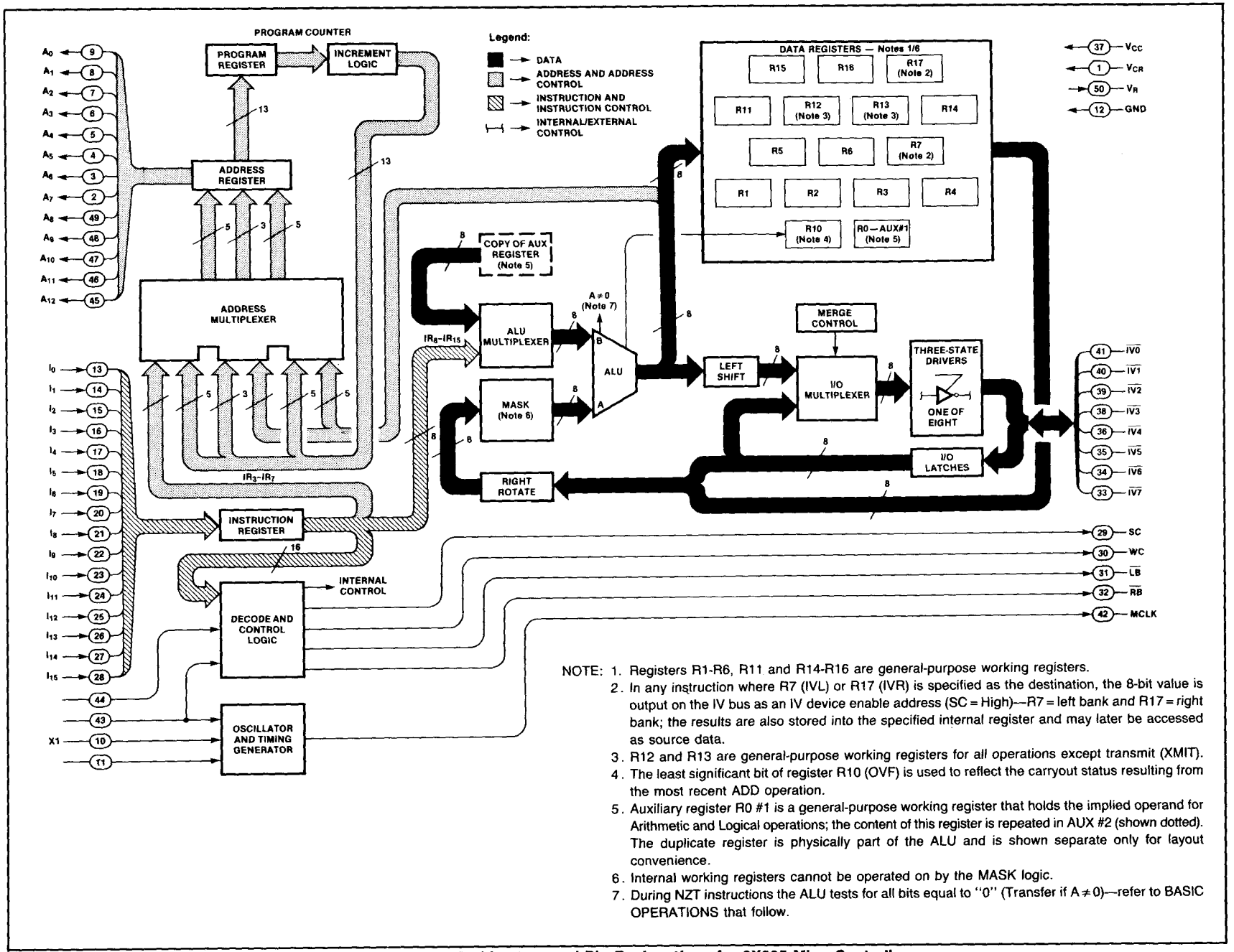
Signetics

8X305
March 1984

8X305 MicroController



NOTE:
1. Registers R1-R6, R11 and R14-R16 are general-purpose working registers.
2. In any instruction where R7 (IVL) or R17 (IVR) is specified as the destination, the 8-bit value is output on the IV bus as an IV device enable address (SC = High)—R7 = left bank and R17 = right bank; the results are also stored into the specified internal register and may later be accessed as source data.
3. R12 and R13 are general-purpose working registers for all operations except transmit (XMIT).
4. The least significant bit of register R10 (OVF) is used to reflect the carryout status resulting from the most recent ADD operation.
5. Auxiliary register R0 #1 is a general-purpose working register that holds the implied operand for Arithmetic and Logical operations; the content of this register is repeated in AUX #2 (shown dotted). The duplicate register is physically part of the ALU and is shown separate only for layout convenience.
6. Internal working registers cannot be operated on by the MASK logic.
7. During NZT instructions the ALU tests for all bits equal to "0" (Transfer if A≠0)—refer to BASIC OPERATIONS that follow.

Figure 1. Architecture and Pin Designations for 8X305 MicroController

MICROCONTROLLER 8X305

FEATURES

- **Fetch, Decode, and Execute a 16-bit Instruction in a minimum of 200 nanoseconds (one machine cycle)**
- **Bit-oriented Instruction set (addressable single-or-multiple bit subfields)**
- **Separate buses for Instruction, Instruction Address and Three-State I/O**
- **Thirteen 8-bit general-purpose working registers**
- **Source/destination architecture**
- **Bipolar low-power Schottky technology/TTL inputs and outputs**
- **On-chip oscillator and timing generation**
- **Single +5V supply**
- **0.9-in. 50-pin DIP**

PRODUCT DESCRIPTION

The Signetics 8X305 MicroController (Figure 1) is a high-speed bipolar microprocessor implemented with low-power Schottky technology. In a single chip, the 8X305 combines speed, flexibility, and a bit-oriented instruction set. These features and other basic characteristics of the chip combine to provide cost-effective solutions for a broad range of applications. The 8X305 is particularly useful in systems that require high-speed bit manipulations — sophisticated controllers, data communications, very fast interface control, and other applications of a similar nature.

The 8X305 can fetch, decode, and execute a 16-bit instruction word in a minimum of 200 nanoseconds. Within one instruction cycle, the 8-bit data-processing path can be programmed to rotate, mask, shift, and/or merge single or multiple bit subfields and, in addition, perform an ALU operation; in the same instruction, an external data field can be input, processed, and output to a specified destination — likewise, single or multiple bit data fields can be internally moved from a given source to a given destination. To summarize, fixed or variable-length data fields can be fetched, processed, operated on by the ALU, and moved to a different location — all in a timeframe of 200 nanoseconds. To interface with I/O and program memory, the 8X305 uses a 13-bit instruction address bus, a 16-bit instruction bus, an 8-bit bidirectional multiplexed I/O data/address bus and a 5-bit I/O control bus.

A wide selection of I/O devices, interface chips, and special-purpose parts are available for systems use. In most applications, the more powerful 8X305 is functionally interchangeable with its predecessor — the 8X300.

ASSOCIATED DOCUMENTATION

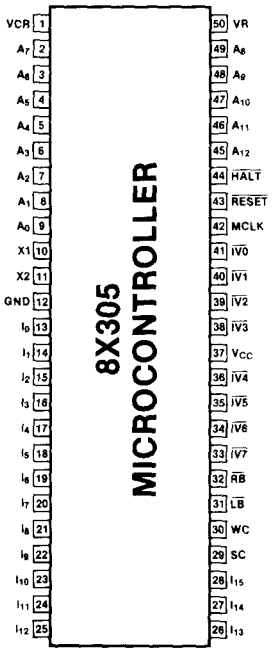
Other documents directly relating to *design* and *applications use* of the 8X305 MicroController are:

- Product Capabilities Manual
- 8X305 Users Manual

These documents and other current literature (Data Sheets, Product Bulletins, Applications Notes, etc.) are available at all Signetics Sales and Service Offices—see rear cover of this data sheet for the office in your locality.

PIN CONFIGURATION

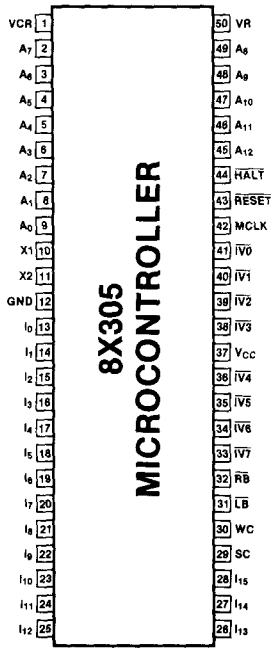
N, I PACKAGE



TOP VIEW

ORDER NUMBERS

8X305N, 8X305I
S8X305I/883B, S8X305I/883C



PIN NO.	IDENTIFIER	FUNCTION
1	VCR	Regulated voltage input from series-pass transistor (2N5320 or equivalent).
2-9, 45-49	$A_0 - A_{12}$	**Program Address Lines:** These active-high outputs permit direct addressing of up to 8192 words of program storage; A_{12} is least significant bit.
10, 11	X1, X2	Timing generator connections for a capacitor, a series resonant crystal, or an external clock source with complementary outputs.
12	GND	Ground.
13-28	$I_0 - I_{15}$	**Instruction Lines:** These active-high input lines receive 16-bit instructions from program storage; I_{15} is least significant bit.
29	SC	**Select Command:** When high (binary 1), an address is being output on pins $\overline{IV0}$ through $\overline{IV7}$.
30	WC	**Write Command:** When high (binary 1), data is being output on pins $\overline{IV0}$ through $\overline{IV7}$.
31	$\overline{LB}$	**Left Bank Control:** When low (binary 0), devices connected to the Left Bank are accessed. (Note. *Typically, the $\overline{LB}$ signal is tied to the $\overline{ME}$ input pin of I/O peripherals).*
32	RB	**Right Bank Control:** When low (binary 0), devices connected to the Right Bank are accessed (Note. *Typically, the $\overline{RB}$ signal is tied to the $\overline{ME}$ input pin of I/O peripherals).*
33-36, 38-41	$\overline{IV0}$-$\overline{IV7}$	**Interface Vector** (Input/Output Bus) — these bidirectional active-low three-state lines communicate data and/or addresses to I/O devices and memory locations. A low voltage level equals a binary "1"; $\overline{IV7}$ is Least Significant Bit.
37	V_{CC}	+5V power supply.
42	MCLK	**Master Clock:** This active-high output signal is used for clocking I/O devices and/or synchronization of external logic.
43	$\overline{RESET}$	When **$\overline{RESET}$** input is low (binary 0), the 8X305 is initialized — sets Program Counter/Address Register to zero and inhibits MCLK. For the period of time $\overline{RESET}$ is low, the Left Bank/Right Bank ($\overline{LB}$/$\overline{RB}$) signals are forced high asynchronously.
44	$\overline{HALT}$	When **$\overline{HALT}$** input is low (binary 0), internal operation of the 8X305 stops at the start of next instruction; MCLK is not inhibited nor is any internal register affected; however, both the Left Bank/Right Bank ($\overline{LB}$/$\overline{RB}$) signals are synchronously driven high during the first quarter of the instruction cycle time and remain high during the time HALT is low.
50	VR	Internally-generated reference output voltage for external series-pass regulator transistor.

