8051/8031

Single-Chip 8-Bit Microcomputer

8031 – Control oriented CPU with RAM and IO

8051 – An 8031 with factory mask-programmable ROM

DISTINCTIVE CHARACTERISTICS

- 4K x 8 ROM
- 128 x 8 RAM
- Four 8-bit ports, 32 I/O lines
- Two 16-bit timer/event counters
- High-performance full-duplex serial channel
- External memory expandable to 128K
- Compatible with 8080 and 8085 peripherals
- Boolean processor
- 8048 architecture enchanced with:
  - Non-paged jumps
  - Direct addressing
  - Four 8-register banks
  - Stack depth up to 128-bytes
  - Multiply, divide, subtract, compare
- Most instructions execute in 1μs
- 4μs multiply and divide

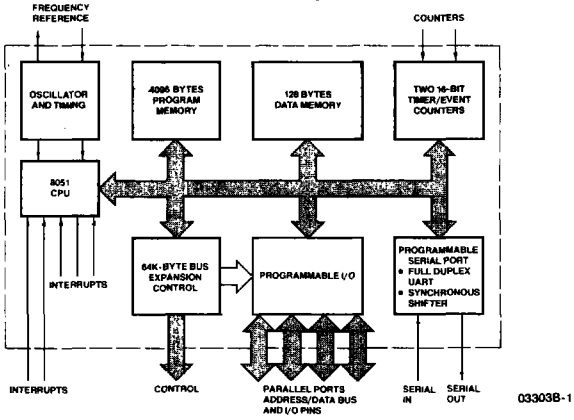
GENERAL DESCRIPTION

The 8051/8031 are members of a family of advanced single-chip microcomputers. The 8051 contains 4K x 8 read-only program memory; 128 x 8 RAM; 32 I/O lines; two 16-bit timer/counters; a five-source, two-priority-level, nested interrupt structure; a serial I/O port for either multi-processor communications, I/O expansion, or full duplex UART; and on-chip oscillator and clock circuits. The 8031 is identical, except that it lacks the program memory. For systems that require extra capability, the 8051 can be expanded using standard TTL compatible memories and the byte oriented 8080 and 8085 peripherals.

The 8051 microcomputer, like its 8048 predecessor, is efficient both as a controller and as a boolean processor. The 8051 has extensive facilities for binary and BCD arithmetic and excels in bit-handling capabilities. Efficient use of program memory results from an instruction set consisting of 44% one-byte, 41% two-byte, and 15% three-byte instructions. With a 12MHz crystal, 58% of the instructions execute in 1μs, 40% in 2μs and multiply and divide require only 4μs. Among the many instructions added to the standard 8048 instruction set are multiply, divide, subtract, and compare.

datasheetdirectory

Figure 1. Block Diagram



ORDERING INFORMATION

Operating Range	Package	12MHz w/ROM	12MHz w/o ROM
Commercial $0°C \leq T_A \leq 70°C$ $V_{CC} = +5V \pm 5\%$ $V_{SS} = 0V$	Hermetic DIP	D8051-XXXXX	D8031
	Molded DIP	P8051-XXXXX	P8031
Industrial $-40°C \leq T_A \leq 85°C$ $V_{CC} = +5V \pm 10\%$ $V_{SS} = 0V$	Hermetic DIP	ID8051-XXXXX	ID8031
Military $-55°C \leq T_A \leq 125°C$ $V_{CC} = +5V \pm 10\%$ $V_{SS} = 0V$	Hermetic DIP	MD8051B-XXXXX	MD8031B

*XXXXX is a five digit ROM code identifier assigned by factory.

9

03303B-MMP

LOGIC SYMBOL

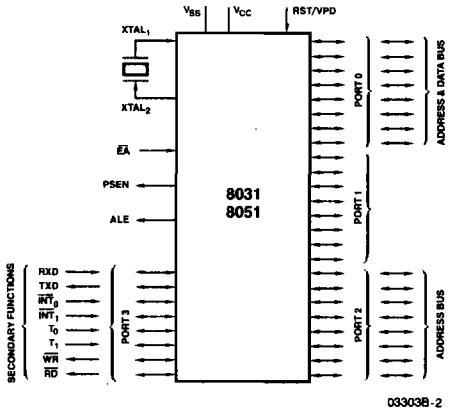


Figure 2.

CONNECTION DIAGRAM
Top View
D-40
P-40

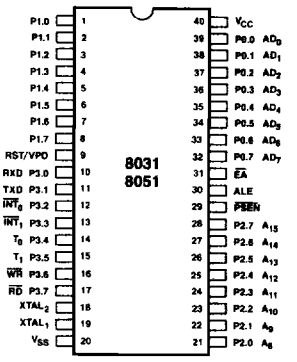


Figure 3. 03303B-3

8051 FAMILY PIN DESCRIPTION

V_{SS}

Circuit ground potential.

V_{CC}

+5V power supply during operation.

PORT 0

Port 0 is an 8-bit open drain bidirectional I/O port. It is also the multiplexed low-order address and data bus when using external memory. It is used for data output during program verification. Port 0 can sink/source eight LS TTL loads.

PORT 1

Port 1 is an 8-bit quasi-bidirectional I/O port. It is used for the low-order address byte during program verification. Port 1 can sink/source four LS TTL loads.

PORT 2

Port 2 is an 8-bit quasi-bidirectional I/O port. It also emits the high-order address byte when accessing external memory. It is used for the high-order address and the control signals during program verification. Port 2 can sink/source four LS TTL loads.

PORT 3

Port 3 is an 8-bit quasi-bidirectional I/O port. It also contains the interrupt, timer, serial port, and $\overline{RD}$ and $\overline{WR}$ pins that are used by various options. The output latch corresponding to a secondary function must be programmed to a one (1) for that function to operate. Port 3 can sink/source four LS TTL loads. The secondary functions are assigned to the pins of Port 3, as follows:

- RXD/data (P3.0). Serial port's receiver data input (asynchronous) or data input/output (synchronous).
- TXD/clock (P3.1). Serial port's transmitter data output (asynchronous) or clock output (synchronous).
- $\overline{INT_0}$ (P3.2). Interrupt 0 input or gate control input for counter 0.
- $\overline{INT_1}$ (P3.3). Interrupt 1 input or gate control input for counter 1
- T_0 (P3.4). Input to counter 0.
- T_1 (P3.5). Input to counter 1.
- $\overline{WR}$ (P3.6). The write control signal latches the data byte from Port 0 into the External Data Memory.
- $\overline{RD}$ (P3.7). The read control signal enables External Data Memory to Port 0.

RST/V_{PD}

A high level on this pin resets the 8051. If V_{PD} is held within its spec (approximately +5V), while V_{CC} drops below spec, V_{PD} will provide standby power to the RAM. When V_{PD} is low, the RAM's current is drawn from V_{CC}. A small internal resistor permits power-on reset using only a capacitor connected to V_{CC}.

ALE

Provides Address Latch Enable output used for latching the address into external memory during normal operation. It is activated every six oscillator periods, except during an external data memory access.

$\overline{PSEN}$

The Program Store Enable output is a control signal that enables the external Program Memory to the bus during external fetch operations. It is activated every six oscillator periods, except during external data memory accesses. Remains high during internal program execution.

$\overline{EA}$

When held at a TTL high level, the 8051 executes instructions from the internal ROM when the PC is less than 4096. When held at a TTL low level, the 8051 fetches all instructions from external Program Memory.

$XTAL_1$

Input to the oscillator's high gain amplifier. Required when a crystal is used. Connect to V_{SS} when external source is used on $XTAL_2$.

$XTAL_2$

Output from the oscillator's amplifier. Input to the internal timing circuitry. A crystal or external source can be used.

THE 8051 FAMILY

The 8051 is a stand-alone high-performance single-chip computer intended for use in sophisticated real-time applications such as instrumentation, industrial control, and intelligent computer peripherals. It provides the hardware features, architectural enhancements, and new instructions that make it a powerful and cost effective controller for applications requiring up to 64K bytes of program memory and/or up to 64K bytes of data storage. A Block Diagram is shown in Figure 1.

The 8031 is a control-oriented CPU without on-chip program memory. It can address 64K-bytes of External Program Memory in addition to 64K-bytes of External Data Memory. For systems requiring extra capability, each member of the 8051 Family can be expanded using standard memories and the byte oriented 8080 and 8085 peripherals. The 8051 is an 8031 with the lower 4K-bytes of Program Memory filled with on-chip mask programmable ROM.

