8X300 DESIGN GUIDE

signetics
a subsidiary of U.S. Philips Corporation

8X300 DESIGN GUIDE

December 1980

Bipolar LSI Division
SIGNETICS Corporation

DSPG Document No. 80-102

Preface

The Signetics 8X300 Design Guide is intended to supply the design engineer with a concentration of valuable and, in some cases, previously unpublished information relative to the implementation of systems using the Signetics 8X300 Microcontroller and its compatible devices.

The material contained within this handbook is arranged in such a manner as to provide the user with the following:

1. An overview of the 8X300 family as a state-of-the-art design tool.

2. Tutorial information relative to the operation and use of the Signetics 8X300 Microcontroller and its compatible devices.

Additional information necessary for the design of systems utilizing the Signetics 8X300 Microcontroller may be obtained from the following Signetics documents:

- Signetics 8X300 Programming Manual
- Signetics Microcontroller Cross Assembly Program Manual
- Signetics Data Sheets for specific devices to be used

CONTENTS

CONTENTS

LIST OF TABLES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont.)

1.0 8X300 FAMILY AS STATE-OF-THE-ART DESIGN TOOLS

1.1 INTRODUCTION

The Signetics 8X300 Microcontroller is a high-speed 8-bit bipolar microprocessor, fabricated using Schottky technology. The device is capable of executing (fetch, decode and execute) a 16-bit instruction in one machine cycle (a minimum of 250 nanoseconds), with a maximum power consumption of less than 2.5 watts. The 8X300 architecture and instruction set are designed to provide a high data throughput with bit-oriented, rather than word-oriented operations. This allows the designer a considerable degree of flexibility in control applications. Under program control, the microprocessor is able to select and test individual bits or groups of bits, make calculations and decisions based upon the selected data, and issue appropriate commands to the system.

Because of its high speed, the 8X300 is able to perform, through software, many operations that would otherwise have to be performed by additional hardware. The resultant lower parts count means less handling and inventory, reduced assembly and testing, lower power consumption and improved reliability.

For more complex applications, Signetics offers a broad range of 8X300 family and compatible devices. These devices are illustrated in Figure 1-1, and further information relative to them may be obtained through the offices listed in the back of this manual.

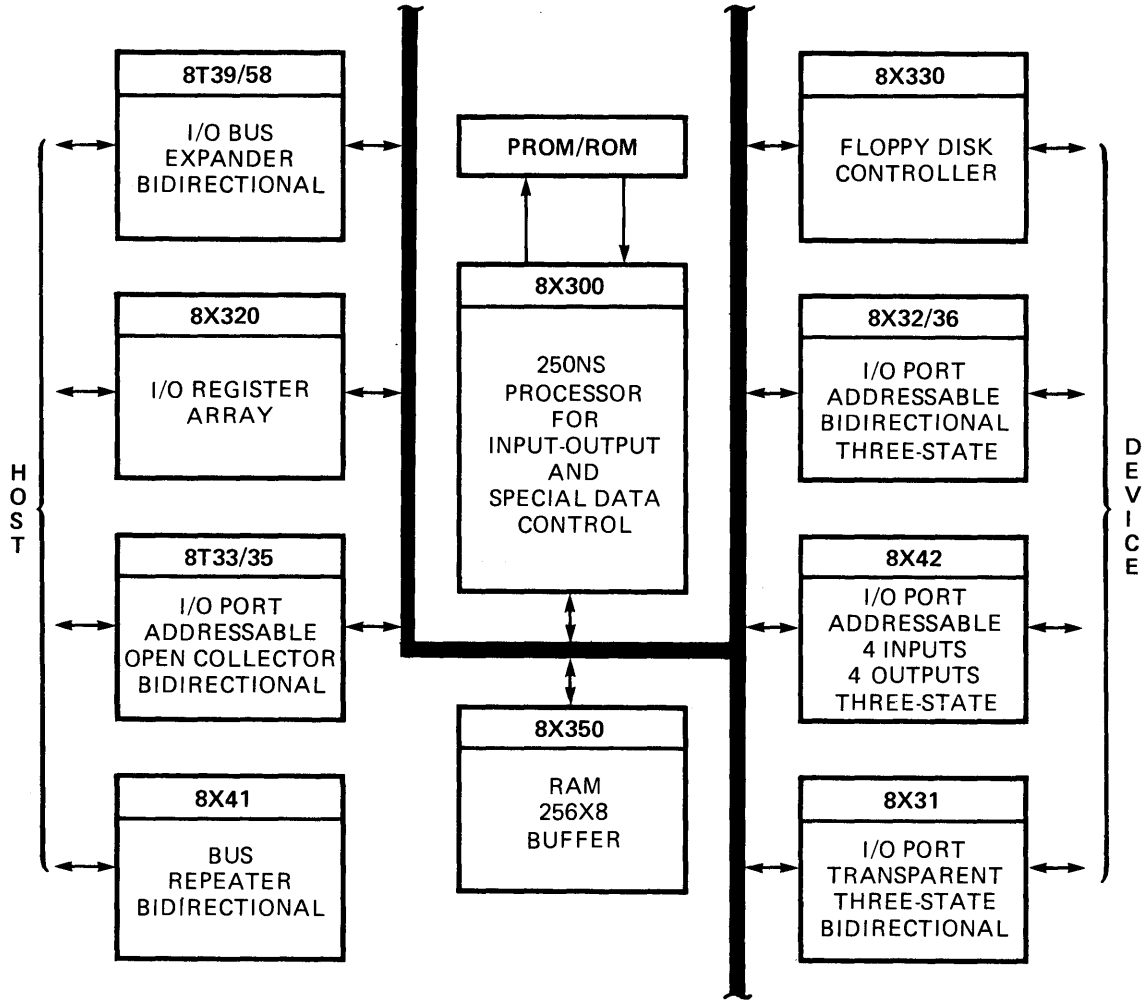


Figure 1-1. 8X300 Family and Compatible Devices

1.2 DESIGN ADVANTAGES OF THE SIGNETICS 8X300 MICROCONTROLLER

The Signetics 8X300 Microcontroller comprises three separate buses: a non-multiplexed 13-bit Instruction Address bus, a non-multiplexed 16-bit Instruction Data bus, and a multiplexed 3-state, 8-bit I/O bus. This dedicated bus structure contributes to the speed and flexibility of the device by allowing more actions to overlap. In brief, the 8X300 offers the designer a degree of speed and versatility previously available only with bit-slice architecture.

The 8X300 may select as its data source either the user I/O bus or one of eight internal read/write registers. Previous to performing the instructed function, the designer can manipulate source data using both rotate and mask functions. The results of the instructed function may then be shifted, masked and merged with the original source data prior to output. All of the above-mentioned operations can be totally executed in 250 nanoseconds, accounting for the designers ability to process 1- to 8-bit variables as easily as MOS microprocessors handle byte-oriented data.

Although the instruction set of the Signetics 8X300 Microcontroller contains only eight major catagories of instructions, a variable operand field within these instructions provides the designer with an extensive set of unique instructions. The inherent simplicity of the instruction set allows several hundred lines of machine code to be written by hand. Where more complex programs are required, a cross assembly program, MCCAP, is available. MCCAP allows the system designer to construct well-documented firmware readily employing all the powerful features of the 8X300 architecture.

The on-chip oscillator and timing generators of the device require only a crystal or single capacitor for reliable operation. If the Signetics 8X300 is to be operated from an external TTL clock, all that is required is an inverting buffer, a non-inverting buffer and two resistors. The device requires only a single plus five-volt power supply for its operation, and its inputs and outputs are TTL.

Figure 1-2 illustrates a typical system configuration.

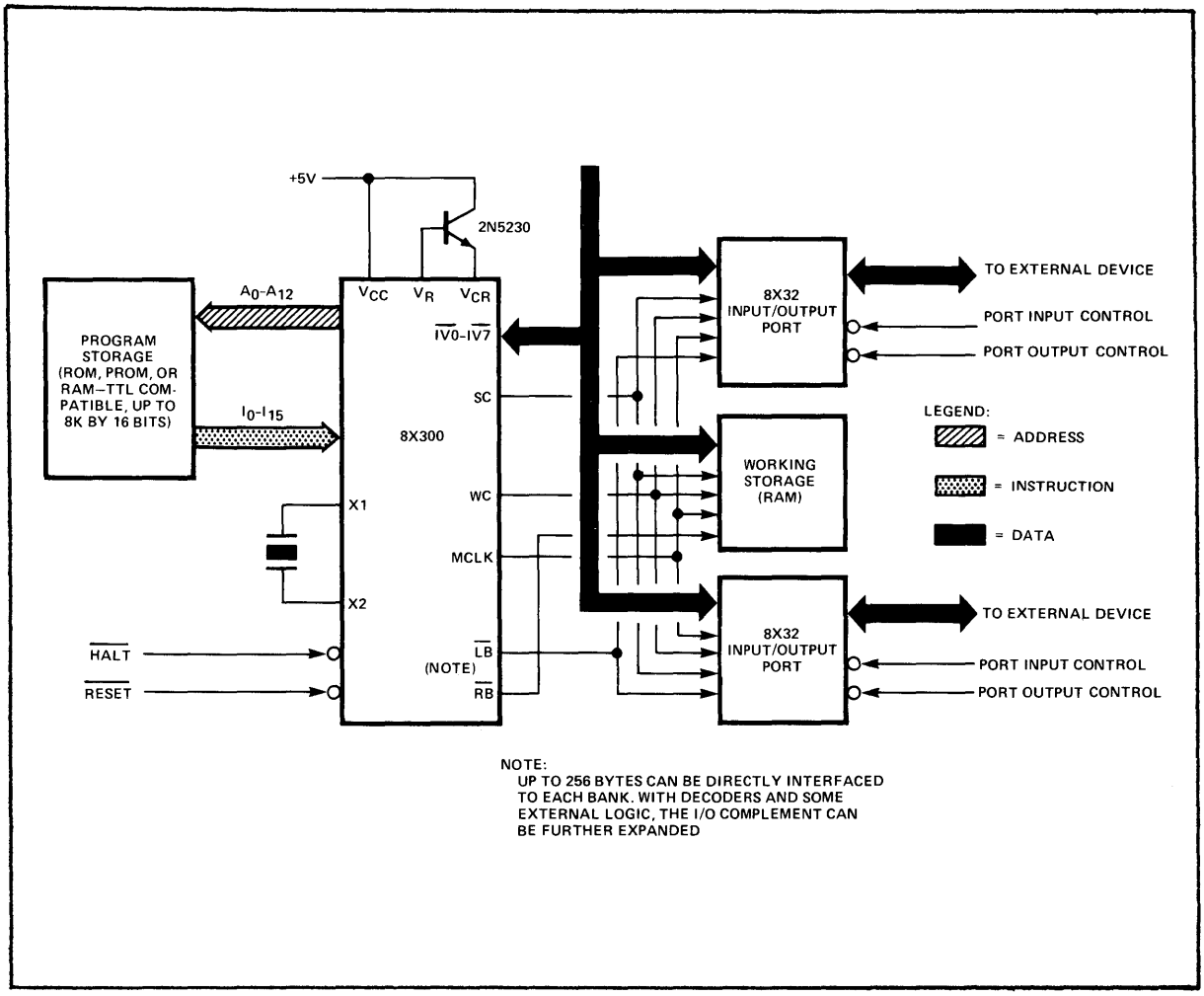


Figure 1-2. Typical System Block Diagram

1.3 DESIGN CONSIDERATIONS

The Signetics 8X300 Microcontroller moves data at 4 million operations per second, while requiring only minimal support and peripheral circuitry. Where speed is of prime importance, the 8X300 offers an economic alternative to random logic or the bit-slice approach. The device reduces both the complexity and cost usually associated with the above mentioned approaches. Anywhere rapid intelligent control is important -- disk drive controllers, CRT displays, data communications, industrial control -- the 8X300 is an excellent choice.

Signetics offers extensive support for the development of systems employing the 8X300. The designer will find aids available in the forms of:

- Development Systems
- Training
- Documentation
- Applications Support

In the area of development systems, Signetics offers an 8X300 Evaluation Kit and an 8X300 Microcontroller Cross Assembly Program (MCCAP).

Training aids include a general introductory movie on the 8X300 and the 8X300 Training Course, a self-paced audio-visual course.

A wide range of 8X300 Family documentation is available. In addition to application notes and data sheets for specific devices, Signetics offers the following manuals:

- The 8X300 Design Guide
- The 8X300 Programming Manual
- The 8X330 Floppy Disk Controller Manual
- The Microcontroller Cross Assembler Program Manual

During the design phase of a project, the Field Applications Engineer is an invaluable source of advice. They are available to offer on-site technical assistance, and are equipped technically to help analyze your application and transform it into a viable hardware/software system. The Field Applications Engineers are supported by the factory-based Product Applications Engineers, who are also available to answer specific technical questions regarding the use of the Signetics 8X300 and its support components.

Additional design tools which are under development and will be released in the near future include -

8X300 In-Circuit Emulator - A hardware development system that converts other development systems and computers into 8X300 development systems.

MDS MCCAP - A version of the 8X300 assembler tailored to execute on the MDS development system.

PDP-11 MCCAP - A version of the 8X300 assembler that executes on PDP-11 and LSI-11 computers.

8X300 Designers Kit - A new version of the current Evaluation Kit which includes writable program storage, an RS232 port, and an key/display panel.

2.0 THEORY OF OPERATION

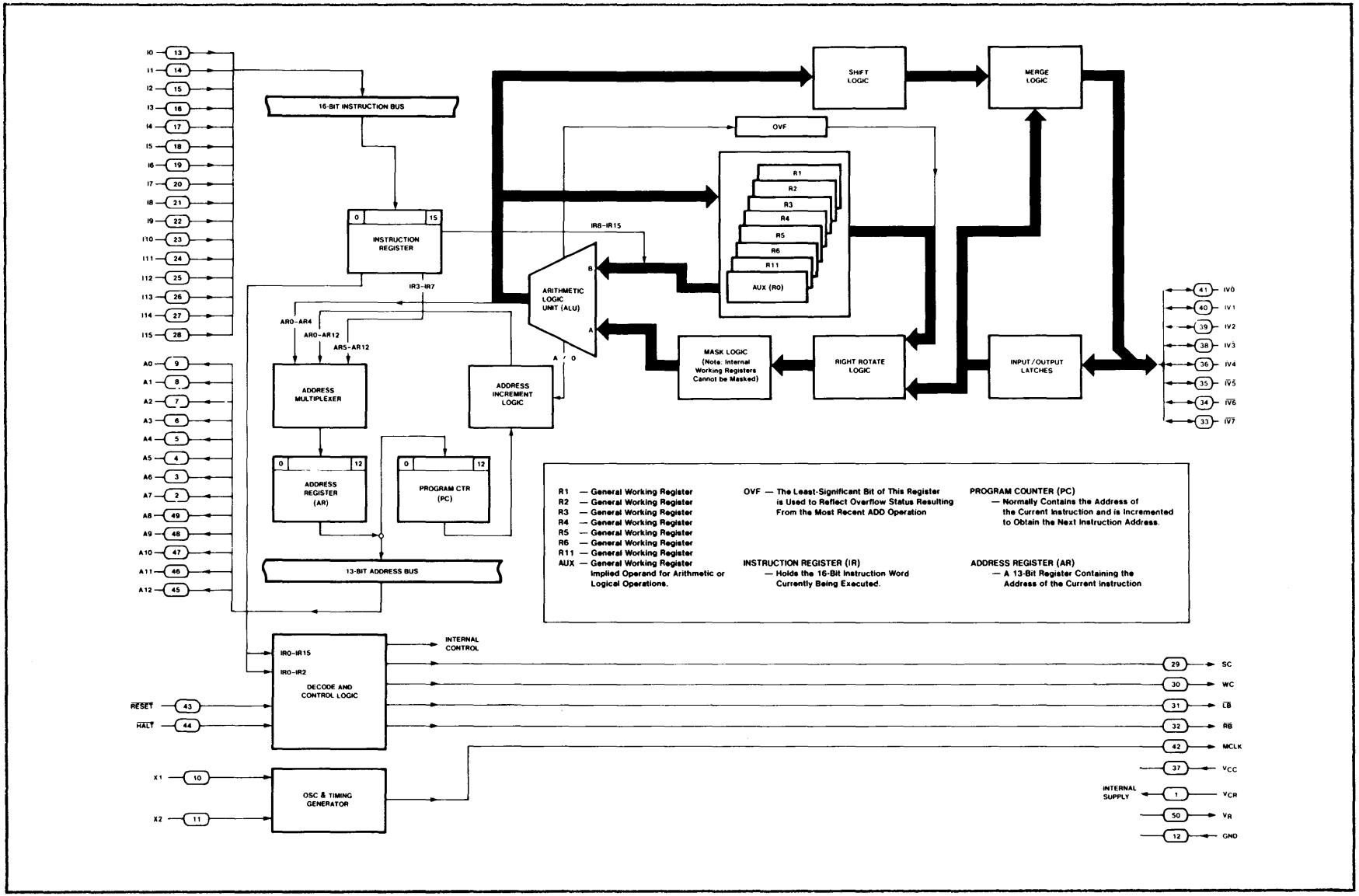


Figure 2-1. Architecture and Pin Designations for 8X300 Microcontroller

2.1 CPU ARCHITECTURE

Figure 2-1 is a simplified block diagram of the Signetics 8X300 illustrating the major internal functional blocks and data paths of the device. As is shown in the illustration, the device includes an Arithmetic Logic Unit (ALU), eight 8-bit working registers, logic for rotate and mask of data, logic for shifting and merging of data, decode and control logic, an address register, a program counter and an instruction register.

2.1.1 Decode and Control Logic

The decode and control logic contained in the Signetics 8X300 interprets the instruction according to its op code and operand fields, and directs the device to perform the specified operation. Additionally, the decode and control logic provides signals (Select Command, Write Command, Left Bank/Right Bank) to control logic external to the device. Such logic includes RAM and I/O interface to user equipment. As shown in Figure 2-2, the decode and control logic also receives control commands ($\overline{\text{RESET}}$ and $\overline{\text{HALT}}$) from logic external to the device.

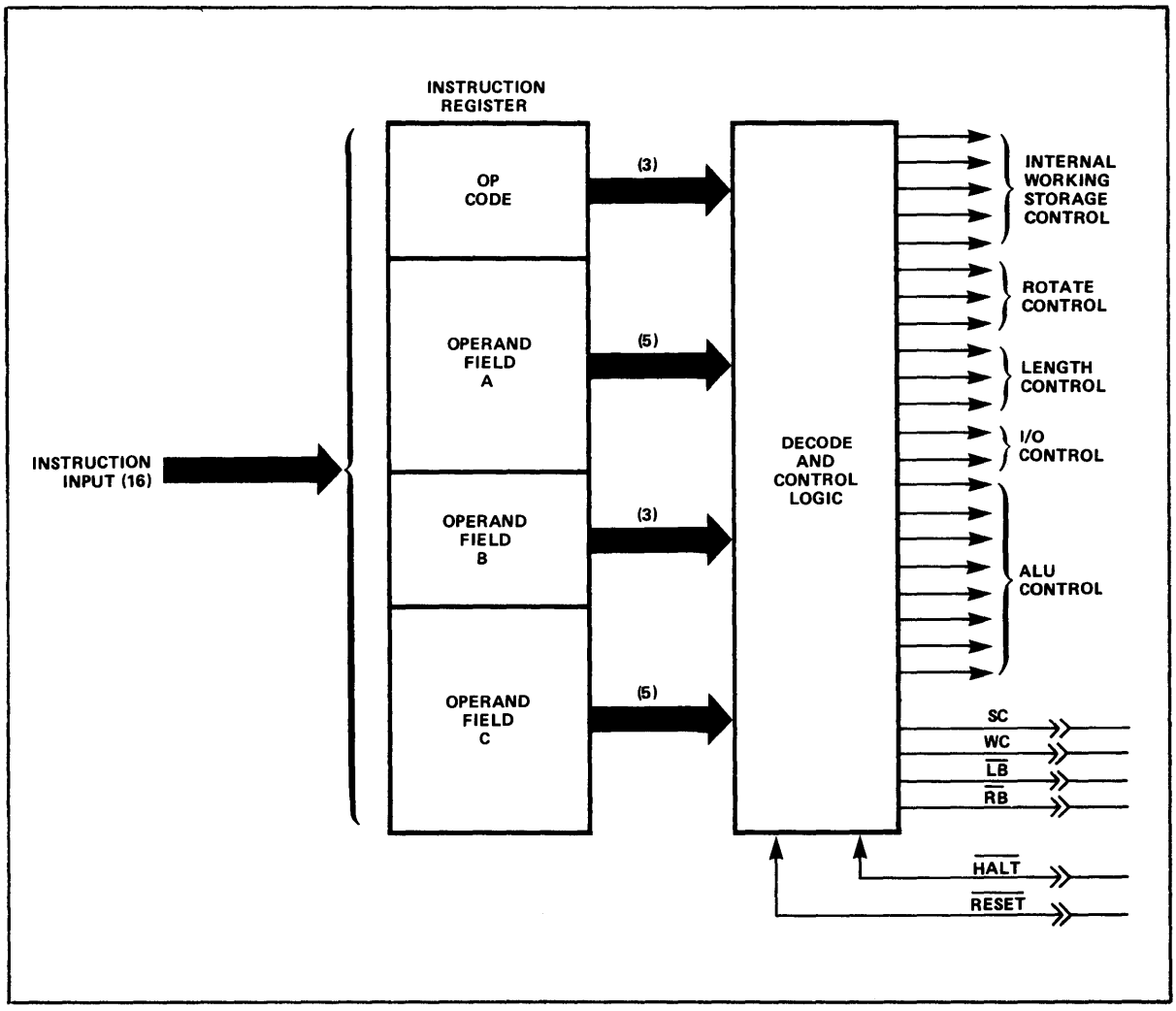


Figure 2-2. Decode and Control Inputs and Outputs

The function of the operand fields vary according to the instruction to be performed. Table 2-1 provides a summary of operand field functions in order of instruction called for. More detailed information on this subject is available in the Signetics 8X300 Programming Manual.

Table 2-1. Operand Field Functions

INSTRUCTIONS		OPERAND FIELD FUNCTION		
		A	B	C
		BITS 3–7	BITS 8–10	BITS 11–15
MOVE, ADD, AND, XOR	REGISTER TO REGISTER REGISTER TO IV BUS ADDRESS	SOURCE	ROTATE	DESTINATION
MOVE, ADD, AND, XOR	REGISTER TO IV BUS IV BUS TO REGISTER IV BUS TO IV BUS IV BUS TO IV BUS ADDRESS	SOURCE	LENGTH	DESTINATION
XEC, NZT	REGISTER	SOURCE	LOCAL ADDRESS	
XEC, NZT	IV BUS	SOURCE	LENGTH	LOCAL ADDRESS
XMIT	REGISTER IV BUS ADDRESS	DESTINATION	LITERAL CONSTANT	
XMIT	IV BUS	DESTINATION	LENGTH	LITERAL CONSTANT
JMP		ADDRESS		

2.1.2 Arithmetic Logic Unit

Referring to Figure 2-3, the Arithmetic Logic Unit (ALU) receives its A-bus inputs from internal registers or I/O (through Rotate/Mask logic) and its B-bus inputs from either the Instruction Register (IR8-IR15) or from the Accumulator (a register containing an exact duplicate of the information contained in the Auxiliary Register, RO) via a multiplexer. The ALU performs one of four functions on

the data. It may ADD,AND or XOR the A and B input data, or it may pass either input data straight through with no modification (MOVE,XMIT). The output data is then transferred to either the Internal Working Storage registers, the I/O Bus through Shift and Merge logic, or the Program Address logic. The ALU also indicates a carry-out condition during eight bit ADD functions by means of a discrete output to the LSB of the Overflow Register, R10. During NZT instructions the ALU tests for all bits "0" (A≠0).

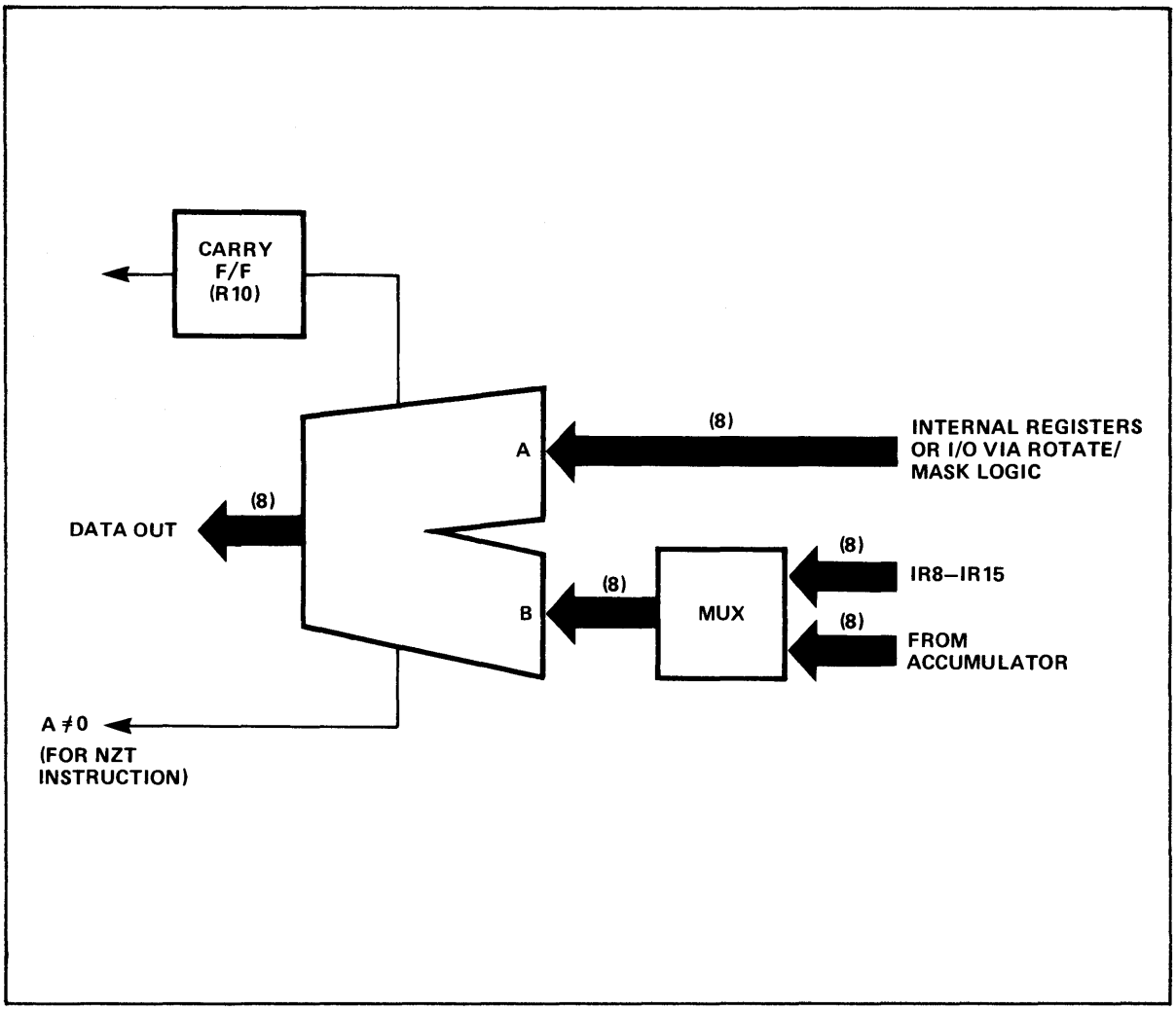


Figure 2-3. ALU Data Inputs and Outputs

2.1.3 Program Address Logic

The data flow within the Program address logic is controlled by the transitions of the internal system clock of the 8X300. Each instruction cycle comprises four of these clock transitions, referred to as quarter-cycles.

The Program Address logic data flow is shown in Figure 2-4. When operating with instructions other than Program Control instructions, the Program Counter data is incremented by the Increment logic during the third quarter-cycle. The incremented data then passes through the lower ports of Multiplexers A, B and C, and is loaded into the address Register. During the fourth quarter-cycle this new address is looped back and loaded into the Program Counter.

The above described data flow becomes altered when operating with Program Control instructions (XEC, NZT, JMP). In these cases, the lower five or eight bits of the address are received from the ALU via the upper ports of Multiplexers A and B. In the case of the JMP instruction, the upper five bits of the address are received from the instruction register through the upper ports of Multiplexer C. The JMP instruction is unconditional and is immediately loaded. In the case of the NZT instruction, the source is first tested for non-zero contents. If the non-zero condition ($A \neq 0$) exists the new address is loaded, otherwise normal program sequencing continues. The XEC instruction differs in that the new address in the Address Register is not looped back to the Program Counter. The result is that unless a JMP is encountered at the executed instruction, the program sequencing resumes operation at the address following the XEC instruction. That is, the program counter, which still contains the address of the XEC instruction, is incremented as normal.

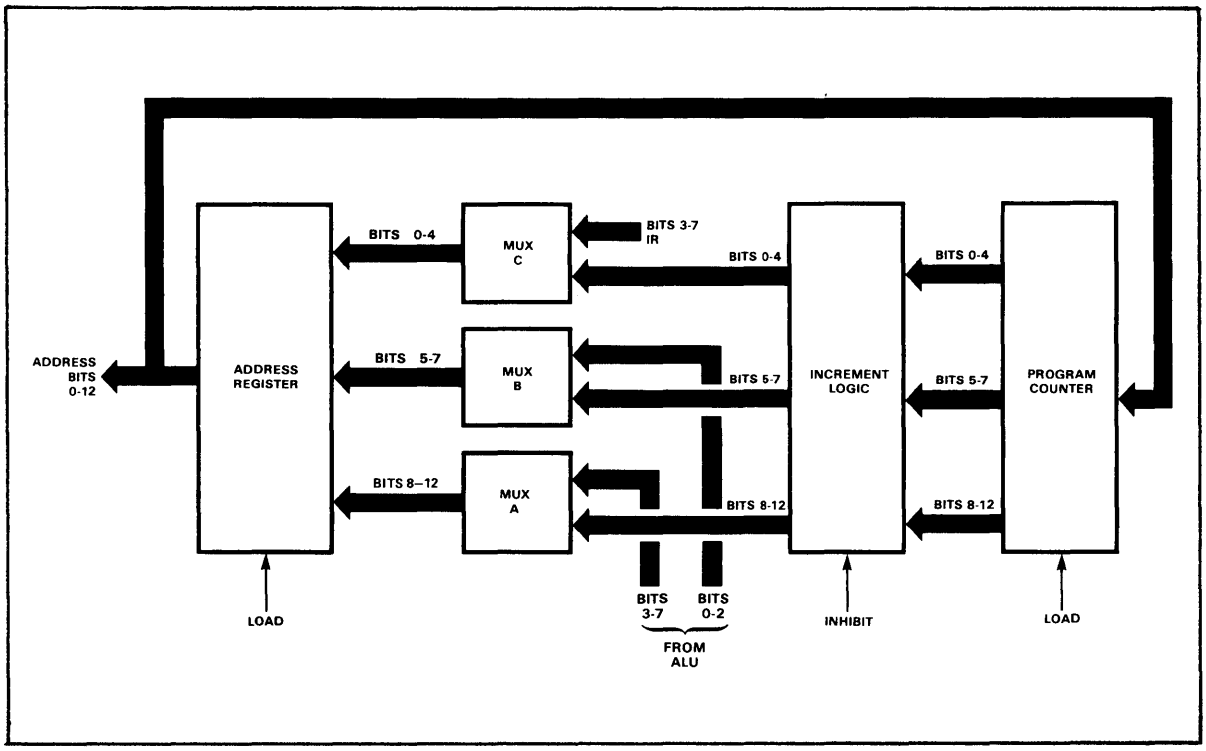


Figure 2-4. Program Address Logic Data Flow

2.1.4 Bus Structure and Control

The 8X300 bus structure, depicted in Figure 2-1, consists of dedicated Instruction and Instruction Address buses, and a multiplexed I/O bus. This bus structure allows direct implementation of a totally independent read only Program Memory of up to 8192 words by 16-bits. The 13-bit Program Control logic allows the order of program execution to be altered by instructions, either unconditionally or under conditions determined from selected data.

Interface with external working storage (RAM) and user equipment is through the I/O bus. This bus carries either address information or data, and is accompanied by two Data-I/O control lines (WC and SC) which indicate to external circuitry which of the two types of information is on the bus. Ordinarily the 8-bit I/O bus would only be capable of addressing a maximum of 156 bytes of memory and/orI/O registers, but through the use of two additional complementary control lines ($\overline{LB}$ and $\overline{RB}$) this capability has been expanded to a maximum of 512. This is accomplished by grouping the memory-I/O into two banks of 256 bytes and/or devices, and using one control line to select each. It should be noted that only one device per bank may be enabled at any given time. Timing for the external memory and I/O is by means of the MCLK signal from the Signetics 8X300. The five control lines function as follows:

1. Select Command (SC) - a high (binary 1) on this line indicates that an address is being output on the I/O bus.

2. Write Command (WC) - a high (binary 1) on this line indicates that data is being output on the I/O bus, to be stored in the previously enabled register/port.

   NOTE: When both SC and WC are low (binary 0) the Microcontroller expects data from the selected device. Both SC and WC high (binary 1) is a condition not generated by the Microcontroller.

3. Left Bank Select ($\overline{LB}$) - a low (binary 0) on this line enables one of two groups of I/O devices (or memory locations). In all following text, this group of Memory-I/O devices is referred to as the Left Bank.

4. Right Bank Select ($\overline{RB}$) - a low (binary 0) on this line enables the second of the previously mentioned two groups of I/O devices (or memory locations). In all following text, this group of Memory-I/O devices is referred to as the Right Bank.

   NOTE: $\overline{LB}$ and $\overline{RB}$ are complementary outputs of a single control line which can be considered as a ninth address bit. Each I/O device (port, memory, etc.) is connected to either the $\overline{LB}$ or $\overline{RB}$ control line.

5. Master Clock (MCLK) - an output signal of the Microcontroller used for clocking of I/O devices and/or synchronization of external logic. The I/O bus data must be stable during the period the MCLK is high.

Figure 2-5 illustrates a simple control system composed of the Signetics 8X300 and its compatible devices. The illustration depicts the wiring configuration of the Program Address, Instruction and I/O buses, and the control lines. In this particular application the Left Bank of devices is composed of I/O ports used for interface with user equipment. The Right Bank of devices comprise the system's external Working Storage. Through program control, the Microcontroller is able to input data from one bank of devices, manipulate this data and output it to the opposite bank of devices, all in one instruction cycle (see Figure 2-6). It should be noted thatthe I/O and Memory devices must have been addressed prior to this instruction cycle, and that an operation from one device to another of the same bank is not normally done during a single instruction cycle. It is a simple matter to input data from a specific device, operate on it and output it again to the same device during one instruction cycle.

2.1.5 Internal Working Storage

Internally, the Signetics 8X300 stores data in eight 8-bit read/write registers. These registers include registers R1 through R6, R11 and the Auxiliary register, R0. The Auxiliary register holds one of the operands used in two-operand instructions. As previously mentioned, an exact duplicate of the Auxiliary register data is also maintained in the Accumulator. This allows the Microcontroller to perform operations which use the Auxiliary register data as both operands. Register 10 stores the carry bit from addition operations in its LSB position. Because the LSB is the only bit used, R10 can only have one of two possible values - binary "1" or binary "0".

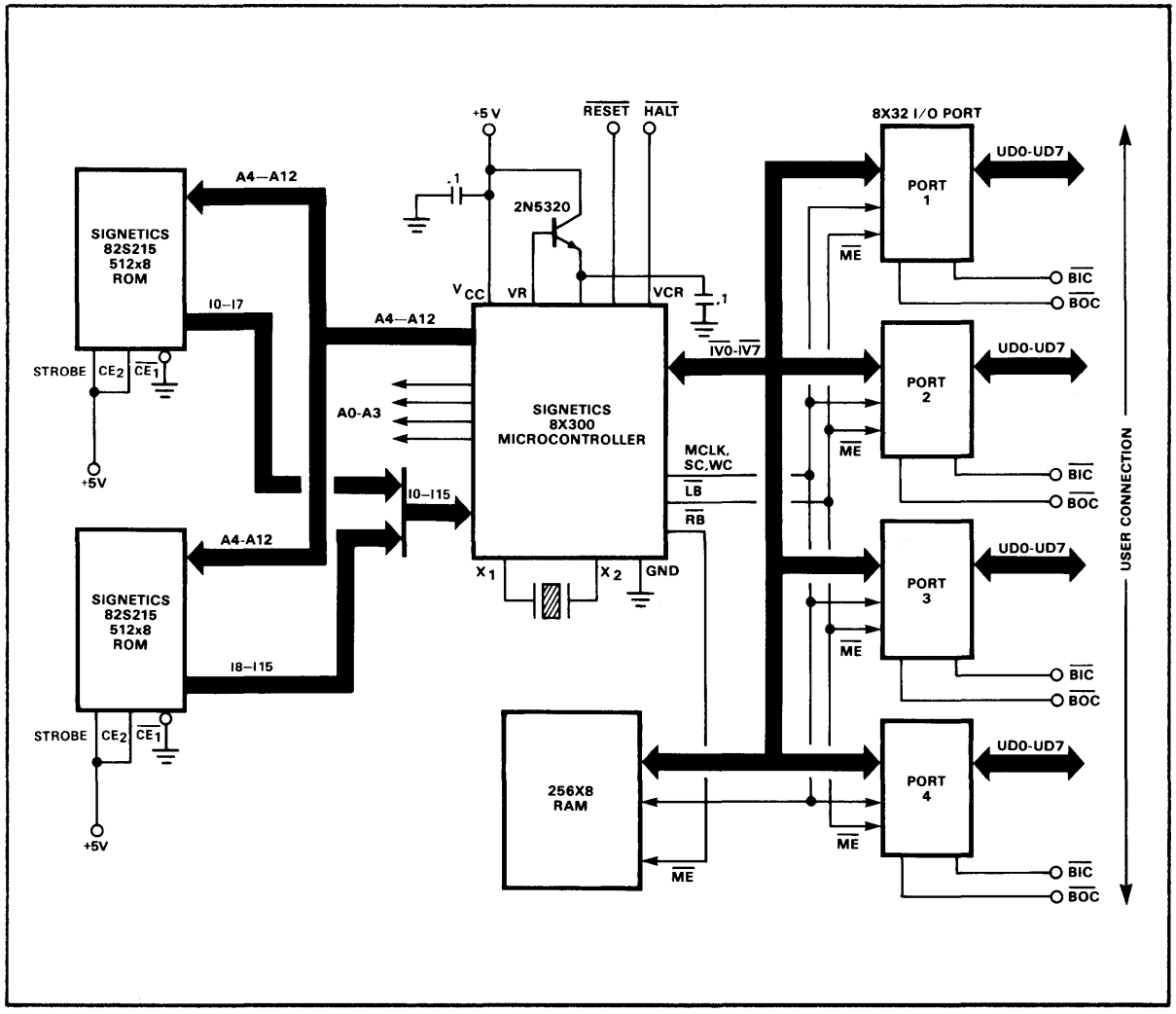


Figure 2-5. Simple Control System

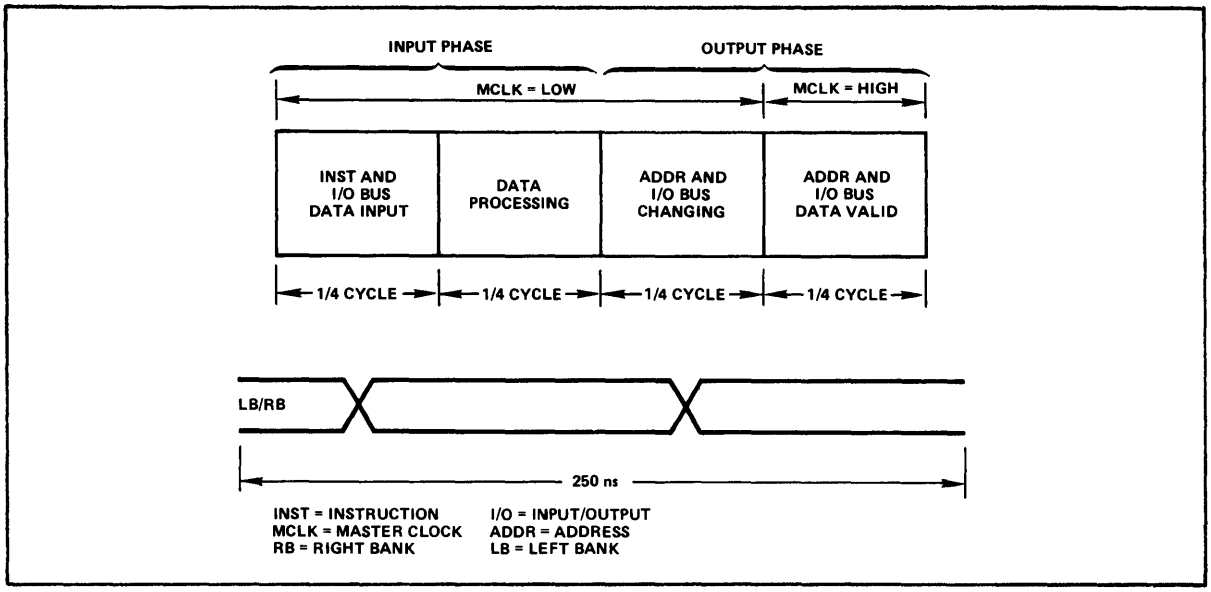


Figure 2-6. Instruction Cycle

2.1.6 Bit Manipulation Logic

The A input of the ALU is preceded by data-rotate and data-mask logic. The combination of right-rotate and mask functions allows selection of one or more bits from a source data field. For instructions where both the source and destination are internal registers, only the rotate function is available, the data being a fixed length of eight bits.

The right-rotate function provides an end-around-shift of one to seven places of the 8-bit source field. In this manner, the least significant bit of the bit string required can be positioned in the least significant position of the data byte, ready for further processing. See Figure 2-7.

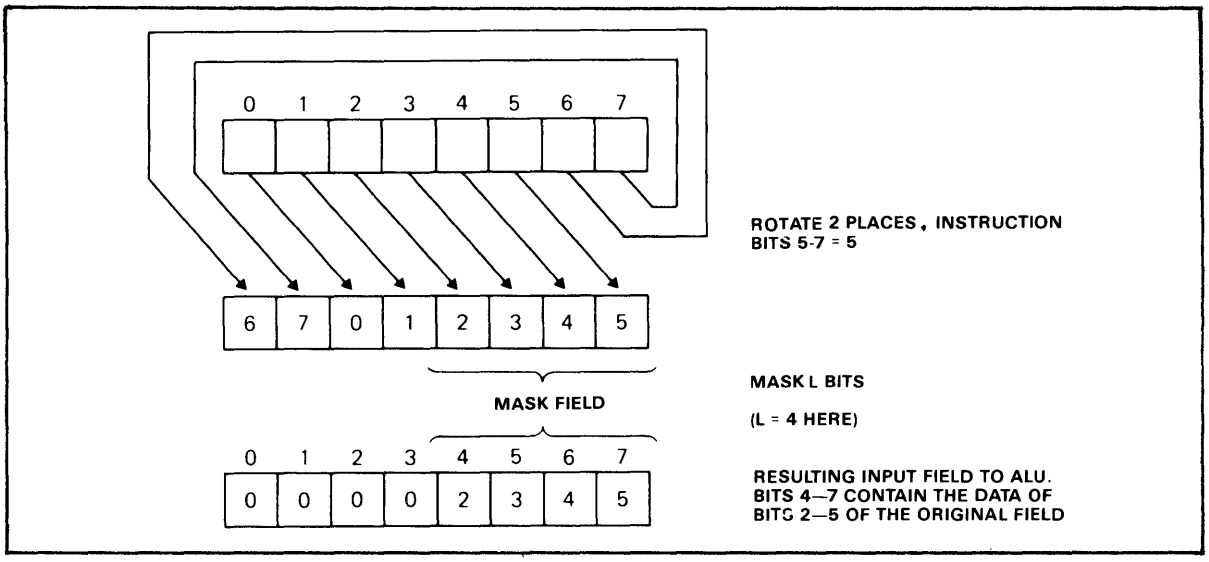


Figure 2-7. Mask Function

The number of places that the data is to be rotated is specified by the R operand field, when present, and by instruction bits 5, 6, and 7 when the source is the I/O bus. These bits specify the bit of the source data field which will be rotated to bit position seven before masking.

The mask function allows selection of the least significant L contiguous bits of the rotated I/O bus source data for subsequent processing. The value L is specified by the L operand field of the instruction. After masking, the least significant bits are output to the ALU, with the remaining bits of the byte set to zero.

The ALU output is followed by the shift and merge logic. These functions allow alteration of the state of a bit string within the I/O bus data byte. The action of the rotate and mask functions ensures that the required processed data is in the least significant bits of the ALU output; the left shift function then aligns the data in the required bit positions prior to merging. See Figure 2-8.

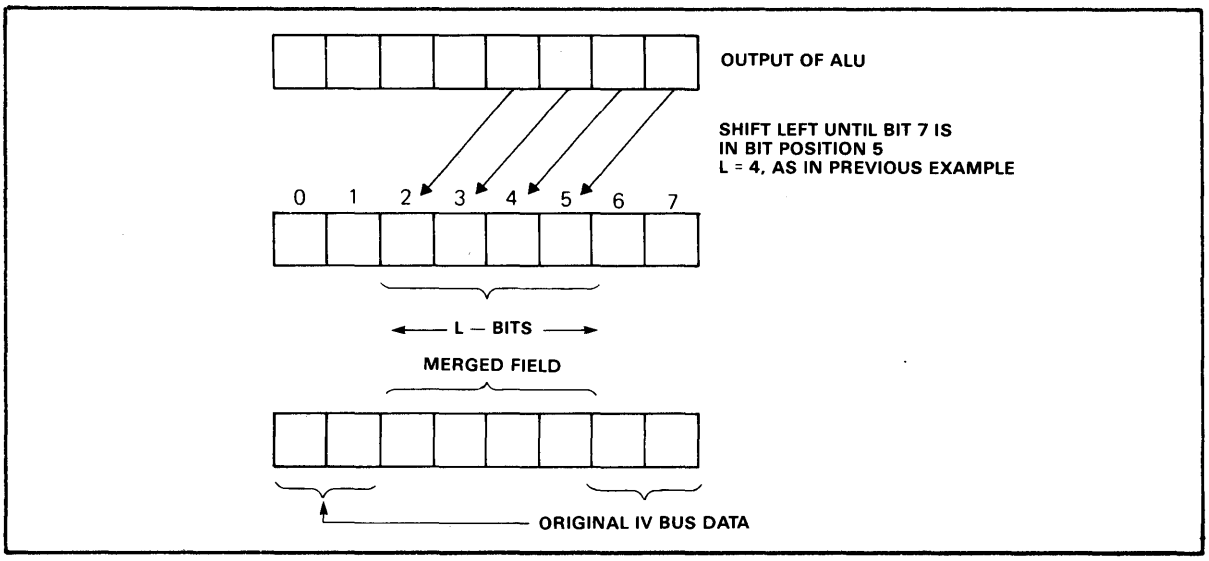


Figure 2-8. Shift and Merge Functions

Because the process is not an end-around-shift, data shifted from position 0, the MSB, is lost. The number of positions to be shifted is determined by instruction bits 13-15: the data is left shifted until the LSB has reached the bit position specified by instruction bits 13-15.

The merge function allows the user to update part of the existing I/O bus data without affecting the remaining parts of the data byte. The length of the bit string to be merged with the existing data is specified by the L operand field, the LSB of the bit string (after shifting) being specified by instruction bits 13-15.

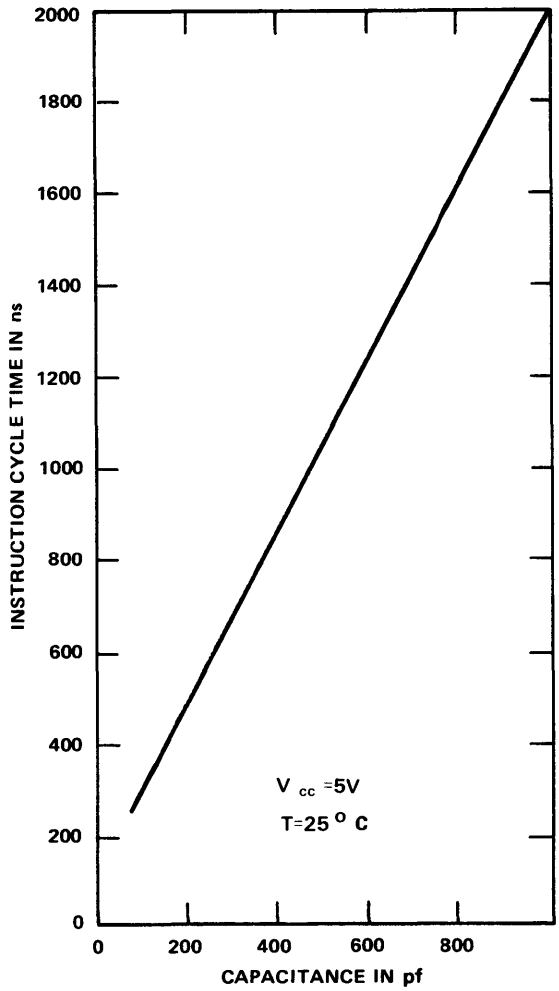
It must be kept in mind that the I/O bus is "read from" and "written to" only in 8-bit bytes. The eight bits of information is read from the I/O bus and latched into the internal I/O latches of the 8X300. Some number of bits may then be rotated, masked and shifted as desired. Prior to output to the I/O bus, the desired bits are then merged with the original information still being retained in the internal I/O latches. For this reason, a single bit cannot be read from one device and written to replace just one bit of another device in one instruction cycle.

2.1.7 Oscillator

The 8X300 contains an on-chip oscillator which generates all necessary clock signals. The oscillator output frequency may be controlled by any one of four methods:

1. External capacitor
2. External crystal
3. Pulse generator
4. TTL clock

The first control method, an external capacitor, is an imprecise way of controlling the speed of the 8X300, and its use should be restricted to low speed applications. In actual operation, the value of the capacitor may be affected by its environmental conditions. Approximate capacitor values are shown in Graph 2-1.



Graph 2-1. Typical Cycle Time Versus Capacitance

The second, and most desirable, method of oscillator control is implemented through the use of a frequency determining crystal. The oscillator is designed such that its gain decreases as its operating frequency increases. When used with a crystal of the proper characteristics, the oscillator has little or no tendency to operate in a harmonic mode. Because of this, no external components other than the crystal are required for frequency control. Table 2-2 lists the specifications of crystals to be used with the 8X300, and their resonant frequency f in Hertz is related to the desired instruction cycle time T in seconds, by the relationship f =2/T.

For example:

The desired instruction cycle time is 250 nanoseconds, or $250X10^{-9}$ seconds.

Therefore:

$$\frac{2}{250X10^{-9}} = 8X10^{6}HZ = 8\ MHz.$$

Table 2-2. Crystal Specifications

TYPE	IMPEDANCE AT FUNDAMENTAL	IMPEDANCE AT HARMONICS AND SPURS
FUNDAMENTAL MODE, SERIES RESONANT	35 OHMS MAXIMUM	50 OHMS MINIMUM

The third method of frequency control is accomplished by driving the oscillator with an external pulse generator. This method is especially useful in applications that require the ability to vary the instruction cycle time. The X1 and X2 inputs of the 8X300 must be connected to the complementary outputs of the pulse generator as shown in Figure 2-9.

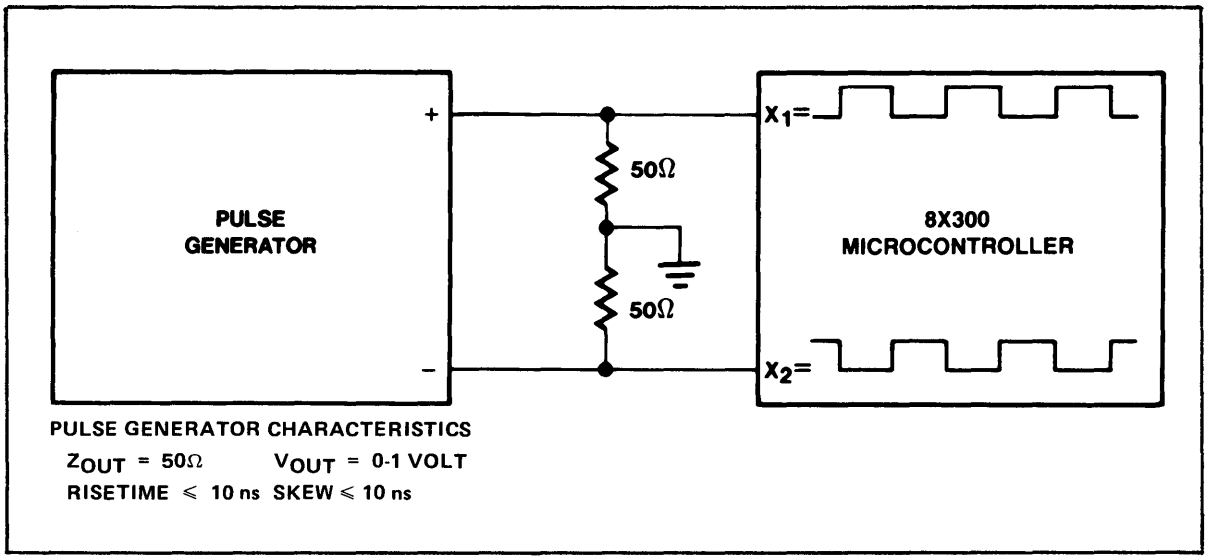


Figure 2-9. Pulse Generator Clocking

The final method of oscillator control involves driving the oscillator with an external TTL clock. The interfacing of a master TTL clock to the 8X300 is illustrated in Figure 2-10.

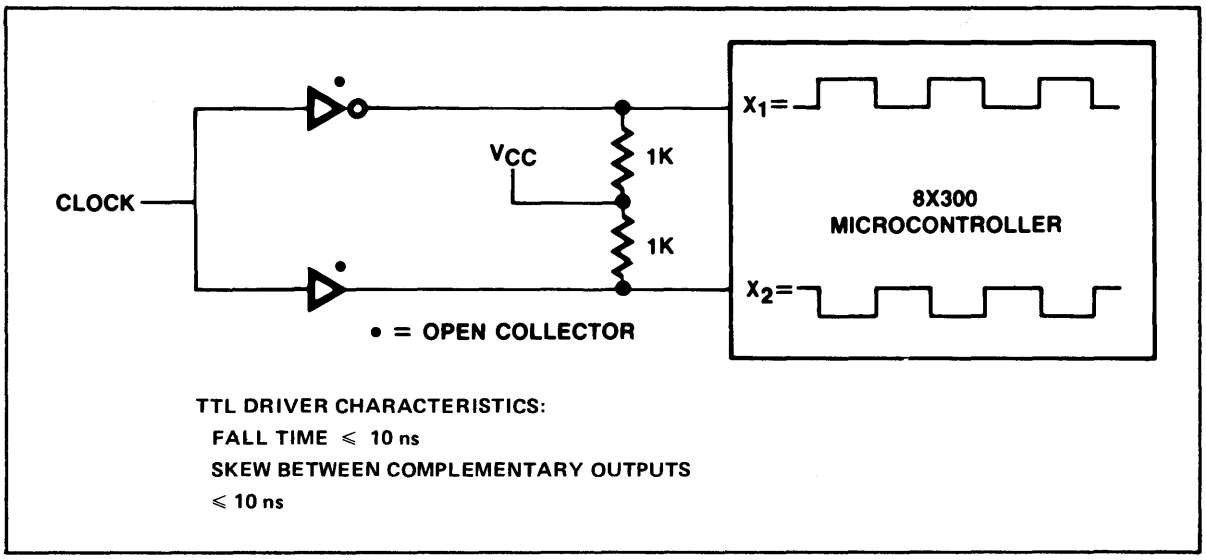


Figure 2-10. TTL Clocking

2.2 8X300 TIMING

To understand the 8X300 timing, it is necessary to know when the internal latches of the device are open and when they are not. Figure 2-11 is a detailed internal block diagram of the 8X300, depicting the latches and registers referred to in the discussions that follow in this section of the manual. Figure 2-12 is a timing chart illustrating the actions of these latches and registers throughout an instruction cycle. Table 2-3 describes the timing specifications of the device. It should be noted that all 8X300 internal latches are level triggered.

2.2.1 Instruction Cycle

The Internal Timing Generator of the 8X300 is depicted in Figure 2-13. Either an external TTL clock or a crystal may be connected to X1 and X2, the inputs to the device's internal oscillator. In conjunction with the sequencer, the output of this oscillator generates all internal timing-control signals. Figure 2-14 shows the relationship of the internal timing signals to the X1 input.

The 8X300 instruction cycle comprises four transitions of the internal system clock. The internal latches are controlled by these transisions, commonly referred to as quarter-cycles. Keep in mind that "quarter-cycles" refer to transitions of the internal system clock, not transitions of X1.

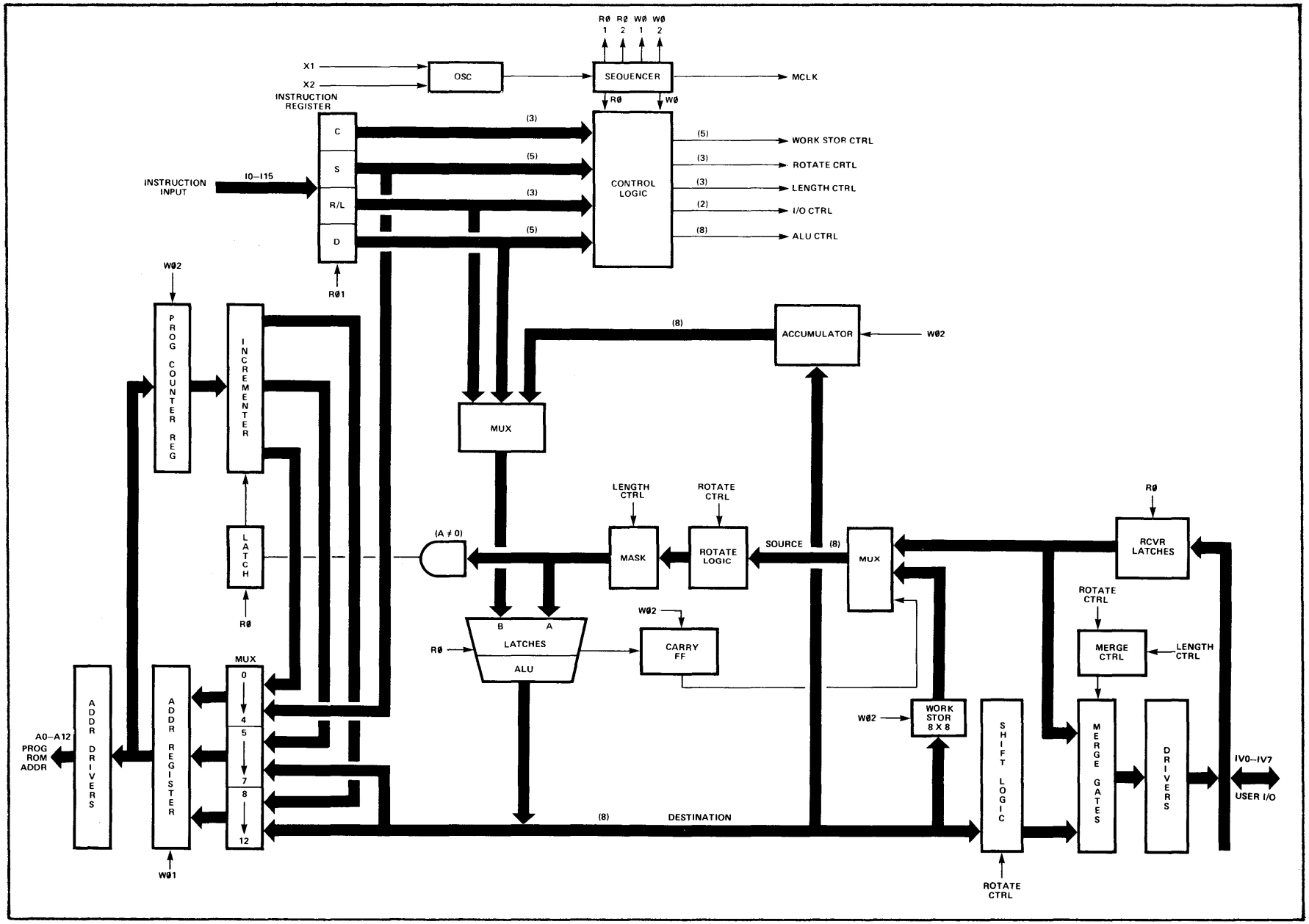
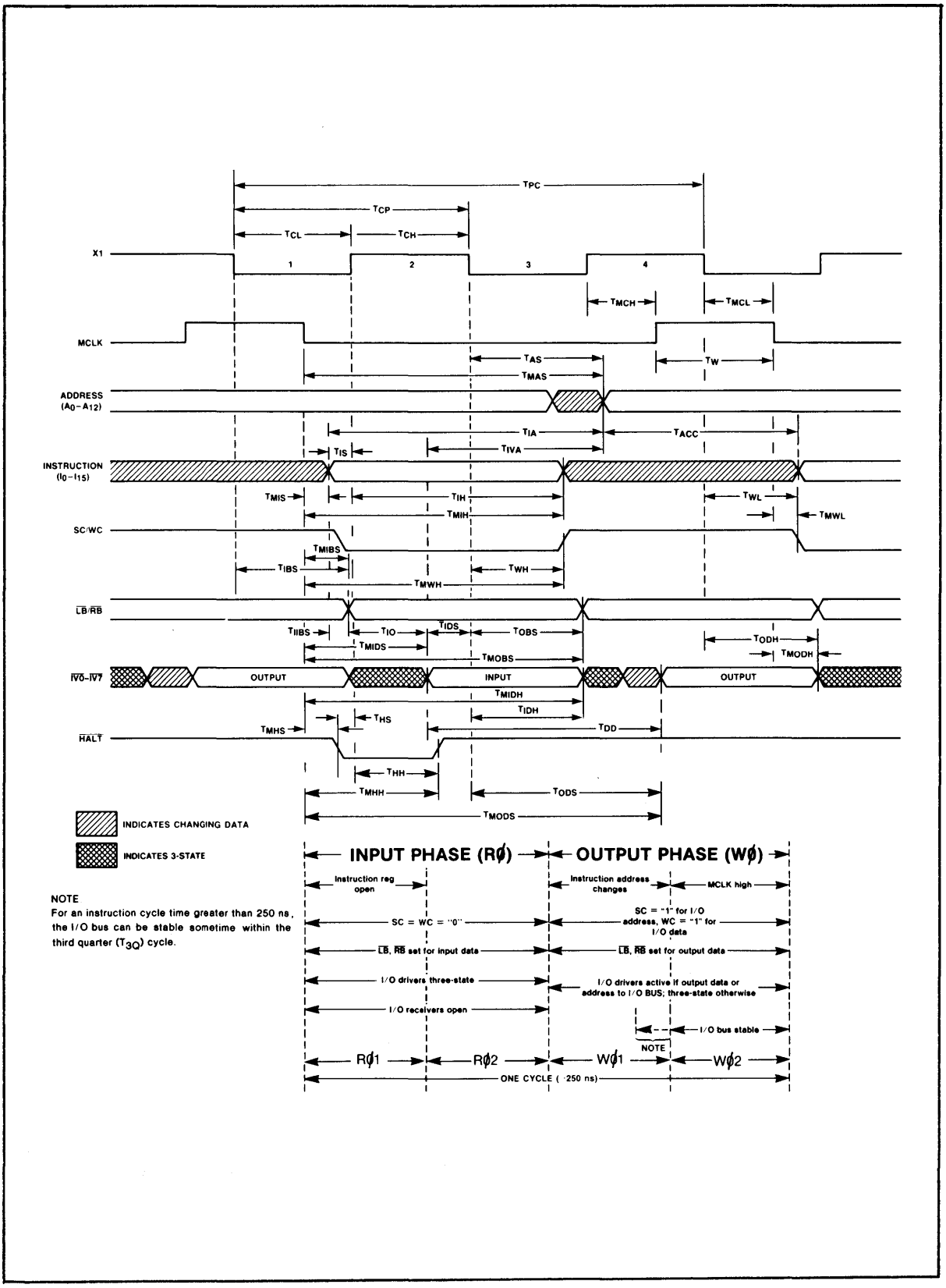


Figure 2-11. 8X300 Detailed Block Diagram



NOTE
For an instruction cycle time greater than 250 ns, the I/O bus can be stable sometime within the third quarter (T_{3Q}) cycle.

