DESCRIPTION

The Signetics 8X300 Interpreter is a monolithic, high-speed micro-processor implemented with bipolar Schottky technology. As the central processing unit, CPU, it allows 16-bit instructions to be fetched, decoded and executed in 300 nanoseconds. A 300 nanosecond instruction cycle requires maximum memory access of 85 nanoseconds, and maximum I/O device access of 40 nanoseconds.

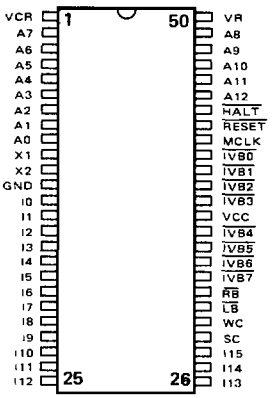
Interpreter instructions operate on 8-bit, parallel data. Logic is distributed along the data path within the Interpreter. Input data can be rotated and masked before being subject to an arithmetic or logical operation; and output data can be shifted and merged with the input data, before being output to external logic. This allows 1- to 8-bit I/O and data memory fields to be accessed and processed in a single instruction cycle.

FEATURES

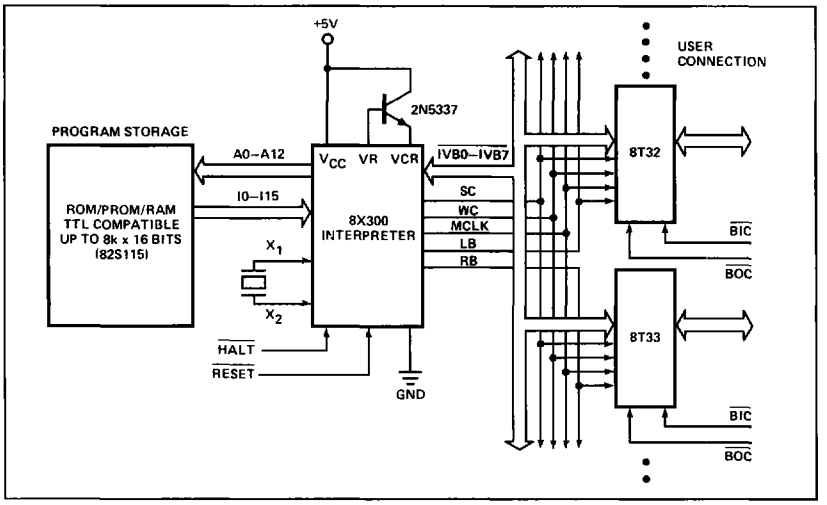
- **225 ns instruction decode and execute delay (with Signetics 8T32/33 I/O port).**
- **Eight 8-bit working registers.**
- **Single instruction access to 1-bit, 2-bit, 3-bit . . . or 8-bit field on I/O bus.**
- **Separate instruction address, instruction, and I/O data busses.**
- **On-chip oscillator.**
- **Bipolar Schottky technology.**
- **TTL inputs and outputs.**
- **Tri-state output on I/O data bus.**
- **+5 volt operation from 0° to 70°C.**

PIN CONFIGURATION

I PACKAGE



TYPICAL SYSTEM CONFIGURATION



PIN DESCRIPTION

PIN	SYMBOL	NAME AND FUNCTION	TYPE
2-9,45-49	A0-A12:	Instruction address lines. A high level equals "1." These outputs directly address up to 8192 words of program storage. A12 is least signigicant bit.	Active high
13-28	I0-I15:	Instruction lines. A high level equals "1." Receives instructions from Program Storage. I_{15} is least significant bit.	Active high
33-36,38-41	$\overline{IVB0}$-$\overline{IVB7}$	Interface Vector (IV) Bus. A low level equals "1." Bidirectional tri-state lines to communicate with I/O devices. $\overline{IVB7}$ is least significant bit.	Three-state Active low
42	MCLK:	Master Clock. Output to clock I/O devices, and/or provide synchronization for external logic	
30	WC:	Write Command. High level output indicates data is being output on the IV Bus.	Active high
29	SC:	Select Command. High level output indicates that an address is being output on the IV Bus.	Active high
31	$\overline{LB}$:	Left Bank. Low level output to enable one of two sets of I/O devices (LB is the complement of RB).	Active low
32	$\overline{RB}$:	Right Bank. Low level output to enable one of two sets of I/O devices (RB is the complement of LB).	Active low
44	$\overline{HALT}$:	Low level is input to stop the Interpreter.	Active low
43	$\overline{RESET}$:	Low level is input to initialize the Interpreter.	Active low
10-11	X1,X2:	Inputs for an external frequency determining crystal. May also be interfaced to logic or test equipment.	
50	VR	Reference Voltage to Pass Transistor.	
1	VCR	Regulated Output Voltage from Pass Transistor	
37	VCC:	5V power connection.	
12	GND:	Ground.	

PROGRAM STORAGE INTERFACE

Program Storage is typically connected to the A0-A12 (A12 is least signigicant bit) and I0-I15 signal lines. 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 interpreter operations which are to follow.