The two pin-compatible versions of this component reduce development problems to a minimum and provide maximum flexibility. The 8051 is suited for low-cost, high volume production; and the 8031 for applications desiring the flexibility of External Program Memory which can be easily modified and updated in the field.

MACRO-VIEW OF THE 8051

On a single die the 8051 microcomputer combines CPU; 4K x 8 read-only program memory; 128 x 8 RAM; 32 I/O lines; two 16-bit timer/event counters; a five-source, two-priority-level, nested interrupt structure; serial I/O port for either multi-processor communications, I/O expansion or full duplex UART; and on-chip oscillator and clock circuits. This section will provide an overview of the 8051 by providing a high-level description of its major elements: the CPU architecture and the on-chip functions peripheral to the CPU. The generic term "8051" is used to refer collectively to the 8031 and 8051.

8051 CPU ARCHITECTURE

The 8051 CPU manipulates operands in four memory spaces. These are the 64K-byte Program Memory, 64K-byte External Data Memory, 256-byte Internal Data Memory and 16-bit Program Counter spaces. The Internal Data Memory address space is further divided into the 128-byte Internal Data RAM and 128-byte Special Function Register (SFR) address spaces shown in Figure 4. Four Register Banks (each with eight registers), 128 addressable bits and the stack reside in the Internal Data RAM. The stack depth is limited only by the available Internal Data RAM and its location is determined by the 8-bit stack pointer. All registers except the Program Counter and the four 8-Register Banks reside in the Special Function Register address space. These memory mapped registers include arithmetic registers, pointers, I/O ports, interrupt system registers, timers, and a serial port. 128 bit locations in the SFR address space are addressable as bits. The 8051 contains 128 bytes of Internal Data RAM and 20 SFRs.

The 8051 provides a non-paged Program Memory address space to accommodate relocatable code. Conditional branches are performed relative to the Program Counter. The register-indirect jump permits branching relative to a 16-bit base register with an offset provided by an 8-bit index register. Sixteen-bit jumps and calls permit branching to any location in the contiguous 64K Program Memory address space.

The 8051 has five methods for addressing source operands: Register, Direct, Register-Indirect, Immediate, and Base-Register-plus-Index-Register-Indirect Addressing. The first three methods can be used for addressing destination operands. Most instructions have a "destination, source" field that specifies the data type, addressing methods, and operands involved. For operations other than moves, the destination operand is also a source operand.

Registers in the four 8-Register Banks can be accessed through Register, Direct, or Register-Indirect Addressing; the 128 bytes of Internal Data RAM through Direct or Register-Indirect Addressing; and the Special Function Registers through Direct Addressing. External Data Memory is accessed through Register-Indirect Addressing. Look-Up-Tables resident in Program Memory can be accessed though Base-Register-plus-Index-Register-Indirect Addressing.

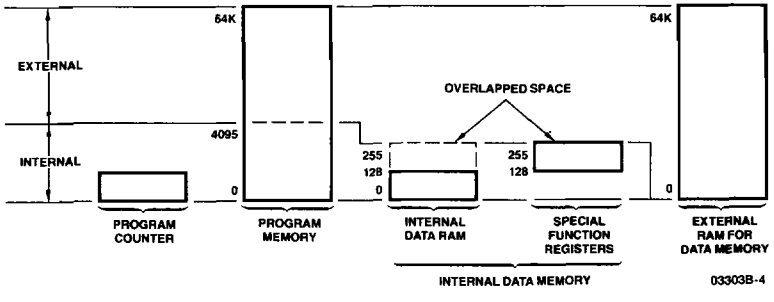
The 8051 is classified as an 8-bit machine since the internal ROM, RAM, Special Function Registers, Arithmetic/Logic Unit, and external data bus are each 8-bits wide. The 8051 performs operations on bit, nibble, byte, and double-byte data types.

The 8051 has extensive facilities for byte transfer, logic, and integer arithmetic operations. It excels at bit handling since data transfer, logic, and conditional branch operations can be performed directly on Boolean variables.

8051 INSTRUCTION SET

The 8051's instruction set is an enhancement of the instruction set familiar to 8048 users. It is enhanced to allow expansion of on-chip CPU peripherals and to optimize byte efficiency and execution speed. Op codes were reassigned to add new high-power operations and to permit new addressing modes which make the old operations more orthogonal. Efficient use of program memory results from an instruction set consisting of 49 single-byte, 45 two-byte, and 17 three-byte instructions. When using a 12MHz oscillator, 64 instructions execute in 1μs and 45

Figure 4. 8051 Family Memory Organization



instructions execute in 2μs. The remaining instructions (multiply and divide) execute in only 4μs. The number of bytes in each instruction and the number of cycles required for execution are listed in Table 1 on pages 14 and 15.

ON-CHIP PERIPHERAL FUNCTIONS

Thus far only the CPU and memory spaces of the 8051 have been described. In addition to the CPU and memories, an interrupt system, extensive I/O facilities, and several peripheral functions are integrated on-chip to relieve the CPU of repetitious, complicated, or time-critical tasks and to permit stringent real-time control of external system interfaces. The extensive I/O facilities include the I/O pins, parallel I/O ports, bidirectional address/data bus, and the serial port for I/O expansion. The CPU peripheral functions integrated on-chip are the two 16-bit counters and the serial port. All of these work together to greatly boost system performance.

INTERRUPT SYSTEM

External events and the real-time-driven on-chip peripherals require service by the CPU asynchronous to the execution of any particular section of code. To tie the asynchronous activities of these functions to normal program execution, a sophisticated multiple-source, two-priority-level, nested interrupt system is provided. Interrupt response latency ranges from 3μs to 7μs when using a 12MHz crystal.

The 8051 acknowledges interrupt requests from five sources: Two from external sources via the $\overline{INT}_0$ and $\overline{INT}_1$ pins, one from each of the two internal counters and one from the serial I/O port. Each interrupt vectors to a separate location in Program Memory for its service program. Each of the five sources can be assigned to either of two priority levels and can be independently enabled and disabled. Additionally all enabled sources can be globally disabled or enabled. Each external interrupt is programmable as either level- or transition-activated and is active-low to allow the "wire or-ing" of several interrupt sources to the input pin. The interrupt system is shown diagrammatically in Figure 5.

I/O FACILITIES

The 8051 has instructions that treat its 32 I/O lines as 32 individually addressable bits and as four parallel 8-bit ports addressable as Ports 0, 1, 2, and 3. Ports 0, 2, and 3 can also assume other functions. Port 0 provides the multiplexed low-order address and data bus used for expanding the 8051 with standard memories and peripherals. Port 2 provides the high-order address bus when expanding the 8051 with external Program Memory or more than 256 bytes of External Data Memory. The pins of Port 3 can be configured individually to provide external interrupt request inputs, counter inputs, the serial port's receiver input and transmitter output, and to generate the control signals used for reading and writing External Data Memory. The generation or use of an alternate function on a Port 3 pin is done automatically by the 8051 as long as the pin is configured as an input. The configuration of the ports is shown on the 8051 Family Logic Symbol of Figure 2.

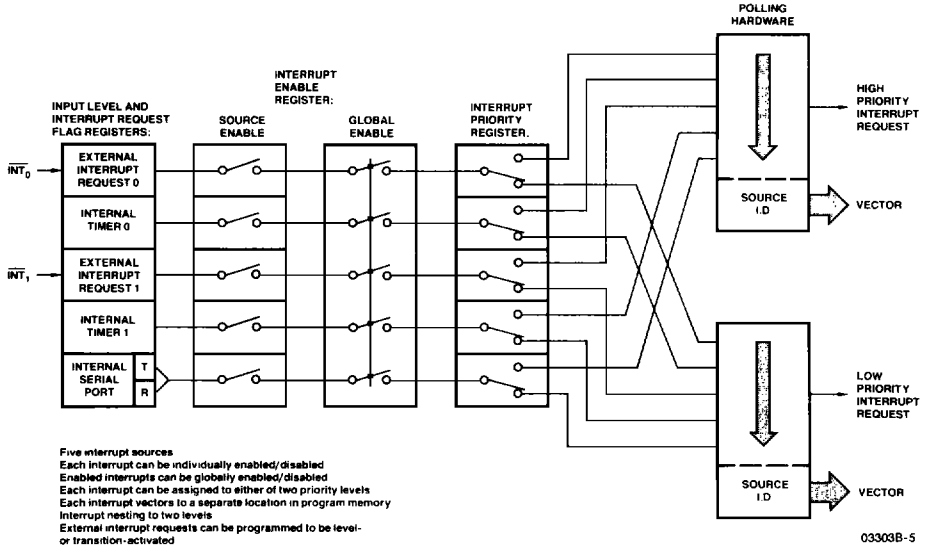
Open Drain I/O Pins

Each pin of Port 0 can be configured as an open drain output or as a high-impedance input. Resetting the microcomputer programs each pin as an input by writing a one (1) to the pin. If a zero (0) is later written to the pin it becomes configured as an output and will continuously sink current. Rewriting the pin to a one (1) will place its output driver in a high-impedance state and configure the pin as an input. Each I/O pin of Port 0 can sink/source eight LS TTL loads.