Figure 2-12. 8X300 Instruction Cycle Timing

Table 2-3. 8X300 AC Electrical Characteristics

AC CHARACTERISTICS (Commercial Part) **CONDITIONS:** V_{CC} = 5V (±5%), V_{IN} = 0V or 3V, 0°C ≤ T_A ≤ 70°C
LOADING: (See test circuits)

PARAMETER (NOTE 1)		LIMITS (INSTRUCTION CYCLE TIME = 250 ns)			LIMITS (INSTRUCTION CYCLE TIME > 250 ns)			UNITS	COMMENTS
		Min	Typ	Max	Min	Typ	Max		
T_{PC}	Processor cycle time	250			250			ns	
T_{CP}	X1 clock period	125			125			ns	
T_{CH}	X1 clock high time	62			62			ns	
T_{CL}	X1 clock low time	62			62			ns	
T_{MCH}	MCLK high delay	31	42	52	31	42	52	ns	
T_{MCL}	MCLK low delay	31	42	52	31	42	52	ns	
T_W	MCLK pulse width	55	62	69	$T_{4Q}-7$	T_{4Q}		ns	Note 2
T_{AS}	X1 falling edge to address stable	50	63	80	50	63	80	ns	Note 7
T_{MAS}	MCLK falling edge to address stable	130	143	160	$T_{1Q}+T_{2Q}+5$	$T_{1Q}+T_{2Q}+18$	$T_{1Q}+T_{2Q}+35$	ns	Notes 2, 3, & 7
T_{IA}	Instruction to address			170			$T_{2Q}+108$	ns	Notes 2, 3, & 8
T_{IVA}	Input data to address			105			105	ns	Notes 3 & 9
T_{IS}	Instruction set-up time (X1 rising edge)	−7			−7			ns	Note 10
T_{MIS}	MCLK falling edge to instruction stable			20			$T_{1Q}-42$	ns	Notes 2, 4, & 10
T_{IH}	Instruction hold time (X1 rising edge)	45			45			ns	Note 11
T_{MIH}	Instruction hold time) (MCLK falling edge)	60			$T_{1Q}-2$			ns	Notes 2 & 11
T_{WH}	X1 falling edge to SC/WC rising edge	40	49	58	40	49	58	ns	
T_{MWH}	MCLK falling edge to SC/WC rising edge	125	130	135	$T_{1Q}+T_{2Q}$	$T_{1Q}+T_{2Q}+5$	$T_{1Q}+T_{2Q}+10$	ns ns	Note 2
T_{WL}	X1 falling edge to SC/WC falling edge	40	49	58	40	49	58	ns	
T_{MWL}	MCLK falling edge to SC/WC falling edge	5	7	15	5	7	15	ns	
T_{IBS}	X1 falling edge to $\overline{LB}/\overline{RB}$ (Input phase)	48	60	70	48	60	70	ns	
T_{MIBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Input phase)	7	17	25	7	17	25	ns	
T_{IIBS}	Instruction to $\overline{LB}/\overline{RB}$ (Input phase)		27	35		27	35	ns	
T_{OBS}	X1 falling edge to $\overline{LB}/\overline{RB}$ (Output phase)	48	60	70	48	60	70	ns	
T_{MOBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Output phase)	132	137	147	$T_{1Q}+T_{2Q}+7$	$T_{1Q}+T_{2Q}+12$	$T_{1Q}+T_{2Q}+22$	ns	Note 2
T_{IDS}	Input data set-up time (X1 falling edge)	25	16		25	16		ns	
T_{MIDS}	MCLK falling edge to input data stable		65	55		$T_{1Q}+T_{2Q}-60$	$T_{1Q}+T_{2Q}-70$	ns	Notes 2 & 5
T_{IDH}	Input data hold time (X1 falling edge)	40	30		40	30		ns	
T_{MDIH}	Input data hold time (MCLK falling edge)	125	112		$T_{1Q}+T_{2Q}$	$T_{1Q}+T_{2Q}-13$		ns	Note 2
T_{ODH}	Output data hold time (X1 falling edge)	55	65	75	55	65	75	ns	
T_{MODH}	Output data hold time (MCLK falling edge)	11	20	25	11	20	25	ns	
T_{ODS}	Output data stable (X1 falling edge)	74	84	94	74	84	94	ns	Notes 12, 14, & 15
T_{MODS}	Output data stable (MCLK falling edge)	150	160	170	$T_{1Q}+T_{2Q}+25$	$T_{1Q}+T_{2Q}+35$	$T_{1Q}+T_{2Q}+45$	ns	Notes 2, 12, 14, & 15

Table 2-3. 8X300 AC Electrical Characteristics (Cont'd)

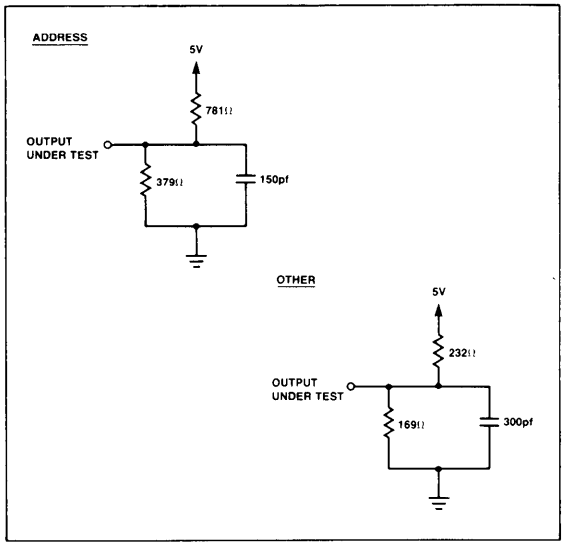
AC CHARACTERISTICS (Commercial Part) (Continued)

CONDITIONS: V_{CC} = 5V (±5%), V_{IN} = 0V or 3V, 0°C ≤ T_A ≤ 70°C
LOADING: (See test circuits)

PARAMETER (NOTE 1)		LIMITS (INSTRUCTION CYCLE TIME = 250 ns)			LIMITS (INSTRUCTION CYCLE TIME > 250 ns)			UNITS	COMMENTS
		Min	Typ	Max	Min	Typ	Max		
T_{DD}	Input data to output data	104	120	136	104	120	136	ns	Notes 13 & 15
T_{HS}	$\overline{HALT}$ set-up time (X1 rising edge)	0			0			ns	
T_{MHS}	MCLK falling edge to $\overline{HALT}$ falling edge			18			$T_{1Q}-44$	ns	Notes 2 & 6
T_{HH}	$\overline{HALT}$ hold time (X1 rising edge)	32			32			ns	
T_{MHH}	$\overline{HALT}$ hold time (MCLK falling edge)	50			$T_{1Q}-12$			ns	Note 2
T_{ACC}	Program storage access time			80				ns	
T_{IO}	I/O port output enable time ($\overline{LB}/\overline{RB}$ to valid IV data input)			30				ns	

NOTES:
1. X1 and X2 inputs are driven by an external pulse generator with an amplitude of 1.5 volts; all timing parameters are measured at this voltage level.
2. Respectively, T_{1Q}, T_{2Q}, T_{3Q}, and T_{4Q} represent time intervals for the first, second, third, and fourth quarter cycles.
3. Capacitive loading for the address bus is 150 picofarads.
4. Same as TIS but referenced to falling edge of MCLK.
5. Same as TIDS but referenced to falling edge of MCLK.
6. Same as THS but referenced to falling edge of MCLK.
7. TAS is obtained by forcing a valid instruction and an I/O bus input to occur earlier than the specified minimum set-up time; the TAS parameter then represents the earliest time that the address bus is valid.
8. TIA is obtained by forcing a valid instruction input to occur earlier than the minimum set-up time.
9. TIVA is obtained by forcing a valid I/O bus input to just meet the minimum set-up time.
10. TMIS represents the set-up time required by internal latches of the 8X300. In system applications, the instruction input may have to be valid before the worst-case set-up time in order for the system to respond with a valid I/O bus input that meets the I/O bus input set-up time (TIDS and TMIDS).
11. TIH represents the hold time required by internal latches of the 8X300. To generate proper LB/RB signals, the instruction must be held valid until the address bus changes.
12. TODS is obtained by forcing a valid I/O bus input to occur earlier than the I/O bus input set-up time (TIDS); this timing parameter represents the earliest time that the I/O output data can be valid.
13. TDD is obtained by forcing a valid I/O bus input to just meet the minimum I/O bus input set-up time; thus timing parameter represents the latest time that the I/O output data can be valid.
14. The minimum figure for these parameters represents the earliest time that I/O bus output drivers of the 8X300 will turn on.
15. For TIDS ≥ 35 ns, TODS or TMODS should be used to determine when the output data is stable.

TEST CIRCUITS



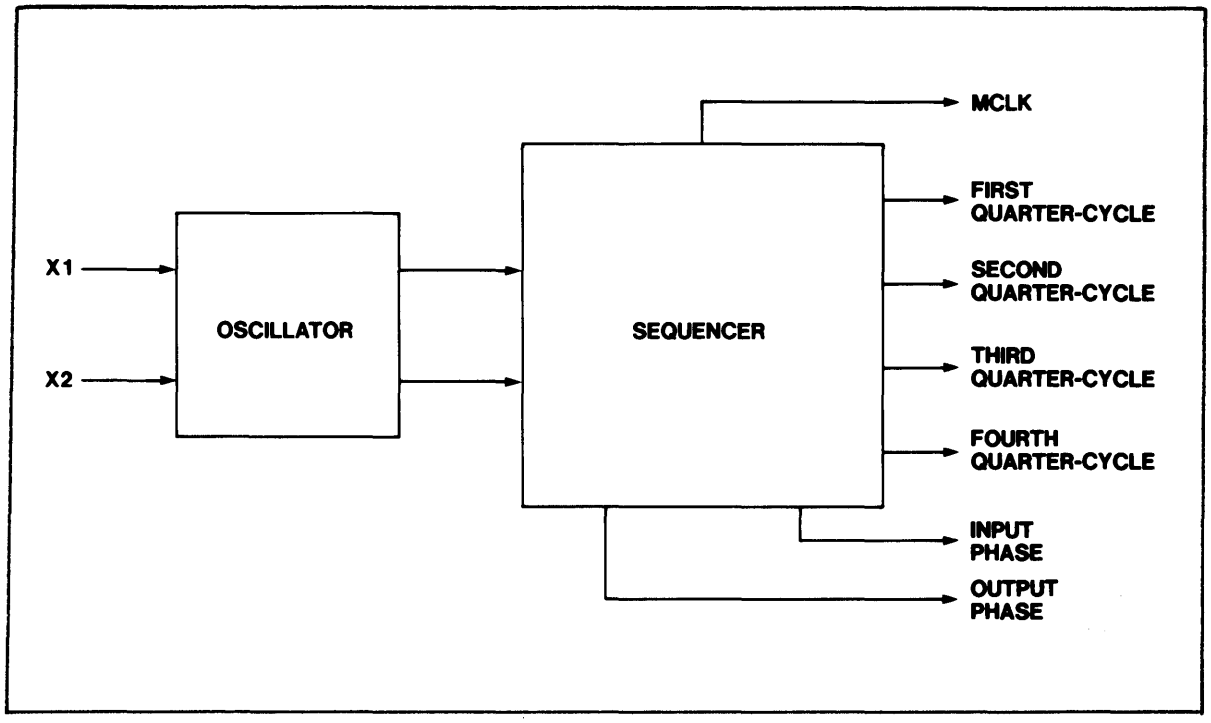
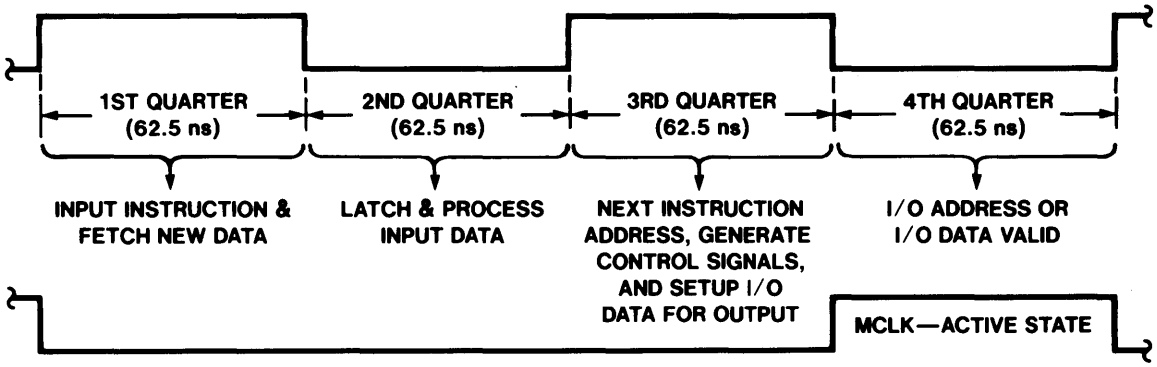


Figure 2-13. Internal Timing Generator



NOTES

1. New instruction must be accepted and latched at end of first quarter cycle.
2. The I/O data latches are open for the first two quarter cycles, that is, for 125-nanoseconds.
3. The address is output during third quarter cycle.
4. I/O bus drivers are active (turned on) during third and fourth quarter cycles.

Figure 2-14. Internal Timing for 250 Nanosecond Instruction Cycle

2.2.2 Timing Calculations

It is an operational requirement of the 8X300 that certain timing events must occur in specific quarter-cycles. In 8X300 systems operating with fast instruction cycle times, most Microcontroller delays are strictly determined by internal gate propagation delays. When operating with slower instruction cycle times, the delays appear to increase due to gating by internal clocks. Figure 2-15 illustrates the timing relationships within the 8X300.

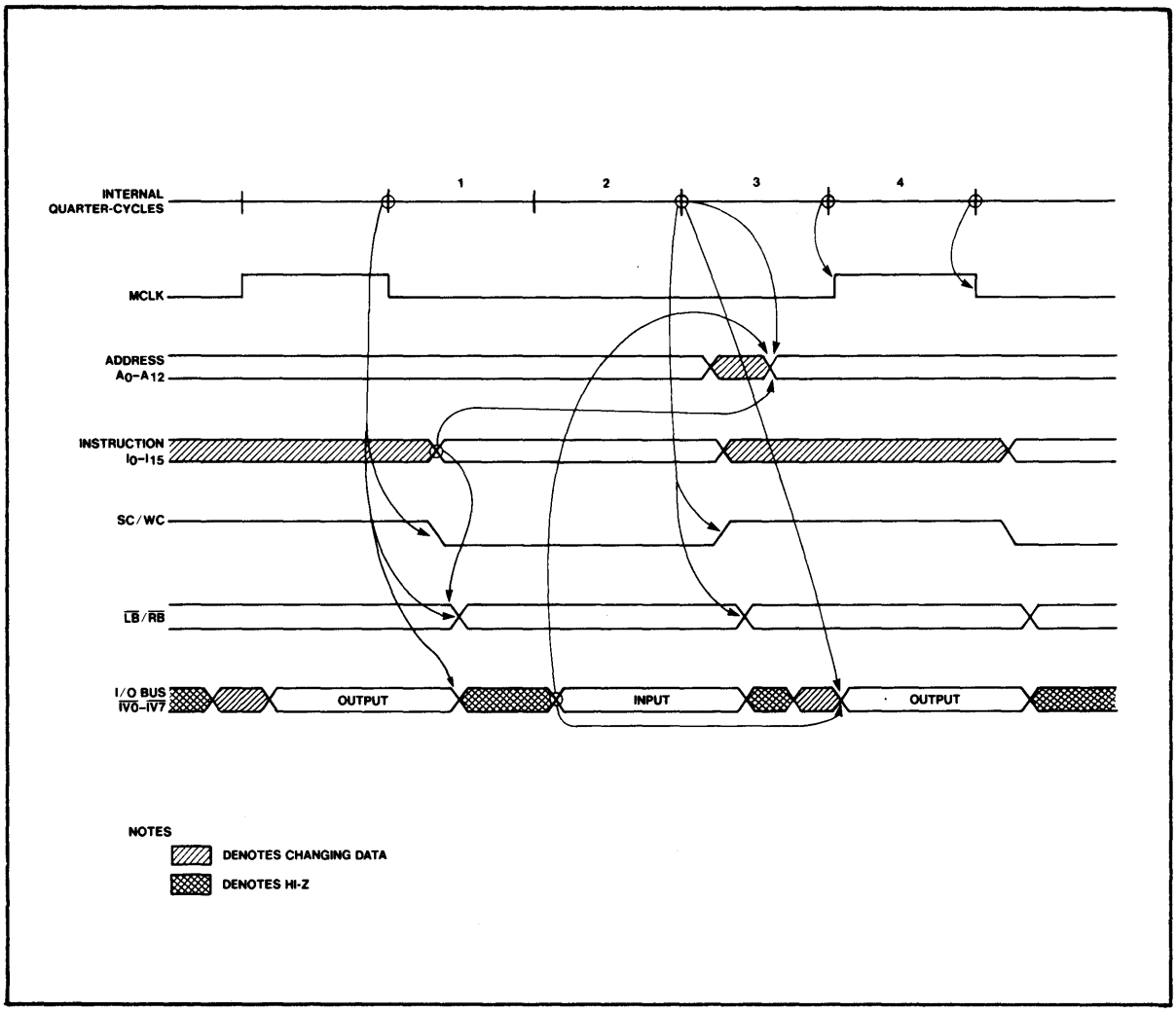


Figure 2-15. 8X300 Timing Relationships

The switching points of the signals illustrated in Figure 2-15 are defined in the following paragraphs:

MCLK: Master clock. MCLK occurs during, and is equal in length to, the fourth quarter-cycle. The leading and trailing edges of MCLK are, respectively, concurrent with the leading and trailing edges of the fourth quarter-cycle.

ADDRESS: Instruction address output signals. The opening and closing of the internal address latches (register) occur, respectively, with the leading and trailing edges of the third quarter-cycle. Therefore, these latches are open only during the third quarter-cycle. The address of the previous instruction cycle remains stable until the trailing edge of the second quarter-cycle, after which time a new address becomes available on the bus. The point at which the new address becomes valid and stable is dependent on the worst case condition of the following considerations:

1. The propagation delay from the address latch input to the address latch output, when the address latch starts to open during the third quarter-cycle (TAS).

2. The propagation delay from the time an instruction becomes stable to the address output (for example, a JMP instruction) (TIA).

3. The propagation delay from the beginning a valid I/O bus input to the address output (TIVA). As an example, an instruction in which bit seven of the selected right bank device is examined and the next instruction is executed if the bit is zero, or the instruction at the current location plus five is executed if the bit is one (NZT RB7,*+5).

INSTRUCTION: Input signals to the instruction latches (register). These latches are open for the duration of the first quarter-cycle only. When considered at the chip level, there is a minimum set-up and hold time requirement with respect to the trailing edge of the first quarter-cycle. At a system level, there are further requirements as to when and how long the instruction inputs must be valid. As is shown in Figure 2-16, the $\overline{LB}/\overline{RB}$ signals are derived directly from the instruction input during the input phase.

To insure the correct $\overline{LB}/\overline{RB}$ signals, the instruction input must be stable until the end of the input phase. Since the address bus is stable for the entire input phase, the instruction output of the program storage should be stable for the entire input phase, thus satisfying the requirements for stability of the $\overline{LB}/\overline{RB}$ signals.

Another factor determining the latest point at which the instruction must become valid is the worst case I/O input set-up time, as shown below:

I/O - data stability depends on the setting of the $\overline{LB}/\overline{RB}$ control lines which, in turn, depend on the stability of the instruction bus. See section on "Timing Considerations" for detailed explanation.

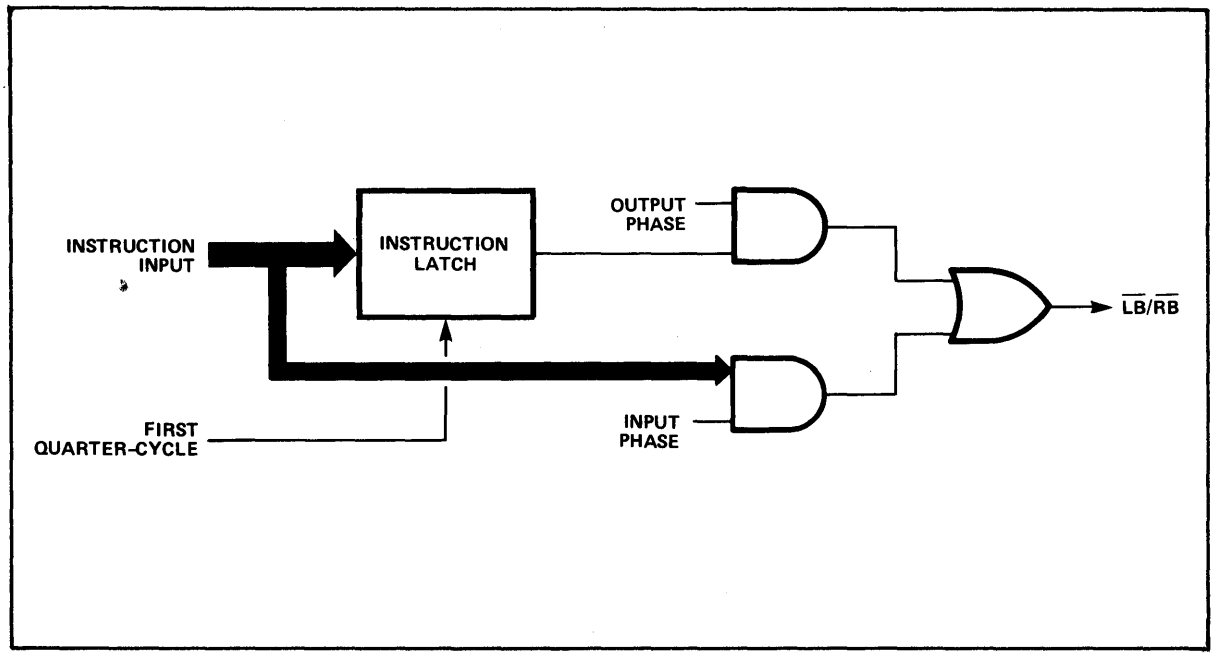


Figure 2-16. $\overline{LB}/\overline{RB}$ Enable Paths

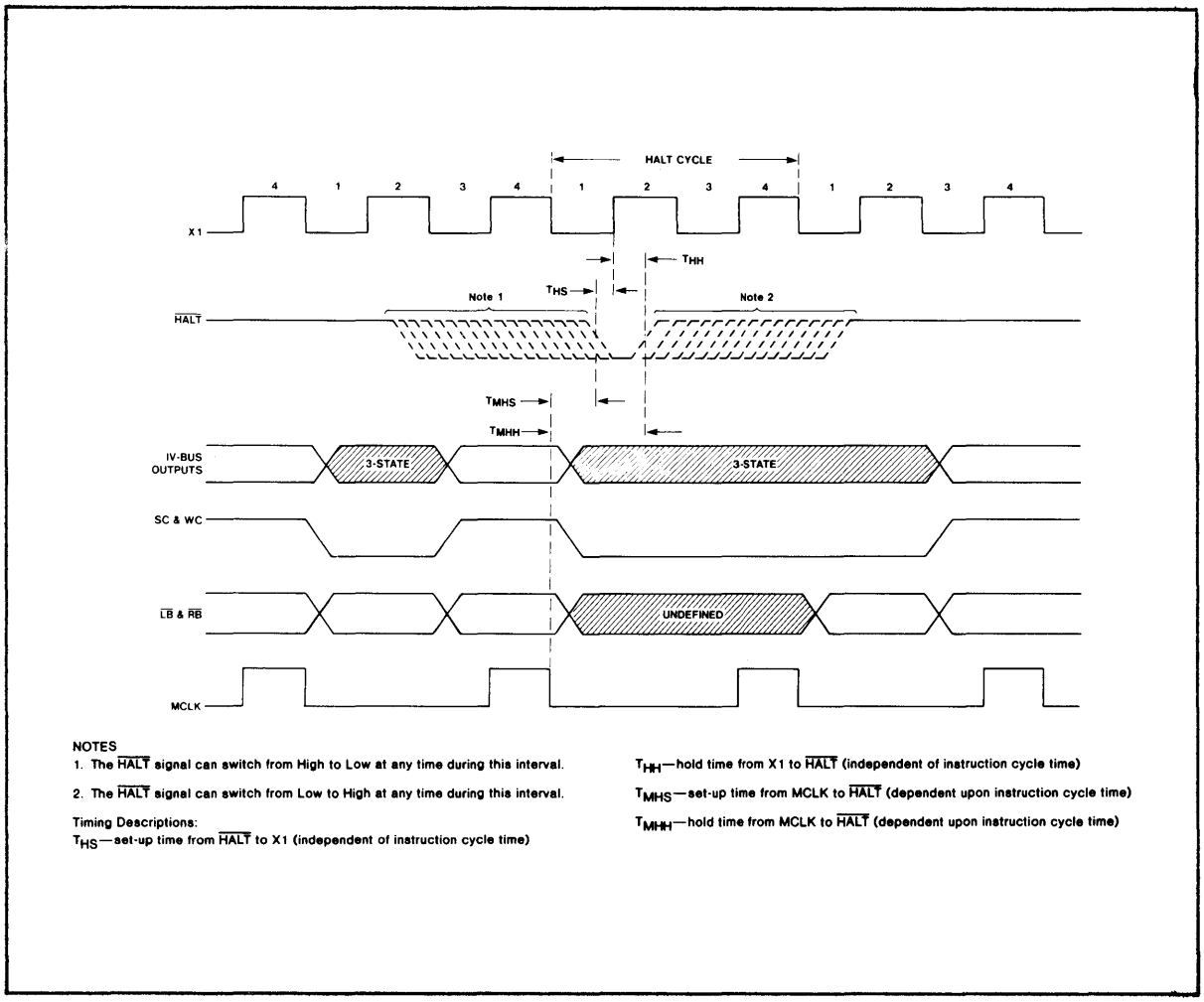
SC/WC: Select command and Write command signals. These signals change state at the leading edges of the first and third quarter-cycles. Keep in mind that SC and WC are never concurrently in a high state.

$\overline{LB}/\overline{RB}$: The complementary left bank and right bank enable signals. When required to change during the input phase, $\overline{LB}/\overline{RB}$ becomes valid after the latest occurance of either the leading edge of the first quarter-cycle (TIBS) or the beginning of a valid instruction (TIIBS). If a change is required during the output phase, $\overline{LB}/\overline{RB}$ is valid after the leading edge of the third quarter-cycle. Refer to Figure 2-16.

I/O Bus: Input/Output Bus. The I/O bus input latches and the ALU input latches are open during the first two quarter-cycles. Allowing some delay after the leading edge of the third quarter-cycle, the I/O bus drivers are turned on, and remain on throughout the fourth quarter-cycle. When operating at 250 nanosecond instruction cycle times, the I/O data may not be valid until the fourth quarter-cycle.

$\overline{HALT}$: Halt Signal. The $\overline{HALT}$ signal is sampled by the 8X300 internal logic during the first quarter-cycle. If the $\overline{HALT}$ signal

goes low (and remains low) during the first quarter-cycle, the current instruction cycle becomes a $\overline{\text{HALT}}$ cycle. A $\overline{\text{HALT}}$ cycle ceases the internal operation of the 8X300, but does not inhibit MCLK or affect any internal registers. During the $\overline{\text{HALT}}$ cycle the I/O bus is in a high-impedance mode, SC and WC are low, and the address bus, A0-A12, retains the address latched in it at the time the $\overline{\text{HALT}}$ was applied. Normal operation resumes with the cycle in which the $\overline{\text{HALT}}$ signal is high when sampled. Refer to Figure 2-17.



NOTES
1. The $\overline{\text{HALT}}$ signal can switch from High to Low at any time during this interval.
2. The $\overline{\text{HALT}}$ signal can switch from Low to High at any time during this interval.

Timing Descriptions:
T_{HS}—set-up time from $\overline{\text{HALT}}$ to X1 (independent of instruction cycle time)
T_{HH}—hold time from X1 to $\overline{\text{HALT}}$ (independent of instruction cycle time)
T_{MHS}—set-up time from MCLK to $\overline{\text{HALT}}$ (dependent upon instruction cycle time)
T_{MHH}—hold time from MCLK to $\overline{\text{HALT}}$ (dependent upon instruction cycle time)

Figure 2-17. $\overline{\text{HALT}}$ Timing

Although $\overline{\text{RESET}}$ timing is not shown in Figure 2-12, it is appropriate that it be described in this portion of text. Figure 2-18 depicts the relationship of $\overline{\text{RESET}}$ to other signals of the 8X300.

When power is first applied to the 8X300, the $\overline{\text{RESET}}$ input of the device must be forced low as part of the initialization process. This

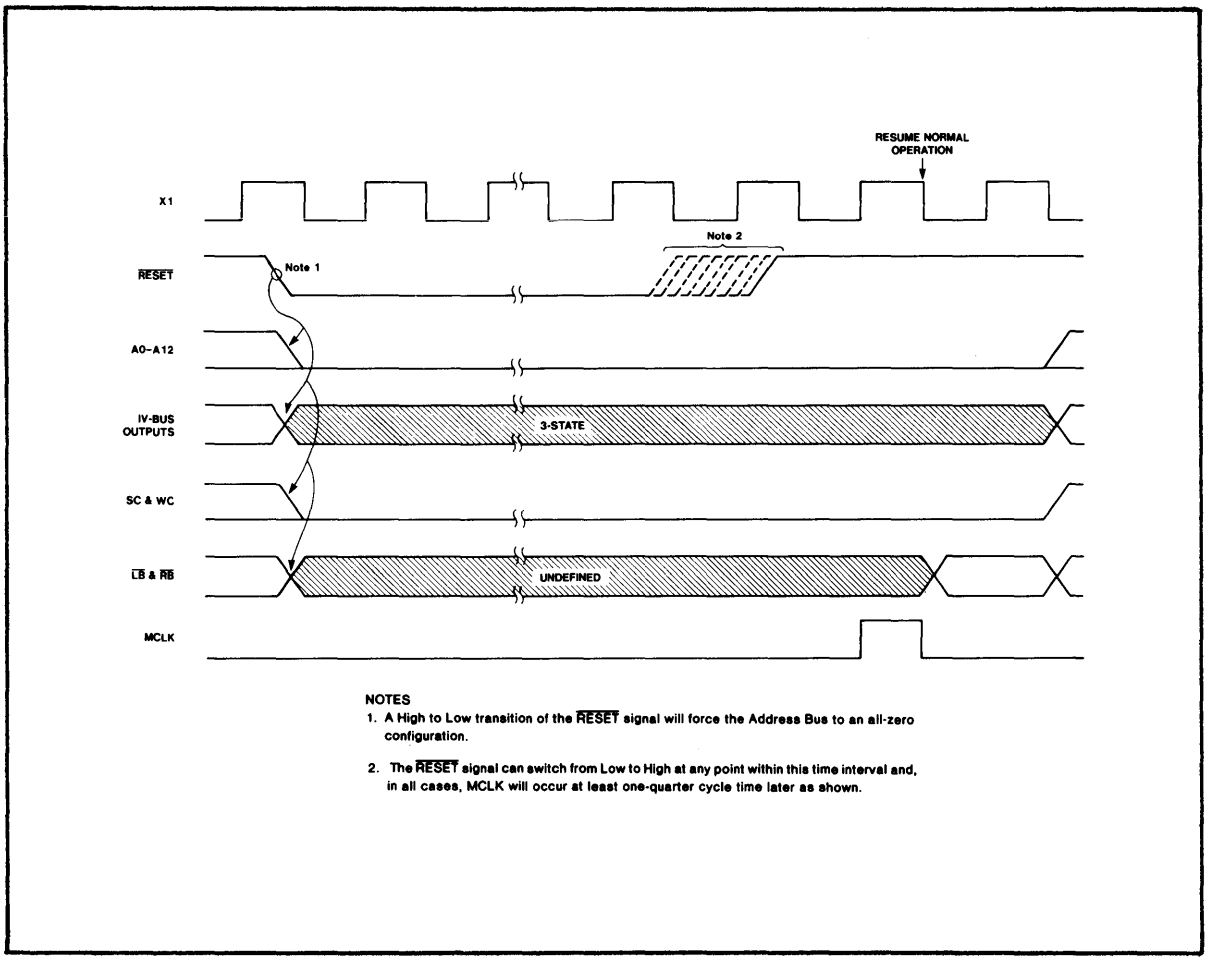


Figure 2-18. RESET Timing

must be done by some external means, such as an R-C network. To ensure proper operation during normal program flow, the RESET input must be forced low for at least one full instruction cycle. When RESET goes low, the following takes place:

1. The Program Counter and Address Register set to zeros asynchronously, with a delay equal to the propagation delay. Their contents remain zeros as long as the RESET input remains low.

2. The I/O bus goes to the high-impedance mode asynchronously, and remains in this state as long as RESET is held low.

3. The SC and WC outputs go low asynchronously, and remain low as long as RESET remains low.

4. The $\overline{LB}/\overline{RB}$ outputs are undefined as long as $\overline{RESET}$ remains low.

5. The MCLK output is inhibited as long as $\overline{RESET}$ is held low. If $\overline{RESET}$ is forced low during the last two quarter-cycles, the MCLK that occurs during that instruction cycle may be shortened.

The $\overline{RESET}$ signal has no affect on any internal registers other than those listed above. When the $\overline{RESET}$ input is allowed to again go high, at least one quarter-cycle later an MCKL (approximately one quarter-cycle long) occurs prior to the resumption of normal operation.

TIMING CONSIDERATIONS (Commercial Part) - As shown in the "AC CHARACTERISTICS" table for this part, Table 2-3, the minimum instruction cycle time is 250 nanoseconds, whereas, the maximum is determined by the on-chip oscillator frequency and can be any value the user chooses. With an instruction cycle time of 250 nanoseconds, the part can be characterized in terms of absolute values; these are shown in the first "LIMITS" column of the table. When the instruction cycle time is greater than 250 nanoseconds, certain parameters are cycle-time dependent; thus, these parameters are specified in terms of the four quarter-cycles (T_{1Q}, T_{2Q}, and T_{4Q}) that make up one instruction cycle -- see the timing diagram Figure 2-12. As the time interval for each instruction cycle increases (becomes greater than 250 nanoseconds), the delay for all parameters that are cycle-time dependent is likewise increased. In some cases, these delays have a significant impact on timing relationships and other areas of systems design; subsequent paragraphs describe these timing parameters and reliable methods of calculation.

Timing parameters for the 8X300 are normally measured with reference to X1 and MCLK; those referenced to MCLK are prefaced with an "M" in the mnemonic-- TMAS, TMIH, and so on. To determine the timing relationship between a particular signal, say "A", and MCLK, the user should, at all times, use the value specified in the table-- DO NOT calculate the value by adding or subtracting two or more parameters that are referenced to X1. When deriving timing relationships between two signals (A to B, etc.) by adding or subtracting the parameter values, the user must consistently use the same parameter reference-- MCLK or X1.
System determinants for the instruction cycle time are:

Propagation delays within the 8X300
Access time of Program Storage
Enable time of Output Port

Normally, the instruction cycle time is constrained by one or more of the following conditions:

Condition 1 -- Instruction or MCLK to $\overline{LB}/\overline{RB}$ (input phase) plus I/O port output enable (TIO) ≤ IV data set-up time (Figure 2-19a)

Condition 2 -- Program storage access time (TACC) plus instruction to $\overline{LB}/\overline{RB}$ (input phase) plus I/O port output enable (TIO) plus IV data (input phase) to address ≤ instruction cycle time (Figure 2-19b).

Condition 3 -- Program Storage access time plus instruction to address ≤ instruction cycle time (Figure 2-19c).

From condition #1 and with an instruction cycle time of 250 nanoseconds, the I/O port output enable time (TIO) can be calculated as follows:

$$TMIBS + TIO \leq TMIDS$$
transposing, $TIO \leq TMIDS - TMIBS$
substituting, $TIO \leq 55nS - 25\ nS$
result, $TIO \leq 30$ nanoseconds

Using 30 nanoseconds for TIO, the constraint imposed by Condition #1 can also be used to calculate the minimum cycle time:

$$TMIBS + TIO \leq TMIDS$$
thus, $25\ nS + 30\ nS \leq T_{1Q} + T_{2Q} - 70$
$25\ nS + 30\ nS \leq \frac{1}{2}$ cycle - 70 therefore,

the worst-case instruction cycle time is 250 nanoseconds. With subject parameters referenced to X1, the same calculations are valid:

$$TIBS + TIO + TIDS \leq \frac{1}{2} \text{ cycle}$$
thus, $70\ nS + 30\ nS + 25\ nS \leq \frac{1}{2}$ cycle therefore,

the worst-case instruction cycle time is again 250 nanoseconds. From Condition #2 and with an instruction cycle time of 250 nanoseconds, the program storage access time can be calculated:

$$TACC + TIIBS + TIO + TIVA \leq 250\ nS$$
transposing, $TACC \leq 250\ nS - TIIBS - TIO - TIVA$
substituting, $TACC \leq 250\ nS - 35\ nS - 30\ nS - 105\ nS$
thus, $TACC \leq 80$ nS hence, for an instruction cycle time of 250 nanoseconds, a program storage access time of 80 nanoseconds is implied. The constraint imposed by Condition #3 can be used to verify the maximum program storage access time:

TIA + TACC ≤ Instruction Cycle

thus, TACC ≤ 250 nS - 170 nS

and, TACC ≤ 80 nS, confirming that a program storage access time of 80 nanoseconds is satisfactory.

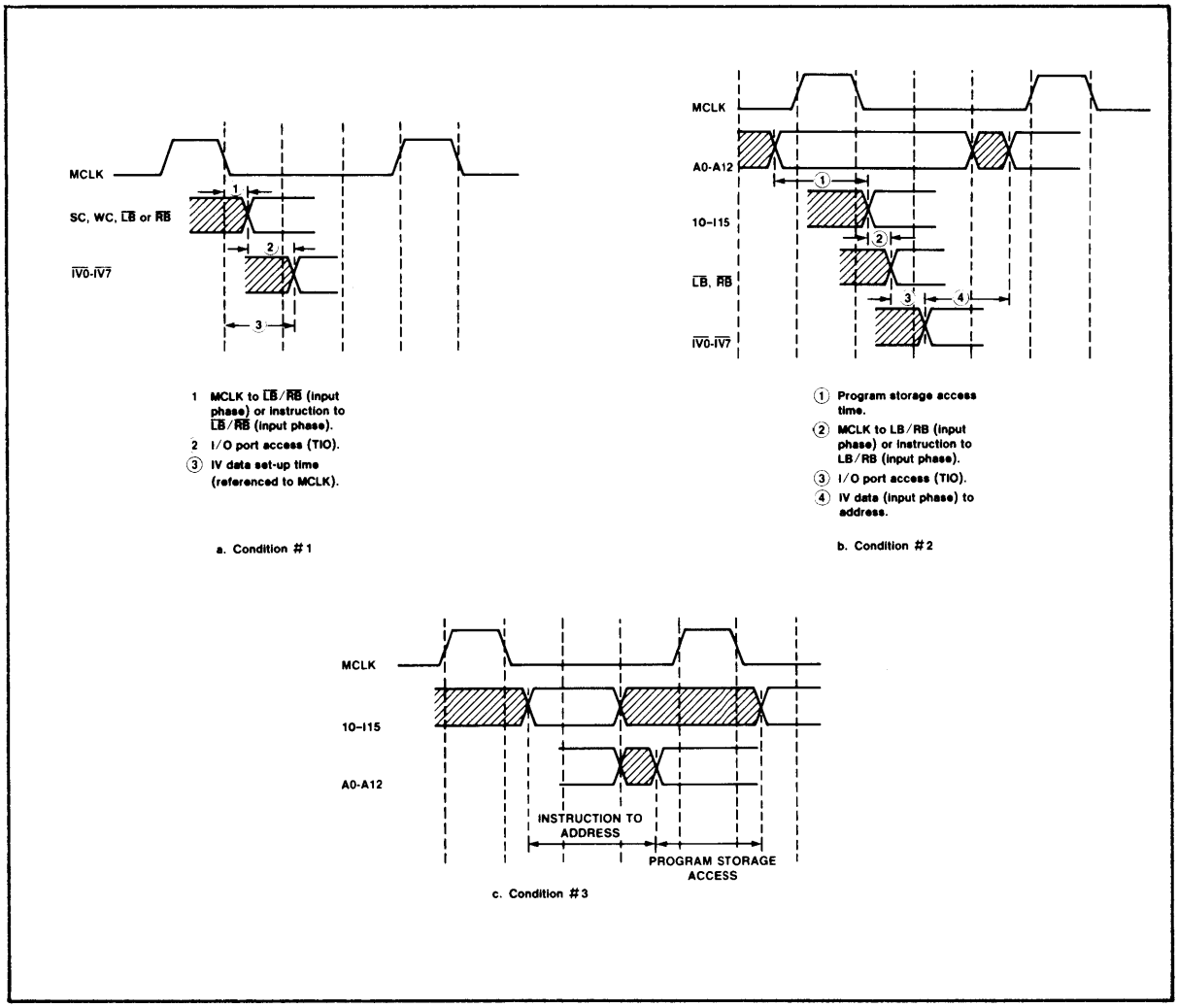


Figure 2-19. Constraints of 8X300 Instruction Cycle Time

For an instruction cycle time of 250 nanoseconds and a program storage access time of 80 nanoseconds (Condition #2), the instruction should be valid 10 nanoseconds before the falling edge of MCLK. This relationship can be derived by the following equation:

$$\begin{aligned} & 250\ nS - TMAS - TACC \\ = \ & 250\ nS - 160\ nS - 80\ nS \\ = \ & 10\ nS \end{aligned}$$

It is important to note that, during the input phase, the beginning of a valid $\overline{LB}/\overline{RB}$ signal is determined by either the instruction to

$\overline{LB}/\overline{RB}$ delay (TIIBS) or the delay time from the falling edge of MCLK to $\overline{LB}/\overline{RB}$ (TMIBS). Assuming the instruction is valid 10 nanoseconds before the falling edge of MCLK and adding the instruction-to-LB/RB delay (TIIBS = 35 nS), the $\overline{LB}/\overline{RB}$ signal will be valid 25 nanoseconds after the falling edge of MCLK. With a fast program-storage memory and with a valid instruction more than 10 nanoseconds before the falling edge of MCLK -- the $\overline{LB}/\overline{RB}$ signal will, due to the TMIBS delay, still be valid 25 nanoseconds after the falling edge of MCLK. Using a worst-case instruction cycle time of 250 nanoseconds, the user cannot gain a speed advantage by selecting a memory with faster access time. Under the same conditions, a speed advantage cannot be obtained by using an I/O port with fast output enable time (TIO) because the address bus will be stable 80 nanoseconds (TAS) after the beginning of the third quarter-cycle -- no matter how early the IV data input is valid.

When operating at slower instruction cycle times (TPC > 250 nS), there are two more timing conditions which must be satisfied in addition to those already mentioned. First, the I/O input data must be stable by the set-up time required by the input latches. The program storage access time (TACC) must be such that

$$TAS + TACC + TIIBS + TIC + TIDS \leq TPC$$

Second, the instruction must be stable by the set-up time required by the instruction latches. The program storage access time must also be such that

$$TAS + TACC + TIS \leq TPC - T_{2Q}$$

where T_{2Q} is the length of the second quarter-cycle (pulse width of X1 high); for symetric clock signals

$$T_{2Q} = \tfrac{1}{4}\ TPC, \text{ or}$$

$$TAS + TACC + TIS \leq .75\ TPC$$

Program storage access time must satisfy the worst case of all of the timing conditions mentioned.

2.3 THE 8X300 INSTRUCTION SET

The 8X300 instruction set comprises eight categories of instruction, each identified by a different OP code value. A variable operand field within these instructions provide the equivalent of 32 instructions. During the process of an instruction, the data may undergo some combinations of rotate, mask, and shift and merge manipulations. Table 2-4 describes the functions that can be performed on the data during the instruction operations. Certain instruction operations perform no function on the data, therefore, they do not appear in this Table (example: XMIT, Register).

The following paragraphs, in conjunction with Table 2.4, provide a brief description of the operations that may be performed by each of the instruction categories. It is suggested that when reading these descriptions, the reader should also refer back to the internal block diagram, Figure 2-11.

Op Code 0 (MOVE): Data from the source register or I/O bus is moved to the destination register or I/O bus. Bit manipulations may be performed as specified in Table 2-4. The source data field remains unchanged after the operation.

Op Code 1 (ADD): Data from the source register or I/O bus is added to the contents of the AUX register (Accumulator). The result is then placed in the destination register or I/O bus. Bit manipulations may be performed as specified in Table 2-4. The source data field and AUX register remain unchanged unless one of those is also the destination. Only during the ADD instruction is the value of Register 10 set. (RIO cannot be specified as a destination.) The high order seven bits always remain zero. The LSB is set to indicate the carry-out of the MSB of the ALU.

NOTE: This does not indicate an overflow condition resulting from excessive magnitude of a two's complement sum.

Op Code 2 (AND): Data from the source register or I/O bus is ANDed with the contents of the AUX register (Accumulator). The result is then placed in the destination register or I/O bus. Bit manipulations may be performed as specified in Table 2-4. The source data field and AUX register remain unchanged unless one of those is also the destination.

Table 2-4. Bit Manipulation Functions

	FUNCTION			
OPERATION	ROTATE	MASK	SHIFT	MERGE
MOVE, ADD, AND, XOR REGISTER – REGISTER	OK	X	X	X
MOVE, ADD, AND, XOR REGISTER – IV BUS ADDRESS	OK	X	X	X
MOVE, ADD, AND, XOR REGISTER – IV BUS	X	X	OK	OK
MOVE, ADD, AND, XOR IV BUS – REGISTER	OK	OK	X	X
MOVE, ADD, AND, XOR IV BUS – IV BUS	OK	OK	OK	OK
MOVE, ADD, AND, XOR IV BUS – IV BUS ADDRESS	OK	OK	X	X
XEC, NZT IV BUS	OK	OK	X	X
XMIT IV BUS	X	X	OK	OK

NOTES:
1. X REPRESENTS A NON-USABLE FUNCTION.
2. XEC, NZT AND XMIT REGISTER, AND XMIT IV BUS ADDRESS HAVE NO BIT MANIPULATIONS PERFORMED DURING THEIR OPERATION.

Op Code 3 (XOR): Data from the source register or I/O bus undergoes an EXCLUSIVE OR operation with the contents of the AUX register (Accumulator). The result is then placed in the destination register or I/O bus. Bit manipulations may be performed as specified in Table 2-4. The source data field and AUX register remain unchanged unless one of those is also the destination.

Op Code 4 (XEC): Executes the instruction at the address formed by replacing the least significant bits of the current address with the sum of the J field and the data in the source register or I/O port.* Bit manipulations may be performed on I/O source data only as specified in Table 2-4. After execution of the instruction at the specified address, instruction execution continues at the address following the XEC instruction, unless the instruction at the specified address caused a jump.

Op Code 5 (NZT): The least significant bits of the program counter are replaced by the J field data if the register or I/O bus specified by the source field has non-zero contents.* Bit manipulations may be performed on I/O destination data only as specified in Table 2-4. The tested data field remains unchanged.

Op Code 6 (XMIT): The data in the J field is placed in the register or I/O port specified as the destination field.* Bit manipulations may be performed as specified in Table 2-4.

Op Code 7 (JMP): The address of the next instruction to be executed is changed to that specified by the 13-bit A field of the instruction.

*NOTE:

If the source field specified a register, the J field contains an eight-bit value. If the source field specifies an I/O port (or memory), the J field contains a five-bit value.

A compendium of information on instruction formats is contained in Appendix B.

2.4 INTERFACE PRINCIPLES

Designing hardware to interface with the 8X300 Microcontroller is primarily a task of selecting external devices that suit the job to be done. Careful consideration must be given to the minimum requirements necessary to accomplish the job. Such requirements might include instruction cycle time, memory size and memory access time. Must the system operate with a 250 nS instruction cycle time, or is 500 nS sufficient? At longer instruction cycle times, slower and less expensive memory may be used. How much External Working Storage is necessary? As External Working Storage increases, the methods of manipulating it become more complex. Which is the most economical method of manipulating larger and/or slower memory - hardware or software?

Signetics offers a broad line of devices that are either specifically designed for use with or are compatible with the 8X300. Table 2-5 provides a listing of these devices, by function. Selection of devices from this list can greatly reduce the time and tasks necessary in 8X300 system design.

The following sections provide general information which will aid designers seeking solutions to specific interface problems.

2.4.1 Memory Interface

It should be pointed out that the program storage and working storage devices listed in Table 2-5 are all bipolar devices. Furthermore, as is shown in Table 2-5, the access times of the majority of devices are within the 80 nanosecond maximum required for an 8X300 operating with an instruction cycle time of 250 nanoseconds. This is not to say that all systems must have bipolar memory devices, or that all systems must operate with a maximum memory access time of 80 nanoseconds. If the system under design is of a nature that allows instruction cycle times greater than 250 nanoseconds, the designer might desire to stretch the instruction cycle time in order to select memory from readily available industry standard, TTL compatible MOS memory devices. It is suggested that the designer review Section 2-2 of this manual prior to attempting calculations of instruction cycle times versus memory access times.

As an alternative to stretching the entire instruction cycle, the designer might consider an oscillator configuration which selectively stretches the instruction quarter-cycles. A suggested configuration is shown in Figure 2-20. This design stretches all positive phases of clock input X1. The system is self-synchronizing, and elongates the memory fetch cycles. An arbitrary oscillator frequency of 6.154 MH_Z and delay time of 18.75 nanoseconds was chosen for purposes of illustration. Other oscillator frequencies and delay times will yield different quarter-cycle time periods.

Table 2-5. 8X300 Compatible Devices

Program Storage (ROM, PROM)

Type	Description			Access Time
82S23/123	OC/TS	32X8	PROM	50 ns
82S126/129	OC/TS	256X4	PROM	50 ns
82S131	TS	512X4	PROM	50 ns
82S115	TS	512X8	PROM	60 ns
82S137	TS	1KX4	PROM	60 ns
82S181	TS	1KX8	PROM	70 ns
82S185	TS	2KX4	PROM	100 ns
82S190/191	OC/TS	2KX8	PROM	80 ns
82S2708(Mil. only)	TS	1KX8	PROM	60 ns

Working Storage (RAM)

82S09	OC	64X9	RAM	50 ns
82S16	TS	256X1	RAM	50 ns
82S116	TS	256X1	RAM	40 ns
8X350	TS	256X8	RAM	35 ns

I/O Devices

8T26A/28	Quad Bus Transceiver
8T31	8-Bit Bidirectional I/O Port
8T32/33/35/36	8-Bit Addressable Bidirectional I/O Port
8T39	Bus Extender
8T58	Transparent Bus Expander
8X41	Bidirectional Bus Extender/Repeater
8X320	Bus Interface Register Array

Communications Devices

2652	Synchronous Data Link Controller
8X01	Cyclic Redundancy Checker
8X330	Floppy Disk Formatter/Controller
9403	64-Bit FIFO Buffer Memory

Integrated Fuse Logic

82S100/101	TS/OC 16X8X48 FPLA
82S102/103	OC/TS FPGA

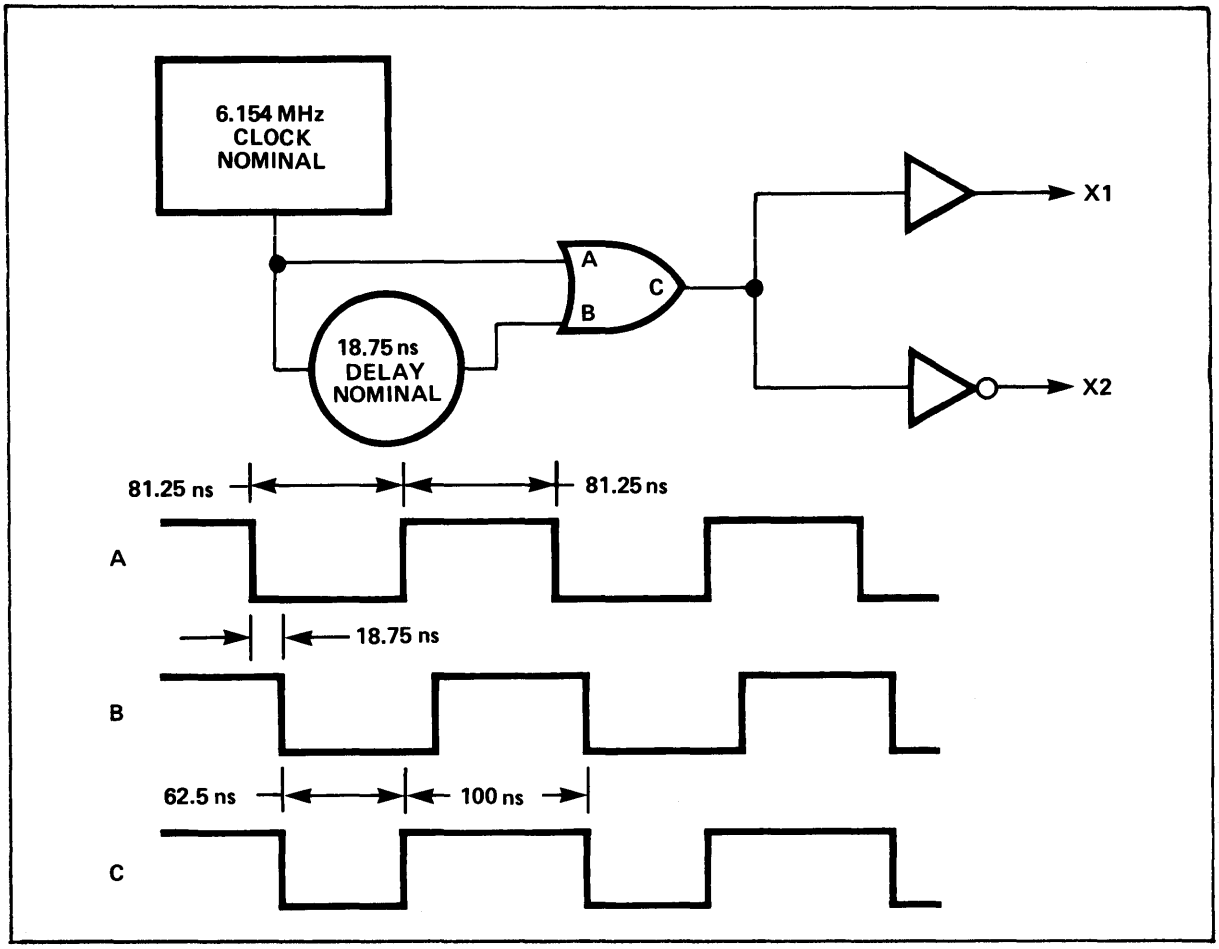


Figure 2-20. Elongated Positive Cycle Clock

Figure 2-21 illustrates a system which operates with up to 64K bytes of industry standard, TTL compatible MOS working storage. The system is implemented through the use of some additional hardware. It should be noted that the program memory should consist of devices with an access time of 80 nanoseconds or less and that the instruction word has been extended from 16 bits to 20 bits. Two of these bits are used as enables for the RAM address latches. The remaining two bits may be used for other control or display drive functions. Although this method requires additional ROMs for the extension, it eliminates the latch select instruction cycle that would be necessary if addressable latches, such as the 8X32, were used as RAM address latches. This saves both in program storage and in total processing time for working storage. It requires that a specific program structure be followed within program storage.

The RAM address latches must be sequentially loaded, the data latch selected, then the data itself loaded into the data latch. If the RAM array is composed of devices with an access time of 250 nanoseconds or less, the RAM address will be stable by the time data is to be written to or read from the RAM array.