The Signetics 82S115 Prom, or any TTL compatible memory, may be used for program storage.

I/O DEVICES INTERFACE

An 8-bit I/O bus, called the Interface Vector (IV) data bus, is used by the Interpreter 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 selected.

Both I/O data and I/O address information can be output on the IV bus. The SC and WC signals are typically used to distinguish between I/O data and I/O address information as follows:

SC	WC	
1	0	I/O address is being output on IV bus
0	1	I/O data is being output on IV bus
0	0	I/O data is expected on the IV bus, as input to the Interpreter
1	1	Not generated by the Interpreter

The Signetics 82sxxx series RAM, and the 8T32/33 may be attached to the IV bus. (See Application Book)

INTERPRETER ARCHITECTURE AND OPERATION SUMMARY

Figure 1 provides a diagram of Interpreter internal architecture, and Table 1 summarizes Interpreter registers.

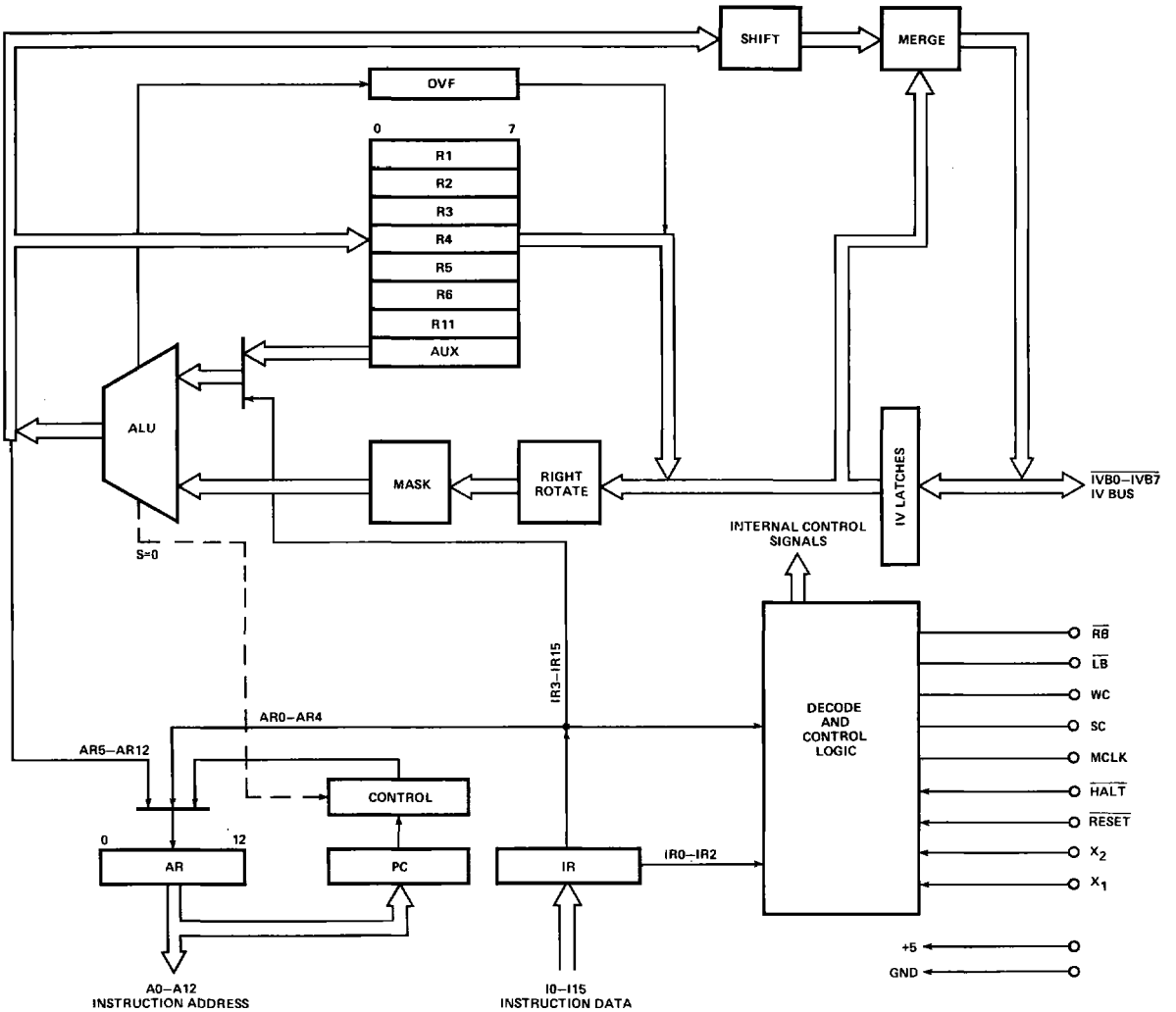


Figure 1

Interpreter Architecture

Table 1

INTERPRETER INTERNAL REGISTERS

Programmable Registers (all 8 bits):

AUX—General working register. Contains second term for arithmetic or logical operations.

R1 —General Working register.

R2 —General working register.