Quasi-Bidirectional I/O Pins

Ports 1, 2, and 3 are quasi-bidirectional buffers. Resetting the microcomputer programs each pin as an input by writing a one (1) to the pin. If a zero (0) is later written to the pin it becomes configured as an output and will continuously sink current. Any pin that is configured as an output will be reconfigured as an input when a one (1) is written to the pin. Simultaneous to this reconfiguration, the output driver of the quasi-bidirectional port will source current for two oscillator periods. Since current is sourced only when a bit previously written to a zero (0) is up-

Figure 5. 8051 Interrupt System



dated to a one (1), a pin programmed as an input will not source current into the TTL gate that is driving it if the pin is later written with another one (1). Since the quasi-bidirectional output driver sources current for only two oscillator periods, an internal pull-up resistor of approximately 20K- to 40K-ohms is provided to hold the external driver's loading at a TTL high level. Ports 1, 2, and 3 can sink/source four LS TTL loads.

Microprocessor Bus

A microprocessor bus is provided to permit the 8051 to solve a wide range of problems and to allow the upward growth of user products. This multiplexed address and data bus provides an interface compatible with standard memories, 8080 peripherals, and the 8085 compatible memories that include on-chip programmable I/O ports and timing functions. These are summarized in the 8051 Microcomputer Expansion Components chart of Figure 6.

When accessing external memory the high-order address is emitted on Port 2 and the low-order address on Port 0. The ALE signal is provided for strobing the address into an external latch. The program store enable ($\overline{PSEN}$) signal is provided for enabling an external memory device to Port 0 during a read from the Program Memory address space. When the MOVX instruction is executed Port 3 automatically generates the read ($\overline{RD}$) signal for enabling an External Data Memory device to Port 0 or generates the write ($\overline{WR}$) signal for strobing the external memory device with the data emitted by Port 0. Port 0 emits the address and data to the external memory through a push/pull driver that can sink/source eight LS TTL loads. At the end of the read/write bus cycle, Port 0 is automatically reprogrammed to its high impedance state and Port 2 is returned to the state it had prior to the bus cycle. The 8051 generates the address, data, and control signals needed by memory and I/O devices in a manner that minimizes the requirements placed on external program and data memories. At 12MHz, the Program Memory cycle time is 500ns and the access times required from stable address and $\overline{PSEN}$ are approximately 320ns and 150ns respectively. The External Data Memory cycle time is 1μs and the access times required for stable address from read ($\overline{RD}$) or write ($\overline{WR}$) command are approximately 600ns and 250ns respectively.

TIMER/EVENT COUNTERS

The 8051 contains two 16-bit counters for measuring timing events and pulse widths, for counting events, as well as for generating precise, periodic interrupt requests. Each can be programmed independently to one of the following three modes:

Mode 0 – similar to an 8048 8-bit timer or counter with divide by 32 prescaler.

Mode 1 – 16-bit time-interval or event counter.

Mode 2 – 8-bit time-interval or event counter with automatic reload upon overflow.

Additionally, counter 0 can be programmed to a mode that divides it into one 8-bit time-interval or event counter and one 8-bit time-interval counter (Mode 3). When counter 0 is in Mode 3, counter 1 can be programmed to any of the three aforementioned modes, although it cannot set an interrupt request flag or generate an interrupt. This mode is useful because counter 1's overflow can be used to pulse the serial port's transmission-rate generator. Along with their multiple operating modes and 16-bit precision, the counters can also handle very high input frequencies. These range from 0.1MHz to 1.0MHz (from 1.2MHz to 12MHz crystal) when programmed for an input that is a division by 12 of the oscillator frequency and from 0Hz to an upper limit of 50KHz to 0.5MHz (from 1.2MHz to 12MHz crystal) when programmed for external inputs. Both internal and external inputs can be gated to the counter by a second external source for directly measuring pulse widths.

Figure 6. 8051 Microcomputer Expansion Components

Category	AMD Part No.	Description	Comments	Program or Data Memory	Crystal Frequency MHz (Max)
Standard EPROMs	2708	1K x 8 450ns Light Erasable	User programmable and erasable.	P	7
	2716-1	2K x 8 350ns Light Erasable		P	8
	2732	4K x 8 450ns Light Erasable		P	8
	2732A-2	4K x 8 200ns Light Erasable		P	12
Standard RAMs	2114A	1K x 4 100ns RAM	Data memory can be easily expanded using standard NMOS RAMs.	D	12
	2148	1K x 4 70ns RAM		D	12
Standard I/O	8212	8-Bit I/O Port	Serves as Address Latch or I/O port.	D	12
	8255A	Programmable Peripheral Interface	Three 8-bit programmable I/O ports.	D	12
	8251A	Programmable Communications Interface	Serial Communications Receiver/Transmitter.	D	12
Standard Peripherals	8286	Bi-directional Bus Driver	8080 and 8085 peripheral devices are compatible with the 8051 allowing easy addition of specialized interfaces.	D	12
	8287	Bi-directional Bus Driver (Inverting)		D	12
	8253A	Programmable Interval Timer		D	12
	8279	Programmable Keyboard/Display Interface (128 Keys)		D	12
Universal Peripheral Interfaces	8041A	ROM Program Memory	Mask programmable to perform custom I/O and control functions	D/P	12/11.7
Memories with on-chip I/O and Peripheral Functions.	8155-2	256 x 8 330ns RAM		D	12

9

The counters are started and stopped under software control. Each counter sets its interrupt request flag when it overflows from all ones to all zeroes (or auto-reload value). The operating modes and input sources are summarized in Figures 7 and 8. The effects of the configuration flags and the status flags are shown in Figures 9 and 10.

SERIAL COMMUNICATIONS

The 8051's serial I/O port is useful for serially linking peripheral devices as well as multiple 8051s through standard asynchronous protocols with full-duplex operation. The serial port also has a synchronous mode for expansion of I/O lines using CMOS and TTL shift registers. This hardware serial communications interface saves ROM code and permits a much higher transmission rate than could be achieved through software. In response to a serial port interrupt request, the CPU has only to read/write the serial port's buffer to service the serial link. A block diagram of the serial port is shown in Figures 11 and 12. Methods for linking UART (universal asynchronous receiver/transmitter) devices are shown in Figure 13 and a method for I/O expansion is shown in Figure 14.

The full-duplex serial I/O port provides asynchronous modes to facilitate communications with standard UART devices, such as printers and CRT terminals, or communications with other 8051s in multi-processor systems. The receiver is double buffered to eliminate the overrun that would occur if the CPU failed to respond to the receiver's interrupt before the beginning of the next frame. The 8051 can generally maintain the serial link at its maximum rate so double buffering of the transmitter is not needed. A minor degradation in transmission rate can occur in rare events such as when the servicing of the transmitter has to wait for a lengthy interrupt service program to complete. In asynchronous modes, false start-bit rejection is provided on received frames. For noise rejection a best two-out-of-three vote is taken on three samples near the center of each received bit.

When interfacing with standard UART devices, the serial channel can be programmed to Mode 1 which transmits/receives a ten-bit frame or programmed to Mode 2 or 3 which transmits/receives an eleven-bit frame as shown in Figure 15. The frame consists of a start bit, eight or nine data bits, and one stop bit. In Modes 1 and 3, the transmission-rate timing circuitry receives a pulse from counter 1 each time the counter overflows. The input to counter 1 can be an external source or a division by 12 of the oscillator frequency. The auto-reload mode of the counter provides communication rates of 122 to 31,250 bits per second (including start and stop bits) for a 12MHz crystal. In Mode 2 the communication rate is a division by 64 of the oscillator frequency yielding a transmission rate of 187,500 bits per second (including start and stop bits) for a 12MHz crystal.