The system designer must ensure that the program instruction becomes valid early enough relative to MCLK in order for the RAM address latch enable bits (I16 and I17) to be latched into the D-latches. In cases where the instruction is not yet valid at the trailing edge of MCKL, the MCLK signal to the D-latches can be slightly delayed by adding two or three buffer stages to the MCKL signal line, just prior to the D-latches

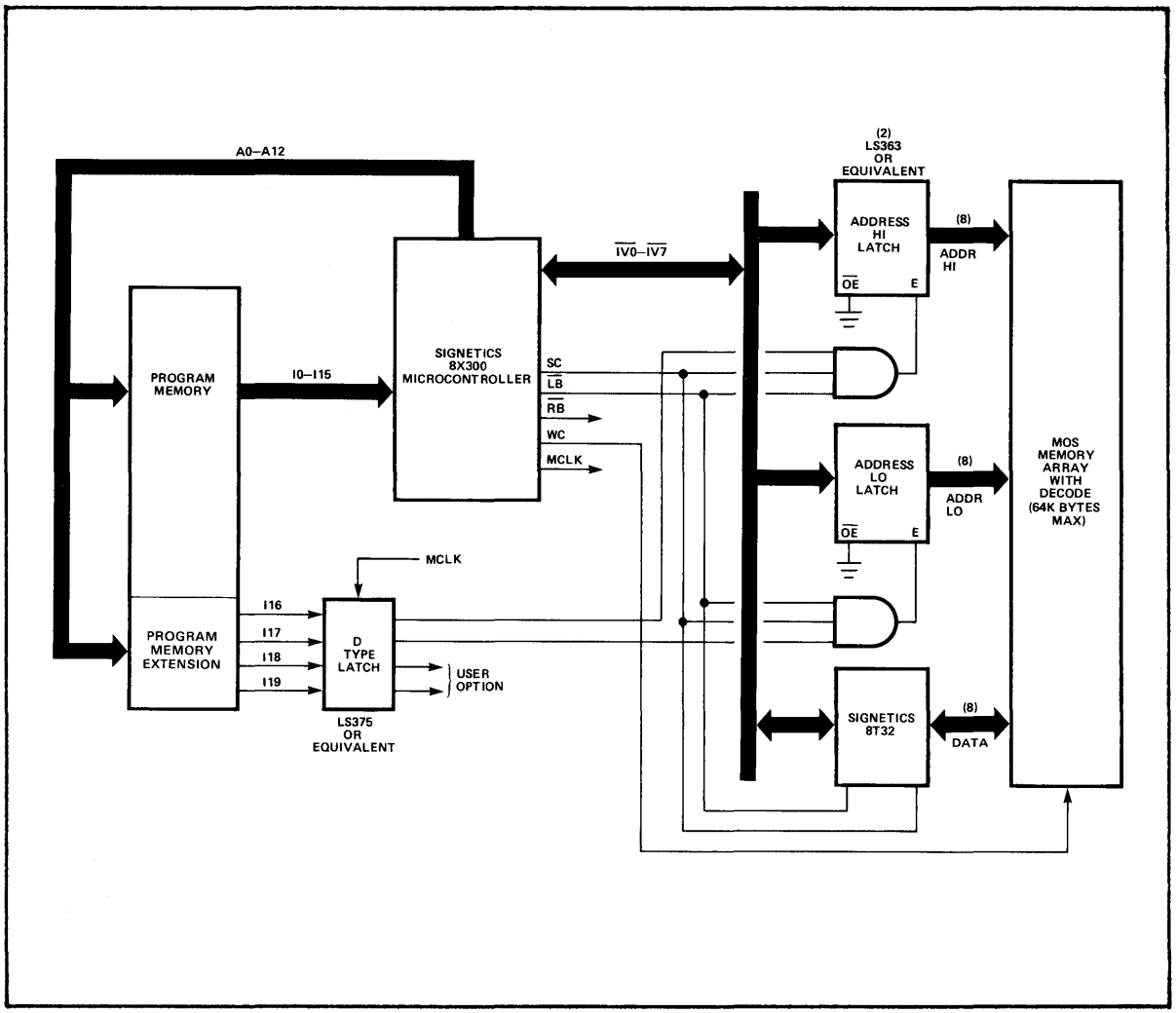


Figure 2-21. Extended Microcode Select

When interfacing program storage composed of ROMs with an access time greater than 80 nanoseconds, the problems facing the designer are quite similar to those encountered when designing for slower working storage. Often, the solution to the problem is the same for both slower working storage and slower program storage. For example, an instruction cycle time of 500 nanoseconds will allow the designer to use either RAMs or ROMs (or both) having an access time of 120 nanoseconds. A solution for slow RAM is that of inserting software delays after each RAM address function. A software delay can be generated by moving a register to itself - the equivalent of a NOP. The drawback to using software delays is that they increase the required program storage. If, due to generation of software delays, program storage size becomes a problem, then the designer might consider hardware generation of delays. Figure 2-22 illustrates a hardware implementation of a delay using a minimum of additional devices.

This configuration is suitable for use with RAM devices having access times of up to 250 nanoseconds. By increasing the number of flip-flops, additional hardware delays may be added, based on the RAM access time requirements.

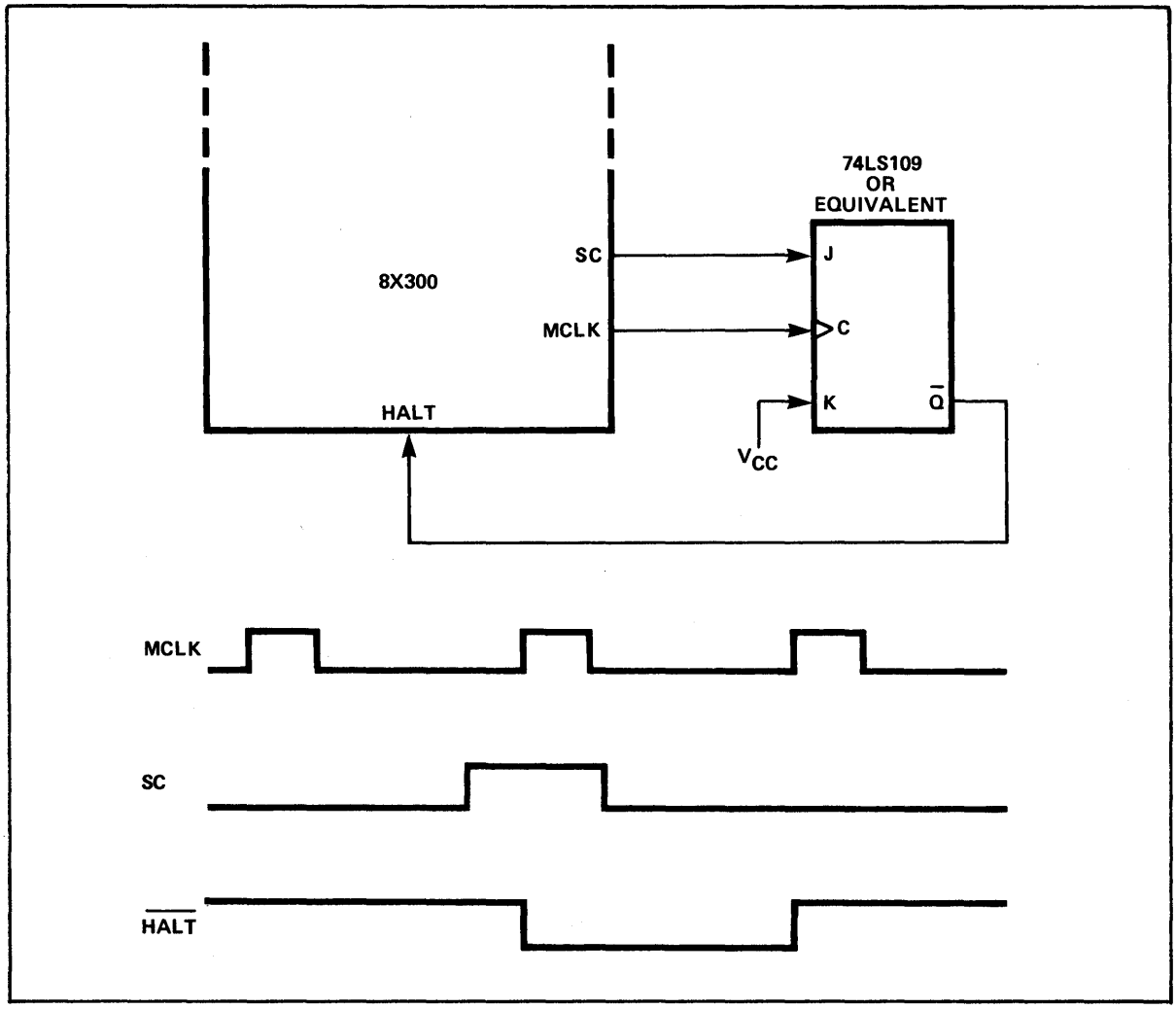


Figure 2-22. Hardware Delay Generation

A variation of this same scheme, which is suitable for use with program storage, is shown in Figure 2-23. In addition, it allows the designer to combine both bipolar and MOS ROMs within the same system. As in the previous example, a flip-flop is used to generate a hardware delay via the HALT input. Also, the program storage has been partitioned into two equal segments - one segment composed of bipolar ROMs, the other segment composed of MOS ROMs. The state of the most significant address bit determines which program storage segment is active. Other decoding schemes can be applied to partition the memory at different ratios.

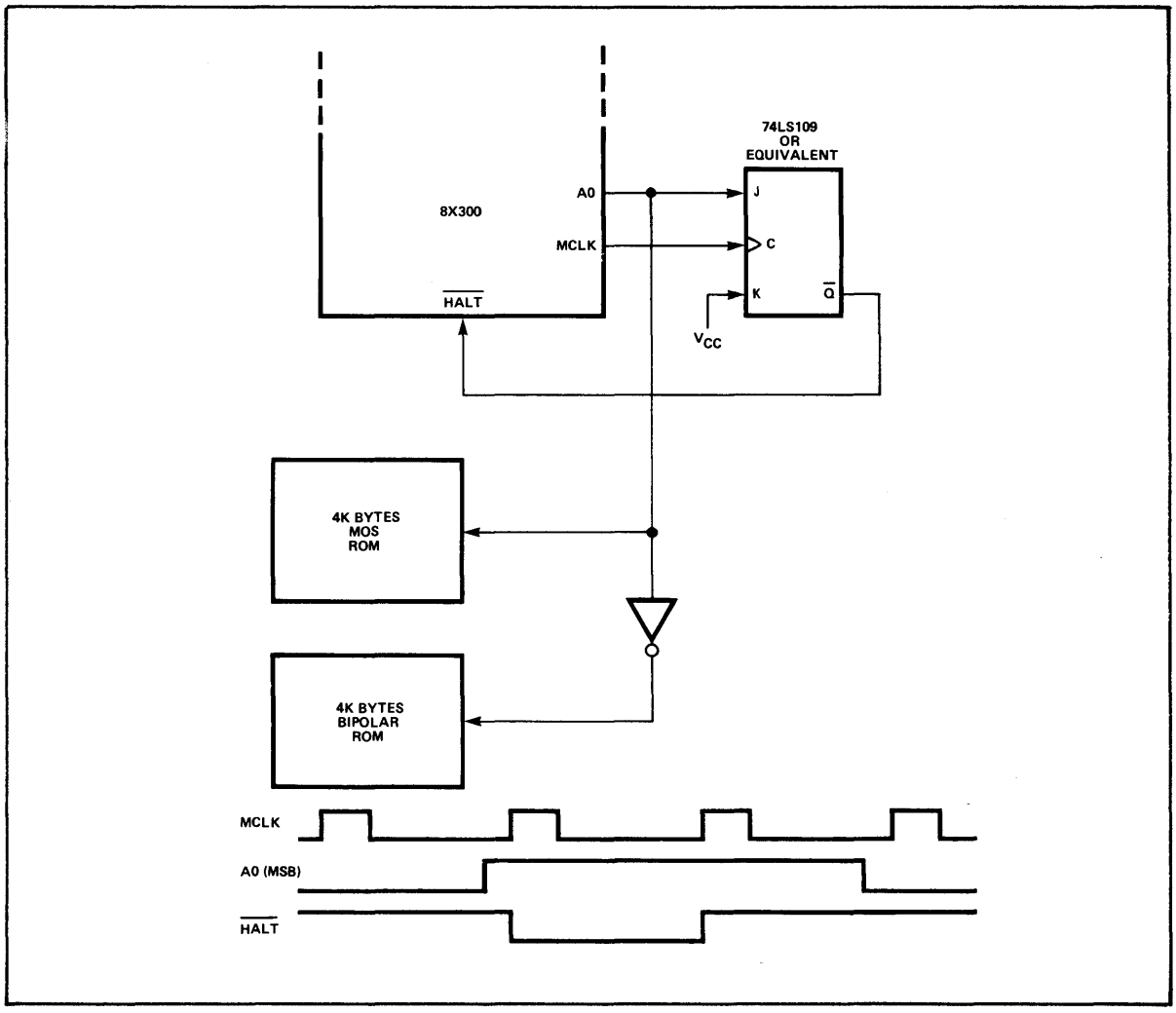


Figure 2-23. Hardware Delay Generation With Program Storage Partitioning

2.4.2 I/O Interface

Typical interfaces to the 8X300 employ the 8X32 latched addressable bidirectional I/O port. This device provides a single connection between the 8X300 I/O bus and the user status and data lines. A unique feature of this I/O port is the way in which it is addressed. Each I/O port has an 8-bit, field programmable address, used to enable the device. When the SC signal is high, data at the microprocessor port is treated as an address. if the address matches the I/O port's internally programmed address, the I/O port is latched into a selected state. The port remains selected until an address which does not match its internal address is presented, at which time the port is disabled. This feature eliminates the need for the additional hardware required for port enable decoding. The 8-bit address field, combined with the LB/RB select bits, allow selection of one of a possible 512 I/O ports.

The only disadvantage of this device is the increase in software overhead and throughput time encountered when addressing is necessary. Faster I/O selection may be implemented by adding bits to the instruction word, similar to that which was done in Figure 2-21. This technique, illustrated in Figure 2-24, allows an I/O port to be selected within the same instruction where it is used. By eliminating the need for the I/O port select instruction, important processor time is saved.

The program memory extension is followed by D-type latches, and the latches in turn followed by a decoder. The outputs of the decoder, ANDed with the bank select signal, comprise the I/O port select circuitry.

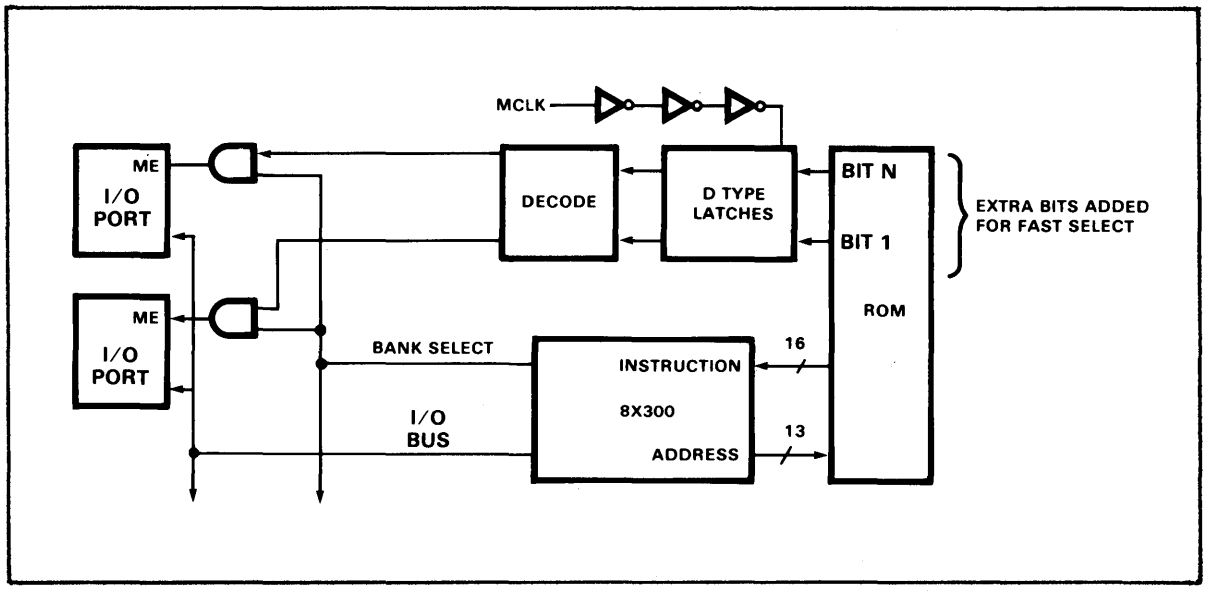


Figure 2-24. Fast I/O Select For Small Systems

This method of fast I/O port selection is suitable for use in small systems. It can be used in the same system with normal select I/O (such as 8X32s), but when using fast I/O select the programmer must ensure that a programmable I/O port has not been left selected on the bank in use.

On large systems extra delays may be encountered, with the result that it may be necessary to program the I/O port enable address in the instruction preceding its usage. A double set of D-latches are used as address hold latches to insure that the address appears sufficiently early in the instruction cycle. Refer to Figure 2-25.

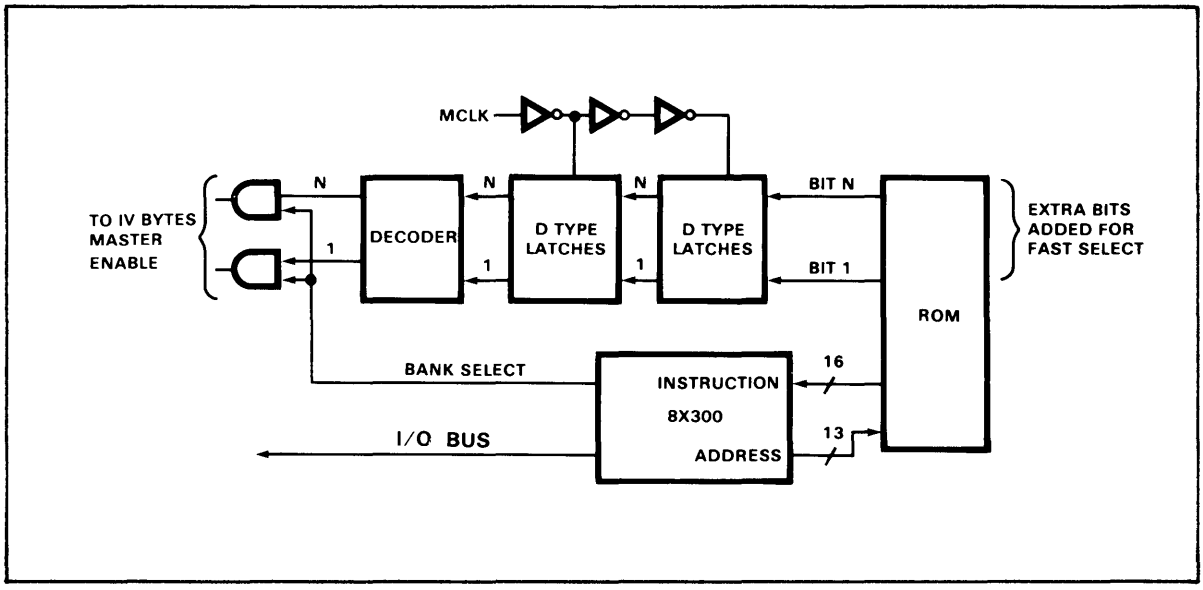


Figure 2-25. Fast I/O Select For Large Systems

In both examples, Figures 2-24 and 2-25, the program memory extension is usually made in increments of four or eight bits. If only one I/O port is to be enabled, then the remaining extension bits may be used for other control or display drive functions. If, on the other hand, a maximum of eight extension bits are decoded, this allows I/O port selection from up to 256 ports.

The preceding descriptions should not imply that up to 512 I/O devices can be directly attached to the 8X300 I/O bus. Special attention must be given to the fan-out and drive capabilities of the 8X300. When a large number of I/O or memory devices are to be driven, consideration should be given to use of the Signetics 8T39 addressable bus expander, and the Signetics 8T58 transparent bus expander. Each of these devices can buffer a block of 16 I/O ports while adding only a single load to the device driving them. The use of these bus expanders may impact system cycle time due to the added delay in the data path. When calculating allowable cycle time, the bus expander delays should be considered as additive to the I/O port delays so that a buffered I/O port actually appears as a slower I/O port.

The left bank of devices in a maximum system might consist of one 8T58 driving sixteen 8T39s, each 8T39 driving sixteen 8X32s. This results in a total of 256 usable I/O ports.

NOTE: If, when operating with asynchronous I/O input data, the data on the IV bus changes arbitrarily during the execution of an instruction such as

```
NZT     PORT, ADDR
```

where PORT references an IV bus source, the resulting branch address may be unpredictable. To avoid this, code

```
MOVE    PORT, REGX
NZT     REGX, ADDR
```

where REGX is a scratch pad register.

2.4.3 Data Conversion

In recent years there has been considerably more demand of engineers to design systems in which analog and digital disciplines coexist. Thus, the field of data conversion was born. Data conversion covers a broad range product applications including automotive and audio, and for this reason it plays a part in our everyday lives. As technology advances, it is to be expected that the product application range will continue to grow. Data conversion systems will find greater usage in a variety of fields, particularly including those of energy conservation and home entertainment.

Operating principles of Analog-to-Digital and Digital-to-Analog converters are well-defined in textbooks and industrial manuals; therefore, the following discussions are aimed at acquainting the user with the problems involved in specifying D-to-A and A-to-D converters, and to aid the user in selection of these devices.

Generally, the greatest problem facing the designer is that of device speed and/or accuracy versus cost. Quite often, as a device's speed increases its accuracy decreases. An increase in either speed or accuracy causes an increase in the cost of the device. For these reasons, a compromise is sometimes called for. The designer must consider speed, accuracy and cost in terms of system objectives. After determining the system objectives the following key specifications may be defined:

1. Accuracy - the actual output level as related to a known reference

2. Resolution - the smallest output increment that can be distinguished

3. Speed - the length of time necessary to perform the desired function (typically, the sampling rate)

4. Stability - the lack of sensitivity of a converter's characteristics to time and temperature.

Once these specifications have been defined then the general price range of the device can be determined.

When interfacing data conversion devices with the 8X300, it must be realized that the 8X300 operating at maximum speed (250 nanosecond instruction cycle) is faster than the majority of data conversion devices currently available. The software and hardware must be arranged in a manner that will economically accommodate the data conversion device being used. For example, the designer might elect to use a Digital-to-Analog converter with a settling time of 500 nanoseconds. System requirements might dictate that the analog output level be held stable for one microsecond. This means that the digital inputs to the D-to-A converter must be held stable for a minimum of 1.5 microseconds. This is easily accomplished using a latched addressable I/O port such as the Signetics 8X32. Refer to Figure 2-26. The digital input word to the Signetics NE5008 is latched into I/O port number three.

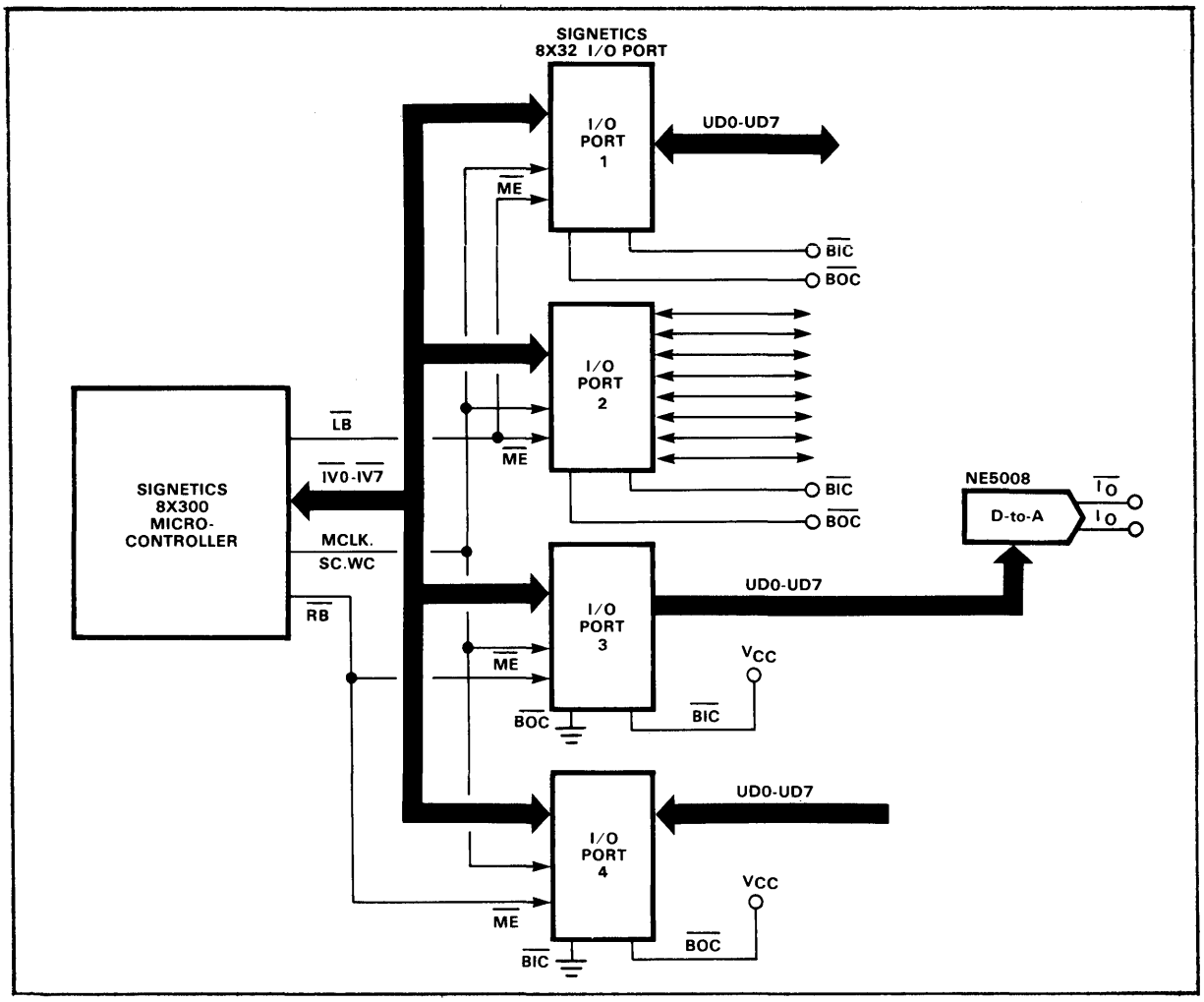


Figure 2-26. Digital-to-Analog Converter Configuration

The above described configuration is easily adapted for use as a successive-approximation Analog-to-Digital converter, and requires a minimum of additional hardware. Referring to Figure 2-27, a comparator has been placed on the output of the D-to-A converter. The output of the comparator is then connected to the UD-7 (LSB) input of I/O port number four.

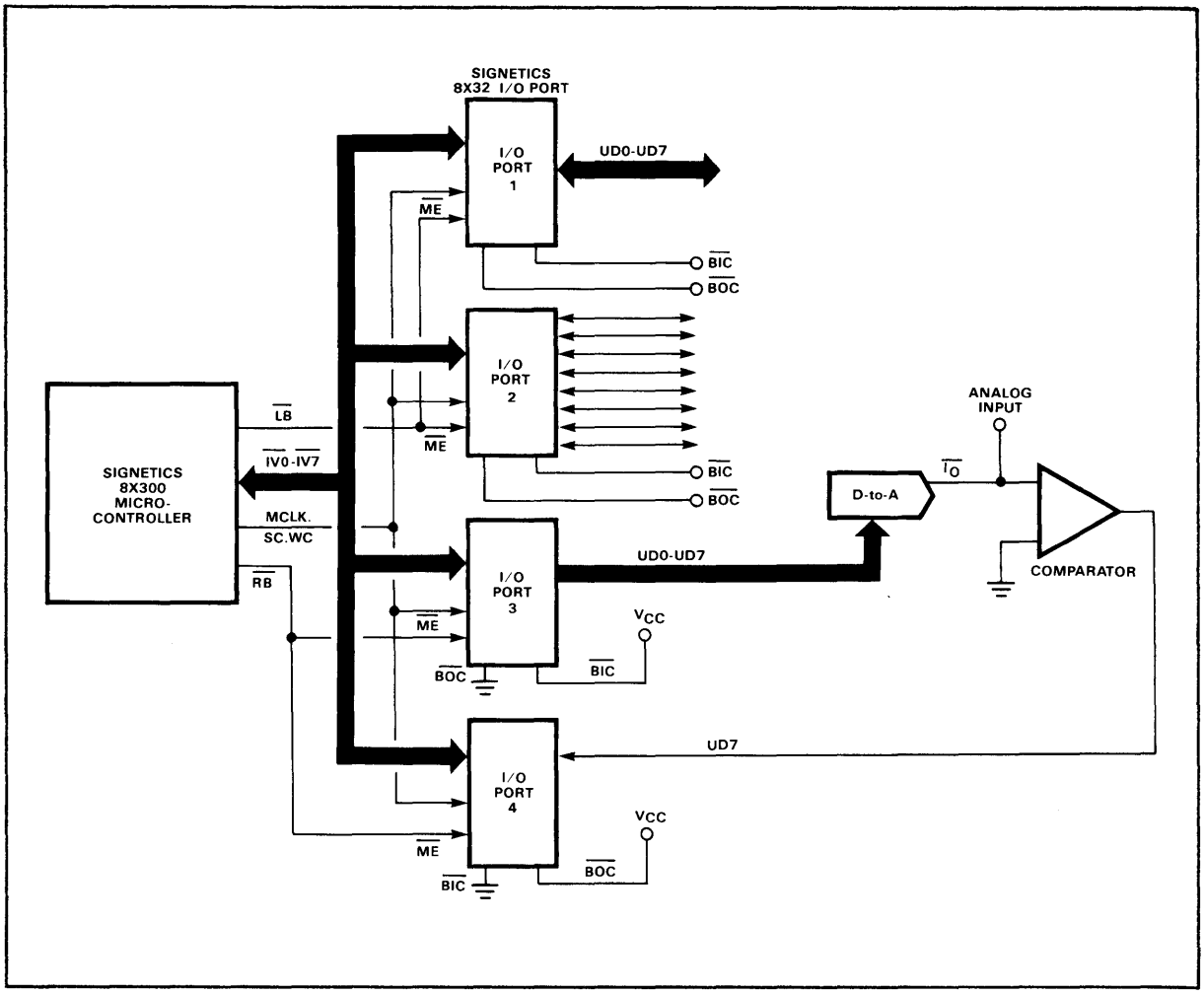


Figure 2-27. Successive - Approximation
Analog-to-Digital Converter Configuration

The objective is to successively turn-on each bit of the D-to-A converter, starting with the most significant bit. An internal register of the 8X300 may be used as a successive-approximation register. As each bit is output to the comparator, it is checked for a compare condition with the analog input. If the D-to-A output compares low with respect to the analog input, a high is stored in the successive-approximation register for this respective bit position. If the D-to-A output compares high with respect to the analog input, a low is stored. After comparison and storage is completed for this bit posititon, the program loops back and the procedure is repeated for the next least significant bit. This process is repeated until all input bits to the D-to-A converter have been processed. Upon completion of the process, the digital word

stored in the successive-approximation register represents the analog input value with respect to the D-to-A converter. The successive-approximation routine is shown in Figure 2-28, Successive-Approximation Flowchart.

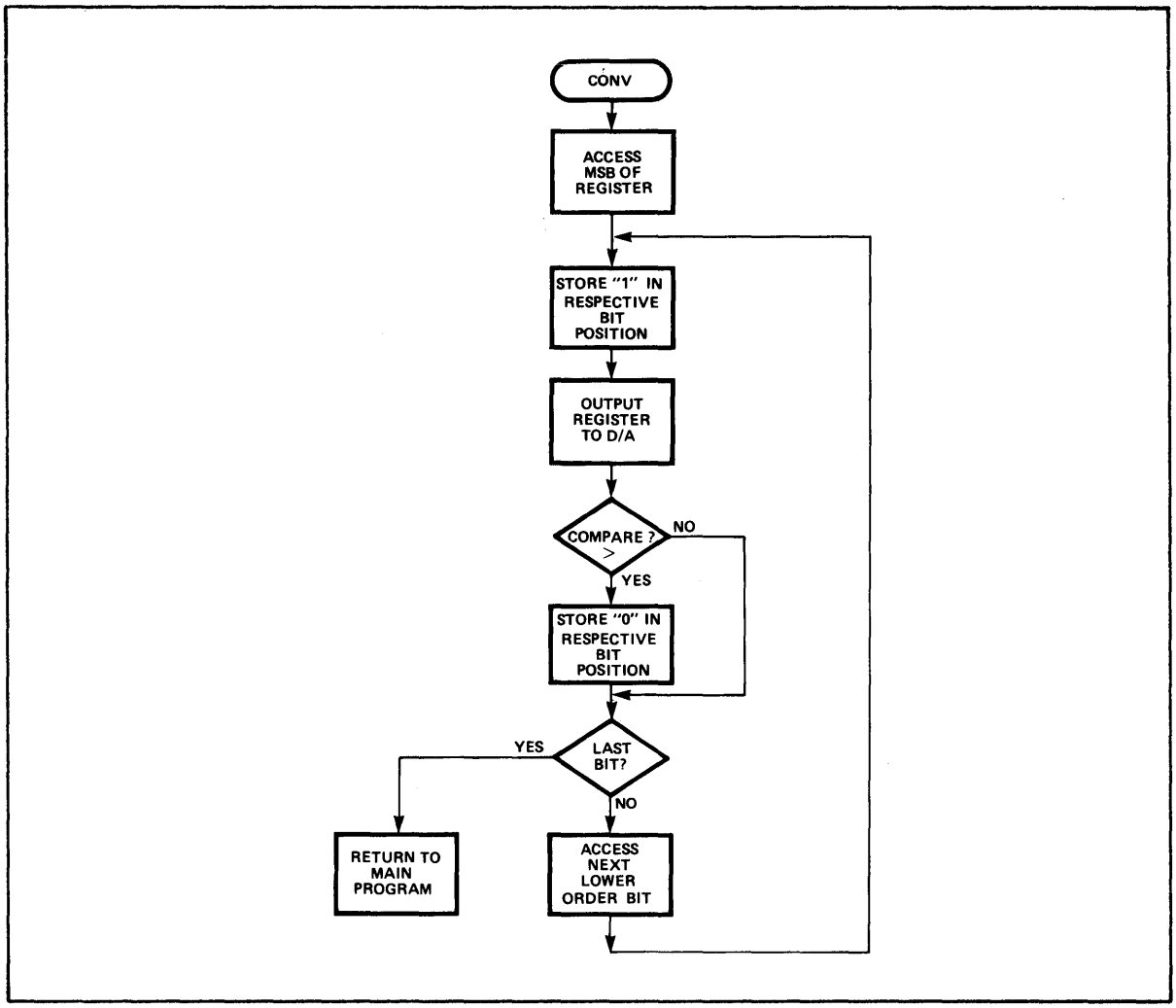


Figure 2-28. Successive-Approximation Flowchart

This type of converter is somewhat slow, with speeds in the range of tens of microseconds. The actual speed is determined by a number of factors, including the settling time of the D-to-A converter and the resolution (8-, 10-, or 16-bits) desired. The latter affects the number of approximations that must be made.

2.5 8X300 INTERRUPT STRUCTURE

Although the 8X300 microprocessor is a very fast device and in many cases external events occurring randomly and influencing the program flow may be discovered by means of a polling technique, there are instances when the system must react immediately to such events.

Because the 8X300 has no interrupt facility, it is necessary to add hardware to obtain such a facility. This hardware should perform the following actions on receipt of an interrupt signal:

- save the current Program Counter contents
- retain the last used IVL and IVR addresses
- make the program jump to a fixed (interrupt) address
- detect a return instruction
- restore the IVL and IVR addresses
- restore the original contents of the Program Counter

Any registers which are used in the interrupt routine should also be saved and restored on return; this is done in software as is the restoration of the IVL and IVR addresses, although the saving is done by hardware. To save and restore various items, a RAM of at least ten bytes is required. However, it is assumed that a 256 byte RAM is available, connected to the right bank.

To provide an interrupt facility for 8X300 systems, it is necessary to add the following hardware:

a) EXECUTE decoding - Interrupts must be inhibited during an EXECUTE instruction

b) Return decoding - 717777 (JMP 17777) is used as a return instruction. The address 17777 must be decoded

c) Interrupt control - The Interrupt signal has to be synchronized with the Master Clock, and various control signals have to be generated

d) Return Address Register

e) Interrupt Vector logic - This logic forces a jump to the interrupt address on the instruction bus

f) IVL, IVR Save logic - It is necessary to retain the last IVL and IVR addresses selected before the interrupt, to be able to restore these addresses on return

g) Priority logic - If more than one interrupt request is given at one time, the priority between them has to be determined and a different interrupt address allocated to each interrupt level.

Figure 2-29 shows the block diagram of the hardware.

On a right bank select instruction, the address is stored in the 8X31 latches. The three-state outputs of the latches drive the address inputs of the eight RAM circuits. The data connections of the RAM and the I/O ports are wired to the I/O bus.

In the interrupt logic, an interrupt signal is sent to the Interrupt Control device. A Return Address Clock is then given to store the current address in the Return Address Register. Generally, the current address is equal to the contents of the Program Counter (PC). Only when an Execute instruction occurs are the contents of the address bus not equal to that of the PC. This condition is detected by decoding the first three bits of the instruction bus, and defers the interrupt. When the interrupt is given, the PROM is disabled and a JMP instruction to the interrupt address is forced on the instruction bus by the Interrupt Vector, coincident with a Vector Enable signal.

During the running of the main program, prior to the interrupt, IVL and IVR addresses are stored in RAM locations 255 and 254 respectively. This is done when the Select Control (SC) signal is active. When an interrupt occurs, the Save Logic is disabled and the last IVL and IVR addresses are retained in locations 255 and 254. Just before the return, these addresses must be fetched to reselect the original left and right I/O ports (or RAM location).

The return instruction is 717777. The address 17777 is decoded and again causes a PROM Disable. The contents of the Return Address Resister are forced on the instruction bus by means of the RETURN ADDRESS ENABLE signal while the three most significant bits are "1". Thus, a JMP to the return address is made.

An INTERRUPT INHIBIT signal may be generated by software and taken from a bit of an I/O port, or it may be generated somewhere else in the system. The interrupt signal is inhibited from generating an interrupt, but the interrupt may remain pending until the INTERRUPT INHIBIT signal is removed.

when an interrupt occurs, a signal INTERRUPT ACKNOWLEDGE is generated. The timing depends upon whether a handshake interrupt or an edge and stobe interrupt is chosen.

The implementation of the interrupt facility as described requires 20 DIL devices.

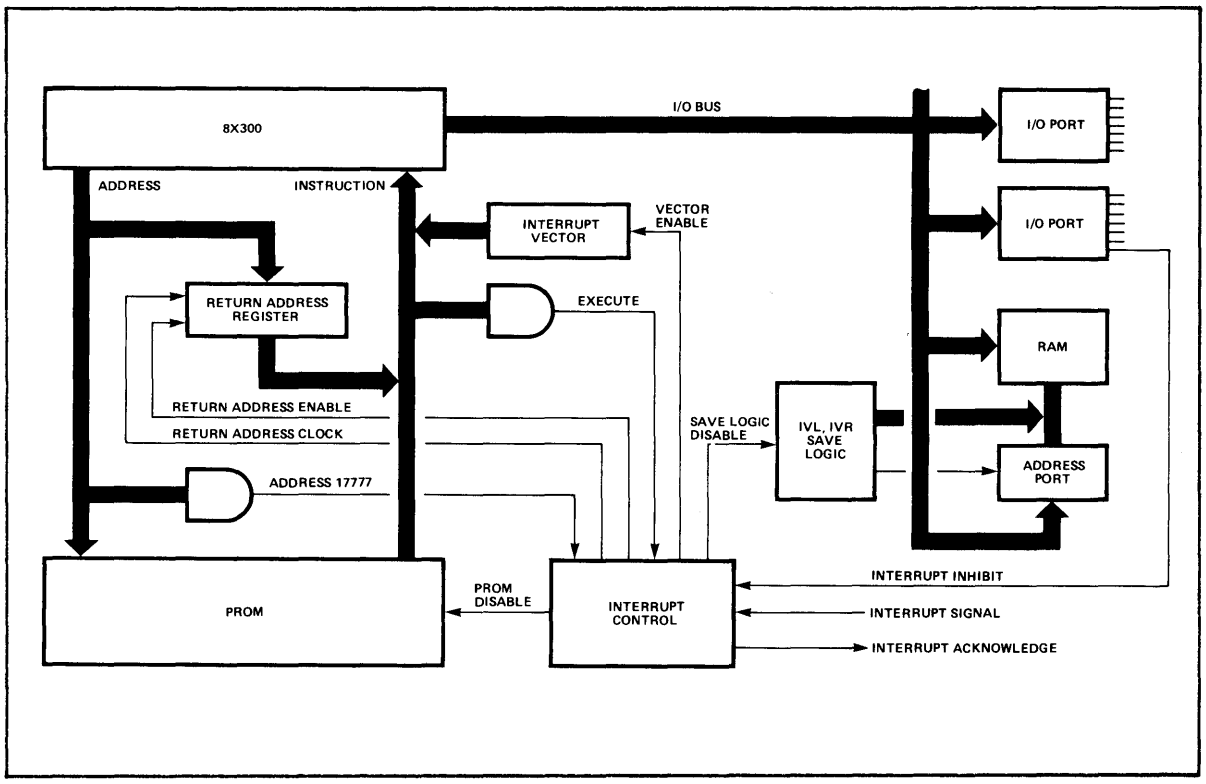


Figure 2-29. Typical Interrupt Block Diagram

2.5.1 Execute and Return Decoding

The EXECUTE and RETURN decoding logic is shown in Figure 2-30. When an EXECUTE instruction (opcode 100) is encountered, the content of the address bus is not equal to that of the Program Counter. At that moment no interrupt may occur. To detect whether an EXECUTE instruction is being performed, however, requires a delay of the trailing edge of MCLK because the instruction bus becomes valid after this trailing edge. Prior to the leading edge of the next MCLK the address changes, causing the instruction to also change.

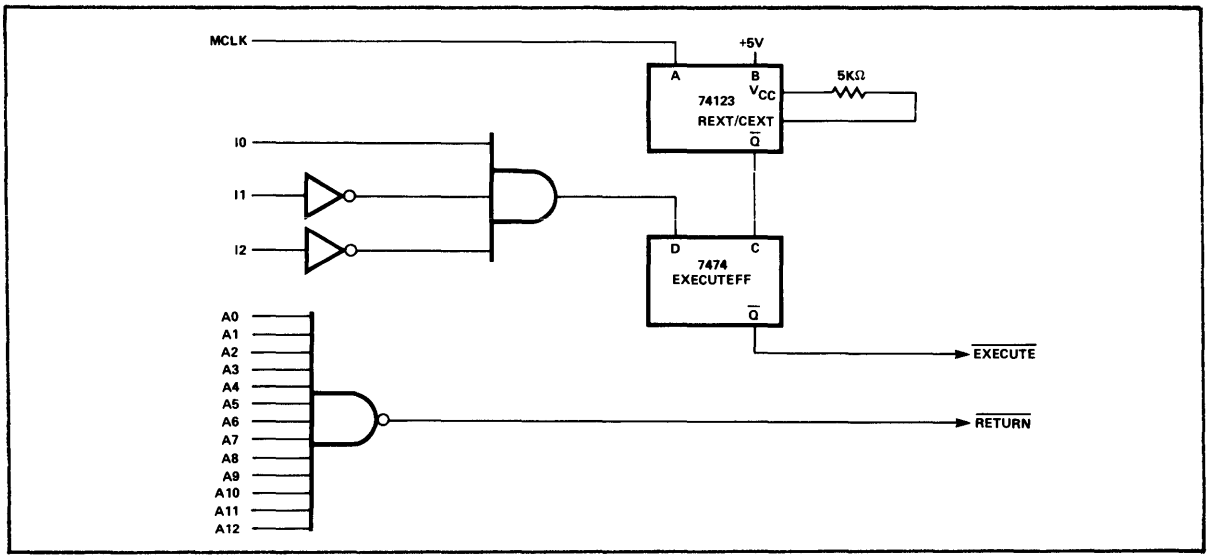


Figure 2-30. Execute and Return Decoding

The instruction is decoded and clocked into a flip-flop about 50 nanoseconds after the trailing edge of MCLK. The output of the flip-flop is used to stop the interrupt (see Figures 2-31 and 2-32).

To save some hardware, it is also possible to inhibit the interrupt by means of software. In that case the hardware decoding and clocking is not required, but before each EXECUTE instruction, one has to make sure that the INTERRUPT INHIBIT signal is active.

At the end of the interrupt routine, an instruction, 717777, has to be given to indicate a return. The address 17777 is on the address bus and is decoded to form the RETURN signal. If 4K or less of program memory is used, the decode logic (13-input gate) is no longer necessary since the MSB(A0) alone may be used to signal a return, that is, a jump to any address 4096.

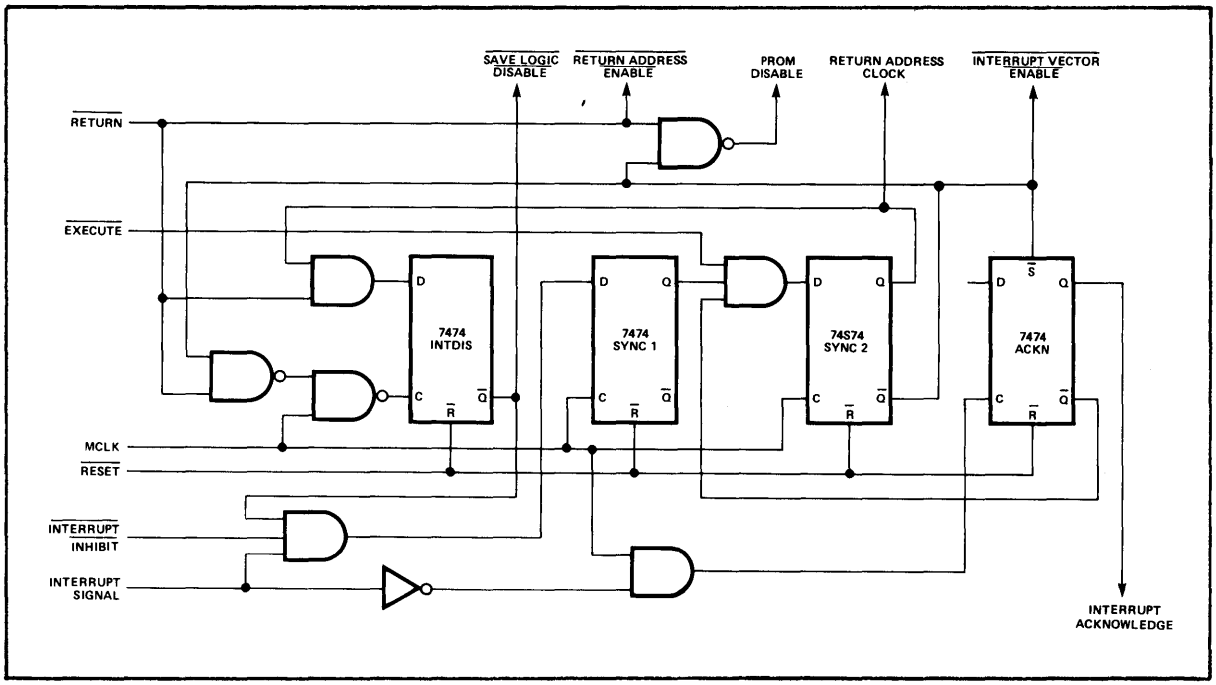


Figure 2-31. Interrupt Control (Handshake)

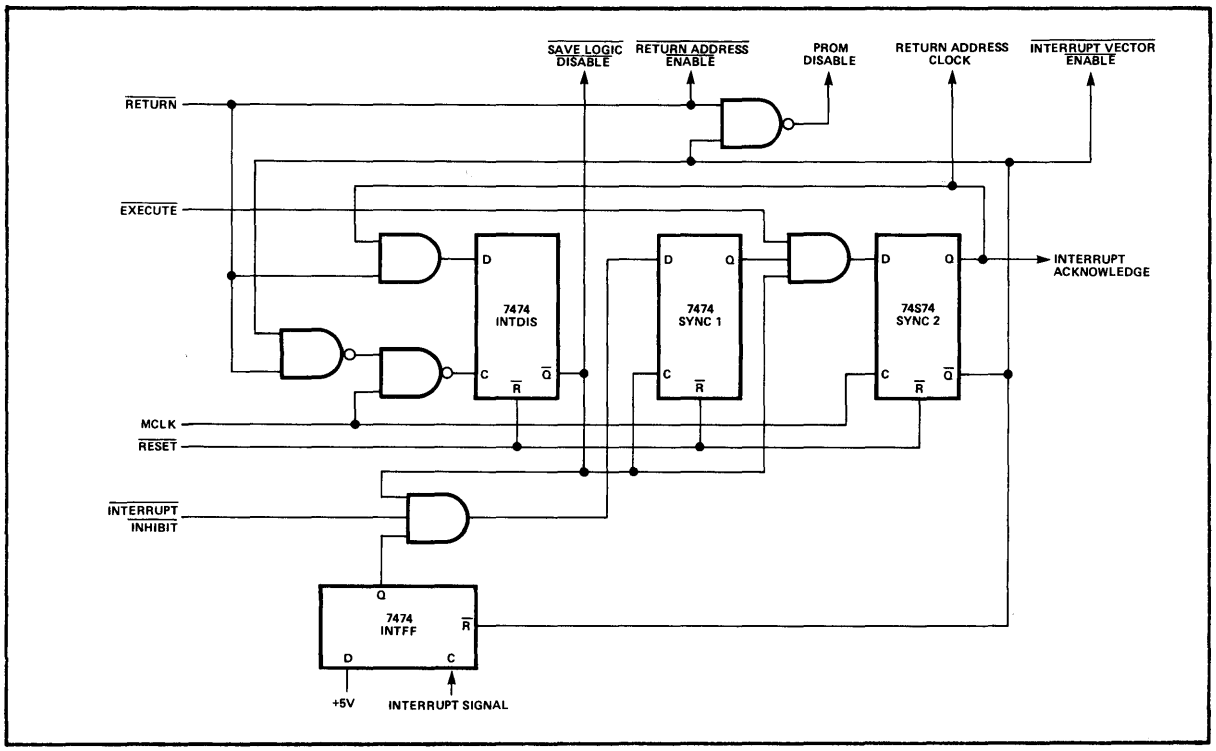


Figure 2-32. Interrupt Control (Edge And Strobe)

2.5.2 Interrupt Control (Handshake)

The timing of the handshake procedure is shown in Figure 2-33. The time between the activation of the INTERRUPT signal and the leading edge of INTERRUPT ACKNOWLEDGE (time A) is T_{cycle} + 10 nanoseconds minimum, and 3 x t_{cycle} + 60 nanoseconds maximum, (when Execute has to be taken into account).

The time between the de-activation of the INTERRUPT signal and the trailing edge of INTERRUPT ACKNOWLEDGE (time B) is 10 nanoseconds mimimum, and 3/4 x t_{cycle} + 50 nanoseconds maximum.

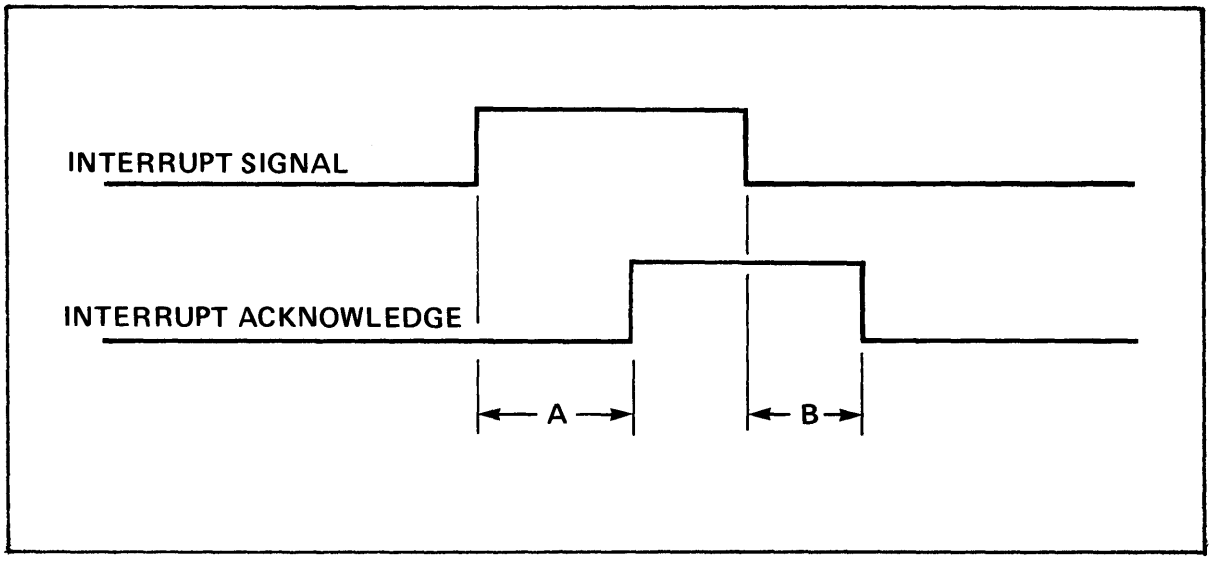


Figure 2-33. Handshake Timing

The INTERRUPT signal must remain active until the INTERUPT ACKNOWLEDGE goes high. The INTERRUPT ACKNOWLEDGE is de-activated when the INTERRUPT signal is no longer active.

When an INTERRUPT signal occurs with no INTERRUPT INHIBIT present, the signal is synchronized in flip-flops SYNC1 and SYNC2 because the output of SYNC1 may be a very small pulse if the INTERRUPT signal occurs very near the leading edge of MCLK. If there is an EXECUTE instruction, the setting of SYNC2 is inhibited for one cycle. When SYNC2 is set, the signals PROM DISABLE, RETURN ADDRESS CLOCK and INTERRUPT VECTOR ENABLE become active. The acknowledge flip-flop (ACKN) is also set.

The interrupt disable flip-flop (INTDIS) is set to 1 at the trailing edge of MCLK. The signal SAVE LOGIC DISABLE may only become active after MCLK, as the previous instruction might have been an I/O select instruction and the I/O address still has to be saved in the RAM. SYNC1 and SYNC2 are reset with the next MCLK because of INTDIS and ACKN, respectively. ACKN remains set as long as the INTERRUPT signal is active, and is reset at the MCLK following de-activation of the INTERRUPT signal.

When a return instruction is given (717777) INTDIS is reset, the PROM disabled and the output of the Return Address Register is forced on the instruction bus. Thus, a jump to the return address occurs. The SAVE LOGIC DISABLE signal is de-activated, so I/O addresses are saved again.

2.5.3 Interrupt Control (Edge and Strobe)

The interrupt control logic is shown in Figure 2-32, with the timing shown in Figure 2-34.

The width of the INTERRUPT signal (time A in Figure 2-34) must be 25 nanoseconds, minimum.

The time between the activation of the INTERRUPT signal and the leading edge of INTERRUPT ACKNOWLEDGE (Time B) is t_{cycle} + 10 nanoseconds, minimum, and 3 x t_{cycle} + 60 nanoseconds, maximum (when EXECUTE has to be taken into account).

The width of INTERRUPT ACKNOWLEDGE (time C) is t_{cycle}.

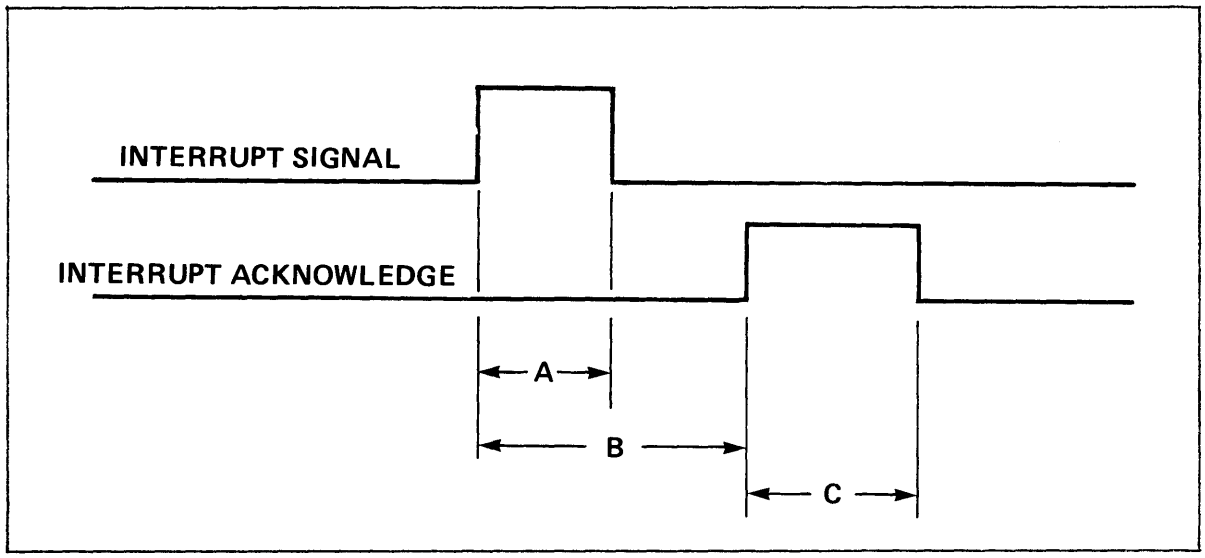


Figure 2-34. Edge and Strobe Timing

The positive edge of the INTERRUPT signal triggers the interrupt flip-flop (INTFF). Then the synchronizaion is accomplished in the same way as in the handshake procedure. INTERRUPT ACKNOWLEDGE is derived from the SYNC2 output.

2.5.4 Return Address Register And Interrupt Vector

The Return Address Register is built up of four 8T10 three-state quad D-type flip-flops. Refer to Figure 2-35. The inputs are connected to the address bus, but the three most significant bits are tied to "1", providing a JMP opcode (111) at the outputs, which are connected to the instruction bus.

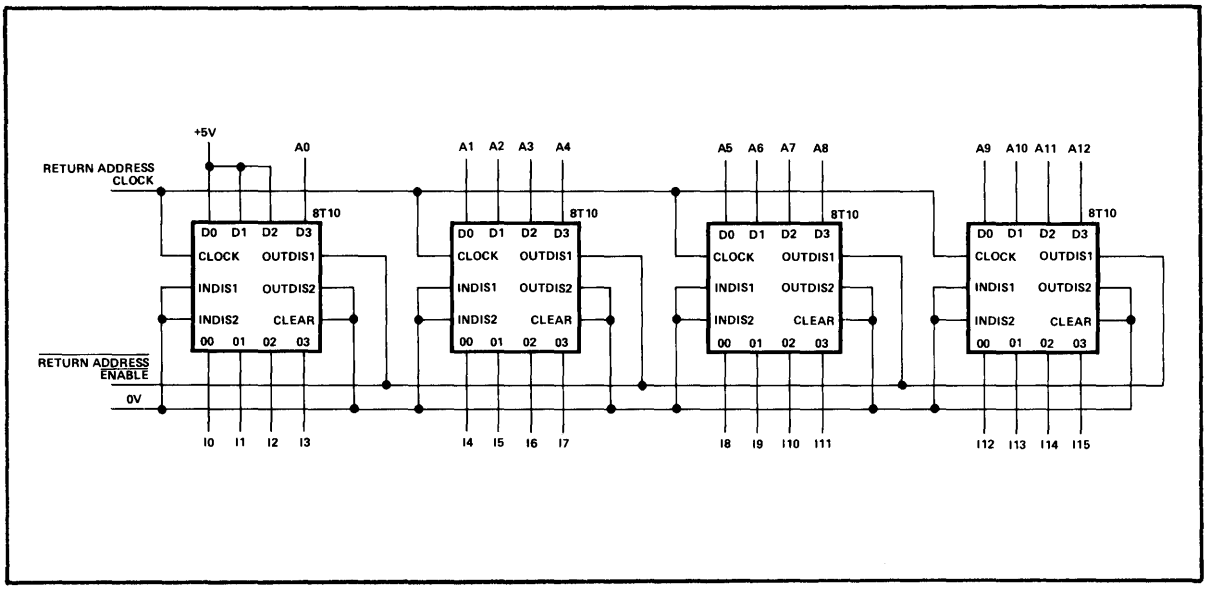


Figure 2-35. Return Address Register

The Interrupt Vector consists of 8T97 tri-state buffers. Refer to Figure 2-36. The inputs may be chosen to be 0 or 1 by means of straps, jumpers, or switches to form the start address of the interrupt routine. The three most significant bits, however, must be "1" to provide a JMP opcode.

In the case of eight levels of interrupt priority, the three least significant bits form the interrupt code.

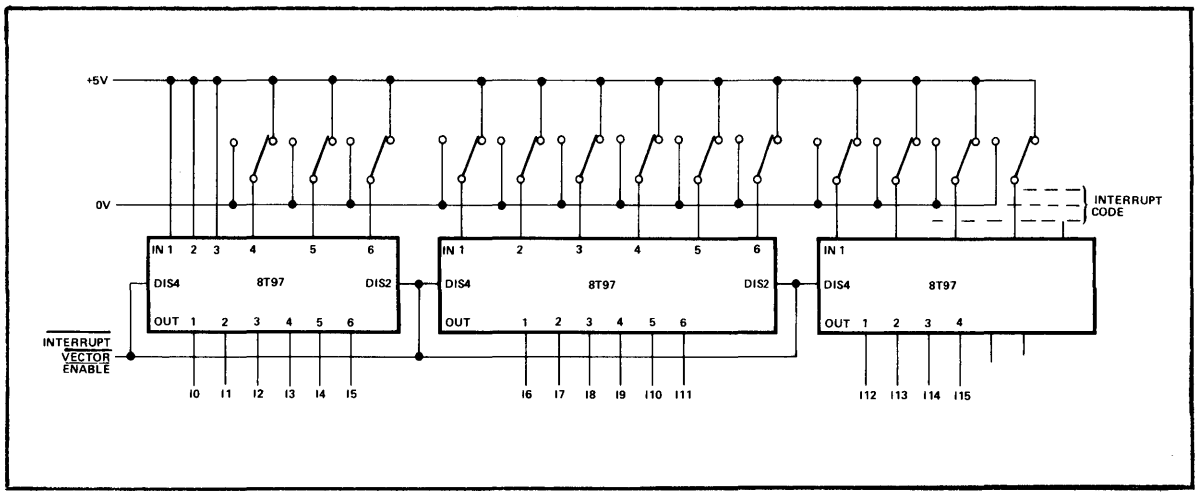


Figure 2-36. Interrupt Vector Logic

At the start of an interrupt, the RETURN ADDRESS CLOCK loads the Return Address Register with the current address, and the INTERRUPT VECTOR ENABLE forces a jump to the interrupt address on the instruction bus.

At the return, RETURN ADDRESS ENABLE forces a jump to the original address on the instruction bus.

2.5.5 IVL, IVR Storage Logic

The IVL and IVR storage logic is illustrated in Figure 2-37. With a 256 byte RAM (eight 82S116's), an 8X31 8-bit latched bidirectional I/O port may be used as address latches. When a right I/O select occurs, the I/O address is latched in the 8X31, and in a following instruction data can be transferred in or out of the RAM.

