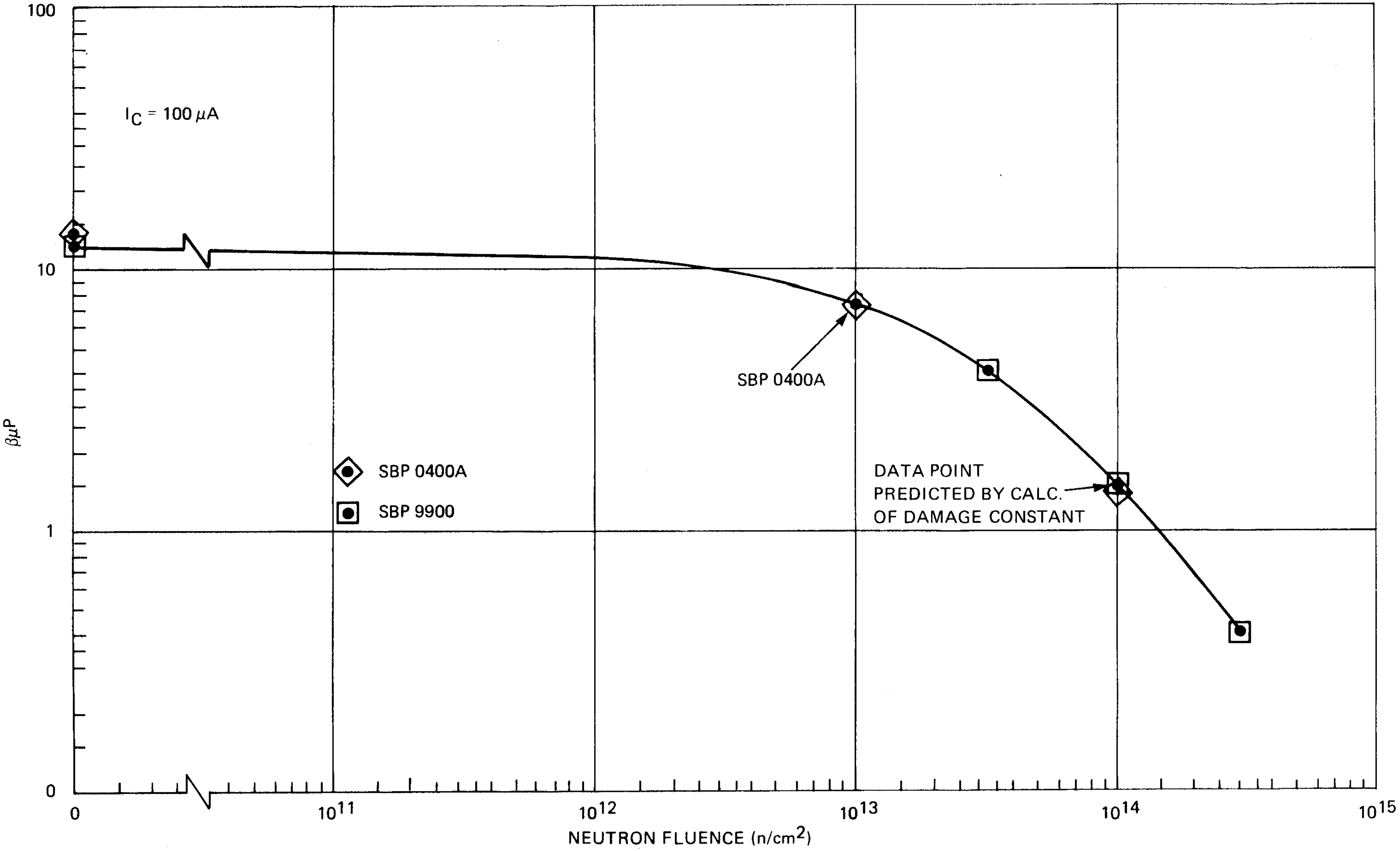


Figure D-14. β μ**P vs Neutrons**

APPENDIX E

HARDWARE/SOFTWARE SYSTEM

DEVELOPMENT AIDS

SBP/TMS 9900 TRANSPORTABLE CROSS-SUPPORT SOFTWARE PRODUCT DESCRIPTION

● PACKAGING

The SBP/TMS 9900 Transportable Cross Support package, which is now available for sales and distribution, is composed of three distinct products: 9900 Cross Assembler, Simulator, and ROM Utility. The part number for the package is *SYS9900/16F.X*. The ***Product Name*** is ***TMS 9900 Transportable Cross-Support Software***. Initially, the package will be manufactured only on ***½ inch, 9 track PE encoded*** (IBM compatible) magnetic tape recorded at ***1600 BPI***. The tape will be ***un-labeled, un-blocked***, with ***80 ASCII bytes per data record*** and will contain ***131 files***. The first file on the tape is a data file which contains:

a) A one-time descriptor for each file on the tape

b) A bill of materials (to verify that the complete package has been received), and

c) An errata list of problems and known solutions for the software version on that tape.

Each file on the tape is terminated by an EOF mark except for the last file which is terminated with a double EOF to indicate ***end-of-logical tape***.

Included in the shipping package is a ***User Manual*** for each of the three programs and an ***Installation Manual*** covering each of the three programs (4 manuals, total).

● OPERATING ENVIRONMENT

The programs are written to conform to ANSI STANDARD X3.0 (1966) 16-BIT FORTRAN and are designed to execute on any minicomputer with the following minimum characteristics.

1) ANSI STANDARD X3.0 (1966) 16-BIT FORTRAN COMPILER

2) Two's complement arithmetic

3) Disc capacity for up to 7 simultaneously active sequential files.

4) A 20K word user program memory partition.

To date, the package has been extensively tested on the TI 990/10 under DX10V2.2, DEC PDP11/10 under RSX-11M(FIN IV PLUS), and System 370/160 under MVS.

● 9900 CROSS ASSEMBLER DESCRIPTION

The SBP/TMS 9900 Assembly Language source is translated by the 9900 Cross Assembler into relocatable linkable SBP/TMS 9900 Object module format. Both the source input and the object output are fully compatible with the FS 990 Prototype Development System and TMS time-sharing services (GE TERMINET, NCSS, and TYMSHARE).

● SBP/TMS 9900 SIMULATOR DESCRIPTION

The Simulator is patterned after and affords extensions to the SBP/TMS 9900 Simulator on GE TERMINET, NCSS and TYMSHARE. Object modules generated by the Cross Assembler along with ''link-control'' statements are input to the first stage of the simulator. The output from this stage, an ***absolute,*** non-relocatable load module, plus simulation/debug control statements to the second stage of simulation. This stage may be operated in ''batchmode'' or interactively (e.g., the simulation/debug control stream is entered to the Simulator from a Keyboard/Display device). In this second phase of Simulation the user's program logic is verified and the program's performance characteristics are ascertained. Performance parameterization is supported for considerations such as target system clock speed, memory characteristics, and I/O part descriptions. Debug features include multiple breakpoints, full instruction trace and snapshots, plus the normal inspect/modify for CPU registers. All program references may be made symbolically, using symbols defined in the user's source program.

● ROM UTILITY DESCRIPTION

When the application program has been satisfactorily verified, the object module may be input to the ROM Utility Program for translation into a format acceptable for production of a gate placement program (preparatory to mass production). Alternatively, the utility may be used to generate a BNPF formatted file which may be input to a PROM Programmer (DATA I/O, etc.) to produce a PROM version of the program. In all, there are 12 acceptable input formats and 12 output formats in support of the TMS 1000 and the SBP/TMS 9900 microprocessors. TABLE I indicates the valid input/output translations supported by the utility.

AMPL . . . A COMPLETE MICROPROCESSOR PROTOTYPING LAB

The AMPL* Microprocessor Prototyping Lab combines the high performance 990 computer with the low-cost flexibility of the floppy disk to provide a complete microprocessor prototyping lab. The AMPL lab provides in-circuit emulation support, logic-state trace and analysis, read-only memory implementation aids, and SBP 9900A software development support. This microprocessor lab provides the user a dedicated design center where 9900-based systems can be developed in an integrated software-hardware design and debug sequence. Substantial savings in design cost and design time in each phase of new product development is ensured with this system.

The microprocessor lab is structured around the FS9900 system, a fellow member of TI's 9900/990 computer family. The FS990 system includes the Model 911 Video Display Terminal dual floppy-disk drive and TX990/TXDS system software license, all packaged in a self-contained single-bay desk. The Texas Instruments Model 810 Printer option for the FS990 system provides the user a fast throughput for software listings, trace data, and other hard-copy output requirements.

In-Circuit Emulation Support

The SBP 9900A in-circuit emulation feature includes the SBP 9900A emulator, SBP 9900A buffer, and SBP 9900A target connector. This feature allows the SBP 9900A microprocessor-based system design engineer to simulate his target system by utilizing the dedicated 4096 words of emulator memory and the SBP 9900A microprocessor emulator. All functions of the proposed system can be simulated except input/output, and benchmark data can be tabulated.

SBP 9900A emulation is designed to aid the design engineer through each stage of his prototype implementation. Emulation control provided by the FS990 system allows the design engineer to step through the developed code, setting breakpoints and instruction traces to start/stop tests at desired points within his code.

Two significant advantages included in the emulation feature are the use of dedicated emulator memory and the ability under interactive software control to switch back and forth between target system memory and emulator memory. The dedicated 4096 16-bit words of emulator memory provides a significant speed advantage over systems which utilize host system memory on a cycle-stealing basis. The faster, dedicated emulator is designed so that this dedicated memory can have precedence over target system memory so that even after target system memory is implemented, 9900 code changes can be quickly evaluated and tested before implementing this change in target system ROM/PROM.

Logic-State Trace

The logic-state trace feature adds a dramatic new dimension to the integration and checkout of the target system.

The FS990 system trace/emulation features interactive on-line control and analysis to provide fast data reduction and programmable emulation control based on the results of this analysis.

The trace feature can be interconnected with the emulator module or it can utilize the general-purpose Trace Data Probe. When interconnected to the emulator, the design engineer can trace 256 events of both address and 16-bit data. The Trace Data Probe provides 20 individual logic-line trace probes.

These probes can be used by the design engineer to trace any TTL logic lines desired in his target system. The sampling rate can be controlled by a 10-megahertz internal clock or by an external clock up to 10 megahertz. Of the 20 trace probes, 4 have a special glitch latch feature and can detect noise pulses down to 10 nanoseconds in width.

In addition to the 20 trace data probes, 4 general-purpose trace clock qualifier probes are provided to allow the user to prequalify trace conditions based on logic-state conditions within his target system. By using the interactive programming features within the host FS990 system, the design engineer can define procedures and functions to automatically process incoming trace data from these events, perform data reduction looking for defined conditions, display or print only the desired results, or branch into other emulation/trace procedures. Thus, for example the design engineer can set qualifying conditions and start trace and emulation in a continuous cycle while looking for those random troublesome noise glitches. Upon detecting a glitch, the trace/emulation cycle can be programmed to pause momentarily, analyze and print conditions, and then continue the trace/emulation sequence looking for the next glitch. This feature can mean tremendous savings in manpower and design checkout time since the full speed and power of the 990 computer is processing the problem.

MPL Microprocessor Prototyping Lab
Standard Kits and Options

FS990 System

The FS990/4 system includes 990/4 with 25K 16-bit words of memory, dual floppy-disk drives, Model 911 Video Display Terminal, TX990/TXDS system software license, one-year software subscription service, hardware installation* within the continental United States, and programming and user manuals. The system is mounted in a single-bay equipment desk (34 inches deep by 31 inches high by 54 inches wide).

FS990/4 System
Part Number 940031-0011

*Single service call for hardware installation is included in FS990 system price for the basic system configuration plus any of above listed options, within a 25-mile radius of the nearest Texas Instruments DSD service office. Travel beyond this radius will be in accordance with TI's then-current standard rates.

FS990 FORTRAN IV OPTION

This option includes FS990 FORTRAN software object license and one-year software subscription service.

FORTRAN IV
Part Number 936044-0003

Printer Option

The Model 810 Printer is a 132-column, 150-cps, 9×7 matrix printer with controller and 30-foot cable.

Model 810 Printer Kit
Part Number 838120-0001

TMS 990/9980 Emulator Options

The TMS 9900/9980 emualtor options include the 9900/9980 emulator, either the 9900 buffer module or 9980 buffer module, AMPL software license, one-year software subscription service, and manuals.

TMS 9900 Emulator Kit
Part Number 949960-0001

TMS 9980 Emulator Kit
Part Number 949960-0002

TMS 9940/SBP 9900A Emulator Options

The in-circuit emulator options for these microprocessors will be available in early 1978. Consult the nearest Texas Instruments sales office for details.

Logic-State Trace Option

The trace option includes trace support for 16 channels of data or address plus 3 control bits when used with SBP 9900A emulator or 20 channels of general-purpose TTL logic data when used with Trace Data Probe. This function requires one of the emulator options.

Trace Data Module
Part Number 949947-0001

Trace Data Probe
Part Number 949915-0001

PROM Programming Option

The 990 PROM Programming Module comes in a tabletop enclosure with controller and cable. Requires one of the adapter panels below:

PROM Programming Module
Part Number 944924-0001

PROM Programming Adapter
Part Number 945135-0001

EPROM Programming Adapter
Part Number 945165-0001

Texas Instruments reserves the right to make changes at any time as required in supplying the best product possible.

Sales and Service Offices of Texas Instruments are located throughout the United States and in major countries overseas. Contact Texas Instruments Incorporated, Digital Systems Division, P.O. Box 1444, Houston, Texas (77001), or call (713) 491-5115, for the location of the office nearest to you.

AMPL Software, an Interactive Control Language

The control of the SBP 9900A emulation and the logic-state functions is provided by AMPL software, a higher level language executing within the FS990 system. This interactive microprocessor prototyping language unifies the diverse 9900 prototyping support capabilities into a user-oriented system. AMPL software has designed-in features to simplify orientation for the new user, yet provides extensive flexibility and support for the experienced user. AMPL software features include:

- Block-structured concepts and simple, concise syntax
- Straightforward interactive evaluation and display of expressions
- User- and system-defined procedures and functions
- Interactive display/modification of target memory with instruction level assembly and conversion from internal binary to assembly instruction mnemonics

- Symbolic target system debug
- Dynamic processing of traced data collected by the emulator and trace modules
- Batch-like execution through predefined disk resident sessions.