R3 —General working register.

R4 —General working register.

R5 —General working register.

R6 —General working register.

R11 —General working register.

Other Registers:

Address Register (AR) —A 13-bit register containing the address of the current instruction.

OVF—The least-significant bit of this register is used to reflect overflow status resulting from the most recent ADD operation (see Instruction Set Summary).

Program Counter (PC) —Normally contains the address of the current instruction and is incremented to obtain the next instruction address.

Instruction Register (IR)—Holds the 16-bit instruction word currently being executed.

INSTRUCTION CYCLE

Each interpreter operation is executed in one instruction cycle, which may be as short as 300ns. The interpreter generates MCLK to synchronize external logic to the instruction cycle. Instruction cycles are subdivided into quarter cycles. MCLK is an output during the last quarter cycle.

During the third quarter cycle of an instruction, an address is output on A0-A12, identifying the location in program storage of the next instruction word. This instruction word defines the next instruction, which must be input on I0-I15 during the first quarter cycle of the next instruction cycle (see Figure 2).

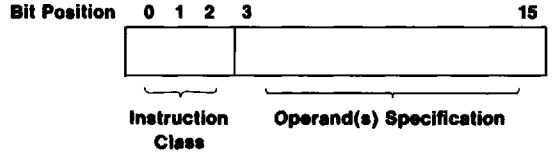
INST. AND IV BUS DATA INPUT	DATA PROCESSING	ADDR. AND IV BUS CHANGING	ADDR. AND IV BUS DATA VALID MCLK=HIGH
← ¼ cycle →	← ¼ cycle →	← ¼ cycle →	← ¼ cycle →

Figure 2

INSTRUCTION CYCLE

Instruction Set Summary

The 16-bit instruction word input on I0-I15 is decoded by the instruction decode logic to implement events that are to occur during the remainder of the instruction cycle. Generally the 16-bit instruction word is decoded as follows:



A detailed usage of the 13 "operand(s) specification" bits is given in following sections.

Three operation code bits allow for eight instruction classes. The eight instruction classes are summarized in Table 2. Each entry is referred to as an "instruction class" because the unique architecture of the interpreter allows a number of powerful variations to be specified by the thirteen operand(s) specification bits. A complete description of instruction formats and some instruction examples are provided in the Applications Guide.

Data Processing

The interpreter architecture includes eight 8-bit working registers, an arithmetic logic unit (ALU), an overflow register, and the 8-bit IV Bus. Internal 8-bit data paths connect the registers and IV Bus to the ALU inputs, and the ALU output to the registers and IV Bus. Data processing logic is distributed along these internal 8-bit data paths. Rotate and mask logic precedes the ALU on the data entry path. Shift and merge logic follows the ALU on the data output path. All four sets of logic can operate on eight data bits in a single instruction cycle. (See Figure 1)

When less than eight bits of data are specified for output to the IV bus by the ALU, the data field (shifted if necessary) is inserted into the prior contents of the IV bus latches. The IV bus latches contain data input at the start of an instruction. This data in the IV bus latches will be specified in the instruction as a) IV bus source data or b) data from an automatic read when the IV bus is specified as a destination. Therefore, IV bus bit positions outside an inserted bit field are unmodified.

Data Addressing

Sources and destinations of data are specified using a 5-bit octal number, as shown in Table 3. The source and/or destination of data to be operated upon is specified in a single instruction word.

Referring to Table 1, the Auxiliary register (address 00) is the implied source of the second argument for ADD, AND or XOR operations.

IVL and IVR are write-only registers used only as a destination. They have addresses and are treated as registers, but in reality they do not exist. When IVL is specified as a destination or the D field = $20\text{-}27_8$, then $\overline{LB}$ = 'low', $\overline{RB}$ = 'high' are generated; when IVR is specified as a destination or the D field = $30\text{-}37_8$, then $\overline{RB}$ = low, $\overline{LB}$ = 'high' are generated.

When IVL or IVR is specified as the destination in an instruction, SC is also activated and data is placed on the IV bus. If IVL or IVR is specified as a source of data, the source data is all zeros.

INSTRUCTION SEQUENCE CONTROL

The Address Register and Program Counter are used to generate addresses for accessing an instruction. The Address Register is used to form the instruction address, and in all but three instructions (XEC, NZT, and JMP) the address is copied into the Program Counter. The instruction address is formed in one of three ways:

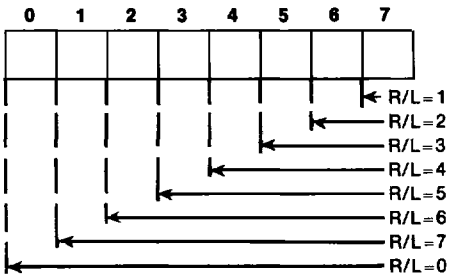
1. For all instructions but the JMP, XEC, and a satisfied NZT, the Program Counter is incremented by one and placed in the Address Register.
2. For the JMP instruction, the full 13-bit address field from the JMP instruction is placed into the Address Register and copied into the Program Counter.
3. For the XEC and NZT instructions, the high order 5- or 8-bits of the Program Counter are combined with 8- or 5-lower-order bits of ALU output (XEC or NZT) and placed in the Address Register. For the NZT instruction, it is also copied into the Program Counter.

