M8051AH, M8031AH

Single-Component 8-Bit Microcomputer

The M8031/AH/M8051AH is Intel's HMOS II version of the high performance 8-bit M8031/M8051 microcomputer. While the M8031AH/M8051AH features the same powerful architecture and instruction set as its HMOS I predecessor, it offers the additional benefit of lower power supply current.

The M8031AH/M8051AH provides a cost-effective solution for those controller applications requiring up to 64K bytes of program and/or 64K bytes of data storage. Specifically, the M8031AH contains 128 bytes of read/write data memory; 32 I/O lines configured as four 8-bit parallel ports; two 16-bit timer/counters; a five-source, two-priority level, nested interrupt structure; a programmable serial I/O port; and an on-chip oscillator with clock circuitry. The M8051AH has all of these M8031AH features plus 4k bytes of nonvolatile read only program memory. Both microcomputers can use standard TTL compatible memories and most byte-oriented MCS-80 and MCS-85 peripherals for additional I/O and memory capabilities.

Rochester Electronics Manufactured Components

Rochester branded components are manufactured using either die/wafers purchased from the original suppliers or Rochester wafers recreated from the original IP. All recreations are done with the approval of the OCM.

Parts are tested using original factory test programs or Rochester developed test solutions to guarantee product meets or exceeds the OCM data sheet.

Quality Overview

- ISO-9001
- AS9120 certification
- Qualified Manufacturers List (QML) MIL-PRF-38535
    - Class Q Military
    - Class V Space Level
- Qualified Suppliers List of Distributors (QSLD)
    - Rochester is a critical supplier to DLA and meets all industry and DLA standards.

Rochester Electronics, LLC is committed to supplying products that satisfy customer expectations for quality and are equal to those originally supplied by industry manufacturers.

The original manufacturer's datasheet accompanying this document reflects the performance and specifications of the Rochester manufactured version of this device. Rochester Electronics guarantees the performance of its semiconductor products to the original OEM specifications. 'Typical' values are for reference purposes only. Certain minimum or maximum ratings may be based on product characterization, design, simulation, or sample testing.

© 2014 Rochester Electronics, LLC. All Rights Reserved 01172014

T-49-19-07

M8051AH/M8031AH
SINGLE-COMPONENT 8-BIT MICROCOMPUTER

Military

T-49-19-59

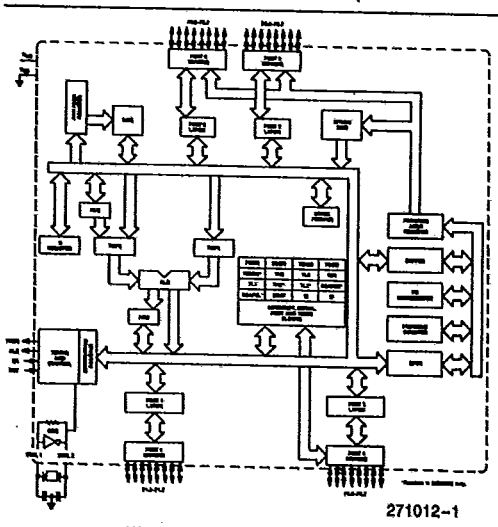
- M8031AH—Control-Oriented CPU with RAM and I/O
- M8051AH—An M8031AH with Factory Mask-Programmable ROM
- Fabricated with Intel's HMOS* II Process
- 12 MHz Operation
- 4K x 8 ROM (M8051AH only)
- 128 x 8 RAM
- 32 I/O Lines (Four 8-Bit Ports)
- Two 16-Bit Timer/Counters
- Programmable Full-Duplex Serial Channel
- 128K Accessible External Memory
- Boolean Processor
- 4 μs Multiply and Divide
- 256 User Bit-Addressable Locations
- 100 mA Typical Supply Current
- Military Temperature Range: −55°C to +125°C (T_C)

The M8031/AH/M8051AH is Intel's HMOS II version of the high performance 8-bit M8031/M8051 microcomputer. While the M8031AH/M8051AH features the same powerful architecture and instruction set as its HMOS I predecessor, it offers the additional benefit of lower power supply current.

The M8031AH/M8051AH provides a cost-effective solution for those controller applications requiring up to 64K bytes of program and/or 64K bytes of data storage. Specifically, the M8031AH contains 128 bytes of read/write data memory; 32 I/O lines configured as four 8-bit parallel ports; two 16-bit timer/counters; a five-source, two-priority level, nested interrupt structure; a programmable serial I/O port; and an on-chip oscillator with clock circuitry. The M8051AH has all of these M8031AH features plus 4K bytes of nonvolatile read only program memory. Both microcomputers can use standard TTL compatible memories and most byte-oriented MCS®-80 and MCS-85 peripherals for additional I/O and memory capabilities.

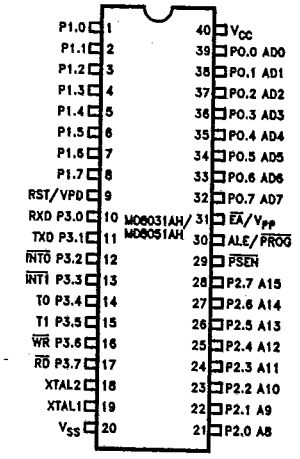
The M8031AH/M8051AH microcomputer, characteristic of the entire MCS-51 family, is efficient in both control and computational type applications. This results from extensive BCD/binary arithmetic and bit-handling facilities. The M8031AH/M8051AH also makes efficient use of its program memory space with an instruction set consisting of 44% one-byte, 41% two-byte, and 15% three-byte instructions. At the 12 MHz CPU operation, over half of the instructions execute in just 1.0 μs, while the longest instructions, multiply and divide, require only 4 μs.

*HMOS is a patented process of Intel Corporation.