The user-defined procedural and function support provided by AMPL software allows the user to develop standard support tasks for emulation, debug, and trace data evaluation which can significantly ease the microprocessor application design efforts.

9900 Software Development

The SBP 9900A software support includes a source editor, two-pass assembler, and link editor. The source editor is an interactive text editor that simplifies the creation and modification of SBP 9900A source code. The two-pass assembler supports the full 9900/990 assembly language and the resultant software can be implemented for the FS990 system and the SBP 9900A-based target system.

FS990 FORTRAN is a valuable aid to the design engineer for statistical, benchmark and other engineering/scientific support. The FS990 FORTRAN meets the American Standard FORTRAN IV (X3.9-1966) requirements.

The FS990 FORTRAN is structured to provide the design engineer capability to utilize FORTRAN subroutines within his target system. Output of the FORTRAN compiler can be defined to produce 990 object which can be linked into other SBP 9900A code for implementation into the target system.

PROM Programming

Part of every 9900 microprocessor-based target system memory will be implemented in nonvolatile read-only memory. This section may contain only the bootstrap loader program to control loading of the system program into RAM memory, or it may contain all of the target system code. The read-only memory support available within the FS990 system includes BNPF and HIGH/LOW formatted output, and PROM/EPROM implementation using the 990 PROM programming module.

The PROM programming module offers the SBP 9900A user capability to implement his target system memory in PROM or EPROM. Bipolar PROM devices which can be implemented include SN74188A, SN74S188, SN74S288, SN74S287, SN74S387, SN74S470, SN74S471, SN74S472, and SN74S473. EPROM device includes the TMS 2708. By using the PROM programming module with the FS990 system, the SBP 9900A user can develop 9900 code, test it through in-circuit emulation, and implement it in PROM or EPROM.

Packaged Systems

The packaged systems are coordinated hardware/software offerings. Users should be aware that the standard packaged systems are offered at a price that is less than the sum of the system components sold separately. In many cases, it will be more economical to select the packaged system that best meets the needs of the user application and add options or leave some features unused. Texas Instruments urges its customers to investigate the possible price advantages of this type of purchase. Note that substitutions are not permitted on the packaged systems.

FS990 Floppy-Disk Systems

FS990 systems provide the customer a facility for development, testing, and execution of applications programs in assembly language or FORTRAN. The dual floppy disks provide an economical mass storage capability for system programs, application programs, and user data. A video display terminal is provided for interactive software development and operator control of application programs.

Licensed TX990/TXDS software package provides a complete family of program development functions that run under control of an operating system executive. After the user programs are developed, they may be run under control of this same executive in the final application. The executive handles the system resource allocations, such as memory space, central processor time, and input/output, so that the user may concentrate on applications rather than system overhead. The operating system provides simplified logical input/output to the standard and optional devices specified for the FS990 systems. The software license includes a one-year software subscription service.

Those users who are developing ROM programs may record their programs in industry standard BNPF or HI-LO format for implementation by a ROM manufacturer, or, with the optional PROM programmer, "burn" PROM devices with the FS990.

Texas Instruments strongly recommends the selection of an optional hard-copy device, such as the Model 810 Printer, to record listings of user programs and to allow offline analysis of results.

Installation of the standard system hardware is included as part of the system price. Conditions of hardware installation are discussed under Customer Services.

FS990 systems are offered in two versions, the FS990/4 system and the FS990/10 system. The FS990/4 uses the economical Model 990/4 Microcomputer as the system central processor. The FS990/10 uses the faster Model 990/10 Minicomputer as the central processor.

FS990/4 Standard Configuration

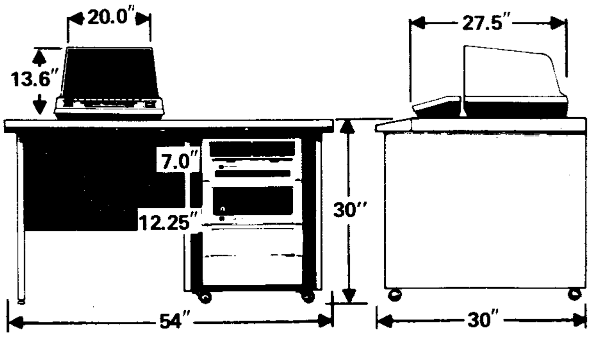
- Model 990/4 Microcomputer with 48K bytes of parity memory in a 13-slot chassis with programmer panel and floppy disk loader/self-test ROM
- Model 911 Video Display Terminal (1920 character) with dual port controller
- Dual FD800 Floppy-Disk Drives
- Attractive, office-style single-bay desk enclosure
- Licensed TX990/TXDS Terminal Executive Development System Software with one-year software subscription service.

FS990/10 Standard Configuration

- Model 990/10 Minicomputer with 64K bytes of error-correcting memory and mapping in a 13-slot chassis with programmer panel and floppy disk loader/self-test ROM
- Model 911 Video Display Terminal (1920 character) with dual port controller
- Dual FD800 Floppy-Disk Drives
- Attractive, office-style single-bay desk enclosure
- Licensed TX990/TXDS Terminal Executive Development System Software with one-year software subscription service.

Options

- Model 810 Printer (Item 630)
- PROM Programmer (Item 130)
- TXDS FORTRAN IV License (Item 240)
- AMPL* Microprocessor Prototyping Lab Kits (Item 120 or 123).



(Refer to the Configuration Guide for packaging constraints.)

*Trademark of Texas Instruments

Extensions

These extensions can be supported by the TX990 terminal executive, if the customer performs a system generation operation. The system generation operation consists of selecting and linking system component modules into a customized operating system. System generation (sysgen) is supported by the documentation and software supplied with the operating system.

NOTE: Although the user may generate a customized operating system that allows execution of application *program on multiple terminals, the program development functions of TX990/TXDS are limited to* one *terminal only.*

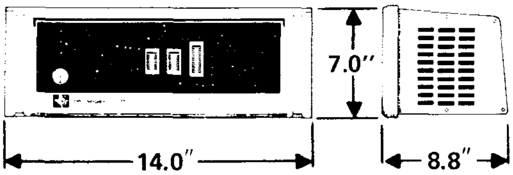
- *Model 804 Card Reader (Item 650)*
- *Up to two additional Model FD800 Floppy-Disk Drives (Item 502 or 503)*
- *Model 2230 Line Printer (Item 640)*
- *Model 2260 Line Printer (Item 641)*
- *Model 733 ASR Data Terminal (Item 621)*
- *Model 743 KSR Data Terminal (Item 620)*
- *Additional Model 911 Video Display Terminals (Items 610-612).*

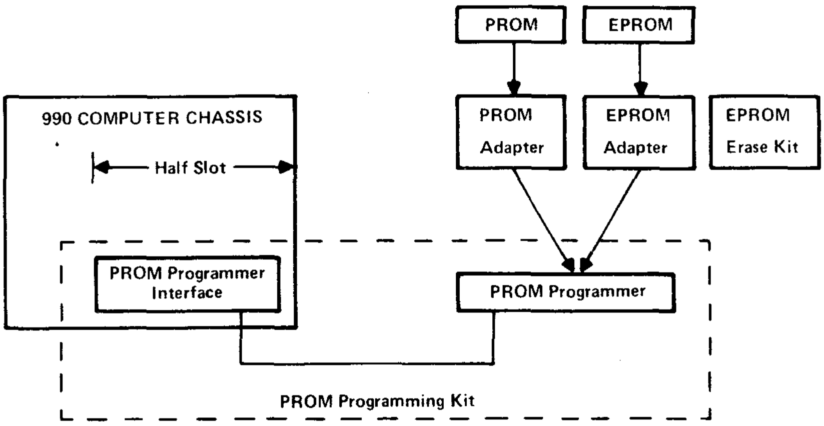
Item	Description	Comments
110	FS990/4 System	Includes licensed software and hardware installation. Refer to software policy in software subsection.
111	FS990/4 System without Software	Does not include software or installation.

Item	Description	Comments
112	FS990/10 System	Includes licensed software and hardware installation. Refer to software policy in software subsection.
113	FS990/10 System without Software	Does not include software or installation.

PROM Programmer

The PROM Programmer is a tabletop device that provides the correct signal levels and timing for programming PROM and EPROM devices. The interface card requires a half slot in the 990 chassis and interfaces to the CRU bus. Adapters plug into the PROM Programmer to provide the correct sockets and signals for particular devices.





The PROM Programming Adapter plugs into the PROM Programmer and programs SN74S287, 471, and 472 devices.

The EPROM Programming Adapter plugs into PROM Programmer and programs TMS 2708 and 2716 devices.

Item 133, EPROM Erase Kit, provides an ultraviolet lamp for erasing EPROM devices.

Item	Description	Comments
130	PROM Programmer Kit (Tabletop)	Ships with adapter selected below mounted in unit
131	PROM Programmer Adapter	Requires Item 130
132	EPROM Programmer Adapter	Requires Item 130
133	EPROM Erase Kit	

DS990 Disk-Based Systems

The DS990 Disk-Based System offers all the program development features of the smaller FS990 system plus many features that facilitate concurrent operation of multiple applications programs run from multiple interactive user terminals. High-level languages are available as options. Processing power for the DS990 systems is supplied by the Model 990/10 Minicomputer, which features TILINE asynchronous high-speed data bus and hardware address mapping for up to 2048K bytes of memory. File management and allocation of resources among multiple tasks is performed by the licensed DX10 Operating System.

Three standard systems are offered and are suited for the following purposes:

- Model 4–Small software development system or medium-scale application system using 10M bytes of disk storage.
- Model 6–Dual 25M-byte disk-based system suitable for medium-scale software development and application systems.
- Model 8–Dual 50M-byte disk-based system intended for medium- to large-scale software development and application systems.

The differences between these systems are based on the amount of disk storage required for the different applications. Physical packaging differences between the models are based on the space requirements of the disk storage units.

Installation of the standard system hardware and DX10 software is included along with the software license as part of the system price. Additional conditions of hardware installation are discussed ι nder Customer Services.

Standard Configuration–All Models

- Model 990/10 Minicc.nputer with mapping, 128K bytes of error-correcting memory in a 13-slot chassis with programmer panel and disk loader ROM
- Model 911 Video Display Terminal (1920 character) with dual-port controller
- Licensed copy of DX10 Operating System on compatible disk media, with one-year software subscription service
- Disk drive(s) and enclosure(s) as determined by model number.

Options–All Models

- Up to 128K bytes of additional error-correcting memory in TILINE expansion chassis (Item 820). For additional memory beyond 256K bytes, consult your field sales engineer.
- Model 911 Video Display Terminals (Items 610-612) up to a maximum of ten. CRU expansion chassis (Item 810) is required for more than four terminals. Contact your TI field sales engineer regarding additional memory requirements for more than four terminals.
- Printer–Model 810, 2230, or 2260 printers (Item 630, 640, or 641); highly recommended for hardcopy output
- Model 804 Card Reader (Item 650)
- Model 979A Magnetic Tape Drive 800/1600 bpi (Item 550 or 552), requires rackmount cabinet
- Model 733 ASR Data Terminal (Item 621)
- Model 743 KSR Data Terminal (Item 620)
- DX10 COBOL License (Item 242)
- DX10 FORTRAN IV License with ISA extensions (Item 241)
- DX10 BASIC License (Item 243)
- DX10 Business BASIC License (Item 244)
- DX10 Sort/Merge Utility License (Item 245).

Documentation

Hardware Manuals

All 990 Computer Family hardware products are fully documented in one or more of three types of manuals: installation and operation, field maintenance, and depot maintenance.

Installation and Operation Manual

The purpose of an installation and operation manual is to present specific information regarding the correct procedures and site preparation required for the successful installation of the hardware peripheral. An overview of the physical, electrical, and operational characteristics of the peripheral, including both the hardware device and its associated computer interface, are included. In addition, the software requirements are detailed to assist the customer who intends to develop customized application software drivers as opposed to using the standard TI operating system(s) device service routine(s).

Field Maintenance Manual

The field maintenance manual outlines the preventive maintenance procedures required to maintain the computer and/or peripherals in good operating condition and presents equipment malfunction troubleshooting techniques. In general, the maintenance procedures are intended to resolve equipment problems at the assembly or board-swap level.

Depot Maintenance Manual

Depot maintenance manuals present detailed electrical and mechanical data to allow circuit-level diagnosis and resolution of equipment malfunctions.

Item Number			
Install.	**Field**	**Depot**	
and Oper.	**Maint.**	**Maint.**	**Description**
902	924	928	990/4 Microcomputer
903	925	929	990/10 Minicomputer
904	*	930	FD800 Floppy Disk
905	*	931	DS31/32 Disk
906	*	–	DS 10 Disk
907	–	933	DS25/50 Disk
908	–	934	979A Tape Controller
909	–	935	979A Tape Transport
910	–	936	911 VDT
911	–	937	743 KSR
912	*	938	733 ASR
913	–	–	810 Printer
914	–	–	2230/2260 Line Printer
915	*	932	804 Card Reader
916	–	939	TMS9900 Emulator and Buffer
917	–	940	Trace Data Module
918	*	941	PROM Programmer
919	–	942	TTY/EIA Module
920	–	943	990 Communications
921	–	944	Autocall Unit
–	–	945	16 I/O EIA Module
–	–	946	16 I/O TTL Module
922	–	947	D/A Converter Module
922	–	947	A/D Converter Module
923	–	–	32 Input/Transition Module
923	–	–	32 Output Module
923	–	–	Digital I/O Panel
–	926	–	990 Peripherals
			990 Family Drawings
–	–	948	Volume I–Processors
–	–	949	Volume II–Peripherals
–	927	–	990 Diagnostic Handbook

*Included in Model 990 Computer Peripheral Equipment Field Maintenance Manual, Item 926.

Software Manuals

990 Computer Family software is fully documented in one or more of the following types of manuals: system operation guides, user guides, programmer's guides, reference cards, and installation procedures.

System Operation Guides

A system operation guide is provided with each of the packaged systems for the 990 Computer Family. This manual links together the hardware components of the system with the software that accompanies it and describes the concepts required to effectively use the system. This information includes installation instructions, procedures for verifying that the system is operating effectively, plus operating instructions for using each of the software packages in conjunction with the hardware included with the system.

User Guides

User guides provide information about individual software packages that are not ordinarily used in conjunction with a specific system. Each guide contains a description of the functions and capabilities of the package as well as detailed instructions for effectively using the package.

