IS80C52
IS80C32

ISSI®

CMOS SINGLE CHIP LOW VOLTAGE 8-BIT MICROCONTROLLER

NOVEMBER 1998

FEATURES

- 80C51 based architecture
- 8K x 8 ROM (IS80C52 only)
- 256 x 8 RAM
- Three 16-bit Timer/Counters
- Full duplex serial channel
- Boolean processor
- Four 8-bit I/O ports, 32 I/O lines
- Memory addressing capability
  – 64K ROM and 64K RAM
- Program memory lock
  – Encrypted verify (32 bytes)
  – Lock bits (2)
- Power save modes:
  – Idle and power-down
- Eight interrupt sources
- Most instructions execute in 0.3 µs
- CMOS and TTL compatible
- Maximum speed: 40 MHz @ Vcc = 5V
- Industrial temperature available
- Packages available:
  – 40-pin DIP
  – 44-pin PLCC
  – 44-pin PQFP

GENERAL DESCRIPTION

The *ISSI* IS80C52 and IS80C32 are high-performance microcontrollers fabricated using high-density CMOS technology. The CMOS IS80C52/32 is functionally compatible with the industry standard 8052/32 microcontrollers.

The IS80C52/32 is designed with 8K x 8 ROM (IS80C52 only); 256 x 8 RAM; 32 programmable I/O lines; a serial I/O port for either multiprocessor communications, I/O expansion or full duplex UART; three 16-bit timer/counters; an eight-source, two-priority-level, nested interrupt structure; and an on-chip oscillator and clock circuit. The IS80C52/32 can be expanded using standard TTL compatible memory.

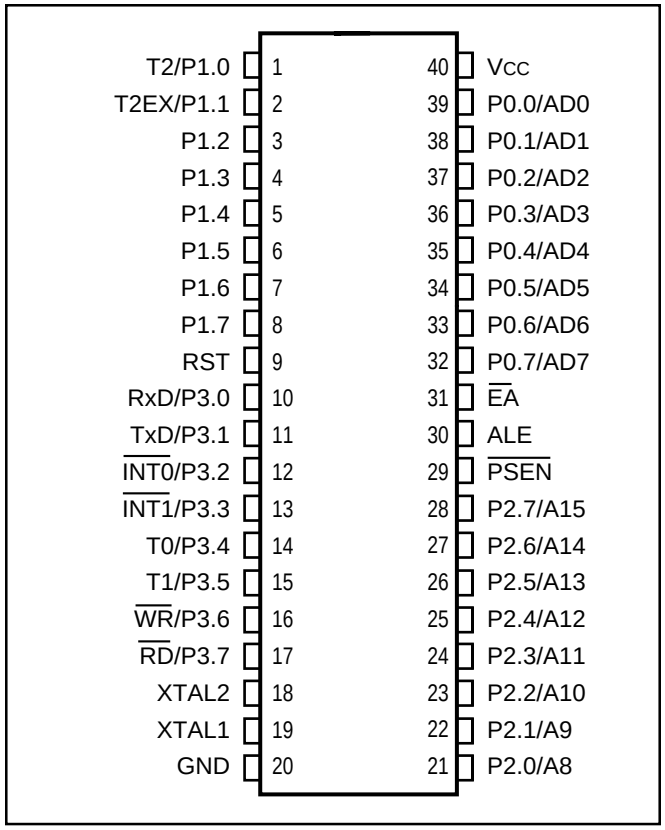

Figure 1. IS80C52/32 Pin Configuration: 40-pin PDIP

ISSI reserves the right to make changes to its products at any time without notice in order to improve design and supply the best possible product. We assume no responsibility for any errors which may appear in this publication. © Copyright 1998, Integrated Silicon Solution, Inc.