271012-1

Figure 1. Block Diagram



271012-2

Figure 2. Pin Configuration

December 1989
Order Number: 271012-003

M8031AH/M8051AH PIN DESCRIPTIONS

V_{CC}

Supply voltage.

V_{SS}

Circuit ground.

PORT 0

Port 0 is an 8-bit open drain bidirectional I/O port. As an output port each pin can sink 8 LS TTL inputs.

Port 0 pins that have 1s written to them float, and in that state can be used as high-impedance inputs.

Port 0 is also the multiplexed low-order address and data bus during accesses to external Program and Data Memory. In this application it uses strong internal pullups when emitting 1s and can source and sink 8 LS TTL inputs.

Port 0 also outputs the code bytes during program verification of the ROM device. External pullups are required during program verification.

PORT 1

Port 1 is an 8-bit bidirectional I/O port with internal pullups. The Port 1 output buffers can sink/source 4 LS TTL inputs. Port 1 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL} on the data sheet) because of the internal pullups.

Port 1 also receives the low-order address bytes during program verification of the ROM device.

PORT 2

Port 2 is an 8-bit bidirectional I/O port with internal pullups. The Port 2 output buffers can sink/source 4 LS TTL inputs. Port 2 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL} on the data sheet) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external Program Memory and during accesses to external Data Memory that use 16-bit addresses (MOVX @DPTR). In this application it uses strong internal pullups when emitting 1s. During accesses to external Data Memory that use 8-bit addresses (MOVX @Ri), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits during program verification of the ROM device.

PORT 3

Port 3 is an 8-bit bidirectional I/O port with internal pullups. The Port 3 output buffers can sink/source 4 LS TTL inputs. Port 3 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL} on the data sheet) because of the pullups.

Port 3 also serves the functions of various special features of the MCS-51 Family, as listed below:

Port Pin	Alternative Function
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (Timer 0 external input)
P3.5	T1 (Timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$

Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory.

$\overline{PSEN}$

Program Store Enable is the read strobe to external Program Memory.

When the device is executing code from external Program Memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external Data Memory.

$\overline{EA}/V_{PP}$

External Access enable $\overline{EA}$ must be strapped to V_{SS} in order to enable any MCS-51 device to fetch code from external Program memory locations 0 to 0FFFH.

XTAL1

Input to the inverting oscillator amplifier.

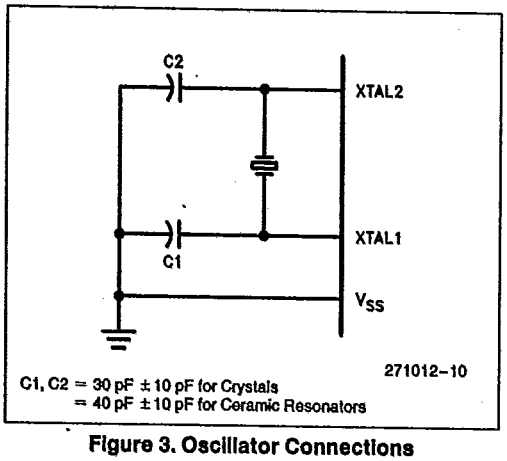


Figure 3. Oscillator Connections

XTAL2

Output from the inverting oscillator amplifier.

OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 3. Either a quartz crystal or ceramic resonator may be used. More detailed information concerning the use of the on-chip oscillator is available in Application Note AP-155, "Oscillators for Microcontrollers," published in the Embedded Controller Handbook.

To drive the device from an external clock source, XTAL1 should be grounded, while XTAL2 is driven, as shown in Figure 4. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum high and low times specified on the Data Sheet must be observed.

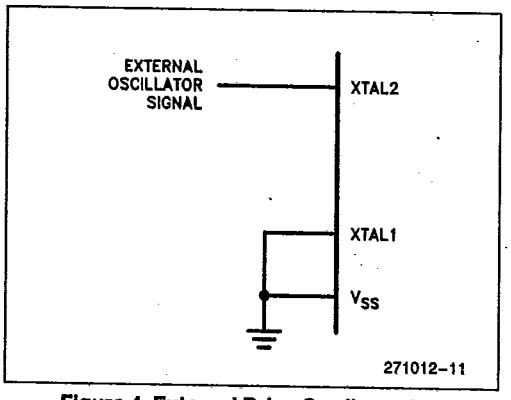


Figure 4. External Drive Configuration

ABSOLUTE MAXIMUM RATINGS*

Case Temperature Under Bias . . . −55°C to +125°C
Storage Temperature . . . . . −65°C to +150°C
Voltage on Any Pin With Respect to Ground (V_{SS}) . . . . . −0.5V to +7V
Power Dissipation 1W

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

Operating Conditions

Symbol	Description	Min	Max	Units
T_C	Case Temperature (Instant On)	−55	+125	°C
V_{CC}	Digital Supply Voltage	4.50	5.50	V

D.C. CHARACTERISTICS (Over Specified Operating Conditions)