With the interrupt circuitry, the 8X31 retains its function as RAM-address latches, but with an I/O select instruction (SC active) the output is disabled and the 8T97 three-state buffers force the address 255 or 254 on the RAM address bus. If the SAVE LOGIC DISABLE is not active, the I/O address on the data bus is written into the RAM at address 255 (left IV) or 254 (right IV) by means of the Write Enable input (WE). If SAVE LOGIC DISABLE is active,

the writing of I/O addresses is inhibited. Only data that has to be written into the RAM at a specifc address will be written by means of the Write Control (WC) signal, if the RB signal is active to enable the RAM by means of the CE1 input. Only one of the eight RAM cells, 82S116, has been drawn in Figure 2-37.

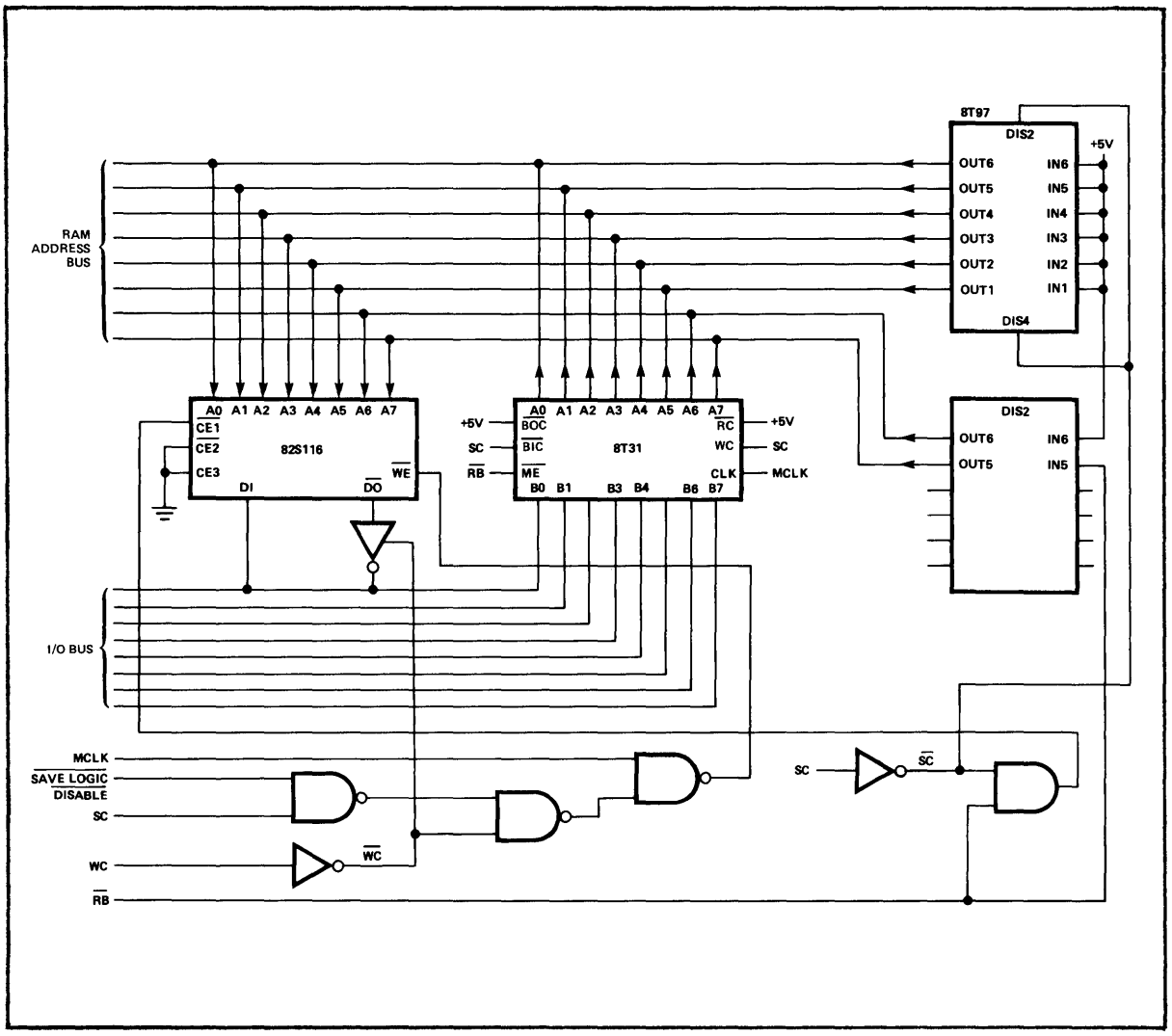


Figure 2-37. IVL, IVR Storage Logic

2.5.6 Priority Interrupt Levels

Figure 2-38 shows part of the logic for an eight level interrupt priority. The remainer of the interrupt control logic is depicted in Figure 2-32. (Edge and Strobe), with the exception that one interrupt flip-flop (INTFF) is replace by a number of flip-flops, INTFFA, INTFFB, etc. A handshake procedure is also possible.

In total, eight interupt requests may be given. As an example, INTREQA is chosen as an edge-trigger interrupt while INTREQH is treated in a handshake procedure.

The eight interrupt request signals, INTREQA through INTREQH, either trigger an interrupt flip-flop or feed directly into a masking gate, with which it is possible to inhibit one or more chosen levels.

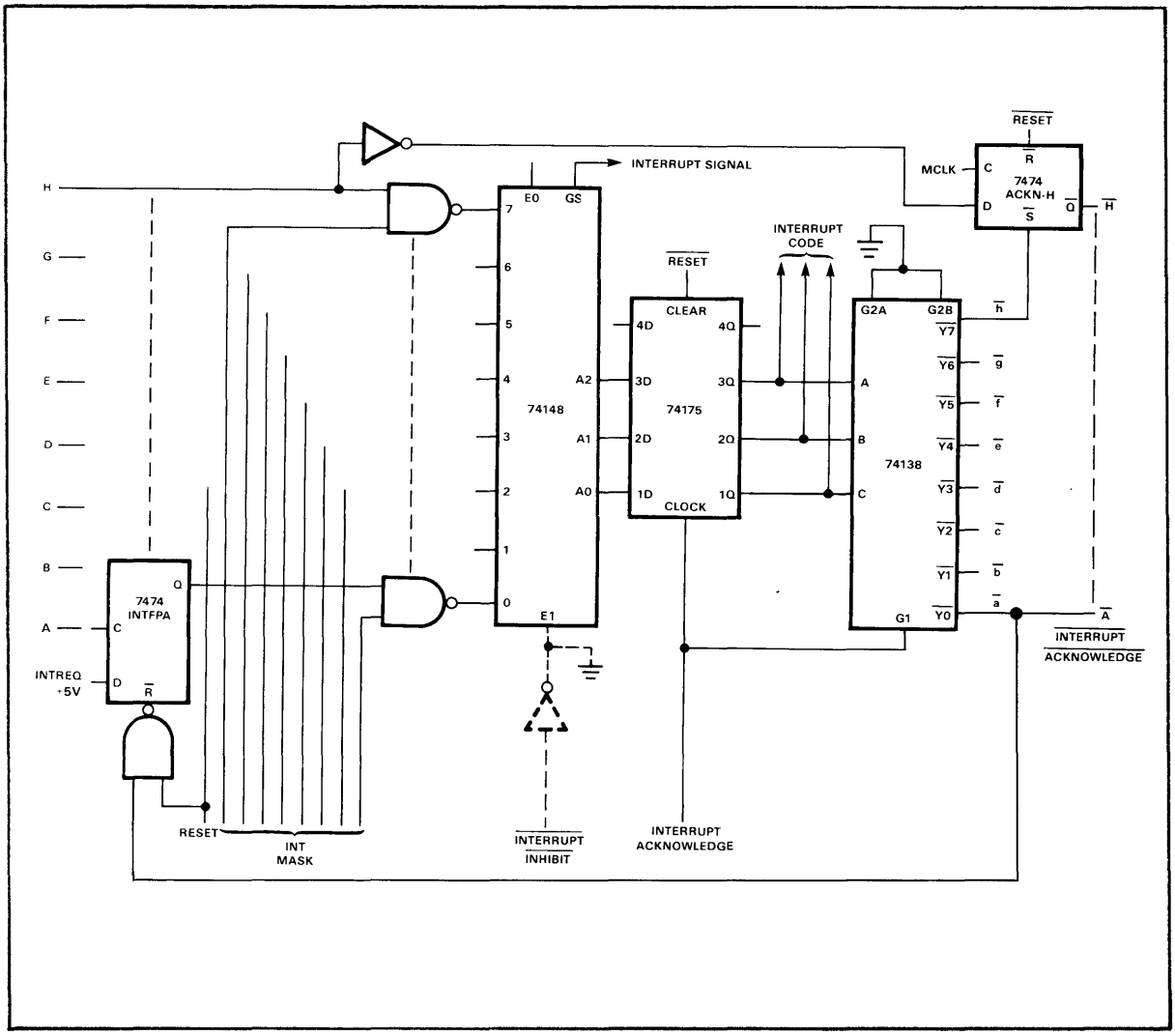


Figure 2-38. Priority Interrupt Level Logic

INT MASK signals come from an I/O port, so that masking may be done by software. A general interrupt inhibit may be applied if a signal is fed to the EI input of the encoder. If this is not requried, the EI input should be tied to ground. The priority of the incoming interrupts is determined in the priority encoder. If at least one interrupt request is active at the input of the encoder, an INTERRUPT signal is generated for the Interrupt Control (Figure 2-32). The code of the active interrupt with the highest priority is clocked into the flip-flops of the 74175, coincident with the INTERRUPT ACKNOWLEDGE from Interrupt Control.

The outputs of the flip-flops are used as the three least significant bits of the Interrupt Vector to provide eight different start addresses for the interrupt routines.
They are also decoded to form an acknowledge signal indicating which interrupt request has been served. In the handshake procedue this signal sets the acknowledge flip-flop (ACKN). The acknowledge flip-flop is reset only when the INTREQ signal is de-activated. With this logic, however, the INTREQ signal must be de-activated before the end of the interrupt routine (return instruction), otherwise, the routine may be repeated.

APPENDIX A

8X300 FAMILY DATA SHEETS

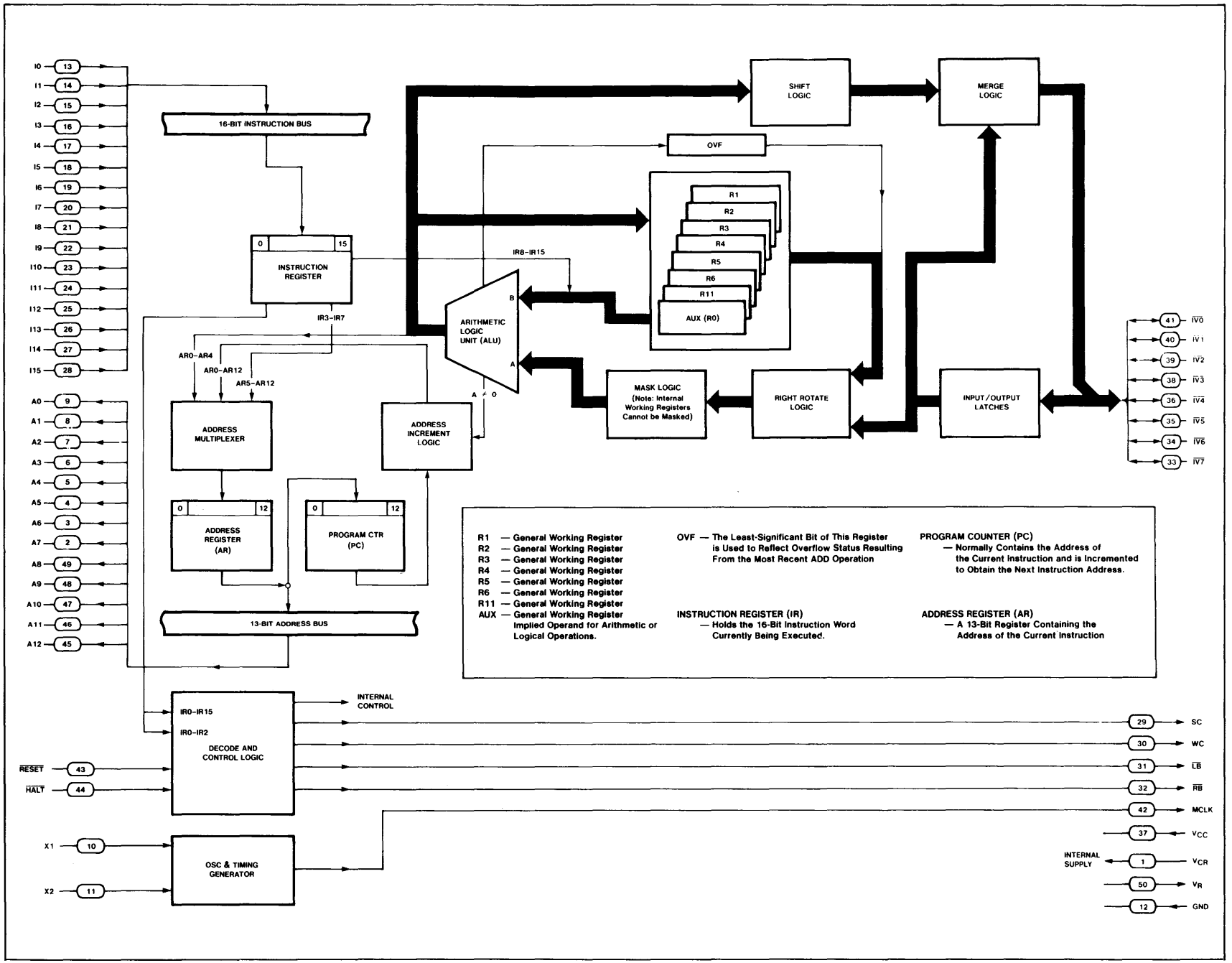


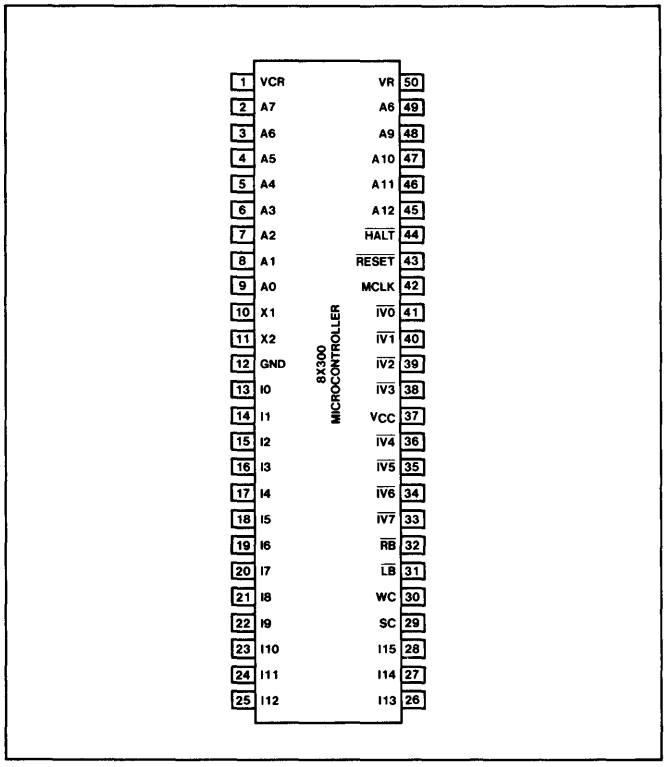
Figure 1. CPU Architecture and PIN Designations For 8X300 Microcontroller

ARCHITECTURAL OVERVIEW

The Signetics 8X300 Microcontroller (Figure 1) is a high-speed bipolar microprocessor implemented with low-power Schottky technology. The 8X300 brings together all the qualities needed—SPEED, FLEXIBILITY, and ECONOMY—for systems design in the many areas that require reliable bit stream management. Consider!—5V operation, TTL bus compatibility, and an on-chip clock—the result, a system with fewer parts. Consider!—the inherent power of LSI logic (programmable Rotate, Mask, Shift, and Merge functions in the data-processing path) and the ability to Fetch, Decode, and Execute a 16-bit instruction in a minimum of 250-nanoseconds—the result, a system with superior bit handling capabilities. Consider!—the 250ns cycle time in conjunction with extended microcode—the result, the flexibility of bit-slice devices with the programming ease of MOS microprocessors. Now, consider the results!—a device tailored to bit-stream management in the areas of Industrial Control, Input/Output Control, and Data Communications.

The 8X300 uses three separate buses—one for 13-bit instruction addresses, one for 16-bit instructions, and a bidirectional 8-bit input/output data bus; except for the I/O bus, there are no time multiplexing of functions.

PIN CONFIGURATION



FEATURES

- **Fetch, Decode, and Execute a 16-bit instruction in a minimum of 250-nanoseconds (one machine cycle)**
- **Bit-oriented instruction set (addressable single-or-multiple bit subfields)**
- **Separate address, instruction, and I/O buses**
- **Source/destination architecture**
- **On-Chip oscillator and timing generation**
- **Eight 8-bit working registers**
- **TTL inputs and outputs**
- **BiPolar Low-Power Schottky technology**
- **3-State I/O bus**
- **Single +5V supply**

ORDERING INFORMATION

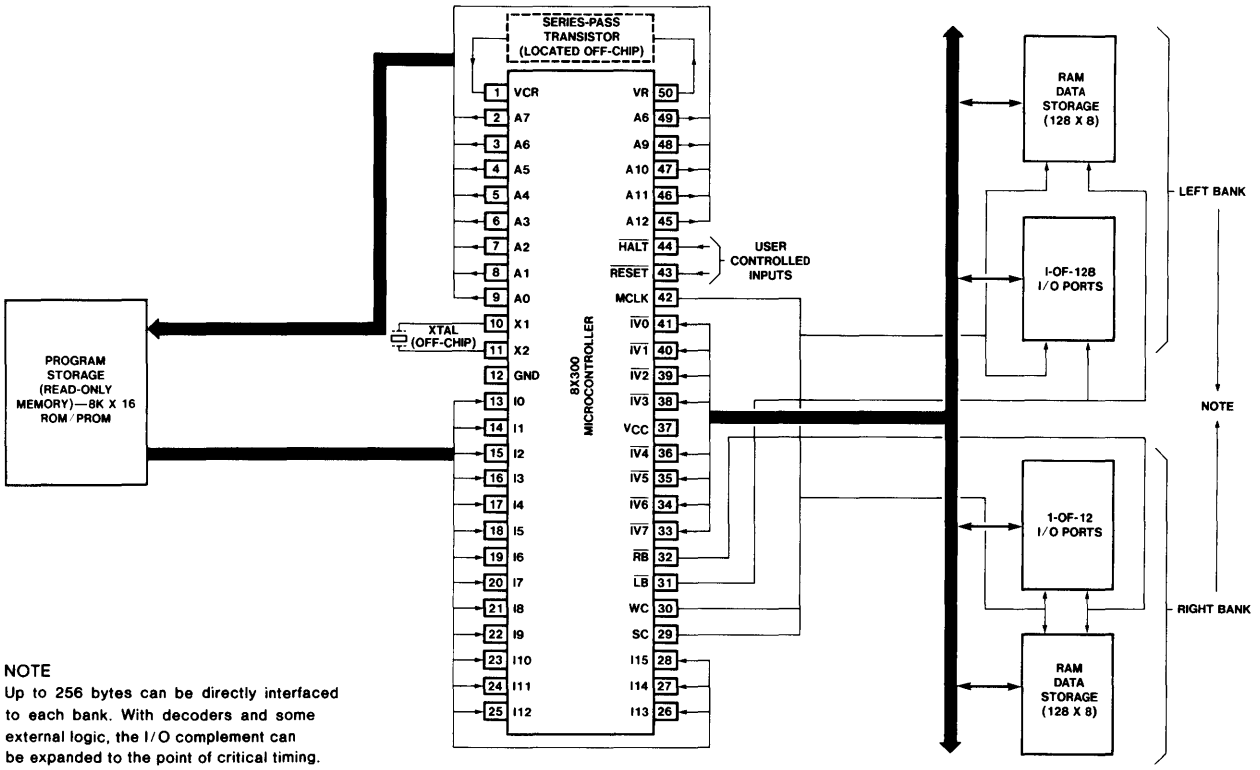
Commercial

Order number: N8X300I
Packaging information: Refer to Signetics price list
Supply voltage: 5V (±5%)
Operating temperature range: 0°C to +70°C

Military

Order number: S8X300-1
Packaging information: Refer to Signetics price list
Supply voltage: 5V (±5%)
Operating temperature range: −40°C to +100°C

Order number: S8X300-2
Packaging information: Refer to Signetics price list
Supply voltage: 5V (±10%)
Operating temperature range: −20°C to +100°C



NOTE
Up to 256 bytes can be directly interfaced to each bank. With decoders and some external logic, the I/O complement can be expanded to the point of critical timing.

PIN NO.	IDENTIFIER	NAME AND FUNCTION	ACTIVE STATE
2-9/45-49	A0-A12	**Program Address Lines:** These outputs permit direct addressing of up to 8192 words of program storage. A high voltage level equals a binary "1"; A12 is Least Significant Bit.	High
13-28	I0-I15	**Instruction Lines:** These input lines receive 16-bit instructions from program storage. A high voltage level equals a binary "1"; I15 is Least Significant Bit.	High
33-36 38-41	$\overline{IV0}$-$\overline{IV7}$	**Input/Output Bus:** These bidirectional three-state lines communicate with up to 512 I/O devices (256 per bank). A low voltage level equals a binary "1"; $\overline{IV7}$ is Least Significant Bit.	Low
10 & 11	X1 & X2	**Connections** for a capacitor, a series-resonant crystal, or an external clock source with complementary outputs. For precise frequency control, a crystal or external source is required.	—
42	MCLK	**Master Clock:** This output is used for clocking I/O devices and/or synchronization of external logic.	High
30	WC	**Write Command:** When signal is high (binary 1), data is being output on pins $\overline{IV0}$-$\overline{IV7}$ of I/O bus.	High
29	SC	**Select Command:** When signal is high (binary 1), an address is being output on pins $\overline{IV0}$-$\overline{IV7}$ of I/O bus.	High
31	$\overline{LB}$	When the $\overline{LB}$ signal is low (binary 0), any one of up-to-256 I/O devices (or memory locations) in the left bank can be accessed. When the address of a particular device (or memory location) matches the address on the IV bus, that particular device (or memory location) is enabled and selected for input/output operations. All addresses on the left bank that do not match are deselected.	Low
32	$\overline{RB}$	When the $\overline{RB}$ signal is low (binary 0), any one of up-to-256 I/O devices (or memory locations) in the right bank can be accessed. When the address of a particular device (or memory location) matches the address on the I/O bus, that particular device (or memory location) is enabled and selected for input/output operations. All addresses on the right bank that do not match are deselected.	Low
43	$\overline{RESET}$	When reset input is low (binary 0), the microcontroller is initialized—sets Program Counter/Address to zero and inhibits MCLK output.	Low
44	$\overline{HALT}$	When halt input is low (binary 0), internal operation of microcontroller stops at the start of next instruction. The stop function does not inhibit MCLK or affect any internal registers.	Low
50	VR	Internally-generated reference output voltage for external series-pass transistor.	—
1	VCR	Regulated voltage input from series-pass transistor (2N5320 or equivalent).	—
12	GND	Circuit ground.	—
37	V_{CC}	Input connection for +5V power.	—

Figure 2. Typical 8X300 System with Pin Definitions

TYPICAL 8X300 SYSTEM HOOKUP

Although the system hookup shown in Figure 2 is of the simplest form, it provides a fundamental look at the 8X300 microcontroller and peripheral relationships. As indicated, program storage can be either ROM or PROM and, by using various addressing-methods/decoding-schemes, memory paging techniques can be easily implemented. Also, by proper bit assignment, some external interface logic and, under software control, the program memory can be used as a storage device for interrupt-service subroutines. The user interface ($\overline{IV0}$ through $\overline{IV7}$) is capable of addressing 256 Input/Output ports and, with the additional bank-select bit ($\overline{LB}$ and $\overline{RB}$), the number of addressable I/O ports is 512—the left bank and right bank each consisting of 256 ports. The I/O ports of each bank can be used in a variety of ways; one of these ways is shown in Figure 2. When $\overline{LB}$ is active low, the left bank can be enabled and, providing there is an address match, anyone of 128 I/O ports or anyone of 128 locations within the RAM memory can be accessed for input/output operations. When $\overline{RB}$ is active low, the same set of conditions are applicable to the right bank. With some sacrifice in speed, any given I/O port can be interfaced to a memory peripheral or other I/O device of the user.

PROGRAM STORAGE INTERFACE

As shown in Figure 2, program storage is connected to output address lines A0 through A12 (A12 = LSB) and input instruction lines I0 through I15. An address output on A0/A12 identifies one 16-bit instruction word in program storage. The instruction word is subsequently input on I0/I15 and defines the microcontroller operations which are to follow.

The Signetics 82S115 PROM or any TTL-compatible memory can be used for program storage. (Note. The worst-case access time depends upon the instruction cycle time, and also, the overall system configuration.)

I/O INTERFACE AND CONTROL

An 8-bit I/O data bus is used by the microcontroller to communicate with two fields of I/O devices. The complementary $\overline{LB}$ and $\overline{RB}$ signals identify which field of the I/O devices is enabled.

Both data and address information are output on the I/O bus. The SC (Select Command) and WC (Write Command) signals distinguish between data and address information as follows:

SC	WC	FUNCTION
High	Low	I/O address is being output on the I/O (IV) bus
Low	High	I/O data is being output on the I/O (IV) bus
Low	Low	Input data expected from selected I/O device
High	High	Invalid (not generated by 8X300)

DATA PROCESSING

From a data processing point of view, the 8X300 microcontroller chip (Figure 1) contains eight 8-bit working registers (R1 through R6, R11, AUXiliary), an arithmetic logic unit (ALU), an overflow register (OVF), rotate/shift/mask/merge logic, and a bidirectional 8-bit I/O bus. Internal 8-bit data paths connect the registers and I/O bus to the ALU inputs, and the ALU output to the registers and I/O bus. Inputs to the ALU are preceded by the data-rotate and data-mask logic and the ALU output is followed by the shift and merge logic. Any one or all of the logic functions can operate on 8-bits of data in a single instruction cycle. Data from the source register can be right-rotated (end around) before processing by the ALU; external data (I/O bus) can also be masked to isolate a portion of the 8-bit field. Since the ALU always processes 8-bits of data, bit positions not specified by the mask operation are filled with zeroes.

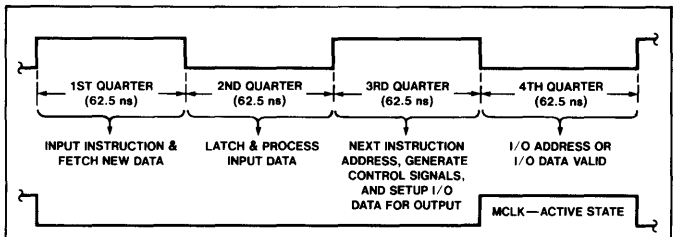
When less than 8-bits of data are specified as output to the I/O bus from the ALU, the data field (shifted and masked, as required) is merged with prior contents of the I/O latches to form the output data. Bit positions of the I/O data not affected by the logic operations are not modified. Depending upon whether an I/O peripheral or an internal register is specified in the instruction as the source of data, the I/O latches contain, respectively, I/O-bus source data or destination data. For instance, when an internal register is specified as a source of data and an I/O peripheral as the destination, data from the peripheral is read into the I/O latches at the start of the instruction cycle; processed data is then merged with contents of the I/O latches to form the I/O output data at the end of the instruction cycle. When an I/O peripheral is specified as both data source and destination, data from the source is used both as the input to the I/O latches and as data to be processed; the processed data is then merged with data from the I/O latches to form the previously-described I/O bus output. If the data source and destination are on opposite banks of the 8X300 bus, the destination data is written with a full 8-bits, since the prior contents were not stored in the I/O latches.

INSTRUCTION CYCLE

Each microcontroller operation is executed in a single instruction cycle. The instruction cycle is divided into quarters with each quarter cycle being as short as 62.5-nanoseconds. Figure 3 shows the general functions that occur during each quarter cycle; specifics regarding minimum/maximum timing and other critical values are described under "Design Parameters" in this data sheet. During the first quarter cycle, a new instruction from program storage is input on signal lines I0 through I15; simultaneously, new data is fetched via the input/output bus ($\overline{IV0}$ through $\overline{IV7}$). At the end of the first quarter cycle, the new instruction is latched in the instruction register and the new I/O data is present at the input of the chip but is not, as yet, latched by the IV latches.

In the second quarter cycle, the I/O data stabilizes and preliminary processing is completed; at the end of this quarter, the IV latches are closed and final processing can be accomplished. During the third quarter cycle, the address for the next instruction is output to the I/O (IV) bus, control signals are generated, and I/O data is setup for the output

phase. During the fourth quarter cycle, a master clock signal (MCLK) generated by the 8X300 is used to latch valid address or data into peripheral devices connected to the IV bus; MCLK is also used to synchronize any external logic with timing circuits of the 8X300. To summarize the action, the first half of the instruction cycle deals primarily with input functions and the second half is mostly concerned with output functions.



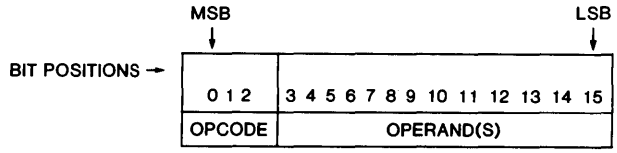
NOTES
1. New instruction must be accepted and latched at end of first quarter cycle.
2. The I/O data latches are open for the first two quarter cycles, that is, for 125-nanoseconds.
3. The address is output during third quarter cycle.
4. I/O bus drivers are active (turned on) during third and fourth quarter cycles.

Figure 3. Instruction Cycle and MCLK with: Crystal = 8MHz and Cycle Time = 250 ns

INSTRUCTION SET

General Format and Basic Operations

The 16-bit instruction word (I0 through 15) from program storage is input to the instruction register (Figure 1) and is subsequently decoded to implement the events to occur during the current instruction cycle. The instruction word is formatted as follows:



Rather than discrete instructions, the three operation code (OP CODE) bits specify eight instruction classes. Each instruction class is subject to a number of powerful variations; these variations are specified by the thirteen operand bits. General areas of control for the eight instruction classes are:

- Arithmetic and Logic Operations (ADD, AND, AND XOR)
- Movement of Data and Constants (MOVE and XMIT)
- Branch or Test (JMP, NZT, and XEC)

Basic operations for each of the eight instruction classes are as follows; a summary of the instruction set is provided in Table 1.

MOVE—data in source register or I/O-bus input is moved to destination register or I/O-bus output. Data can be shifted any number of places and/or masked to any length.

ADD—data in source register or I/O-but input is added to content of AUX (R0) register and the result is placed in the destination register or I/O-bus output. Data can be shifted and/or masked, as required.

Table 1. Summary of 8X300 Instruction Set

INSTRUC CLASS	OPCODE	FORMATS	DESCRIPTION	I/O CONT SIG	STATE OF CONTROL SIGNAL DURING INSTRUCTION CYCLE: INPUT PHASE (INSTRUCTION INPUT & DATA PROCESSING)	STATE OF CONTROL SIGNAL DURING INSTRUCTION CYCLE: OUTPUT PHASE (ADDRESS & I/O BUS)
MOVE	0	**F1: Register to Register** 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | R | D Invalid values of "S": 07_8, 17_8, 20_8-37_8 Invalid values of "D": 10_8, 20_8-37_8	(S) → D Move content of internal register specified by S-field to internal register specified by D-field. Prior to the "MOVE" operation, right-rotate contents of internal source register by octal value (0 through 7) defined by the R-field.	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L X X	H if "D" = 07_8, 17_8 L H if "D" = 17_8 L if "D" = 07_8
		F2: I/O Bus to Register 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | L | D Valid values of "S": 20_8-37_8 Invalid values of "D": 10_8, 20_8-37_8	Move right-rotated I/O bus (source) data specified by the S-field to internal register specified by the D-field. The L-field specifies the length of source data starting from the LSB-position and, if less than 8-bits, the remaining bits are filled with zeroes.	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L L if "S" = 20_8-27_8 H if "S" = 30_8-37_8	H if "D" = 07_8, 17_8 L H if "D" = 17_8 L if "D" = 07_8
		F2: Register to I/O Bus 0 1 2 3 | 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | L | D Invalid values of "S": 07_8, 17_8, 20_8, 37_8 Valid values of "D": 20_8-37_8	Move contents of internal register specified by the S-field to the I/O latches. Before outputting on I/O bus, data is shifted as specified by the least significant octal digit of the D-field and the bits specified by the L-field are merged with the latched I/O data.	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L L if "D" = 20_8-27_8 H if "D" = 30_8-37_8	L H L if "D" = 20_8-27_8 H if "D" = 30_8-37_8
		F2: I/O Bus to I/O Bus 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | L | D Valid values of "S": 20_8-37_8 Valid values of "D": 20_8-37_8	Move right rotated I/O-bus (source) data specified by the S-field to the I/O latches. Before outputting on I/O bus, shift data as specified by the D-field; then merge source and latched I/O data as specified by the L (length) field.	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L L if "D" = 20_8-27_8 H if "D" =30_8-37_8	L H L if "D" = 20_8-27_8 H if "D" = 30_8-37_8

Table 1. Summary of 8X300 Instruction Set (Continued)

INSTRUC CLASS	OPCODE	FORMATS	DESCRIPTION	I/O CONT SIG	STATE OF CONTROL SIGNAL DURING INSTRUCTION CYCLE	
					INPUT PHASE (INSTRUCTION INPUT & DATA PROCESSING)	OUTPUT PHASE (ADDRESS & I/O BUS)
ADD	1	Same as MOVE instruction class	(S) plus (AUX) → D Same as MOVE instruction class except that contents of AUX (R0) register are ADDed to the source data. If there is a "carry" from MSB, then OVF (overflow) = 1, otherwise OVF = 0.	Same as MOVE instruction class		
AND	2	Same as MOVE instruction class	(S) ∧ (AUX) → D Same as MOVE instruction class except that contents of AUX (R0) register are ANDed with source data.	Same as MOVE instruction class.		
XOR	3	Same as MOVE instruction class	(S) ⊕ (AUX) → D Same as MOVE instruction class except that contents of AUX (R0) register are exclusively ORed with source data.	Same as MOVE instruction class.		
XEC	4	**F3: Register Immediate** 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | S | J Invalid values of "S": 07_8, 17_8, 20_8-37_8 Valid values of "J": 000_8-377_8	**Execute instruction at current page address offset by J (literal) + (S). Return to normal instruction flow unless a branch is encountered.** Execute instruction at an address determined by replacing the low-order 8-bits of the Program Counter with the following derived sum: • Value of literal (J-field) plus • Contents of internal register specified by S-field The PC is not incremented and the overflow status (OVF) is not changed.	SC = WC = $\overline{LB}$ =	L L X	L L X
		F4: I/O Bus Immediate 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | L | J Valid values of "S": 20_8-37_8 Valid values of "J": 00_8-37_8	Execute instruction at an address determined by replacing the low-order 5-bits of Program Counter with the following derived sum: • 5-bit value of literal (J-field) plus • Value of rotated source data specified by S-field (The L-field specifies the length of source data starting from the LSB-position and, if less than 8-bits, the remaining bits are filled with zeros; the Program Counter is not incremented and the overflow status (OVF) is not changed.)	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L L if "S" = 20_8-27_8 H if "S" = 30_8-37_8	L L X X
NZT	5	**F3: Register Immediate** 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | S | J Invalid values of "S": 07_8, 17_8, 20_8-37_8 Valid values of "J": 000_8-377_8	**If data specified by the S-field is not equal to zero, jump to current page address offset by value of J-field; otherwise, increment the Program Counter.** If contents of internal register specified by S-field is **non-zero**, transfer to address determined by replacing the low-order 8-bits of Program Counter with "J", otherwise, increment PC.	SC = WC = $\overline{LB}$ =	L L X	L L X
		F4: I/O Bus Immediate 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | L | J Valid values of "S": 20_8-37_8 Valid values of "J": 00_8-37_8	If right-rotated I/O bus data is **non-zero**, transfer to address determined by replacing low-order 5-bits of Program Counter with "J", otherwise, increment PC. (The L-field specifies the length of source I/O data starting from the LSB-position and, if less than 8-bits, the remaining bits are filled with zeroes.)	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L L if "S" = 20_8-27_8 H if "S" = 30_8-37_8	L L X X
XMIT	6	**F3: Register Immediate** 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | D | J Invalid values of "D": 10_8, 20_8-37_8 Valid values of "J": 000_8-377_8	Transmit J → D Transmit and store 8-bit binary pattern in J-field to internal register specified by D-field.	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L X X	H if D = 07_8 or 17_8 L H if D = 17_8 L if D = 07_8
		F4: I/O Bus Immediate 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | D | L | J Valid values of "D": 20_8-37_8 Valid values of "J": 00_8-37_8	Transmit binary pattern in J-field to I/O bus. Before putting data on I/O bus, shift literal value "J" as specified by the D-field and merge bits specified by the L-field with existing I/O bus data. If the L-field specifies more than 5-bits starting from the LSB-position, all remaining bits are set to zero.	SC = WC = $\overline{LB}$ = $\overline{LB}$ =	L L L if D = 20_8-27_8 H if D = 30_8-37_8	L H L if D = 20_8-27_8 H is D = 30_8-37_8
JMP	7	**F5: Address Immediate** 0 1 2 | 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | A Valid values of A: 00000_8-17777_8	Jump to address in program storage specified by A-field; this address is loaded into the Address Register and the Program Counter.	SC = WC = $\overline{LB}$ =	L L X	L L X

NOTES

* $\overline{RB}$ is complement of $\overline{LB}$, X = Undefined

AND—data in source register or I/O-bus input is ANDed with content of AUX (R0) register and the result is placed in the destination register or I/O-bus output. Data can be shifted and/or masked, as required.

XOR—data in source register or I/O-bus input is exclusively ORed with contents of AUX (R0) register and the result is placed in the destination register or I/O-bus output. Data can be shifted and/or masked, as required.

XMIT—immediate data field of instruction word replaces data in destination register or I/O-bus output.

XEC—executes instruction at the program address which is formed by replacing the least significant bits of the last address with the sum of:

- Literal (J) field value of instruction plus,
- Value of data in source register or I/O-bus input.

NZT—least significant bits of program address are replaced by literal (J) field of instruction if the source register or I/O-bus is not equal to zero.

JMP—program address is replaced by address field of the instruction word.

Instruction Fields

As shown in Table 1, each instruction contains an operations code (OPCODE) field and from one-to-three operand fields. The operand fields are: Source (S), Destination (D), Rotate/Length (R/L), Literal (J), and Address (A). The OPCODE and operand fields are briefly described in the following paragraphs.

Operations Code Field: The three-bit OPCODE field specifies one of eight classes of 8X300 instructions; octal designations for this field and operands for each instruction class are shown in Table 1.

Source (S) and Destination (D) Fields: The five-bit (S) and (D) fields specify the source and destination of data for the operation defined by the OPCODE field. The AUXiliary (R0) register is an implied second operand for the ADD, AND, and XOR instructions, each of which require two source fields. That is, instructions of the form:

ADD X, Y

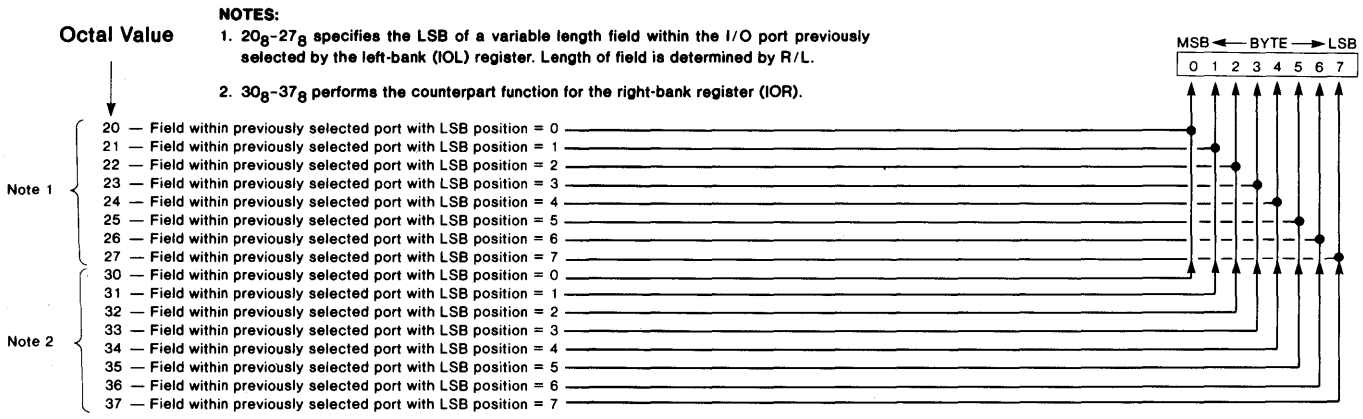
imply a third operand, say Z, located in the AUX (R0) register. Thus, the operation for the preceding expression is actually (X + Z), with the result stored in Y. The (S) and/or (D) fields can specify an internal 8X300 register or any one-to-eight bit I/O field; octal values for these registers and Source/Destination field assignments are provided in Table 2.

Table 2. Octal Addresses of 8X300 Registers and Address/Bit Assignments of Source/Destination Fields

Octal Value	8X300 Register	Octal Value	8X300 Register
00	Auxiliary (R0)	10	OVF (Overflow Register)—used only as a source
01	R1	11	R11
02	R2	12	Unassigned
03	R3	13	Unassigned
04	R4	14	Unassigned
05	R5	15	Unassigned
06	R6	16	Unassigned
07	*IOL Register—Left Bank I/O Address Register; Used only as destination	17	*IOR Register—Right Bank I/O Address Register; Used only as destination

NOTE

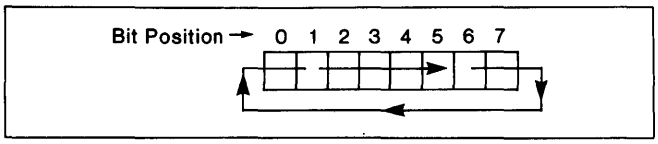
*If IOL or IOR is specified as a source of data, the source data is all zeroes.



Rotate (R) and Length (L) Field: The three-bit R/L field performs one of two functions, specifying either the field length (L) or a right-rotate (R). For a given instruction, the specified function depends upon the contents of the source (S) and destination (D) fields.

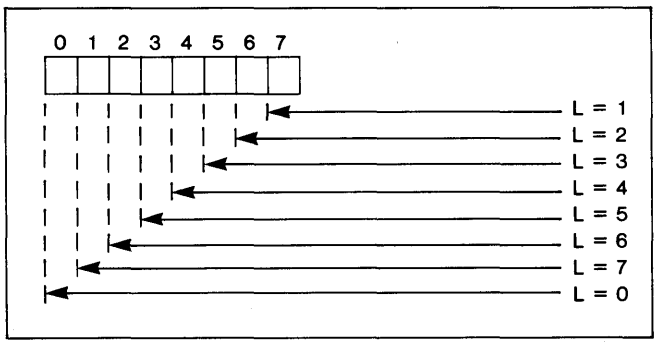
- When an internal register is specified by both the source and destination fields, the (R) field is invoked and it specifies a right-rotate of the data specified in the (S) field—see accompanying diagram. The source-register data (up to eight-bits) is right-rotated within one instruction cycle. (The right-rotate function is implemented on the bus and not in the source register.)

RIGHT-ROTATE FUNCTION



- When either or both of the source and destination fields specify a variable-length I/O data field, the (L) field specifies the length of the I/O data field—see accompanying diagram. If the source field specifies an I/O address (20_8-37_8) and the destination field specifies an internal register (00_8-06_8, 07_8, 11_8, or 17_8), the L-field specifies the length of source data; the source data is formed by right-rotating the I/O bus data according to the source address (Table 2) and then masking result as specified by L-field. If length is less than eight-bits, all remaining bits are set to zero prior to processing data in the ALU. If the source field specifies an internal register (00_8,-06_8, 10_8,or 11_8) and the destination field specifies I/O bus data (20_8-37_8), the L field specifies the length of the destination data. To form the destination data, the ALU output is left-shifted according to the destination address (Table 2) and then masked to the required length—see DATA LENGTH SPECIFICATION. The destination data is merged with data in the I/O latches to finalize the I/O bus data. Hence, a one-to-eight bit destination data field can be inserted into the existing eight-bit I/O port without modifying surrounding bits. If both the source and destination fields specify I/O bus data (20_8-37_8), the L-field specifies the length of both the source and destination data.

DATA LENGTH SPECIFICATION



To form the source data, the I/O bus data is right-rotated according to the source address (Table 2) and then masked to the required length—see preceding DATA LENGTH SPECIFICATION. If length is less than eight-bits, all remaining bits are set to zero before processing in the ALU. To form the destination data, the ALU output is left-shifted according to the destination address (Table 2) and masked to the required length specification. The destination data is then merged into the I/O bus data that was used to obtain the source; thus, if the source and destination addresses are on the same bank, the I/O bus data written to the destination register appears unmodified, except for bits changed during the shift-and-mask operations. If the source and destination addresses refer to different banks, the destination register is changed to contain the contents of the source register in those bit positions not affected by the destination data.

J-Field: The 5-bit or 8-bit (J) field is used to load a literal value (contained in the instruction) into a register, into a variable I/O data field, or to modify the low-order bits of the Program Counter. The bit-length of the (J) field is implied by the (S) field in the XEC, NZT, and XMIT instructions, based on the following considerations.

- When the source (S) field specifies an internal register, the literal value of the J-field is an 8-bit binary number.
- When the source (S) field specifies a variable I/O data field, the literal value of the J-field is a 5-bit binary number.

A-Field: The 13-bit (A) field is an address field which allows the 8X300 to directly address up 8192 locations in Program Storage memory.

INSTRUCTION SEQUENCE CONTROL

Formation of Instruction Address

The Address Register and Program Counter are used to generate addresses for accessing an instruction from program storage. The instruction address is formed in any one of four ways:

- For all except the JMP, XEC, and a "satisfied" NZT instruction, the Program Counter is incremented by one and placed in the Address Register.
- For the JMP instruction, the 13-bit A-field contained in the JMP instruction word replaces the contents of both the Address Register and Program Counter.
- For the XEC instruction, the Address Register is loaded with the high-order bits of the Program Counter modified as follows:
  XEC using I/O Bus Data: low order 5-bits of ALU output replaces counterpart bits in Address Register.
  XEC using Data from Internal Register: low order 8-bits of ALU output replaces counterpart bits in Address Register.
  The Program Counter is not modified for either of the above conditions.
- For a "satisfied" NZT instruction, the low order 5-bits (NZT source is I/O Bus Data) or low order 8/bits (NZT source is an Internal Register) of both the Address Register and Program Counter are loaded with the literal value specified by J-field of the instruction word.

Data Addressing

The source and/or destination addresses of the data to be operated upon are specified as part of the instruction word. As shown in Table 3, source/destination addresses are specified using a five-bit address (00_8 through 37_8). When the most significant octal digit is a 0 or 1, the source and/or destination address is an internal register; if the most significant digit is a 2 or 3, an I/O bus address is indicated—2 specifying a left-bank (LB) address and 3 specifying a right-bank (RB) address. The least significant octal digit (0 through 7) indicates either a specific internal register address or positioning information for the least significant bit when specifying I/O bus data. Referring to Table 1, the AUXiliary register (00) is the implied source of the second argument for the ADD, AND, and XOR operations. IOL (destination address 07_8) and IVR (destination address 17_8) provide a means of routing address information to I/O registers. With IOL or IOR specified as the destination address, the data is placed on the I/O bus during the output phase of the instruction cycle. Simultaneously, a select command (SC) is generated to inform all I/O devices that information on the I/O bus is to be considered as an I/O address. Since IOL and IOR are not harware registers, they should never be specified as a source address.

Control outputs $\overline{LB}$ and $\overline{RB}$ are used to partition I/O bus devices into two fields of 256 addresses. With LB in the active-low state and a source address of 20_8-27_8, the left bank of I/O devices are enabled during the input phase of the instruction cycle. With $\overline{RB}$ in the active-low state and a source address of 30_8-37_8, the right bank of devices are enabled. During the output phase, $\overline{RB}$ is low if the destination address is IOR (17_8) or 30_8-37_8; $\overline{LB}$ is low if the destination address is IOL (07_8) or 20_8-27_8. Each address field ($\overline{LB}$ and $\overline{RB}$) can have a different I/O device selected; thus, two devices can be directly accessed within one instruction cycle.

Table 3. Source/Destination Addresses

Source and/or Destination Field (Octal)	Source/Destination
00	AUXiliary register (R0)
01-06	Working registers R1-R6, respectively
07	IOL Left-bank enable (Destination only)
10	Overflow status—OVF (Source only)
11	Working register R11
17	IOR Right-bank enable (Destination only)
2N (N = 0, 1, 2, 3 4, 5, 6, or 7)	If a source, I/O data is right-rotated (7 − N) bits and then masked as specified by the L-field. $\overline{LB}$ = low and $\overline{RB}$ = high generated during input phase.
	If a destination, I/O data is left-shift (7 − N) bits and merged (specified by L-field) with data contained in the I/O latches. $\overline{LB}$ = low and $\overline{RB}$ = high generated during output phase.
3N (N = 0, 1, 2, 3, 4, 5, 6, or 7)	If a source, I/O data is right-rotated (7 − N) bits and then masked as specified by the L-field. $\overline{LB}$ = high and $\overline{RB}$ = low generated during input phase.
	If a destination, I/O data is left-shifted (7 − N) bits and merged (specified by L-field) with data contained in the I/O latches. $\overline{LB}$ = high and $\overline{RB}$ = low generated during output phase.

DESIGN PARAMETERS

Hardware design of an 8X300-based system largely consists of the following operations:

- **Selecting and interfacing a Program Storage device—ROM, PROM, etc. (Pins 2 through 9 and 45 through 49 for 13-bit address interface; Pins 13 through 28 for 16-bit instruction interface.)**
- **Selecting and interfacing Input/Output devices—RAM, Multiplexers, I/O Ports, and other eight-bit addressable I/O devices. (Pins 33 through 36 and pins 38 through 41 for eight-bit I/O interface.)**
- **Choosing and implementing System Clock—Capacitor-Controlled, Crystal-Controlled, or Externally-Driven. (Pins 10 and 11 for System Clock interface.)**
- **Selection of 5-volt power supply and off-chip series-pass transistor.**
- **External logic, as required, to meet the control requirements of a particular application.**

All information required for easy implementation of these design requirements is provided under the following captions.

- **DC Characteristics**
- **AC Characteristics**
- **Timing Considerations**
- **Clock Considerations**
- **HALT/RESET Logic**
- **Voltage Regulator**

DC CHARACTERISTICS (Commercial Part) $4.75V \leq V_{CC} \leq 5.25V$, $0°C \leq T_A \leq 70°C$

PARAMETER		TEST CONDITIONS	LIMITS			UNIT	COMMENTS
			Min	Typ	Max		
V_{CC}	Supply voltage		475	5.0	5.25	V	5V ± 5%; pin 37 only
V_{IH}	High level input voltage		0.6 2.0		2.0	V	X1 and X2 All other pins
V_{IL}	Low level input voltage				0.4 0.8	V	X1 and X2 All other pins
V_{OH}	High level output voltage	V_{CC} = min; I_{OH} = −3mA	2.4	3.0		V	
V_{OL}	Low level output voltage	V_{CC} = min; I_{OL} = 6mA V_{CC} = min; I_{OL} = 16mA		0.39 0.39	0.55 0.55	V	A0 through A12 All other outputs
V_{CR}	Regulator voltage	V_{CC} = 5V		3.1		V	From series-pass transistor
V_{IC}	Input clamp voltage	V_{CC} = min; I_{IN} = −10mA			−1.5	V	Crystal inputs X1 and X2 do not have internal clamp diodes.
I_{IH}	High-level input current	V_{CC} = max; V_{IH} = 0.6V V_{IH} = 4.5V	1.0	1.7 1	3.0 50	mA μA	X1 and X2 All other pins
I_{IL}	Low-level input current	V_{CC} = max; V_{IL} = 0.4V		 −0.13 −0.67 −0.23	−3 −0.2 −1.6 −0.4	mA	X1 and X2 $\overline{IV0}$-$\overline{IV7}$ I0-I15 $\overline{HALT}$ and $\overline{RESET}$
I_{OS}	Short circuit output current	V_{CC} = max; V_{CR} = V_{CRH} (Note: At any time, no more than one output should be connected to ground.)	−30		−140	mA	All output pins
I_{CC}	Supply current	V_{CC} = max; V_{CR} = V_{CRH}			150	mA	
I_{REG}	Regulator control	V_{CC} = 5.0V	−14		−21	mA	
I_{CR}	Regulator current	V_{CC} = max			250	mA	

NOTES:
1. Operating temperature ranges are guaranteed after thermal equilibrium has been reached.
2. All voltages measured with respect to ground terminal.

AC CHARACTERISTICS (Commercial Part)

CONDITIONS: V_{CC} = 5V (±5%), V_{IN} = 0V or 3V, $0°C \leq T_A \leq 70°C$
LOADING: (See test circuits)

PARAMETER (NOTE 1)		LIMITS (INSTRUCTION CYCLE TIME = 250 ns)			LIMITS (INSTRUCTION CYCLE TIME > 250 ns)			UNITS	COMMENTS
		Min	Typ	Max	Min	Typ	Max		
T_{PC}	Processor cycle time	250			250			ns	
T_{CP}	X1 clock period	125			125			ns	
T_{CH}	X1 clock high time	62			62			ns	
T_{CL}	X1 clock low time	62			62			ns	
T_{MCH}	MCLK high delay	31	42	52	31	42	52	ns	
T_{MCL}	MCLK low delay	31	42	52	31	42	52	ns	
T_W	MCLK pulse width	55	62	69	T_{4Q}−7	T_{4Q}		ns	Note 2
T_{AS}	X1 falling edge to address stable	50	63	80	50	63	80	ns	Note 7

AC CHARACTERISTICS (Commercial Part) (Continued)

CONDITIONS: V_{CC} = 5V (±5%), V_{IN} = 0V or 3V, 0°C ≤ T_A ≤ 70°C
LOADING: (See test circuits)

PARAMETER (NOTE 1)		LIMITS (INSTRUCTION CYCLE TIME = 250 ns)			LIMITS (INSTRUCTION CYCLE TIME > 250 ns)			UNITS	COMMENTS
		Min	Typ	Max	Min	Typ	Max		
T_{MAS}	MCLK falling edge to address stable	130	143	160	$T_{1Q}+T_{2Q}$ +5	$T_{1Q}+T_{2Q}$ +18	$T_{1Q}+T_{2Q}$ +35	ns	Notes 2, 3, & 7
T_{IA}	Instruction to address			170			T_{2Q}+108	ns	Notes 2, 3, & 8
T_{IVA}	Input data to address			105			105	ns	Notes 3 & 9
T_{IS}	Instruction set-up time (X1 rising edge)	−7			−7			ns	Note 10
T_{MIS}	MCLK falling edge to instruction stable			20			T_{1Q}−42	ns	Notes 2, 4, & 10
T_{IH}	Instruction hold time (X1 rising edge)	45			45			ns	Note 11
T_{MIH}	Instruction hold time) (MCLK falling edge)	60			T_{1Q}−2			ns	Notes 2 & 11
T_{WH}	X1 falling edge to SC/WC rising edge	40	49	58	40	49	58	ns	
T_{MWH}	MCLK falling edge to SC/WC rising edge	125	130	135	$T_{1Q}+T_{2Q}$	$T_{1Q}+T_{2Q}$ +5	$T_{1Q}+T_{2Q}$ +10	ns ns	Note 2
T_{WL}	X1 falling edge to SC/WC falling edge	40	49	58	40	49	58	ns	
T_{MWL}	MCLK falling edge to SC/WC falling edge	5	7	15	5	7	15	ns	
T_{IBS}	X1 falling edge to $\overline{LB}/\overline{RB}$ (Input phase)	48	60	70	48	60	70	ns	
T_{MIBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Input phase)	7	17	25	7	17	25	ns	
T_{IIBS}	Instruction to $\overline{LB}/\overline{RB}$ (Input phase)		27	35		27	35	ns	
T_{OBS}	X1 falling edge to $\overline{LB}/\overline{RB}$ (Output phase)	48	60	70	48	60	70	ns	
T_{MOBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Output phase)	132	137	147	$T_{1Q}+T_{2Q}$ +7	$T_{1Q}+T_{2Q}$ +12	$T_{1Q}+T_{2Q}$ +22	ns	Note 2
T_{IDS}	Input data set-up time (X1 falling edge)	25	16		25	16		ns	
T_{MIDS}	MCLK falling edge to input data stable		65	55		$T_{1Q}+T_{2Q}$ −60	$T_{1Q}+T_{2Q}$ −70	ns	Notes 2 & 5
T_{IDH}	Input data hold time (X1 falling edge)	40	30		40	30		ns	
T_{MDIH}	Input data hold time (MCLK falling edge)	125	112		$T_{1Q}+T_{2Q}$	$T_{1Q}+T_{2Q}$ −13		ns	Note 2
T_{ODH}	Output data hold time (X1 falling edge)	55	65	75	55	65	75	ns	
T_{MODH}	Output data hold time (MCLK falling edge)	11	20	25	11	20	25	ns	
T_{ODS}	Output data stable (X1 falling edge)	74	84	94	74	84	94	ns	Notes 12, 14, & 15
T_{MODS}	Output data stable (MCLK falling edge)	150	160	170	$T_{1Q}+T_{2Q}$ +25	$T_{1Q}+T_{2Q}$ +35	$T_{1Q}+T_{2Q}$ +45	ns	Notes 2, 12, 14, & 15

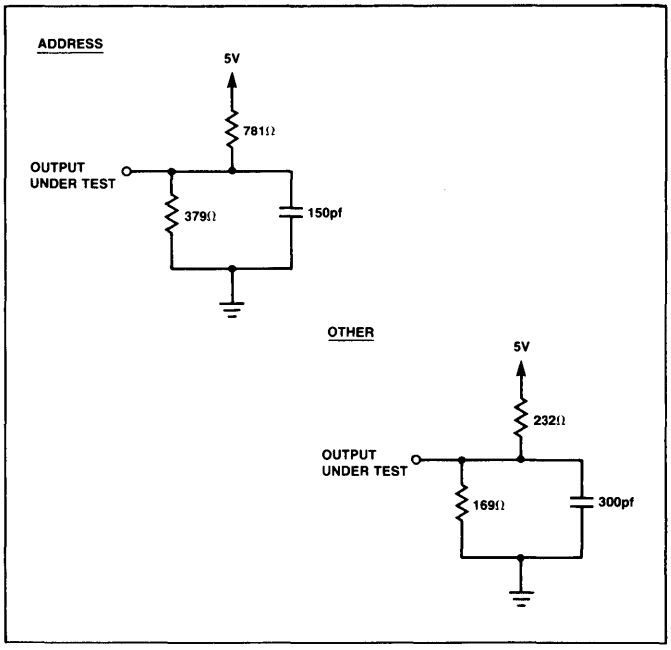
AC CHARACTERISTICS (Commercial Part) (Continued)

CONDITIONS: V_{CC} = 5V (±5%), V_{IN} = 0V or 3V, 0°C ≤ T_A ≤ 70°C
LOADING: (See test circuits)

PARAMETER (NOTE 1)		LIMITS (INSTRUCTION CYCLE TIME = 250 ns)			LIMITS (INSTRUCTION CYCLE TIME > 250 ns)			UNITS	COMMENTS
		Min	Typ	Max	Min	Typ	Max		
T_{DD}	Input data to output data	104	120	136	104	120	136	ns	Notes 13 & 15
T_{HS}	$\overline{HALT}$ set-up time (X1 rising edge)	0			0			ns	
T_{MHS}	MCLK falling edge to $\overline{HALT}$ falling edge			18			$T_{1Q}-44$	ns	Notes 2 & 6
T_{HH}	$\overline{HALT}$ hold time (X1 rising edge)	32			32			ns	
T_{MHH}	$\overline{HALT}$ hold time (MCLK falling edge)	50			$T_{1Q}-12$			ns	Note 2
T_{ACC}	Program storage access time			80				ns	
T_{IO}	I/O port output enable time ($\overline{LB}/\overline{RB}$ to valid IV data input)			30				ns	

NOTES:
1. X1 and X2 inputs are driven by an external pulse generator with an amplitude of 1.5 volts; all timing parameters are measured at this voltage level.
2. Respectively, T_{1Q}, T_{2Q}, T_{3Q}, and T_{4Q} represent time intervals for the first, second, third, and fourth quarter cycles.
3. Capacitive loading for the address bus is 150 picofarads.
4. Same as TIS but referenced to falling edge of MCLK.
5. Same as TIDS but referenced to falling edge of MCLK.
6. Same as THS but referenced to falling edge of MCLK.
7. TAS is obtained by forcing a valid instruction and an I/O bus input to occur earlier than the specified minimum set-up time; the TAS parameter then represents the earliest time that the address bus is valid.
8. TIA is obtained by forcing a valid instruction input to occur earlier than the minimum set-up time.
9. TIVA is obtained by forcing a valid I/O bus input to just meet the minimum set-up time.
10. TMIS represents the set-up time required by internal latches of the 8X300. In system applications, the instruction input may have to be valid before the worst-case set-up time in order for the system to respond with a valid I/O bus input that meets the I/O bus input set-up time (TIDS and TMIDS).
11. TIH represents the hold time required by internal latches of the 8X300. To generate proper LB/RB signals, the instruction must be held valid until the address bus changes.
12. TODS is obtained by forcing a valid I/O bus input to occur earlier than the I/O bus input set-up time (TIDS); this timing parameter represents the earliest time that the I/O output data can be valid.
13. TDD is obtained by forcing a valid I/O bus input to just meet the minimum I/O bus input set-up time; thus timing parameter represents the latest time that the I/O output data can be valid.
14. The minimum figure for these parameters represents the earliest time that I/O bus output drivers of the 8X300 will turn on.
15. For TIDS ≥ 35 ns, TODS or TMODS should be used to determine when the output data is stable.