Figure 2. Designations and Descriptions for Pins of 8X305 MicroController.

FUNCTIONAL OPERATION

Typical System Configuration

Although the system hookup shown in Figure 3 is of the simplest form, it provides a fundamental look at the 8X305 MicroController and peripheral relationships. As indicated, the 8X305 can directly address up to 8K words of program storage — either ROM or PROM. The user interface ($\overline{IV0}$ through $\overline{IV7}$) is capable of uniquely addressing 256 Input/Output locations and, with additional bank bits ($\overline{LB}$, $\overline{RB}$), this number is expanded to 512 — each bank comprising 256 addressable locations. The addressable locations of each bank can be used in a variety of ways; a simple method of implementation is shown in Figure 3. When $\overline{LB}$ is active low, the left bank is enabled and any one of 256 locations within the RAM memory can be accessed for input/output operations. A similar set of "enable/access" conditions are applicable to the right bank when $\overline{RB}$ is active low.

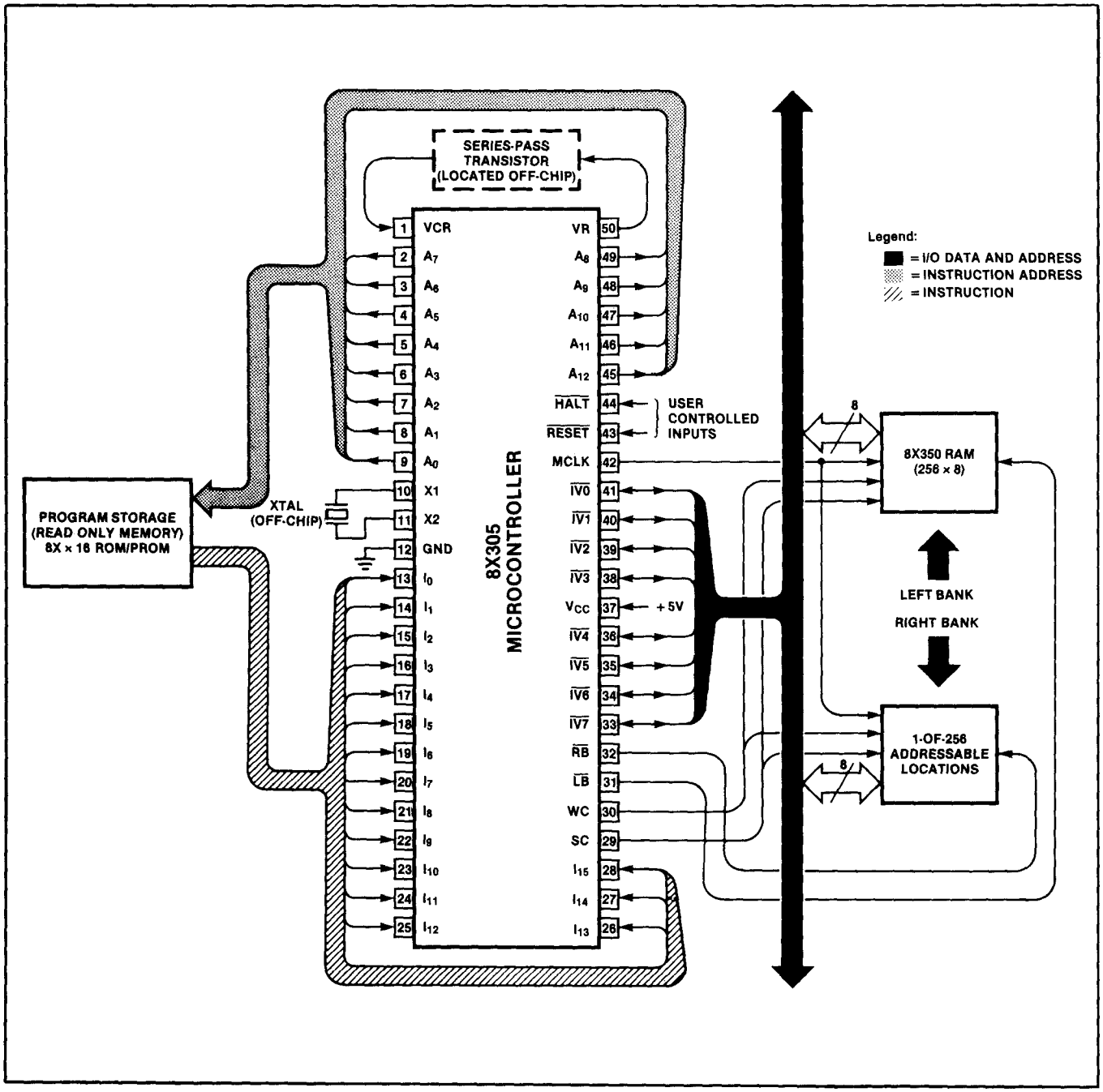


Figure 3. Typical 8X305 System Hookup

BASIC OPERATIONS OF 8X305

Refer to a later discussion of "Instruction Fields" for a detailed examination of all operand fields and subdivisions thereof—"S" (S_0, S_1), "D" (D_0, D_1), "R", "L", "J", and "A".

MOVE OPERATIONS

REGISTER-TO-REGISTER

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
R0-R17 as specified by "S" field of instruction.	Right rotate as specified by "R" field of instruction.	No operation.	No operation.	R0-R7, R11-R17 as specified by "D" field of instruction.

REGISTER-TO-$\overline{IV}$ BUS

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
R0-R17 as specified by "S" field of instruction.	No operation.	No operation.	Shift and merge as specified by "D_0" and "L" fields of instruction.	Variable length field of $\overline{IV}$ bus—Left Bank ($\overline{LB}$) or Right Bank ($\overline{RB}$) as specified by "D_0" and "D_1" fields of instruction.

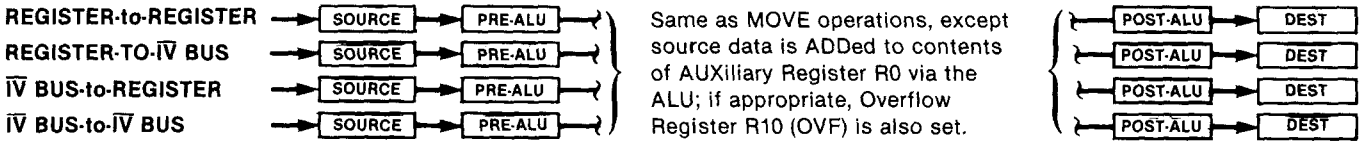
$\overline{IV}$ BUS-TO-REGISTER

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
Variable length field of $\overline{IV}$ bus—Left Bank ($\overline{LB}$) or Right Bank ($\overline{RB}$) as specified by "S_1" and "S_0" fields of instruction.	Right rotate and mask as specified by "S_0" and "L" fields of instruction.	No operation.	No operation.	R0-R7, R11-R17 as specified by "D" field of instruction.

$\overline{IV}$ BUS-TO-$\overline{IV}$ BUS

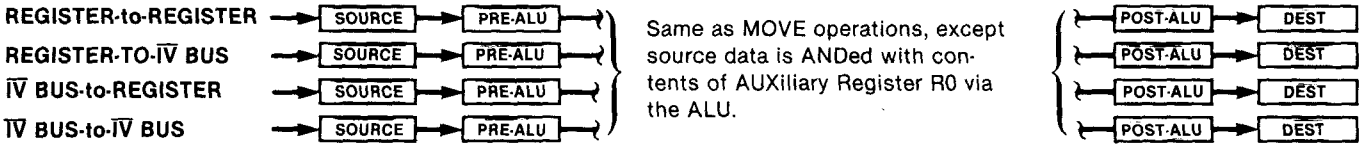
SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
Variable length field of $\overline{IV}$ bus—Left Bank ($\overline{LB}$) or Right Bank ($\overline{RB}$) as specified by "S_1" and "S_0" fields of instruction.	Right rotate and mask as specified by "S_0" and "L" fields of instruction.	No operation.	Shift and merge as specified by "D_0" and "L" fields of instruction.	Variable length field of $\overline{IV}$ bus—Left Bank ($\overline{LB}$) or Right Bank ($\overline{RB}$) as specified by "D_0" and "D_1" fields of instruction.

ADD OPERATIONS



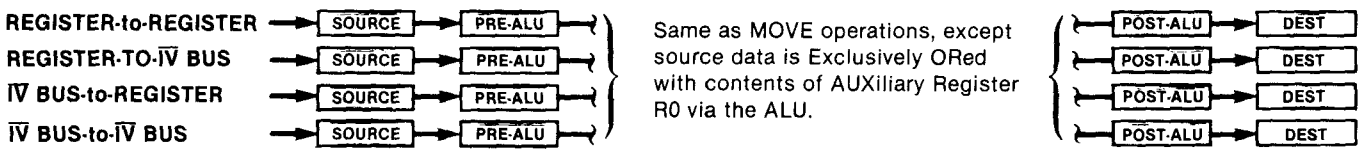
Same as MOVE operations, except source data is ADDed to contents of AUXiliary Register R0 via the ALU; if appropriate, Overflow Register R10 (OVF) is also set.

AND OPERATIONS



Same as MOVE operations, except source data is ANDed with contents of AUXiliary Register R0 via the ALU.

EXCLUSIVE OR (XOR) OPERATIONS



Same as MOVE operations, except source data is Exclusively ORed with contents of AUXiliary Register R0 via the ALU.