Figure 7. Timer/Event Counter Modes 0, 1 and 2

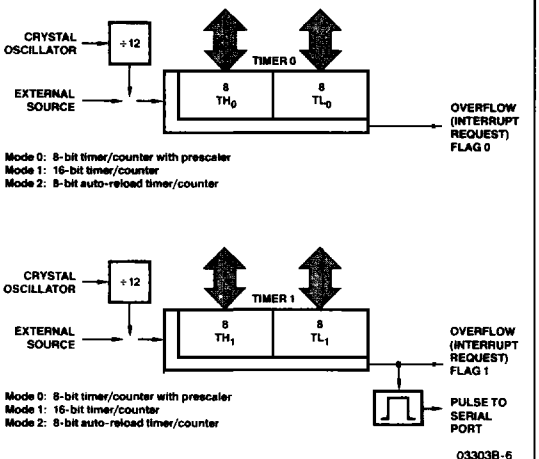


Figure 8. Timer/Event Counter 0 in Mode 3

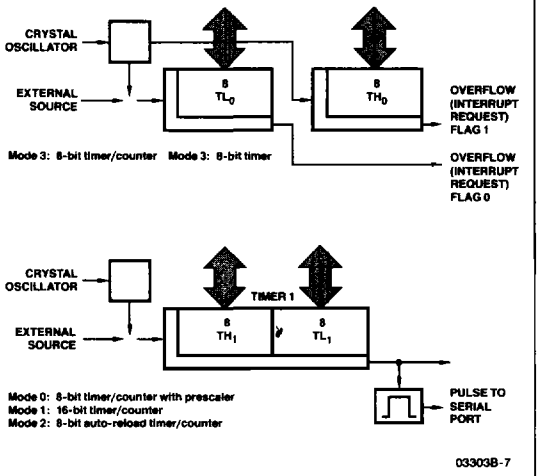


Figure 9. Timer/Counter 0 Control and Status Flag Circuitry

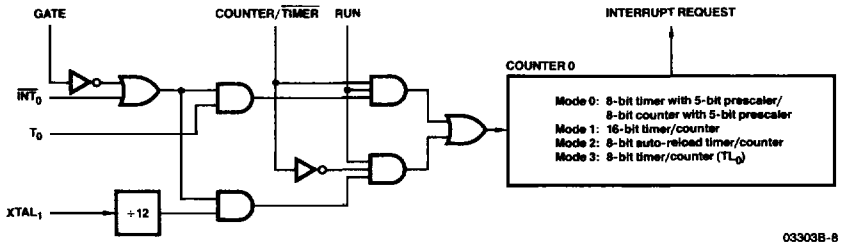


Figure 10. Timer/Counter 1 Control and Status Flag Circuitry

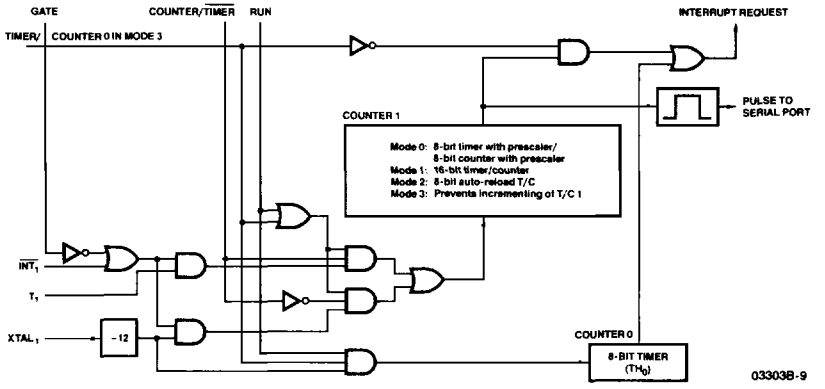


Figure 11. Serial Port – Synchronous Mode 0

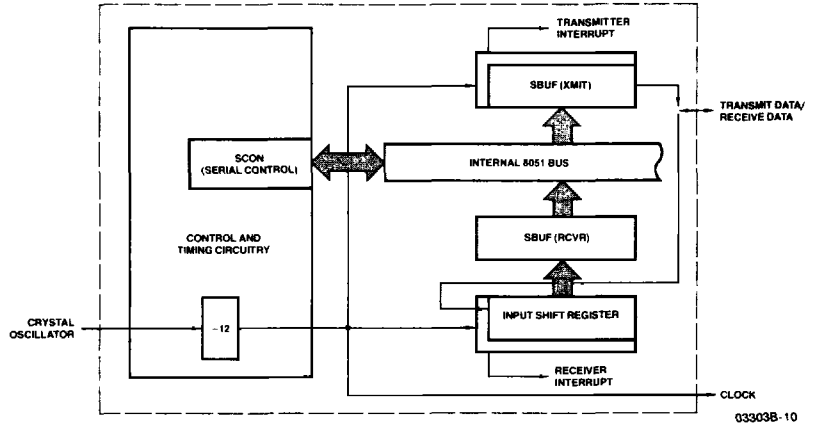
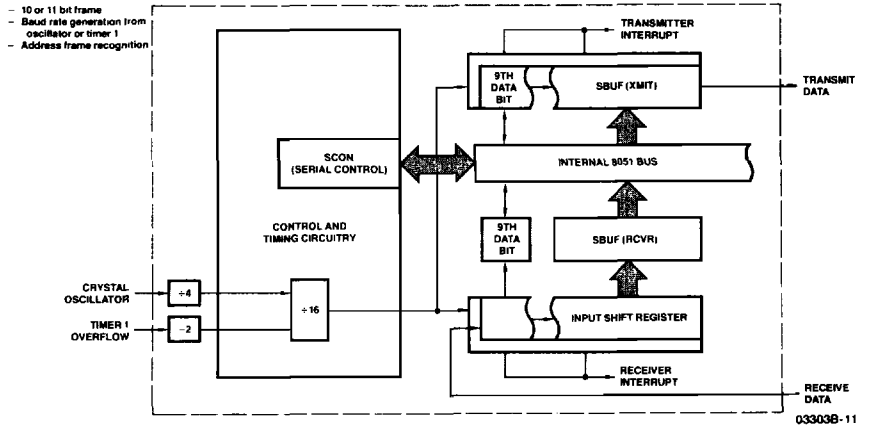


Figure 12. Serial Port – UART Modes 1, 2 and 3



9

Figure 13. UART Interfacing Schemes

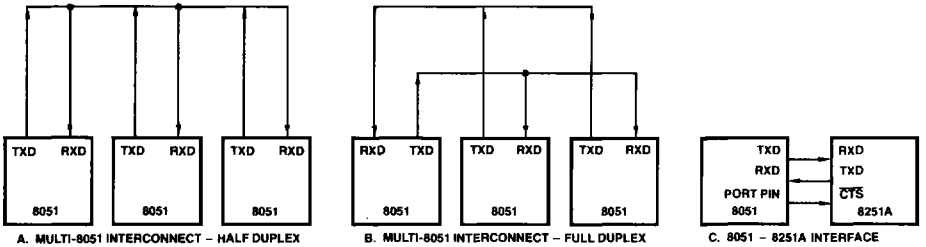


Figure 14. I/O Expansion Technique

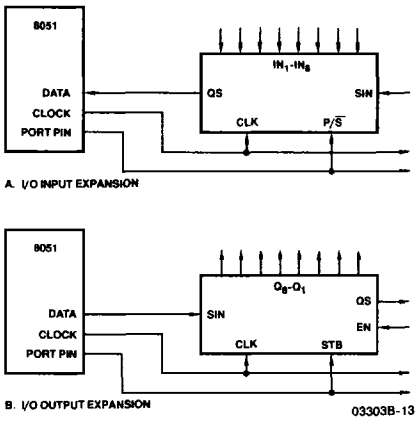
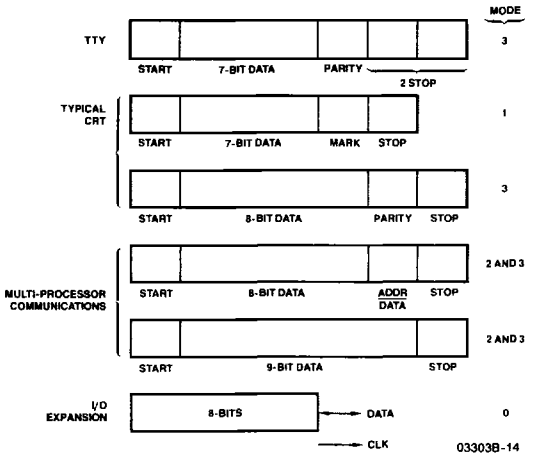


Figure 15. Typical Frame Formats



Distributed processing offers a faster, more powerful system than a single CPU can provide. This results from a hierarchy of interconnected processors, each with its own memories and I/O. In a multiprocessing environment, a single host 8051 controls other slave 8051s configured to operate simultaneously on separate portions of a program. The interconnected 8051s reduce the load on the host processor and result in a lower-cost system of data transmission. This form of distributed processing is especially effective in a complex process where controls are required at physically separated locations.