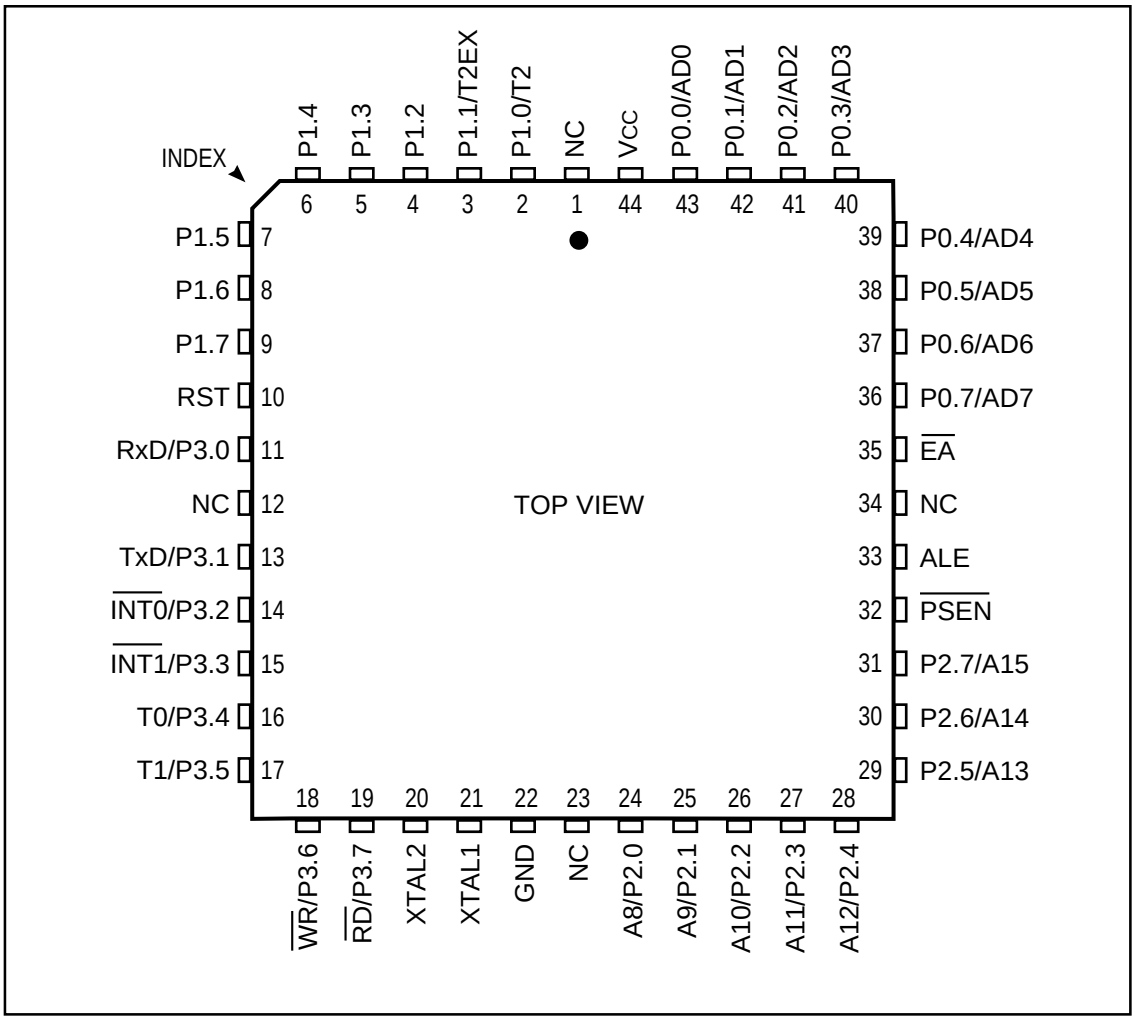

Figure 2. IS80C52/32 Pin Configuration: 44-pin PLCC