TEST CIRCUITS



DC CHARACTERISTICS (Military Part)

S8X300-1 $-40°C \leq TC \leq 100°C$ $V_{CC} = 5V \pm 5\%$
S8X300-2 $-20°C \leq TC \leq 100°C$ $V_{CC} = 5V \pm 10\%$

PARAMETER		TEST CONDITIONS	LIMITS			UNIT
			Min	Typ	Max	
V_{IH}	High level input voltage X1, X2 All others		 0.6 2.0			 V V
V_{IL}	Low level input voltage X1, X2 All others				 0.4 0.8	 V V
V_{IC}	Input clamp voltage (Notes 1 & 5)	V_{CC} = min $I_I = -10mA$			-1.5	V
I_{IH}	High level input current X1, X2 All others	 V_{CC} = max V_{IH} = 0.6V V_{CC} = max V_{IH} = 4.5V			 3.0 0.05	 mA
I_{IL}	Low level input current X1, X2	 V_{CC} = max V_{IL} = 0.4V			 -3.0	 mA
	$\overline{IV0}$-$\overline{IV7}$	V_{CC} = max V_{IL} = 0.4V			-0.3	mA
	I0-I15	V_{CC} = max V_{IL} = 0.4V			-1.6	mA
	$\overline{HALT}$, $\overline{RESET}$	V_{CC} = max V_{IL} = 0.4V			-0.4	mA
V_{OL}	Low level output voltage A0-A12 All others	 V_{CC} = min $I_L = 4.25mA$ V_{CC} = min $I_{OL} = 16mA$			 0.55 0.55	 V V
V_{OH}	High level output voltage	V_{CC} = min $I_{OH} = -3mA$	2.4			V
I_{OS}	Short circuit output current (Note 2)	V_{CC} = max	-30		-140	mA
I_{CC}	Supply current (Note 4)	V_{CC} = max			160	mA
I_{REG}	Regulator control	V_{CC} = 5.0V	-14		-21	mA
I_{CR}	Regulator current	V_{CC} = max			285	mA
I_{CR}	Regulator current	$TC \geq 25°C$ V_{CC} = max			330	mA
V_{CR}	Regulator voltage	$TC < 25°C$ (Note 3)		3.1		V

NOTES:
1. Crystal inputs X1 and X2 do not have clamp diodes.
2. Only one output may be grounded at a time.
3. From series-passed transistor under the following conditions:
   VCC = Max, HALT = RESET = ADDRESS = IVX = 0.0V, all other pins open.
4. Pin 37 only.
5. Test each input one at a time.
6. All voltages are with respect to ground terminal.
7. The operating temperature ranges are guaranteed after thermal equilibrium has been reached.
8. Storage temperature $-65°C$ to $+150°C$.

AC CHARACTERISTICS (Military Part)

CONDITIONS: S8X300-1—V_{CC} = 5V (±5%) -40°C ≤ T_C ≤ 100°C
S8X300-2—V_{CC} = 5V (±10%) -20°C ≤ T_C ≤ 100°C

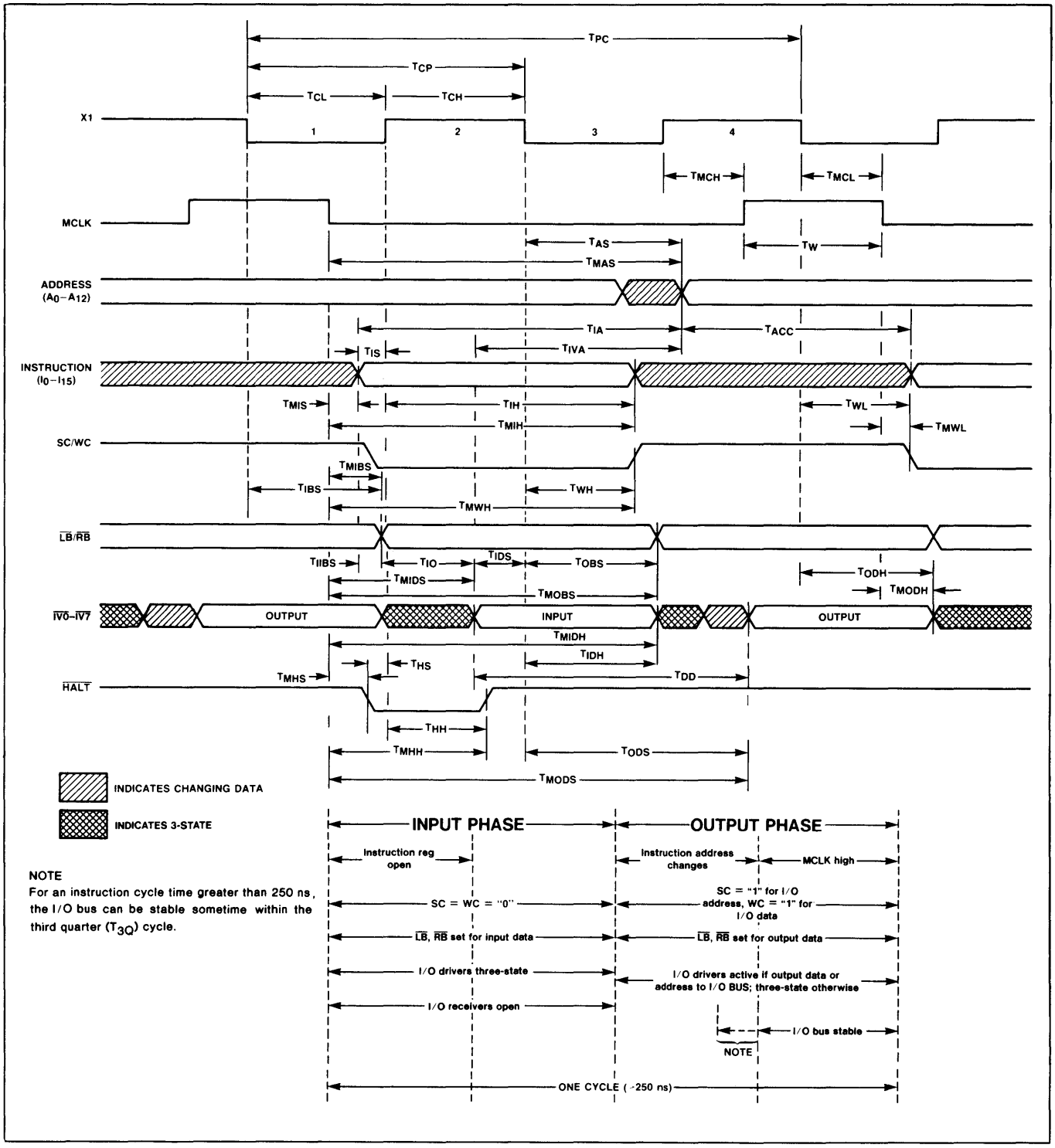
PARAMETER		TEST CONDITIONS (NOTES 1 & 2)	LIMITS			UNIT
			Min	Typ	Max	
Clock:						
T_{PC}	Processor cycle time		300			ns
T_{CP}	X1 clock period		150			ns
T_{CH}	X1 clock high time		62			ns
T_{CL}	X1 clock low time		62			ns
Controls:						
T_{HS}	$\overline{HALT}$ set-up time (X1 rising edge)		0			ns
T_{HH}	$\overline{HALT}$ hold time (X1 rising edge)		50			ns
Instructions:						
T_{AS}	X1 falling edge to address stable	CL = 100pF	35		92	ns
T_{IS}	Instruction set-up time (X1 rising edge)		0			ns
T_{IH}	Instruction hold time (X1 rising edge)		50			ns
T_{MCH}	MCLK high delay	X1 = 2.0V	20		55	ns
T_{MCL}	MCLK low delay	X1 = 2.0V	20		55	ns
T_{WH}	X1 falling edge to SC/WC rising edge				80	ns
T_{WL}	X1 falling edge to SC/WC falling edge				80	ns
T_{IIBS}	Instruction to $\overline{LB}/\overline{RB}$ (input phase)				52	ns
T_{IBS}	X1 falling edge to $\overline{LB}/\overline{RB}$ (input phase)		24			ns
T_{OBS}	X1 falling edge to $\overline{LB}/\overline{RB}$ (output phase)				90	ns
T_{IDS}	Input data set-up time (X1 falling edge)		36			ns
T_{IDH}	Input data hold time (X1 falling edge)		50			ns
T_{ODS}	Output data stable (X1 falling edge)				125	ns
T_{ODH}	Output data hold time (X1 falling edge)		35		85	ns
T_{ACC}	Instruction access time	Provided by worst case timing	80			ns
T_{IO}	Data I/O access time	Provided by worst case timing	40			ns

NOTES:
1. Operating temperature ranges are guaranteed after thermal equilibrium has been reached.
2. Unless otherwise noted CL = 300 pF, VIN = 3V.

TIMING CONSIDERATIONS (Commercial Part)

As shown in the "AC CHARACTERISTICS" table for this part, the minimum instruction cycle time is 25 ns, whereas, the maximum is determined by the on-chip oscillator frequency and can be any value the user chooses. With an instruction cycle time of 250 ns, the part can be characterized in terms of absolute values; these are shown in the first "LIMITS" column of the table. When the instruction cycle time is greater than 250 ns, certain parameters are cycle-time dependent; thus, these parameters are specified in terms of the four quarter cycles (T_{1Q}, T_{2Q}, T_{3Q}, and T_{4Q}) that make up one instruction cycle—see 8X300 TIMING DIAGRAM. As the time interval for each instruction cycle increases (becomes greater than 250 ns), the delay for all parameters that are cycle-time dependent is likewise increased. In some cases, these delays have a significant impact on timing relationships and other areas of systems design; subsequent paragraphs describe these timing parameters and reliable methods of calculation.

8X300 TIMING DIAGRAM



NOTE
For an instruction cycle time greater than 250 ns, the I/O bus can be stable sometime within the third quarter (T_{3Q}) cycle.

Timing parameters for the 8X300 are normally measured with reference to X1 or MCLK; those referenced to MCLK are prefaced with an "M" in the mnemonic—TMAS, TMIH, and so on. To determine the timing relationship between a particular signal, say "A" and MCLK, the user should, at all times, use the value specified in the table—DO NOT calculate the value by adding or subtracting two or more parameters that are referenced to X1. When deriving timing relationships between two signals (A to B, etc.) by adding or subtracting the parameter values, the user must consistently use the same parameter reference—MCLK or X1.

System determinants for the instruction cycle time are:

- Propagation delays within the 8X300
- Access time of Program Storage
- Enable time of the I/O port

Normally, the instruction cycle time is constrained by one or more of the following conditions:

Condition 1—Instruction or MCLK to $\overline{LB}/\overline{RB}$ (input phase) plus I/O port access time (TIO) $\leq$ IV data set-up time (Figure 4a).

Condition 2—Program storage access time (TACC) plus instruction to $\overline{LB}/\overline{RB}$ (input phase) plus I/O port access time (TIO) plus IV data (input phase) to address $\leq$ instruction time (Figure 4b).

Condition 3—Program storage access time plus instruction to address $\leq$ instruction cycle time (Figure 4c).

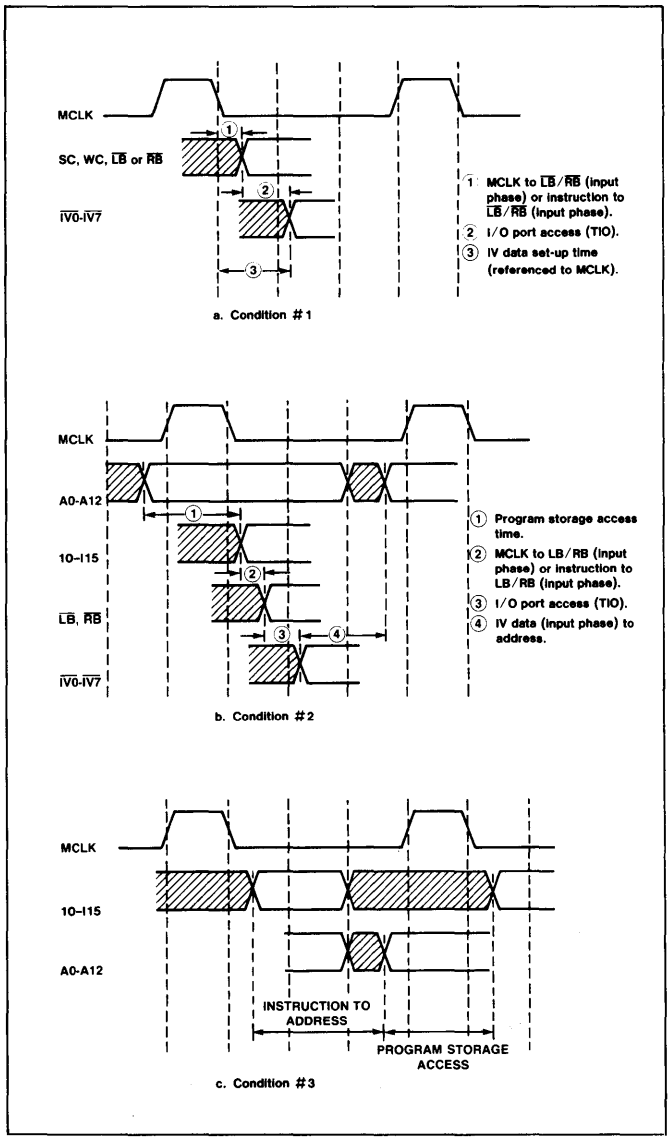


Figure 4. Constraints of 8X300 Instruction Cycle Time

From condition #1 and with an instruction cycle time of 250 ns, the I/O port access time (TIO) can be calculated as follows:

$$TMIBS + TIO \leq TMIDS$$
transposing, $TIO \leq TMIDS - TMIBS$
substituting, $TIO \leq 55ns - 25ns$
result, $TIO \leq 30\ ns$

Using 30 ns for TIO, the constraint imposed by condition #1 can also be used to calculate the minimum cycle time:

$$TMIBS + TIO \leq TMIDS$$
thus, $25ns + 30ns \leq T_{1Q} + T_{2Q} - 70$
$25ns + 30ns \leq \frac{1}{2}$ cycle $- 70$ therefore, the worst-case instruction cycle time is 250 ns. With subject parameters referenced to X1, the same calculations are valid:

$$TIBS + TIO + TIDS \leq \frac{1}{2}\ cycle$$
thus, $70ns + 30ns + 25ns \leq \frac{1}{2}$ cycle therefore, the worst-case instruction cycle time is again 250 ns. From condition #2 and with an instruction cycle time of 250 ns, the program storage access time can be calculated:

$$TACC + TIIBS + TIO + TIVA \leq 250ns$$
transposing, $TACC \leq 250ns - TIIBS - TIO - TIVA$
substituting, $TACC \leq 250ns - 35ns - 30ns - 105ns$
thus, $TACC \leq 80ns$ hence, for an instruction cycle time of 250 ns, a program storage access time of 80 ns is implied. The constraint imposed by condition #3 can be used to verify the maximum program storage access time:

$$TIA + TACC \leq \text{Instruction Cycle}$$
thus, $TACC \leq 250ns - 170ns$
and, $TACC \leq 80ns$, confirming that a program storage access time of 80 ns is satisfactory.

For an instruction cycle time of 250 ns and a program storage access time of 80 ns (Condition #2/Figure 4b), the instruction should be valid 10 ns before the falling edge of MCLK. This relationship can be derived by the following equation:

$$250ns - TMAS - TACC$$
$$= 250ns - 160ns - 80ns$$
$$= 10ns$$

It is important to note that, during the input phase, the beginning of a valid $\overline{LB}/\overline{RB}$ signal is determined by either the instruction to $\overline{LB}/\overline{RB}$ delay (TIIBS) or the delay from the falling edge of MCLK to $\overline{LB}/\overline{RB}$ (TMIBS). Assuming the instruction is valid 10 ns before the falling edge of MCLK and adding the instruction-to-LB/RB delay (TIIBS) = 30ns), the $\overline{LB}/\overline{RB}$ signal will be valid 25 ns after the falling edge of MCLK. With a fast program storage memory and with a valid instruction more than 10 ns before the falling edge of MCLK—the $\overline{LB}/\overline{RB}$ signal will, due to the TMIBS delay, still be valid 25 ns after the falling edge of MCLK. Using a worst-case instruction cycle time of 250 ns, the user cannot gain a speed advantage by selecting a memory with faster access time. Under the same conditions, a speed advantage cannot be obtained by using an I/O port with fast access time (TIO) because the address bus will be stable 80 ns (TAS) after the beginning of the third quarter cycle—no matter how early the IV data input is valid.

Internal Timing and Timing Relationships

All timing and timing-control signals of the 8X300 are generated by the oscillator and sequencer shown in Figure 5. The sequencer outputs direct and control all of the timing parameters specified in the TIMING DIAGRAM. Observe that each input quarter cycle bears a fixed relationship to X1 via the propagation delay.

General and interactive timing relationships pertaining to I/O signals of the 8X300 are shown in Figure 6. Example—in the input phase, the switching point of the $\overline{LB}/\overline{RB}$ signal is caused by the worst-case delay from the instruction to $\overline{LB}/\overline{RB}$ or from the beginning of the first internal quarter cycle to $\overline{LB}/\overline{RB}$; the two arrows pointing to the $\overline{LB}/\overline{RB}$ transition indicate this "either/or" dependency. This information coupled with tabular values and the TIMING DIAGRAM provides the user with the wherewithal to calculate any and all system timing parameters.

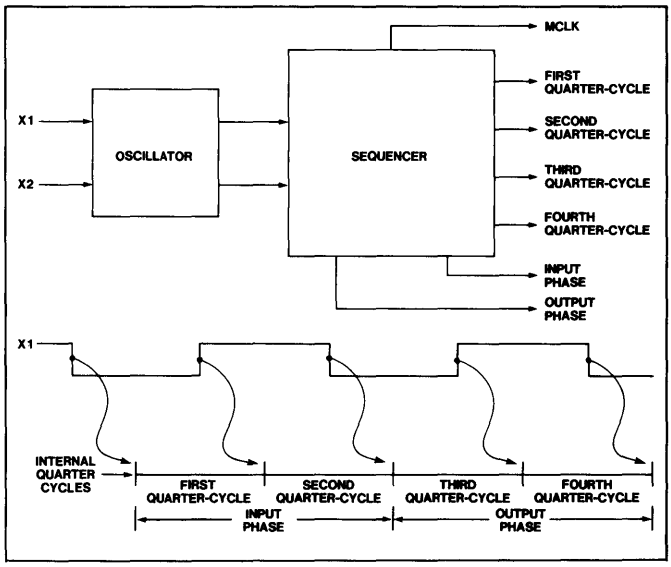


Figure 5. Timing and Timing Control Signals of the 8X300

CLOCK CONSIDERATIONS

The on-chip oscillator and timing-generation circuits of the 8X300 can be controlled by any one of the following methods:

Capacitor: if timing is not critical
Crystal: if precise timing is required
External Drive: if application requires that the 8X300 be synchronized with system clock

Capacitor Timing: A non-polarized ceramic or mica capacitor with a working voltage equal to or greater than 25-volts is recommended. The lead lengths of capacitor should be approximately the same and as short as possible; also, the timing circuits should not be in close proximity to external sources of noise. For various capacitor (C_X) values, the cycle time can be approximated as:

C_X (in pF)	APPROXIMATE CYCLE TIME
100	300 ns
200	500 ns
500	1.1 μs
1000	2.0 μs

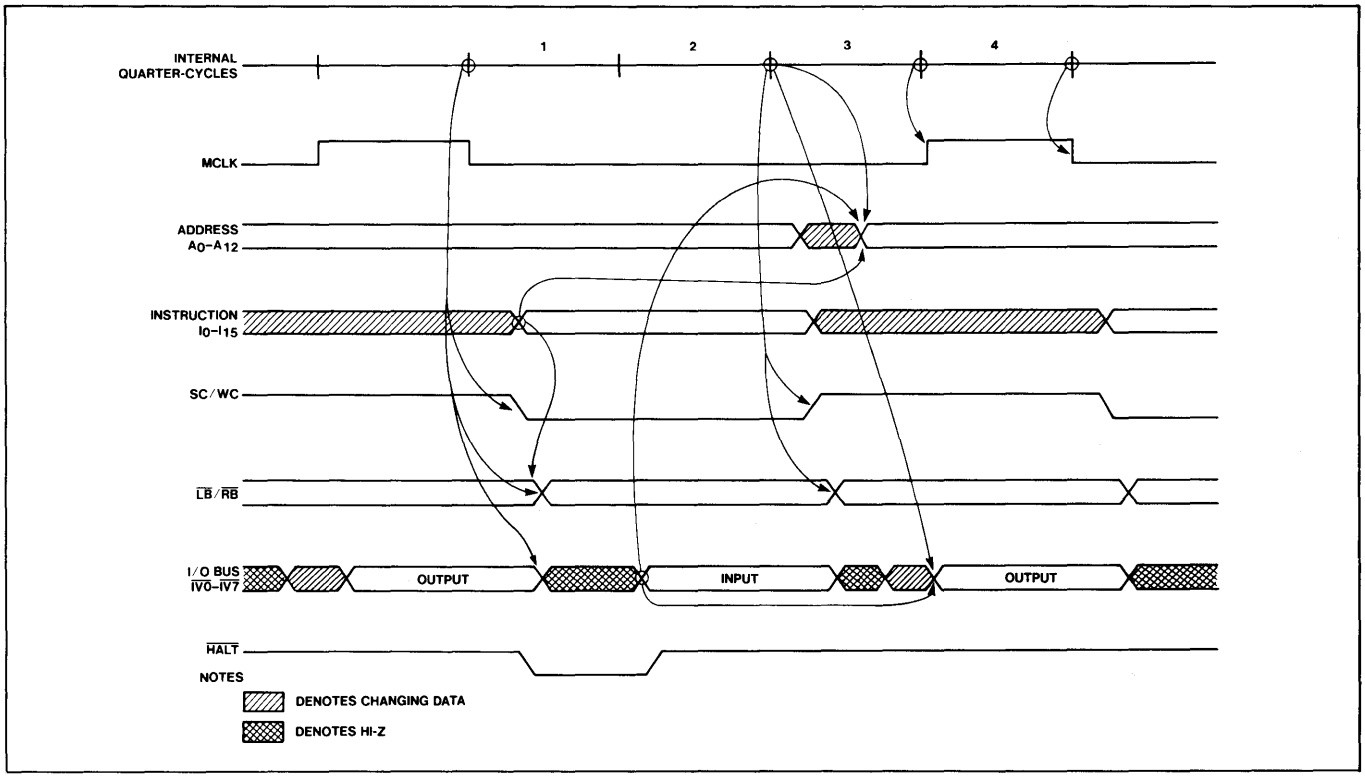


Figure 6. Timing Relationships of 8X300 I/O Signals

Crystal Timing: When a crystal is used, the on-chip oscillator operates at the resonant frequency (f_O) of the crystal; the series-resonant quartz crystal connects to the 8X300 via pins 10 (X1) and 11 (X2). The lead lengths of the crystal should be approximately equal and as short as possible; also, the timing circuits should not be in close proximity to external sources of noise. The crystal should be hermetically sealed (HC type can) and have the following electrical characteristics:
Type: Fundamental mode, series resonant
Impedance at Fundamental: 35-ohms maximum
Impedance at Harmonics and Spurs: 50-ohms minimum

The resonant frequency (f_O) of the crystal is related to the desired cycle time (T) by the equation $f_O = 2/T$; for a cycle time of 250 ns, f_O = 8MHz.

Using an External Clock: The 8X300 can be synchronized with an external clock by simply connecting appropriate drive circuits to the X1/X2 inputs. Figure 7 shows how the on-chip oscillator can be driven from the complementary outputs of a pulse generator. In applications where the microcontroller must be driven from a master clock, the X1/X2 lines can be interfaced to TTL logic as shown in Figure 8.

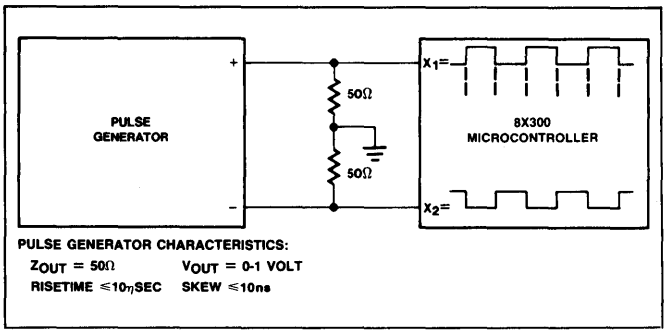


Figure 7. Clocking with a Pulse Generator

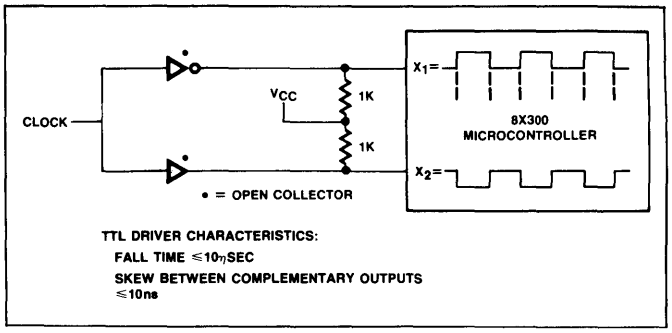


Figure 8. Clocking with TTL

RESET Logic

The $\overline{\text{RESET}}$ line (pin 43) can be driven from a high (inactive) state to a low (active) state at any time with respect to the system clock, that is, the reset function is asynchronous. To ensure proper operation, the $\overline{\text{RESET}}$ line should be held low (active) for one full instruction time. When the line is driven from a high state to an active-low state, several events occur—the precise instant of occurrence is basically a function of the propagation delay for that particular event. As shown in the acompanying $\overline{\text{RESET}}$ timing diagram, these events are:

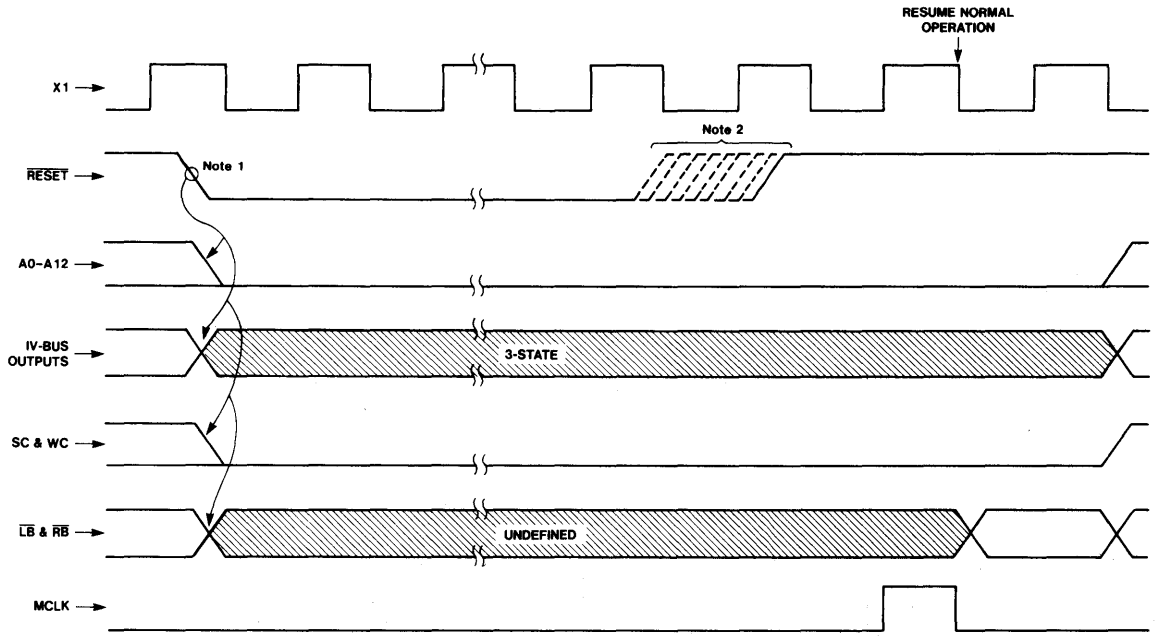
- **The Program Counter and Address Register are set to an all-zero configuration and remain in that state as long as the $\overline{\text{RESET}}$ line is low. Other than PC and AR, reset does not affect other internal registers.**
- **The input/output (IV) bus goes three-state and remains in that mode as long as the $\overline{\text{RESET}}$ line is low.**
- **The Select Command and Write Command signals are driven low and remain inactive as long as the $\overline{\text{RESET}}$ line is low.**
- **The Left Bank/Right Bank signals are undefined for the period in which the $\overline{\text{RESET}}$ line is low.**

During the time $\overline{\text{RESET}}$ is active-low, MCLK is inhibited; moreover, if the $\overline{\text{RESET}}$ line is driven low during the last two quarter cycles, MCLK can be shortened for that particular machine cycle. When $\overline{\text{RESET}}$ line is driven high (inactive)—one-quarter to one full instruction cycle later—MCLK appears just before normal operation is resumed. The $\overline{\text{RESET}}$/MCLK relationship is clearly shown by "B" in the timing diagram. As long as the $\overline{\text{RESET}}$ line is active-low, the $\overline{\text{HALT}}$ signal (described next) is not sampled by internal logic of the 8X300.

HALT Logic

The $\overline{\text{HALT}}$ signal is sampled via internal chip logic at the end of the first internal quarter of each instruction cycle. If, when sampled, the HALT signal is active-low, a halt is immediately executed and the current instruction cycle is terminated; however, the halt cycle does not inhibit MCLK nor does it affect any internal registers of the 8X300. As long as the $\overline{\text{HALT}}$ line is active-low, the SC and WC lines are low (inactive) and the input/output (IV) bus remains in the three-state mode of operation. The halt cycle continues until, when again sampled, the $\overline{\text{HALT}}$ line is found to be high; at this time, normal operation is resumed. Timing for the halt signal is shown in the accompanying diagram.

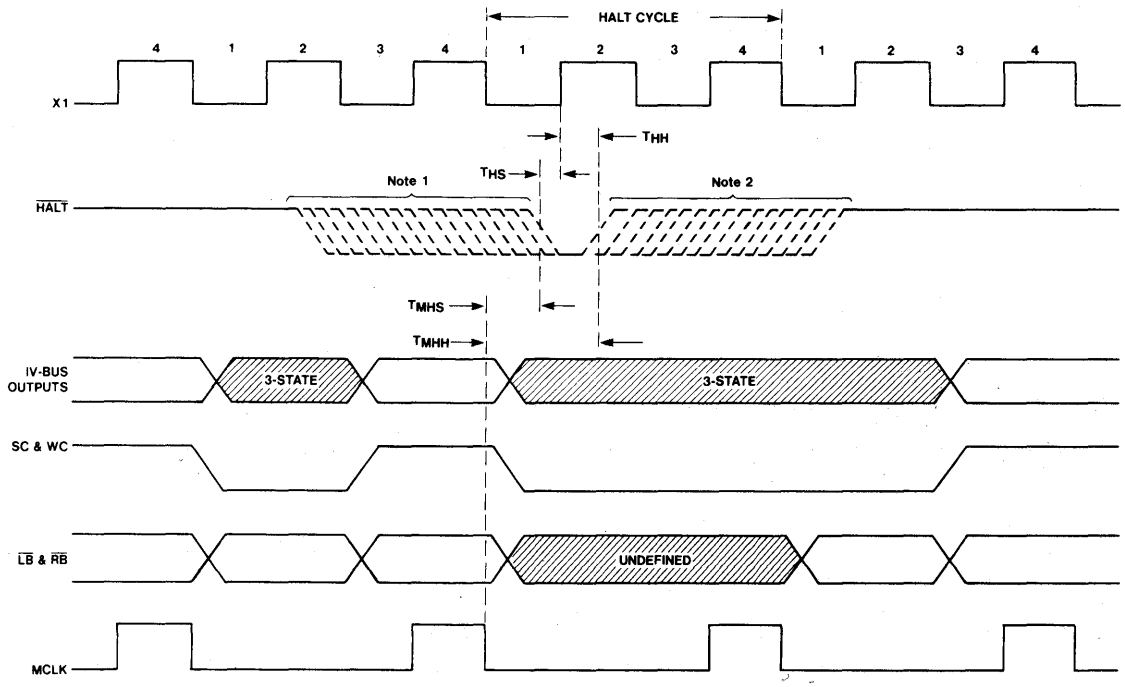
RESET TIMING DIAGRAM



NOTES

1. A High to Low transition of the RESET signal will force the Address Bus to an all-zero configuration.
2. The RESET signal can switch from Low to High at any point within this time interval and, in all cases, MCLK will occur at least one-quarter cycle time later as shown.

HALT TIMING DIAGRAM



NOTES

1. The HALT signal can switch from High to Low at any time during this interval.
2. The HALT signal can switch from Low to High at any time during this interval.

Timing Descriptions:

T_{HS}—set-up time from HALT to X1 (independent of instruction cycle time)

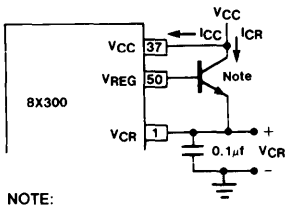
T_{HH}—hold time from X1 to HALT (independent of instruction cycle time)

T_{MHS}—set-up time from MCLK to HALT (dependent upon instruction cycle time)

T_{MHH}—hold time from MCLK to HALT (dependent upon instruction cycle time)

VOLTAGE REGULATOR

All internal logic of the 8X300 is powered by an on-chip voltage regulator that requires an external series-pass transistor. Electrical specifications for the off-chip power transistor and a typical hook-up are shown in the accompanying diagram. To minimize lead inductance, the transistor should be as close as possible to the 8X300 package and the emitter should be ac-grounded via a 0.1-microfarad ceramic capacitor.



PARAMETER	CONDITIONS	LIMITS
h_{fe}	V_{CE} = 2V; 100mA < I_C < 500mA	50
BE_{ON}	V_{CE} = 5V; I_C = 500mA	1V
V_{CESAT}	I_C = 500mA; I_B = 50mA	0.5V
BV_{CEO}		15V
f_t		30MHz

NOTE:
Typical approved parts—2N5320, 2N5337

ARCHITECTURAL OVERVIEW

The Signetics 8X320 Bus Interface Register Array (Figure 1) is a dual-port RAM memory designed for use betwen a host processor and a peripheral processor. Specifically, the register array provides a convenient and economical interface between the 8X300 Microcontroller (secondary port) and User's Host System (primary port); the host can be almost any bus-oriented device—another processor, a minicomputer, or a mainframe computer. The host has 8-bit (byte) or 16-bit (word) access to the primary port; data can be read-from or written-into any memory location as determined by the primary-port address and control lines. The secondary port (8X300 bus) consists of eight input/output lines and four bus control lines. To implement the secondary-port interface, an 8-bit memory location is addressed during one machine cycle and, during another cycle, data is read or written under control of the secondary (8X300) processor. Both primary and secondary ports feature three-state outputs and both ports are bidirectional.

Besides the convenience and economy of a two-port memory, the array also provides simple handshake control via two 8-bit flag registers, logic to facilitate DMA transfers, and a write-protect feature for the primary port in both byte and word modes of operation.

FEATURES

- **16-Byte/2-Port interface**
- **8 or 16 Bit primary-port (Host) interface (user selectable)**
- **8-Bit secondary-port interface**
- **Two 8-Bit flag registers (handshake control)**
- **DMA or programmed I/O operation**
- **Two Three-State Bidirectional Ports**
- **Secondary Port is bus compatible with 8X300**
- **Single 5V supply**
- **40-Pin package**

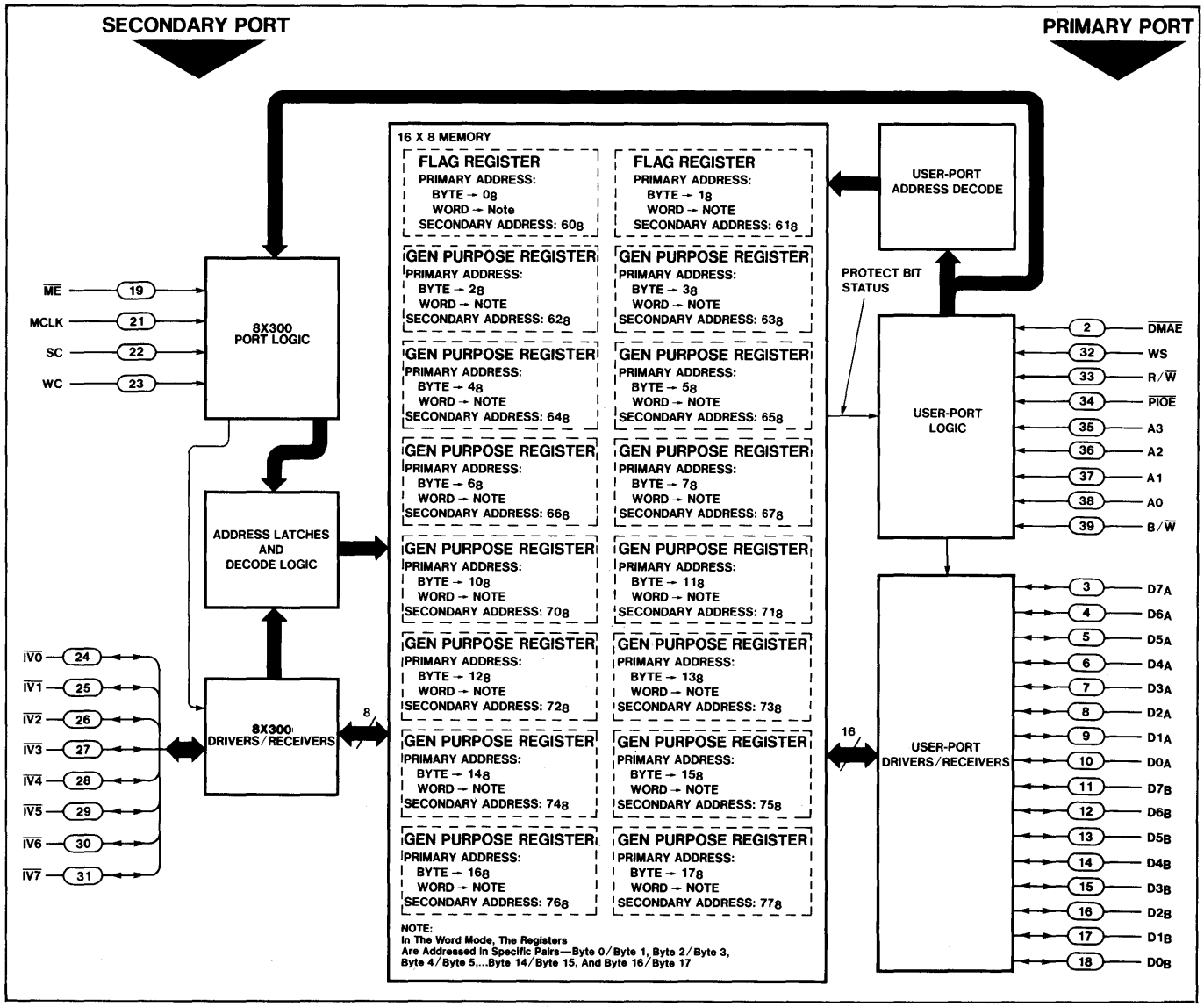
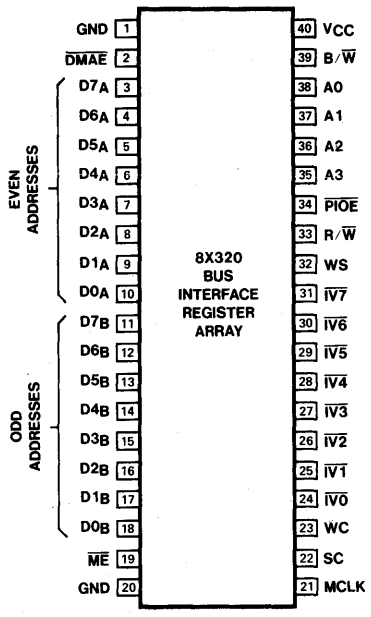


Figure 1. Block Diagram of 8X320 Bus Interface Register Array



PIN NO.	PARAMETER		FUNCTION
1, 20	GND	Ground	Circuit ground.
2	$\overline{DMAE}$	Direct Memory Access Enable	Enables primary port to facilitate DMA transfers; does not affect secondary port.
3-18	$D0_A$-$D7_A$/ $D0_B$-$D7_B$	Primary Data Port	Sixteen 3-state lines used for data transfers to-and-from the primary data port; most significant bit is $D0_A$ and least significant bit is $D7_B$.
19	$\overline{ME}$	Master Enable	Enables secondary port when active low ($\overline{ME}$).
21	MCLK	Master Clock	When MCLK is high, and 8X320 is enabled ($\overline{ME}$), a register location may be either selected or written-into under control of SC and WC.
22	SC	Select Command	With SC high, WC low, MCLK high and $\overline{ME}$ low, data on $\overline{IV0}$ through $\overline{IV7}$ is interpreted as an address. If any one of the 16 register addresses (60_8-77_8) matches that on the I/O (IV) bus, that particular register is selected and remains selected until another address on the same bank (i.e. $\overline{ME}$ = low) is output on the I/O bus—at which time, the old register is deselected and a new register may or may not be selected.
23	WC	Write Command	With WC high, SC low, MCLK high, and $\overline{ME}$ low, the selected register stores contents of $\overline{IV0}$-$\overline{IV7}$ as data.
24-31	$\overline{IV0}$-$\overline{IV7}$	Secondary Data Port	Eight 3-state lines used to transfer data or I/O address to-and-from the secondary data port; most significant bit is $\overline{IV0}$ and least significant bit is $\overline{IV7}$.
32	WS	Write Strobe	When active high, data appearing at the primary port ($D0_A$-$D7_A$/$D0_B$-$D7_B$) is stored in the register array if the primary port is in the *write* mode.
33	R/$\overline{W}$	Read/Write Control	When this signal is high, primary port is in *read* mode; when signal is low, primary port is in *write* mode.
34	$\overline{PIOE}$	Programmed I/O Enable	When active low, primary port operates in programmed input/output mode with register to be read-from or written-into selected by A0-A3.
35-38	A0-A3	Primary Port Address Select	Selects register or register-pair that primary port is to read-from or write-into. Most significant bit is A3; least significant bit is A0.
39	B/$\overline{W}$	Byte/Word	When signal is high, the primary port operates in the byte (8-bit) mode; when signal is low, the primary port operates in the word (16-bit) mode.
40	V_{CC}	Power	+5 volts.

All barred symbols ($\overline{DMAE}$, etc.) denote signals that are asserted (or active) when low (logical 0); signals that are not barred are asserted in the high state (logical 1).

OPERATING CHARACTERISTICS

Memory Organization

Memory and address correlation for the 16-register array is shown in Figure 2. From the primary port, the sixteen 8-bit registers can be addressed in either byte (8-bit) or word (16-bit) format; in the word mode, the registers are addressed in pairs—$0_8/1_8$, $2_8/3_8$, $4_8/5_8$,... $14_8/15_8$, and $16_8/17_8$. From the secondary port, all registers are addressed in byte format—60_8 through 77_8. The memory consists of two 8-bit flag registers and fourteen 8-bit general-purpose registers. The flag registers facilitate information transfers between the two ports and, in addition, they protect certain registers from being written into from the primary port.

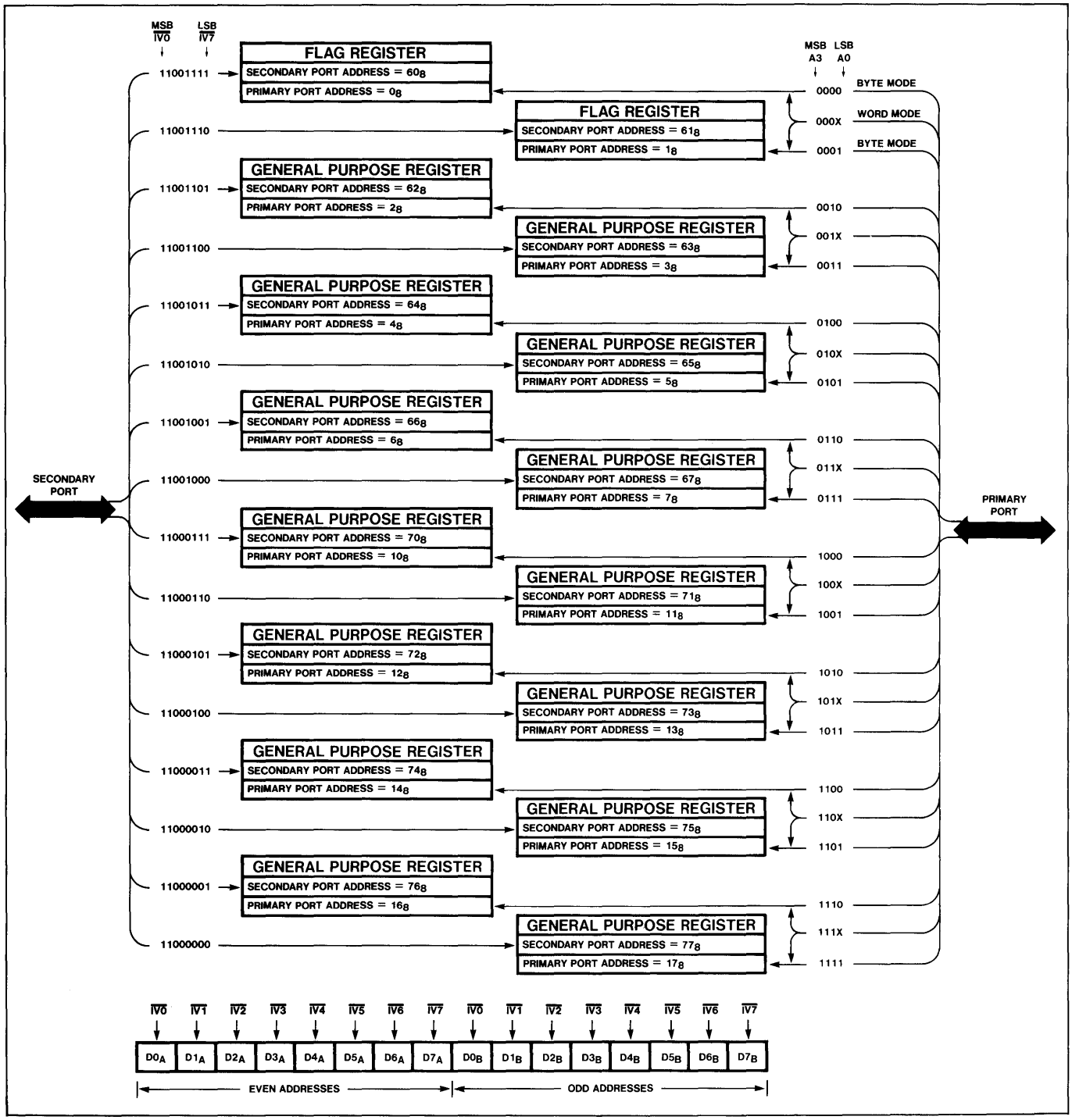


Figure 2. Memory and Address Organization for the 8X320

In either byte or word mode, the write-protect logic, implemented by bits F0 and F1 of register 60_8, inhibits the primary port from writing into addresses 16_8 and 17_8. The write-protect bits (F0 and F1) can be either read or written from the secondary port; the bits are read-only from the primary port.

As shown in Table 1, flag bits F2 through F7 of 60_8 and F0 through F7 of 61_8 are controlled by the fourteen general-purpose registers. When any one of these registers is written into by either port, the corresponding flag bit for that register is automatically set by internal logic of the 8X320. When information is read from any register, the corresponding flag bit must be reset by user software. Except for the write-protect bits, all other flag bits can be read or written from the primary or the secondary port.

Table 1. Control of the Two Flag Registers

Flag Registers	60_8 (0_8)	F2 F3 F4 F5 F6 F7
	61_8 (1_8)	F0 F1 F2 F3 F4 F5 F6 F7
Octal Address of Controlling Byte	Primary	2 3 4 5 6 7 10 11 12 13 14 15 16 17
	Secondary	62 63 64 65 66 67 70 71 72 73 74 75 76 77

FUNCTION AND CONTROL OF PRIMARY PORT

The primary port provides an 8-bit (byte) or 16-bit (word) interface between the 16-byte memory and the user's host system. If the host is an 8-bit system (or 16-bit system operating in Byte mode), the sixteen bidirectional I/O lines must be tied together ($D0_A$ to $D0_B$, $D1_A$ to $D1_B$, . . . and $D7_A$ to $D7_B$); when data is input or output on $D0_A$ through $D7_A$, the remaining eight lines ($D0_B$ through $D7_B$) are high-Z and vice-versa.

Other than the Byte/$\overline{\text{Word}}$ control line, specific operating characteristics of the primary port are controlled by two signals—$\overline{\text{PIOE}}$ (Programmed I/O Enable) and $\overline{\text{DMAE}}$ (Direct Memory Access Enable). When $\overline{\text{PIOE}}$ is active (low) and $\overline{\text{DMAE}}$ is inactive (high), the primary port operates in the programmed I/O mode—refer to Table 2; in this mode of operation, the register to be read-from or written-into is determined by four address lines (A0 through A3) and the Byte/$\overline{\text{Word}}$ control line—see Figure 2 and Table 3. In the DMA mode of operation, A1, A2, and A3 are not used; data is read-from or written-into preassigned registers: bytes 16_8 (76_8) and 17_8 (77_8) for the byte mode of operation and bytes 14_8 (74_8)/15_8 (75_8) and 16_8 (76_8)/ 17_8 (77_8) for the word mode of operation. In both cases, switching between bytes 16_8 and 17_8 in the byte mode and 14_8/15_8 and 16_8/17_8 in the word mode is controlled by A0 (the least significant address bit). Refer to Table 4.

Table 2. Mode Control of Primary Port

MODE	$\overline{\text{PIOE}}$	$\overline{\text{DMAE}}$
Disabled (output high-Z)	1	1
Programmed I/O	0	1
DMA	X	0

X = Don't Care

Table 3 defines programmed I/O operation of the primary port in terms of read/$\overline{\text{write}}$ functions and Byte/$\overline{\text{Word}}$ control. In the byte mode, data is read-from or written-into the even addresses (0_8, 2_8, 4_8, 6_8, 10_8, 12_8, 14_8, and 16_8) via data lines $D0_A$ through $D7_A$; data is read-from or written-into odd addresses (1_8, 3_8, 5_8, 7_8, 11_8, 13_8, 15_8, and 17_8) via data lines $D0_B$ through $D7_B$. When A0 is low (logical 0), even addresses are selected and when A0 is high (logical 1), odd addresses are selected; thus, A0 is the LSB of a 4-bit address. In the word mode, the state of A0 is irrelevant, since both the odd and even bytes are, simultaneously, read-from or written-into; thus, a register pair is selected by a 3-bit address, A1 being the LSB.

In the DMA mode of operation with $\overline{\text{DMAE}}$ set to 0 and other conditions satisfied, data is directly transferred to-or-from specified memory locations under control of Byte/$\overline{\text{Word}}$, R/$\overline{\text{W}}$, and A0. The state of the Byte/$\overline{\text{Word}}$ control line determines whether the data word is 8-bits or 16-bits. The A0 address line correlates eight of the sixteen data lines ($D0_A$-$D7_A$ or $D0_B$-$D7_B$) with the proper byte/word location. Thus, in the word mode, the exchange of data between the mem-

Table 3. Primary Port Operating in Programmed I/O Mode

MODE	B/$\overline{\text{W}}$	A0	$D0_A$-$D7_A$ (Even Addresses)	$D0_B$-$D7_B$ (Odd Addresses)
Read	0 (Word)	X	Stored Data	Stored Data
Read	1 (Byte)	0	Stored Data	HI-Z
Read	1 (Byte)	1	HI-Z	Stored Data
Write	0 (Word)	X	Write	Write
Write	1 (Byte)	0	Write	No Change
Write	1 (Byte)	1	No Change	Write

X = Don't Care

Table 4. DMA Operation of the Primary Port

MODE	BYTE/WORD	A0	$D0_A$-$D7_A$	$D0_B$-$D7_B$
Read	0 (Word)	0	Data stored in byte 14_8	Data stored in byte 15_8
Read	0 (Word)	1	Data stored in byte 16_8	Data stored in byte 17_8
Read	1 (Byte)	0	Data stored in byte 16_8	HI-Z
Read	1 (Byte)	1	HI-Z	Data stored in byte 17_8
Write	0 (Word)	0	Write to byte 14_8	Write to byte 15_8
Write	0 (Word)	1	Write to byte 16_8	Write to byte 17_8
Write	1 (Byte)	0	Write to byte 16_8	HI-Z
Write	1 (Byte)	1	HI-Z	Write to byte 17_8

ory and the port occurs via $D0_A$-$D7_A$ for bytes 14_8 and 16_8 and via $D0_B$-$D7_B$ for bytes 15_8 and 17_8. The byte mode of operation is similar, except that the unused eight lines are high-Z.

FUNCTION AND CONTROL OF SECONDARY PORT

The secondary port provides an 8-bit interface between the sixteen memory registers and the 8X300 (or other processor). As shown in Table 5, the secondary-port interface is controlled by five input signals and a status latch. The status latch is set when SC is high (MCLK high / $\overline{ME}$ low) and a valid memory address (60_8-77_8) is presented to the 8X320 via the secondary data port ($\overline{IV0}$-$\overline{IV7}$). The latch is cleared by internal logic when an invalid memory address is presented at the secondary port. In all read / write operations from the secondary port, the status latch acts like a master enable; data can be transferred only if the status latch is set.

Table 5. Functional Control of Secondary Port

$\overline{ME}$	SC[1]	WC[1]	MCLK	R/$\overline{W}$	STATUS LATCH	FUNCTION OF SECONDARY BUS
L	L	L	X	X	Set	Output data from 8X320 memory to 8X300.
L	L	H	H	H	Set	Data from 8X300 is input and read-into a previously-selected memory location of the 8X320 (Note 2).
L	L	H	H	L	Set	With the primary port in the write mode (R/$\overline{W}$ = 0), the secondary port is overridden and cannot write to the same register addressed by the primary port; however, the register addressed by the primary port can be read and any other register can be read-from or written-into from the secondary port (Note 2).
L	H	L	H	X	X	Data transmitted to the secondary port via the IV bus is interpreted as an address; if address is within range of 60_8-77_8 the memory status latch is subsequently set.
L	L	H	L	X	X	Inactive
L	H	L	L	X	X	Inactive
L	L	X	X	X	Not Set	Inactive
H	X	X	X	X	X	Inactive

Notes:
1. The SC and WC lines should never both be high at the same time; the 8X300 processor never generates this condition.
2. During read or write operations, the same register can be simultaneously addressed from either port. For any write operation by both ports on the same register, the primary port has priority; other than this, the 8X320 does not indicate error conditions or resolve conflicts.
3. X = Don't Care

DC CHARACTERISTICS $0°C \leq T_A \leq 70°C$; $4.75V \leq V_{CC} \leq 5.25V$

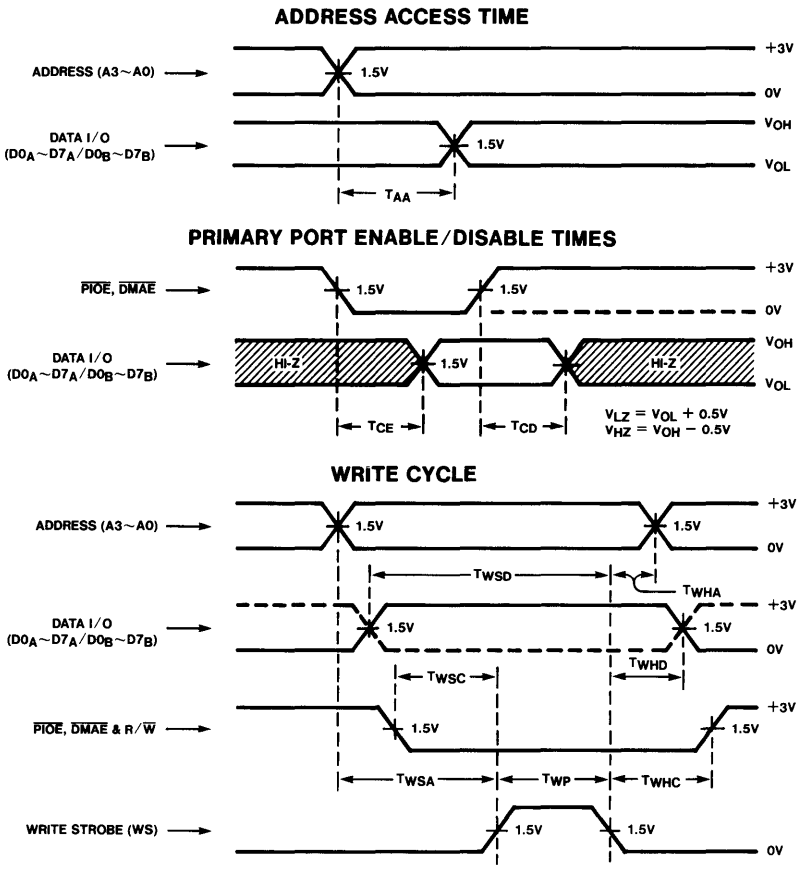
PARAMETER		TEST CONDITIONS[1, 2]	LIMITS			UNIT
			Min	Typ	Max	
V_{CC}	Supply voltage		4.75	5	5.25	V
$V_{IN\ (L)}$	Low level input voltage				0.80	V
$V_{IN\ (H)}$	High level input voltage		2.0			V
V_{OL}	Low level output voltage	V_{CC} = 4.75V; I_{OL} = 16mA			0.55	V
V_{OH}	High level output voltage	V_{CC} = 4.75V; I_{OH} = −3mA	2.40			V
V_{CL}	Input clamp voltage	I_I = −5mA			−1.00	V
I_{CC}	Supply current	V_{CC} = 5.25V (Both ports high-Z)			270	mA
I_{OS}	Short circuit output current[3]	V_{CC} = 4.75V	−20		−100	mA
$I_{IN\ (L)}$	WC, MCLK, SC, & $\overline{ME}$	V_{CC} = 5.25V; V_{IL} = 0.50V			−1.0	mA
$I_{IN\ (L)}$	B/$\overline{W}$	V_{CC} = 5.25V; V_{IL} = 0.50V			−1.6	mA
$I_{IN\ (L)}$	A0-A3	V_{CC} = 5.25V; V_{IL} = 0.50V			−1.0	mA
$I_{IN\ (L)}$	$\overline{DMAE}$	V_{CC} = 5.25V; V_{IL} = 0.5V			−800	µA
$I_{IN\ (L)}$	WS, $\overline{PIOE}$, & R/$\overline{W}$	V_{CC} = 5.25V; V_{IL} = 0.5V			−400	µA
$I_{IN\ (L)}$	$\overline{IV0}$-$\overline{IV7}$	V_{CC} = 5.25V; V_{IL} = 0.5V			−400 each line	µA
$I_{IN\ (L)}$	$D0_A$-$D7_A$/$D0_B$-$D7_B$	V_{CC} = 5.25V; V_{IL} = 0.5V			−400 each line	µA
$I_{IN\ (H)}$	WC, SC, MCLK, & $\overline{ME}$	V_{CC} = 5.25V; V_{IH} = 5.25V			100	µA
$I_{IN\ (H)}$	B/$\overline{W}$	V_{CC} = 5.25V; V_{IH} = 5.25V			240	µA
$I_{IN\ (H)}$	A0	V_{CC} = 5.25V; V_{IH} = 5.25V			120	µA
$I_{IN\ (H)}$	A1-A3	V_{CC} = 5.25V; V_{IH} = 5.25V			60	µA
$I_{IN\ (H)}$	$\overline{DMAE}$	V_{CC} = 5.25V; V_{IH} = 5.25V			120	µA
$I_{IN\ (H)}$	WS, $\overline{PIOE}$, & R/$\overline{W}$	V_{CC} = 5.25V; V_{IH} = 5.25V			60	µA
$I_{IN\ (H)}$	$\overline{IV0}$-$\overline{IV7}$ and $D0_A$-$D7_A$/$D0_B$-$D7_B$	V_{CC} = 5.25V; V_{IH} = 5.25V			100	µA

Notes:
1. Operating temperature ranges are guaranteed after terminal equilibrium has been reached.
2. All voltages are measured with respect to ground terminal.
3. Short only one output at a time.