In Modes 2 and 3 interprocessor communication is facilitated by the automatic wake-up of slave processors through interrupt driven address-frame recognition. The protocol for interprocessor communications is shown in Figure 16. In synchronous mode (Mode 0) the high speed serial port provides an efficient, low-cost method of expanding I/O lines using standard TTL and CMOS shift registers. The serial channel provides a clock output for synchronizing the shifting of bits to/from an external register. The data rate is a division by 12 of the oscillator frequency and hence is 1M bits per second at 12MHz.

Figure 16. Protocol for Multi-Processor Communications

1. Slaves – Configure serial port to interrupt CPU if the received ninth data bit is a one (1).
2. Master – Transmit frame containing address in first 8 data bits and set ninth data bit (i.e., ninth data bit designates address frame).
3. Slaves – Serial port interrupts CPU when address frame is received. Interrupt service program compares received address to its address. The slave which has been addressed reconfigures its serial port to interrupt the CPU on all subsequent transmissions.
4. Master – Transmit control frames and data frames (these will be accepted only by the previously addressed slave).

ABSOLUTE MAXIMUM RATINGS above which useful life may be impaired*

Storage Temperature	−65 to +150°C
Voltage on Any Pin with Respect to Ground	−0.5 to +7.0V
Power Dissipation	2W

The products described by this specification include internal circuitry designed to protect input devices from damaging accumulations of static charge. It is suggested, nevertheless, that conventional precautions be observed during storage, handling, and use in order to avoid exposure to excessive voltages.

*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.

8051/8031 DC CHARACTERISTICS

Parameters	Description	Test Conditions	Min	Typ	Max	Units
V_{IL}	Input Low Voltage		−0.5		0.8	V
V_{IH}	Input High Voltage (Except RST/V_{PD} and $XTAL_2$)		2.0		V_{CC} +0.5	V
V_{IH1}	Input High Voltage to RST/V_{PD} for Reset, $XTAL_2$	$XTAL_1$ to V_{SS}	2.5			V
V_{PD}	Power Down Voltage to RST/V_{PD}	V_{CC} = 0V	4.5		5.5	V
V_{OL}	Output Low Voltage, Ports 1, 2, 3 (Note 1)	I_{OL} = 1.6mA			0.45	V
V_{OL1}	Output Low Voltage, Port 0, ALE, $\overline{PSEN}$ (Note 1)	I_{OL} = 3.2mA			0.45	V
V_{OH}	Output High Voltage, Ports 1, 2, 3	I_{OH} = −80μA	2.4			V
V_{OH1}	Output High Voltage, Port 0, ALE, $\overline{PSEN}$	I_{OH} = −400μA	2.4			V
I_{IL}	Logical 0 Input Current, $XTAL_2$, Ports 1, 2, 3	$XTAL_1$ at V_{SS} V_{IL} = 0.45V			−800	μA
I_{IH1}	Input High Current to RST/V_{PD} for Reset	V_{IN} = V_{CC} − 1.5V			500	μA
I_{LI}	Input Leakage Current to Port 0, $\overline{EA}$	0 < V_{IN} < V_{CC}			10	μA
I_{CC}	Power Supply Current			125	160	mA
I_{PD}	Power Down Current			10	20	mA
C_{IO}	Capacitance of I/O Buffer	fc = 1MHz			10	pF

Note 1. V_{OL} *is degraded when the 8051 rapidly discharges external capacitance. This AC noise is most pronounced during emission of address data. When using external memory, locate the latch or buffer as close to the 8051 as possible.*

8051 PROGRAM VERIFICATION

To ensure correct factory masked ROM, the following procedure may be followed. The address of the location to be verified is input on Port 1 (pins 1 through 8) while Port 2 (pins 21 through 28) and PSEN (pin 29) are held low. RST/VPD (pin 9) and ALE (pin 30) are held high. The data to be verified is read out through Port 0 (pins 32 through 39).

Datum	Emitting Ports	Time Interval	Degraded I/O Lines	V_{OL} (peak) (max)
Address	P2, P0	T3, T9	P1, P3	0.8V
Write Data	P0	T6	P1, P3, ALE	0.8V

9

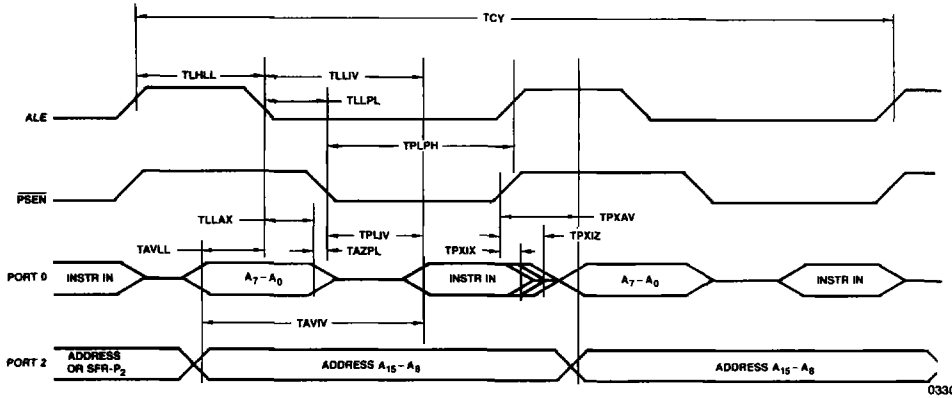
8051/8031 AC CHARACTERISTICS (C_L for Port 0, ALE and $\overline{PSEN}$ Outputs = 100pF; C_L for All Other Outputs = 80pF)

Parameter	Description	12MHz Clock		Variable Clock 1/TCLCL = 1.2MHz to 12MHz		Units
		Min	Max	Min	Max	
PROGRAM MEMORY						
TCY	Min Instruction Cycle Time (Note 3)	1.0		12TCLCL	12TCLCL	ns
TLHLL	ALE Pulse Width	127		2TCLCL-40		ns
TAVLL	Address Setup to ALE	53		TCLCL-30		ns
TLLAX	Address Hold After ALE (Note 1)	48		TCLCL-35		ns
TLLIV	ALE to Valid Instruction In		233		4TCLCL-100	ns
TLLPL	ALE to $\overline{PSEN}$	58		TCLCL-25		ns
TPLPH	$\overline{PSEN}$ Pulse Width	215		3TCLCL-35		ns
TPLIV	$\overline{PSEN}$ to Valid Instruction In		125		3TCLCL-125	ns
TPXIX	Input Instruction Hold After $\overline{PSEN}$	0		0		ns
TPXIZ	Input Instruction Float After $\overline{PSEN}$ (Note 2)		63		TCLCL-20	ns
TPXAV	Address Valid After $\overline{PSEN}$ (Note 2)	75		TCLCL-8		ns
TAVIV	Address to Valid Instruction In		302		5TCLCL-115	ns
TAZPL	Address Float to $\overline{PSEN}$	0		0		ns
EXTERNAL DATA MEMORY						
TRLRH	$\overline{RD}$ Pulse Width	400		6TCLCL-100		ns
TWLWH	$\overline{WR}$ Pulse Width	400		6TCLCL-100		ns
TLLAX	Address Hold After ALE (Note 1)	132		2TCLCL-35		ns
TRLDV	$\overline{RD}$ to Valid Data In		250		5TCLCL-165	ns
TRHDX	Data Hold After $\overline{RD}$	0		0		ns
TRHDZ	Data Float After $\overline{RD}$		97		2TCLCL-70	ns
TLLDV	ALE to Valid Data In		517		8TCLCL-150	ns
TAVDV	Address to Valid Data In		585		9TCLCL-165	ns
TLLWL	ALE to $\overline{WR}$ or $\overline{RD}$	200	300	3TCLCL-50	3TCLCL+50	ns
TAVWL	Address to $\overline{WR}$ or $\overline{RD}$	203		4TCLCL-130		ns
TWHLH	$\overline{WR}$ or $\overline{RD}$ High to ALE High	43	123	TCLCL-40	TCLCL+40	ns
TDVWX	Data Valid to $\overline{WR}$ Transition	33		TCLCL-50		ns
TQVWH	Data Setup Before $\overline{WR}$	433		7TCLCL-150		ns
TWHQX	Data Hold After $\overline{WR}$	33		TCLCL-50		ns
TRLAZ	Address Float After $\overline{RD}$		0		0	ns

Notes: 1. TLLAX for access to Program Memory is different from TLLAX for access to Data Memory.
2. Interfacing the 8051 to devices with float times up to 75ns is permissible. This limited bus contention will not cause any damage to Port 0 drivers.
3. TCY is the minimum instruction cycle time which consists of 12 oscillator clocks or two ALE cycles.