TABLE 2
INSTRUCTION SET SUMMARY

INSTRUCTION MNEMONIC	OP CODE	FORMATS	DESCRIPTION	I/O CONTROL SIGNALS	←INSTRUCTION CYCLE→ INSTRUCTION INPUT AND DATA PROCESSING	ADDRESS/IV BUS OUTPUT
MOVE	0	Register to Register 0 2 3 7 8 10 11 15 0 | S | R/L | D S≠07,17,20-37$_8$ D≠10,20-37$_8$	(S)→D Move contents of register specified by S to register specified by D. Right rotate contents of register S by R/L places before operation.	SC= WC= LB/RB= 1 if D=17	0 0 1 if D=17	1 if D=07,17 0 1 if D=17
		IV Bus to Register: 0 2 3 7 8 10 11 15 0 | S | R/L | D S=20-37$_8$ D≠10,20-37$_8$	Move right rotated IV bus (source) data specified by S to register specified by D. R/L specifies the length of source data with most significant bits set to zero.	SC= WC= LB/RB= LB/RB=	0 0 0 if S=20-27 1 if S=30-37	1 if D=07,17 0 1 if D=17 1 if D=17
		Register to IV Bus: 0 2 3 7 8 10 11 15 0 | S | R/L | D S≠07,17,20-37$_8$ D=20-37$_8$	Move contents of register specified by S to the IV bus. Before placement on IV bus, data is shifted as specified by D, and R/L bits merged with existing IV bus data.	SC= WC= LB/RB= LB/RB=	0 0 x x	0 1 0 if D=20-27 1 if D=30-37
		IV Bus to IV Bus: 0 2 3 7 8 10 11 15 0 | S | R/L | D S=20-37$_8$ D=20-37$_8$	Move right rotated IV bus data (sources) specified by S to the IV bus. Before placement on IV bus, data is shifted or specified by D and R/L specifies the legnth of source data and of destination data merged with existing IV bus data.	SC= WC= LB/RB= LB/RB=	0 0 0 if S=20-27 1 if S=30-37	0 1 0 if D=30-37 1 if D=30-37
ADD	1	SAME AS MOVE	(S) plus (AUX) →D Same as MOVE but contents of AUX ADDed to the source data. If carry from most significant bit then OVF=1, otherwise OVF=0		SAME AS MOVE	
AND	2	SAME AS MOVE	(S) ∧ (AUX) →D Same as MOVE but contents of AUX ANDed with source data.		SAME AS MOVE	
XOR	3	SAME AS MOVE	(S) ⊕ (AUX) →D Same as MOVE but contents of AUX exclusive ORed with source data.		SAME AS MOVE	
XEC	4	Register Immediate: 0 2 3 7 8 15 4 | S | I S≠07,17,20-37$_8$ I=000-377$_8$	Execute instruction at current page address offset by I + (S). EXECute the instruction at the address determined by catenating 5 high order bits of PC with the 8 bit sum of I and register specified by S. PC is not incremented.	SC= WC= LB/RB	0 0 x	0 0 x
		IV Bus Immediate: 0 2 3 7 8 10 11 15 4 | S | R/L | I S=20-37$_8$ I=00-37$_8$	EXECute the instruction at the address determined by catenating 8 high order bits of PC with the 5 bit sum of I and rotated IV bus data (source) specified by S. R/L specifies length of source data with most significant bits set to zero. PC is not incremented.	SC= WC= LB/RB= LB/RB=	0 0 0 if S=20-27 1 if S=30-37	0 0 x x

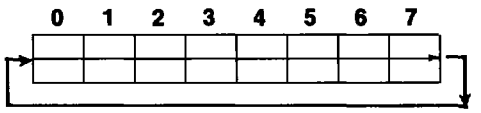
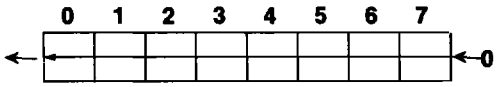
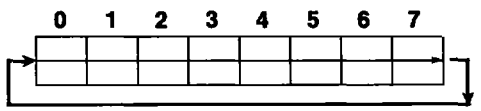
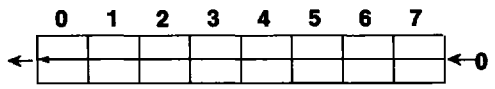
TABLE 2
INSTRUCTION SET SUMMARY