Symbol	Parameter	Min	Max	Unit	Comments
V_{IL}	Input Low Voltage	−0.5	0.7	V	
V_{IH}	Input High Voltage (Except RST/V_{PD} and XTAL2)	2.2	V_{CC} + 0.5	V	
V_{IH1}	Input High Voltage to RST/V_{PD} For RESET, XTAL2	2.8	V_{CC} + 0.5	V	XTAL1 to V_{SS}
V_{PD}	Power Down Voltage to RST/V_{PD}	4.5	5.5	V	V_{CC} = 0V
V_{OL}	Output Low Voltage Ports 1, 2, 3 (Note 1)		0.45	V	I_{OL} = 1.2 mA
V_{OL1}	Output Low Voltage Port 0, ALE, $\overline{PSEN}$ (Note 1)		0.45	V	I_{OL} = 2.4 mA
V_{OH}	Output High Voltage Ports 1, 2, 3	2.4		V	I_{OH} = −60 µA
V_{OH1}	Output High Voltage Port 0, ALE, $\overline{PSEN}$	2.4		V	I_{OH} = −400 µA
I_{IL}	Logical 0 Input Current Ports 1, 2, 3		−800	µA	V_{in} = 0.45V
I_{IL2}	Logical 0 Input Current for XTAL2		−3.2	mA	XTAL1 = V_{SS}, V_{in} = 0.45V
I_{LI}	Input Leakage Current to Port 0, $\overline{EA}$		±10	µA	0.45V < V_{in} < V_{CC}
I_{IH1}	Input High Current to RST/V_{PD} For RESET		600	µA	V_{in} < (V_{CC} − 1.5V)
I_{CC}	Power Supply Current		140	mA	All Outputs Disconnected
I_{PD}	Power Down Current		15	mA	V_{CC} = 0V
C_{IO}	Capacitance of I/O Buffer		10	pF	f_c = 1 MHz, T_C = 25°C

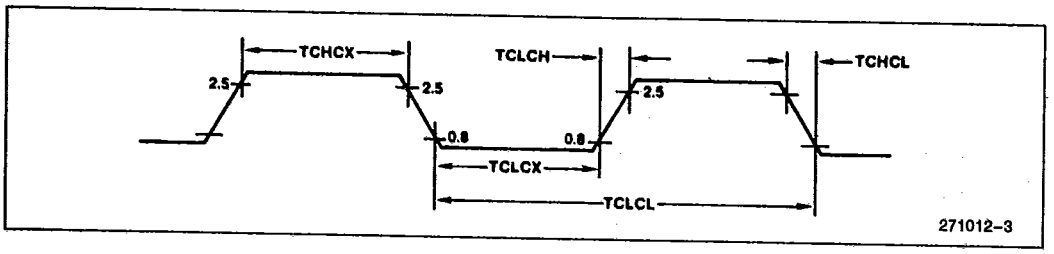
NOTES:
1. Capacitive loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the V_{OL}s of ALE and Ports 1 and 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100 pF), the noise pulse on the ALE line may exceed 0.8V. In such cases it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.

Datum	Emitting Ports	Degraded I/O Lines	V_{OL} (Peak) (Max)
Address	P2, P0	P1, P3	0.8V
Write Data	P0	P1, P3, ALE	0.8V

T-99-19-07
T-99-19-59

EXTERNAL CLOCK DRIVE CHARACTERISTICS (XTAL2)

Symbol	Parameter	Variable Clock Freq = 3.5 MHz to 12 MHz		Unit
		Min	Max	
TCLCL	Oscillator Period	83.3	285.7	ns
TCHCX	High Time	20		ns
TCLCX	Low Time	20		ns
TCLCH	Rise Time		20	ns
TCHCL	Fall Time		20	ns



A.C. CHARACTERISTICS

C_L for Port 0, ALE and $\overline{PSEN}$ Outputs = 100 pF; C_L for all other outputs = 80 pF

EXTERNAL PROGRAM MEMORY CHARACTERISTICS

Symbol	Parameter	12 MHz Clock			Variable Clock 1/TCLCL = 3.5 MHz to 12 MHz		
		Min	Max	Unit	Min	Max	Unit
TLHLL	ALE Pulse Width	112		ns	2TCLCL−55		ns
TAVLL	Address Setup to ALE	28		ns	TCLCL−55		ns
TLLAX	Address Hold After ALE	33		ns	TCLCL−50		ns
TLLIV	ALE to Valid Instr In		218	ns		4TCLCL−115	ns
TLLPL	ALE to $\overline{PSEN}$	43		ns	TCLCL−40		ns
TPLPH	$\overline{PSEN}$ Pulse Width	190		ns	3TCLCL−60		ns
TPLIV	$\overline{PSEN}$ to Valid Instr In		110	ns		3TCLCL−140	ns
TPXIX	Input Instr Hold After $\overline{PSEN}$	0		ns	0		ns
TPXIZ	Input Instr Float After $\overline{PSEN}$		48	ns		TCLCL−35	ns
TPXAV	Address Valid After $\overline{PSEN}$	58		ns	TCLCL−25		ns
TAVIV	Address to Valid Instr In		287	ns		5TCLCL−130	ns
TPLAZ	$\overline{PSEN}$ Low to Address Float		20	ns		20	ns

A.C. CHARACTERISTICS

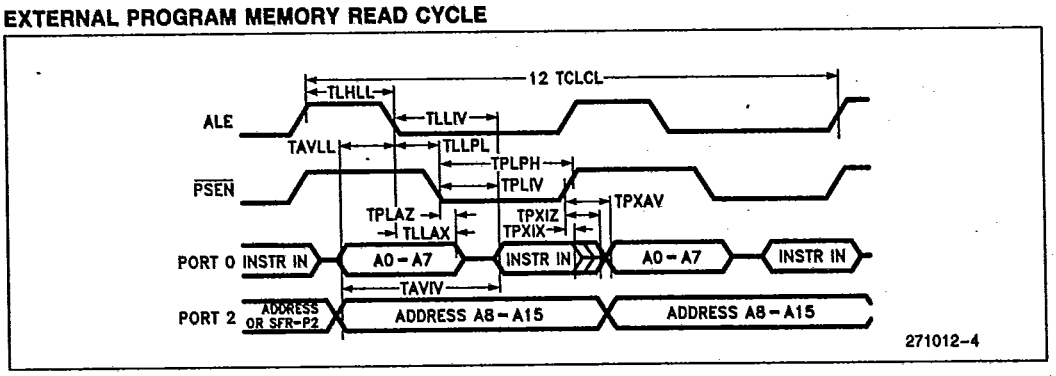
C_L for Port 0, ALE and $\overline{PSEN}$ Outputs = 100 pF; C_L for all other outputs = 80 pF (Continued)