Programmer's Guides

Programmer's guides provide complete, detailed coverage concerning an operating system or programming language. These guides provide all the information an experienced programmer requires to interface with the 990 Computer Family through the subject medium.

Reference Cards

These handy pocket-size cards condense the essential information necessary to program the computer. These cards list the instruction set and give formats for the different instructions, as well as summarizing other helpful concepts.

Installation Procedures

These procedures supply the information necessary to initially install a software package into a system.

See the price list for descriptions, part numbers, and pricing of all available software manuals. Note that at least one set of applicable manuals is supplied with each software package.

Software

This section provides a product description for each of the standard software products available for the 990 Computer Family. Information is provided under the following headings:

Description—A product description summary for each product listing key features

Applications—Intended applications environment

Components—Software components included

Prerequisites—Minimum hardware and software configuration required to use the software

Expansion—Maximum system expansion configuration

More detailed information about 990 software products is located in 990 software data sheets available upon request from your local TI field sales engineer.

For pricing and ordering information including part number, price, format, and distribution media, refer to the appropriate item number in the price list. Each software product is available only on the media specifically listed for that product.

The item number convention followed for all 990 Computer Family software products is as follows:

Item	Description
210-219	Memory Resident System Software
220-229	Floppy-Disk System Software
230-239	Disk-Based System Software
240-249	Languages and Utilities

Software License

990 software is classified in three categories: Category A licensed software, Category B licensed software, and unlicensed software. Licensed software products are furnished under a Program License Agreement whereby Texas Instruments retains ownership of the software and licenses the right to use it in accordance with specified terms and conditions.

Category A Software—The software is purchased once and there is no additional charge for limited reproduction.

Category B Software—Licensed per CPU. A license fee must be paid once for every CPU on which the software will reside.

Unlicensed software—The software is purchased without licensing restrictions.

For further details, consult the price list and the TI Program License Agreement. Copies of the agreement may be obtained from your local TI sales office.

Software Support

Subscription Service—990 licensed software includes a subscription service providing software updates for one year following purchase. Annual renewal subscription service is available after the initial one-year term. Update subscription service provides an updated version of the software and a one-year subscription service for customers who allow their subscriptions to lapse. For further details, consult the software policy section of the price list.

No subscription service is provided for unlicensed software. Software updates are available for purchasers of the unlicensed package.

Training—Regularly scheduled programming courses on the 990 computer family are offered by the TI Education and Development Center. Self-study courses are also available. Consult the education subsection under Customer Service.

Software Installation—The software license of 990 system software, where so specified in the price list, will include installation support. This support will consist of verifying TI software operation in accordance with the TI-supplied installation procedure.

Hardware installation by a TI customer engineer is a prerequisite for software installation support.

Travel expense for software installation will be subject to the same terms and conditions applying to hardware installation described under Customer Services.

Delivered Software

Software is provided on the media specifically listed for that product in the price list. A complete set of user manuals is provided.

In addition to object format, source packages for most 990 software are available for developing custom functions. A source package requires assembly and linking on the appropriate 990 program development system prior to execution. Unless indicated otherwise for a particular package, a DS990 system is required. Pricing for source software can be obtained from local TI sales offices.

Item 210 EX990 Operating System

The EX990 Operating System provides a memory-resident, multitasking executive designed to minimize memory size and execution time through user specification of standard, modified, or user-supplied system modules. Standard capabilities include two levels of priority task scheduling, RAM/ROM partitioning, internal interrupt handling, supervisor calls, simplified 911 VDT, 733 ASR, and EIA I/O, operator communications, interval timer processing, and diagnostics.

EX990 is a system building block designed to support dedicated applications on the smaller members of the 990 family (TMS 9900 and 990/4). User tasks may be developed and linked to the EX990 system using one of the standard 990 program development systems (FS990 or DS990).

Item 211 TX990 Operating System

TX990 is a memory-resident, multitasking executive constructed of linkable object modules (many of which are optional) that allow the system to be tailored to support only the features desired. Features include four levels of priority task scheduling, interrupt handling, I/O processing, supervisor call processing, operator communications, and floppy-disk file management for sequential and relative record files.

TX990 provides a memory-resident target system for dedicated assembly-language or FORTRAN IV application programs developed with TXDS or DX10 software systems. Multiple application tasks can be linked with TX990 and a single task can be installed dynamically.

Item 220 TXDS Floppy-Disk System

TXDS software is a floppy-disk system including the TX990 multitasking executive with floppy-disk file management for sequential and relative record files, a complete set of assembly-language software development utilities, plus copy/concatenate, PROM programmer, BNPF/high-low dump, and IBM diskette conversion utilities. The TXDS control program provides prompts and interactive program control. TXDS also provides the host system for the AMPL software. FORTRAN IV is supported as an option.

TXDS provides single-user program development capabilities in assembly language or FORTRAN IV and provides a target system for floppy disk-based OEM applications. TX990 can be separated from TXDS to provide a memory-resident executive system for programs developed under TXDS or DX10. Firmware development is provided if the optional PROM Programmer hardware is selected.

Item 230 DX10 Disk System

DX10 is a general-purpose, multitasking, disk-based operating system featuring multikey-indexed file management, multiterminal performance, and program development support in both batch and interactive modes. Concurrent tasks are dynamically scheduled at four priority levels, incorporating shared procedures, overlays, and roll in/roll out. The DX10 system command interpreter provides a high-level interactive language to perform prompting and verification functions. Over 160 system commands serve a range of functions from disk backup and restore to initializing time and date.

DX10 is designed for a wide range of commercial and industrial applications on the 990/10 minicomputer. FORTRAN, COBOL, BASIC, and Business BASIC languages are available for application programming. A Sort/Merge package is also available. Programs can also be developed using assembly language or FORTRAN to execute on the smaller floppy-disk-based or memory-resident members of the 990 family for OEM applications.

Components	Prerequisities	Expansion	Comment
The minimum EX990 system occupying approximately 1K bytes of memory consists of the task scheduler, two task supervisor calls, and the EX990 system data base. Other modules are optional. EX990 is supplied in source format only.	TMS 9900 Microprocessor or 990/4 is required with a minimum of 1K bytes of RAM. System generation requires use of a 990 development system (FS990 or DS990).	64K bytes maximum memory on 990/10 or 56K bytes maximum on 990/4; 911 video display terminal; 733 ASR/743 KSR data terminal; 810 printer; Standard EIA Interface device	Source format only
A minimum TX990 requires a task scheduler, interrupt handler, and supervisor call interface. Optional modules include additional supervisor calls, logical I/O, operator communications, and floppy-disk file management.	990/4 or 990/10 computers are required with at least 8K-byte memory for a minimum TX990 system. Development requires the use of a DX10 or TXDS system.	64K bytes maximum memory on 990/10 and 56K bytes maximum memory on 990/4; 911 video display terminals; 733 ASR/743 KSR data terminal; 810, 2230, or 2260 printer; FD800 floppy disk; 804 card reader	TX990 is also included as part of the TXDS system software, Item 220.
TXDS software includes the TX990 Operating System, Two-Pass Assembler, Text Editor, Linker, Debug Package, Sysgen Program, and Floppy-Disk File Management. Additional utilities include the IBM floppy format Conversion, PROM Programmer, BNPF/High-Low Dump, and Cross Reference utilities.	990/4 or 990/10 computer with 48K-byte memory; 911 video display terminal; dual FD800 floppy-disk drives and ROM loader	56K-byte maximum memory on 990/4 or 64K-byte maximum memory on 990/10; 911 video display terminals (single user only for development); four FD800 floppy-disk drives; 810, 2230, or 2260 printer; 733 ASR/743 KSR terminal; 804 card reader; PROM programmer; TXDS FORTRAN IV; AMPL software and hardware	TXDS software is included as part of the FS990/4 and FS990/10 systems (Item 110 and 112).
DX10 system executive; logical I/O including extended 911 VDT support, interpreter for interactive, and batch operation; program development utilities including Interactive Text Editor, Macro Assembler, Link Editor, and Debug package; system log; and file management including sequential, relative record, and multikey-indexed files.	990/10 with mapping; 128K bytes of memory; 911 video display terminal; DS10 disk or dual DS25 or DS50 disks; selectable system disk ROM loader	2M bytes maximum memory; 911 video display terminals; 810, 2230, 2260 printers; 733 ASR/743 KSR terminals; 979A mag tape drives; 800 or 1600 bpi; DS31, DS10, DS25, DS50 disk drives; FD800 floppy-disk drives (physical I/O only); 804 card reader; FORTRAN IV; COBOL; BASIC or Business BASIC; Sort/Merge software	DX10 software is included as part of the DS990 systems (Items 140, 141, 144, and 146).

Description **Application**

Item 240 TXDS FORTRAN IV

TXDS FORTRAN is designed to allow development of FORTRAN programs on 990 TXDS floppy-disk software systems, such as the FS990 system. The TXDS FORTRAN IV compiler is an enhancement of ANSI standard X3.9-1966 including optimization, debug options, and program development features.

TXDS FORTRAN compiler operates under the TXDS system software. Compiler output can be linked to other 990 code for execution on 990 target systems under control of TXDS/TX990 system software or standalone with user-supplied I/O.

Item 241 DX10 FORTRAN IV

DX10 FORTRAN IV is an enhancement of ANSI FORTRAN IV (X3.9-1966) adding ISA recommended extensions (ISA-S61.1-1975 and S61.2-1976) and other useful program development aids including direct access I/O and debug options. The DX10 FORTRAN optimizing compiler operates under the DX10 Disk System Software, which allows concurrent execution of many FORTRAN programs. Overlay capability is provided.

DX10 FORTRAN compiler operates under the DX10 Disk System Software. Run-time library allows execution of compiled FORTRAN programs under DX10 and TX990 Operating Systems, limited standalone (no operating system), or with a user-written operating system.

Item 242 DX10 COBOL

The DX10 COBOL compiler/interpreter is a business-oriented, multiuser, high-level computer language that conforms to ANSI COBOL subset X3.23-1974 (level 1 nucleus, table handling, and sequential I/O) and adds useful higher level extensions including interactive video display terminal I/O, relative- and key-indexed I/O, library, segmentation, interprogram communications, and debug aids.

The DX10 COBOL multiterminal,computer/interpreter operates under the DX10 Disk System Software and is structured to handle large data files and typical business records.

Item 243 DX10 BASIC

DX10 BASIC is a floating-point, scientific-oriented version of Dartmouth BASIC with extensions. Features include integer and real data type, expanded string manipulation, external subroutine CALL capability, matrix arithmetic, mathematical function library, formatted I/O, and sequential and relative file I/O support.

DX10 BASIC is an interactive language designed for simultaneous use by multiple users operating under the DX10 Disk System Software. Each user may code, compile, load, execute, and debug programs.

Item 244 DX10 Business BASIC

DX10 Business BASIC is similar to DX10 BASIC with the following exceptions: decimal arithmetic replaces real, matrix, and trigonometric operations. Key-indexed files support is added.

DX10 Business BASIC is designed for developing business applications operating under the DX10 Disk System Software.

Components	Prerequisites	Expansion	Comment
TXDS FORTRAN compiler and function run-time library	FS990 system; or 990/4 with 48K-byte memory or 990/10 with 64K-byte memory; 911 video display terminal; dual FD800 floppy disks; TXDS system software		DX10 FORTRAN IV also incorporates all the modules in TXDS FORTRAN IV.
FORTRAN IV compiler and run-time library	Compiler–DS990 system or 990/10 disk system with 128K-byte memory; 911 video display terminal; DX10 Disk System Software		DX10 FORTRAN IV also incorporates all the modules in TXDS FORTRAN IV.
COBOL compiler and run-time interpreter	DS990 system or 990/10 disk system with 128K-bytes memory; 911 video display terminal; DX10 Disk System Software.		
DX10 BASIC language processor.	DS990 System; or 990/10 disk system with 128K-byte memory; 911 video display terminal: DX10 Disk System Software		
Business BASIC language processor.	DS990 System; or 990/10 disk system with 128K-byte memory; 911 video display terminal; DX10 Disk System Software		

Item 245 DX10 Sort/Merge Utility

The DX10 Sort/Merge utility provides full, address, key and summary sorts and full and summary merges on up to five input files. Sequence, record length, record selection, and reformatting are all user selectable. The replacement selection algorithm efficiently handles large volumes of data and nearly sorted data.

Sort/Merge operates under the DX10 Disk System Software and is accessible as a utility under DX10 in interactive and batch mode or called from COBOL, FORTRAN, BASIC, Business BASIC, or assembly-language programs. The floppy disk is not supported by DX10 Sort/Merge.

Item 246 990 Diagnostic Kit

The 990 diagnostic package includes stand-alone demonstration tests for the 990/4 and 990/10 computer families and their associated hardware devices.

There are two general categories of tests, depending on the type of hardware being tested. Those tests that require operator interaction utilize an operator-oriented test monitor. Those tests that do repetitive tests not suited for operator interaction are performed as stand-alone tests.

Each diagnostic package, except the disk kits, contains all available 990/4, 990/10, and standard peripheral tests in object code format with documentation. The disk kits support only the 990/10 and its peripherals.

The 990 diagnostic package supports 990/4 and 990/10 users who desire to perform their own equipment maintenance.

Central Processors

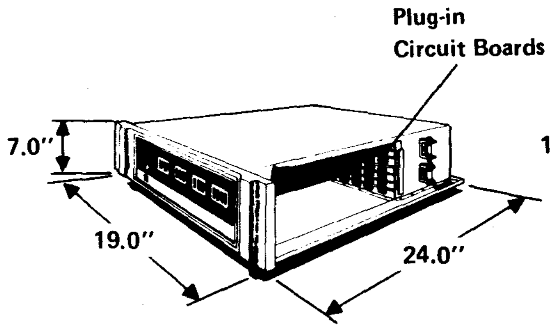
Model 990/4 Microcomputer

The Model 990/4 Microcomputer is a complete, low-cost computer packaged on a single circuit board. When supplied with a front panel, chassis (including power supply), and add-on memory modules, the 990/4 is a flexible, powerful, and inexpensive solution to a wide range or processing and control problems. The 990/4 CPU board includes on-board real-time clock, hardware multiply/divide, power-fail auto-restart logic, 8 vectored interrupts, and a command-driven CRU bus for up to 4096 input and 4096 output lines. The CPU board contains 8K bytes of dynamic RAM with parity. On-board sockets accommodate up to 2K bytes of PROM (SN74S287), ROM, or static RAM (TMS 4043). The 990/4 CPU can address expansion memory boards.