EXECUTE (XEC) OPERATIONS

XEC, REGISTER

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
R0-R17 as specified by "S" field of instruction.	No operation.	Add source data to 8-bit field specified by instruction literal ($0 \leq J \leq 377_8$).	No operation.	Replace 8 LSB of Address Register* with 8-bit sum from ALU. *PGM CTR unchanged.

XEC, $\overline{IV}$ BUS

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
Left or Right Bank of $\overline{IV}$ bus as specified by "S" field of instruction.	Rotate and mask as specified by "S_0" and "L" fields of instruction.	Add masked field of source data to 5-bit literal specified by "J" field of instruction ($0 \leq J \leq 37_8$).	No operation.	Replace 5 LSB of Address Register* with 5-bit sum from ALU. *PGM CTR unchanged.

NON-ZERO TRANSFER (NZT) OPERATIONS

NZT, REGISTER

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DESTINATION
R0-R17 as specified by "S" field of instruction.	No operation.	Test contents of source register for all zeroes.	If S = 0, increment PC by 1; if S≠0, replace 8 LSB of AR and PC with literal specified by "J" field of instruction.

NZT, $\overline{IV}$ BUS

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DESTINATION
Left or Right Bank of $\overline{IV}$ bus as specified by "S" field of instruction.	Rotate and mask as specified by "S_0" and "L" fields of instruction.	Test contents of masked field for all zeroes.	If S = 0, increment PC by 1; if S≠0, replace 5 LSB of AR and PC with literal specified by "J" field of instruction.

TRANSMIT (XMIT) OPERATIONS

XMIT, REGISTER

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
$0 \leq J \leq 377_8$ – value specified by "J" field of instruction.	No operation.	No operation.	No operation.	R0-R7, R11, R14-R17. Load 8-bit integer specified by "J" field into register specified by "D" field.

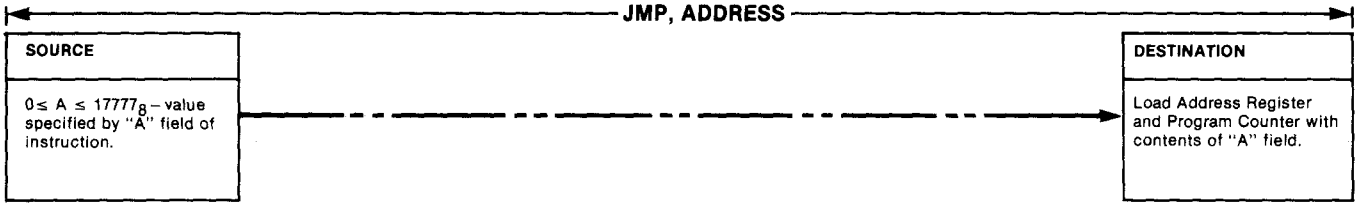
XMIT 8-BITS IMMEDIATE, $\overline{IV}$ BUS

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
$0 \leq J \leq 377_8$ – value specified by "J" field of instruction.	No operation.	No operation.	No operation.	Left (R12) or Right (R13) Bank of $\overline{IV}$ bus as specified by "D" field of instruction.

XMIT VARIABLE-BIT FIELD IMMEDIATE $\overline{IV}$ BUS

SOURCE	DATA PROCESSING (PRE-ALU)	ALU	DATA PROCESSING (POST-ALU)	DESTINATION
$0 \leq J \leq 37_8$ – value specified by "J" field of instruction.	No operation.	No operation.	Shift and merge source data as specified by "D_0" and "L" fields of instruction.	Left or Right Bank of $\overline{IV}$ bus as specified by "D" field of instruction.

JUMP (JMP) OPERATION



Program Storage Interface

As shown in Figure 3, program storage is connected to output address lines A_0 through A_{12} (A_{12} = LSB) and input instruction lines I_0 through I_{15}. An address output on A_0/A_{12} identifies one 16-bit instruction word in program storage. The instruction word is subsequently input on I_0/I_{15} and defines the MicroController operation which is to follow — one instruction word equals one completed operation. Any TTL-compatible memory can be used for program storage provided the worst-case access time is compatible with the instruction cycle time used for the application — see timing section for appropriate calculations.

I/O Interface and Control

An 8-bit bidirectional I/O bus, referred to as the Interface Vector ($\overline{IV}$) bus, provides a communication link between the MicroController and the two banks of I/O devices. The $\overline{LB}$ (Left Bank) and $\overline{RB}$ (Right Bank) control signals identify which bank is enabled; when both $\overline{LB}$ and $\overline{RB}$ are high (inactive), neither bank is enabled and the $\overline{IV}$ bus is inactive (three-state). A functional analysis of the Left and Right Bank signals is shown below:

$\overline{LB}$	$\overline{RB}$	FUNCTION
Low	Low	This state is not generated by the 8X305.
Low	High	Enable left bank devices.
High	Low	Enable right bank devices.
High	High	Disable all devices; $\overline{IV}$ bus is three-state.

Both data and I/O address information are multiplexed on the $\overline{IV}$ bus. The SC (Select Command) and WC (Write Command) signals distinguish between data and I/O address information as follows:

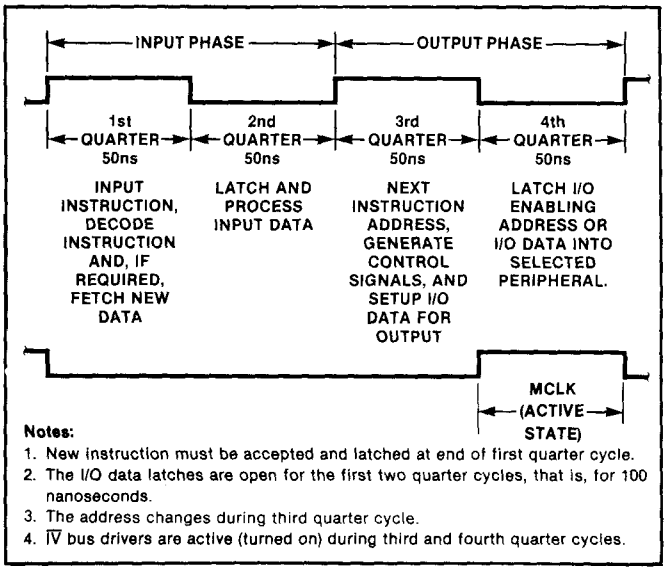
LB/RB	SC	WC	FUNCTION
High	Low	Low	The $\overline{IV}$ bus is three-state and not looking for input data.
Low	Low	Low	The $\overline{IV}$ bus is reading input data.
Low	Low	High	Data is being output.
Low	High	Low	Address is being output.
X	High	High	This condition is never generated.

Data Processing

Basically, the data processing path of the 8X305 consists of the Rotate/Mask logic, the Arithmetic Logic Unit (ALU), the Shift/ Merge functions, on-chip memory (sixteen 8-bit registers), and the bidirectional $\overline{IV}$ bus interface with its associated driver circuits and internal latches. The on-board memory and the $\overline{IV}$ bus are connected to both inputs and outputs of the ALU via internal 8-bit data paths — see Figure 1. Inputs to the ALU are preceded by right-rotate and data-mask functions; the ALU output is followed by the left-shift and merge operations. Depending on the desired operation, any one or all of the functions (Rotate/Mask/Shift/Merge) can operate on 8 bits of data in a single instruction cycle. For a summary of all data-processing capabilities, refer to BASIC OPERATIONS OF THE 8X305 described earlier in this data sheet.

Instruction Cycle

Each operation of the 8X305 is executed in a single instruction cycle. The instruction cycle is internally divided into four equal parts — each part being as short as 50 nanoseconds. Figure 4 shows the general functions that



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 100 nanoseconds.
3. The address changes during third quarter cycle.
4. $\overline{IV}$ bus drivers are active (turned on) during third and fourth quarter cycles.

Figure 4. Instruction Cycle and MCLK with: Crystal = 10MHz and Cycle Time = 200 nanoseconds.

occur during each quarter cycle; specifics regarding minimum/maximum timing and other critical values are described later in this data sheet. During the first quarter cycle, a new instruction from program storage is input via I_0-I_{15} and decoded. If an I/O operation is indicated, new data is fetched from a specified internal register or via the $\overline{IV}$ bus. At the end of the first quarter cycle, the new instruction is latched into the instruction register.

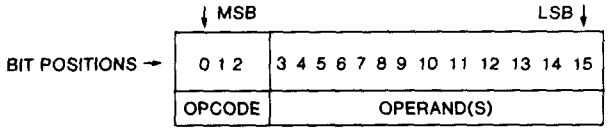
In the second quarter cycle, the I/O input data stabilizes and preliminary processing is completed; at the end of this quarter, the $\overline{IV}$ latches close and final processing can be accomplished, thus completing the input phase of the instruction cycle. During the third quarter cycle, the address for the next instruction is output to the instruction address bus, $\overline{IV}$ control signals are generated, and both data and destination are setup for the remainder of the output phase. During the fourth quarter cycle, a master clock signal (MCLK) generated by the 8X305 is used to latch either the I/O-enabling address or the I/O data into peripheral devices connected to the $\overline{IV}$ bus; MCLK can also be used to synchronize any external logic with timing circuits of the 8X305. 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.

INSTRUCTION SET

General Format and Operating Principles

The 16-bit instruction word (I_0 through I_{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 general format for each instruction word is as follows:



The 3-bit operation code (OPCODE) define any one of eight classes of instructions; variations within each class are specified by the remaining thirteen operand bits. The eight instruction classes can be separated into two control areas — *data* and *program;* general functions within these areas are:

- Data Control —
  - ADD, AND, XOR } Arithmetic and Logic Operations
  - MOVE, XMIT } Movement of Data and Constants
- Program Control
  - XEC, NZT, JMP } Branch or Test

Instruction Fields

As shown in Table 1, each instruction word consists of an operation code (OPCODE) field and from one to three operand fields. The possible operand fields are: Source (S), Destination (D), Rotate/Length (R/L), Literal (J), and Address (A). The OPCODE and operand fields are described in the paragraphs that follow the table.