INSTRUCTION MNEMONIC	OP CODE	FORMATS	DESCRIPTION	I/O CONTROL SIGNALS	←INSTRUCTION CYCLE→ INSTRUCTION INPUT AND DATA PROCESSING	ADDRESS/IV BUS OUTPUT
NZT	5	Register Immediate: 0 2 3 7 8 15 5 | S | I S≠07,17,20-37$_8$ I=000-377$_8$	If (S) ≠ 0, jump to current page address offset by I; otherwise PC + 1 → PC If contents of register specified by S is Non Zero then Transfer to address determined by catenating 5 high order bits of PC with I; otherwise increment PC.	SC= WC= $\overline{LB}$/RB=	0 0 x	0 0 x
		IV Bus Immediate: 0 2 3 7 8 10 11 15 5 | S | R/L | I S=20-37$_8$ I=00-37$_8$	If right rotated IV bus data (source) is Non Zero then Transfer to address determined by catenating 8 high order bits of PC with I; otherwise increment PC.	SC= WC= $\overline{LB}$/RB= $\overline{LB}$/RB=	0 0 0 if S=20-27 1 if S=30-37	0 0 x x
XMIT	6	Register Immediate: 0 2 3 7 8 15 6 | D | I D≠20-37$_8$ I=000-377$_8$	Transmit I → D TRANSMIT and store 8 bit binary pattern I to register specified by D.	SC= WC= $\overline{LB}$/RB=	0 0 x	1 if D=07,17 0 1 if D=17
		IV BUS IMMEDIATE 0 2 3 7 8 10 11 15 6 | D | R/L | I D=20-37$_8$ I=00-37$_8$	TRANSMIT binary pattern I to IV bus. Before placement on IV bus, literal I is shifted as specified by D and R/L bits merged with existing IV bus data.	SC= WC= $\overline{LB}$/RB= $\overline{LB}$/RB=	0 0 x x	0 1 0 if S=20-27 1 if S=30-37
JMP	7	Address Immediate: 0 2 3 15 7 | A A=00000-17777$_8$	Jump to Program Address A JUMP to program storage address A. A is stored in the address register (AR).	SC= WC=$\overline{LB}$/RB	0 0 x	0 0 x



IV BUS DATA LENGTH SPECIFICATION

TABLE 3
DATA SOURCE/DESTINATION ADDRESS

S AND/OR D FIELD SPECIFICATION (OCTAL)	SOURCE/DESTINATION
00	Auxiliary Register (AUX)
01 to 06	Work registgers (R1 to R6) respectively
07	IVL write-only register (destination only)
10	Overflow status (OVF) — source only
11	Working register (R11)
17	IVR write-only register (destination only)
2N (N=0,1,2, 3,4,5,6,7)	a. If a source, IV bus data right rotated (7 — N) bits and masked (specified by R/L). $\overline{LB}$='low' and $\overline{RB}$='high' generated. **IV Bus Source Data**  b. If a destination, IV bus data left shifted (7 — N) bits and merged (specified by R/L). $\overline{LB}$='low' and $\overline{RB}$='high' generated. **IV Bus Destination Data** 
3N (N=0,1,2, 3,4,5,6,7)	a. If a source, IV bus data right rotated (7 — N bits and masked (specified by R/L). $\overline{LB}$='high' and $\overline{RB}$='low' generated. **IV Bus Source Data**  b. If a destination, IV bus data left shifted (7 — N) bits and merged (specified by R/L). $\overline{LB}$='high' and $\overline{RB}$='low' generated. **IV Bus Destination Data** 

SYSTEM DESIGN USING THE INTERPRETER

Designing hardware around the 8X300 Interpreter reduces to selecting a program storage devicer (ROM, PROM, etc.), selecting I/O devices (IV BYTE, MULTIPLEXERS, RAM, etc.), selecting clock mode (system driven or crystal controlled) and interfacing the Interpreter to these components, as shown in Figure 3.

SYSTEM CLOCK

The Interpreter has an integrated oscillator which generates all necessary clock signals. The oscillator is designed to connect directly to a series resonant quartz crystal via pins X1 and X2. The crystal resonant frequency, f, is related to the desired cycle time, T, by the relationship f=2/T. For a 300 ns system, f=6.667 MHz.

In lower speed applications where the cycle time need not be precisely controlled, a capacitor may be connected between X1 and X2 to drive the oscillator. If cycle time is to be varied, X1 and X2 should be driven from complementary outputs of a pulse generator. Figure 4 shows a typical configuration. For systems where the Interpreter is to be driven from a master clock, the X1 and X2 lines may be interfaced to TTL logic as shown in Figure 5.

CRYSTAL CHARACTERISTICS

Type:	*Fundamental mode, series resonant*
Impedance at Fundamental:	35 ohms maximum
Impedance at harmonics and spurs:	50 ohms minimum

HALT, RESET SIGNALS

$\overline{\text{HALT}}$:

A low level at the $\overline{\text{HALT}}$ input causes the Interpreter to stop processing after completion of the current instruction (end of quarter cycle when MCLK is high). $\overline{\text{HALT}}$ does not inhibit MCLK or affect any internal registers. Normal operations begins with the next complete instruction cycle after the $\overline{\text{HALT}}$ input goes high.

$\overline{\text{RESET}}$:

A low level at the $\overline{\text{RESET}}$ input sets the program counter and address register to zero and inhibits MCLK. $\overline{\text{RESET}}$ must be applied for at least one full instruction cycle to insure both registers are cleared. MCLK occurs on first instruction cycle and normal operation begins with the second instruction cycle after the $\overline{\text{RESET}}$ input goes high.