EXTERNAL DATA MEMORY CHARACTERISTICS

Symbol	Parameter	12 MHz Clock			Variable Clock 1/TCLCL = 3.5 MHz to 12 MHz		
		Min	Max	Unit	Min	Max	Unit
TRLRH	$\overline{RD}$ Pulse Width	400		ns	6TCLCL-100		ns
TWLWH	$\overline{WR}$ Pulse Width	400		ns	6TCLCL-100		ns
TLLAX	Address Hold After ALE	33		ns	TCLCL-50		ns
TRLDV	$\overline{RD}$ to Valid Data In		232	ns		5TCLCL-185	ns
TRHDX	Data Hold After $\overline{RD}$	0		ns	0		ns
TRHDZ	Data Float After $\overline{RD}$		82	ns		2TCLCL-85	ns
TLLDV	ALE to Valid Data In		497	ns		8TCLCL-170	ns
TAVDV	Address to Valid Data In		565	ns		9TCLCL-185	ns
TLLWL	ALE to $\overline{WR}$ or $\overline{RD}$	185	315	ns	3TCLCL-65	3TCLCL+65	ns
TAVWL	Address to $\overline{WR}$ or $\overline{RD}$	188		ns	4TCLCL-145		ns
TWHLH	$\overline{WR}$ or $\overline{RD}$ High to ALE High	18	148	ns	TCLCL-65	TCLCL+65	ns
TDVWX	Data Valid to $\overline{WR}$ Transition	8		ns	TCLCL-75		ns
TQVWH	Data Setup Before $\overline{WR}$	508		ns	7TCLCL-75		ns
TWHQX	Data Hold After $\overline{WR}$	18		ns	TCLCL-65		ns
TRLAZ	$\overline{RD}$ Low to Address Float		20	ns		20	ns

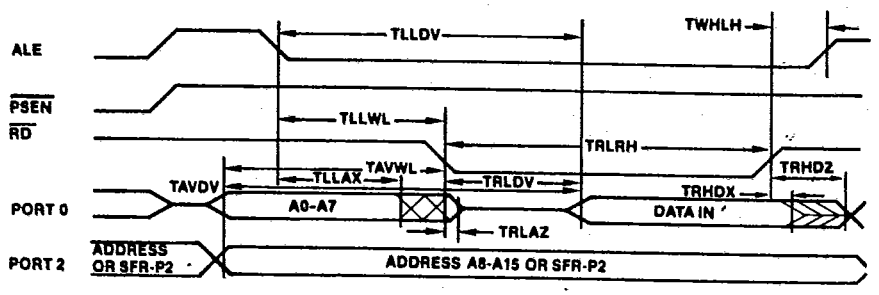
A.C. TIMING DIAGRAMS

EXTERNAL PROGRAM MEMORY READ CYCLE

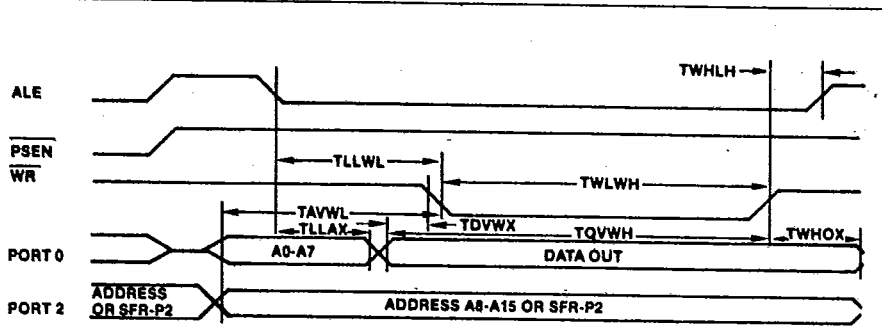


A.C. TIMING DIAGRAMS (Continued)