Table 1. FUNCTIONAL DESCRIPTION OF INSTRUCTION SET

INSTRUCTION WORD	DESCRIPTION	STATE OF CONTROL SIGNAL DURING INSTRUCTION CYCLE — SEE FIGURE 4		
		CONTROL SIGNAL	INPUT PHASE	OUTPUT PHASE
CLASS = **MOVE** OPCODE = **0** OPERATION = **(S) → D**				
Register-to-Register 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S | R | D $S = 00_8$-17_8 $D = 00_8$-07_8, 11_8-17_8	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	L	H if $D = 07_8, 17_8$
		WC	L	L
		$\overline{LB}$	H	L if $D = 07_8$
		$\overline{RB}$	H	L if $D = 17_8$
Register-to-$\overline{IV}$ Bus (Note) 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S | L | D (D_1 : D_0) $S = 00_8$-17_8 $D = 20_8$-37_8	Move contents of internal register specified by the S-field to the $\overline{IV}$ bus. Before outputting on $\overline{IV}$ 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	L	L
		WC	L	H
		$\overline{LB}$	L if $D = 20_8$-27_8	L if $D = 20_8$-27_8
		$\overline{RB}$	L if $D = 30_8$-37_8	L if $D = 30_8$-37_8

Table 1. FUNCTIONAL DESCRIPTION OF INSTRUCTION SET (Continued)

INSTRUCTION WORD	DESCRIPTION	STATE OF CONTROL SIGNAL DURING INSTRUCTION CYCLE — SEE FIGURE 3		
		CONTROL SIGNAL	INPUT PHASE	OUTPUT PHASE
CLASS = MOVE OPCODE = 0 OPERATION = (S) → D				
$\overline{IV}$ Bus-to-Register (Note) 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S (S_1 : S_0) | L | D S = 20_8-37_8 D = 00_8-07_8, 11_8-17_8	Move right-rotated $\overline{IV}$ 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 zeros.	SC WC $\overline{LB}$ $\overline{RB}$	L L L if S = 20_8-27_8 L if S = 30_8-37_8	H if D = 07_8, 17_8 L L if D = 07_8 L if D = 17_8
$\overline{IV}$ Bus-to-$\overline{IV}$ Bus (Note) 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S (S_1 : S_0) | L | D (D_1 : D_0) S = 20_8-37_8 D = 20_8-37_8	Move right-rotated $\overline{IV}$ bus (source) data specified by the S-field to the I/O latches. Before outputting on $\overline{IV}$ 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 $\overline{LB}$ $\overline{RB}$	L L L if S = 20_8-27_8 L if S = 30_8-37_8	L H L if D = 20_8-27_8 L if D = 30_8-37_8
CLASS = ADD OPCODE = 1 OPERATION = (S) + (AUX) → D				
Same as MOVE instruction class	Same as MOVE instruction class except that contents of AUX (R0) register are ADDed to the source data. If there ia a "carry" from MSB, then R10 (OVF) = 1 (overflow), otherwise OVF = 0.	Same as MOVE instruction class		
CLASS = AND OPCODE = 2 OPERATION = (S) ∧ (AUX) → D				
Same as MOVE instruction class	Same as MOVE instruction class except that contents of AUX (R0) register are ANDed with source data.	Same as MOVE instruction class		
CLASS = XOR OPCODE = 3 OPERATION = (S) ⊕ (AUX) → D				
Same as MOVE instruction class	Same as MOVE instruction class except that contents of AUX (R0) register are Exclusively ORed with source data.	Same as MOVE instruction class		
CLASS = XEC OPCODE = 4 OPERATION = Refer to Description				
Register Immediate 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S | J S = 00_8-17_8 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 Address Register 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 $\overline{LB}$ $\overline{RB}$	L L H H	L L H H
$\overline{IV}$ Bus Immediate (Note) 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S (S_1 : S_0) | L | J S = 20_8-37_8 J = 00_8-37_8	Execute instruction at an address determined by replacing the low-order 5 bits of Address Register 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 $\overline{LB}$ $\overline{RB}$	L L L if S = 20_8-27_8 L if S = 30_8-37_8	L L H H
CLASS = NZT OPCODE = 5 OPERATION = Refer to Description				
Register Immediate 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 OPCODE | S | J S = 00_8-17_8 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 Address Register and Program Counter with "J", otherwise, increment PC.	SC WC $\overline{LB}$ $\overline{RB}$	L L H H	L L H H

Table 1. FUNCTIONAL DESCRIPTION OF INSTRUCTION SET (Concluded)

INSTRUCTION WORD	DESCRIPTION	STATE OF CONTROL SIGNAL DURING INSTRUCTION CYCLE — SEE FIGURE 3		
		CONTROL SIGNAL	INPUT PHASE	OUTPUT PHASE
CLASS = **NZT** OPCODE = **5** OPERATION = **Refer to Description**				
$\overline{IV}$ Bus Immediate (Note) 0 1 2 | 3 4 5 6 7 | 8 9 10 | 11 12 13 14 15 OPCODE | S | L | J S_1 ⋮ S_0 $S = 20_8$-37_8 $J = 00_8$-37_8	If right-rotated and masked IV bus is non-zero, transfer to address determined by replacing low-order 5 bits of Address Register and 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 zeros.)	SC	L	L
		WC	L	L
		$\overline{LB}$	L if $S = 20_8$-27_8	H
		$\overline{RB}$	L if $S = 30_8$-37_8	H
CLASS = **XMIT** OPCODE = **6** OPERATION = **J → D**				
XMIT, Register 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | D | J $D = 00_8$-06_8, 11_8, 14_8-16_8 $J = 000_8$-377_8	Store 8-bit value specified by "J" into register specified by "D".	SC	L	L
		WC	L	L
		$\overline{LB}$	H	H
		$\overline{RB}$	H	H
XMIT, $\overline{IV}$ Bus Address 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | D | J $D = 07_8$, 17_8 $J = 000_8$-377_8	Enable I/O device on the bank specified by "D", whose address is the 8-bit integer specified by "J". Address "J" is stored in register "D".	SC	L	H
		WC	L	L
		$\overline{LB}$	H	L if $D = 07_8$
		$\overline{RB}$	H	L if $D = 17_8$
XMIT 8 Bits Immediate, $\overline{IV}$ Bus (Note) 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | D | J $D = 12_8$-13_8 $J = 000_8$-377_8	Store value of 8-bit integer in the previously enabled I/O port, at the bank destination ($\overline{LB}$ or $\overline{RB}$) specified by "D". Contents of R12 or R13 remain unchanged.	SC	L	L
		WC	L	H
		$\overline{LB}$	H	L if $D = 12_8$
		$\overline{RB}$	H	L if $D = 13_8$
XMIT Variable Bit Field Immediate, $\overline{IV}$ Bus (Note) 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | D | L | J D_1 ⋮ D_0 $D = 20_8$-37_8 $J = 00_8$-37_8	Transmit Least Significant "L" bits of "J" field to "L-bit" field of $\overline{IV}$ bus specified by "D"; if "L" is greated than 5 bits, the MSB bits of destination field is filled with zeros.	SC	L	L
		WC	L	H
		$\overline{LB}$	L if $D = 20_8$-27_8	L if $D = 20_8$-27_8
		$\overline{RB}$	L if $D = 30_8$-37_8	L if $D = 30_8$-37_8
CLASS = **JMP** OPCODE = **7** OPERATION = **Refer to Description**				
Address Immediate 0 1 2 | 3 4 5 6 7 | 8 9 10 11 12 13 14 15 OPCODE | A $A = 00000_8$-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	L	L
		WC	L	L
		$\overline{LB}$	H	H
		$\overline{RB}$	H	H

Note:
S_0 specifies the LSB of rotated input data field
S_1 specifies the bank of $\overline{IV}$ bus from which source data will be input
D_0 specifies bit position in I/O device with which LSB of processed data will be aligned and
D_1 specifies the bank of $\overline{IV}$ bus which will be the destination.

Operations Code Field. The 3-bit OPCODE field specifies one of eight classes of 8X305 instructions; octal designations for this field and operands for each instruction class are shown in the preceding table.

Source (S) and Destination (D) Fields. The 5-bit "S" and "D" fields specify the source and destination, respectively, for whatever operation is defined by the OPeration CODE. The "S" and/or "D" fields can specify an internal 8X305 register or any one-to-eight bit field within an I/O device; octal values and source/destination field assignments for all internal registers are shown in Table 2.

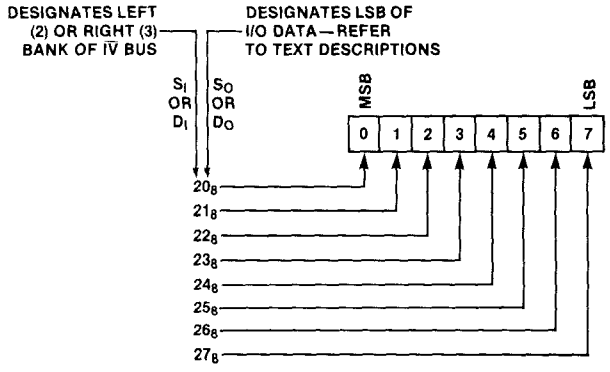
Table 2. OCTAL ADDRESSES AND SOURCE/DESTINATION FIELDS FOR 8X305 REGISTERS

ADDRESS	REGISTER DESIGNATION	SOURCE	DESTINATION	ADDRESS	REGISTER DESIGNATION	SOURCE	DESTINATION
00_8	R0 (AUX)—General purpose register	X	X	10_8	R10 (OVF—Overflow register)	X	
01_8	R1—General purpose register	X	X	11_8	R11—General purpose register	X	X
02_8	R2—General purpose register	X	X	12_8	R12—General purpose register (Note)	X	X
03_8	R3—General purpose register	X	X	13_8	R13—General purpose register (Note)	X	X
04_8	R4—General purpose register	X	X	14_8	R14—General purpose register	X	X
05_8	R5—General purpose register	X	X	15_8	R15—General purpose register	X	X
06_8	R6—General purpose register	X	X	16_8	R16—General purpose register	X	X
07_8	R7—Special purpose register (refer to next paragraph)	X	X	17_8	R17—Special purpose register (refer to next paragraph)	X	X

Note:
R12 and R13 function as general purpose working registers for all operations except transmit (XMIT). During a transmit instruction where R12 or R13 is the destination, the 8-bit "J" field is immediately transferred to the $\overline{IV}$ bus; for this operation, the contents of the designated register remain unchanged.