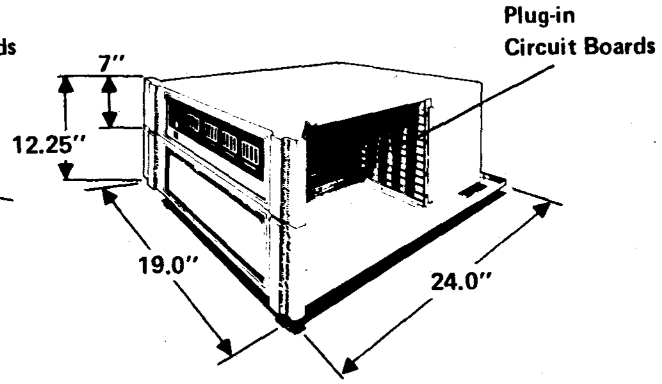
The basic function of the 990/4 loader is to initialize the computer from a "cold start" (power turned on with memory empty). The data to be loaded must be resident in a suitable peripheral connected to the 990/4. The standard loader provides (1) diskette load from FD800 drive, (2) cassette load from 733 ASR, (3) CPU self-test, and (4) programmer panel management utility. The CPU self-test function is automatically initiated during the load procedure and executes a series of short software test routines (such as MULT/DIV) to determine the operational status of the CPU. Should the self-test function detect a malfunction, the fault indicator on the front panel remains lighted and the CPU is prevented from executing all other software. All loader functions are contained in 1K byte of preprogrammed PROM memory.

This subsection contains the information required to select the desired chassis, front panel, standby power supply, mounting, and expansion memory options. The computer is shipped in assembled form. A hardware reference manual is included with all Model 990/4 Microcomputers.

6-Slot Chassis



13-Slot Chassis



Chassis Options

6-Slot–Furnishes all power, cooling air, and backplane wiring for 990/4 CPU (uses one slot) and 5 additional full-size standard interface circuit boards. Maximum operating temperature is 50°C.

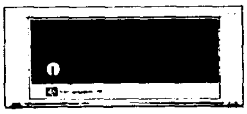
13-Slot–Furnishes all power, cooling air, and backplane wiring for 990/4 CPU (uses one slot) and 12 additional full-size standard interface circuit boards. Maximum operating temperature is 40°C.

Front Panel Options

Operator Panel–Provides key-lock OFF/ON/LOAD switch and POWER and FAULT indicators.

Programmer Panel–Provides key-lock OFF/ON/ENABLE switch and POWER and FAULT indicators plus additional indicators and switches for manual control of CPU from front panel. The switches provide capability to load and read memory and registers. Manual control can be disabled by key-lock switch. Panel control is implemented by firmware in ROM loader. Programmer panel should be used where operation requires loading or examining memory content such as during software debug or debugging of custom interface designs.

Programmer Panel



Operator Panel

990 Front Panel Options

Components	Prerequisites	Expansion	Comment
Control Statement Compiler Module, Record Selection and Reformatting Module, Sort Module, Merge Module, and Summary Module.	DS990 System; or 990/10 disk system with 128K-byte memory; 911 video display terminal; DX10 Disk System Software		
Cassette Package--Each cassette of the kit contains multiple versions of a particular diagnostic. For example, the object code for the 990/10, 990/4, and front panel versions of the line printer diagnostic would reside on one cassette. *Card Package*—The kit includes a card deck for each diagnostic. *Floppy Disk Package*—Each diskette contains multiple diagnostics and is labeled to show their contents. Each diskette also contains software to help the operator specify, load, and execute the resident diagnostics. *DS31, DS10, DS25, DS50 Disk Packages*—All diagnostics related to the 990/10 and its peripherals are packaged on a single disk pack. A software package is provided to help the operator specify, load, and execute the resident diagnostics.	990/4 or 990/10 computer; I/O device compatible with selected input media; ROM loader for selected I/O device		

Power Option

Standby Power Supply—When computer operation is interrupted by power failure, dynamic RAM memory content is preserved for 1 hour at 25°C (for 56K bytes installed). This option furnishes battery, charger, and regulated memory standby power. This option does not require additional slots within chassis.

No Standby—Power loss results in loss of all data in RAM.

NOTE: Power-fail/auto-restart logic requires that the user implement a coherent shutdown and restart, which is not automatically assured by standby power supply, 990/4 CPU board 2K-byte static memory content (if installed by user) is not maintained by standby power supply; however, the 8K-byte dynamic RAM content is maintained.

Model 990/4 Configuration Table

Item	Option	Chassis	Panel
300	No Standby	6-Slot Chassis	Operator Panel
301	Standby Power Supply		
302	No Standby	13-Slot Chassis	
303	Standby Power Supply		
304	No Standby	6-Slot Chassis	Programmer Panel
305	Standby Power Supply		
306	No Standby	13-Slot Chassis	
307	Standby Power Supply		

Mounting Options

A pair of slides for mounting one chassis, either 6-slot or 13-slot, in standard 19-inch equipment cabinets is available to allow convenient extension of the chassis from cabinetry for service.

Dustcovers for tabletop operation are also available.

Item	Description
310	990 Computer Chassis Rackmount Slide Set
311	6-Slot Chassis Tabletop Dustcover
312	13-Slot Chassis Tabletop Dustcover

NOTE: TI maintenance rates apply only to units ordered with the above listed dustcovers or mounted with slide rails in cabinetry.

Memory Options

Two types of expansion semiconductor memory modules are offered for Model 990/4 Microcomputers: a dynamic RAM module with parity with 8K to 40K bytes and an EPROM module with 2K to 16K bytes.

Dynamic RAM modules consist of a single full-size circuit board with 8K to 40K bytes of read/write storage. These modules include write-protect feature and parity with provisions to light on-board LED and interrupt CPU in case of fault. On-board switches set memory address on 8K boundaries. Memory cycle time is 667 nanoseconds.

Item	Description
320	990/4 Parity Memory Module, 8K Bytes
321	990/4 Parity Memory Module, 16K Bytes
322	990/4 Parity Memory Module, 24K Bytes
323	990/4 Parity Memory Module, 32K Bytes
324	990/4 Parity Memory Module, 40K Bytes

EPROM memory module consists of a single full-size circuit board with 2K bytes of TMS 2708 EPROM memory and sockets for an additional 14K bytes. On-board switches set memory address on 2K boundaries. Content may be erased and reprogrammed via the PROM Programming Kit (Item 130) with the EPROM Programming Adaptor (Item 132) and EPROM Erase Kit (Item 133). Cycle time is 667 nanoseconds. Maintenance is for controller only; TI cannot be responsible for EPROM content.

Item	Description
325	EPROM Memory Module

SHIPPING NOTE: Memory modules ordered on the same purchase order with a Model 990/4 Microcomputer will be installed in the computer chassis for shipment.

Model 990/10 Minicomputer

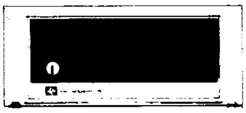
The Model 990/10 Minicomputer is a high-performance minicomputer with an instruction set that is fully upward compatible with the Model 990/4 Microcomputer and TMS 9900 Microprocessor. The 990/10 is implemented with TTL MSI circuits on two full-size (11" x 14") circuit boards. One board contains the AU and the other board contains the memory interface circuits. The two boards are interconnected across the top edge by two short cables and must be adjacent in the CPU chassis. The two CPU boards are located in the top two slots of the chassis. There are two versions of the memory interface board: one offering slightly higher speed with 64K bytes total memory capacity, and the other implementing a mapping scheme that increases total memory capacity to 2048K bytes. Both versions feature a high-speed multiuser bus structure, the TILINE, that supports high-speed peripherals such as disks and magnetic tapes.

Both versions of the 990/10 provide 1K byte of PROM program loader on the memory interface board; but loaders are not interchangeable between the unmapped and mapped versions. The standard loader included with 990/10 without mapping provides diskette load from FD800 drive, cassette load from 733 ASR, CPU self-test, and programmer panel management utility. The standard loader included with 990/10 with mapping provides disk load from moving-head disk, cassette load from 733 ASR, card reader load, magnetic tape load, and programmer panel management utility.

Model 990/10 Minicomputer features include on-board real-time clock, integer hardware multiply/divide, power fail/auto restart logic, 16 vectored interrupts, 16 extended operations (XOPs), high-speed TILINE multiuser bus, CRU bus for decoding up to 4096 input lines and 4096 output lines, and interface for operator/programmer panel.

The chassis is the 13-slot chassis (CPU uses top 2 slots) with self-contained power supply for CPU and standard interface cards. Chassis dimensions are 12.25" H x 19.50" W x 30.00" D. There are multiple options in front panels and power supplies. The maximum operation temperature is 40°C.

Programmer Panel



Operator Panel

990 Front Panel Options

Chassis Options

Operator Panel–Provides key-lock OFF/ON/LOAD switch and POWER and FAULT indicators.

Programmer Panel–Provides operator panel functions plus switchboard indicator lights for full manual control of 990/10 from front panel. This includes ability to load and read registers and memory. Panel control is implemented by firmware in ROM loader. Programmer Panel should be used where operation requires loading or examining memory content, debugging custom designs, etc.

Power Options

Standby Power Supply–When computer operation is interrupted by power failure, up to 64K bytes of RAM memory content is preserved for 1 hour at 25°C. This option furnishes battery, charger, and regulated memory standby power. This option does not require additional slots within chassis.

No Standby–RAM memory content is lost in the event of power failure.

NOTE: Power-fail, auto-restart logic requires that the user implement a coherent shutdown and restart, which is not automatically assured by standby power supply.

Model 990/10 Configuration Table

Item	Option	Panel	Description
400	No Standby	Operator Panel	990/10 without Mapping (64K Memory Space)
401	Standby Power Supply		
402	No Standby	Programmer Panel	
403	Standby Power Supply		
404	No Standby	Operator Panel	990/10 with mapping (2048K Memory Space)
405	Standby Power Supply		
406	No Standby	Programmer Panel	
407	Standby Power Supply		

Mounting Options

A pair of extending slides is available for mounting one chassis in an equipment cabinet.

A dustcover suitable for office environments is an option for tabletop mounting.

Item	Description
310	990 Computer Chassis Rackmount Slides
312	13-Slot Chassis Tabletop Dustcover

NOTE: TI maintenance rates apply only to units provided with a suitable dustcover or mounted with extending slides in an equipment cabinet.

Memory Options

TI offers three semiconductor memory options for Model 990/10 Microcomputers. The different types may be intermixed on a single CPU up to the address and power limits of the CPU. The modules are:

- EPROM Memory Module, Item 325 (see 990/4 section)
- Parity memory module including controller with 32K bytes per module.
- Error-correcting memory based on 4K x 1 RAM integrated circuits consisting of controller board with 16K bytes and one add-on board with 16K, 32K, or 48K bytes. One 2-board module set provides up to 64K bytes. Use multiple module sets for larger memories.

Parity Memory

The parity memory module is a single full-size board (11" x 14") with 32K bytes of read/write storage. It includes a parity feature with provision to light on-board LEDs and interrupt the CPU in case of fault. On-board switches map memory address anywhere in TILINE address space on 8K boundaries. Memory cycle time is 725 nanoseconds without mapping, 825 nanoseconds with mapping.

Item	Description
413	Parity Memory Module, 32K Bytes

Error-Correcting Memory

Error-correcting memory modules consist of a controller circuit board and one expansion circuit board. The controller contains a TILINE interface, error-correcting circuitry, 16K bytes of dynamic RAM, and refresh circuitry. The module can be expanded by adding one expander board with 16K, 32K, or 48K bytes of memory. The controller and expander are interconnected by a top edge connector that requires that the boards be adjacent. One-bit errors light a correctable-error LED on the controller board and are corrected by the circuits. Two-bit errors light a non-correctable-error LED and interrupt the CPU. Switches on the controller board set the memory address anywhere in TILINE address space on 8K boundaries. Controller and expander memory occupy contiguous addresses. Top-edge connector is included with expansion module. Memory cycle time is 825 nanoseconds without mapping, 925 nanoseconds with mapping.

NOTE: Texas Instruments strongly recommends error-correcting memory for systems with more than 48K bytes of memory.

Item	Description	Comments
420	Error-Correcting Memory Controller, 16K Bytes	
421	Error-Correcting Expansion Memory, 16K Bytes	Requires Item 420
422	Error-Correcting Expansion Memory, 32K Bytes	Requires Item 420
423	Error-Correcting Expansion Memory, 48K Bytes	Requires Item 420

SHIPPING NOTE: Memory modules ordered on the same purchase order with a Model 990/10 Minicomputer will be installed in the computer chassis for shipment.

APPENDIX F

SBP 9900A-BASED

TM 990/110M MICROCOMPUTER

TM 990/110M Microcomputer

The TM 9900/110M is an assembled, tested microcomputer module utilizing as its CPU the powerful, ruggedized, I^2L 16-bit SBP 9900A microprocessor. With RAM and ROM/EPROM included on board as well as programmable serial and parallel I/O, the TM 990/100M is a powerful single-board microcomputer. Since all address, data, and control lines are brought to the board connectors, the board can be expanded to use the entire capabilities of the SBP 9900A in larger systems. The TM 990/110M is also being reformatted to meet the form factor/ruggedness requirements of the U.S. Navy's standard electronic module (SEM) program.

Features

- SBP 9900A 16-bit CPU
- 1024 words of 4045 Static RAM
- 1K words of 2708 EPROM, expandable to 4K words using 2716 EPROMs
- SBP 9960/SBP 9961 programmable system interface
- TMS 9902 asynchronous communications controller
- EIA or TTY terminal interface option
- Fully expandable bus structure
- Designed to fit the 990/510 chassis
- TIBUG operating monitor.

Operation

The TM 990/110M microcomputer is a software compatible member of the SBP 9900A/990 family. The SBP 9900A is used as a CPU to provide 16 bits of processing power with a minicomputer instruction set which includes multiply and divide. The TM 990/110M module is designed for 3 MHz operation, utilizing the full 16 levels of prioritized interrupts and the advanced memory-to-memory architecture of the SBP 9900A. Additionally, the bus structures are set up to take advantage of the full 64K byte memory addressing capability of the SBP 9900A and the nonmultiplexed memory, I/O and interrupt buses.