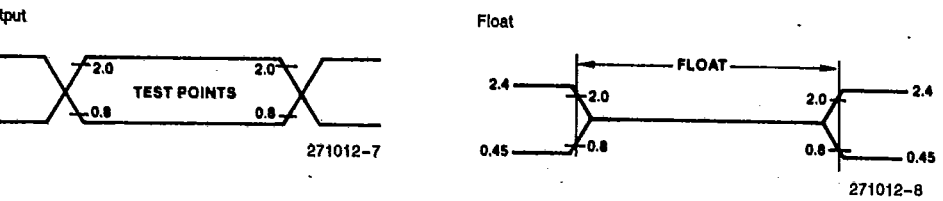
EXTERNAL DATA MEMORY READ CYCLE



EXTERNAL DATA MEMORY WRITE CYCLE

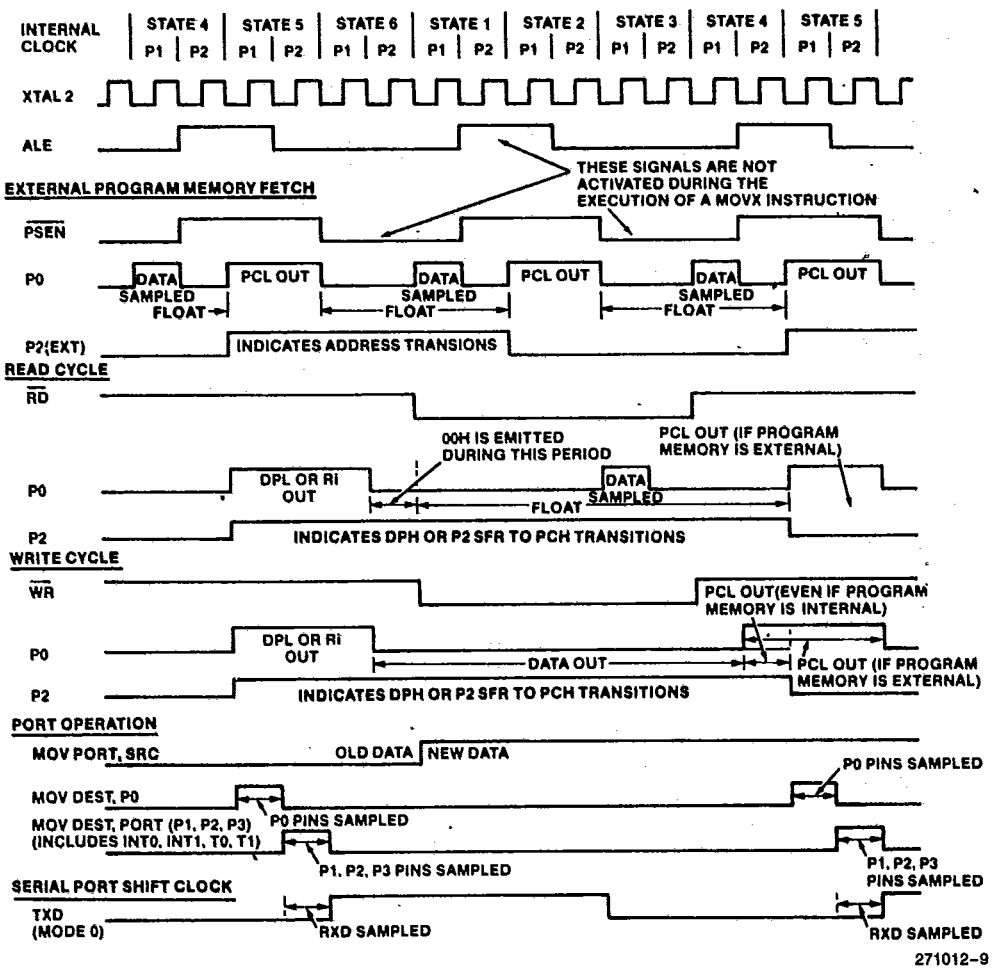


A.C. TESTING INPUT/OUTPUT, FLOAT WAVEFORMS



AC inputs during testing are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0". For timing purposes, the float state is defined as the point at which a P0 pin sinks 2.4 mA or sources 400 μA at the voltage test levels.

CLOCK WAVEFORMS



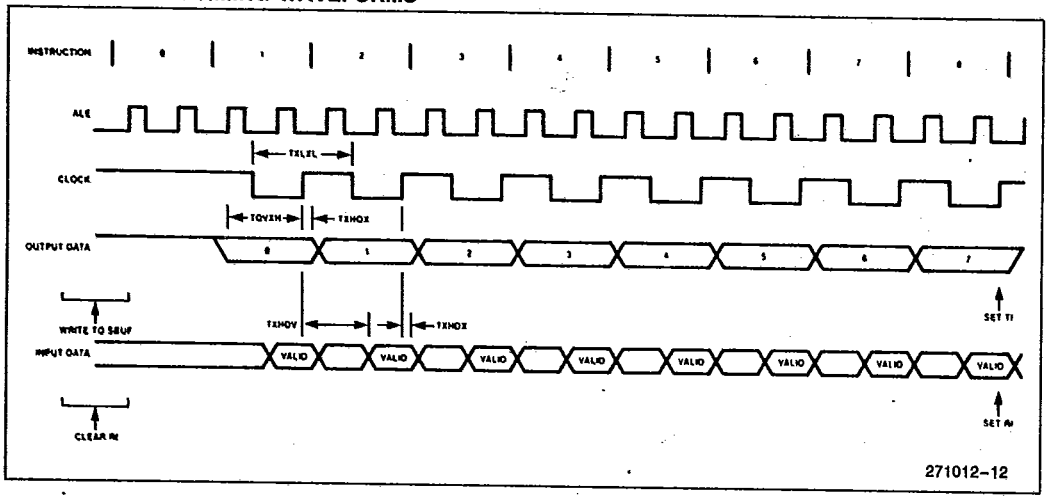
271012-9

This diagram indicates when signals are clocked internally. The time it takes the signals to propagate to the pins, however, ranges from 25 to 125 ns. This propagation delay is dependent on variables such as temperature and pin loading. Propagation also varies from output to output and component to component.

SERIAL PORT TIMING—SHIFT REGISTER MODE

Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
TXLXL	Serial Port Clock Cycle Time	1.0		12TCLCL		μs
TQVXH	Output Data Setup to Clock Rising Edge	700		10TCLCL−133		ns
TXHQX	Output Data Hold after Clock Rising Edge	50		2TCLCL−117		ns
TXHDX	Input Data Hold after Clock Rising Edge	0		0		ns
TXHDV	Clock Rising Edge to Input Data Valid		700		10TCLCL−133	ns

SHIFT REGISTER TIMING WAVEFORMS



271012-12

Table 1. MCS®-51 Instruction Set Description

ARITHMETIC OPERATIONS				
Mnemonic		**Description**	**Byte**	**Cyc**
ADD	A, Rn	Add register to Accumulator	1	1
ADD	A, direct	Add direct byte to Accumulator	2	1
ADD	A, @Ri	Add indirect RAM to Accumulator	1	1
ADD	A, #data	Add immediate data to Accumulator	2	1
ADDC	A, Rn	Add register to Accumulator with Carry	1	1
ADDC	A, direct	Add direct byte to A with Carry flag	2	1
ADDC	A, @Ri	Add indirect RAM to A with Carry flag	1	1
ADDC	A, #data	Add immediate data to A with Carry flag	2	1
SUBB	A, Rn	Subtract register from A with Borrow	1	1
SUBB	A, direct	Subtract direct byte from A with Borrow	2	1
SUBB	A, @Ri	Subtract indirect RAM from A with Borrow	1	1
SUBB	A, #data	Subtract immed data from A with Borrow	2	1
INC	A	Increment Accumulator	1	1
INC	Rn	Increment register	1	1
INC	direct	Increment direct byte	2	1
INC	@Ri	Increment indirect RAM	1	1
INC	DPTR	Increment Data Pointer	1	2
DEC	A	Decrement Accumulator	1	1
DEC	Rn	Decrement register	1	1
DEC	direct	Decrement direct byte	2	1
DEC	@Ri	Decrement indirect RAM	1	1
MUL	AB	Multiply A & B	1	4
DIV	AB	Divide A by B	1	4
DA	A	Decimal Adjust Accumulator	1	1