EXAMPLE:

A specific example of a control system, using the 8X300 Interpreter — four 8T32/33 IV Bytes, and two 82S215 ROMs is shown in Figure 3. Only eight components are required to build this system which contains 512 words of program storage, 32 TTL I/O connection points, and operates at a 300-ns instruction cycle time.

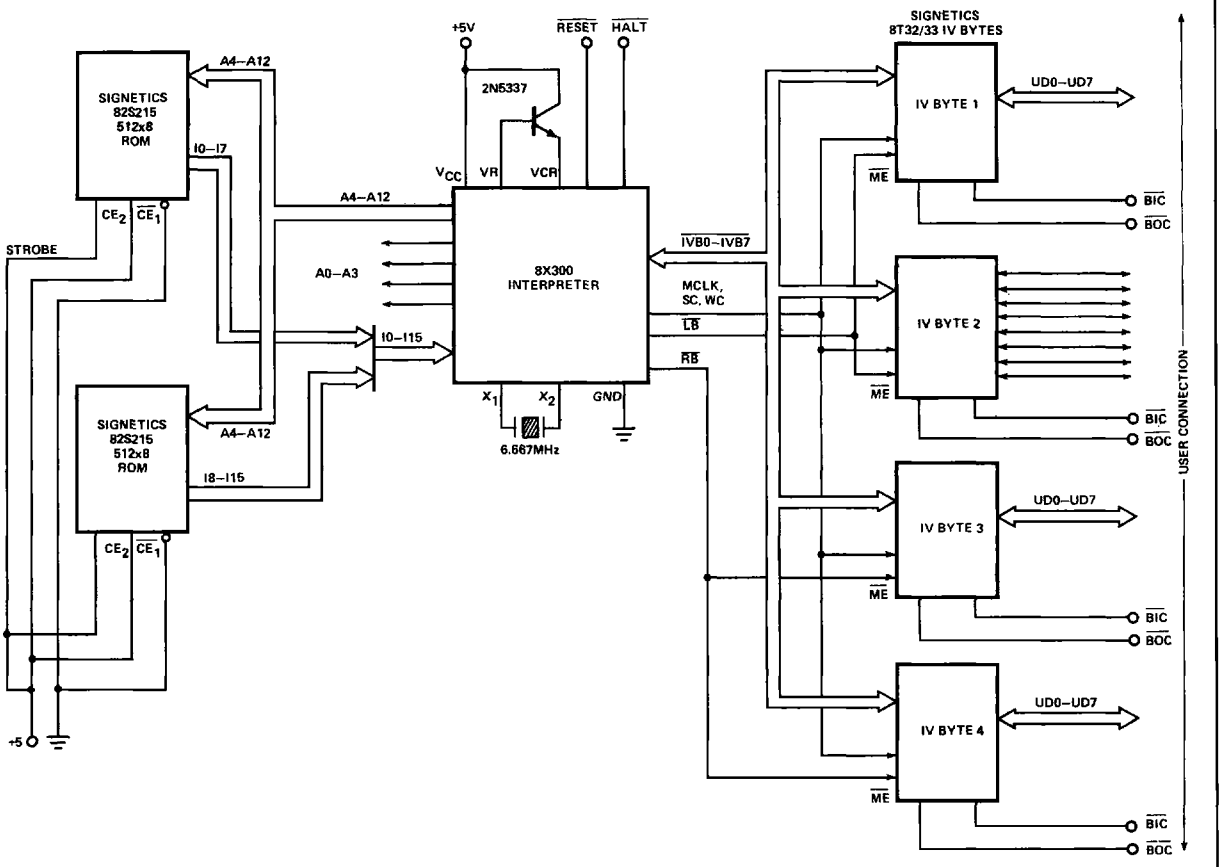
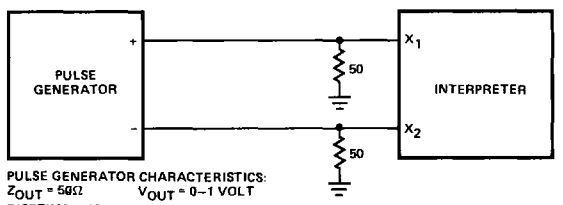


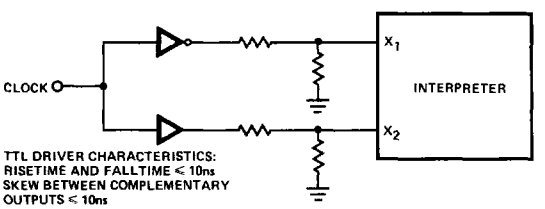
FIGURE 3
EXAMPLE OF CONTROL SYSTEM



PULSE GENERATOR CHARACTERISTICS:
$Z_{OUT} = 50\Omega$ $V_{OUT} = 0–1$ VOLT
RISETIME ≤ 10ns SKEW ≤ 10ns
COMPLEMENTARY OUTPUTS

NOTE: ALL RESISTORS VALUES ARE TYPICAL AND IN OHMS.