AC CHARACTERISTICS OF PRIMARY PORT

$0°C \leq T_A \leq 70°C$; $4.75V \leq V_{CC} \leq 5.25V$

Loading: See Test Circuit

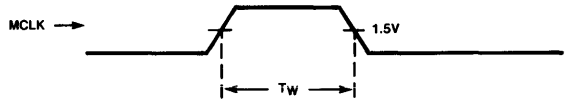


	PARAMETER	FROM	TO	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
T_{AA}	Address access time	A3-A0	$D0_A$-$D7_A$/ $D0_B$-$D7_B$			40	nS
T_{CE}	Primary port enable time	↓ $\overline{PIOE}$ ↓ $\overline{DMAE}$	$D0_A$-$D7_A$/ $D0_B$-$D7_B$			30	nS
T_{CD}	Primary port disable time	↑ $\overline{PIOE}$ ↑ $\overline{DMAE}$	$D0_A$-$D7_A$/ $D0_B$-$D7_B$			35	nS
T_{WSA}	Address setup time	A3-A0	↑ WS	40			nS
T_{WHA}	Address hold time	↓ WS	A3-A0	10			nS
T_{WSD}	Primary port data setup time	$D0_A$-$D7_A$/ $D0_B$-$D7_B$	↓ WS	30			nS
T_{WHD}	Primary port data hold time	↓ WS	$D0_A$-$D7_A$/ $D0_B$-$D7_B$	10			nS
T_{WSC}	Write mode control setup time	$\overline{PIOE}$ $\overline{DMAE}$ R/$\overline{W}$	↑ WS ↑ WS ↑ WS	30 40 30			nS nS nS
T_{WHC}	Write mode control hold time	↓ WS	$\overline{PIOE}$ $\overline{DMAE}$ R/$\overline{W}$	10 10 10			nS nS nS
T_{WP}	Write strobe pulse width			25			nS

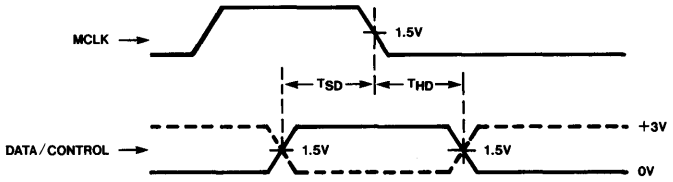
AC CHARACTERISTICS OF SECONDARY PORT

$0°C \leq T_A \leq 70°C$, $4.75V \leq V_{CC} \leq 5.25V$
Loading: See Test Circuit

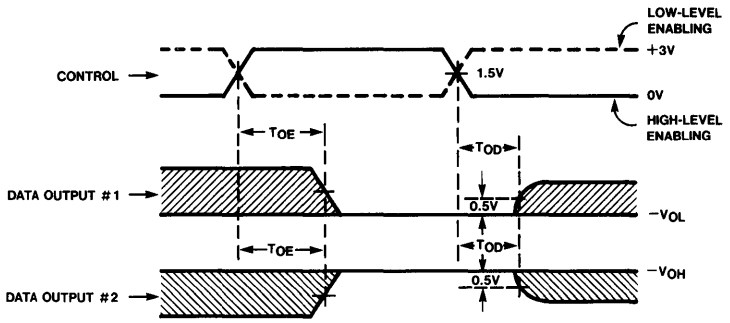
CLOCK PULSE WIDTH



SETUP AND HOLD TIMES

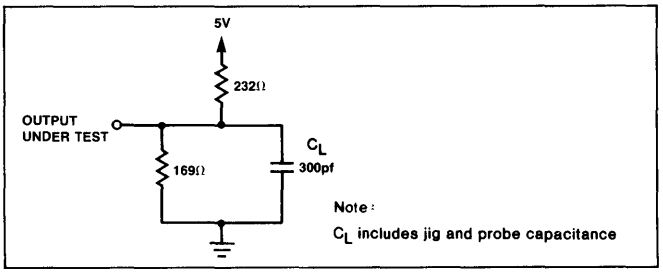


OUTPUT ENABLE/DISABLE TIMES (TRI-STATE OUTPUTS)



	PARAMETER	FROM	TO	LIMITS			UNIT
				Min	Typ	Max	
t_W	MCLK pulse width			30			nS
T_{SD}	Data setup time	$\overline{IV0}$-$\overline{IV7}$	↓ MCLK	35			nS
	$\overline{ME}$ Setup time	$\overline{ME}$	↓ MCLK	30			nS
	SC Setup time	SC	↓ MCLK	30			nS
	WC Setup time	WC	↓ MCLK	30			nS
T_{HD}	Data hold time	↓ MCLK	$\overline{IV0}$-$\overline{IV7}$	0			nS
	$\overline{ME}$ Hold time	↓ MCLK	$\overline{ME}$	0			nS
	SC Hold time	↓ MCLK	SC	0			nS
	WC Hold time	↓ MCLK	WC	0			nS
T_{OE}	Output enable	$\overline{ME}$, SC, or WC	$\overline{IV0}$-$\overline{IV7}$			30	nS
T_{OD}	Output disable	$\overline{ME}$, SC, or WC	$\overline{IV0}$-$\overline{IV7}$			25	nS

TEST CIRCUIT



ORDERING INFORMATION

Order number: N8X320
Packaging information: refer to Signetics price list
Supply voltage: 5V (±5%)
Operating temperature range: 0°C to +70°C

ARCHITECTURAL OVERVIEW

The Signetics 8X330 Floppy Disk Formatter/Controller (Figure 1) is a monolithic peripheral device of the 8X300 family. The chip uses Bipolar-Schottky/I²L-Technology and some very unique features to provide 8X330 customers with a competitive edge in both simple and complicated disk-controller designs. The competitive advantage is measurable in terms of "systems parts count", "error-correction capabilities", and "overall design concepts" that are applications oriented. Except for a crystal, a capacitor, an external transistor acting as a series-pass element for the on-chip voltage regulator, an active low-pass filter, and an optional off-chip voltage controlled oscillator (refer to Features and Option), the 8X330 contains all processing circuits and the required control logic to encode/decode double-density (MFM/M²FM) and single-density (FM) codes. Even the data-separation and write-precompensation logic are located on the chip; in addition, 16-bytes of scratch-pad RAM are provided for storage of various control/status parameters.

FEATURES

- **Single or double density encoding/decoding**
- **On-chip data separator**
- **Programmable:**
  - **FM, MFM, and M²FM encoding/decoding**
  - **Preamble Polarity**
  - **Data transfer rate**
  - **Address mark encoding/decoding**
  - **Sector length**
  - **Output port (7-bits disk command)**
  - **Input port (5-bits disk status)**
- **Write Precompensation with on/off control**
- **On-chip phase lock loop**
- **CRC generator with software-controlled error correction capabilities**
- **40-pin package**
- **+5 volt operation**

OPTION: External Voltage Controlled Oscillator (VCO). For critical applications, window margins can be improved by as much as 6% with the use of an external VCO.

NOTES
1. Components shown with dotted lines are located off-chip.
2. Certain bits of the status/control registers connected to the I/O bus are 'read-only'; this distinction is not shown in the diagram.
3. Greater detail for the more complicated blocks (Phase-Lock Loop, Data Separator, etc.) are shown in other areas of this data sheet.

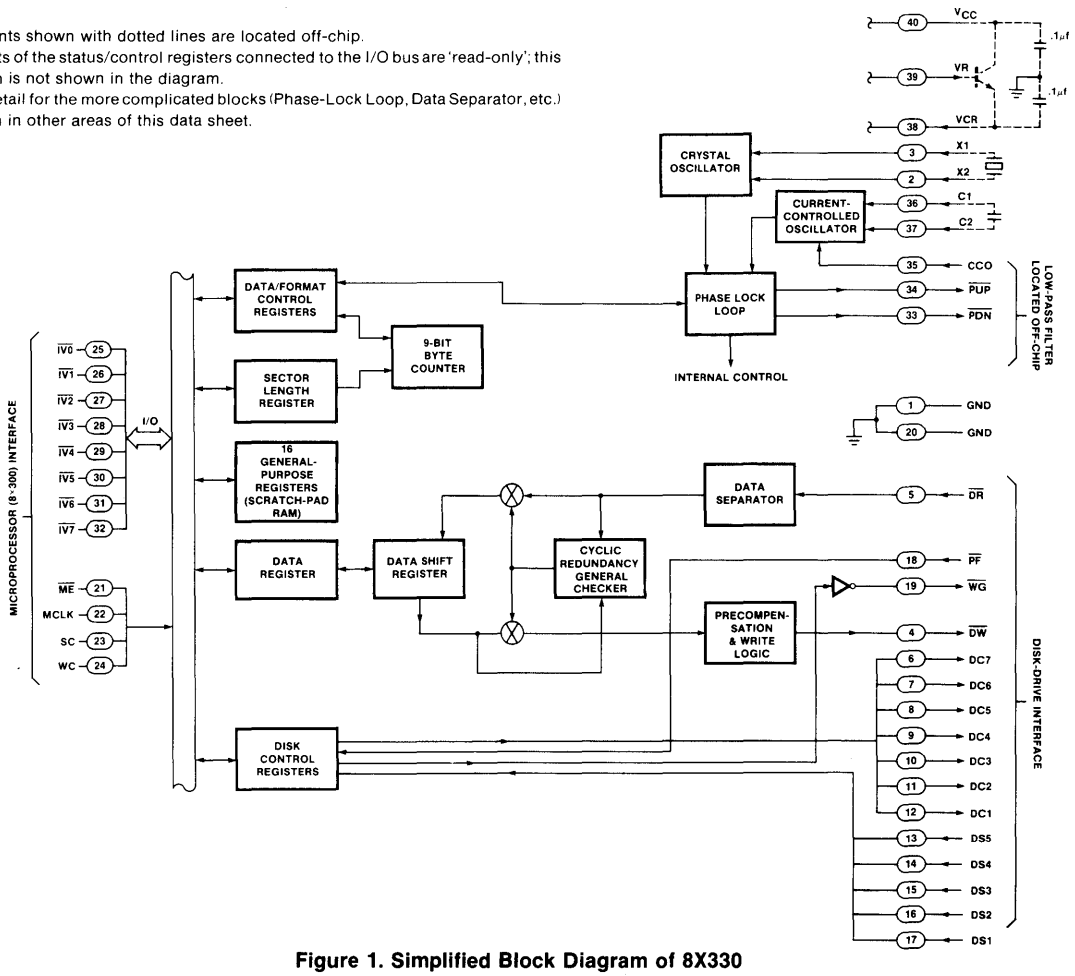
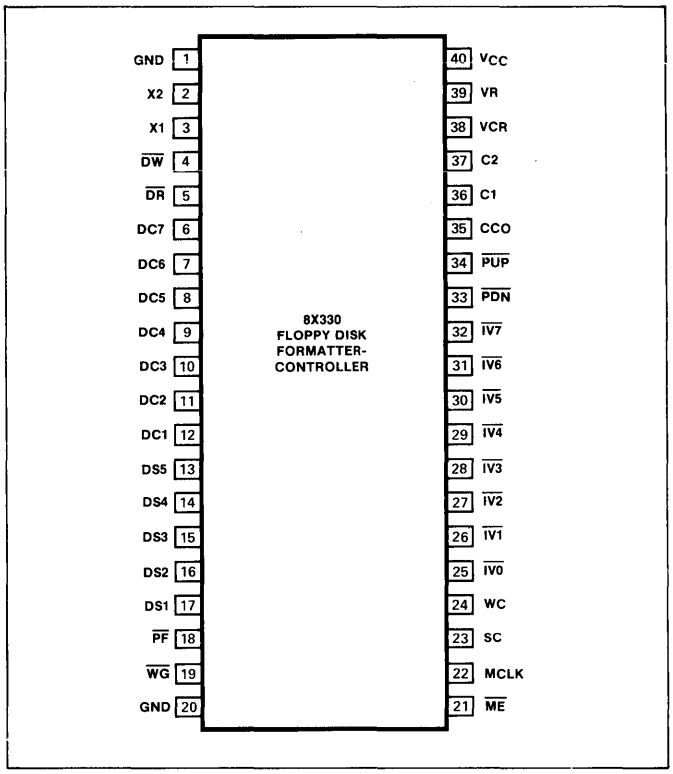


Figure 1. Simplified Block Diagram of 8X330

8X330 PACKAGE/PIN DESIGNATIONS



PIN NO.	MNEMONIC & DEFINITION		FUNCTION
1, 20	GND	Ground	Circuit ground
2, 3	X1, X2	Crystal inputs	Inputs from a crystal that determines frequency of an on-chip crystal oscillator.
4	$\overline{DW}$	Data write	A series of negative-going pulses transmitted to the disk drive. The data write signal produces pulses (with precompensation, if required) for data and clock in accordance with the applicable encoding rules (FM, MFM or M²FM).
5	$\overline{DR}$	Data read	Negative-going pulses transmitted from the disk drive to a Schmitt-trigger input of the 8X330; these pulses represent encoded data and clock from the disk media.
6-12	DC1-DC7	Disk commands	Seven outputs from the 8X330 that allow general-purpose control, of one or more disk drives.
13-17	DS1-DS5	Disk status	Five general-purpose Schmitt-trigger inputs from the disk drive (or drives) that provide status information for the 8X300.
18	$\overline{PF}$	Power fail	Schmitt-trigger input from external logic that is active (low) when the "user-sensed" power supply voltage drops below a predetermined value.
19	$\overline{WG}$	Write gate	When active (low), this 40-milliampere open-collector output enables writing to the disk media. When $\overline{PF}$ is low, the write gate is inhibited during periods of power supply uncertainty.
21	$\overline{ME}$	Master enable	When this input signal is active (low), the 8X330 can be accessed and enabled by the 8X300. (Refer to the $\overline{LB}$ and $\overline{RB}$ pinout discriptions of the 8X300 for further detail.)
22	MCLK	Master clock	When active high and with $\overline{ME}$ in the active-low state, this input signal provides a means whereby the I/O output from the 8X300 is interpreted as an enabling address (provided there is an address match) or as input data (if one of the 8X330 registers has already been selected).
23	SC	Select command	When this signal is active (high), the information output on pins $\overline{IV0}$-$\overline{IV7}$ of the 8X300 is interpreted as an address input by the 8X330.
24	WC	Write command	When this signal is active (high), the information output on pins $\overline{IV0}$-$\overline{IV7}$ of the 8X300 is interpreted as input data by the 8X330.
25-31	$\overline{IV0}$-$\overline{IV7}$	Input/output lines	Eight three-state input/output lines that provide bidirectional data transfers between the 8X300 and the enabled I/O device; $\overline{IV7}$ is the *Least Significant Bit.*
33	$\overline{PDN}$	Pump down output	Open-collector output of on-chip phase detector which indicates (by a negative-going, quantized, pulse-width modulated signal) that internal CCO frequency is too high.
34	$\overline{PUP}$	Pump up output	Open-collector output of on-chip phase detector which indicates (by a negative-going, quantized, pulse-width modulated signal) that internal CCO frequency is too low.
35	CCO	Frequency Control Input for Current-Controlled Oscillator	Variable input current from external low-pass filter that controls the frequency of the oscillator.
36-37	C1, C2	Capacitor input terminals	Inputs for capacitor that determines center frequency of the current-controlled oscillator.
38	VCR	Regulated supply voltage	DC voltage input from emitter of external series-pass transistor; this voltage powers internal logic of chip.
39	VR	Reference voltage	Reference voltage output to base of series-pass transistor; this reference controls VCR.
40	V_{CC}	Supply voltage	+5 volt power.

SYSTEM INTERFACE

A typical floppy disk controller using an 8X300 microcontroller and the 8X330 is shown in Figure 2. The non-shaded portion of this particular configuration can service the command, status, and input/output requirements of two double-sided disk drives and, under software supervision, the system can read/write single-density (FM) or double-density (MFM/M²FM) codes. Interface requirements are simple—on one hand, consisting of the 8X300 microcontroller and, on the other, the two disk drives. All 8X330 control and data registers directly linked to the microprocessor interface (Figure 1) are addressable and appear to the 8X300 as simple I/O ports; a 13-bit address bus and a 16-bit instruction bus provide communications between the 8X300 and up to 8K of microprogram storage.

The disk-drive interface consists of seven (7) output control lines (DC1-DC7), five (5) input status lines (DS1-DS5), a write gate ($\overline{WG}$), a data- write output ($\overline{DW}$), and a data-read input ($\overline{DR}$). The twelve command/status lines are not dedicated; thus, the user can assign system functions to best suit a given application. As shown in Figure 2, all control lines except $\overline{WG}$ are buffered to accommodate a reasonable distance between the controller and the disk media; the Write Gate, being a 40-milliampere output, requires no buffering.

As shown by the shaded part of Figure 2, the control and status lines can be expanded with peripheral hardware—the 8T32 (in this example) being only one method of implementation. Using this particular technique, one I/O port is totally dedicated to output control, whereas, the other port is totally dedicated to input status. With additional hardware and supporting software, the disk-drive system can be expanded without limit; however, from a point of being practical, five or six drives is sufficient for most applications. By using the programmable features of the 8X330, the user can emphasize and prioritize those system parameters that are most important—economics, reliability and/or speed.

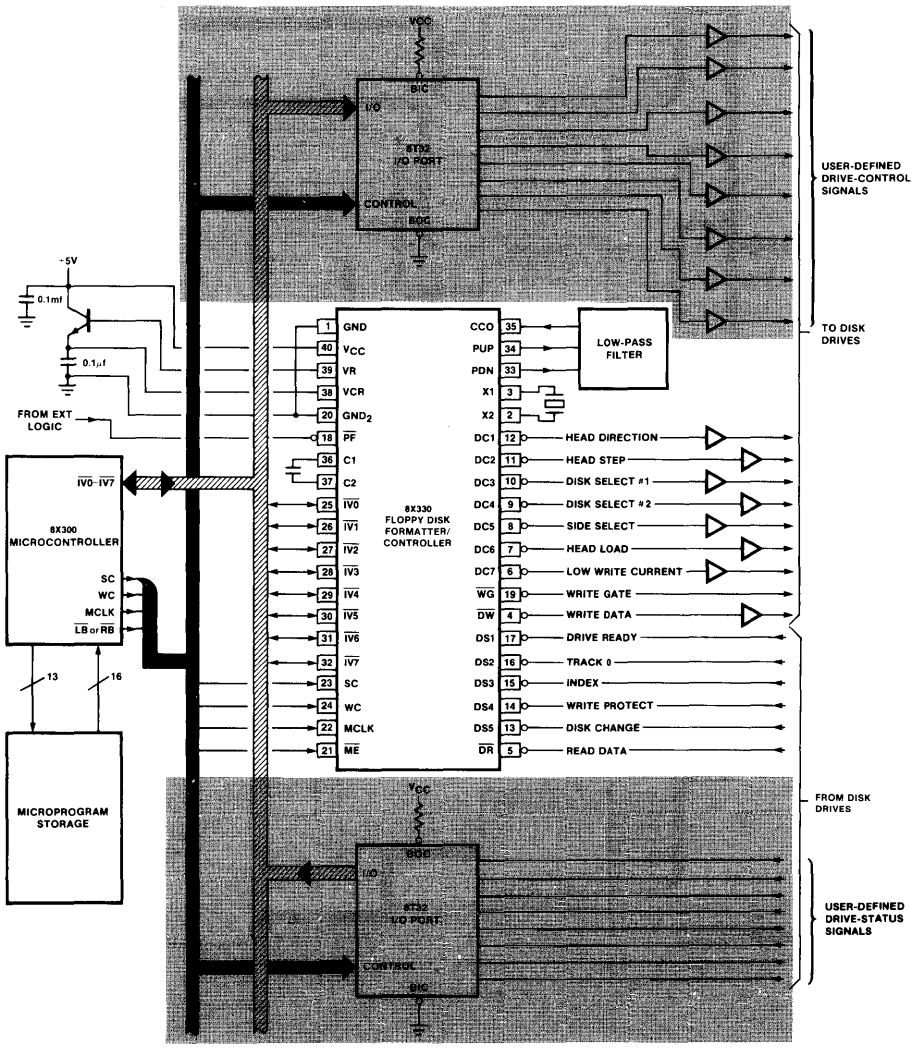


Figure 2. Typical Interface Using an 8X300 Microcontroller

FUNCTIONAL OPERATION

As shown in Figure 2, the interface between the 8X300 and 8X330 consists of twelve (12) lines—$\overline{IV0}$-$\overline{IV7}$, SC, WC, MCLK, and $\overline{ME}$; the Master Enable ($\overline{ME}$) input (pin 21) is driven from either the $\overline{LB}$ (Left Bank) or $\overline{RB}$ (Right Bank) output of the 8X300. An expanded view of this interface is shown in Figure 3 and, as indicated, the 8X330 appears as a number of addressable registers (110_8-127_8 and 132_8-137_8) under input/output control of the 8X300. These registers are used for general-purpose storage, data-transfer operations, disk commands, disk status, and various control functions. Design-oriented information for these registers and other data-processing/logic functions of the 8X330 are described in the paragraphs that follow; in all of these registers, bit 0 is the Most Significant Bit (MSB).

NOTE
When power is first applied to the 8X330, the Disk Command lines (DC1-DC7), the Write Gate ($\overline{WG}$) output, and contents of Command/Status Register #2 (CSR #2) are set to 1 (high). The wakeup state of all other bits is undefined.

General-Purpose Register File

This general-purpose (scratch pad) memory is directly accessible by the 8X300 and is used to store system variables such as track address, sector address and other necessary parameters. The sixteen 8-bit registers (110_8-127_8) provide sufficient on-chip memory to accommodate a minimum of two disk drives; the maximum number of drives that this non-dedicated memory file can support depends on several factors—system configuration, reliability requirements, economic constraints, and so on. Because of the on-chip file, all other system memory can be dedicated to the purpose of handling data to-and-from the disk media.

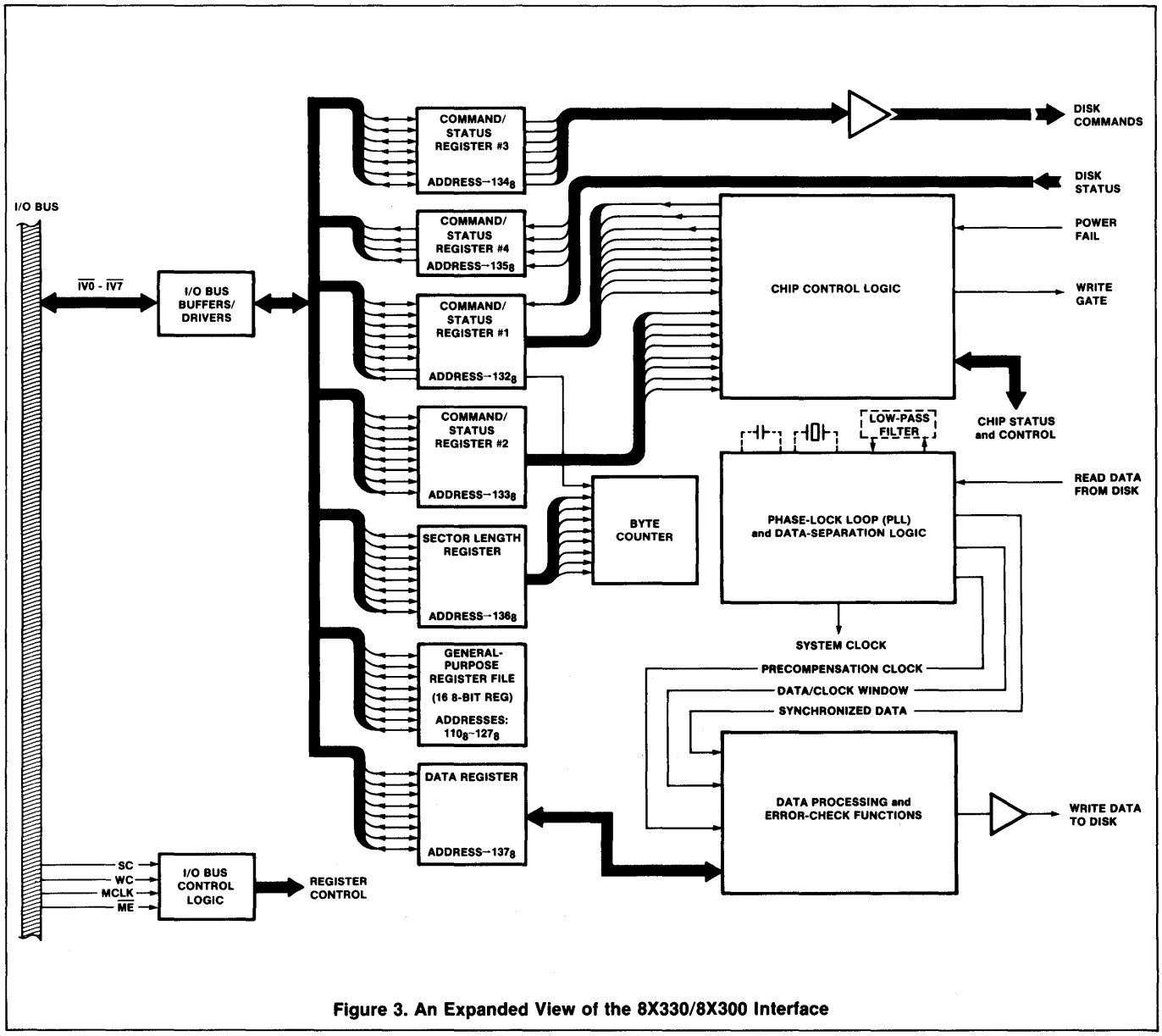


Figure 3. An Expanded View of the 8X330/8X300 Interface

Command/Status Register #1 (CSR 1/Address 132_8)

The disk status (read) or disk-command (write) contents of this register are interpreted as follows; unless otherwise indicated, all bits of CSR 1 are read/write from the I/O bus.

Bit 0 (Write Gate Enable)
Enables write gate output ($\overline{WG}$/pin 19) to disk drive(s)—the write gate ($\overline{WG}$) cannot be enabled unless the PF input pin (18) is high. When the WGE bit is set to 0, the $\overline{WG}$ ouput pin is low (enabled); when WGE is set to 1, the $\overline{WG}$ output is high (disabled). If the PF input goes low while the $\overline{WG}$ output is low, the $\overline{WG}$ output will go high and the Write Gate Enable bit is reset to 1.

Bit 1 (CRC Enable)
When set to 1, permits internal CRC register to compute remainders on the data stream in either read or write modes of operation. When set to 0, the CRC register becomes the source of data. A change in the CRC Enable bit does not become effective until the "next BYTRA flag appears" following the command bit change—refer to description of CSR 1/Bit 6.

Bit 2 (Data Register Control)
When set to 0, contents of data register consists of interleaved data-and-clock bits; starting from the MSB ($\overline{IVO}$) position, register contents are: Clock 1, Data 1, Clock 2, Data 2, Clock 3, Data 3, Clock 4, and Data 4. When writing an address mark, the appropriate data/clock pattern is loaded into the data register by the 8X300. Since each byte of data from the processor becomes an interleaved pattern (4-bits of data and 4-bits of clock) in the 8X330 data register, two bytes from the processor are required to write each full byte of address mark to the drive—eight bit cells with each cell containing a possible data and/or clock transition, or a total of 16 bit positions. When writing address marks, the normal on-chip clock insertion circuitry of the 8X330 is inhibited; thus, the user is free to define any clock/data pattern for the address mark.

When reading address marks, the data register is loaded with data and clock representing four bit cells from the disk media. The information in the data register can then be compared with the expected address mark by the 8X300 on a nibble-by-nibble basis. When the DRC bit is set to 1, the data register contains separated data (no clocks). A state change in this bit does not become effective until the "next BYTRA flag appears" following the state-change command.

Bit 3 (Sync Enable)
The Sync Enable bit allows the on-chip data separator to obtain bit and byte synchronization; this bit also controls initialization of the CRC Register. With the 8X330 in Read mode and with Bit 3 set to 0, bit synchronization occurs. The Preamble field is assumed to be all "zeroes" or all "ones" as determined by the Preamble Select bit (CSR 2/Bit 4).

When the proper number of preamble bytes, as determined by the disk-control program, have been found, the Sync Enable bit should be changed, under program direction, to a 1. This puts the 8X330 in the Address-Mark search mode. Accordingly, all bits of the CRC Register are preset to 1, the BYte TRAnsfer flag is inhibited, and the 8X330 examines the data stream for an Address Mark. The Address Mark is detected by observing the data and clock bits to find a change in the normal Preamble pattern. Byte synchronization is achieved by assuming that the change occurred in the bit cell determined by two Bit Select bits (CSR 2/Bits 2 and 3).

When the pattern change is found indicating the start of an Address Mark, the 8X330 starts CRC computation and synchronizes BYTRA to the byte boundaries. Note that the 8X330 presumes an Address Mark by finding a change in the preamble pattern; however, it is up to the 8X300 to read the Address Mark and to establish its validity or non-validity.

In write mode, setting the Sync Enable bit to 0 presets all bits of the CRC Register to 1. Setting the Sync Enable bit to 1 allows CRC computation to begin at the next byte boundary.

Bit 4 (Load Counter)
When set to 1, transfers 8-bits of data from Sector Length register and 1-bit (MSB) of data from Byte Counter (refer to next description) to 9-bit Byte Counter. Loading of the 9-bit Byte Counter is effective one bit-cell time after the Load Counter bit is set to 1. In both the read and write modes of operation, the Byte Counter is incremented by BYTRA. The Load Counter bit is self-clearing and always returns a 0 when read.

NOTE
The Load Counter bit must be set one or more instruction cycles *after* setting the Byte Counter MSB, that is, bits 4 and 5 of CSR 1 cannot be set during th same instruction cycle.

Bit 5 (Byte Counter MSB)
This bit is used to set and monitor the state of the ninth (MSB) bit in the Byte Counter; reading this bit always returns the current state of MSB in the Byte Counter. The MSB of the Byte Counter is set to the value of CSR 1/Bit 5 when the Load Counter bit (CSR 1/Bit 4) is asserted—refer to preceding description.

NOTE
The Byte Counter MSB must be set one or more instruction cycles *before* the Load Counter bits—bits 4 and 5 of CSR 1 cannot be set during the same instruction cycle.

Bit 6 (BYTRA)
During a disk read operation, the BYte TRAnsfer flag is automatically set to 0 when 8-bits of information are transferred from the Data Shift Register to the Data Register —see Figure 1. During a disk write operation, BYTRA is automatically set to 0 when 8-bits are transferred from the Data Register to the Data Shift Register. BYTRA (a read-

only bit) is reset to a 1 when the Data Register (address 137_8) is selected by the user's program. During read/write operations, the 1-to-0 transition of the BYTRA flag increments the Byte Counter to keep count of bytes read or bytes written. All read-only bits of the 8X330 are designed to remain stable during the monitor period; thus, to read a status change of BYTRA, Disk-Status bit, the Byte Counter MSB, or other read-only bit requires a two-instruction loop similar to:

```
TEST    SEL   CSR 1
        NZT   BYTRA,   TEST
```

Bit 7 (Disk Status 1)
Reflects state (0 or 1) of input DS1 (pin 17); this is a user-definable read-only bit.

NOTE
A high input on any one of the Disk Status lines of the 8X330 is read by the 8X300 program as a logical 1 and a low input on the status lines is read as a logical 0.

Command/Status Register #2 (CSR 2/Address 133_8)

The disk status (read) or disk-command (write) contents of this register are interpreted as follows:

Bit 0 (Precompensation Enable)
This command bit determines whether or not precompensation is applied to the data stream being written onto the disk. When set to 0, precompensation is inhibited. When set to 1 and with double-density encoding, write precompensation is applied to the following data/clock bit patterns:

Precomp Time	Data/Clock Pattern (in Data Shift Reg)			Bit Being Written	Bits Already Written to Disk		
2T (Late)	0	1	**0**	1	0	0	0
2T (Late)	0	1	**0**	1	0	0	1
2T (Early)	1	0	**0**	1	0	1	0
2T (Early)	0	0	**0**	1	0	1	0

where, $T = \frac{1}{\text{crystal frequency}}$ if bit 7 of CSR 2 ($1/2F$) = 1

$T = \frac{2}{\text{crystal frequency}}$ if bit 7 of CSR 2 ($1/2F$) = 0

Bit 1 (Read Mode)
When set to 0, the 8X330 reads data from the disk and transfers it to the Data Register; when set to 1, data from the Data Register is transferred to the disk, provided the Write Gate Enable bit (CSR1/Bit 0) is set to 1. With WGE set to 0 and the Read Mode bit set to 1, the current-controlled oscillator is forced to lock onto the crystal oscillator; this technique is used during a data-read operation to ensure rapid acquisition of the disk data.

Bits 2,3 (Bit Selects 1 and 0)
Together with the Sync Enable (CSR 1/Bit 3), these two bits allow the user to establish byte boundaries for the data stream; this is done in the following way. After bit synchronization is established, and the preamble pattern is verified, the 8X330 looks for a change in the normal preamble pattern. As shown in the following truth table, Bit Select 1 (Bit 2) and Bit Select 0 (Bit 3) identifies the bit cell within the first nibble of the first Address-Mark byte in which the first deviation from the normal preamble is expected. BYTRA is always referenced to bit cell 0.

BS 0	BS 1	Bit Cell
0	0	0
0	1	1
1	0	2
1	1	3

Bit 4 (Preamble Select)
This bit is used only for bit sychronization—refer to CSR 1/Bit 3. With Bit 1 of CSR 2 set to 0 (Read Mode) and the Preamble Select bit set to 0, the preamble field is assumed to be all zeroes; with the Preamble Select it set to 1, the preamble field is assumed to be all ones. In either case, preamble validity is determined by the 8X300.

Bits 5,6 (E1 and E2)
Together, E1 and E2 select the encoding scheme used to write data on the disk—refer to truth table that follows.

E1	E2	Encoding Scheme
1	X	FM
0	0	MFM
0	1	M²FM

x = don't care

Bit 7 (1/2F)
This bit allows the data transfer rate to be changed without modification of the frequency-selective components in the data-separation logic; thus, differences in data transfer rates between standard-and-mini floppies can be accommodated via software—no component or other hardware changes. Assuming an 8 MHz crystal and with the 1/2F bit set to 1, the data transfer rate is 250K-bits per second in the single-density (FM) mode and 500K-bits per second in the double-density (MFM/M²FM) mode. When set to 0, the transfer rates are halved—125K-bits and 250K-bits, respectively. When using frequencies other than 8 MHz, the data transfer rate is determined as follows:

Bit 7 ($1/2F$)	Single-Density (FM)	Double-Density (MFM/M²FM)
0	$\frac{\text{xtal freq}}{64}$	$\frac{\text{xtal freq}}{32}$
1	$\frac{\text{xtal freq}}{32}$	$\frac{\text{xtal freq}}{16}$

Command/Status Register #3 (CSR 3/Address 134_8)

This register contains seven bits (Bit 0 through Bit 6) which determines the state of the disk-command outputs; writing to Bit 7 has no effect and reading Bit 7 always returns a zero. When a logical "1" is specified by the 8X300 program for a given disk-command line, a high will appear at the output of the 8X330 for that particular command line. Each bit and the output pin it controls are summarized below.

Bit (CSR 3)	Control Function	Pkg Pin No.
0	DC1 Output	12
1	DC2 Output	11
2	DC3 Output	10
3	DC4 Output	9
4	DC5 Output	8
5	DC6 Output	7
6	DC7 Output	6

Command/Status Register #4 (CSR 4/Address 135_8)

This register contains four bits (Bit 0 through Bit 3) which reflect the state of the disk-status inputs to the 8X330; reading all other bits (4 through 7) always returns a zero. These read-only bits and the reflected status they represent are as follows; the information specified by notation for Bit 7/CSR 1 is applicable to these input lines.

Bit (CSR 4)	Control Function	Pkg Pin No.
0	DS2 Input	16
1	DS3 Input	15
2	DS4 Input	14
3	DS5 Input	13

Sector Length Register—Address 136_8

This register contains the load value for the lower eight (LSBs) bits of the Byte Counter. Data is transferred from the Sector Length Register to the Byte Counter under control of Load Counter Bit in CSR 1. When the contents of this register are transferred to another location via a read or write commands, the original holding of data is not lost; thus, if the same data is to be used more than once, a repetitive read or write can be implemented without reloading the register.

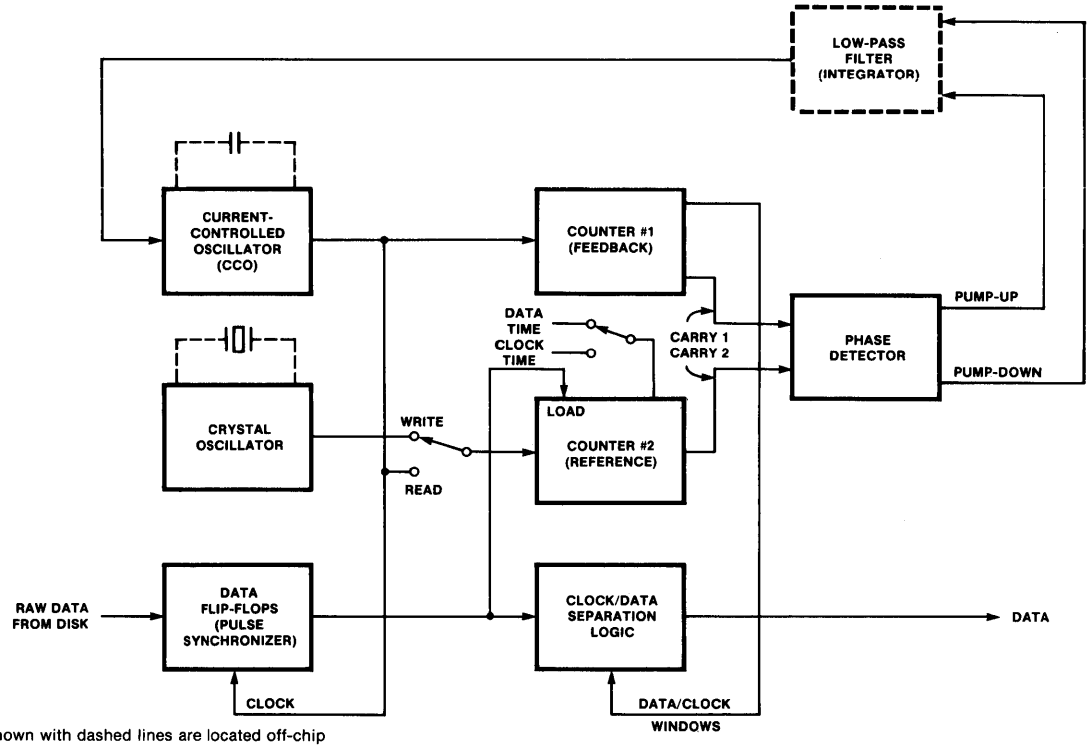
Data Register—Address 137_8

Together with the Data Shift Register, the Data Register is used for bidirectional transfer of data between the 8X330 and the I/O bus. All transfers to-and-from this register are made in conjunction with Bit 6 (BYTRA—Byte Transfer Flag) of CSR 1. When the Data Register Control bit (CSR 1/Bit 2) is set to 0, the content of this register is interleaved with four bits of data and four bits of clock. When data is transferred from the Data Register to the Data Shift Register, the original content of the Data Register is not lost.

Phase Lock Loop (PLL) and Data Separation Logic

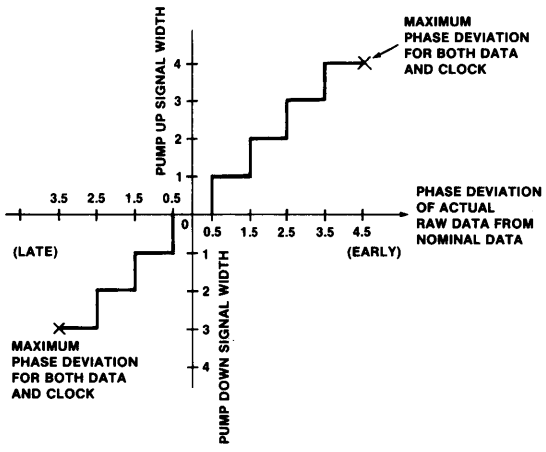
An expanded view of the phase-lock loop and the data-separation logic is shown in Figure 4. Basically, the PLL consists of two counters, a phase detector, and a feedback loop containing a low-pass filter (off-chip) that controls a phase-locked oscillator (CCO). In simplified form, the data-separation logic consists of data flip-flops (pulse synchronizer) and other circuits required to separate data and clock transitions. In the read mode, the output of the phase-locked oscillator (CCO) is applied to the clock inputs of counter #1, counter #2, and the pulse synchronization circuits. Essentially, the frequencies of the two counters are identical (phase relationships may or may not be identical); to maintain proper frequencies and to continuously correct for any phase deviations, the following actions occur.

Preset values which represent, respectively, nominal mid-points of the clock and data windows are present at counter #2 and, when an output appears at the pulse synchronizer, these preset values are entered. The count sequence for both counters is from "0 to F"; hence, the phase difference between Carry 1 (counter #1) and Carry 2 (counter #2) actually corresponds to any phase deviation between the CCO and the synchronized data from the disk. The phase detector measures the phase difference between the two carry inputs and produces a series of quantized pulses whose widths are proportional to the phase error at the end of each counting cycle. After integration by the low-pass filter, a current proportional to the phase error is applied to the current-controlled oscillator. Accordingly, the CCO is driven in a direction (pump-up or pump-down) to correct any phase difference between the synchronized disk data and the feedback-controlled clock. Phase detector characteristics for both single-and-double density formats are shown in Figures 5 and 6.



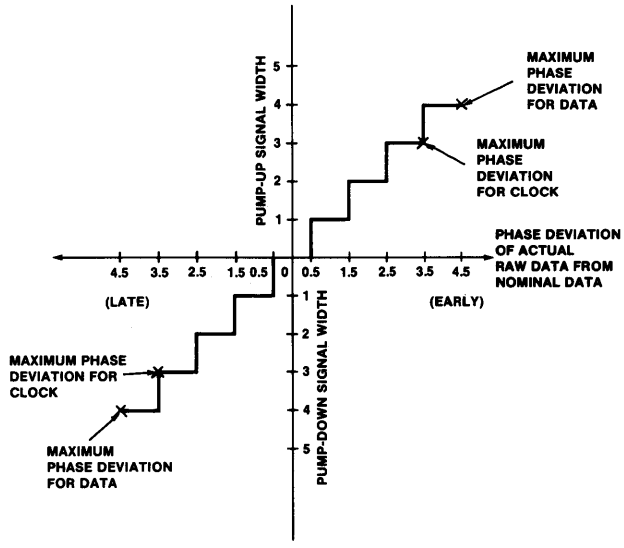
NOTE
Components shown with dashed lines are located off-chip

Figure 4. Simplified Block of Phase-Lock Loop



NOTE
All horizontal/vertical units are specified in CCO clock periods; for Standard Floppy FM Format, one CCO clock period equals 250 nanoseconds

Figure 5. Phase Detector Characteristic for Single-Density (FM) Format



NOTE
All horizontal/vertical units are specified in CCO clock periods; for standard floppy double density (MFM/M²FM), one CCO clock period equals 125 nanoseconds

Figure 6. Phase Detector Characteristic for Double-Density (MFM/M²FM) Format

Data Processing and Error-Check Functions

These functions of the 8X330 are summarized in Figures 7 and 8. The read/write operations are software-controlled by previously-described bits of command/status registers CSR1 and CSR2. For the sake of simplicity, control lines and much of the control logic associated with the data processing and error-check functions are omitted in the read/write diagrams.

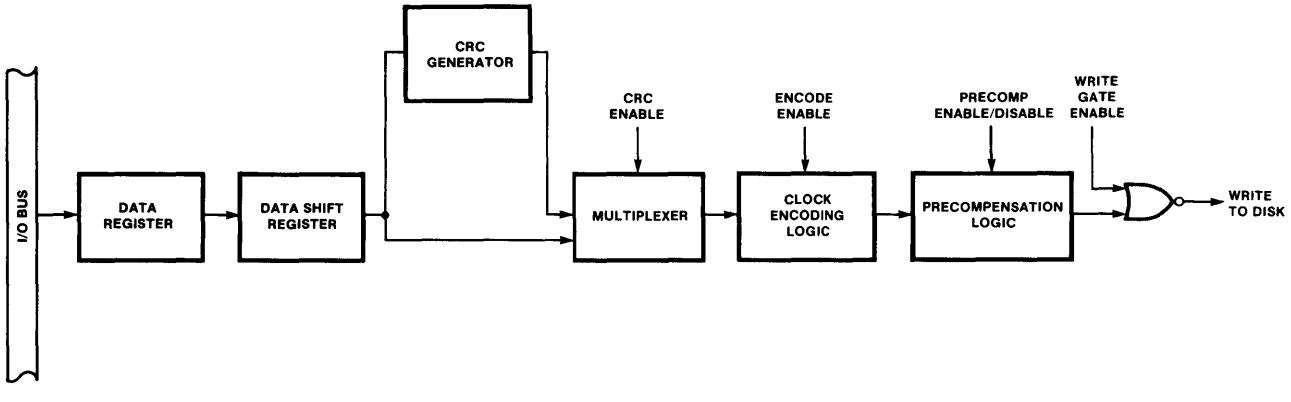


Figure 7. Simplified Block of Data Processing and Error Check Functions—Write Mode

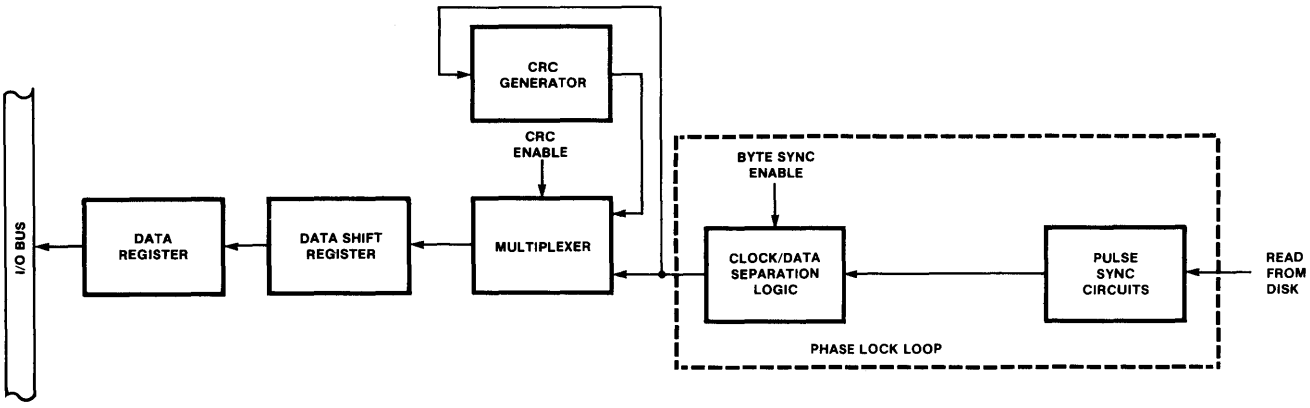


Figure 8. Simplified Block of Data Processing and Error Check Functions—Read Mode

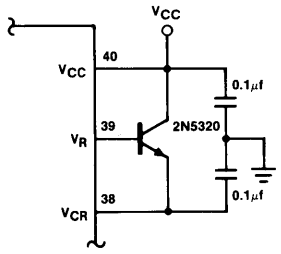
DC CHARACTERISTICS V_{CC} = 5V (±5%), T_A = 0°C to 70°C

PARAMETER		TEST CONDITIONS	LIMITS			UNITS	COMMENTS
			Min	Typ	Max		
V_{IH}	High level input voltage		2.0		V_{CC}	V	For all inputs except X1, X2, C1, C2, CCO, and V_{CR}
V_{IL}	Low level input voltage		−1.0		.8	V	
V_{CC}	Supply voltage		4.75	5.0	5.25	V	5V (±5%)
V_{CR}	Regulator voltage	V_{CC} = 5V		3.1		V	From series-pass transistor
V_{CL}	Input clamp voltage	V_{CC} = Min I_{IN} = −5mA	−1.0			V	Inputs X1, X2, C1, C2, and CC0 do not have internal clamp diodes.
V_{OH}	High level output voltage	V_{CC} = Min; I_{OH} = −0.4mA	2.7			V	DC1 through DC7 (Pins 6-12) & $\overline{DW}$ (Pin 4)
		V_{CC} = Min; I_{OH} = −3mA	2.4			V	$\overline{IV0}$-$\overline{IV7}$ (Pins 25-32)
V_{OL}	Low level output voltage	V_{CC} = Min; I_{OL} = 8mA			0.5	V	DC1 through DC7 (Pins 6-12); $\overline{PUP}$, $\overline{PDN}$ (Pins 33, 34); $\overline{DW}$ (Pin 4)
		V_{CC} = Min; I_{OL} = 16mA			0.55	V	$\overline{IV0}$-$\overline{IV7}$ (Pins 25-32)
		V_{CC} = Min; I_{OL} = 40mA			0.55	V	$\overline{WG}$ (Pin 19)
I_{CEX}	Open-collector leakage current with output set to 1.	V_{CC} = Min; V_{OUT} = V_{CC}			100	μA	$\overline{WG}$ (Pin 19); $\overline{PUP}$ (Pin 34); $\overline{PDN}$ (Pin 33)
I_{IH}	High level input current	V_{CC} = Max; V_{IN} = 2.7V			20	μA	DS1-DS5 (Pins 13-17); $\overline{PF}$ (Pin 18); $\overline{DR}$ (Pin 5)
					40	μA	$\overline{ME}$ (Pin 21); MCLK (Pin 22); SC (Pin 23); WC (Pin 24)
		V_{CC} = Max; V_{IN} = 5.25V; CCO (Pin 35) input current = 0mA	1		4	mA	With C1 (Pin 36) under test, C2 (Pin 37) is open and, vice-versa.
		V_{CC} = Max; V_{IN} = 5.25V CCO (Pin 35) input current = 1mA	0		2	mA	
		V_{CC} = Max; V_{IN} = 0.6V	1		4	mA	With X1 (Pin 2) under test, X2 (Pin 3) is open and, vice-versa.
		V_{CC} = Max; V_{IN} = 4.5V			50	μA	$\overline{IV0}$ - $\overline{IV7}$ (pins 25-32)
V_{CCO}	Input voltage for current-controlled oscillator	V_{CC} = 5V; T_A = 25°C CCO input current (Pin 35) = 300 μA		750		mV	

DC CHARACTERISTICS (Cont'd) V_{CC} = 5V (±5%), T_A = 0°C to 70°C

PARAMETER		TEST CONDITIONS	LIMITS			UNITS	COMMENTS
			Min	Typ	Max		
I_{IL}	Low level input current	V_{CC} = Max; V_{IN} = 0.4V			−400	µA	DS1-DS5 (Pins 13-17); $\overline{PF}$ (Pin 18); $\overline{DR}$ (Pin 5)
					−800	µA	$\overline{ME}$ (Pin 21); MCLK (pin 22); SC (Pin 23); WC (Pin 24)
			−1		−4	mA	X1 (Pin 2), X2 (Pin 3), C1 (Pin 36), and C2 (Pin 37)—with X1 under test, X2 is open and, vice-versa; the same relationship exists for C1 and C2.
		V_{CC} = Max V_{IN} = 0.5V			−550	µA	$\overline{IV0}$-$\overline{IV7}$ (Pins 25-32)
I_{OS}	Output short-circuit current	V_{CC} = Max; Output = "1"; V_{OUT} = "0".	−15		−100	mA	DC1-DC7 (Pins 6-12) & $\overline{DW}$ (Pin 4)
		(NOTE At any time, no more than one output should be connected to ground)	−30		−140	mA	$\overline{IV0}$-$\overline{IV7}$ (Pins 25-32)
I_{CC}	(Pin 40)	V_{CC} = Max			200	mA	
I_{CR}		V_{CC} = Max			250	mA	
I_{REG}	(Pin 39)	V_{CC} = 5V; V_{CR} = 0V & V_R = 2V	−16		−24	mA	

NOTES
1. Operating temperature ranges are guaranteed after thermal equilibrium has been reached.
2. All voltages measured with respect to ground terminal.
3. Unless otherwise specified, each test requires that V_{CR} be supplied through a series-pass transistor as shown in the accompanying drawing.



AC CHARACTERISTICS V_{CC} = 5V (±5%), T_A = 0°C to 70°C

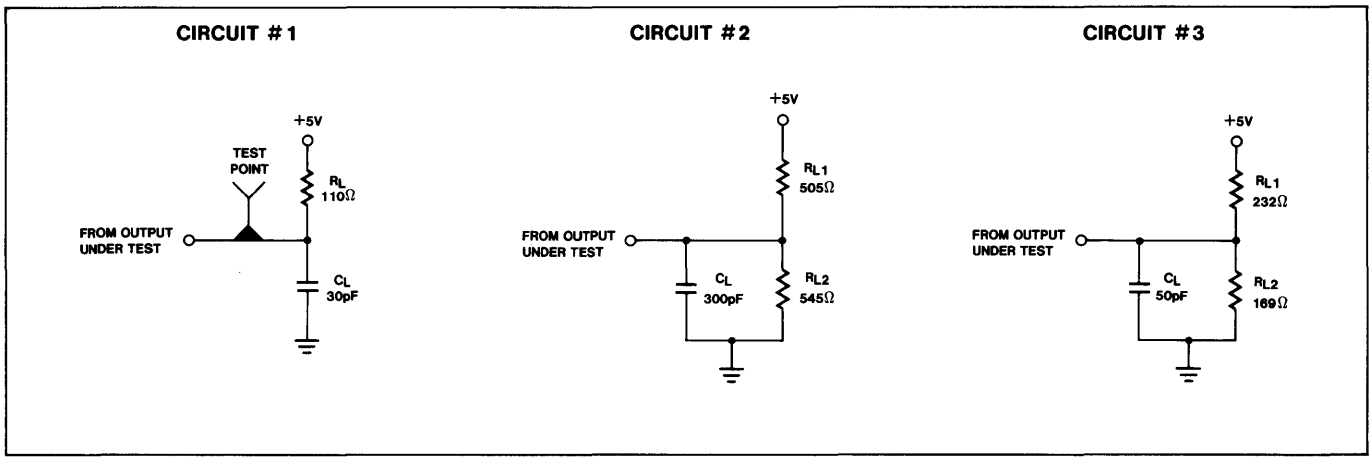
MNEMONIC	REFERENCE	INPUT	OUTPUT	Min	Typ	Max	COMMENTS
t_{PD}		$\overline{PF}$ ↓	$\overline{WG}$ ↑			30ns	Refer to Note 3 and Test Loading Circuit #1.
t_{PD}		MCLK ↑	$\overline{WG}$ ↑			70ns	
t_{PD}		MCLK ↑	$\overline{WG}$ ↓			70ns	
t_{PD}		MCLK ↑	DC1-7 ↑			70ns	Refer to Note 3 and Test Loading Circuit #2.
t_{PD}		MCLK ↑	DC1-7 ↓			70ns	
t_{pw}		$\overline{PF}$ ↓		50ns			
t_{pw}		$\overline{DR}$ ↓		50ns			
t_{pw}			$\overline{DW}$ ↓				Note 1
t_{SETUP}	SC ↓	Input on DS1-5		55ns			Note 2
t_{SETUP}	WC ↓	Input on DS1-5		55ns			Note 2
t_{SETUP}	$\overline{ME}$ ↓	Input on DS1-5		55ns			Note 2
t_{HOLD}	SC ↓	Input on DS1-5		0ns			Note 2
t_{HOLD}	WC ↓	Input on DS1-5		0ns			Note 2
t_{HOLD}	$\overline{ME}$ ↓	Input on DS1-5		0ns			Note 2
t_{OE}—$\overline{ME}$, SC & WC		$\overline{ME}$ ↓ SC ↓ WC ↓	I/O bus			25ηsec	Refer to Test Loading Circuit #3.
t_{OD}—$\overline{ME}$, SC & WC		$\overline{ME}$ ↑ SC ↑ WC ↑	I/O bus (three-state)			30ηsec	
t_W (MCLK pulse width)		MCLK ↓		45ns			
t_{SD} (data setup time)	MCLK ↓	I/O bus		50ns			
t_{SD} ($\overline{ME}$ setup time)	MCLK ↓	$\overline{ME}$		45ns			
t_{SD} (SC setup time)	MCLK ↓	SC		45ns			
t_{SD} (WC setup time)	MCLK ↓	WC		45ns			
t_{HD} (data hold time)	MCLK ↓	I/O bus		0ns			

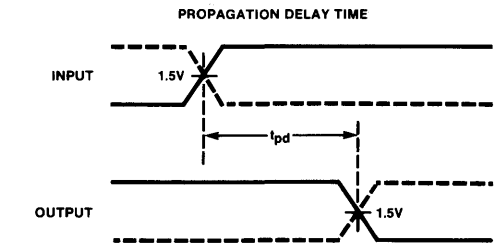
AC CHARACTERISTICS (Cont'd) V_{CC} = 5V (±5%), T_A = 0°C to 70°C

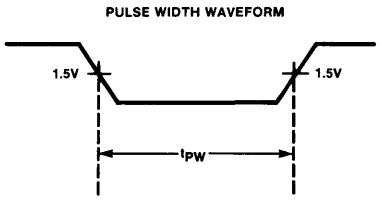
MNEMONIC	REFERENCE	INPUT	OUTPUT	Min	Typ	Max	COMMENTS
t_{HD} ($\overline{ME}$ hold time)	MCLK ↓	$\overline{ME}$		0ns			
t_{HD} (SC hold time)	MCLK ↓	SC		0ns			
t_{HD} (WC hold time)	MCLK ↓	WC		0ns			

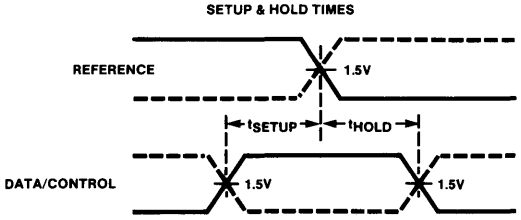
NOTES
1. Write pulse width = $2/F_{XTAL}$, that is, for 8MHz crystal, t_{pw} = 250ηsec (typical)
2. Changes on DS1-5 are not stored in read mode ($\overline{ME}$ = 0, SC = 0, and WC = 0)
3. During the period when MCLK is high, measurement is made with $\overline{ME}$ = Low, SC = Low, and WC = High.