LOGICAL OPERATIONS				
Mnemonic		**Description**	**Byte**	**Cyc**
ANL	A, Rn	AND register to Accumulator	1	1
ANL	A, direct	AND direct byte to Accumulator	2	1
ANL	A, @Ri	AND indirect RAM to Accumulator	1	1
ANL	A, #data	AND immediate data to Accumulator	2	1
ANL	direct, A	AND Accumulator to direct byte	2	1
ANL	direct, #data	AND immediate data to direct byte	3	2
ORL	A, Rn	OR register to Accumulator	1	1
ORL	A, direct	OR direct byte to Accumulator	2	1
ORL	A, @Ri	OR indirect RAM to Accumulator	1	1
ORL	A, #data	OR immediate data to Accumulator	2	1
ORL	direct, A	OR Accumulator to direct byte	2	1
ORL	direct, #data	OR immediate data to direct byte	3	2
XRL	A, Rn	Exclusive-OR register to Accumulator	1	1
XRL	A, direct	Exclusive-OR direct byte to Accumulator	2	1
XRL	A, @Ri	Exclusive-OR indirect RAM to A	1	1
XRL	A, #data	Exclusive-OR immediate data to A	2	1
XRL	direct, A	Exclusive-OR Accumulator to direct byte	2	1
XRL	direct, #data	Exclusive-OR immediate data to direct	3	2
CLR	A	Clear Accumulator	1	1
CPL	A	Complement A	1	1
RL	A	Rotate Accumulator Left	1	1
RLC	A	Rotate A Left through the Carry flag	1	1
RR	A	Rotate Accumulator Right	1	1
RRC	A	Rotate A Right through Carry flag	1	1
SWAP	A	Swap nibbles within the Accumulator	1	1

DATA TRANSFER				
Mnemonic		**Description**	**Byte**	**Cyc**
MOV	A, Rn	Move register to Accumulator	1	1
MOV	A, direct	Move direct byte to Accumulator	2	1
MOV	A, @Ri	Move indirect RAM to Accumulator	1	1
MOV	A, #data	Move immediate data to Accumulator	2	1
MOV	Rn, A	Move Accumulator to register	1	1
MOV	Rn, direct	Move direct byte to register	2	2
MOV	Rn, #data	Move immediate data to register	2	1
MOV	direct, A	Move Accumulator to direct byte	2	1
MOV	direct, Rn	Move register to direct byte	2	2
MOV	direct, direct	Move direct byte to direct	3	2
MOV	direct, @Ri	Move indirect RAM to direct byte	2	2

Table 1. MCS®-51 Instruction Set Description (Continued)

DATA TRANSFER (Continued)

Mnemonic		Description	Byte	Cyc
MOV	direct, #data	Move immediate data to direct byte	3	2
MOV	@Ri, A	Move Accumulator to indirect RAM	1	1
MOV	@Ri, direct	Move direct byte to indirect RAM	2	2
MOV	@Ri, #data	Move immediate data to indirect RAM	2	1
MOV	DPTR, #data16	Load Data Pointer with a 16-bit constant	3	2
MOVC	A, @A+DPTR	Move Code byte relative to DPTR to A	1	2
MOVC	A, @A+PC	Move Code byte relative to PC to A	1	2
MOVX	A, @Ri	Move External RAM (8-bit addr) to A	1	2
MOVX	A, @DPTR	Move External RAM (16-bit addr) to A	1	2
MOVX	@Ri, A	Move A to External RAM (8-bit addr)	1	2
MOVX	@DPTR, A	Move A to External RAM (16-bit addr)	1	2
PUSH	direct	Push direct byte onto stack	2	2
POP	direct	Pop direct byte from stack	2	2
XCH	A, Rn	Exchange register with Accumulator	1	1
XCH	A, direct	Exchange direct byte with Accumulator	2	1
XCH	A, @Ri	Exchange indirect RAM with A	1	1
XCHD	A, @Ri	Exchange low-order Digit ind RAM with A	1	1

BOOLEAN VARIABLE MANIPULATION

Mnemonic		Description	Byte	Cyc
CLR	C	Clear Carry flag	1	1
CLR	bit	Clear direct bit	2	1
SETB	C	Set Carry flag	1	1
SETB	bit	Set direct Bit	2	1
CPL	C	Complement Carry flag	1	1
CPL	bit	Complement direct bit	2	1
ANL	C, bit	AND direct bit to Carry flag	2	2
ANL	C, /bit	AND complement of direct bit to Carry	2	2
ORL	C/bit	OR direct bit to Carry flag	2	2
ORL	C/bit	OR complement of direct bit to Carry	2	2
MOV	C, /bit	Move direct bit to Carry flag	2	1
MOV	bit, C	Move Carry flag to direct bit	2	2