Memory

The on-board memory includes both an EPROM/ROM section and a static RAM section. Four sockets are available for TMS 2708, TMS 2716 EPROM or TMS 4700, TMS 4732 ROM operation. Using the available jumper option, all four sockets can be populated with TMS 2716's, providing a maximum on-board EPROM capability of 4K words. The static RAM area consists of 1024 words of memory. Four TMS 4045's are included. The cycle time of this memory is 0.667 microseconds. The address map is shown in Figure 1; the minimum area of EPROM/RAM area may not be used for off-board expansion. DMA control lines are also accessible on the bus.

Interrupts and Timers

Fifteen maskable interrupts plus the reset and load trap vectors are implemented. Table 1 shows the implementation. The SBP 9961 handles all 15 external interrupts which can be generated from either the bus connector or the SBP 9961 I/O bus. The SBP 9961 enables each level to be individually maskable under program control. Additionally, level 3 can be programmed to use the interval timer in the SBP 9961. Level 4 can be generated from the TMS 9902 as an interval timer or for three other serial interface conditions (see the *TMS 9902 Data Manual*). Two programmable timers, therefore, are available on board.

I/O

The serial I/O and the parallel I/O are handled over the dedicated I/O bus of the SBP 9900A, the communications register unit (CRU). Table 2 lists the address assignments within the dedicated 4K CRU address space. The TMS 9902 acts as the controller for this asynchronous serial interface. The character length, baud rate (75 to 38,400), and parity and stop bits are programmable. Three optional types of interface are supported:

- EIA
- 20 mA neutral current loop TTY
- Private wire differential line driver/receiver.

The TM 990/110M board is delivered complete with a 25-pin RS-232 type female connector, and is jumper selectable to support EIA or TTY operation. The differential line driver is normally unpopulated (see *Options*). Also, the TMS 9903 synchronous communications controller can be utilized since the TMS 9902/9903 are socket compatible.

The parallel I/O is handled by the SBP 9900/SBP 9961 pair; 16 parallel lines are all interfaced to the top edge connector which has 40 pins on 0.100 inch centers. Additionally, eight parallel lines are interfaced to the bus connectors. All I/O lines are equipped with pullup resistors.

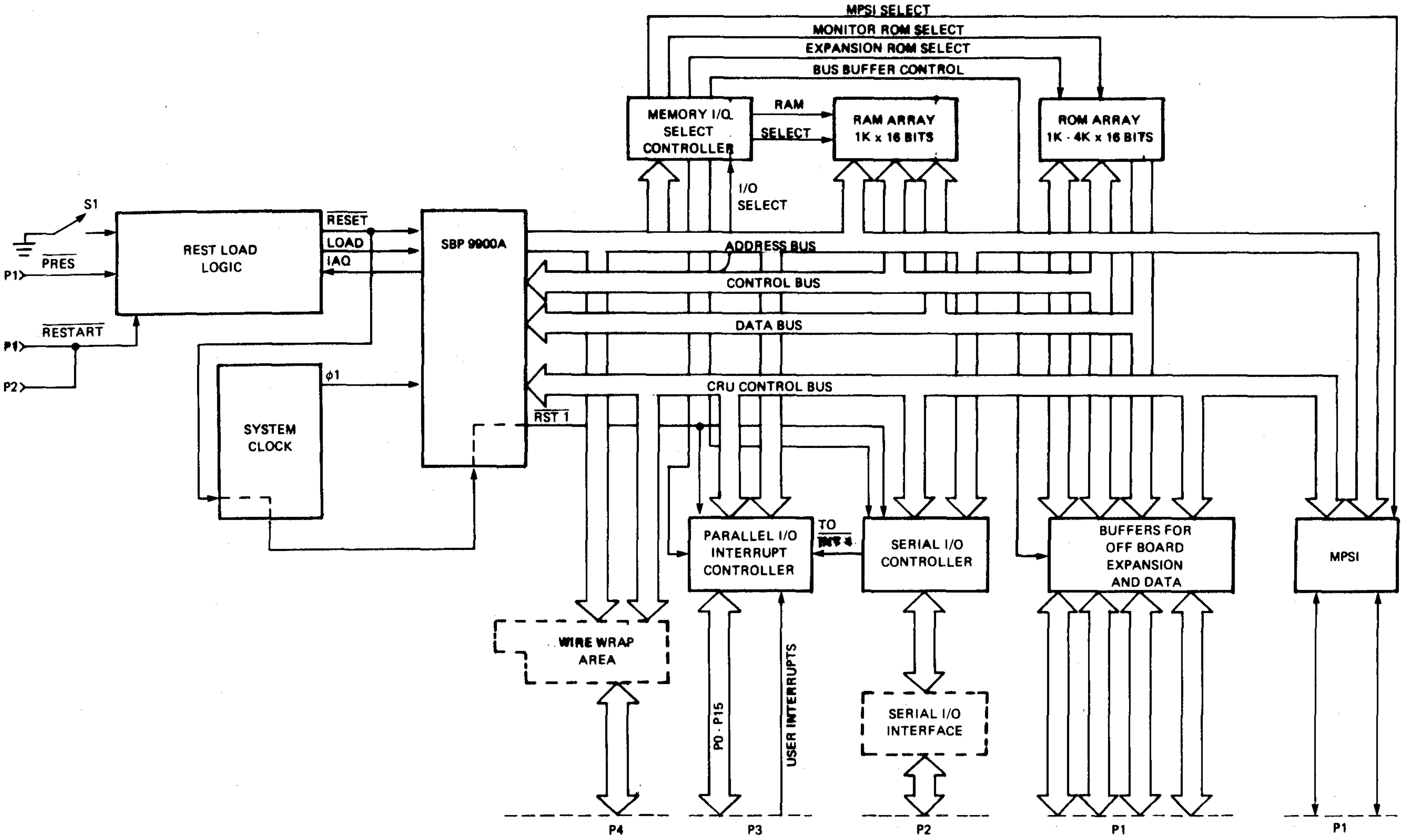


Figure F-1. Memory Address Map

TABLE 1. INTERRUPTS

INTERRUPT FUNCTION	LEVEL
0	Reset or PRES
1	External Device
2	External Device
3	Clock or External
4	Serial Int. or Ext.
5-15	External Devices
Load	Restart

TABLE 2. CRU ADDRESS MAP

BASE ADDRESS (REGISTER 12)	CRU BIT NUMBER		FUNCTION
0080_{16}	40_{16}	$5F_{16}$	On-Board Serial I/O Port (TMS 9902)
0100_{16}	80_{16}	$9F_{16}$	On-Board 16 I/O Parallel Interface, Interrupt Status Register, Interrupt Mask Register, and Interval Timer (TMS 9901)
0000_{16}	00_{16}	$3F_{16}$	Reserved for On-Card Expansion
$00C0_{16}$	60_{16}	$7F_{16}$	
0140_{16}	$A0_{16}$	FF_{16}	
200_{16}	100_{16}	FFF_{16}	Off-Board CRU

TIBUG

The TIBUG monitor TM 990/401-1 is normally supplied preprogrammed in the populated TMS 2708 EPROM's (see *Options*). Its operation is described in the *User's Manual* or the TM 990 Series literature.

Options

The TM 990/110M-1 board is equipped with two TMS 2708's preprogrammed with the TIBUG monitor, and the serial I/O is jumper selectable as EIA port or a TTY interface. The TM 990/110M-2 board is populated with two blank EPROM's, and a private wire differential line driver interface is populated instead of the TTY interface. Other software or accessories, such as the line by line assembler and the microterminal, may be ordered under separate part numbers.

Specifications

CPU: SBP 9900A

Instruction Set — 69 instructions
8, 16, or 32-bit operation
3 MHz System Clock

Interrupts: 16 levels — 15 may be external

Interval Timers: Two (in SBP 9961 and TMS 9902)

	Resolution	Maximum Interval
SBP 9961	21.3 μsec	349 msec
TMS 9902	64 μsec	16.32 msec

Memory: 16-bit word configuration

On-board EPROM/ROM, 1K words, expandable to 4K
On-board RAM, 1024 words
Off-board expansion to full 32K words

I/O

Parallel: 16 lines (expandable to 4K total I/O using CPU)

Serial: Asynchronous Controller TMS 9902
5-8 bit character
Programmable data rate, stop bits, parity

Baud Rates (bps):		
	75	2400
	110	4800
	150	9600
	300	19,200
	600	38,400
	1200	

Interfaces

Bus:	Data and Address	Three-state TTL compatible buffered output
	Control	TTL compatible
Parallel I/O and Interrupts:		TTL compatible
Serial I/O:		RS-232, 20 mA current loop or differential line driver

Software

TIBUG monitor self-contained in EPROM

Mating Connectors

Bus	100 pin, 0.125 in.	TI	H321150 (wire wrap)
		TI	H322150 (solder tail)
Parallel I/O	40 pin, 0.100 in.	TI	H311120 (wire wrap)
		3M	3464-0001 (no ears)

APPENDIX G

DYNAMIC MEMORY

Memory applications in conditioned environments requiring large bit storage can use 4K or 16K dynamic memories for low cost, low power consumption, and high bit density. SBP 9900A systems requiring 4K words or more of RAM can economically use the 4096-bit TMS 4050, the 16,384-bit TMS 4070, or any of the dynamic RAMs shown in Table G-1.

G.1 REFRESH

Dynamic RAMs must be refreshed periodically to avoid loss of stored data. The RAM data cells are organized into a matrix of rows and columns with on-chip gating to select the addressed bit. Refresh of the 4K RAM cell matrix is accomplished by performing a memory cycle at each of the 64 row addresses at least every two milliseconds. The 16K RAM has 128 row addresses. Performing a memory cycle at any cell on a row refreshes all cells in the row, thus allowing the use of arbitrary column addresses during refresh.

G.2 REFRESH MODES

There are several dynamic memory refresh techniques which can be used for an SBP 9900A system. If the system accesses at least one cell of each row every two milliseconds, then no additional refresh circuitry is required. A CRT controller which refreshes the display periodically is an example of such a system.

Refresh control logic must be included, however, in many systems since the system cannot otherwise ensure that all rows are refreshed every two milliseconds.

The dynamic memory in such SBP 9900A systems can be refreshed in either the block, cycle-stealing, or transparent mode.

TABLE G-1: DYNAMIC RAMS

DEVICE	ORGANIZATION	PACKAGE
TMS 4030	4096 × 1	22-PIN
TMS 4050	4096 × 1	18-PIN
TMS 4051	4096 × 1	18-PIN
TMS 4060	4096 × 1	22-PIN
TMS 4070	16,384 × 1	16-PIN

G.2.1 BLOCK REFRESH. The block mode of refresh halts the SBP 9900A every two milliseconds and sequentially refreshes each of the rows. The block technique halts execution for a 128 (4K) or 256 (16K) clock cycle period every two milliseconds. Some SBP 9900A systems cannot use this technique because of the possible slow response to priority interrupts or because of the effect of the delay during critical timing of I/O routines.

G.2.2 CYCLE STEALING. The cycle stealing mode of refresh ''steals'' a cycle from the system periodically to refresh one row. The refresh interval is determined by the maximum refresh time and the number of rows to be refreshed. The 4K dynamic RAMs have 64 rows to be refreshed every two milliseconds and thus require a maximum cycle stealing interval of 31.2 microseconds.

A refresh cycle may occur whenever the refresh circuitry is requesting a refresh cycle and when the CPU is not addressing memory. Since the refresh cycle is not synchronized to the CPU the refresh cycle (2 clock cycles) may overlap into a CPU initiated memory access cycle. The refresh controller must make memory not ready by negating the READY signal, thus ''stealing'' a cycle from the CPU. Even if this interference occurs during each refresh cycle, a maximum of 64 clock cycles are ''stolen'' for refresh every two milliseconds for a 4K dynamic memory refresh. Note that during a refresh cycle, the SBP 9900A should be externally isolated from the memory bus via 3-state buffers.

G.2.3 TRANSPARENT REFRESH. The transparent refresh mode synchronizes the refresh cycle to the SBP 9900A memory cycle. The rising edge of $\overline{\text{MEMEN}}$ marks the end of a memory cycle immediately preceding a non-memory cycle. The $\overline{\text{MEMEN}}$ rising edge can initiate a refresh cycle with no interference with memory cycles. The refresh requirement does not interfere with the system throughput since only non-memory cycles are used for the refresh cycles. The worst case SBP 9900A instruction execution sequence (all divides) will guarantee the complete refresh of a 4K or 16K dynamic RAM within two milliseconds. Note that during a refresh cycle, the SBP 9900A should be externally isolated from the memory bus via 3-state buffers.

While the transparent refresh mode eliminates degradation of system performance, the system power consumption can be higher since the RAMs are refreshed more often than required. As many as one-half of the SBP 9900As machine cycles can be refresh cycles, resulting in multiple refresh cycles for each row during the refresh interval. This situation can be corrected by adding a timer to determine the start of the refresh interval and an overflow detector for the refresh row counter. When every row has been refreshed during an interval, the refresh circuit is disabled until the beginning of the next interval. Since each row is refreshed only once, the system power consumption is reduced to a minimum.

Direct memory access using $\overline{\text{HOLD}}$ should guarantee that sufficient non-memory cycles are available for refresh during large block transfers. An additional refresh timer can be used to block HOLDA in order to provide periodic refresh cycles.

G.2.4 REFRESH CONTROLLER FOR TMS 4051 4K × 1 DYNAMIC RAM. A refresh controller for the TMS 4051 4K dynamic RAM is shown in Figure G-1. This refresh controller operates in a mode which is similar to the cycle stealing refresh mode except that the refresh controller is synchronized to the SBP 9900A through the use of the $\overline{\text{MEMEN}}$ and $\overline{\text{CYCEND}}$ signals thus allowing transparent refresh at a fixed interval. The $\overline{\text{REGNT}}$ signal does not become true unless the SBP 9900A is in the first clock period of an ALU cycle. Therefore completion of the refresh cycle before the SBP 9900A initiates a memory cycle is guaranteed. Since refresh occurs only when REFREQ becomes active, not at the completion of each memory cycle, this mode is dissimilar to the transparent mode discussed in paragraph G.2.3. However, this mode allows for all of the advantages of transparent mode refresh with the added feature of reduced power consumption since multiple row refresh during a refresh interval has been eliminated.

Figure G-2 shows the timing for the controller in Figure G-1. Note that refresh is granted ($\overline{\text{REGNT}}$) whenever $\overline{\text{MEMEN}}$, $\overline{\text{CYCEND}}$, and REFREQ are all high. This condition occurs only during the first clock period of an ALU or CRU cycle. The chip enable timing (CE) is per TMS 4051 specifications and is initiated for both SBP 9900A memory access and refresh cycles. The dynamic memory row address is the output of a modulo-64 counter (SN54LS393). The counter is incremented each refresh cycle in order to refresh the rows sequentially.