In instructions where $R7_8$ (IVL) or $R17_8$ (IVR) is specified as the destination, the 8-bit value is output on the $\overline{IV}$ bus as an I/O device address or memory location; register R7 selects the Left Bank and register R17 selects the Right Bank. The results are also stored into the specified internal register ($R7_8$ or $R17_8$) and may later be accessed as source data. When the $\overline{IV}$ bus is specified as a source and/or destination, the "S" and "D" fields are split into two parts, that is,

- Source (S) = S_1, S_0 and Destination (D) = D_1, D_0 where,
  S_0 specifies the LSB of rotated input data field
  S_1 specifies the bank of $\overline{IV}$ bus from which source data will be input
  D_0 specifies bit position in I/O device with which LSB of processed data will be aligned and
  D_1 specifies the bank of $\overline{IV}$ bus which will be the destination

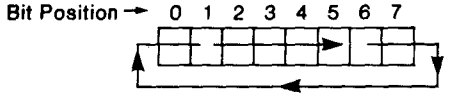


Notes:
1. The field length of 0-to-8 bits is specified by the "L" field.
2. For the Right Bank, 30_8-37_8 perform equivalent I/O functions.

Rotate (R) and Length (L) Field. The 3-bit R/L field performs one of two functions, specifying either the field length (L) for I/O operations or a right-rotate (R) for internal operations. 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 8 bits) is right-rotated during the "input phase" of the instruction cycle (Figure 4) and this function is always performed prior to any ALU operation. (*Note: The right-rotate function is implemented on the bus and not in the source register.*)

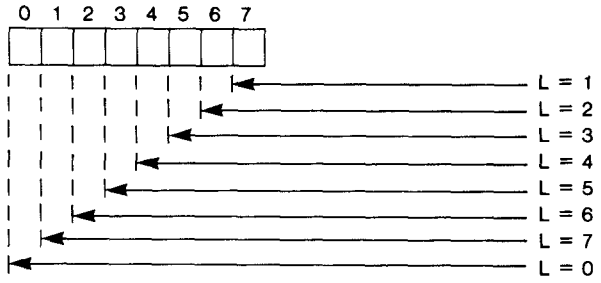
RIGHT-ROTATE FUNCTION



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 following diagram. If the source field specifies an $\overline{IV}$ address (20_8-37_8) and the destination field specifies an internal register (00_8-07_8, 11_8-17_8), the "L" field specifies the length of source data; the source data is formed by right-rotating the $\overline{IV}$ bus data according to the source address and then masking result as specified by the "L" field. If length is less than 8 bits, all remaining bits are set to zero prior to processing data in the ALU. If the source field

specifies an internal register (00_8–17_8) and the destination field specifies $\overline{IV}$ 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 and then masked to the required length — see $\overline{IV}$ DATA LENGTH SPECIFICATION. The destination data is merged with data in the I/O latches to finalize the $\overline{IV}$ bus data. Hence, a one-to-eight bit destination data field can be inserted into the existing 8-bit I/O port without modifying surrounding bits. If both the source and destination fields specify $\overline{IV}$ bus data (20_8–37_8), the "L" field specifies the length of both the source and destination data.

$\overline{IV}$ DATA LENGTH SPECIFICATION
(No Rotate Function Specified)



To form the source data, the $\overline{IV}$ bus input data is right-rotated according to the source address and then masked to the required length—see $\overline{IV}$ DATA LENGTH SPECIFICATION. If length is less than 8 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 and masked to the required length specification. The destination data is then merged into the $\overline{IV}$ bus data that was used to obtain the source; thus, if the source and destination addresses are on the same bank, the $\overline{IV}$ bus data written to the destination I/O Port appears unmodified, except for bits changed during the shift-and-mask operations. If the source and destination addresses refer to different banks, the destination I/O Port is changed to contain the contents of the source I/O Port 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" and "L" fields in the XEC, NZT, and XMIT instructions, based on the following conditions:

- 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 8X305 to directly branch to any of the 8192 locations in Program Storage memory.

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 one of the following 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 the Program Counter.
- For the XEC instruction, the Address Register is loaded with bits from the Program Counter modified as follows:

  XEC using $\overline{IV}$ 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 $\overline{IV}$ 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 the "J" field of 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 earlier, source/destination addresses are specified using a 5-bit code (00_8–37_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 $\overline{IV}$ bus operation is indicated — 2 specifying a Left-Bank ($\overline{LB}$) operation and 3 specifying a Right-Bank ($\overline{RB}$) operation. 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 $\overline{IV}$ bus data. Referring to Table 1, AUXiliary register R0 (00_8) is the implied source

of the second argument for the ADD, AND, and XOR operations. IVL register R7 and IVR register R17 (destination addresses 07_8 and 17_8, respectively) provide a means of routing enabling address information to I/O peripherals. With IVL or IVR specified as the destination address, data is placed on the $\overline{IV}$ 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 $\overline{IV}$ bus is to be considered as an I/O address. Since the contents of IVL and IVR are preserved, either register may later be accessed as a source of data.

Control outputs $\overline{LB}$ and $\overline{RB}$ are used to partition I/O bus devices into two fields of 256 addresses. With $\overline{LB}$ 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, $\overline{LB}$ is low if the destination address is 07_8 or 20_8–27_8, whereas $\overline{RB}$ is low if the destination address is 17_8 or 30_8–37_8. Each address field ($\overline{LB}$ and $\overline{RB}$) can have a different I/O device selected, that is, data can be transferred from a device in one bank to a device in the other in one instruction cycle.

DESIGN PARAMETERS

Hardware design of an 8X305-based system largely consists of the following operations:

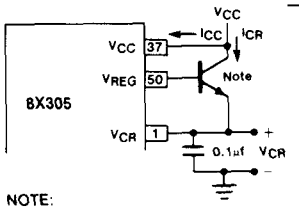
- Selecting and interfacing a Program Storage device — ROM, PROM, etc.
- Selecting and interfacing input/output devices — RAM, Ports, and other 8-bit addressable I/O devices.
- Choosing and implementing System Clock — Capacitor-Controlled, Crystal-Controlled, or Externally-Driven.
- Selection of an off-chip series-pass transistor.

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

- Ordering Information
- Voltage Regulator
- DC Characteristics
- AC Characteristics
- Timing Considerations
- Clock Considerations
- $\overline{HALT/RESET}$ Logic

VOLTAGE REGULATOR

All internal logic of the 8X305 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 8X305 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
VBE_{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

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

ABSOLUTE MAXIMUM RATINGS

Storage Temperature (T_{STG}) ratings are from -65 to $+150°C$

PIN	DESCRIPTION	RATING	UNIT	PIN	DESCRIPTION	RATING	UNIT
V_{CC} X1, X2	Supply voltage Crystal input voltage	+7.0 2.0	V V	All other pins	Logic input voltage	5.5	V

PARAMETER		TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT	COMMENTS
V_{CC}	Supply voltage		4.75	5.0	5.25	V	
V_{IH}	High level input voltage		0.6 2.0		2.0 5.5	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			V	
V_{OL}	Low level output voltage	V_{CC} = min; $I_{OL} = 6mA$ V_{CC} = min; $I_{OL} = 16mA$			0.55 0.55	V	A_0 through A_{12} All other outputs
V_{CR}	Regulator voltage	$V_{CC} = 5V$		3.1 2.9		V	$T_A = 0°C$ $T_A = 70°C$
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 $V_{IH} = 0.6V$ $V_{IH} = 4.5V$			4.0 50	mA μA	X1 and X2 All other pins
I_{IL}	Low-level input current	V_{CC} = max; $V_{IL} = 0.4V$			−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; (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			180 195	mA	$T_A = 70°C$ $T_A = 0°C$
I_{REG}	Regulator control	$V_{CC} = 5.0V$	−10		−25	mA	Max available base drive for series-pass transistor
I_{CR}	Regulator current	V_{CC} = max			200 230	mA	$T_A = 70°C$ $T_A = 0°C$

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

LOADING: (See test circuits)

PARAMETER (Note 1)		LIMITS (INSTRUCTION CYCLE TIME = 200ns) Min	Typ	Max	LIMITS (INSTRUCTION CYCLE TIME > 200ns) Min	Typ	Max	UNITS	COMMENTS
T_{PC}	Processor cycle time	200			200			ns	
T_{CP}	X1 clock period	100			100			ns	
T_{CH}	X1 clock high time	50			50			ns	
T_{CL}	X1 clock low time	50			50			ns	
T_{MCL}	MCLK low delay	15		40	15		40	ns	
T_W	MCLK pulse width	40		60	$T_{4Q} - 10$		$T_{4Q} + 10$	ns	Note 2
T_{MODO}	Output driver turn on time MCLK falling edge	125		145	$T_{1Q} + T_{2Q} + 25$		$T_{1Q} + T_{2Q} + 45$	ns	Note 9

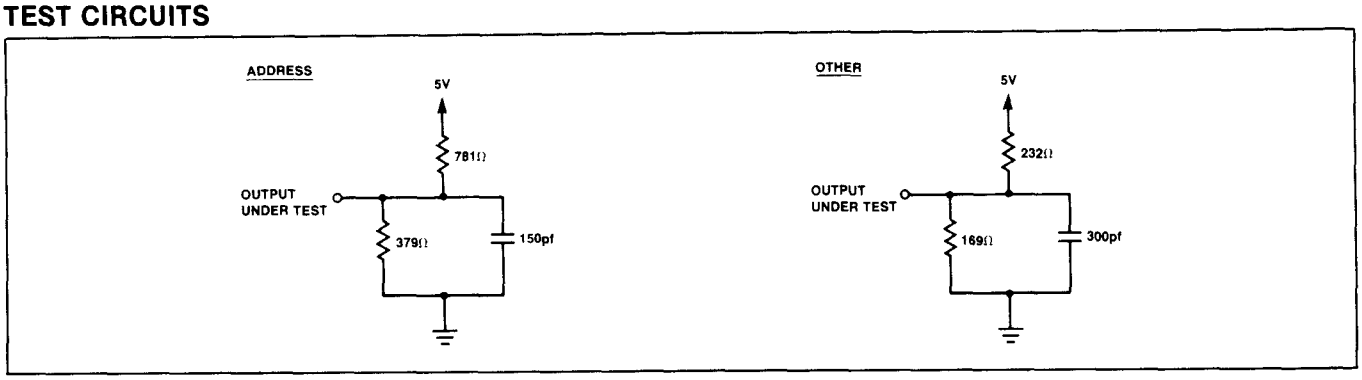
AC CHARACTERISTICS (Commercial Part) (Continued)