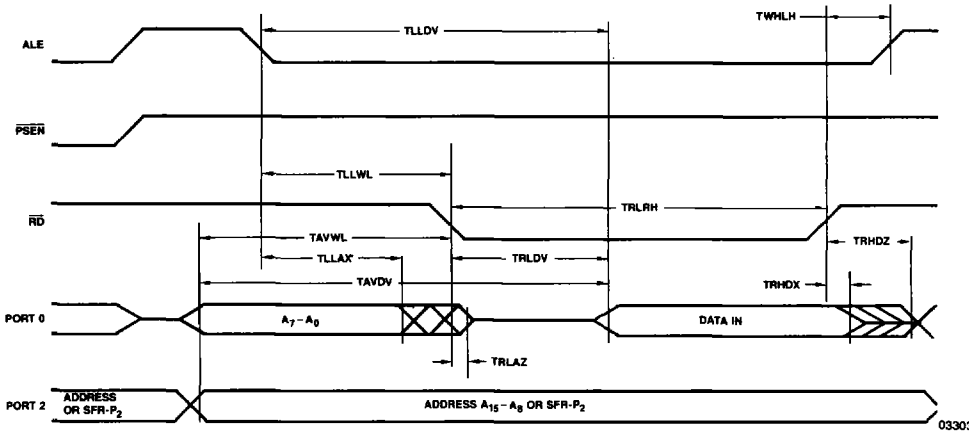
WAVEFORMS

PROGRAM MEMORY READ CYCLE



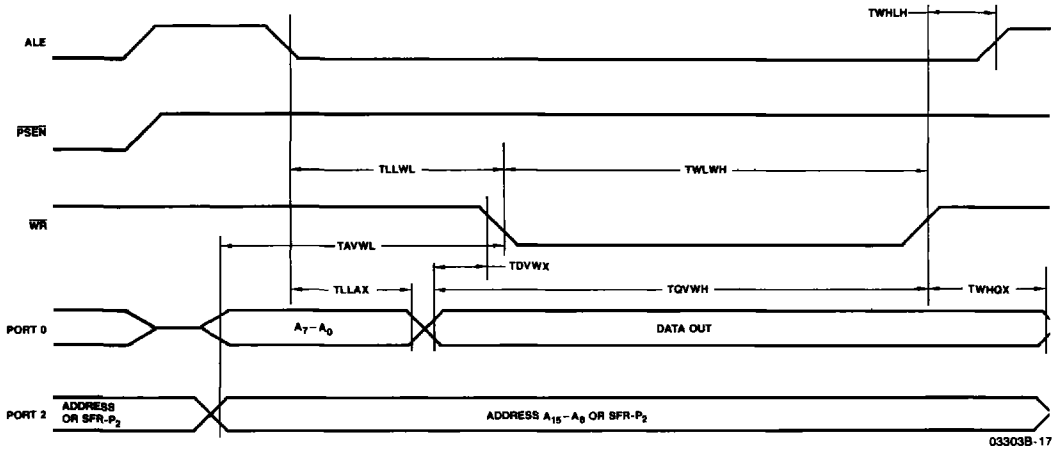
03303B-15

DATA MEMORY READ CYCLE



03303B-16

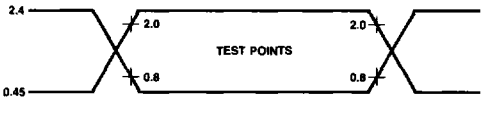
DATA MEMORY WRITE CYCLE



03303B-17

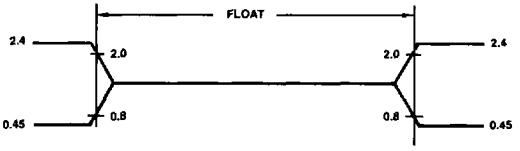
AC TESTING INPUT, OUTPUT, FLOAT WAVEFORMS

INPUT/OUTPUT



03303B-18

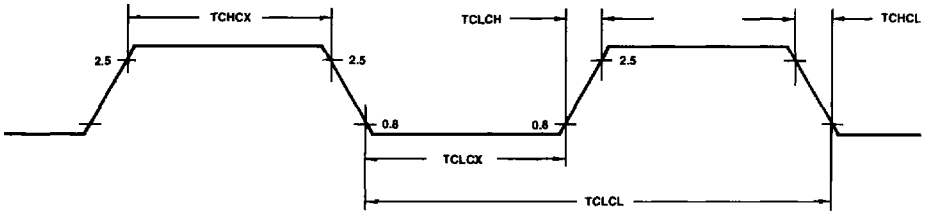
FLOAT



03303B-19

AC testing inputs 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 P_0 pin sinks 3.2mA or sources 400μA at the voltage test levels.