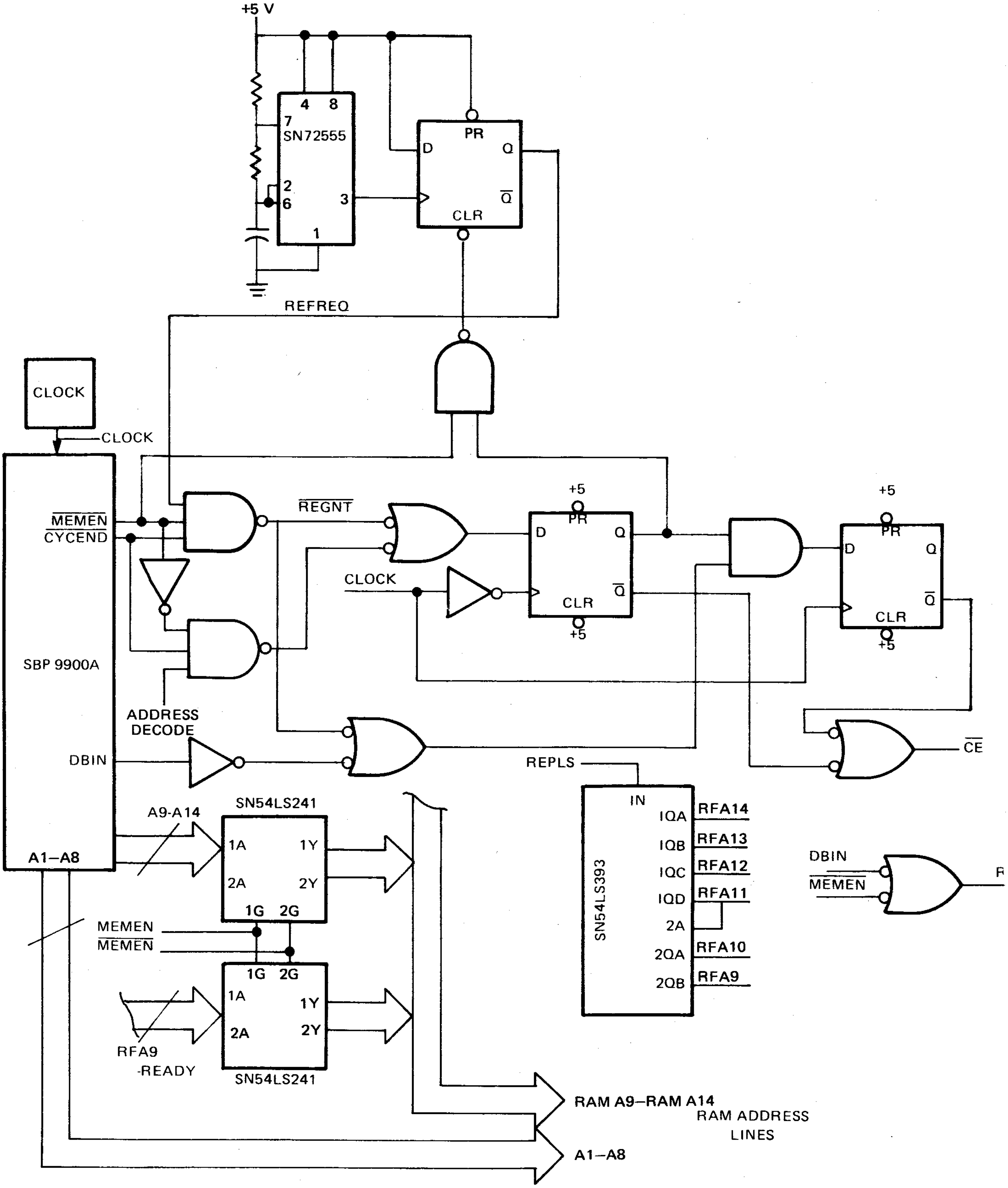

+5 V
4
8
7
SN72555
2
6
3
1
PR
D
Q
$\overline{Q}$
CLR
REFREQ
CLOCK
CLOCK
$\overline{MEMEN}$
$\overline{CYCEND}$
SBP 9900A
$\overline{REGNT}$
+5
PR
D
Q
$\overline{Q}$
CLR
+5
CLOCK
+5
PR
D
Q
$\overline{Q}$
CLR
+5
ADDRESS
DECODE
DBIN
$\overline{CE}$
REPLS
IN
SN54LS393
IQA
IQB
IQC
IQD
2A
2QA
2QB
RFA14
RFA13
RFA12
RFA11
RFA10
RFA9
A9-A14
A1–A8
SN54LS241
1A
2A
1Y
2Y
1G
2G
MEMEN
$\overline{MEMEN}$
1G
2G
1A
2A
1Y
2Y
RFA9
-READY
SN54LS241
DBIN
$\overline{MEMEN}$
R
RAM A9—RAM A14
RAM ADDRESS
LINES
A1—A8

Figure G-1

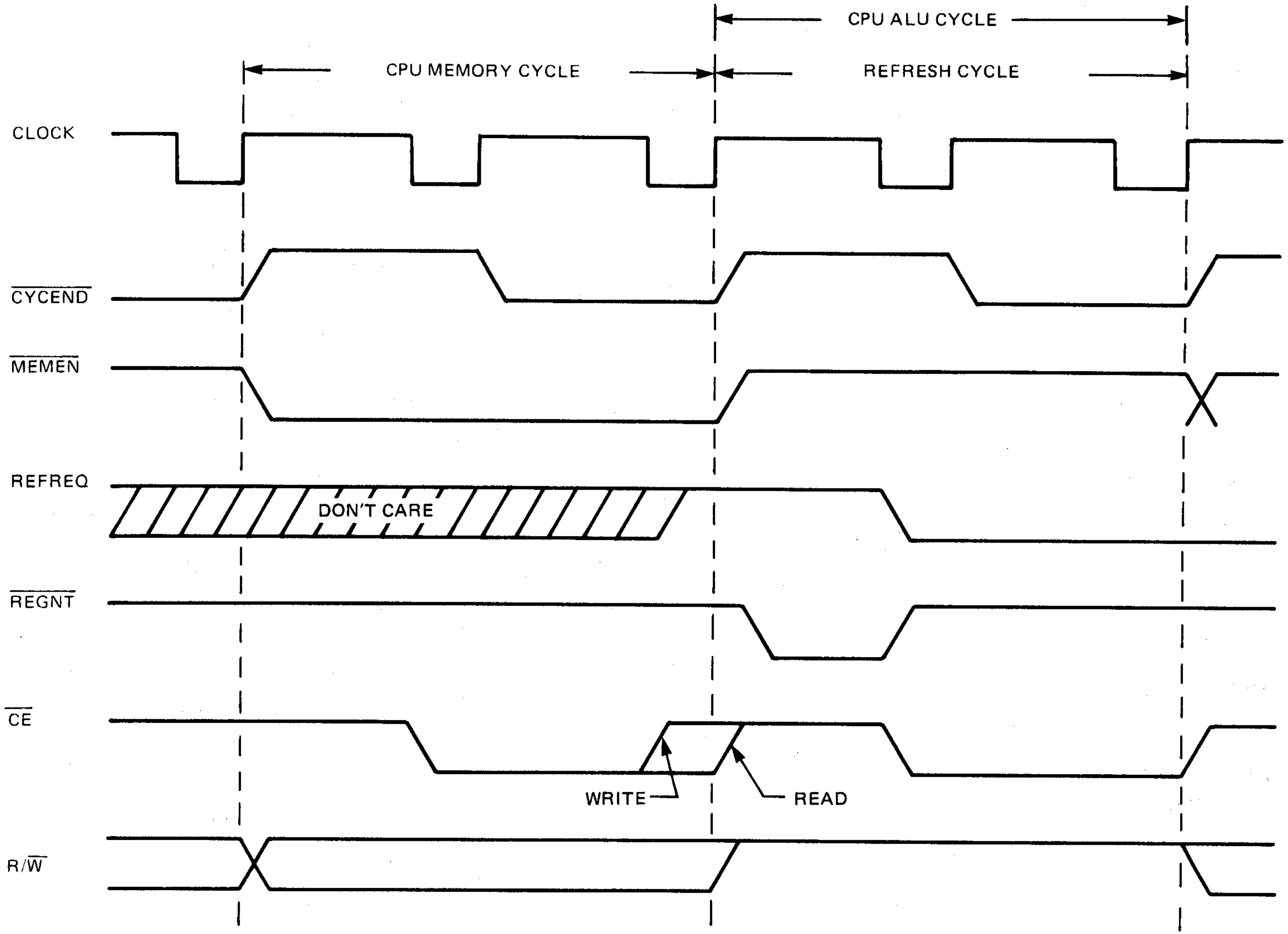

CPU ALU CYCLE
CPU MEMORY CYCLE
REFRESH CYCLE
CLOCK
$\overline{\text{CYCEND}}$
$\overline{\text{MEMEN}}$
REFREQ
DON'T CARE
$\overline{\text{REGNT}}$
$\overline{\text{CE}}$
WRITE
READ
R/$\overline{\text{W}}$

Figure G-2

APPENDIX H

DATA SHEETS

LINEAR INTEGRATED CIRCUITS

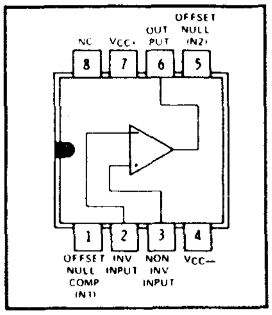
TYPES TL061, TL061A, TL061B, TL062, TL062A, TL062B, TL064, TL064A, TL064B, TL066, TL066A, TL066B LOW-POWER JFET-INPUT OPERATIONAL AMPLIFIERS

SEPTEMBER 1977

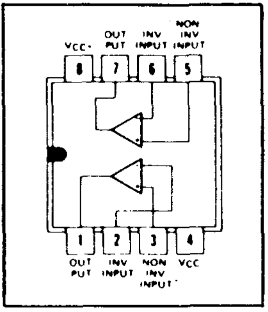
24 DEVICES COVER COMMERCIAL, INDUSTRIAL, AND MILITARY TEMPERATURE RANGES

- Very Low Power Consumption
- Typical Supply Current . . . 150 μA
- Wide Common-Mode and Differential Voltage Ranges
- Low Input Bias and Offset Currents
- Output Short-Circuit Protection
- High Input Impedance . . . JFET-Input Stage
- Internal Frequency Compensation
- Latch-Up-Free Operation
- High Slew Rate . . . 3.5 V/μs Typ

TL061, TL061A, TL061B
JG OR P DUAL-IN-LINE
PACKAGE (TOP VIEW)



TL062, TL062A, TL062B
JG OR P DUAL-IN-LINE
PACKAGE (TOP VIEW)

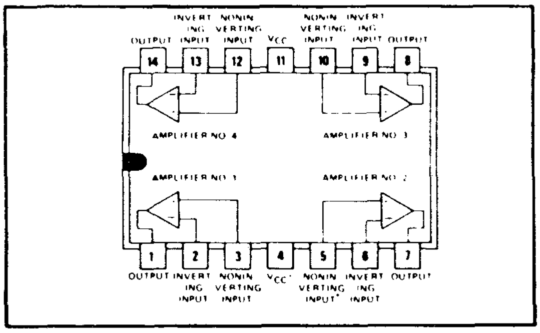


description

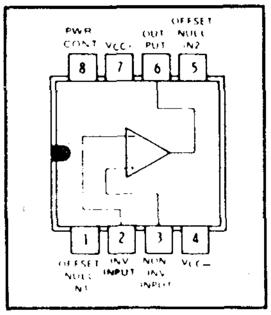
The JFET-input operational amplifiers of the TL061 series are designed as low-power versions of the TL081 series amplifiers. They feature high input impedance, wide bandwidth, high slew rate, and low input offset and bias currents. The TL061 series feature the same terminal assignments as the TL071 and TL081 series, except the TL066 has an additional power control function (pin 8) that allows the user to control the power dissipation of the circuit. This is achieved by connecting a resistor between the power-control terminal and V_{CC-}. For normal operation, the power-control terminal is connected directly to V_{CC-}.

Device types with an "M" suffix are characterized for operation over the full military temperature range of −55°C to 125°C, those with an "I" suffix are characterized for operation from −25°C to 85°C, and those with a "C" suffix are characterized for operation from 0°C to 70°C.