CONDITIONS: $4.75V \leq V_{CC} \leq 5.25V$; $0°C \leq T_A \leq 70°C$
LOADING: (See test circuits)

PARAMETER (Note 1)		LIMITS (INSTRUCTION CYCLE TIME = 200ns)			LIMITS (INSTRUCTION CYCLE TIME > 200ns)			UNITS	COMMENTS
		Min	Typ	Max	Min	Typ	Max		
T_{DI}	Output driver turn-on time (SC/WC rising edge)	20			20			ns	Note 10
T_{DD}	Input data to output data	85		105	85		105	ns	
T_{MHS}	MCLK falling edge to $\overline{HALT}$ falling edge			30			$T_{1Q} - 20$	ns	Note 2
T_{MHH}	$\overline{HALT}$ hold time (MCLK falling edge)	65			$T_{1Q} + 15$			ns	Note 2
T_{ACC}	Program storage access time			60				ns	
T_{IO}	I/O port output enable time ($\overline{LR}/\overline{RB}$ to valide $\overline{IV}$ data input)			30				ns	
T_{MAS}	MCLK falling edge to address stable			140			$T_{1Q} + T_{2Q} + 40$	ns	Notes 2, 3 & 4
T_{IA}	Instruction to address			140			$T_{2Q} + 90$	ns	Notes 2, 3 & 5
T_{IVA}	Input data to address			85			85	ns	Notes 3 & 6
T_{MIS}	MCLK falling edge to instruction stable			30			$T_{1Q} - 20$	ns	Notes 2 & 10
T_{MIH}	Instruction hold time (MCLK falling edge)	55			$T_{1Q} + 5$			ns	Notes 2 & 8
T_{MWH}	MCLK falling edge to SC/WC rising edge	105		125	$T_{1Q} + T_{2Q} + 5$		$T_{1Q} + T_{2Q} + 25$	ns	Note 2
T_{MWL}	MCLK falling edge to SC/WC falling edge	5		15	5		15	ns	
T_{MIBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Input phase)	10		25	10		25	ns	
T_{IIBS}	Instruction to $\overline{LB}/\overline{RB}$ (Input phase)			25			25	ns	
T_{MOBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Output phase)	115		145	$T_{1Q} + T_{2Q} + 15$		$T_{1Q} + T_{2Q} + 45$	ns	Note 2
T_{MIDS}	MCLK falling edge to input data stable			55			$T_{1Q} + T_{2Q} - 45$	ns	Note 2
T_{MIDH}	Input data hold time (MCLK falling edge)	115			$T_{1Q} + T_{2Q} + 15$			ns	Note 2
T_{MODH}	Output data hold time (MCLK falling edge)	11			11			ns	
T_{MODS}	Output data stable (MCLK falling edge)	130		150	$T_{1Q} + T_{2Q} + 30$		$T_{1Q} + T_{2Q} + 50$	ns	Note 2

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. T_{MAS} is obtained by forcing a valid instruction and an I/O bus input to occur earlier than the specified minimum set up time.
5. T_{IA} is obtained by forcing a valid instruction input to occur earlier than the minimum set up time.
6. T_{IVA} is obtained by forcing a valid I/O bus input to meet the minium set up time.
7. T_{MIS} represents the setup time required by internal latches of the 8X305. 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 (T_{IDS} and T_{MIDS}).
8. T_{MIH} represents the hold time required by internal latches of the 8X305. To generate proper $\overline{LB}/\overline{RB}$ signals, the instruction must be held valid until the address bus changes.
9. The minimum figure for these parameters represents the earliest time that I/O bus output drivers of the 8X305 will turn on.
10. This parameter represents the latest time that the output drivers of the input device should be turned off.

TEST CIRCUITS



ABSOLUTE MAXIMUM RATINGS

Storage Temperature (T_{STG}) ratings are from −65 to +150°C

PIN	DESCRIPTION	RATING	UNIT	PIN	DESCRIPTION	RATING	UNIT
V_{CC}	Supply voltage	+7.0	V	All other pins	Logic input pins	5.5	V
X1, X2	Crystal input voltage	2.0	V				

DC CHARACTERISTICS (Military Part)

4.5V ≤ V_{CC} ≤ 5.5V, −55°C ≤ T_C ≤ +125°C

PARAMETER		TEST CONDITIONS	LIMITS Min	LIMITS Typ	LIMITS Max	UNIT	COMMENTS
V_{CC}	Supply voltage		4.5	5.0	5.5	V	
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			V	
V_{OL}	Low level output voltage	V_{CC} = min; I_{OL} = 6mA V_{CC} = min; I_{OL} = 16mA			0.55 0.55	V	A_0 through A_{12} All other outputs
V_{CR}	Regulator voltage	V_{CC} = 5V		3.5 3.1 2.6		V	T_C = −55°C T_C = 0°C T_C = 125°C
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			4.0 50	mA μA	X1 and X2 All other pins
I_{IL}	Low-level input current	V_{CC} = max; V_{IL} = 0.4V			−3 −0.3 −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; (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			175 205	mA	T_C = 125°C T_C = −55°C
I_{REG}	Regulator control	V_{CC} = 5.0V	−10		−25	mA	Max available base drive for series-pass transistor
I_{CR}	Regulator current	V_{CC} = max			180 260	mA	T_C = 125°C T_C = −55°C

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

AC CHARACTERISTICS (Military Part)

CONDITIONS: $4.5V \leq V_{CC} \leq 5.5V$; $-55°C \leq T_C \leq 125°C$
LOADING: (See test circuits)

PARAMETER (Note 1)		LIMITS (INSTRUCTION CYCLE TIME = 250ns)			LIMITS (INSTRUCTION CYCLE TIME > 250ns)			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_{MCL}	MCLK low delay	15		40	15		40	ns	
T_W	MCLK pulse width	47		72	$T_{4Q} - 15$		$T_{4Q} + 10$	ns	Note 2
T_{MODO}	Output driver turn-on time (MCLK falling edge)	145		175	$T_{1Q} + T_{2Q} + 20$		$T_{1Q} + T_{2Q} + 50$	ns	Note 9
T_{DI}	Output driver turn-on time (SC/WC rising edge)	20			20			ns	Note 10
T_{DD}	Input data to output data	80		115	80		115	ns	
T_{MHS}	MCLK falling edge to $\overline{HALT}$ falling edge			40			$T_{1Q} - 22$	ns	Note 2
T_{MHH}	$\overline{HALT}$ hold time (MCLK falling edge)	80			$T_{1Q} + 18$			ns	Note 2
T_{ACC}	Program storage access time			90				ns	
T_{IO}	I/O port output enable time ($\overline{LB}/\overline{RB}$ to valid $\overline{IV}$ data input)			40				ns	
T_{MAS}	MCLK falling edge to address stable			160			$T_{1Q} + T_{2Q} + 35$	ns	Notes 2, 3 & 4
T_{IA}	Instruction to address			160			$T_{2Q} + 98$	ns	Notes 2, 3 & 5
T_{IVA}	Input data to address			90			90	ns	Notes 3 & 6
T_{MIS}	MCLK falling edge to instruction stable			40			$T_{1Q} - 22$	ns	Notes 2 & 10
T_{MIH}	Instruction hold time (MCLK falling edge)	70			$T_{1Q} + 8$			ns	Notes 2 & 8
T_{MWH}	MCLK falling edge to SC/WC rising edge	127		154	$T_{1Q} + T_{2Q} + 2$		$T_{1Q} + T_{2Q} + 29$	ns	Note 2
T_{MWL}	MCLK falling edge to SC/WC falling edge	5		25	5		25	ns	
T_{MIBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Input phase)	10		35	10		35	ns	
T_{IIBS}	Instruction to $\overline{LB}/\overline{RB}$ (Input phase)			30			30	ns	
T_{MOBS}	MCLK falling edge to $\overline{LB}/\overline{RB}$ (Output phase)	140		170	$T_{1Q} + T_{2Q} + 15$		$T_{1Q} + T_{2Q} + 45$	ns	Note 2
T_{MIDS}	MCLK faling edge to input data stable			75			$T_{1Q} + T_{2Q} - 50$	ns	Note 2
T_{MIDH}	Input data hold time (MCLK falling edge)	140			$T_{1Q} + T_{2Q} + 15$			ns	Note 2
T_{MODH}	Output data hold time (MCLK falling edge)	11			11			ns	
T_{MODS}	Output data stable (MCLK falling edge)	150		180	$T_{1Q} + T_{2Q} + 25$		$T_{1Q} + T_{2Q} + 55$	ns	Note 2

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. T_{MAS} is obtained by forcing a valid instruction and an I/O bus input to occur earlier than the specified minimum set up time.
5. T_{IA} is obtained by forcing a valid instruction input to occur earlier than the minimum set up time.
6. T_{IVA} is obtained by forcing a valid I/O bus input to meet the minium set up time.
7. T_{MIS} represents the setup time required by internal latches of the 8X305. 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 (T_{IDS} and T_{MIDS}).
8. T_{MIH} represents the hold time required by internal latches of the 8X305. To generate proper $\overline{LB}/\overline{RB}$ signals, the instruction must be held valid until the address bus changes.
9. The minimum figure for these parameters represents the earliest time that I/O bus output drivers of the 8X305 will turn on.
10. This parameter represents the latest time that the output drivers of the input device should be turned off.

TIMING CONSIDERATIONS (Commercial Part)

As shown in the AC CHARACTERISTICS table for the commercial part, the minimum instruction cycle time is 200 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 200 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 200 nanoseconds, 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 8X305 TIMING DIAGRAM. As the time interval for each instruction cycle increases (becomes greater than 200 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 8X305 are normally measured with reference to MCLK.

System determinants for the instruction cycle time are:

- Propagation delays within the 8X305
- 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) ≤ $\overline{IV}$ data set up time (Figure 5a).

Condition 2 — Program storage access time (TACC) plus instruction to $\overline{LB}/\overline{RB}$ (input phase) plus I/O port access time (TIO) plus $\overline{IV}$ data (input phase) to address ≤ instruction cycle time (Figure 5b).

Condition 3 — Program storage access time plus instruction to address ≤ instruction cycle time (Figure 5c).