FIGURE 4
CLOCKING WITH A PULSE GENERATOR



TTL DRIVER CHARACTERISTICS:
RISETIME AND FALLTIME ≤ 10ns
SKEW BETWEEN COMPLEMENTARY OUTPUTS ≤ 10ns

FIGURE 5
CLOCKING WITH TTL

SYSTEM TIMING

The system instruction cycle time is determined by program storage access time, I/O register data/control delays, and Interpreter propagation delays. Instruction cycle time is normally constrained by two major propagation delay paths:

1. Program storage access time + instruction -to- address stable delay or

   Program storage access time + I/O control input delay + I/O device access time, + IV Bus to address stable delay

II. I/O control input delay + I/O device access time.

These propagation path delay times must be consistent with the Interpreter internal clock times.

Interpreter internal clock intervals occur every quarter cycle of a complete instruction cycle. The Interpreter output MCLK is high during the last quarter cycle of every instruction cycle. Interpreter input operations (instruction data, IV Bus data) occur during the first two quarter cycles (INPUT PHASE). Interpreter output operations (address, IV Bus data) occur during the last two quarter cycles (OUTPUT PHASE). Figure 6 illustrates typical timing waveforms for an instruction cycle. Interpreter propagation delays are shown in Figure 6.

Propagation path II delay time must be less than one quarter cycle. Interpreter delay times which are applicable are: MCLK TO SC/WC INPUT CONTROL and MCLK TO LB/RB INPUT CONTROL. These delays occur during the first part of the first quarter cycle as shown in Figure 6 and correspond to I/O control input delays.

The maximum I/O device access time is the difference between one quarter cycle time and the I/O control input delay. Using the delay values, the required I/O device access time is determined by the following equations:

35ns + I/O device access ≤ ¼(cycle time) EQ1

25ns + I/O device access ≤ ¼(cycle time) EQ2

For a 300-ns instruction cycle time, I/O device access times must be less than 40ns and 50ns respectively. The Signetics 8T32/33 IV Byte is an I/O register which satisfies the I/O device access time constraints.

Propagation path I determines the allowable program storage access time for a given instruction cycle time. The program storage access time is the smaller of these two equations:

Program storage access ≤ cycle time — instruction to address stable EQ3

Program storage access ≤ cycle time — (I/O device access + I/O control input delay ≤ IV Bus to address stable delay) EQ4

Therefore, a cycle time of 300 ns requires a program storage access time of 85ns or less.

Tradeoffs can be made between I/O device access time and program storage access time. If the I/O device access and program storage access times are less than the limits determined in equations EQ1,EQ2 and EQ3, then EQ4 can be used to trade I/O device and program storage access times.

Propagation delays during the OUTPUT PHASE usually do not limit instruction cycle times. MCLK is normally used to control data entry into I/O devices on the IV Bus during the last two quarter cycles. The user must insure that data set-up time requirements of I/O devices are satisfied. Data output on the IV Bus will be stable for the duration of MCLK if the I/O device access time and instruction cycle time satisfy the following equation:

I/O device access + I/O control input delay + ½(cycle time) ≤ MCLK to IV Bus stable delay.

If the above inequality is not satisfied, the IV Bus data may be changing during MCLK.