PROGRAM AND MACHINE CONTROL

Mnemonic		Description	Byte	Cyc
ACALL	addr11	Absolute Subroutine Call	2	2
LCALL	addr16	Long Subroutine Call	3	2
RET		Return from subroutine	1	2
RETI		Return from interrupt	1	2
AJMP	addr11	Absolute Jump	2	2
LJMP	addr16	Long Jump	3	2
SJMP	rel	Short Jump (relative addr)	2	2
JMP	@A+DPTR	Jump indirect relative to the DPTR	1	2
JZ	rel	Jump if Accumulator is Zero	2	2
JNZ	rel	Jump if Accumulator is Not Zero	2	2
JC	rel	Jump if Carry flag is set	2	2
JNC	rel	Jump if No Carry flag	2	2
JB	bit, rel	Jump if direct Bit set	3	2
JB	bit, rel	Jump if direct Bit Not set	3	2
JBC	bit, rel	Jump if direct Bit is set & Clear bit	3	2
CJNE	A, direct,rel	Compare direct to A & Jump if Not Equal	3	2
CJNE	A, #data,rel	Comp, immed, to A & Jump if Not Equal	3	2
CJNE	Rn, #data,rel	Comp, immed, to reg & Jump if Not Equal	3	2
CJNE	@Ri, #data, rel	Comp, immed, to ind, & Jump if Not Equal	3	2
DJNZ	Rn, rel	Decrement register & Jump if Not Zero	2	2
DJNZ	direct, rel	Decrement direct & Jump if Not Zero	3	2
NOP		No operation	1	1

NOTES ON DATA ADDRESSING MODES:

Rn — Working register R0–R7

direct — 128 internal RAM locations, any I/O port, control or status register

@Ri — Indirect internal RAM location addressed by register R0 or R1

#data — 8-bit constant included in instruction

#data16 — 16-bit constant included as bytes 2 & 3 of instruction

bit — 128 software flags, any I/O pin, control or status bit

NOTES ON PROGRAM ADDRESSING MODES:

addr16 — Destination address for LCALL & LJMP may be anywhere within the 64-K program memory address space

addr11 — Destination address for ACALL & AJMP will be within the same 2-K page of program memory as the first byte of the following instruction

rel — SJMP and all conditional jumps include an 8-bit offset byte, Range is +127–128 bytes relative to first byte of the following instruction

All mnemonics copyrighted© Intel Corporation 1979

Table 2. Instruction Opcodes in Hexadecimal Order

Hex Code	Number of Bytes	Mnemonic	Operands
00	1	NOP	
01	2	AJMP	code addr
02	3	LJMP	code addr
03	1	RR	A
04	1	INC	A
05	2	INC	data addr
06	1	INC	@R0
07	1	INC	@R1
08	1	INC	R0
09	1	INC	R1
0A	1	INC	R2
0B	1	INC	R3
0C	1	INC	R4
0D	1	INC	R5
0E	1	INC	R6
0F	1	INC	R7
10	3	JBC	bit addr, code addr
11	2	ACALL	code addr
12	3	LCALL	code addr
13	1	RRC	A
14	1	DEC	A
15	2	DEC	data addr
16	1	DEC	@R0
17	1	DEC	@R1
18	1	DEC	R0
19	1	DEC	R1
1A	1	DEC	R2
1B	1	DEC	R3
1C	1	DEC	R4
1D	1	DEC	R5
1E	1	DEC	R6
1F	1	DEC	R7
20	3	JB	bit addr, code addr
21	2	AJMP	code addr
22	1	RET	
23	1	RL	A
24	2	ADD	A,#data
25	2	ADD	A,data addr
26	1	ADD	A,@R0
27	1	ADD	A,@R1
28	1	ADD	A,R0
29	1	ADD	A,R1
2A	1	ADD	A,R2
2B	1	ADD	A,R3
2C	1	ADD	A,R4
2D	1	ADD	A,R5
2E	1	ADD	A,R6
2F	1	ADD	A,R7
30	3	JNB	bit addr, code addr
31	2	ACALL	code addr
32	1	RETI	

Hex Code	Number of Bytes	Mnemonic	Operands
33	1	RLC	A
34	2	ADDC	A,#data
35	2	ADDC	A,data addr
36	1	ADDC	A,@R0
37	1	ADDC	A,@R1
38	1	ADDC	A,R0
39	1	ADDC	A,R1
3A	1	ADDC	A,R2
3B	1	ADDC	A,R3
3C	1	ADDC	A,R4
3D	1	ADDC	A,R5
3E	1	ADDC	A,R6
3F	1	ADDC	A,R7
40	2	JC	code addr
41	2	AJMP	code addr
42	2	ORL	data addr,A
43	3	ORL	data addr,#data
44	2	ORL	A,#data
45	2	ORL	A,data addr
46	1	ORL	A,@R0
47	1	ORL	A,@R1
48	1	ORL	A,R0
49	1	ORL	A,R1
4A	1	ORL	A,R2
4B	1	ORL	A,R3
4C	1	ORL	A,R4
4D	1	ORL	A,R5
4E	1	ORL	A,R6
4F	1	ORL	A,R7
50	2	JNC	code addr
51	2	ACALL	code addr
52	2	ANL	data addr,A
53	3	ANL	data addr,#data
54	2	ANL	A,#data
55	2	ANL	A,data addr
56	1	ANL	A,@R0
57	1	ANL	A,@R1
58	1	ANL	A,R0
59	1	ANL	A,R1
5A	1	ANL	A,R2
5B	1	ANL	A,R3
5C	1	ANL	A,R4
5D	1	ANL	A,R5
5E	1	ANL	A,R6
5F	1	ANL	A,R7
60	2	JZ	code addr
61	2	AJMP	code addr
62	2	XRL	data addr,A
63	3	XRL	data addr,#data
64	2	XRL	A,#data
65	2	XRL	A,data addr

Table 2. Instruction Opcodes in Hexadecimal Order (Continued)