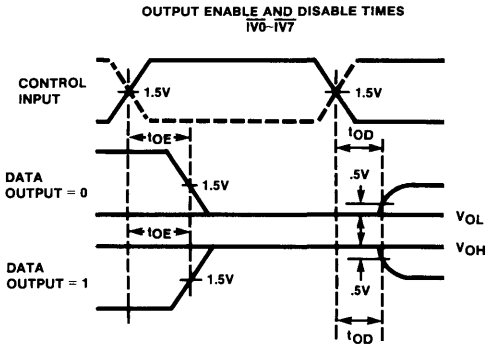
TEST LOADING CIRCUIT











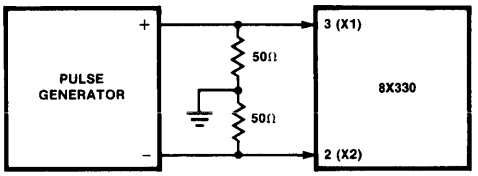
CLOCK REQUIREMENTS

Crystal Oscillator. The on-chip crystal oscillator circuit is designed for operation using an external series-resonant quartz crystal; alternately the crystal oscillator can be driven with complementary outputs of a pulse generator or interfaced to a master clock source via TTL logic—see accompanying circuits. When a crystal is used, the on-chip oscillator operates at the resonant frequency (f.) of the crystal; the crystal connects to the 8X330 via pins 3 (X1) and 2 (X2). The lead lengths of the crystal should be approximately equal and as short as possible; also, avoid close proximity to all potential noise sources. The crystal should be hermetically sealed (HC type can) and have the following electrical characteristics:

Type: Fundamental mode, series resonant
Impedance at Fundamental: 35-ohms maximum
Impedance at Harmonics and Spurs: 50-ohms minimum

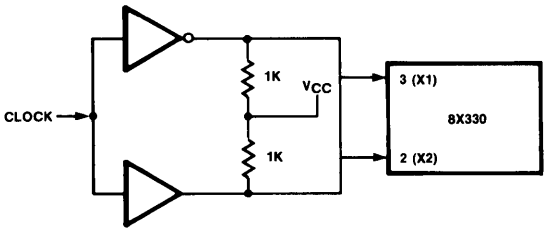
When the crystal oscillator is externally-driven, typical waveforms are as follows:

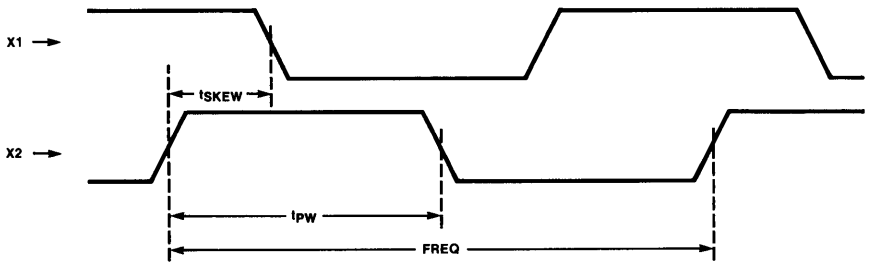
CLOCKING XTAL OSC WITH PULSE GEN



PULSE GEN CHAR:
Z_{OUT} = 50Ω
V_{OUT} = 0–2V
RISETIME ≤ 10ηSEC
SKEW ≤ 10ηSEC
COMPLEMENTARY OUTPUTS

CLOCKING XTAL OSC WITH OPEN-COLLECTOR TTL

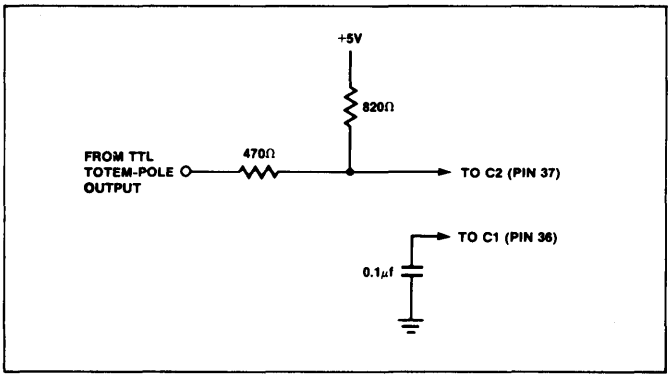




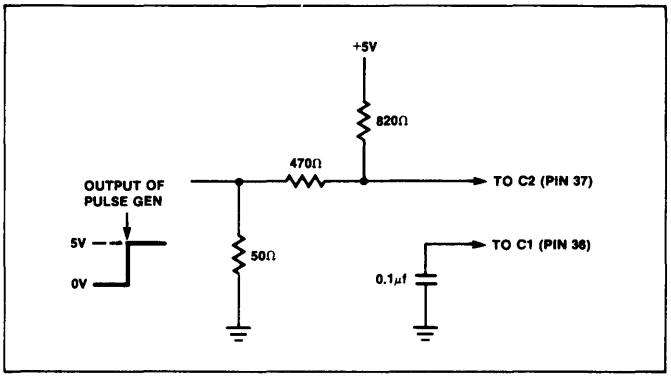
FREQUENCY = 8 MHz (TYP)/16 MHz (MAX)
t_{PW} = 50% (TYP)
t_{SKEW} ≤ 10ηSEC

Current-Controlled Oscillator (CCO).
A non-polarized ceramic or mica capacitor is recommended for the current-controlled oscillator. The capacitor connects to the 8X330 via pins 37 (C2) and 36 (C1); lead lengths of the capacitor should be approximately the same and as short as possible. When the input current to the CCO is near zero (maximum frequency), the capacitor value should be chosen so that the high-limit rest frequency of the oscillator does not exceed 24 MHz. If the rest frequency is higher than 24 MHz, synchronization of the CCO with the crystal oscillator just prior to the read operation, may be impeded. The curves in Figure 9 (current-versus-frequency) and Figure 8 (capacitance-versus-frequency) show how these design parameters affect operation of the CCO over a temperature range of 0° C to 70° C. A suitable test circuit for verification/validation of the current-controlled oscillator is also shown in Figure 10. Like the crystal oscillator, the CCO can be driven with the TTL output of a pulse generator or interfaced to a master clock via TTL logic—see accompanying diagrams.

CLOCKING WITH OPEN-COLLECTOR TTL



CLOCKING WITH PULSE GENERATOR



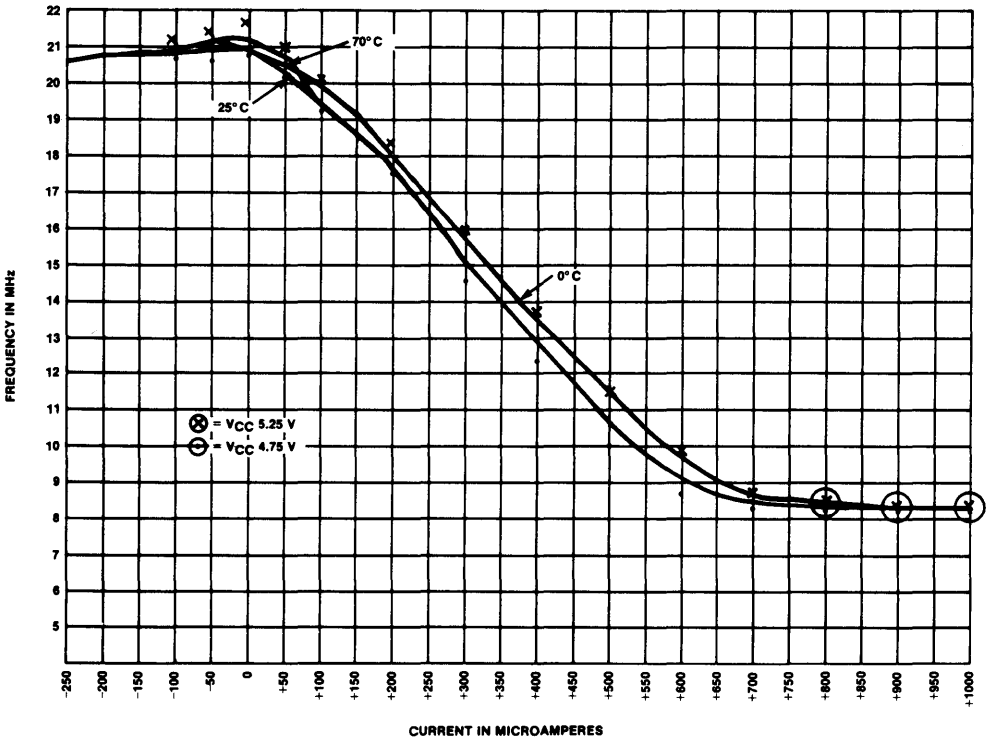


Figure 9. Current-versus-Frequency with: V_{CC} = 5V and Capacitance = 25 Picofarads

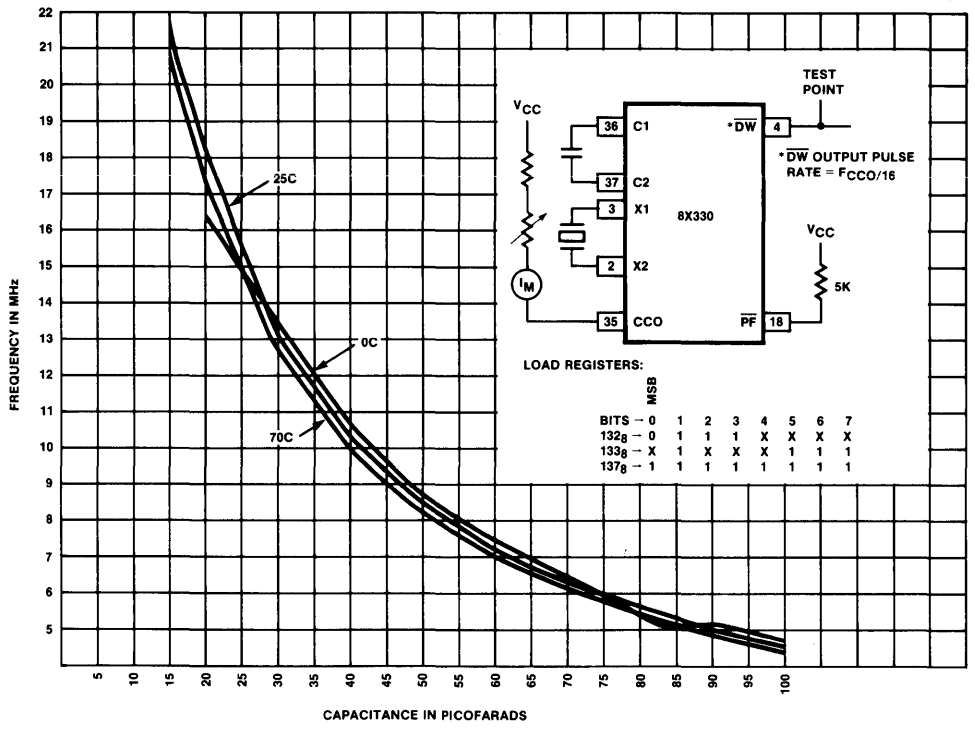
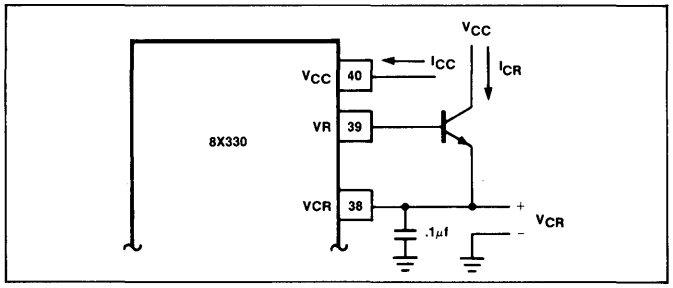


Figure 10. Capacitance-versus-Frequency with: V_{CC} = 5V, VCR = 2.5V, and I = 300μA

VOLTAGE REGULATOR

All internal logic of the 8X330 is powered by an on-chip voltage regulator that requires an external series-pass transistor. Electrical specifications for the off-chip power transistor and a typical hook-up are shown in accompanying diagrams. To minimize lead inductance, the transistor should be as close as possible to the 8X330 package and the emitter should be ac-grounded via a 0.1-microfarad capacitor.

TYPICAL HOOK-UP



ELECTRICAL SPECIFICATIONS

*PARAMETER	CONDITIONS	LIMITS
h_{fe}	V_{CE} = 2V	>50
V_{BEON}	V_{CE} = 5V/I_C = 500mA	<1V
V_{CESAT}	I_C = 500 mA/I_B = 50 mA	<0.5V
BV_{CEO}		>8V
f_t		>30 MHz

*Medium power NPN silicon (0°<T_A<70°C) recommended parts: 2N5320, 2N5337

ORDERING INFORMATION

Order number: N8X330
Packaging information: Refer to Signetics price list
Supply voltage: 5V (±5%)
Operating temperature range: 0°C to +70°C

DESCRIPTION

The 8X350 bipolar RAM is designed principally as a working storage element in an 8X300 based system. Internal circuitry is provided for direct use in 8X300 applications. When used with the 8X300, the RAM address and data buses are tied together and connected to the IV bus of the system.

The data inputs and outputs share a common I/O bus with 3-state outputs.

The 8X350 is available in commercial and military temperature ranges. For the commercial temperature range (0°C to +75°C) specify N8X350-F, and for the military temperature range (−55°C to +125°C) specify S8X350-F.

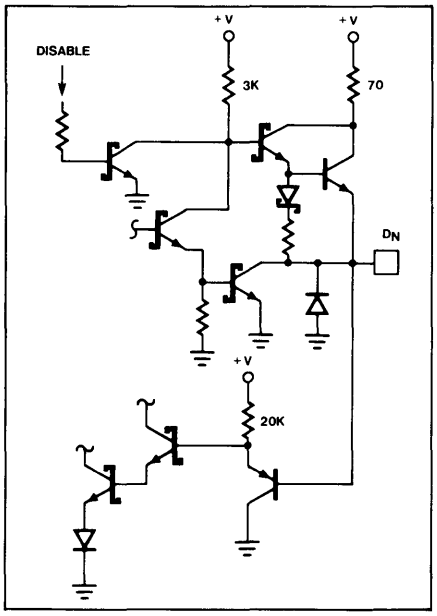
FEATURES

- **On-chip address latches**
- **3-state outputs**
- **Schottky clamped TTL**
- **Internal control logic for 8X300 system**
- **Directly interfaces with the 8X300 bipolar microprocessor with no external logic**
- **May be used on left or right bank**

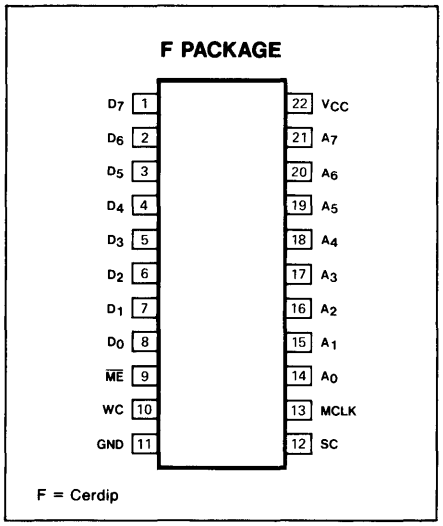
APPLICATIONS

- **8X300 working storage**

TYPICAL I/O STRUCTURE



PIN CONFIGURATION



F = Cerdip

ABSOLUTE MAXIMUM RATINGS

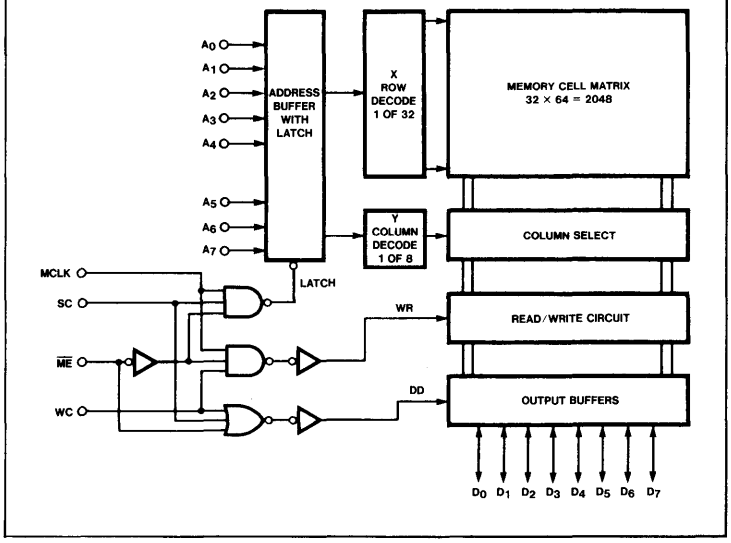
	PARAMETER[1]	RATING	UNIT
V_{CC}	Supply voltage	+7	Vdc
V_{IN}	Input voltage	+5.5	Vdc
	Output voltage		Vdc
V_{OH}	High	+5.5	
V_O	Off-state	+5.5	
	Temperature range		°C
T_A	Operating		
	N8X350	0 to +75	
	S8X350	−55 to +125	
T_{STG}	Storage	−65 to +150	

TRUTH TABLE

Note X = Don't care

MODE	$\overline{ME}$	SC	WC	MCLK	BUSSED DATA/ADDRESS LINES
Hold address Disable data out	1	X	X	X	High Z
Input new address	0	1	0	1	Address
Hold address					High Z
Disable data out	0	1	0	0	High Z
Hold address Write data	0	0	1	1	Data in
Hold address Disable data out	0	0	1	0	High Z
Hold address Read data	0	0	0	X	Data out
Undefined state[12]	0	1	1	1	—
Hold address[12] Disable data out	0	1	1	0	High Z

BLOCK DIAGRAM



DC ELECTRICAL CHARACTERISTICS[2]

N8X350: $0°C \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$

S8X350: $-55°C \leq T_A \leq +125°C$, $4.75V \leq V_{CC} \leq 5.25V$

PARAMETER		TEST CONDITIONS	N8X350			S8X350			UNIT
			Min	Typ	Max	Min	Typ	Max	
	Input voltage								V
V_{IL}	Low[1]	V_{CC} = Min			.85			.80	
V_{IH}	High[1]	V_{CC} = Max	2.0			2.0			
V_{IC}	Clamp[1,3]	V_{CC} = Min, I_{IN} = −12mA			−1.2			−1.2	
	Output voltage	V_{CC} = Min							V
V_{OL}	Low[1,4]	I_{OL} = 9.6mA			0.5			.5	
V_{OH}	High[1,5]	I_{OH} = −2mA	2.4			2.4			
	Input current								μA
I_{IL}	Low	V_{IN} = 0.45V			−100			−150	
I_{IH}	High	V_{IN} = 5.5V			25			50	
	Output current								μA
$I_{O(OFF)}$	High Z state	$\overline{ME}$ = High, V_{OUT} = 5.5 V			40			60	μA
		$\overline{ME}$ = High, V_{OUT} = 0.5 V			−100			−100	
I_{OS}	Short circuit[3,6]	SC = WC, $\overline{ME}$ = Low, V_{OUT} = 0V, Stored High	−20		−70	−15		−85	mA
I_{CC}	V_{CC} supply current[7]	V_{CC} = Max			185			185	mA
	Capacitance	$\overline{ME}$ = High, V_{CC} = 5.0V							pF
C_{IN}	Input	V_{IN} = 2.0V		5			5		
C_{OUT}	Output	V_{OUT} = 2.0V		8			8		

AC ELECTRICAL CHARACTERISTICS [2,9]

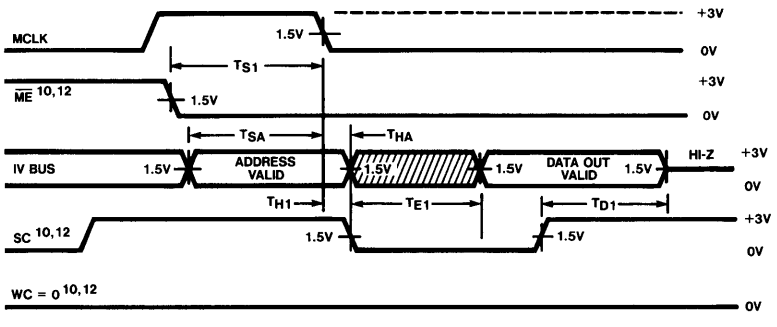
N8X350: $0°C \leq T_A \leq +75°C$, $4.75V \leq V_{CC} \leq 5.25V$

S8X350: $-55°C \leq T_A \leq +125°C$, $4.75V \leq V_{CC} \leq 5.25V$

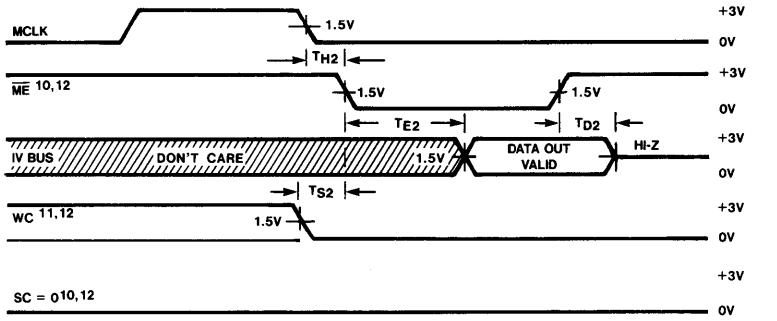
PARAMETER		TO	FROM	N8X350			S8X350			UNIT
				Min	Typ	Max	Min	Typ	Max	
	Enable time									ns
T_{E1}	Output	Data out	SC−			35			40	
T_{E2}	Output	Data out	$\overline{ME}$−			35			40	
	Disable time									ns
T_{D1}	Output	Data out	SC+			35			40	
T_{D2}	Output	Data out	$\overline{ME}$+			35			40	
	Pulse width									ns
T_W	Master clock[8]			40			50			
	Setup and hold time									ns
T_{SA}	Setup time	MCLK−	Address	30			40			
T_{HA}	Hold time	Address	MCLK−	5			10			
T_{SD}	Setup time	MCLK−	Data in	35			45			
T_{HD}	Hold time	Data in	MCLK−	5			10			
T_{S3}	Setup time	MCLK−	$\overline{ME}$−	40			50			
T_{H3}	Hold time	$\overline{ME}$+	MCLK−	5			5			
T_{S1}	Setup time	MCLK−	$\overline{ME}$−	30			40			
T_{H2}	Hold time	$\overline{ME}$−	MCLK−	5			5			
T_{S2}	Setup time	$\overline{ME}$−	SC−,WC−	0			5			
T_{H1}	Hold time	SC−	MCLK−	5			5			
T_{H4}	Hold time	WC−	MCLK−	5			5			

TIMING DIAGRAMS

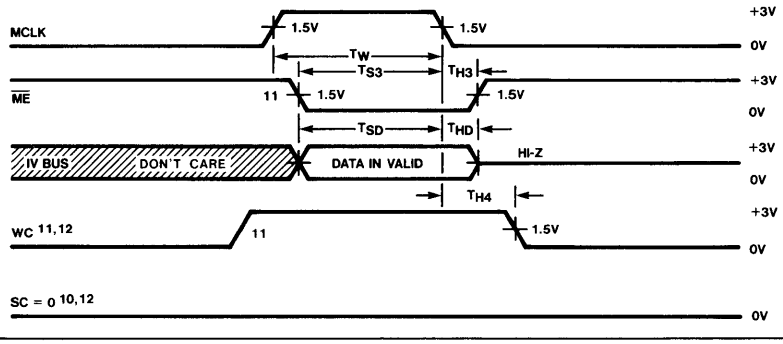
TRANSPARENT ADDRESS READ CYCLE



LATCHED ADDRESS READ CYCLE



LATCHED ADDRESS WRITE CYCLE



NOTES

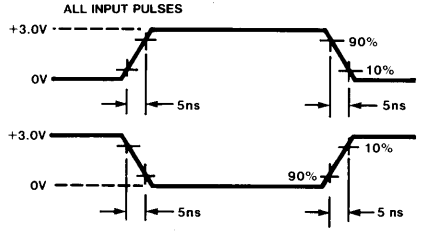
1. All voltage values are with respect to network ground terminal.
2. The operating ambient temperature ranges are guaranteed with transverse air flow exceeding 400 linear feet per minute and a 2-minute warm-up.
   Typical thermal resistance values of the package at maximum temperature are:
   Θ_{JA} junction to ambient at 400fpm air flow - 50°C/watt
   Θ_{JA} junction to ambient - still air - 90°C/watt
   Θ_{JA} junction to case - 20°C/watt
3. Test each pin one at a time.
4. Measured with a logic low stored Output sink current is supplied through a resistor to V_{CC}.
5. Measured with a logic high stored.
6. Duration of the short circuit should not exceed 1 second.
7. I_{CC} is measured with the write enable and memory enable inputs grounded, all other inputs at 4.5V and the output open.
8. Minimum required to guarantee a Write into the slowest bit.
9. Applied to the 8X300 based system with the data and address pins tied to the IV Bus.
10. SC + $\overline{ME}$ = 1 to avoid bus conflict.
11. WC + $\overline{ME}$ = 1 to avoid bus conflict.
12. The SC and WC outputs from the 8X300 are never at 1 simultaneously.

TIMING DEFINITIONS

T_{S1}	Required delay between beginning of Master Enable low and falling edge of Master Clock.
T_{SA}	Required delay between beginning of valid address and falling edge of Master Clock.
T_{HA}	Required delay between falling edge of Master Clock and end of valid Address.
T_{H1}	Required delay between falling edge of Master Clock and when Select Command becomes low.
T_{E1}	Delay between beginning of Select Command low and beginning of valid data output on the IV Bus.
T_{D1}	Delay between when select Command becomes high and end of valid data output on the IV Bus.
T_{H2}	Required delay between falling edge of Master Clock and when Master Enable becomes low.
T_{E2}	Delay between when Master Enable becomes low and beginning of valid data output on the IV Bus.
T_{D2}	Delay between when Master Enable becomes high and end of valid data output on the IV Bus.
T_{S2}	Required delay between when Select Command or Write Command becomes low and when Master Enable becomes low.
T_W	Minimum width of the Master Clock pulse.
T_{S3}	Required delay between when Master Enable becomes low and falling edge of Master Clock.
T_{H3}	Required delay between falling edge of Master Clock and when Master Enable becomes high.
T_{SD}	Required delay between beginning of valid data input on the IV Bus and falling edge of Master Clock.
T_{HD}	Required delay between falling edge of Master Clock and end of valid data input on the IV Bus.
T_{H4}	Required delay between falling edge of Master Clock and when Write Command becomes low.

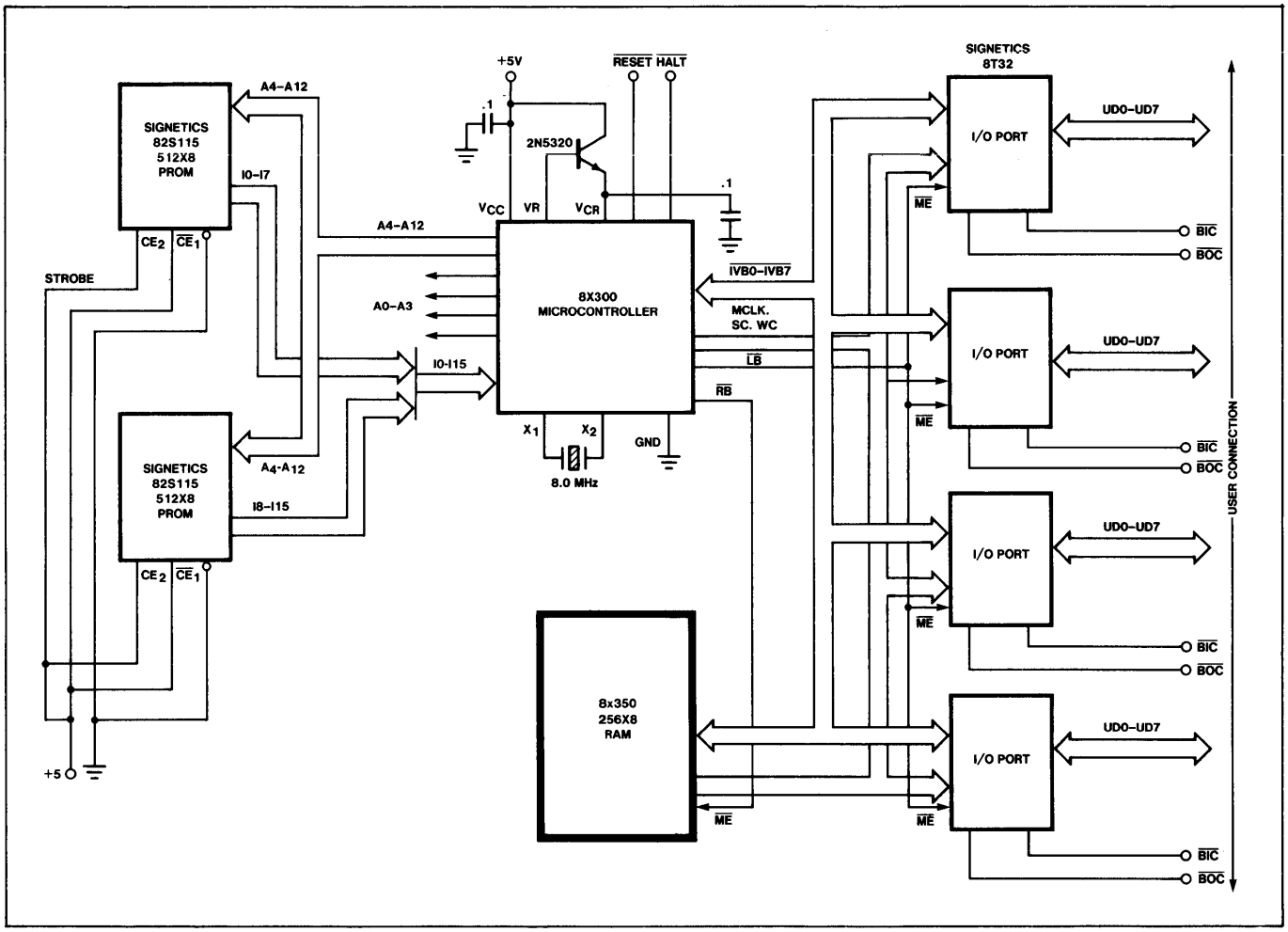
VOLTAGE WAVEFORM

INPUT PULSES



Measurements: All circuit delays are measured at the +1.5V level of inputs and output.

TYPICAL 8X350 APPLICATION



PRODUCT DESCRIPTION

Both the 8T31 and 8X31 are 8-bit bidirectional data registers designed to function as Input/Output interface elements in microprocessor systems. The two parts are functionally identical; however, the 8X31 is housed in a 0.4-inch Slim-Line package which makes it ideally suited for applications where board space is limited.

Each part contains eight clocked data latches that are accessible from either a *microprocessor* port or a *user* port. Separate I/O control is provided for each port. The two ports operate independently, except that when both are attempting to input data into the data latches, the User port (UD0-UD7) has priority. The master enable ($\overline{ME}$) signal enables or disables the microprocessor bus regardless of the state of the other inputs but has no effect on the user bus.

A unique feature of these parts is their ability to start up in a predetermined state. If the clock is maintained at a level of less than 0.8 volts until the power supply reaches 3.5 volts, all bits of the user port will wakeup at a "logic 1" level and those of the microprocessor port will wakeup in the high-impedance state.

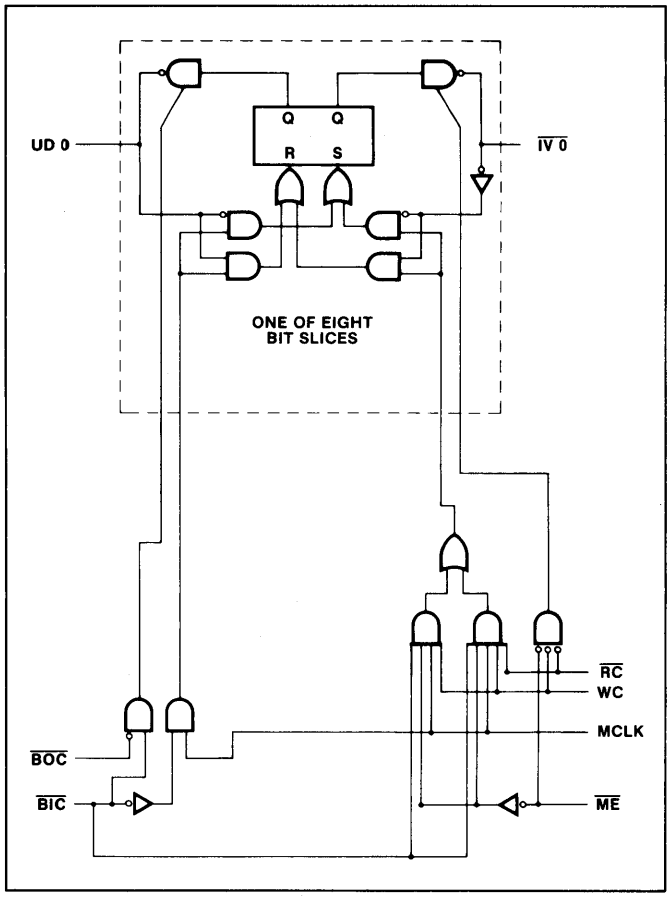
FEATURES

- **Dual bidirectional ports**
- **Independent port operation (User-port priority for data entry)**
- **User data input synchronous**
- **At power-up, User-port outputs are high and Microprocessor-port outputs are high-Z**
- **Three-state TTL outputs for high-drive capabilities**
- **Directly compatible with 8X300 Microcontroller**
- **Single +5V supply**
- **Slim (0.4 in.) 24-pin DIP (8X31 only)**

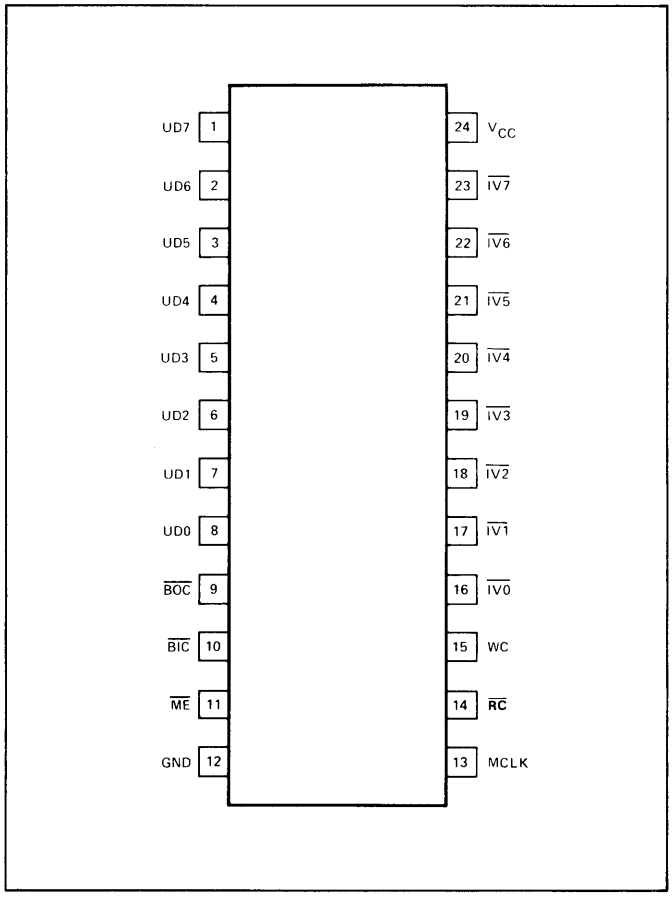
ORDERING INFORMATION

Order number: N8T31N, N8X31N
Packaging information: Refer to Signetics price list
Supply voltage: 5V (±5%)
Operating temperature range: 0°C to +70°C

BLOCK DIAGRAM



PIN CONFIGURATION



PIN DESIGNATION

PIN	SYMBOL	NAME AND FUNCTION	TYPE
1-8	UD0-UD7:	User Data I/O Lines. Bidirectional data lines to communicate with user's equipment.	Acitve high three-state
16-23	$\overline{IV0}$-$\overline{IV7}$:	Microprocessor Bus. Bidirectional data lines to communicate with controlling digital system.	Active low three-state
10	$\overline{BIC}$:	Input Control. User input to control writing into the I/O Port from the user data lines.	Active low
9	$\overline{BOC}$:	Output Control. User input to control reading from the I/O Port onto the user data lines.	Active low
11	$\overline{ME}$:	Master Enable. System input to enable or disable all other system inputs and outputs. It has no effect on user inputs and outputs.	Active low
15	WC:	Write Command. When WC is high, stores contents of $\overline{IV0}$-$\overline{IV7}$ as data.	Active high
14	$\overline{RC}$	Read Command. When RC is low, data is presented on $\overline{IV0}$-$\overline{IV7}$.	Active low
13	MCLK:	Master Clock. Input to strobe data into the latches. See function tables for details.	Active high
24	V_{CC}:	5V power connection.	
12	GND:	Ground.	

$\overline{BIC}$	$\overline{BOC}$	MCLK	USER DATA BUS FUNCTION
H	L	X	Output Data
L	X	H	Input Data
H	H	X	Inactive

H = High Level L = Low Level X = Don't care

Table 1 USER PORT CONTROL FUNCTION

$\overline{ME}$	$\overline{RC}$	WC	MCLK	$\overline{BIC}$	MICROPROCESSOR BUS FUNCTION
L	L	L	X	X	Output Data
L	X	H	H	H	Input Data
X	H	L	X	X	Inactive
X	X	H	X	L	Inactive
H	X	X	X	X	Inactive

Table 2 MICROPROCESSOR PORT CONTROL FUNCTION

USER DATA BUS CONTROL

The activity of the user data bus is controlled by the $\overline{BIC}$ and $\overline{BOC}$ inputs as shown in Table 1.

The user data input is a synchronous function with MCLK. A low level on the $\overline{BIC}$ input allows data on the user data bus to be written into the data latches only if MCLK is at a high level.

To avoid conflicts at the data latches, input from the microprocessor port is inhibited when $\overline{BIC}$ is at a low level. Under all other conditions the two ports operate independently.

MICROPROCESSOR BUS CONTROL

As is shown in Table 2, the activity of the microprocessor port is controlled by the $\overline{ME}$, $\overline{RC}$, WC and $\overline{BIC}$ inputs, as well as the state of an internal status latch. $\overline{BIC}$ is included to show user port priority over the microprocessor port for data input.

BUS OPERATION

Data written into the 8T31/8X31 from one port will appear inverted when read from the other port. Data written into the 8T31/8X31 from one port will not be inverted when read from the same port.

DC ELECTRICAL CHARACTERISTICS

V_{CC} = 5V ± 5%, 0°C ≤ T_A ≤ 70°C unless otherwise specified.

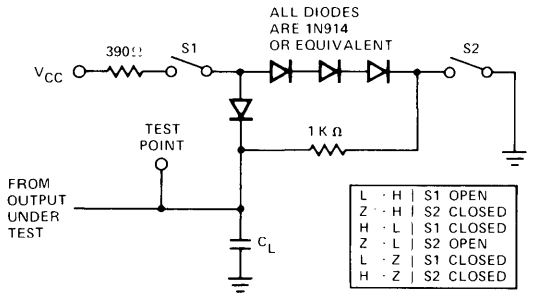
PARAMETER		TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
Input voltage:						V
V_{IH}	High		2.0			
V_{IL}	Low				.8	
V_{IC}	Clamp	I_I = -5mA			-1	
Output voltage:		V_{CC} = 4.75V				V
V_{OH}	High		2.4			
V_{OL}	Low				.55	
Input current[1]:		V_{CC} = 5.25V				μA
I_{IH}	High	V_{IH} = 5.25V		<10	100	
I_{IL}	Low	V_{IL} = .5V		-350	-550	
Output current[2]:						mA
I_{OS}	Short circuit	V_{CC} = 4.75V				
	UD bus		10			
	IV bus		20			
I_{CC}	VCC supply current	V_{CC} = 5.25V		100	150	mA

NOTES

1. The input current includes the three-state/open collector leakage current of the output driver on the data lines.
2. Only one output may be shorted at a time.

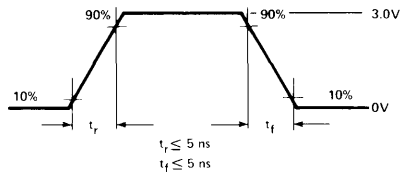
PARAMETER MEASUREMENT INFORMATION

LOAD CIRCUIT FOR THREE-STATE OUTPUTS

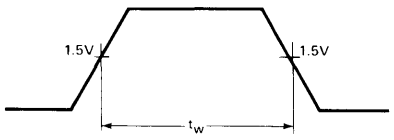


NOTE: C_L includes fixture capacitance

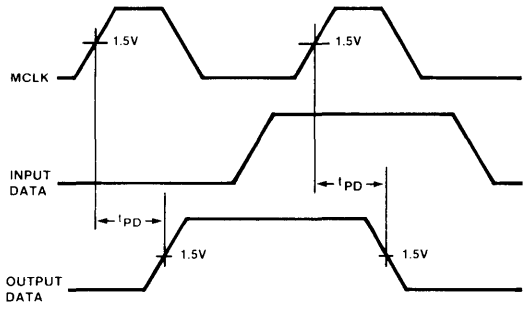
INPUT WAVEFORM



CLOCK PULSE WIDTH



DATA DELAY TIMES
Clock Referenced



AC ELECTRICAL CHARACTERISTICS $0°C \le T_A \le 70°C$, $V_{CC} = 5V \pm 5\%$

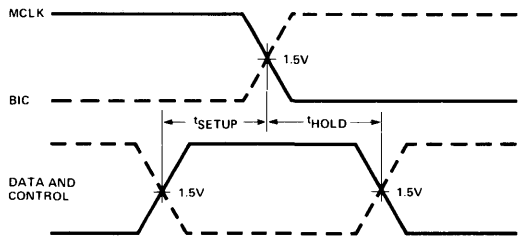
PARAMETER		INPUT	TEST CONDITION	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
t_{PD}	User data delay[1]	UD X			25	38	ns
		MCLK	C_L = 50pF		45	61	ns
t_{OE}	User output enable	$\overline{BOC}$	C_L = 50pF	18	26	47	ns
t_{OD}	User output disable	$\overline{BIC}$	C_L = 50pF	18	28	35	ns
		$\overline{BOC}$		16	23	33	ns
t_{PD}	µP data delay[1]	$\overline{IV}$ X	C_L = 50pF		38	53	ns
		MCLK			48	61	ns
t_{OE}	µP output enable	$\overline{ME}$	C_L = 50pF	14	19	25	ns
		$\overline{RC}$					
		WC					
t_{OD}	µP output disable	$\overline{ME}$	C_L = 50pF	13	17	32	ns
		$\overline{RC}$					
		WC					
t_W	Minimum pulse width	MCLK		40			ns
t_{SETUP}	Minimum setup time[2]	$\overline{UD}$ X[3]		15			ns
		$\overline{BIC}$		25			
		$\overline{IV}$ X		55			
		$\overline{ME}$		30			
		$\overline{RC}$		30			
		WC		30			
t_{HOLD}	Minimum hold time[2]	$\overline{UD}$ X[3]		25			ns
		$\overline{BIC}$		10			
		$\overline{IV}$ X		10			
		$\overline{ME}$		5			
		$\overline{RC}$		5			
		WC		5			

NOTES

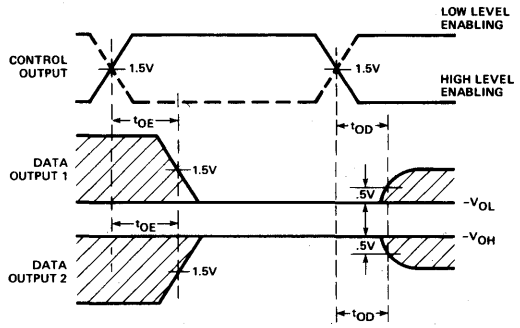
1. Data delays referenced to the clock are valid only if the input data is stable at the arrival of the clock and the hold time requirement is met.
2. Set up and hold times given are for "normal" operation. $\overline{BIC}$ setup and hold times are for a user write operation. $\overline{RC}$ setup and hold times are for an I/O Port select operation. $\overline{ME}$ and WC setup and hold times are for a microprocessor bus write operation.
3. Times are referenced to MCLK.

VOLTAGE WAVEFORMS

SETUP AND HOLD TIMES



OUTPUT ENABLE AND DISABLE TIMES
(Three-State Outputs)



WAVEFORM #1 IS FOR AN OUTPUT WITH INTERNAL CONDITIONS SUCH THAT THE OUTPUT IS LOW WHEN THE THREE-STATE DRIVER IS ENABLED. WAVEFORM #2 IS FOR THE OPPOSITE CONDITION.

PRODUCT IDENTITY

8T32— Three-state, field-programmable (addresses 0-255), synchronous user port

8X32— Three-state, preprogrammed addresses (0-15), synchronous user port in slim-line (0.4 in.) package

8T33— Open-collector, synchronous user port

8T35— Open-collector, asynchronous user port

8T36— Three-state, field-programmable (addresses 0-255), asynchronous user port

8X36— Three-state, preprogrammed addresses (0-15), asynchronous user port in slim-line (0.4 in.) package

8X42— Three-state, 4-input/4-output, preprogrammed addresses (0-15), synchronous user port

ORDERING INFORMATION

All of the above parts can be ordered as field-programmable or with addresses preassigned at the factory. To order, use the following part-number format.

N8XYY-ZZZ-P where,
N8TYY-ZZZ-P

P = F for Ceramic package and
NA for Plastic package

ZZZ = Any address from 000 through 255 (decimal)—256 available addresses for each 8T-- part. For an 8X-- part, preassigned addresses are limited to 000 through 015 (decimal)—16 available addresses.

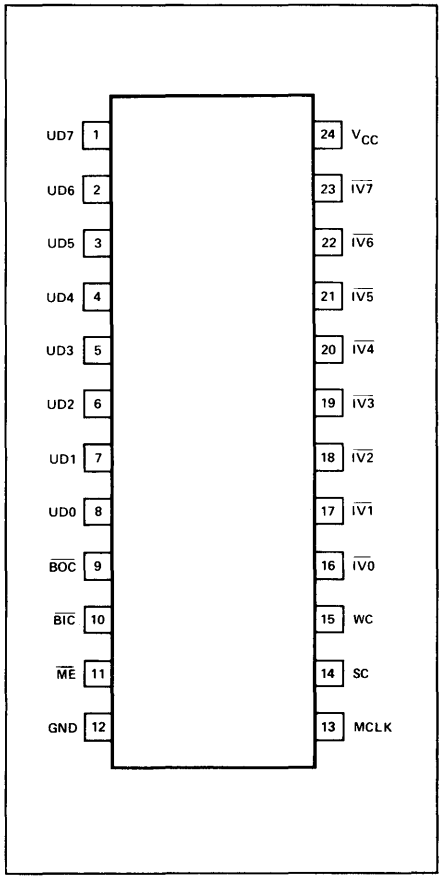
YY = I/O port version.

FEATURES

- **Dual bidirectional ports (except 8X42)**
- **Independent port operation (user-port priority for data entry)**
- **User data input available as synchronous (8T32/8X32/8X42/8T33) or as asynchronous (8T35/8T36/8X36)**
- **User data bus available with three-state (8T32/8X32, 8T36/8X36, and/or 8X42) or open-collector (8T33 and/or 8T35) outputs**
- **At power-up, user-port outputs are high and microprocessor-port outputs are high-z; status latch (from address compare) is also cleared at power-up**
- **Three-state TTL outputs for high-drive capabilities**
- **Directly compatible with 8X300 microcontroller**
- **Single +5V supply**
- **Slim (0.4 in.) 24-pin DIP (8X32/8X36/8X42 only)**

A stock of 8T32s and 8T36s with addresses "1" through "10" are maintained in inventory; with a longer lead time, a small quantity of addresses "11" through "50" are also available.

PIN CONFIGURATION



PRODUCT DESCRIPTIONS

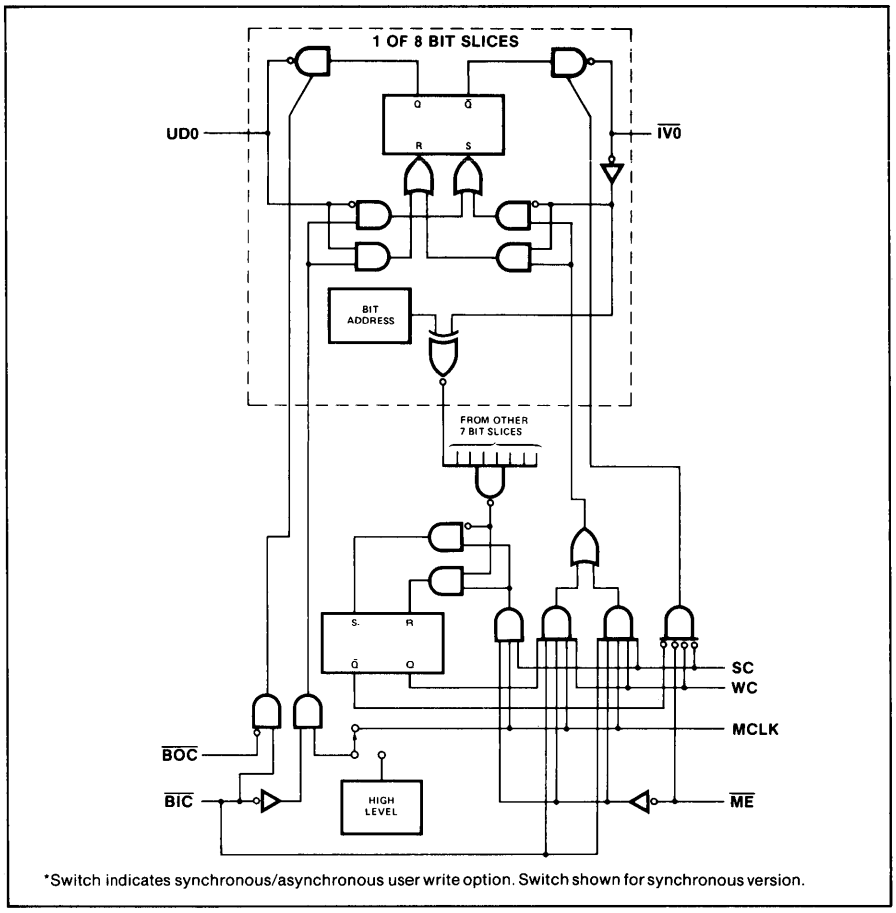
8T32/8T33/8T35/8T36. Each of these I/O Bytes is an addressable and bi-directional register designed for use as an interface element in any system with TTL-compatible buses. (*Note.* Since these I/O Bytes are frequently used with the 8X300 Microcontroller and its associated Interface Vector bus, the 8T32-8T36 family of parts are commonly called $\overline{IV}$ Bytes.) Each I/O Byte contains eight identical data latches (Bits 0 through 7); the latches are accessed from either of two 8-bit ports—one port connecting to the microprocessor (8X300) and the other port connecting to the user device. Separate controls are provided for each port and the two ports operate independently, except when both attempt to input data at the same time; in this case, the user port bus has priority.

The address of each I/O Byte is field-programmable and the microprocessor port is accessed when a valid address is received; the user port is accessible at all times. A selected Byte is automatically deselected when the address of another I/O Byte is sensed on the address/data bus. A Master Enable ($\overline{ME}$) input is available for use as a ninth address bit, allowing direct access to 512 I/O Bytes without address decoding.

A unique feature of these parts is their ability to start up in a predetermined state. If the clock is maintained at a level of less than 0.8 volts until the power supply reaches 3.5 volts, all bits of the user port will wakeup at a "logic 1" level and those of the microprocessor port will wakeup in the high-impedance state.

8X32/8X36. Except for package differences and the fact that the 8X32 and 8X36 are available with preprogrammed addresses (0

TYPICAL BLOCK DIAGRAM



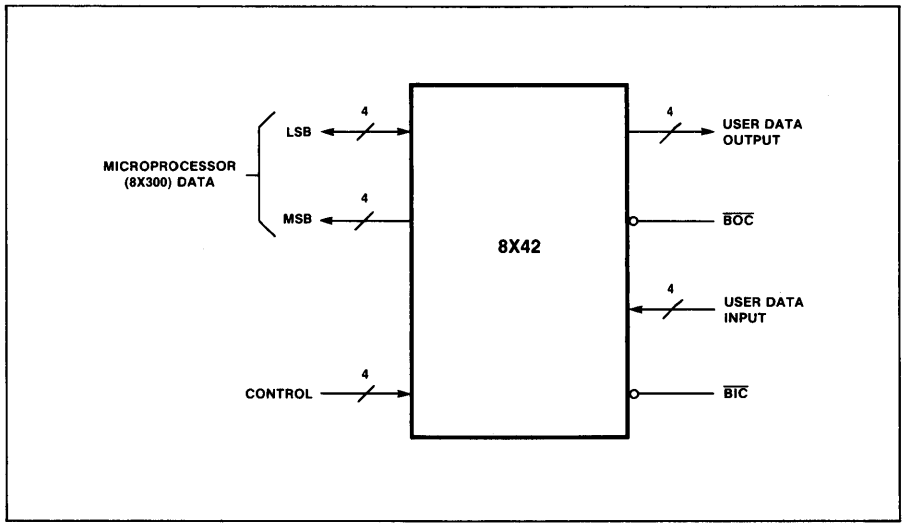
*Switch indicates synchronous/asynchronous user write option. Switch shown for synchronous version.

through 15), these two devices are functionally identical to their 8Txx counterparts; design parameters and interface requirements are exactly the same.

8X42. This part differs from the 8X32 in the following ways:

- Bits 0-3 (pins 8-5, respectively) are dedicated user inputs controlled by Byte Input Control ($\overline{BIC}$/pin 10). Byte Output Control ($\overline{BOC}$/pin 9) does not affect these bits and the user output drivers do not exist for these bits.
- Bits 4-7 (pins 4-1, respectively) are dedicated user outputs whose three-state drivers are controlled by $\overline{BOC}$; $\overline{BIC}$ has no control over these bits.
- Pin numbers, software requirements, and AC characteristics of the 8X42 are identical to those of the 8X32, ergo 8T32. A simplified block diagram of the 8X42 is shown in the accompanying diagram.

SIMPLIFIED DIAGRAM OF 8X42



PIN DESCRIPTION

PIN	SYMBOL	NAME AND FUNCTION	TYPE
1-8	UD0-UD7:	User Data I/O Lines. Bidirectional data lines to communicate with user's equipment. Either tri-state or open collector outputs are available.	Active high
16-23	$\overline{IV0}$-$\overline{IV7}$:	Microprocessor Bus. Bidirectional data lines to communicate with controlling digital system (microprocessor).	Active low three-state
10	$\overline{BIC}$:	Input Control. User input to control writing into the I/O Port from the user data lines.	Active low
9	$\overline{BOC}$:	Output Control. User input to control reading from the I/O Port onto the user data lines.	Active low
11	$\overline{ME}$:	Master Enable. System input to enable or disable all other system inputs and outputs. It has no effect on user inputs and outputs.	Active low
15	WC:	Write Command. When WC is high and SC is low, I/O Port, if selected, stores contents of $\overline{IV0}$-$\overline{IV7}$ as data.	Active high
14	SC:	Select Command. When SC is high and WC is low, data on $\overline{IV0}$-$\overline{IV7}$ is interpreted as an address. I/O Port selects itself if its address is identical to μP bus data; it de-selects itself otherwise.	Active high
13	MCLK:	Master Clock. Input to strobe data into the latches. See function tables for details.	Active high
24	VCC:	5V power connection.	
12	GND:	Ground.	

$\overline{BIC}$	$\overline{BOC}$	MCLK	USER DATA BUS FUNCTION	
			8T32/8X32/8T33	8T35/8T36/8X36
H	L	X	Output Data	Output Data
L	X	H	Input Data	Input Data
L	X	L	Inactive	Input Data
H	H	X	Inactive	Inactive

H = High Level L = Low Level X = Don't care

Table 1 USER PORT CONTROL FUNCTION

$\overline{ME}$	SC	WC	MCLK	$\overline{BIC}$	STATUS LATCH	I/O PORT FUNCTION
L	L	L	X	X	SET	Output Data
L	L	H	H	H	SET	Input Data
L	H	L	H	X	X	Input Address
L	H	H	H	L	X	Input Address
L	H	H	H	H	X	Input Data and Address
L	X	H	L	X	X	Inactive
L	H	X	L	X	X	Inactive
L	L	H	H	L	X	Inactive
L	L	X	X	X	Not Set	Inactive
H	X	X	X	X	X	Inactive

Table 2 MICROPROCESSOR PORT CONTROL FUNCTION

USER DATA BUS CONTROL

The activity of the user data bus is controlled by the $\overline{BIC}$ and $\overline{BOC}$ inputs as shown in Table 1.

For the 8T32, 8X32, 8T33, and 8X42, user data input is a synchronous function with MCLK. A low level on the $\overline{BIC}$ input allows data on the user data bus to be written into the data latches only if MCLK is at a high level. For the 8T35, 8T36 and 8X36, user data input is an asynchronous function. A low level on the $\overline{BIC}$ input allows data on the user data bus to be latched regardless of the level of the MCLK input. Note that when the 8T35, 8T36, or 8X36 is used with the 8X300 Microcontroller, care must be taken to insure that the Microprocessor bus is stable when it is being read by the 8X300 Microcontroller.

To avoid conflicts at the Data Latches, input from the Microprocessor Port is inhibited when $\overline{BIC}$ is at a low level. Under all other conditions the two ports operate independently.

MICROPROCESSOR BUS CONTROL

As is shown in Table 2, the activity of the microprocessor port is controlled by the $\overline{ME}$, SC, WC and $\overline{BIC}$ inputs, as well as the state of an internal status latch. $\overline{BIC}$ is included to show user port priority over the microprocessor port for data input.

Each I/O Port's status latch stores the result of the most recent I/O Port select; it is set when the I/O Port's internal address matches the Microprocessor Bus. It is cleared when an address that differs from the internal address is presented on the Microprocessor Bus.

In normal operation, the state of the status latch acts like a master enable; the microprocessor port can transfer data only when the status latch is set.

When SC and WC are both high, data on the Microprocessor Bus is accepted as data, whether or not the I/O Port was selected. The data is also interpreted as an address. The I/O Port sets its select status if its address matches the data read when SC and WC were both high; it resets its select status otherwise.

BUS OPERATION

Data written into the I/O Port from one port will appear inverted when read from the other port. Data written into the I/O Port from one port will not be inverted when read from the same port.

AC ELECTRICAL CHARACTERISTICS $0°C \le T_A \le 70°C$, $V_{CC} = 5V \pm 5\%$

PARAMETER	INPUT	TEST CONDITION	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
t_{PD} User data delay (Note 1)	UD X	C_L = 50pF		25	38	ns
	MCLK*			45	61	
	$\overline{BIC}$†			40	55	
t_{OE} User output enable	$\overline{BOC}$	C_L = 50pF	18	26	47	ns
t_{OD} User output disable	$\overline{BIC}$	C_L = 50pF	18	28	35	ns
	$\overline{BOC}$		16	23	33	
t_{PD} µP data delay (Note 1)	$\overline{IV}$ X	C_L = 50pF		38	53	ns
	MCLK			48	61	
t_{OE} µP output enable	$\overline{ME}$	C_L = 50pF	14	19	25	ns
	SC					
	WC					
t_{OD} µP output disable	$\overline{ME}$	C_L = 50pF	13	17	32	ns
	SC					
	WC					
t_W Minimum pulse width	MCLK		40			ns
	$\overline{BIC}$†		35			
t_{SETUP} Minimum setup time	UD X□	(Note 2)	15			ns
	$\overline{BIC}$*		25			
	$\overline{IV}$ X		55			
	$\overline{ME}$		30			
	SC		30			
	WC		30			
t_{HOLD} Minimum hold time	UD X□	(Note 2)	25			ns
	$\overline{BIC}$*		10			
	$\overline{IV}$ X		10			
	$\overline{ME}$		5			
	SC		5			
	WC		5			

* Applies for 8T32, 8X32 and 8T33 only.

† Applies for 8T35, 8T36 and 8X36 only.

□ Times are referenced to MCLK for 8T32, 8X32 and 8T33, and are referenced to $\overline{BIC}$ for 8T35, 8T36 and 8X36.

NOTES:

1. Data delays referenced to the clock are valid only if the input data is stable at the arrival of the clock and the hold time requirement is met.
2. Set up and hold times given are for "normal" operation. $\overline{BIC}$ setup and hold times are for a user write operation. SC setup and hold times are for an I/O Port select operation. WC setup and hold times are for a Microprocessor Bus write operation. $\overline{ME}$ setup and hold times are for both IV write and select operations.

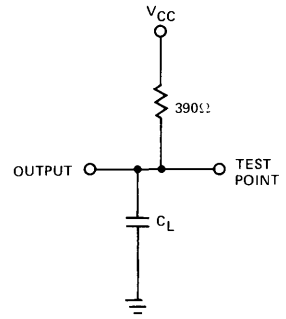
DC ELECTRICAL CHARACTERISTICS $0°C \leq T_A \leq 70°C$, $V_{CC} = 5V \pm 5\%$

PARAMETER		TEST CONDITIONS	LIMITS			UNITS
			Min	Typ	Max	
V_{IH}	High-level input voltage		2.0		5.5	V
V_{IL}	Low-level input voltage		-1.0		.8	V
V_{CL}	Input clamp voltage	$I_I = -5mA$			-1.0	V
I_{IH}	High-level input current[1]	$V_{CC} = 5.25V$ $V_{IH} = 5.25V$		<10	100	μA
I_{IL}	Low level input current[1]	$V_{CC} = 5.25V$ $V_{IL} = .5V$		-350	-550	μA
V_{OL}	Low-level output voltage	$V_{CC} = 4.75V$ $I_{OL} = 16mA$			.55	V
V_{OH}	High-level output voltage	$V_{CC} = 4.75V$ $I_{OH} = -3.2mA$	2.4			V
I_{OS}	Short-circuit output current[2] UD bus IV bus	 $V_{CC} = 4.75V$ $V_{CC} = 4.75V$	 10 20			 mA mA
I_{CC}	Supply current	$V_{CC} = 5.25V$		100	150	mA

NOTES
1. The input current includes the Three-state/Open Collector leakage current of the output driver on the data lines.
2. Only one output may be shorted at a time.
3. These limits do not apply during address programming.

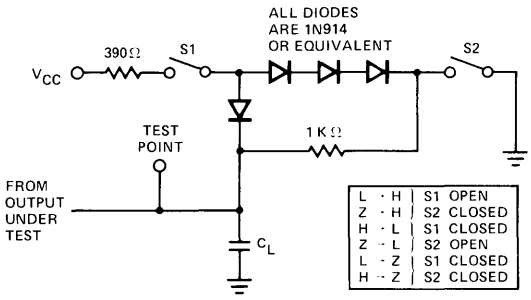
Absolute Maximum Ratings:
Supply voltage[3] 7V
Input voltage[3] 5.5V

TEST LOAD CIRCUIT (OPEN COLLECTOR OUTPUTS)



NOTE: C_L includes fixture capacitance.

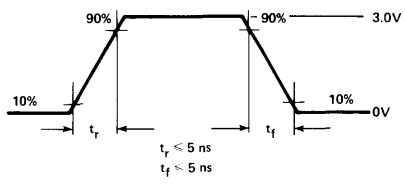
TEST LOAD CIRCUIT (THREE-STATE OUTPUTS)



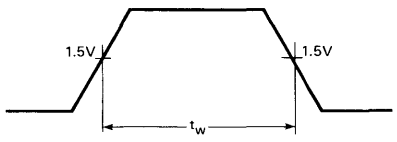
NOTE: C_L includes fixture capacitance.