TL064, TL064A, TL064B
J OR N DUAL-IN-LINE
PACKAGE (TOP VIEW)



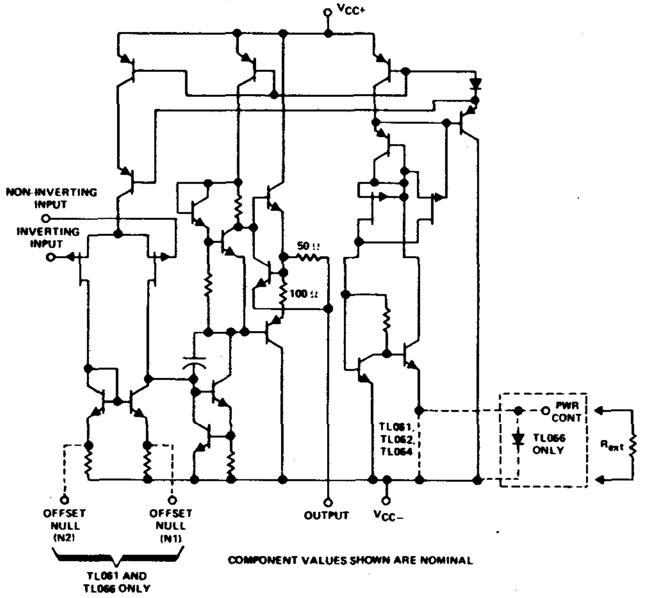
TL066, TL066A, TL066B
JG OR P DUAL-IN-LINE
PACKAGE (TOP VIEW)



NC—No internal connection

TYPES TL061, TL061A, TL061B, TL062, TL062A, TL062B, TL064, TL064A, TL064B, TL066, TL066A, TL066B LOW-POWER JFET-INPUT OPERATIONAL AMPLIFIERS

schematic (each amplifier)



COMPONENT VALUES SHOWN ARE NOMINAL

DISSIPATION DERATING TABLE

PACKAGE	POWER RATING	DERATING FACTOR	ABOVE T_A
J	680 mW	8 2 mW/ C	67 C
JG	680 mW	6 6 mW/ C	47 C
N	680 mW	9 2 mW/ C	76 C
P	680 mW	8 0 mW/ C	65 C

For TL066 only

$$I_{CC} \approx \frac{2.85}{2.85 + R_{ext}} \cdot I_{CC(0)}$$

where R_{ext} is in kΩ
and $I_{CC(0)} = I_{CC}$ with $R_{ext} = 0$

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

		TL06_M TL06_AM	TL06_I	TL06_C TL06_AC TL06_BC	UNIT
Supply voltage, $V_{CC}+$ (see Note 1)		18	18	18	V
Supply voltage, $V_{CC}-$ (see Note 1)		−18	−18	−18	V
Differential input voltage (see Note 2)		±30	±30	±30	V
Input voltage (see Notes 1 and 3)		±15	±15	±15	V
Voltage at power-control terminal of TL066 with respect to V_{CC-}		0.5	0.5	0.5	V
Duration of output short circuit (see Note 4)		Unlimited	Unlimited	Unlimited	
Continuous total dissipation at (or below) 25°C free-air temperature (see Note 5)		680	680	680	mW
Operating free-air temperature range		−55 to 125	−25 to 85	0 to 70	C
Storage temperature range		−65 to 150	−65 to 150	−65 to 150	C
Lead temperature 1/16 inch from case for 60 seconds	J or JG Package	300	300	300	C
Lead temperature 1/16 inch from case for 10 seconds	N or P Package	260	260	260	C

NOTES: 1. All voltage values, except differential voltages, are with respect to the zero reference level (ground) of the supply voltages where the zero reference level is the midpoint between V_{CC+} and V_{CC-}.
2. Differential voltages are at the noninverting input terminal with respect to the inverting input terminal.
3. The magnitude of the input voltage must never exceed the magnitude of the supply voltage or 15 volts, whichever is less.
4. The output may be shorted to ground or to either supply. Temperature and/or supply voltages must be limited to ensure that the dissipation rating is not exceeded.
5. For operation above 25°C, free air temperature, refer to Dissipation Derating Table.

operating characteristics, $V_{CC\pm} = \pm 15$ V, $T_A = 25°C$

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
SR	Slew rate at unity gain	$V_I = 10$ V, $C_L = 100$ pF,	$R_L = 10$ kΩ, $A_{VD} = 1$		3.5		V/μs
t_r	Rise time	$V_I = 20$ mV,	$R_L = 10$ kΩ,		0.15		μs
	Overshoot factor	$C_L = 100$ pF,	$A_{VD} = 1$		10%		

electrical characteristics, $V_{CC\pm}$ = ±15 V

PARAMETER		TEST CONDITIONS†		TL06_M TL06_AM			TL06_ I			TL06_C TL06_AC TL06_BC			UNIT
				MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
V_{IO}	Input offset voltage	R_S = 50 Ω, T_A = 25°C	'61, '62, '64, '66		3	6		3	6		3.	15	mV
			'61A, '62A, '64A, '66A		2	3					3	6	
			'61B, '62B, '64B, '66B								2	3	
		R_S = 50 Ω, T_A = full range	'61, '62, '64, '66			9			9			20	
			'61A, '62A, '64A, '66A			5						7.5	
			'61B, '62B, '64B, '66B									5	
α_{VIO}	Temperature coefficient of input offset voltage	R_S = 50 Ω,	T_A = full range		10			10			10		μV/°C
I_{IO}	Input offset current‡	T_A = 25°C	'61, '62, '64, '66		5	100		5	100		5	200	pA
			'61A, '62A, '64A, '66A		5	100					5	100	
			'61B, '62B, '64B, '66B								5	100	
		T_A = full range	'61, '62, '64, '66			20			10			5	nA
			'61A, '62A, '64A, '66A			20						3	
			'61B, '62B, '64B, '66B									3	
I_{IB}	Input bias current‡	T_A = 25°C	'61, '62, '64, '66		30	200		30	200		30	400	pA
			'61A, '62A, '64A, '66A		30	200					30	200	
			'61B, '62B, '64B, '66B								30	200	
		T_A = full range	'61, '62, '64, '66			50			20			10	nA
			'61A, '62A, '64A, '66A			50						7	
			'61B, '62B, '64B, '66B									7	
V_{ICR}	Common-mode input voltage range	T_A = 25°C	'61, '62, '64, '66	±12			±12			±10			V
			'61A, '62A, '64A, '66A	±12						±12			
			'61B, '62B, '64B, '66B							±12			
V_{OPP}	Maximum peak-to-peak output voltage swing	T_A = 25°C,	R_L = 10 kΩ	20	26		20	26		20	26		V
		T_A = full range,	R_L ≥ 10 kΩ	20			20			20			
A_{VD}	Large-signal differential voltage amplification	R_L ≥ 10 kΩ, V_O = ±10 V, T_A = 25°C	'61, '62, '64, '66	4	10		4	10		3	10		V/mV
			'61A, '62A, '64A, '66A	4	10					4	10		
			'61B, '62B, '64B, '66B							4	10		
		R_L ≥ 10 kΩ, V_O = ±10 V, T_A = full range	'61, '62, '64, '66	4			4			3			
			'61A, '62A, '64A, '66A	4						4			
			'61B, '62B, '64B, '66B							4			
B_1	Unity-gain bandwidth	T_A = 25°C			1			1			1		MHz
r_i	Input resistance	T_A = 25°C			10^{12}			10^{12}			10^{12}		Ω
CMRR	Common-mode rejection ratio	R_S ≤ 10 kΩ, T_A = 25°C	'61, '62, '64, '66	80	86		80	86		70	76		dB
			'61A, '62A, '64A, '66A	80	86					80	86		
			'61B, '62B, '64B, '66B							80	86		
k_{SVR}	Supply voltage rejection ratio ($\Delta V_{CC\pm}/\Delta V_{IO}$)	R_S ≤ 10 kΩ, T_A = 25°C	'61, '62, '64, '66	80	95		80	95		70	95		dB
			'61A, '62A, '64A, '66A	80	95					80	95		
			'61B, '62B, '64B, '66B							80	95		
P_D	Total power dissipation (each amplifier)	No load, T_A = 25°C,	No signal, See Note 6		4.5	6		4.5	6		4.5	7.5	mW
I_{CC}	Supply current (per amplifier)	No load, T_A = 25°C	No signal, See Note 6		150	200		150	200		150	250	μA
V_{o1}/V_{o2}	Channel separation	A_{VD} = 100,	T_A = 25°C		120			120			120		dB

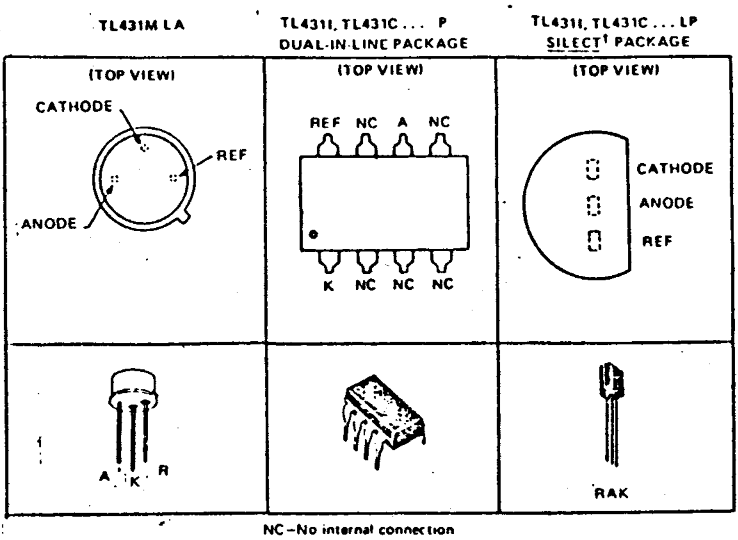
† All characteristics are specified under open-loop conditions unless otherwise noted. Full range for T_A is −55°C to 125°C for TL06_M and TL06_AM; −25°C to 85°C for TL06_I; and 0°C to 70°C for TL06_C, TL06_AC, and TL06_BC.

‡ Input bias currents of a FET-input operational amplifier are normal junction reverse currents, which are temperature sensitive. Pulse techniques must be used that will maintain the junction temperature as close to the ambient temperature as is possible.

NOTE 6: For the TL066 supply current is measured with the power-control terminal (pin 8) connected to V_{CC-}.

TL431C, TL431I, TL431M ADJUSTABLE PRECISION SHUNT REGULATORS

- Temperature-Compensated for Operation Over the Full Rated Operating Temperature Range
- Programmable Output Voltage
- Low Dynamic Output Impedance
- Fast Turn-On/Turn-Off Response
- Sink Current Capability from 1 mA to 100 mA
- Low Output Noise



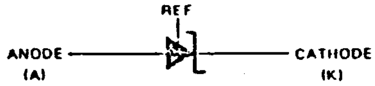
NC–No internal connection

description

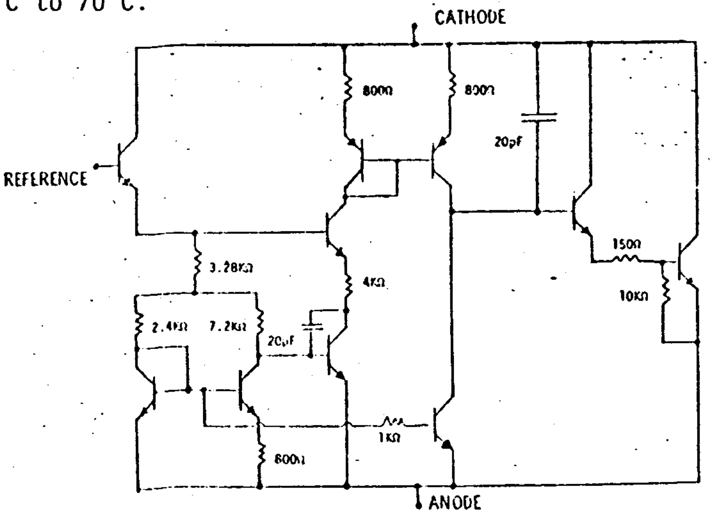
The TL431 is a three terminal adjustable regulator with guaranteed thermal stability over applicable temperature ranges. The output voltage may be set to any value between V_{REF} (≈ 2.5 V) and 36 V with two external resistors. The device has a 0.2 Ω typical dynamic output impedance. Active output circuitry provides a very sharp turn-on characteristic, making these devices excellent replacements for zener diodes in many applications.

The TL431M is characterized for operation from -55°C to 125°C, the TL431I from -40°C to 85°C, and the TL431C from 0°C to 70°C.

functional block diagram



schematic



absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Cathode voltage (see Note 1)		37 V
Continuous cathode current		-10 mA to 150 mA
Reference voltage		-5 V
Reference input current		10 mA
Continuous power dissipation at (or below) 25°C free-air temperature (see Note 2):	LA package	600 mW
	P package	1000 mW
	LP package	775 mW
Operating free-air temperature range:	TL431C	0°C to 70°C
	TL431I	-40°C to 85°C
	TL431M	-55°C to 125°C
Storage temperature range		-65°C to 150°C
Lead temperature 1/16 inch from case for 60 seconds:	LA package	300°C
Lead temperature 1/16 inch from case for 10 seconds:	LP or P package	260°C

recommended operating conditions

	MIN	MAX	UNIT
Cathode voltage, V_Z	V_{REF}	36	V
Cathode current, I_Z	1	100	mA

NOTES: 1. All voltage values are with respect to the anode terminal unless otherwise noted.

2. For operation above 25°C free-air temperature, refer to the Dissipation Derating Curves.

electrical characteristics at 25°C free-air temperature (unless otherwise noted)

PARAMETER		TEST FIGURE	TEST CONDITIONS		TL431M MIN	TL431M TYP	TL431M MAX	TL431I MIN	TL431I TYP	TL431I MAX	TL431C MIN	TL431C TYP	TL431C MAX	UNIT
V_{REF}	Reference voltage	1	$V_Z = V_{REF}$, $I_{IN} = 10$ mA		2.415	2.485	2.555	2.415	2.485	2.555	2.415	2.485	2.555	V
V_{DEV}	Deviation of reference input voltage over temperature	1	$V_Z = V_{REF}$, $I_{IN} = 10$ mA, T_A = full range†, see Note 3			20	43		15	30		8	16	mV
αV_{REF}	Average temperature coefficient of reference voltage	1	$V_Z = V_{REF}$, $I_{IN} = 10$ mA, T_A = full range†, see Note 4			±50	±100		±50	±100		±50	±100	ppm/°C
ΔV_{REF}	Change in reference voltage	2	$I_Z = 10$mA	V_Z from V_{REF} to 10 V		10	20		10	20		10	20	mV
				V_Z from 10 V to 36 V		25	50		25	50		25	50	
I_{REF}	Reference input current	2	$R_1 = 10$kΩ, $R_2 = \infty$, $I_{IN} = 10$ mA	$T_A = 25$°C		2.5	5		2.5	5		2.5	5	µA
				T_A = full range†			10			10			8	
$I_{Z\ MIN}$	Minimum cathode current for regulation	1	$V_Z = V_{REF}$			0.4	1		0.4	1		0.4	1	mA
$I_{Z\ OFF}$	Off-state current	3	$V_Z = 36$ V, $V_{REF} = 0$ V			0.05	0.5		0.05	0.5		0.05	0.5	µA
r_Z	Dynamic output impedance	1	$V_Z = V_{REF}$, Frequency = 0 Hz, see Note 5			0.2	0.5		0.2	0.5		0.2	0.5	Ω