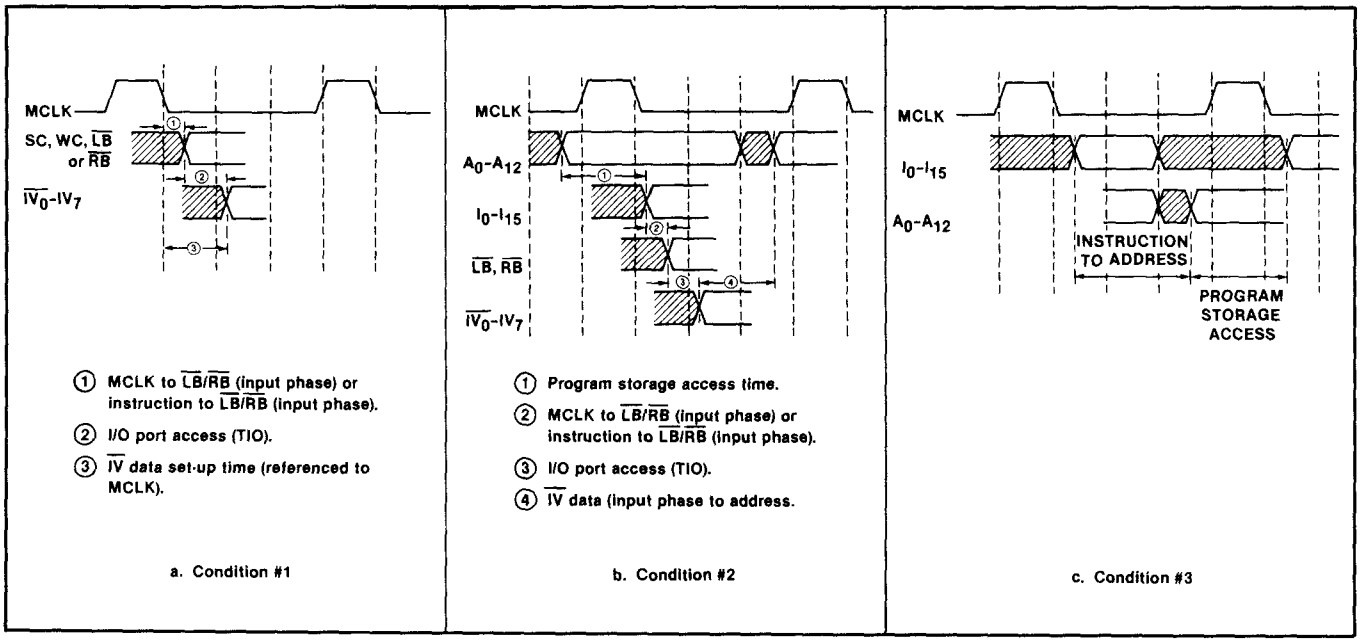
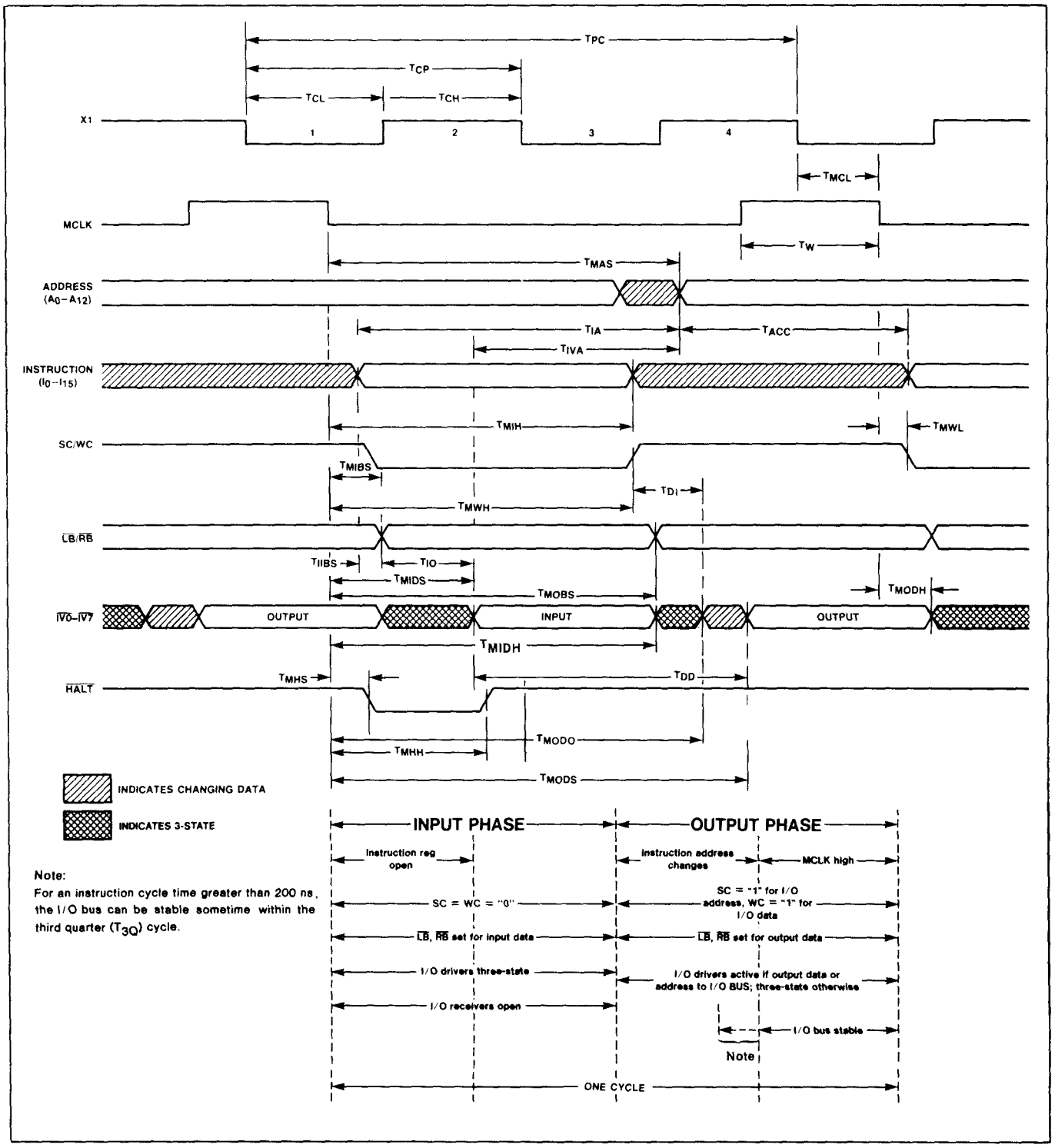


Figure 5. Constraints of 8X305 Instruction Cycle Time

8X305 TIMING DIAGRAM



Note:
For an instruction cycle time greater than 200 ns, the I/O bus can be stable sometime within the third quarter (T_{3Q}) cycle.

From condition #1 and with an instruction cycle time of 200ns, 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, $TIO \leq 30ns$

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

$$TMIBS + TIO \leq TMIDS$$
thus, $25ns + 30ns \leq T_{1Q} + T_{2Q} - 45$
$25ns + 30ns \leq 1/2$ cycle $- 45$ therefore,

the worst-case instruction cycle time is 200ns. With subject parameters referenced to X1, the same calculations are valid:

$$TIBS + TIO + TIDS \leq 1/2 \text{ cycle}$$
thus, $45ns + 30ns + 25ns \leq 1/2$ cycle therefore,

the worst-case instruction cycle time is again 200ns. From condition #2 and with an instruction cycle time of 200ns, the program storage access time can be calculated:

$$TACC + TIIBS + TIO + TIVA \leq 200ns$$
transposing, $TACC \leq 200ns - TIIBS - TIO - TIVA$
substituting, $TACC \leq 200ns - 25ns - 30ns - 85ns$
thus, $TACC \leq 60ns$ hence, for an instruction cycle time of 200ns, a program storage access time of 60ns 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, $TACC \leq 200ns - 140ns$
and, $TACC \leq 60ns$, confirming that a program storage access time of 60ns is satisfactory.

For an instruction cycle time of 200ns and a program storage access time of 60ns (Condition #2/Figure 5b), the instruction should be valid at the falling edge of MCLK. This relationship can be derived by the following equation:

$$200ns - TMAS - TACC$$
$$= 200ns - 140ns - 60ns$$
$$= 0ns$$

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 at the falling edge of MCLK and adding the instruction-to-$\overline{LB}/\overline{RB}$ delay (TIIBS = 25ns), the $\overline{LB}/\overline{RB}$ signal will be valid 25ns after the falling edge of MCLK. With a fast program storage memory and with a valid instruction before the falling edge of MCLK — the $\overline{LB}/\overline{RB}$ signal will, due to the TMIBS delay, still be valid 25ns after the falling edge of MCLK. Using a worst-case instruction cycle time of 200ns, 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 55ns (TAS) after the beginning of the third quarter cycle — no matter how early the $\overline{IV}$ data input is valid.