EXTERNAL CLOCK DRIVE $XTAL_2$

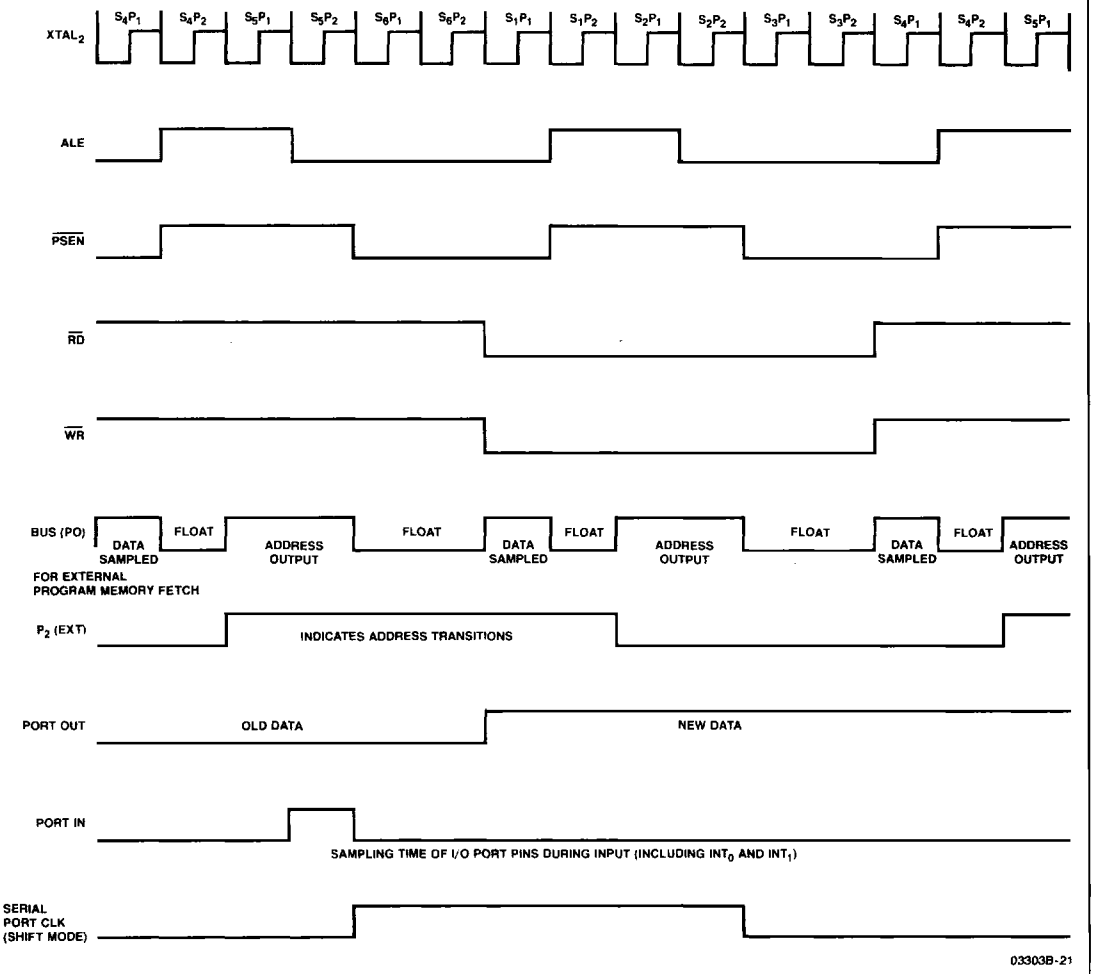


03303B-20

CLOCKING DETAILS

Symbol	Parameters	Variable Clock Freq = 1.2MHz to 12MHz		Unit
		Min	Max	
TCLCL	Oscillator Period	83.3	833.3	ns
TCHCX	High Time	20	TCLCL-TCLCX	ns
TCLCX	Low Time	20	TCLCL-TCHCX	ns
TCLCH	Rise Time		20	ns
TCHCL	Fall Time		20	ns

WAVEFORMS



03303B-21

All internal timing is referenced to the internal time states shown at the top of the page. This waveform represents the signal on the X_2 input of the oscillator. This diagram represents when these signals are actually clocked within the chip. However, the time it takes a signal to propagate to the pins is in the range of 50 – 150ns. Prop delays are dependent on many variables, such as temperature, pin loading. Even the different signals vary. Typically though, /RD and /WR have prop delays of approximately 50ns and the other timing signals approximately 85ns, at room temperature, fully loaded. These differences in prop delays between signals have been integrated into the timing specs.

TABLE 1. 8051/8031 INSTRUCTION SET

INSTRUCTIONS THAT AFFECT FLAG SETTINGS*

Instruction	Flag			Instruction	Flag		
	C	OV	AC		C	OV	AC
ADD	X	X	X	CLR C	O		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C, bit	X		
MUL	O	X		ANL C,/bit	X		
DIV	O	X		ORL C, bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C, bit	X		
RLC	X			CJNE	X		
SETB C	1						

Interrupt Response Time: To finish execution of current instruction, respond to the interrupt request, push the PC and to vector to the first instruction of the interrupt service program requires 38 to 81 oscillator periods (3 to 7μs @ 12MHz).

*Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

DATA TRANSFER (Note 1)

Mnemonic		Description	Bytes	Cycle
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 byte	3	2
*MOV	direct,@Ri	Move indirect RAM to direct byte	2	2
*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	Move 16-bit constant to Data Pointer	3	2
*MOVC	A,@A+ DPTR	Move Code byte relative to DPTR to Accumulator	1	2
*MOVC	A,@A+PC	Move Code byte relative to PC to Accumulator	1	2
MOVX	A,@Ri	Move External RAM (8-bit address) to Accumulator	1	2
*MOVX	A,@DPTR	Move External RAM (16-bit address) to Accumulator	1	2
MOVX	@Ri,A	Move Accumulator to External RAM (8-bit address)	1	2
*MOVX	@DPTR,A	Move Accumulator to External RAM (16-bit address)	1	2
*PUSH	direct	Push direct byte onto stack	2	2
*POP	direct	Pop direct byte off of 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 Accumulator	1	1
XCHD	A,@Ri	Exchange indirect RAM's least sig nibble with A's LSN	1	1

*New operation not provided by 8048/8049 Family

All mnemonics copyrighted© Intel Corporation 1980.

BOOLEAN VARIABLE MANIPULATION

Mnemonic		Description	Byte	Cycle
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		
*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

LOGIC

Mnemonic		Description	Bytes	Cycle
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 Accumulator	1	1
XRL	A,#data	Exclusive-OR immediate data to Accumulator	2	1
*XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	1

Note 1: Special care should be taken (particularly with the 8031) when using the MOV instruction. The MOV instruction should *not* be used to move data on port 0 nor 2 when these ports are used to address external memory.

TABLE 1. 8051/8031 INSTRUCTION SET (Cont.)

LOGIC (Cont.)

Mnemonic		Description	Bytes	Cycle
*XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	2
CLR	A	Clear Accumulator	1	1
CPL	A	Complement Accumulator	1	1
RL	A	Rotate Accumulator Left	1	1
RLC	A	Rotate Accumulator Left through Carry Flag	1	1
RR	A	Rotate Accumulator Right	1	1
RRC	A	Rotate Accumulator Right through Carry Flag	1	1
SWAP	A	Exchange nibbles within Accumulator	1	1

ARITHMETIC

Mnemonic		Description	Bytes	Cycle
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 Accumulator with Carry Flag	2	1
ADDC	A,@Ri	Add indirect RAM and Carry Flag to Accumulator	1	1
ADDC	A,#data	Add immediate data and Carry Flag to Accumulator	2	1
*SUBB	A,Rn	Subtract register from Accumulator with Borrow	1	1
*SUBB	A,direct	Subtract direct byte from Accumulator with Borrow	2	1
*SUBB	A,@Ri	Subtract indirect RAM from Accumulator with Borrow	1	1
*SUBB	A,#data	Subtract immediate data from Accumulator 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
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
*INC	DPTR	Increment Data Pointer	1	2
*MUL	AB	Multiply Accumulator times B	1	4
*DIV	AB	Divide Accumulator by B	1	4
DA	A	Decimal Adjust Accumulator	1	1

OTHER

Mnemonic	Description	Bytes	Cycle
NOP	No Operation	1	1

CONTROL TRANSFER (BRANCH)

Mnemonic		Description	Bytes	Cycle
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 carry is not set	2	2
*JB	bit,rel	Jump relative if direct bit is set	3	2
*JNB	bit,rel	Jump relative if direct bit is not set	3	2
*JBC	bit,rel	Jump relative if direct bit is set, then clear bit	3	2
*CJNE	A,direct,rel	Compare direct byte to Accumulator and Jump if not Equal	3	2
*CJNE	A,#data,rel	Compare immediate to Accumulator and Jump if not Equal	3	2
*CJNE	Rn,#data, rel	Compare immediate to reg and Jump if not Equal	3	2
*CJNE	@Ri, #data,rel	Compare immed. to indirect RAM and Jump if not Equal	3	2
DJNZ	Rn,rel	Decrement register and Jump if not zero	2	2
*DJNZ	direct,rel	Decrement direct byte and Jump if not zero	3	2

CONTROL TRANSFER (SUBROUTINE)

Mnemonic		Description	Bytes	Cycle
ACALL	addr11	Absolute Subroutine Call	2	2
LCALL	addr16	Long Subroutine Call	3	2
RET		Return from Subroutine Call	1	2
RETI		Return from Interrupt Call	1	2

Notes on data addressing modes:

- Rn – Working register R0 – R7 of the currently selected Register bank.
- 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 and 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 and LJMP may be anywhere within the 64-Kilobyte program memory address space.
- addr11 – Destination address for ACALL and AJMP will be within the same 2-Kilobyte 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.