PARAMETER MEASUREMENT INFORMATION

dc test circuits

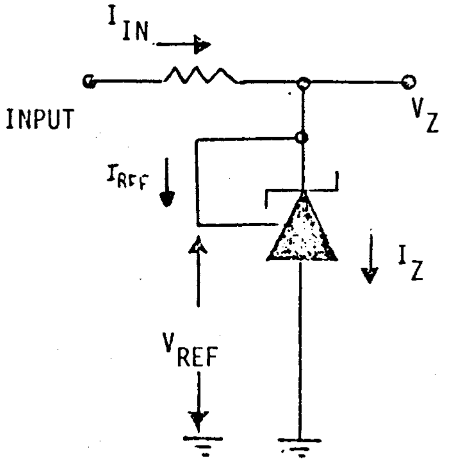


FIGURE 1 - TEST CIRCUIT FOR $V_Z = V_{REF}$

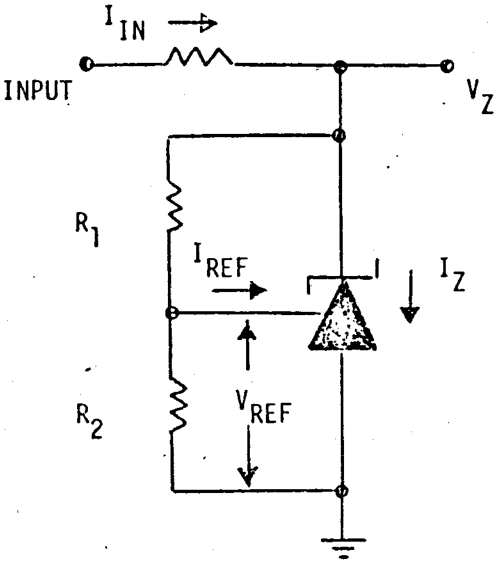


FIGURE 2 - TEST CIRCUIT FOR $V_Z > V_{REF}$

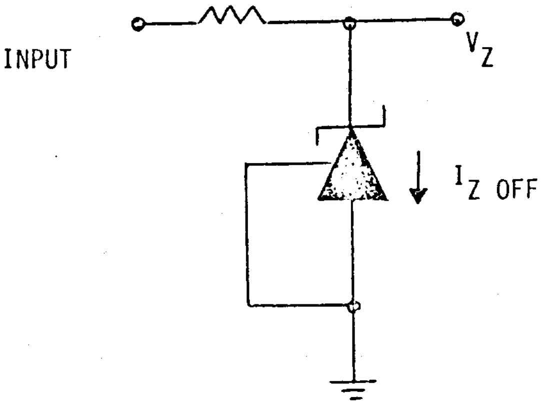
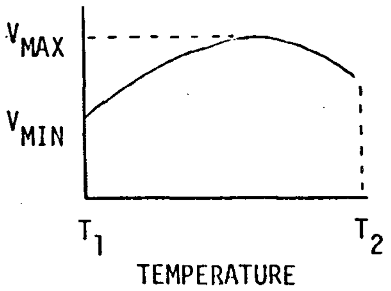


FIGURE 3 - TEST CIRCUIT FOR OFF-STATE CURRENT

+Full range is -55°C to 125°C for the TL431M, -40°C to 85°C for the TL431I, and 0°C to 70°C for the TL431C.

NOTES: 3. Deviation of reference input voltage, V_{DEV}, is defined as the maximum variation of the reference input voltage over the full temperature range.



$$V_{DEV} = V_{MAX} - V_{MIN}$$

4. The average temperature coefficient of the reference input voltage, αV_{REF}, is defined as:

$$\alpha V_{REF} = \pm \frac{\left[\dfrac{V_{MAX} - V_{MIN}}{V_{REF}(\text{at } 25^\circ\text{C})}\right] 10^6}{T_2 - T_1} \qquad \frac{\text{ppm}}{^\circ\text{C}}$$

where $T_2 - T_1$ = Full range temperature change.

αV_{REF} can be positive or negative depending on whether the slope is positive or negative.

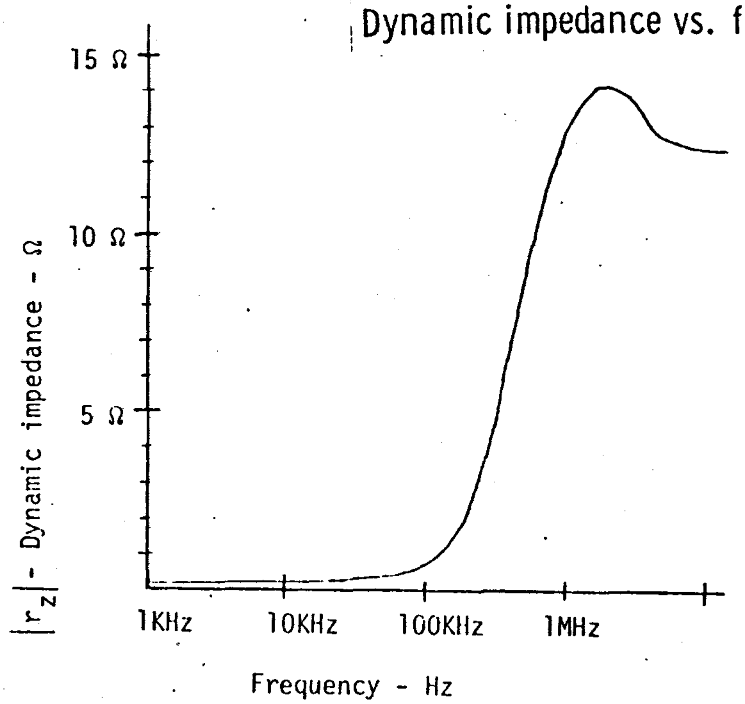
5. The dynamic output impedance, r_z, is defined as:

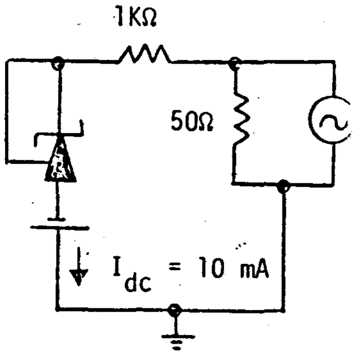
$$r_z = \frac{\Delta V_Z}{\Delta I_Z}$$

For $V_Z > V_{REF}$, the dynamic output impedance, r_z', is defined as:

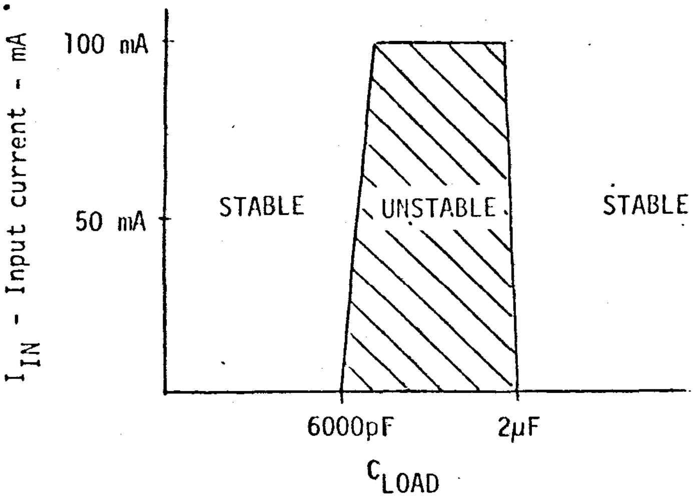
$$r_z' = \frac{\Delta V_Z'}{\Delta I_Z'} \approx r_z \left[1 + \frac{R_1}{R_2}\right]$$

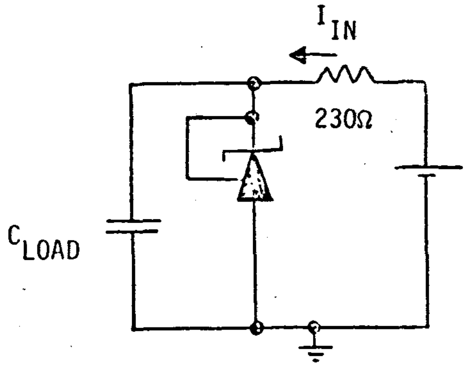
Dynamic impedance vs. frequency



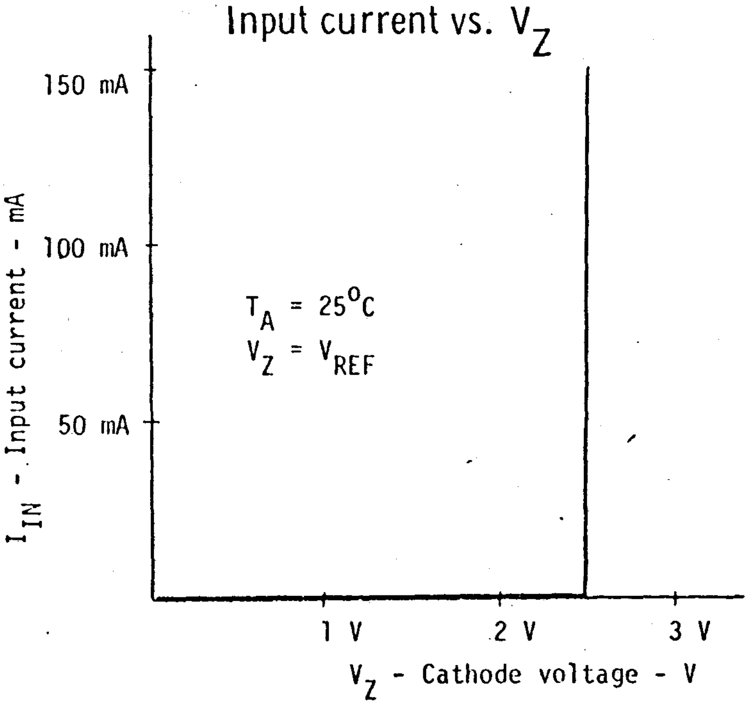


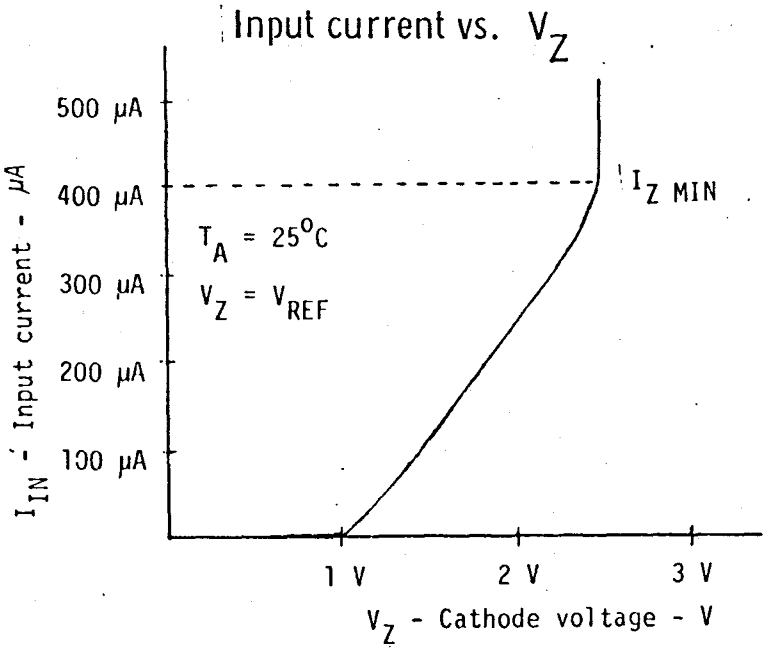
Range of instability



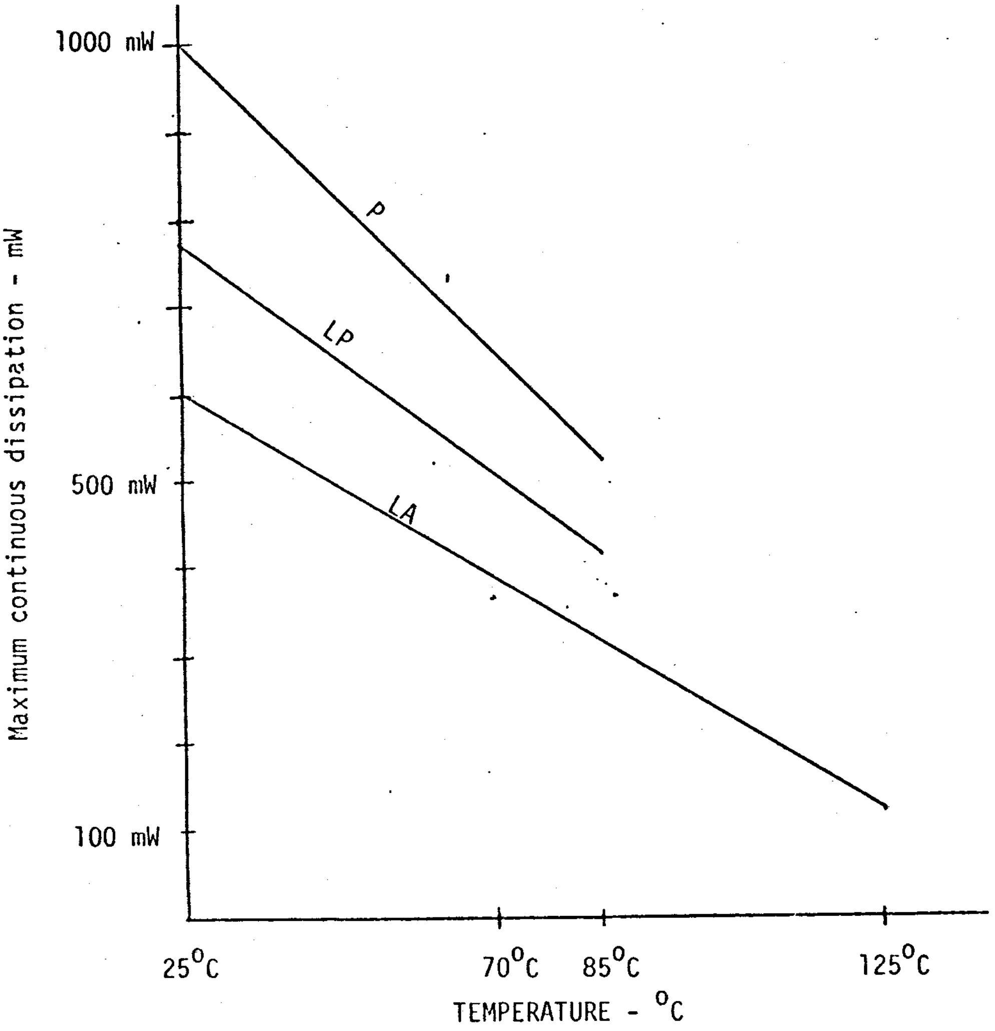


TYPICAL CHARACTERISTICS





THERMAL INFORMATION


1000 mW
500 mW
100 mW
Maximum continuous dissipation - mW
P
LP
LA
25°C
70°C
85°C
125°C
TEMPERATURE - °C