CLOCK CONSIDERATIONS

The on-chip oscillator and timing-generation circuits of the 8X305 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 8X305 be driven from a 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	300ns
200	500ns
500	1.1μs
1000	2.0μs

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 8X305 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:

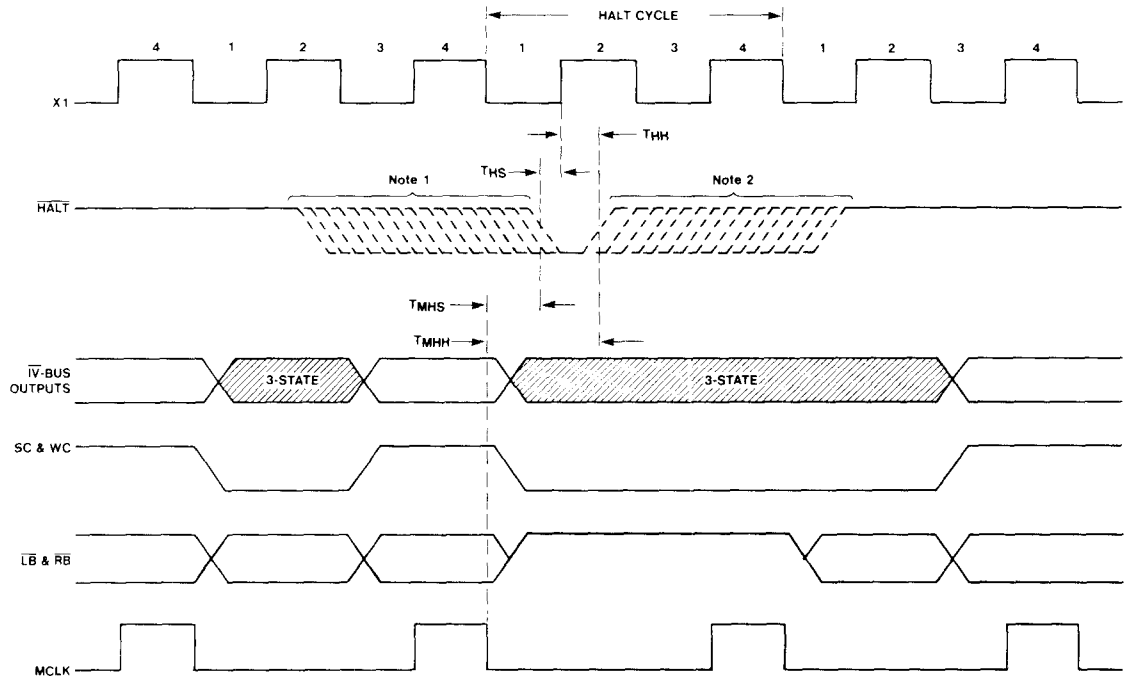
Type — Fundamental mode, series resonant
Impedance at Fundamental — 35 ohms maximum
Impedance at Harmonics and Spurs — 50 ohms minimum

The resonant frequency (f_o) of the crystal is related to the desired cycle time (T) by the equation: $f_o = 2/T$; thus, for a cycle time of 200 nanoseconds, $f_o = 10MHz$.

$\overline{HALT}$ Logic

The $\overline{HALT}$ signal is sampled via internal chip logic at the end of the first internal quarter of each instruction cycle. If, when sampled, the $\overline{HALT}$ 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 8X305. As long as the $\overline{HALT}$ line is active-low, the SC and WC lines are low (inactive), the Left Bank ($\overline{LB}$)/Right Bank ($\overline{RB}$) signals are high (inactive), and the $\overline{IV}$ bus remains in the three-state mode of operation. Normal operation resumes at the next cycle in which $\overline{HALT}$ is high when sampled — see $\overline{HALT}$ TIMING DIAGRAM.

HALT TIMING DIAGRAM



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)

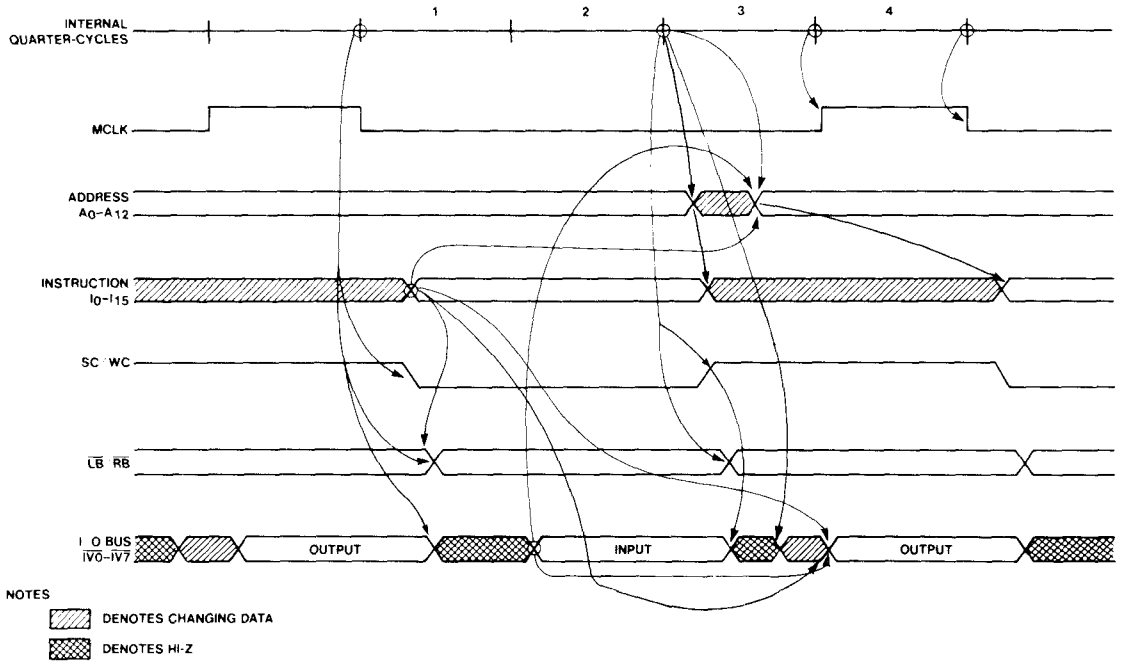


Figure 7. Timing Relationships of 8X305 I/O Signals

Using an External Clock. The 8X305 can be synchronized with an external clock by simply connecting appropriate drive circuits to the X1/X2 inputs. Figure 8 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 9.

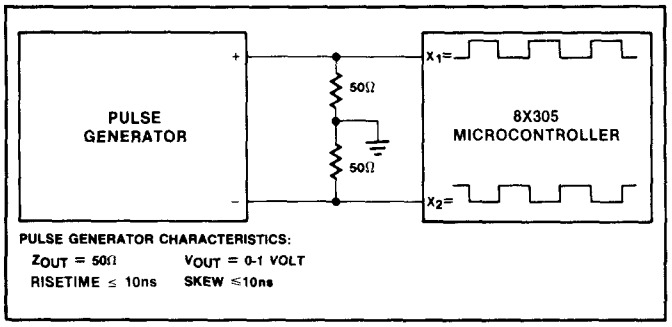


Figure 8. Clocking with a Pulse Generator

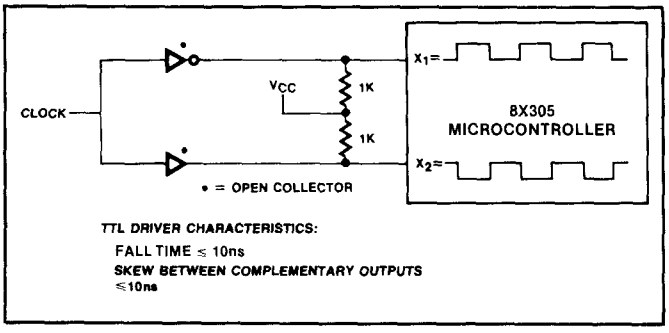


Figure 9. Clocking with TTL

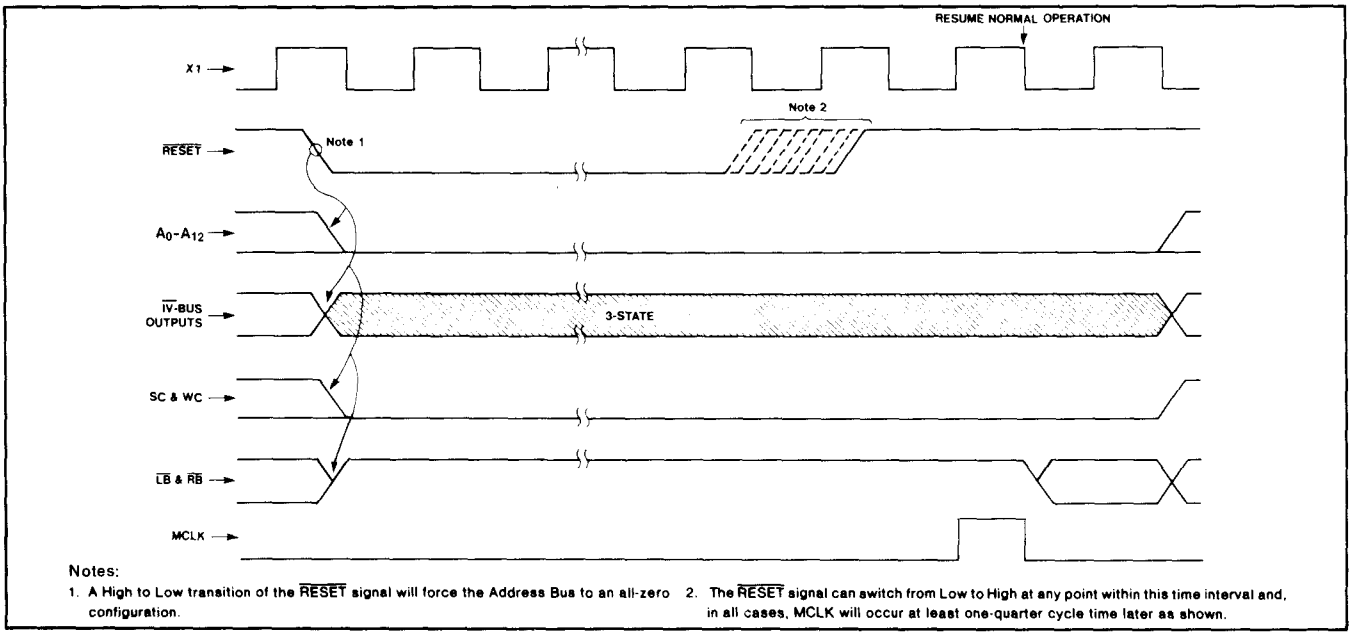
RESET Logic

RESET (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, RESET must 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 RESET TIMING DIAGRAM, these events are:

- The Program Counter and Address Register are set to address zero and remain in that state as long as the 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 condition as long as the RESET line is low.
- The Select Command and Write Command signals are driven low and remain low as long as the RESET line is low.
- The Left Bank/Right Bank (LB/RB) signals are forced high asynchronously for the period in which the RESET line is low.

During the time RESET is active-low, MCLK is inhibited; moreover, if the RESET line is driven low during the last two quarter cycles, MCLK may be shortened for that particular machine cycle. When RESET line is driven high (inactive)—one quarter to one full instruction cycle later, MCLK appears just before normal operation is resumed. The RESET/MCLK relationship is clearly shown by "B" in the timing diagram. As long as the RESET line is active-low, the HALT signal (described next) is not sampled by internal logic of the 8X305.

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.

signetics
a subsidiary of U.S. Philips Corporation

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

98-3001-400

Printed in U.S.A. BA/CRCR/10M0384