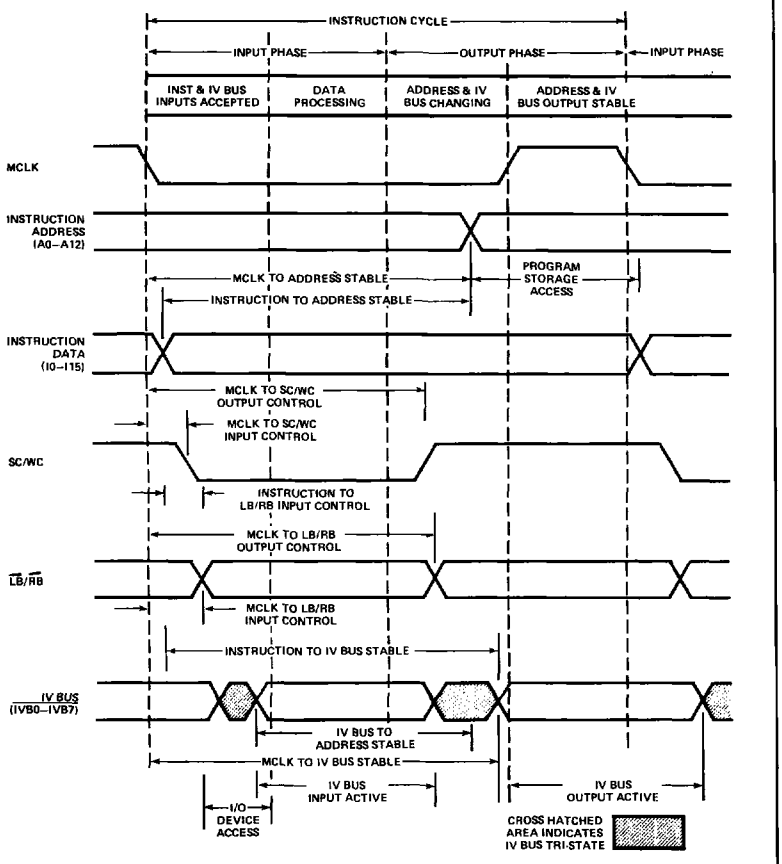


Figure 6
TYPICAL INSTRUCTION CYCLE TIMING

ABSOLUTE MAXIMUM RATINGS

Supply Voltage V_{CC} 7V
Logic Input Voltage 5.5V
Crystal Input Voltage 2V

AC ELECTRICAL CHARACTERISTICS

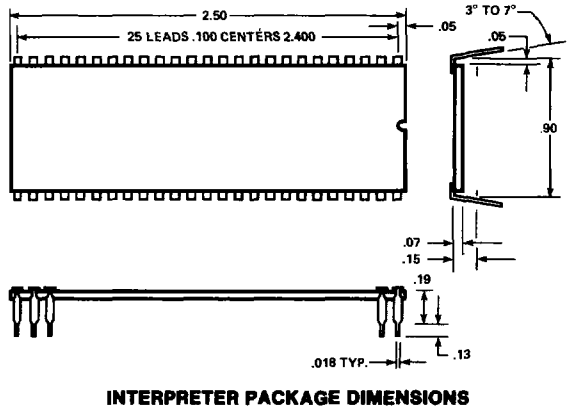
DELAY DESCRIPTION	PROPAGATION DELAY LIMIT	CYCLE TIME LIMIT
X1 falling edge to MCLK falling edge		
MCLK to SC/WC input control	25ns	
MCLK to SC/WC output control *		.5(CYC) + 25ns
INSTRUCTION to $\overline{LB}/\overline{RB}$ input control	35ns	
MCLK to $\overline{LB}/\overline{RB}$ input control	35ns	
MCLK to $\overline{LB}/\overline{RB}$ output control		.5 (CYC) + 35ns
INSTRUCTION to IV BUS stable	225ns	
MCLK to IV BUS stable	225ns	
IV BUS input stable to IV BUS output stable	150ns	
INSTRUCTION to ADDRESS stable	215ns	
MCLK to ADDRESS stable	215ns	.5 (CYC) + 40ns
IV BUS to ADDRESS stable	140ns	
MCLK falling edge to $\overline{HALT}$ falling edge	¼(CYC)—40ns (max)	
MCLK falling edge to $\overline{HALT}$ rising edge	¼(CYC)—40ns	
MCLK falling edge to $\overline{RESET}$ falling edge	½(CYC) (max)	
$\overline{RESET}$ rising edge to first MCLK	0 to 1 CYC (max)	

Limits apply for V_{CC} = 5V ≤ 5% and 0°C ≤ T_A ≤70°C.
Loading on ADDRESS outputs ≤ 100pF and other outputs ≤ 300pF
*IV Bus outputs remain Hi Z for at least 20ns after SC/WC output control

DC ELECTRICAL CHARACTERISTICS

PARAMETER	TEST CONDITIONS	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNITS
V_{IH} High-level input voltage					
X1, X2		.6			V
All others		2			V
V_{IL} Low-level input voltage					
X1, X2				.4	V
All others				.8	V
V_{CL} Input clamp voltage (Note 1)	V_{CC} = 4.75V, I_I = -10mA			-1.5	V
I_{IH} High-level input current					
X1, X2	V_{CC} = 5.25V, V_{IH} = .6V		2700		μA
All others	V_{CC} = 5.25V, V_{IH} = 4.5V		<1	50	μA
I_{IL} Low-level input current					
X1, X2	V_{CC} = 5.25V, V_{IL} = .4V		-2500		μA
$\overline{IVBO}$-7	V_{CC} = 5.25V, V_{IL} = .4V		-140	-200	μA
I0-I15	V_{CC} = 5.25V, V_{IL} = .4V		-880	-1600	μA
$\overline{HALT}$, $\overline{RESET}$	V_{CC} = 5.25V, V_{IL} = .4V		-230	-400	μA
V_{OL} Low-level output voltage					
A0-A12	V_{CC} = 4.75V, I_{OL} = 4.25mA		.35	.55	V
All others	V_{CC} = 4.75V, I_{OL} = 16mA		.35	.55	V
V_{OH} High-level output voltage	V_{CC} = 4.75V, I_{OH} = 3mA	2.4			V
I_{OS} Short circuit output current (Note 2)	V_{CC} = 5.25V	-30		-140	mA
V_{CC} Supply voltage		4.75	5	5.25	V
I_{CC} Supply current	V_{CC} = 5.25V		300	450	mA

NOTES:
1. Crystal inputs X1 and X2 do not have clamp diodes.
2. Only one output may be grounded at a time.
3. (Limits apply for V_{CC} = 5V ± 5% and 0°C<T_A<70°C unless specified otherwise.)



INTERPRETER PACKAGE DIMENSIONS