Hex Code	Number of Bytes	Mnemonic	Operands
66	1	XRL	A,@R0
67	1	XRL	A,@R1
68	1	XRL	A,R0
69	1	XRL	A,R1
6A	1	XRL	A,R2
6B	1	XRL	A,R3
6C	1	XRL	A,R4
6D	1	XRL	A,R5
6E	1	XRL	A,R6
6F	1	XRL	A,R7
70	2	JNZ	code addr
71	2	ACALL	code addr
72	2	ORL	C,bit addr
73	1	JMP	@A+DPTR
74	2	MOV	A,#data
75	3	MOV	data addr,#data
76	2	MOV	@R0,#data
77	2	MOV	@R1,#data
78	2	MOV	R0,#data
79	2	MOV	R1,#data
7A	2	MOV	R2,#data
7B	2	MOV	R3,#data
7C	2	MOV	R4,#data
7D	2	MOV	R5,#data
7E	2	MOV	R6,#data
7F	2	MOV	R7,#data
80	2	SJMP	code addr
81	2	AJMP	code addr
82	2	ANL	C,bit addr
83	1	MOVC	A,@A+PC
84	1	DIV	AB
85	3	MOV	data addr, data addr
86	2	MOV	data addr,@R0
87	2	MOV	data addr,@R1
88	2	MOV	data addr,R0
89	2	MOV	data addr,R1
8A	2	MOV	data addr,R2
8B	2	MOV	data addr,R3
8C	2	MOV	data addr,R4
8D	2	MOV	data addr,R5
8E	2	MOV	data addr,R6
8F	2	MOV	data addr,R7
90	3	MOV	DPTR,#data
91	2	ACALL	code addr
92	2	MOV	bit addr,C
93	1	MOVC	A,@A+DPTR
94	2	SUBB	A,#data
95	2	SUBB	A,data addr
96	1	SUBB	A,@R0
97	1	SUBB	A,@R1
98	1	SUBB	A,R0
99	1	SUBB	A,R1
9A	1	SUBB	A,R2
9B	1	SUBB	A,R3
9C	1	SUBB	A,R4
9D	1	SUBB	A,R5
9E	1	SUBB	A,R6
9F	1	SUBB	A,R7
A0	2	ORL	C,/bit addr
A1	2	AJMP	code addr
A2	2	MOV	C,bit addr
A3	1	INC	DPTR
A4	1	MUL	AB
A5		reserved	
A6	2	MOV	@R0,data addr
A7	2	MOV	@R1,data addr
A8	2	MOV	R0,data addr
A9	2	MOV	R1,data addr
AA	2	MOV	R2,data addr
AB	2	MOV	R3,data addr
AC	2	MOV	R4,data addr
AD	2	MOV	R5,data addr
AE	2	MOV	R6,data addr
AF	2	MOV	R7,data addr
B0	2	ANL	C,/bit addr
B1	2	ACALL	code addr
B2	2	CPL	bit addr
B3	1	CPL	C
B4	3	CJNE	A,#data,code addr
B5	3	CJNE	A,data addr,code addr
B6	3	CJNE	@R0,#data,code addr
B7	3	CJNE	@R1,#data,code addr
B8	3	CJNE	R0,#data,code addr
B9	3	CJNE	R1,#data,code addr
BA	3	CJNE	R2,#data,code addr
BB	3	CJNE	R3,#data,code addr
BC	3	CJNE	R4,#data,code addr
BD	3	CJNE	R5,#data,code addr
BE	3	CJNE	R6,#data,code addr
BF	3	CJNE	R7,#data,code addr
C0	2	PUSH	data addr
C1	2	AJMP	code addr
C2	2	CLR	bit addr
C3	1	CLR	C
C4	1	SWAP	A
C5	2	XCH	A,data addr
C6	1	XCH	A,@R0
C7	1	XCH	A,@R1
C8	1	XCH	A,R0
C9	1	XCH	A,R1
CA	1	XCH	A,R2
CB	1	XCH	A,R3

Table 2. Instruction Opcodes In Hexadecimal Order (Continued)

Hex Code	Number of Bytes	Mnemonic	Operands
CC	1	XCH	A,R4
CD	1	XCH	A,R5
CE	1	XCH	A,R6
CF	1	XCH	A,R7
D0	2	POP	data addr
D1	2	ACALL	code addr
D2	2	SETB	bit addr
D3	1	SETB	C
D4	1	DA	A
D5	3	DJNZ	data addr,code addr
D6	1	XCHD	A,@R0
D7	1	XCHD	A,@R1
D8	2	DJNZ	R0,code addr
D9	2	DJNZ	R1,code addr
DA	2	DJNZ	R2,code addr
DB	2	DJNZ	R3,code addr
DC	2	DJNZ	R4,code addr
DD	2	DJNZ	R5,code addr
DE	2	DJNZ	R6,code addr
DF	2	DJNZ	R7,code addr
E0	1	MOVX	A,@DPTR
E1	2	AJMP	code addr
E2	1	MOVX	A,@R0
E3	1	MOVX	A,@R1
E4	1	CLR	A
E5	2	MOV	A,data addr
E6	1	MOV	A,@R0
E7	1	MOV	A,@R1
E8	1	MOV	A,R0
E9	1	MOV	A,R1
EA	1	MOV	A,R2
EB	1	MOV	A,R3
EC	1	MOV	A,R4
ED	1	MOV	A,R5
EE	1	MOV	A,R6
EF	1	MOV	A,R7
F0	1	MOVX	@DPTR,A
F1	2	ACALL	code addr
F2	1	MOVX	@R0,A
F3	1	MOVX	@R1,A
F4	1	CPL	A
F5	2	MOV	data addr,A
F6	1	MOV	@R0,A
F7	1	MOV	@R1,A
F8	1	MOV	R0,A
F9	1	MOV	R1,A
FA	1	MOV	R2,A
FB	1	MOV	R3,A
FC	1	MOV	R4,A
FD	1	MOV	R5,A
FE	1	MOV	R6,A
FF	1	MOV	R7,A