VOLTAGE WAVEFORMS

INPUT WAVEFORM

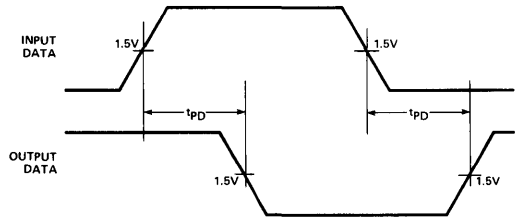


CLOCK PULSE WIDTH

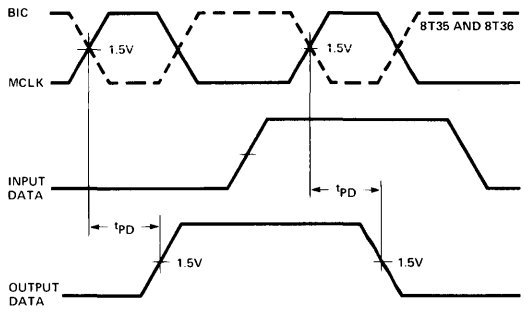


VOLTAGE WAVEFORMS (Cont'd)

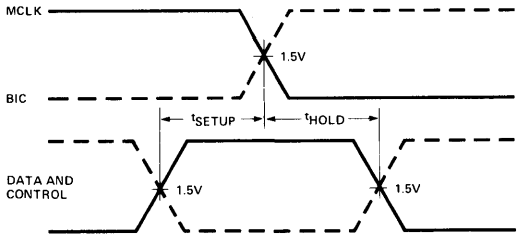
DATA DELAY TIMES
Input Data Reference



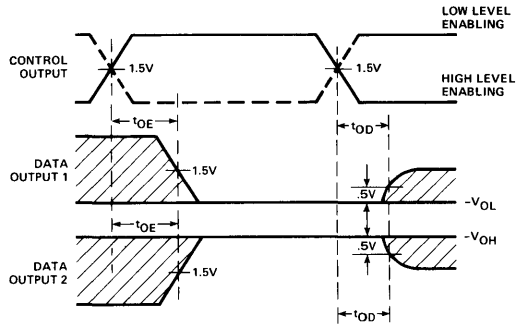
DATA DELAY TIMES
Clock Referenced



SETUP AND HOLD TIMES



OUTPUT ENABLE AND DISABLE TIMES
(Three-State Outputs)



Waveform #1 is for an output with internal conditions such that the output is Low when the three-state driver is enabled. Waveform #2 is for the opposite condition.

ADDRESS PROGRAMMING

The I/O Port is manufactured such that an address of all high levels (>2V) on the Microprocessor Bus inputs matches the Port's internal address. To program a bit so a low-level input (<0.8V) matches, the following procedure should be used:

1. Set all control inputs to their inactive state ($\overline{BIC}$ = $\overline{BOC}$ = $\overline{ME}$ = V_{CC}, SC = WC = MCLK = GND). Leave all Microprocessor Bus I/O pins open.
2. Raise V_{CC} to 7.75V ± .25V.
3. After V_{CC} has stabilized, apply a single programming pulse to the user data bus bit where a low-level match is desired. The voltage should be limited to 18V; the current should be limited to 75mA. Apply the pulse as shown in Figure 1.
4. Return V_{CC} to 0V. (Note 1).
5. Repeat this procedure for each bit where a low-level match is desired.
6. Verify that the proper address is programmed by setting the Port's status latch ($\overline{IV0}$-$\overline{IV7}$ = desired address, $\overline{ME}$ = WC = L, SC = MCLK = H). If the proper address has been programmed, data presented at the µP bus will appear inverted on the user bus outputs. (Use normal V_{CC} and input voltage for verification.)

After the desired address has been programmed, a second procedure must be followed to isolate the address circuitry. The procedure is:

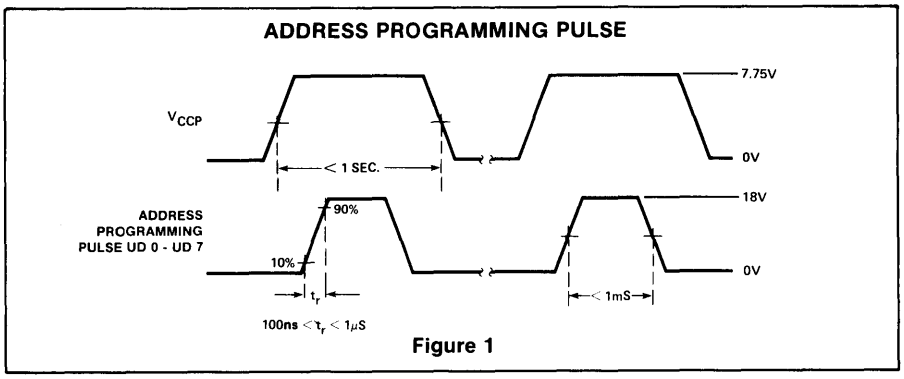


Figure 1

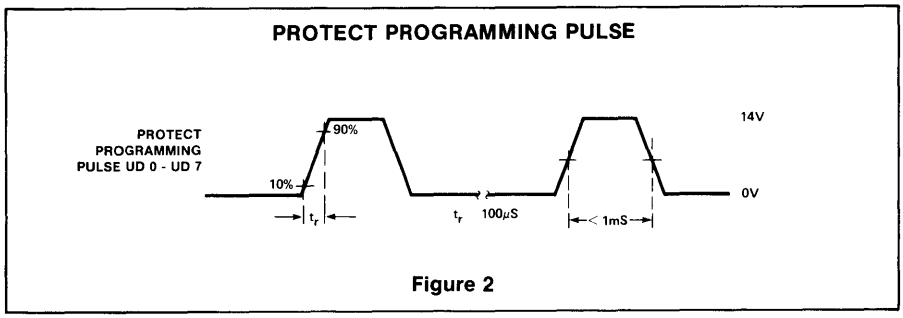


Figure 2

1. Set V_{CC} and all control inputs to 0V. (V_{CC} = $\overline{BIC}$ = $\overline{BOC}$ = $\overline{ME}$ = SC = WC = MCLK = 0V). Leave all Microprocessor Bus I/O pins open.
2. Apply a protect programming pulse to every user data bus pin, one at a time. The voltage should be limited to 14V; the current should be limited to 150mA. Apply the pulse as shown in Figure 2.
3. Verify that the address circuitry is isolated by applying 7V to each user data bus pin and measuring less than 1mA of input current. The conditions should be the same as in step 1 above. The rise time on the verification voltage must be slower than 100µs.

PROGRAMMING SPECIFICATIONS[1]

	PARAMETER	TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNITS
V_{CCP}	Programming supply voltage					
	Address		7.5		8.0	V
	Protect			0		V
I_{CCP}	Programming supply current	V_{CCP} = 8.0V			250	mA
	Max time V_{CCP} > 5.25V				1.0	s
	Programming voltage					
	Address		17.5		18.0	V
	Protect		13.5		14.0	V
	Programming current					
	Address				75	mA
	Protect				150	mA
	Programming pulse rise time					
	Address		.1		1	µs
	Protect		100			µs
	Programming pulse width		.5		1	ms

NOTE

1. If all programming can be done in less than 1 second, V_{CC} may remain at 7.75V for the entire programming cycle.

APPLICATIONS

Figure 3 shows some of the various ways to use the I/O Port in a system. By controlling the $\overline{BIC}$ and $\overline{BOC}$ lines, the device may be used for the input and output of data, control, and status signals. I/O Port 1 functions bidirectionally for data transfer and I/O Port 2 provides a similar function for discrete status and control lines. I/O Ports 3 and 4 serve as dedicated output and input ports, respectively.

APPLICATIONS

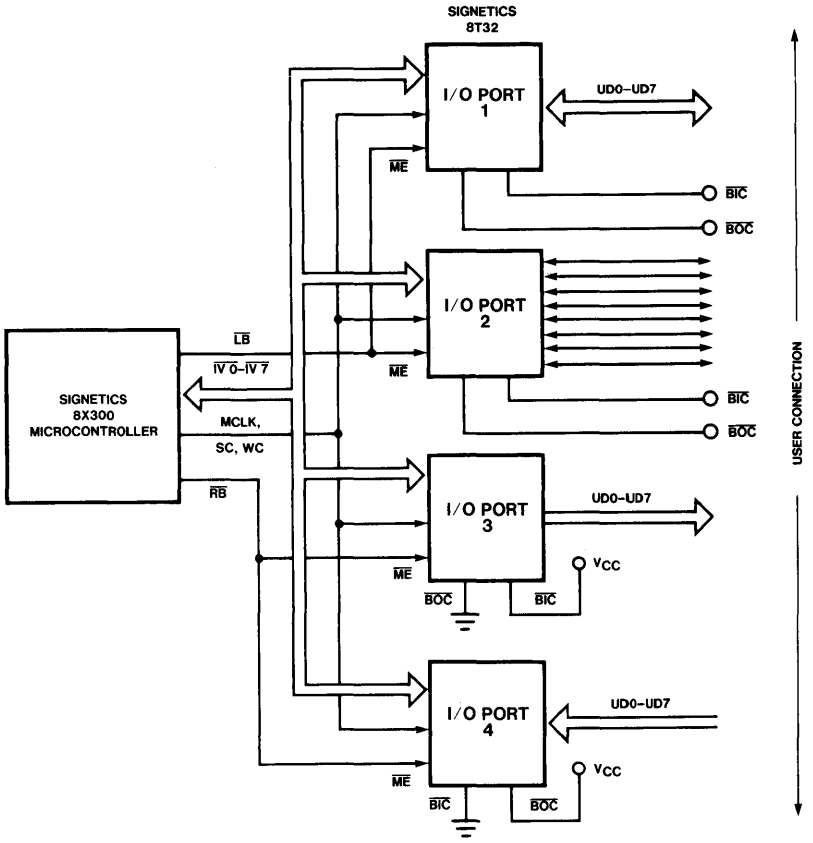


Figure 3

DESCRIPTION

The Bus Expander is specifically designed to increase the I/O capability of 8X300 systems previously limited by fanout considerations. The bus expander serves as a buffer between the 8X300 and blocks of I/O devices. Each bus expander can buffer a block of 16 I/O ports while only adding a single load to the 8X300.

FEATURES

- **15ns max propagation delay**
- **Bidirectional**
- **Three-state outputs on both ports**
- **Pre-programmed address range**

APPLICATIONS

The 8T39 Bus Expander is designed to be used with the 8X300 microprocessor to allow increased I/O capability in those systems previously limited by fanout considerations. Figure 1 shows a typical arrangement of the bus expander in an 8X300 system. Each expander services I/O ports whose address is within the range of the expander. Other I/O ports or working storage may be directly connected to the bus as shown.

The bus expander is not limited to use with the 8X300, but may be applied in any system which uses a combined address/data bus.

8T39 ADDRESSING

During normal operation of the 8X300 when an I/O port address is being sent on the IV Bus (SC is high), the I/O port will examine all eight bits of the microprocessor bus for an address compare. Since the 8T39 is used to buffer blocks of I/O ports, only the four most significant bits are examined by the 8T39 for an address compare.

Note that redundant addresses are not programmed into separate devices. Rather, a discrete device (such as the 8T39-03) may be wired for any address requiring two 1 bits and two 0 bits in the address. The various address ranges for this same device are obtained by permuting the high order (DI0 and DO0 are MSB) data lines accordingly. Both input and output lines must be redefined in order to maintain data and address integrity on the extended bus. Table 1 summarizes the 8T39 addressing.

Address functions are specified with the convention that bit 0 is the MSB and bit 7 is the LSB. The DI microprocessor bus address decoding is active low.

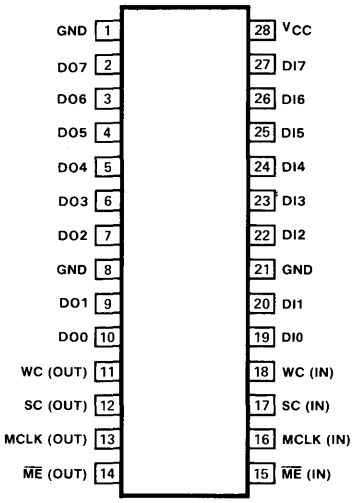
FUNCTIONAL DESCRIPTION

The Bus Expander contains eight sets of non-inverting bidirectional tri-state drivers for the bus data bits, four non-inverting unidirectional drivers for I/O port control, and necessary control logic. The control logic is required to maintain the proper directional transfer of bus data as dictated by the states of the I/O port control signals and the currently enabled I/O port. Each bus expander is programmed during manufacturing to respond to a specific block of I/O port addresses. Only I/O ports with addresses in the range of a given bus expander may be connected to that expander. A bus expander may be used on either left bank or right bank. Multiple expanders on the same bank must have different address ranges; however, expanders with the same address range can be connected if they are on different banks. Systems may be configured with I/O ports connected directly to the 8X300, as well as I/O ports connected through a bus expander; however, no unbuffered I/O port may have an address within the span of a bus expander on the same bank.

Addition of bus expanders may impact system cycle time due to the added delay in the data path. For the purposes of calculating allowable cycle time as described in the 8X300 data sheet, the bus expander delays may be considered additive to the I/O port delays so that a buffered I/O port simply appears as a slower I/O port.

PIN CONFIGURATION

I,XL PACKAGE



PIN DESIGNATION

PIN NO.	SYMBOL	NAME AND FUNCTION	TYPE
2-7,9,10	DO0-DO7	I/O port data bus	Active low, three-state
11	WC(OUT)	Write command output	Active high
12	SC(OUT)	Select command output	Active high
13	MCLK(OUT)	Master clock output	Acitve high
14	$\overline{ME}$(OUT)	Master enable output	Active low
15	$\overline{ME}$(IN)	Master enable input	Active low
16	MCLK(IN)	Master clock input	Active high
17	SC(IN)	Select command output	Active high
18	WC(IN)	Write command output	Active high
19,20,22-27	DI0-DI7	Microprocessor data bus	Active low, three-state
1,8,21	GND	Ground	
28	V_{CC}	+5 volt supply	

PART TYPE	ADDRESS PATTERN MSB(0) LSB(7)	ADDRESS BLOCKS Octal
8T39-00	0000XXXX	0-17
8T39-01	0001XXXX	20-37, 40-57, 100-117, 200-217
8T39-03	0011XXXX	60-77, 120-137, 220-237, 140-157, 240-257, 300-317
8T39-07	0111XXXX	160-177, 260-277, 320-337, 340-357
8T39-17	1111XXXX	360-377

Table 1 8T39 ADDRESSING SUMMARY

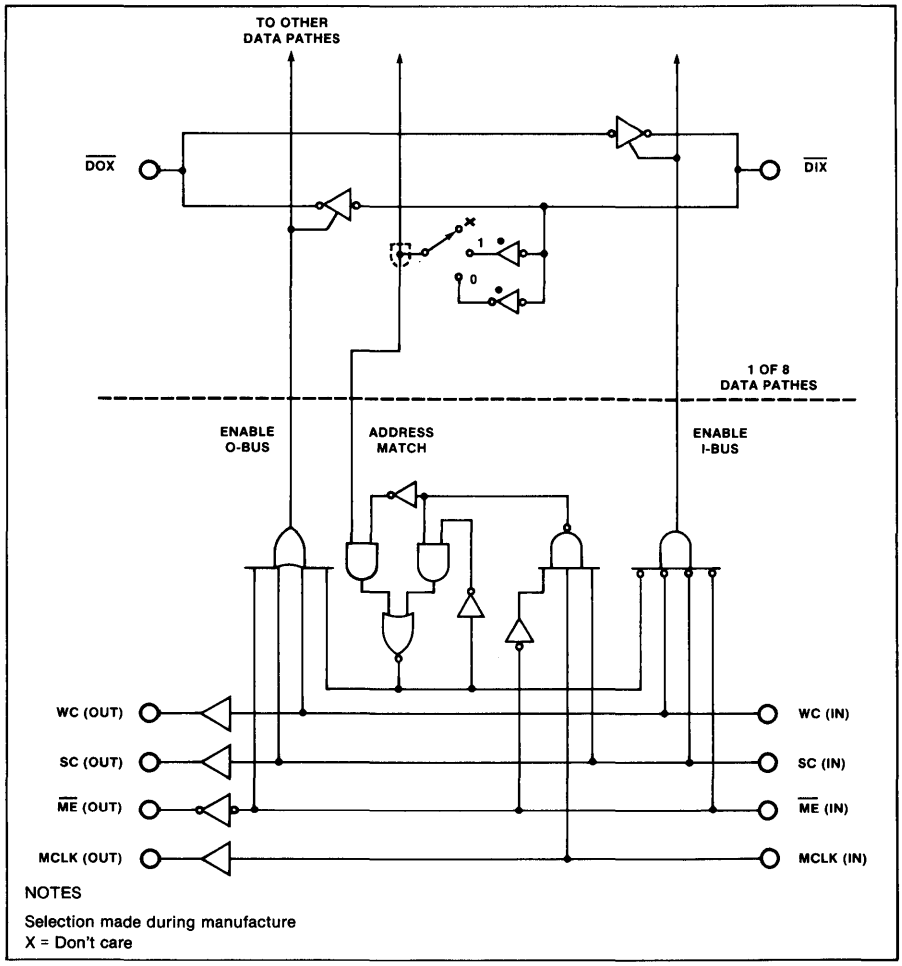
TRUTH TABLE

$\overline{ME}$	SC	WC	MCLK	SELECT LATCH	DATA TRANSFER DIRECTION	ADDRESS* COMPARISON
L	L	L	X	Set	DI Bus ← DO Bus	No
L	L	L	X	Not set	DI Bus → DO Bus	No
L	L	H	X	X	DI Bus → DO Bus	No
L	H	X	L	X	DI Bus → DO Bus	No
L	H	X	H	X	DI Bus → DO Bus	Yes
H	X	X	X	X	DI Bus → DO Bus	No

NOTES

*When an address comparison is made, the select latch is set if the data on the DI Bus is within the manufactured address range of the IV Bus Expander. Otherwise, the select latch is cleared.

FUNCTIONAL BLOCK DIAGRAM



NOTES

Selection made during manufacture
X = Don't care

ABSOLUTE MAXIMUM RATINGS[1]

PARAMETER		RATING	UNIT
V_{CC}	Power supply voltage	+7	Vdc
V_{IN}	Input voltage	+5.5	Vdc
V_O	Off-state output voltage	+5.5	Vdc
T_A	Operating temperature range	0 to +70	°C
T_{STG}	Storage temperature range	-65 to +150	°C

ORDERING INFORMATION

The Bus Expander is ordered by specifying the following part number:

N8T39-XX P

- P = { I - Ceramic Package; XL - Epoxy Package
- XX = Address Range As Determined From Table 1

DC ELECTRICAL CHARACTERISTICS V_{CC} = 5V ± 5%, 0°C ≤ T_A ≤ 70°C

PARAMETER		TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
	Input voltage					V
V_{IL}	Low				.8	
V_{IH}	High	-5mA at V_{CC} min	2.0			
V_{IC}	Clamp				-1	
	Output voltage	V_{CC} = 4.75V				V
V_{OL}	Low	I_{OL} = 16mA			.55	
V_{OH}	High	I_{OH} = -3.2mA	2.4			
	Input current	V_{CC} = 5.25V				uA
I_{IL}	Low[1]	V_{IL} = .5V			-250	
I_{IH}	High[1]	V_{IH} = 5.25V		< 10	100	
I_{OS}	Short circuit output current	V_{CC} = 4.75V	-40			mA
I_{CC}	Supply current	V_{CC} = 5.25V			200	mA

AC ELECTRICAL CHARACTERISTICS V_{CC} = 5V ± 5%, 0°C ≤ T_A ≤ 70°C, C_L = 300pF[2]

PARAMETER		TO	FROM	TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
t_{pd}	Propagation Delay Data	DOX DIX	DIX DOX				15	ns
t_{pd}	Control Propagation Delay	$\overline{ME}$ (out) MCLK (out) SC (out) WC (out)	$\overline{ME}$ (in) MCLK (in) SC (in) WC (in)				15	
t_{oe}	Data Output Enable	DIX DOX	$\overline{ME}$ (in) SC (in) WC (in)		28		56	ns
t_{od}	Data Output Disable	DIX DOX	$\overline{ME}$ (in) SC (in) WC (in)		15			ns
t_{setup}	Adverse Setup Time[3]	DIX DOX	DIX $\overline{ME}$ (in) MCLK (in) SC (in) WC (in)		54			ns
t_{hold}	Address Hold Time[3]	DIX DOX	DIX $\overline{ME}$ (in) MCLK (in) SC (in) WC (in)		3			ns

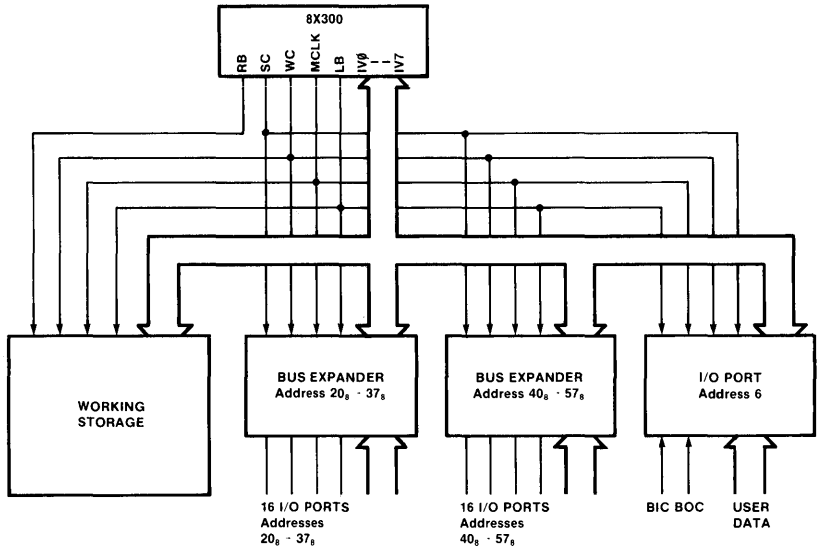
NOTES

1. Includes tri-state leakage.
2. Minimum clock width ≈ 50ns.
3. All set up and hold times are referenced to the trailing edge of the clocking input MCLK.

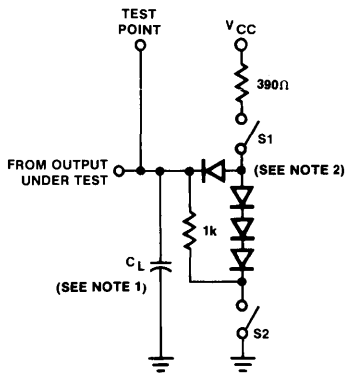
PRELIMINARY SPECIFICATION 8T39-I,XL

TYPICAL APPLICATIONS

USING 2 BUS EXPANDERS TO GIVE 33 I/O PORTS PLUS WORKING STORAGE



TEST LOAD CIRCUIT



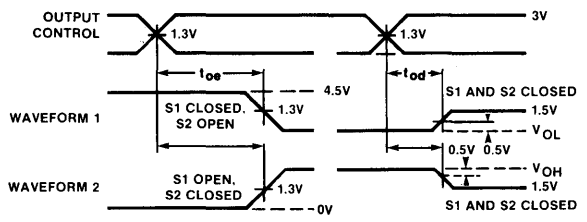
All resistors values are typical and in ohms.

NOTES

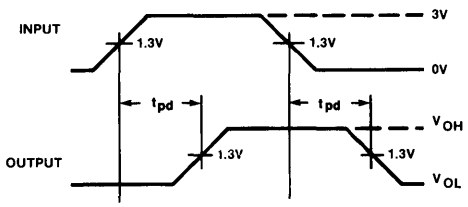
1. C_L includes probe and jig capacitance.
2. All diodes are 1N916 or 1N3064.

VOLTAGE WAVEFORMS

PROPAGATION DELAY TO THREE-STATE OUTPUTS



PROPAGATION DELAY TIMES



PRODUCT DESCRIPTION

The Signetics Asynchronous Bidirectional Bus Extender and Repeater (SABER) is a general purpose device ideal for system bus expansion applications. SABER consists of eight data channels, each with one pair of terminals (A_i and B_i); each data channel can be operated independently.

The device requires no external controls since all intelligence is internally generated; thus, operation of the device is completely autonomous. The first logic low signal that occurs on one channel terminal (A_i or B_i) will be repeated on the corresponding terminal (B_i or A_i) of the same channel.

The 8X41 is designed for use in open-collector bus systems where high speed and low-current inputs/high-current outputs are required. In system configurations, the discrete capabilities of SABER can be expanded by parallel connection to service any number of bits. To provide reliable operation and integrity of data transfers, all channels are disabled by an on-chip power monitor whenever V_{CC} falls below approximately 4V.

FUNCTIONAL OPERATION

The 8X41 (Figure 1) consists of eight functionally independent yet logically identical channels. Each channel consists of two bus terminals (A_i and B_i); each terminal is internally connected to an open-collector driver and a high-impedance receiver. The monitoring state of each channel is defined when both terminals (A_i and B_i) are "high"; in this state, the internal logic of SABER continually examines the A and B bus signals to determine signal direction—A_i to B_i or B_i to A_i. A low signal occurring at either of the two terminals causes the open-collector driver on the opposite terminal to follow suit; hence, the signal is repeated by the 8X41. For each channel, latches L1 and L2 determine signal direction. As shown in the truth table for these latches, there is no transmission of data when both signals are low; however, this condition should never occur during normal system operation.

The internal automatic direction control can be overridden by either or both of the common disable inputs—$\overline{DBA}$ and $\overline{DAB}$. When $\overline{DBA}$ is driven low ($\overline{DAB}$ = high), the B_i to A_i path is interrupted and SABER becomes a unidirectional repeater in the A_i to B_i direction only. With these conditions reversed ($\overline{DAB}$ = low and $\overline{DBA}$ = high), the A_i to B_i path is interrupted and the chip functions as a unidirectional repeater in the B_i to A_i direction. When both control signals are low, data passage is inhibited in both directions. Refer to the I/O truth table for all possible input/output conditions.

DESIGN FEATURES

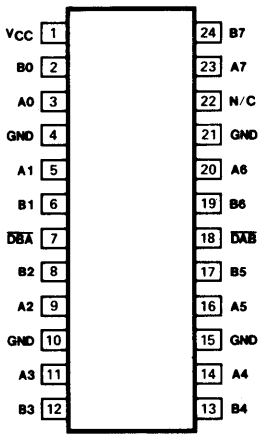
- **Intelligent bidirectional bus repeater with self-generating or external control**
- **Eight independent channels**
- **Open-collector outputs (meets DEC UNIBUS* specifications)**
- **TTL compatible**
- **High speed (30-nanoseconds max)**
- **Expandable to any number of bits**
- **High input impedance for every operating value of V_{CC}**
- **Low input current (less than 100-microamperes); high output current (up to 70-milliamperes)**
- **24-pin DIP**
- **+5V supply**

USE AND APPLICATION

- **Minicomputers**
- **Microcomputers MOS/Bipolar**
- **Communications**
- **Signal buffer**
- **Bus fan-out extensions**
- **Distributed processing**
- **Bidirectional bus connector/isolator**

*Trademark of the Digital Equipment Corporation

8X41 PACKAGE/PIN DESIGNATIONS



PIN NO.	IDENTIFIER	FUNCTION
1	V_{CC}	+5V power
3, 5, 9, 11, 14, 16, 20, and 23	A_0-A_7	A-bus input/output
2, 6, 8, 12, 13, 17, 19, and 24	B_0-B_7	B-bus input/output
4, 10, 15, and 21	GND	Circuit ground
7 & 18	$\overline{DBA}$ & $\overline{DAB}$	Input/output control-see TRUTH TABLE in Figure 1
22	N/C	No connection

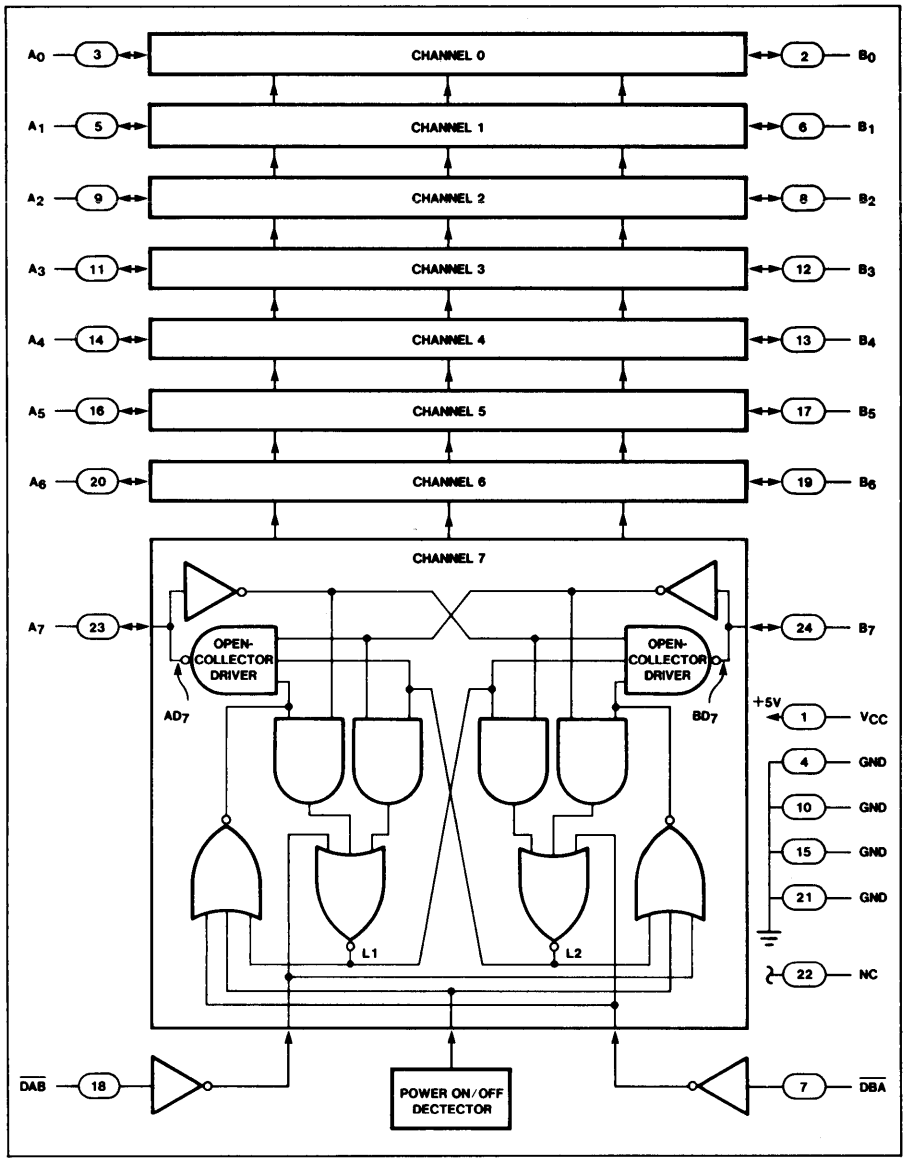


Figure 1. Logic Diagram of 8X41 Asynchronous Bidirectional Bus Extender and Repeater

INPUT/OUTPUT TRUTH TABLE

EXTERNAL CONTROLS		INPUT SIGNALS		OUTPUT DRIVER SIGNALS	
$\overline{DAB}$	$\overline{DBA}$	A_i	B_i	AD_i	BD_i
H	H	L	L	H	H
H	H	L	H	H	L
H	H	H	L	L	H
H	H	H	H	H	H
H	L	L	L	H	L
H	L	L	H	H	L
H	L	H	L	H	H
H	L	H	H	H	H
L	H	L	L	L	H
L	H	L	H	H	H
L	H	H	L	L	H
L	H	H	H	H	H
L	L	X	X	H	H

$\overline{DBA}$	$\overline{DAB}$	FUNCTION
Ø	Ø	Data transmission inhibited
Ø	1	$A_i \rightarrow B_i$
1	Ø	$A_i \leftarrow B_i$
1	1	$A_i \rightarrow B_i$ $A_i \leftarrow B_i$

i = Channel 0, 1, 2, 3, 4, 5, 6, or 7
$A_i \rightarrow B_i$ = Data transmission from A_i to B_i
$A_i \leftarrow B_i$ = Data transmission from B_i to A_i

TRUTH TABLE FOR INTERNAL LATCHES

LATCHES		DIRECTION OF DATA
L1	L2	
1	1	Monitoring state
1	Ø	A_i to B_i
Ø	1	B_i to A_i
Ø	Ø	No transmission

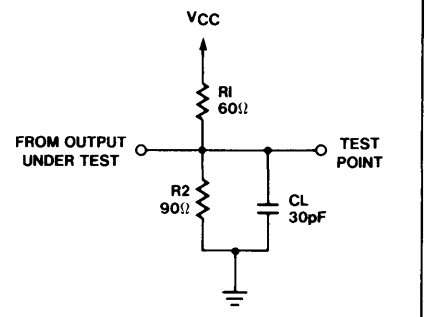
NOTES
A_i = External signal
AD_i = Output A driver
B_i = External signal
BD_i = Output B driver
X = Don't care

DC CHARACTERISTICS V_{CC} = 5V (±5%); T_A = 0°C to 70°C

PARAMETER	DESCRIPTION	TEST CONDITIONS	LIMITS			UNITS
			Min	Typ	Max	
V_{OL}	Bus output low voltage (driver ON)	I_{OL} = 70 mA; V_{CC} = Min			0.5	V
*V_B	Bus input threshold voltage (driver OFF)		1.3		1.7	V
V_{IH} ($\overline{DBA}$, $\overline{DAB}$ only)	High level input voltage		2.0			V
V_{IL} ($\overline{DBA}$, $\overline{DAB}$ only)	Low level input voltage				0.8	V
V_{IC}	Input clamp voltage	V_{CC} = Min; I_{IL} = −18mA			−1.5	V
V_{PD}	Power ON/OFF detector threshold voltage		3.7		4.35	V
I_{IH} ($\overline{DBA}$, $\overline{DAB}$ only)	High level input current	V_{CC} = Max; V_{IN} = 2.7V			20	μA
I_{IL} ($\overline{DBA}$, $\overline{DAB}$ only)	Low level input current	V_{CC} = Max; V_{IN} = 0.4V			−0.4	mA
I_I	Bus input current (driver OFF)	V_{CC} = Max; V_B = 2.5V*			100	μA
		V_{CC} = Max; V_B = 0V*			−20	μA
I_{OFF}	Bus leakage current (power OFF)	V_{CC} = 0V; V_B = 2.5V*			100	μA
I_{CC}	Supply current	V_{CC} = Max; A_0-A_7 = Low or B_0-B_7 = Low and $\overline{DBA}$ = $\overline{DAB}$ = High		145	180	mA

*V_B = V_{BUS}

LOAD CIRCUIT FOR OUTPUTS



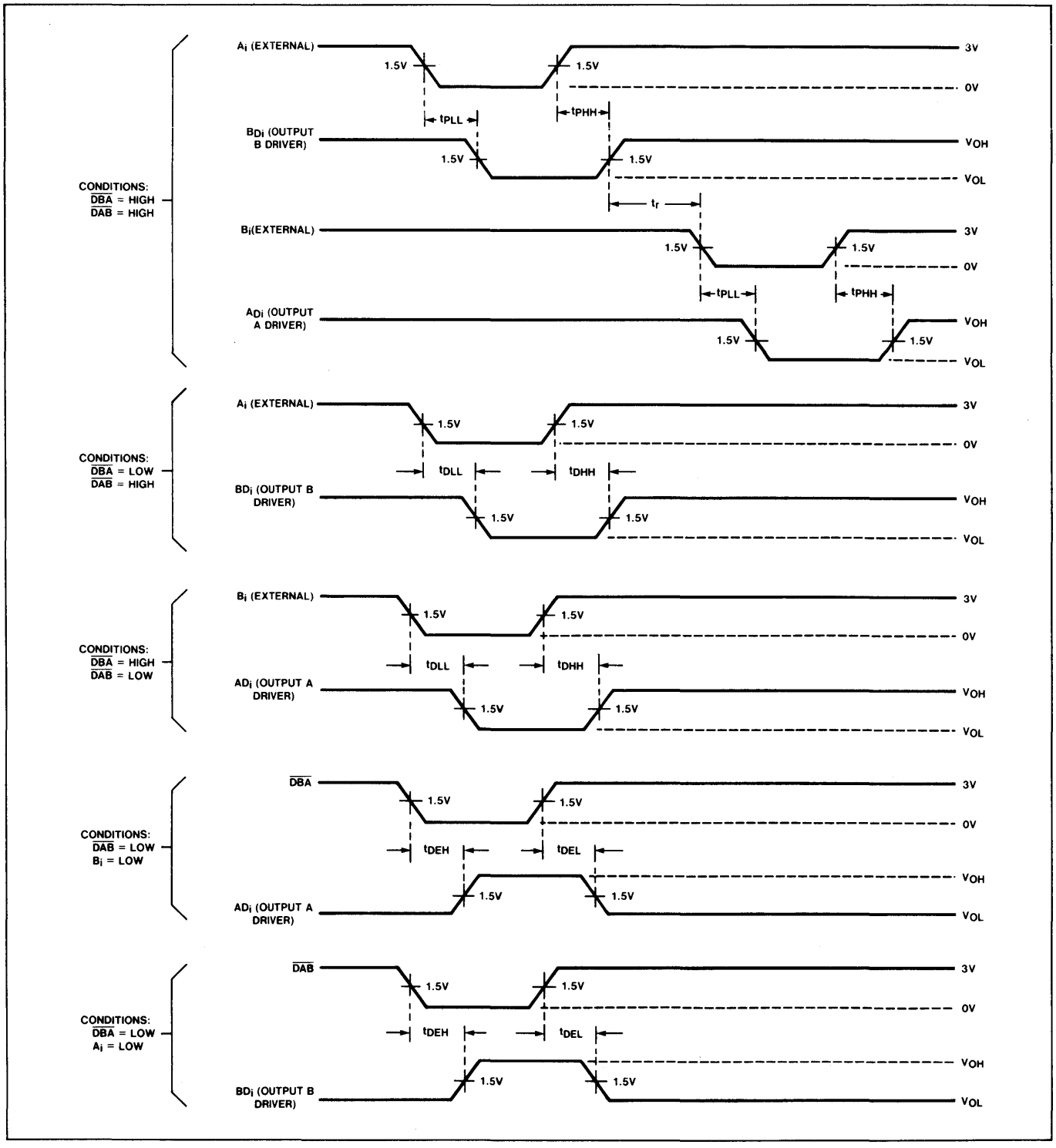
NOTE
C_L includes probe and jig capacitance

AC CHARACTERISTICS V_{CC} = 5V (±5%); T_A = 0°C to 70°C

PARAMETER	DESCRIPTION	FROM	TO	TEST CONDITIONS	LIMITS			UNITS
					Min	Typ	Max	
t_{PLL}	Propagation delay	Low A_i Low B_i	Low BD_i Low AD_i	$\overline{DBA}$ = $\overline{DAB}$ = High			30	ηsec
t_{PHH}	Propagation delay	High A_i High B_i	High BD_i High AD_i	$\overline{DBA}$ = $\overline{DAB}$ = High			30	ηsec
t_{DHH}	Propagation delay	High A_i	High BD_i	$\overline{DBA}$ = Low; $\overline{DAB}$ = High			25	ηsec
		High B_i	High AD_i	$\overline{DAB}$ = Low; $\overline{DBA}$ = High			25	ηsec
t_{DLL}	Propagation delay	Low A_i	Low BD_i	$\overline{DBA}$ = Low; $\overline{DAB}$ = High			25	ηsec
		Low B_i	Low AD_i	$\overline{DAB}$ = Low; $\overline{DBA}$ = High			25	ηsec
t_{DEH}	Propagation delay	Low $\overline{DBA}$	High AD_i	$\overline{DAB}$ = Low; B_i = Low			30	ηsec
t_{DEL}	Propagation delay	High $\overline{DBA}$	Low AD_i	$\overline{DAB}$ = Low; B_i = Low			30	ηsec
t_{DEH}	Propagation delay	Low $\overline{DAB}$	High BD_i	$\overline{DBA}$ = Low; A_i = Low			30	ηsec
t_{DEL}	Propagation delay	High $\overline{DAB}$	Low BD_i	$\overline{DBA}$ = Low; A_i = Low			30	ηsec
t_r	Recovery time (see timing diagram)	—	—	$\overline{DBA}$ = $\overline{DAB}$ = High		20		ηsec

NOTES
A_i = External signal AD_i = Output A driver B_i = External signal BD_i = Output B driver

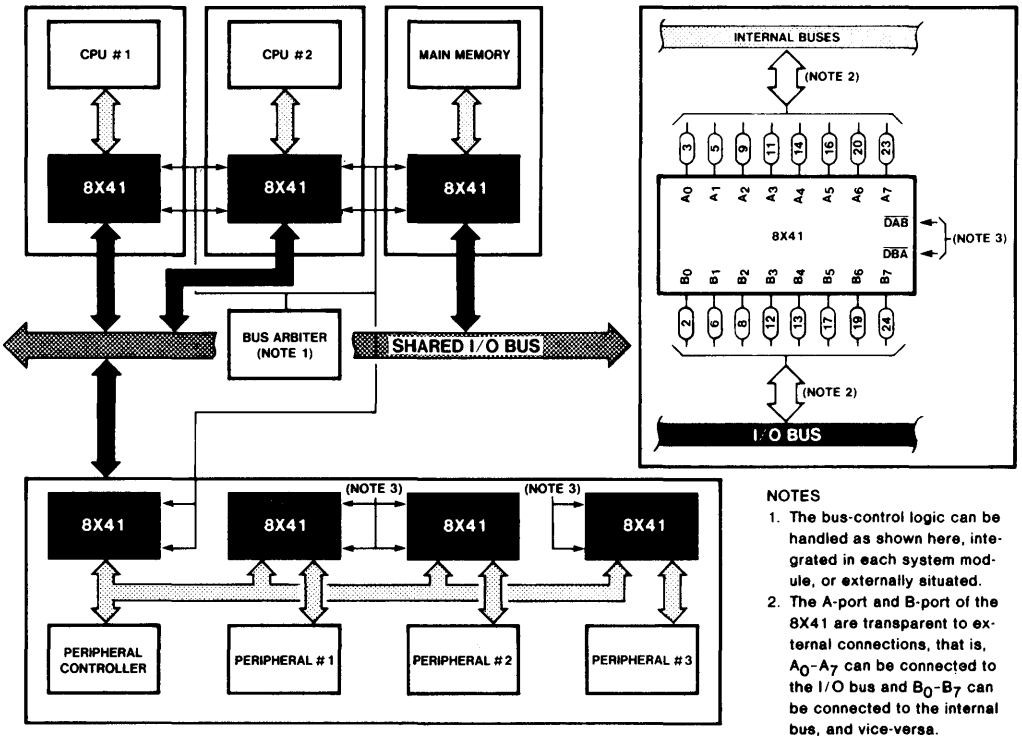
8X41 TIMING DIAGRAM



ORDERING INFORMATION

Order number: N8X41N
Packaging information: Refer to Signetics price list
Supply voltage: 5V (±5%)
Operating temperature range: 0°C to +70°C

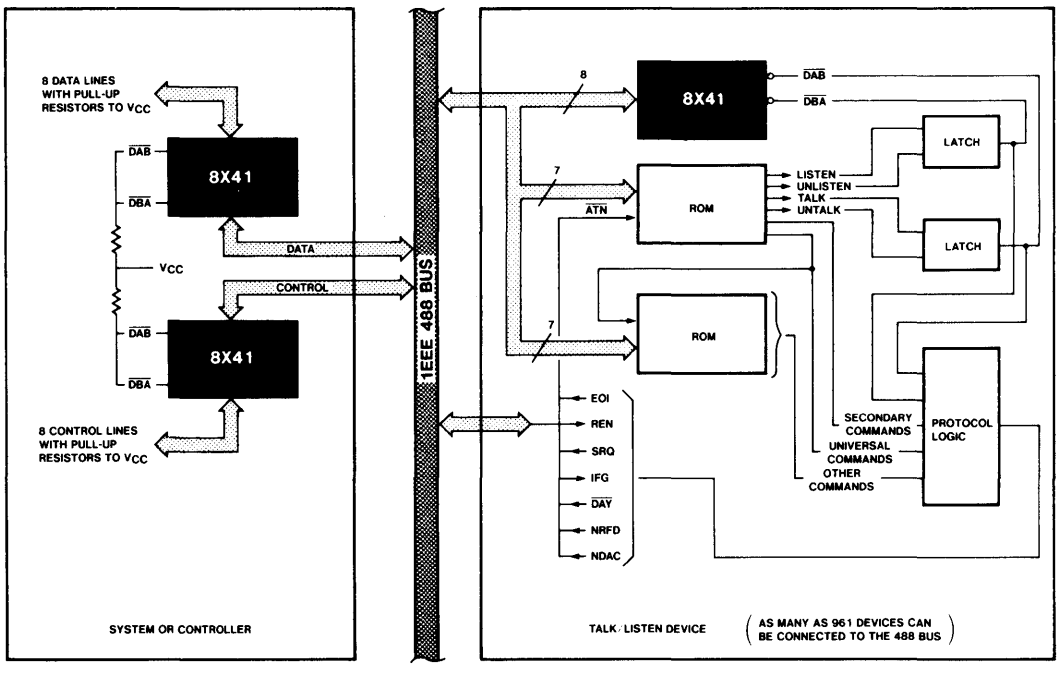
USING THE 8X41 IN A BUS-SHARED CONFIGURATION



NOTES

1. The bus-control logic can be handled as shown here, integrated in each system module, or externally situated.
2. The A-port and B-port of the 8X41 are transparent to external connections, that is, A_0-A_7 can be connected to the I/O bus and B_0-B_7 can be connected to the internal bus, and vice-versa.
3. Refer to tables in Figure 1 for data control capabilities.

INTERFACING 8X41 TO IEEE 488 BUS



8T58-I,XL

DESCRIPTION

The Bus Expander is specifically designed to increase the I/O capability of 8X300 systems previously limited by fanout considerations. The bus expander serves as a buffer between the 8X300 and blocks of I/O devices. Each bus expander can buffer a block of 16 I/O ports while only adding a single load to the 8X300.

FEATURES

- **15ns max propagation delay**
- **Bidirectional**
- **Three-state outputs on both ports**

FUNCTIONAL DESCRIPTION

The Bus Expander contains eight sets of non-inverting bidirectional tri-state drivers for the bus data bits, four non-inverting undirectional drivers for I/O port control, and necessary control logic. The control logic is required to maintain the proper directional transfer of bus data as dictated by the states of the I/O port control signals. A bus expander may be used on either left bank or right bank. Systems may be configured with I/O ports connected directly to the 8X300, as well as I/O ports connected through a bus expander.

Addition of bus expanders may impact system cycle time due to the added delay in the data path. For the purposes of calculating allowable cycle time as described in the 8X300 data sheet, the bus expander delays may be considered additive to the I/O port delays so that a buffered I/O port simply appears as a slower I/O port.

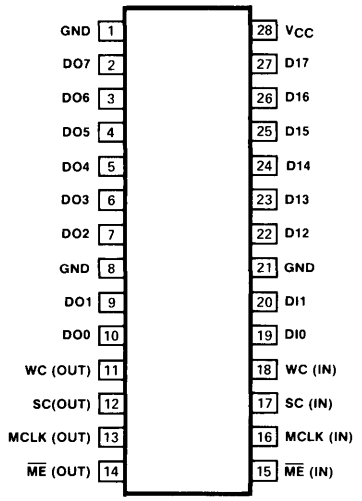
APPLICATIONS

The 8T39 Bus Expander is designed to be used with the 8X300 microprocessor to allow increased I/O capability in those systems previously limited by fanout considerations. Figure 1 shows a typical arrangement of the bus expander in an 8X300 system. Other I/O ports or working storage may be directly connected to the bus as shown.

The bus expander is not limited to use with the 8X300, but may be applied in any system which uses a combined address/data bus.

PIN CONFIGURATION

I,XL PACKAGE



μP Side

TRUTH TABLE

$\overline{ME}$	SC	WC	DATA TRANSFER DIRECTION	ADDRESS COMPARISON
L	L	L	DI Bus ← DO Bus	No
L	L	H	DI Bus → DO Bus	No
L	H	X	DI Bus → DO Bus	No
H	X	X	DI Bus → DO Bus	No

PIN DESIGNATION

PIN NO.	SYMBOL	NAME & FUNCTION	TYPE
2-7,9,10	DO0-DO7	I/O port data bus	Active low, three-state
11	WC(OUT)	Write command output	Active high
12	SC(OUT)	Select command output	Active high
13	MCLK(OUT)	Master clock input	Active high
14	$\overline{ME}$(OUT)	Master enable output	Active low
15	$\overline{ME}$(IN)	Master enable input	Active low
16	MCLK(IN)	Master clock input	Active high
17	SC(IN)	Select command output	Active high
18	WC(IN)	Write command output	Active high
19,20,22-27	DI0-DI7	Microprocessor data bus	Active low, three-state
1,8,21	GND	Ground	
28	V_{CC}	+5 volt supply	

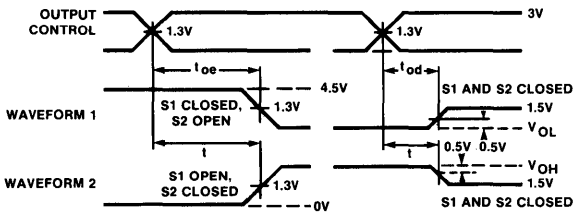
DC ELECTRICAL CHARACTERISTICS V_{CC} = 5V ±5%, 0°C ≤ T_A ≤ 70°C

PARAMETER		TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT
	Input voltage					V
V_{IL}	Low				.8	
V_{IH}	High		2.0			
V_{IC}	Clamp	-5mA at V_{CC} min			-1	
	Output voltage	V_{CC} = 4.75V				V
V_{OL}	Low	I_{OL} = 50mA			.55	
V_{OH}	High	I_{OH} = -3.2mA	2.4			
	Input current	V_{CC} = 5.25V				μA
I_{IL}	Low*	V_{IL} = .5V			-250	
I_{IH}	High*	V_{IH} = 5.25V		<10	100	
I_{OS}	Short circuit output current	V_{CC} = 4.75V	-40			mA
I_{CC}	Supply current	V_{CC} = 5.25V			200	mA

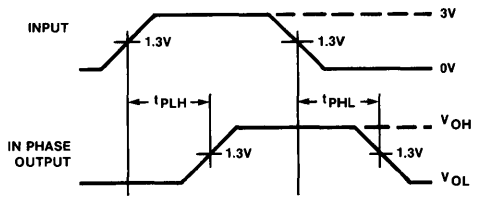
*Includes 3-State leakage.

VOLTAGE WAVEFORMS

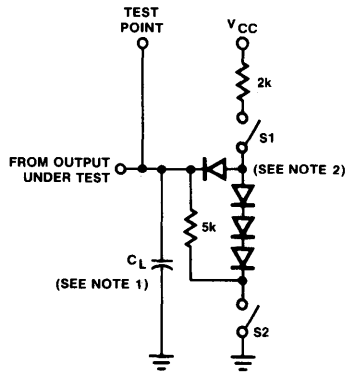
PROPAGATION DELAY TO THREE-STATE OUTPUTS



PROPAGATION DELAY TIMES



TEST LOAD CIRCUIT



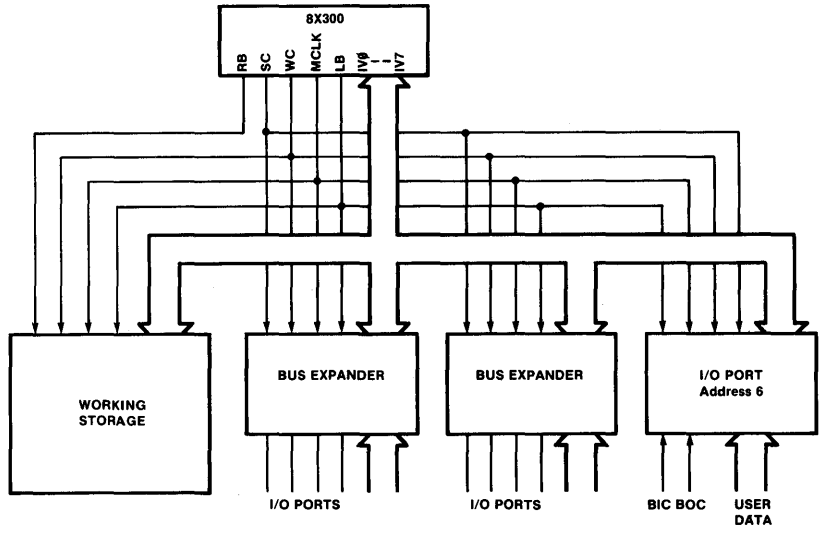
All resistors values are typical and in ohms.

NOTES

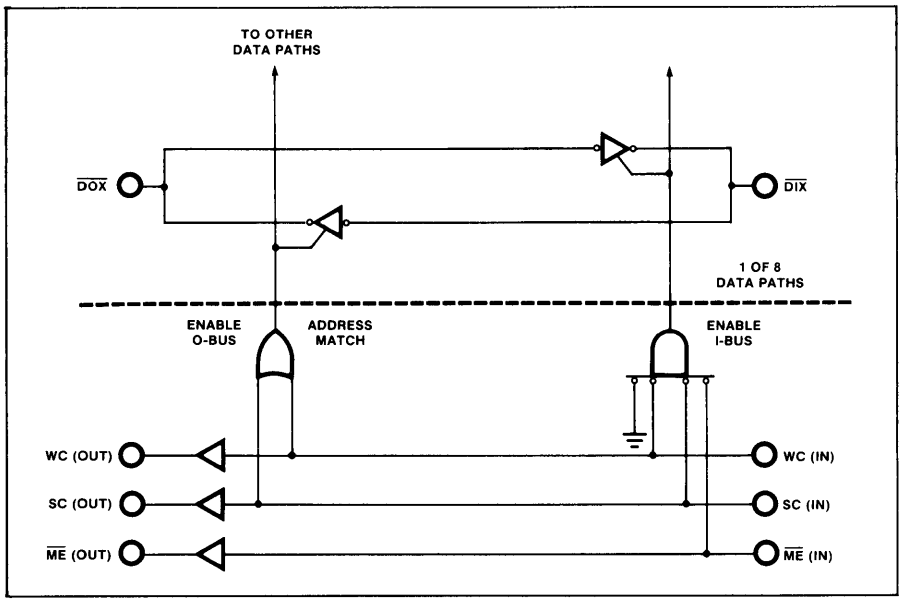
1. C_L includes probe and jig capacitance.
2. All diodes are 1N916 or 1N3064.

TYPICAL APPLICATION

USING 2 BUS EXPANDERS TO CREATE 33 I/O PORTS PLUS WORKING STORAGE



FUNCTIONAL BLOCK DIAGRAM



ABSOLUTE MAXIMUM RATINGS

PARAMETER		RATING	UNIT
V_{CC}	Power supply voltage	+7	Vdc
V_{IN}	Input voltage	+5.5	Vdc
V_O	Off-state output voltage	+5.5	Vdc
T_A	Operating temperature range	0 to +70	°C
T_{STG}	Storage temperature range	-65 to +150	°C

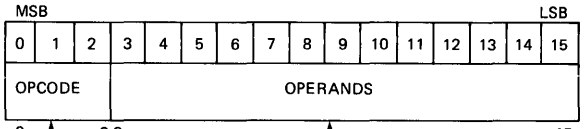
AC ELECTRICAL CHARACTERISTICS $V_{CC} = 5V \pm 5\%$, $0°C \leq T_A \leq 70°C$, $C_L = 300pF$

PARAMETER		TO	FROM	TEST CONDITIONS	LIMITS			UNIT
					Min	Typ	Max	
t_{pd}	Path delay Data	DOX DIX	DIX DOX				15	ns
t_{pd}	Control	$\overline{ME}$(OUT) MCLK(OUT) SC(OUT) WC(OUT)	$\overline{ME}$(IN) MCLK(IN) SC(IN) WC(IN)				15	ns
t_{oe}	Data Output Enable	DIX DOX	$\overline{ME}$(IN) SC(IN) WC(IN)		28		56	ns
t_{od}	Data Output Disable	DIX DOX	$\overline{ME}$(IN) SC(IN) WC(IN)		15			

APPENDIX B

INSTRUCTION FORMATS

GENERAL FORMAT



OPCODE	CMD. OPERATION	TYPE	FORMAT
0	MOVE	DATA TRANSFER	I II
1	ADD	ARITHMETIC/LOGIC	I II
2	AND		
3	XOR		
4	XEC	PROGRAM CONTROL	III IV
5	NZT		
6	XMIT	DATA TRANSFER	III IV
7	JMP	PROGRAM CONTROL	V

OCTAL FORMATS DESIGNATION

FORMAT I

	0 1 2	3 4 5 6 7	8 9 10	11 12 13 14 15
OPCODE / OPERANDS	OPCODE	OPERANDS		
BIT	0 1 2	3 4 5 6 7	8 9 10	11 12 13 14 15
$VALUE_8$	0 - 3	0 - 1 | 0 - 7	0 - 7	0 - 1 | 0 - 7
DEFINITION	MOVE ADD AND XOR	S	R	D

S = R (BIT 3 = Ø)
D = R (BIT 11 = Ø)
R/L = ROTATE

FORMAT II

	0 1 2	3 4 5 6 7	8 9 10	11 12 13 14 15
OPCODE / OPERANDS	OPCODE	OPERANDS		
BIT	0 1 2	3 4 5 6 7	8 9 10	11 12 13 14 15
$VALUE_8$	0 - 3	0 - 3 | 0 - 7	0 - 7	0 - 3 | 0 - 7
DEFINITION	MOVE ADD AND XOR	S	L	D

S = R OR I/0
D = R OR I/0
R/L = LENGTH

FORMAT III

	0 1 2	3 4 5 6 7	8 9 10 11	12 13 14 15
OPCODE / OPERANDS	OPCODE	OPERANDS		
BIT	0 1 2	3 4 5 6 7	8 9 10 11	12 13 14 15
$VALUE_8$	4 - 6	0 - 1 | 0 - 7	0 - 3 | 0 - 7	0 - 7
DEFINITION	XEC NZT XMIT	S*	J	

S = R
I = 8 BITS

TYPICAL INSTRUCTION: 605300
BIT PATTERN: 8 | 9
1 | 1

FORMAT IV

	0 1 2	3 4 5 6 7	8 9 10	11 12 13 14 15
OPCODE / OPERANDS	OPCODE	OPERANDS		
BIT	0 1 2	3 4 5 6 7	8 9 10	11 12 13 14 15
$VALUE_8$	4 - 6	2 - 3 | 0 - 7	0 - 7	0 - 3 | 0 - 7
DEFINITION	XEC NZT XMIT	S*	L	J

S = I/0
R/L = LENGTH
I = 5 BITS

TYPICAL INSTRUCTION: 627305
BIT PATTERN: 8 | 9 | 10
0 | 1 | 1

FORMAT V

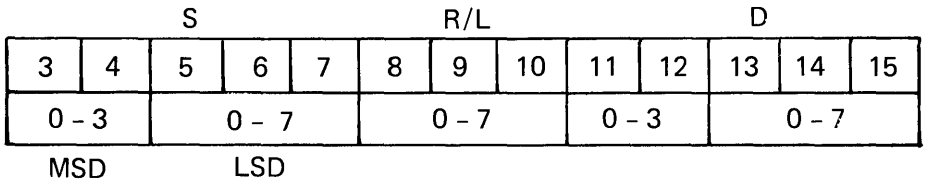
	0 1 2	3 4 5 6 7 8 9 10 11 12 13 14 15
OPCODE / OPERANDS	OPCODE	OPERANDS
BIT	0 1 2	3 4 5 6 7 8 9 10 11 12 13 14 15
$VALUE_8$	7	0-1 | 0 - 7 | 0 - 7 | 0 - 7 | 0 - 7
DEFINITION	JMP	A

A = 13 BITS

OPERANDS

- S = SOURCE (REG OR I/0)
- D = DESTINATION (REG OR I/0)
- R/L = ROTATE/LENGTH
  - R IF (REG → REG)
  - L IF (IF S OR D = I/0)
- I = LITERAL CONSTANT (IMMEDIATE)
- A = ADDRESS (PROGRAM STORAGE)

*D FOR XMIT INSTRUCTIONS



REGISTER SPECIFICATION

OCTAL VALUE	DEFINITION	
00	Auxiliary Register RØ. Also processed as second term during execution of ALU commands	
01	R1	WORKING REGISTERS
02	R2	
03	R3	
04	R4	
05	R5	
06	R6	
07	IVL	Pseudo-register. Specifies address selection to LEFT bank I/O. Can only be specified as a destination.
10	OVF	Overflow Register. Source specified only Carry stored in Bit 7 location
11	R11	WORKING REGISTER
12		UNASSIGNED
13		
14		
15		
16		
17	IVR	Pseudo-register. Specifies address selection to RIGHT bank I/O. Can only be specified as a destination.

IV BUS DATA SPECIFICATION

OCTAL VALUE	DEFINITION
2X	Designates I/O data on the LEFT bank previously selected by IVL pseudo-register.
3X	Designates I/O data on the RIGHT bank previously selected IVR pseudo-register.

where X = 0 to 7, the location of the least significant bit within the byte of selected I/O. The actual field to be processed is variable from 1 to 8 bits, depending upon the R/L operand.*

* R/L OPERAND

R = Right rotation of data specified in source field R number of places. Both Source and Destination fields must specify registers.

L = Length of data field (in bits) within the accessed byte to be manipulated.

a subsidiary of **U.S. Philips Corporation**

Signetics Corporation
811 East Arques Avenue
P.O. Box 409
Sunnyvale, California 94086
Telephone 408/739-7700

Printed in USA
10M1280
300177