*New operation not provided by 8048/8049 Family

9

TABLE 2. INSTRUCTION OPCODES IN HEXADECIMAL ORDER

Hex Code	Bytes	Mnemonic	Operands	Hex Code	Bytes	Mnemonic	Operands
00	1	NOP		42	2	ORL	Data addr,A
01	2	AJMP	Code addr	43	3	ORL	Data addr,#data
02	3	LJMP	Code addr	44	2	ORL	A,#data
03	1	RR	A	45	2	ORL	A,data addr
04	1	INC	A	46	1	JRL	A,@R0
05	2	INC	Data addr	47	1	ORL	A,@R1
06	1	INC	@R0	48	1	ORL	A,R0
07	1	INC	@R1	49	1	ORL	A,R1
08	1	INC	R0	4A	1	ORL	A,R2
09	1	INC	R1	4B	1	ORL	A,R3
0A	1	INC	R2	4C	1	ORL	A,R4
0B	1	INC	R3	4D	1	ORL	A,R5
0C	1	INC	R4	4E	1	ORL	A,R6
0D	1	INC	R5	4F	1	ORL	A,R7
0E	1	INC	R6	50	2	JNC	Code addr
0F	1	INC	R7	51	2	ACALL	Code addr
10	3	JBC	Bit addr,code addr	52	2	ANL	Data addr,A
11	2	ACALL	Code addr	53	3	ANL	Data addr,#data
12	3	LCALL	Code addr	54	2	ANL	A,#data
13	1	RRC	A	55	2	ANL	A,data addr
14	1	DEC	A	56	1	ANL	A,@R0
15	2	DEC	Data addr	57	1	ANL	A,@R1
16	1	DEC	@R0	58	1	ANL	A,R0
17	1	DEC	@R1	59	1	ANL	A,R1
18	1	DEC	R0	5A	1	ANL	A,R2
19	1	DEC	R1	5B	1	ANL	A,R3
1A	1	DEC	R2	5C	1	ANL	A,R4
1B	1	DEC	R3	5D	1	ANL	A,R5
1C	1	DEC	R4	5E	1	ANL	A,R6
1D	1	DEC	R5	5F	1	ANL	A,R7
1E	1	DEC	R6	60	2	JZ	Code addr
1F	1	DEC	R7	61	2	AJMP	Code addr
20	3	JB	Bit addr,code addr	62	2	XRL	Data addr,A
21	2	AJMP	Code addr	63	3	XRL	Data addr,#data
22	1	RET		64	2	XRL	A, #data
23	1	RL	A	65	2	XRL	A,data addr
24	2	ADD	A,#data	66	1	XRL	A,@R0
25	2	ADD	A,data addr	67	1	XRL	A,@R1
26	1	ADD	A,@R0	68	1	XRL	A,R0
27	1	ADD	A,@R1	69	1	XRL	A,R1
28	1	ADD	A,R0	6A	1	XRL	A,R2
29	1	ADD	A,R1	6B	1	XRL	A,R3
2A	1	ADD	A,R2	6C	1	XRL	A,R4
2B	1	ADD	A,R3	6D	1	XRL	A,R5
2C	1	ADD	A,R4	6E	1	XRL	A,R6
2D	1	ADD	A,R5	6F	1	XRL	A,R7
2E	1	ADD	A,R6	70	2	JNZ	Code addr
2F	1	ADD	A,R7	71	2	ACALL	Code addr
30	3	JNB	Bit addr,code addr	72	2	ORL	C,bit addr
31	2	ACALL	Code addr	73	1	JMP	@A+DPTR
32	1	RETI		74	2	MOV	A,#data
33	1	RLC	A	75	3	MOV	Data addr,#data
34	2	ADDC	A,#data	76	2	MOV	@R0,#data
35	2	ADDC	A,data addr	77	2	MOV	@R1,#data
36	1	ADDC	A,@R0	78	2	MOV	R0,#data
37	1	ADDC	A,@R1	79	2	MOV	R1,#data
38	1	ADDC	A,R0	7A	2	MOV	R2,#data
39	1	ADDC	A,R1	7B	2	MOV	R3,#data
3A	1	ADDC	A,R2	7C	2	MOV	R4,#data
3B	1	ADDC	A,R3	7D	2	MOV	R5,#data
3C	1	ADDC	A,R4	7E	2	MOV	R6,#data
3D	1	ADDC	A,R5	7F	2	MOV	R7,#data
3E	1	ADDC	A,R6	80	2	SJMP	Code addr
3F	1	ADDC	A,R7	81	2	AJMP	Code addr
40	2	JC	Code addr	82	2	ANL	C,bit addr
41	2	AJMP	Code addr	83	1	MOVC	A,@A+PC

TABLE 2. INSTRUCTION OPCODES IN HEXADECIMAL ORDER (Cont.)

Hex Code	Bytes	Mnemonic	Operands	Hex Code	Bytes	Mnemonic	Operands
84	1	DIV	AB	C2	2	CLR	Bit addr
85	3	MOV	Data addr,data addr	C3	1	CLR	C
86	2	MOV	Data addr,@R0	C4	1	SWAP	A
87	2	MOV	Data addr,@R1	C5	2	XCH	A,data addr
88	2	MOV	Data addr,R0	C6	1	XCH	A,@R0
89	2	MOV	Data addr,R1	C7	1	XCH	A,@R1
8A	2	MOV	Data addr,R2	C8	1	XCH	A,R0
8B	2	MOV	Data addr,R3	C9	1	XCH	A,R1
8C	2	MOV	Data addr,R4	CA	1	XCH	A,R2
8D	2	MOV	Data addr,R5	CB	1	XCH	A,R3
8E	2	MOV	Data addr,R6	CC	1	XCH	A,R4
8F	2	MOV	Data addr,R7	CD	1	XCH	A,R5
90	3	MOV	DPTR,#data	CE	1	XCH	A,R6
91	2	ACALL	Code addr	CF	1	XCH	A,R7
92	2	MOV	Bit addr,C	D0	2	POP	Data addr
93	1	MOVC	A,@A+DPTR	D1	2	ACALL	Code addr
94	2	SUBB	A,#data	D2	2	SETB	Bit addr
95	2	SUBB	A,data addr	D3	1	SETB	C
96	1	SUBB	A,@R0	D4	1	DA	A
97	1	SUBB	A,@R1	D5	3	DJNZ	Data addr,code addr
98	1	SUBB	A,R0	D6	1	XCHD	A,@R0
99	1	SUBB	A,R1	D7	1	XCHD	A,@R1
9A	1	SUBB	A,R2	D8	2	DJNZ	R0,code addr
9B	1	SUBB	A,R3	D9	2	DJNZ	R1,code addr
9C	1	SUBB	A,R4	DA	2	DJNZ	R2,code addr
9D	1	SUBB	A,R5	DB	2	DJNZ	R3,code addr
9E	1	SUBB	A,R6	DC	2	DJNZ	R4,code addr
9F	1	SUBB	A,R7	DD	2	DJNZ	R5,code addr
A0	2	ORL	C,/bit addr	DE	2	DJNZ	R6,code addr
A1	2	AJMP	Code addr	DF	2	DJNZ	R7,code addr
A2	2	MOV	C,bit addr	E0	1	MOVX	A,@DPTR
A3	1	INC	DPTR	E1	2	AJMP	Code addr
A4	1	MUL	AB	E2	1	MOVX	A,@R0
A5		Reserved		E3	1	MOVX	A,@R1
A6	2	MOV	@R0,data addr	E4	1	CLR	A
A7	2	MOV	@R1,data addr	E5	2	MOV	A,data addr
A8	2	MOV	R0,data addr	E6	1	MOV	A,@R0
A9	2	MOV	R1,data addr	E7	1	MOV	A,@R1
AA	2	MOV	R2,data addr	E8	1	MOV	A,R0
AB	2	MOV	R3,data addr	E9	1	MOV	A,R1
AC	2	MOV	R4,data addr	EA	1	MOV	A,R2
AD	2	MOV	R5,data addr	EB	1	MOV	A,R3
AE	2	MOV	R6,data addr	EC	1	MOV	A,R4
AF	2	MOV	R7,data addr	ED	1	MOV	A,R5
B0	2	ANL	C,/bit addr	EE	1	MOV	A,R6
B1	2	ACALL	Code addr	EF	1	MOV	A,R7
B2	2	CPL	Bit addr	F0	1	MOVX	@DPTR,A
B3	1	CPL	C	F1	2	ACALL	Code addr
B4	3	CJNE	A,#data,code addr	F2	1	MOVX	@R0,A
B5	3	CJNE	A,data addr,code addr	F3	1	MOVX	@R1,A
B6	3	CJNE	@R0,#data,code addr	F4	1	CPL	A
B7	3	CJNE	@R1,#data,code addr	F5	2	MOV	Data addr,A
B8	3	CJNE	R0,#data,code addr	F6	1	MOV	@R0,A
B9	3	CJNE	R1,#data,code addr	F7	1	MOV	@R1,A
BA	3	CJNE	R2,#data,code addr	F8	1	MOV	R0,A
BB	3	CJNE	R3,#data,code addr	F9	1	MOV	R1,A
BC	3	CJNE	R4,#data,code addr	FA	1	MOV	R2,A
BD	3	CJNE	R5,#data,code addr	FB	1	MOV	R3,A
BE	3	CJNE	R6,#data,code addr	FC	1	MOV	R4,A
BF	3	CJNE	R7,#data,code addr	FD	1	MOV	R5,A
C0	2	PUSH	Data addr	FE	1	MOV	R6,A
C1	2	AJMP	Code addr	FF	1	MOV	R7,A