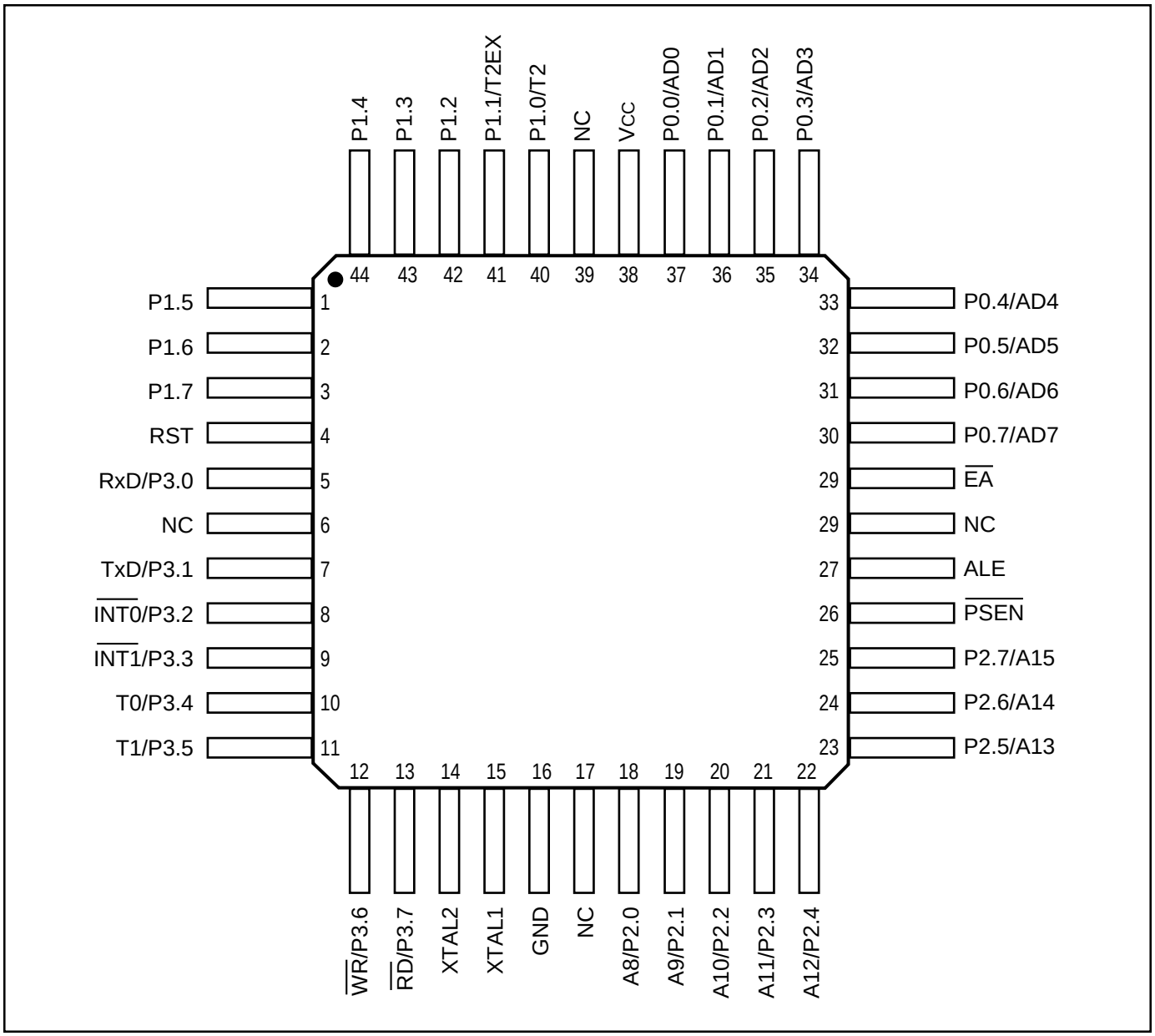

Figure 3. IS80C52/32 Pin Configuration: 44-pin PQFP

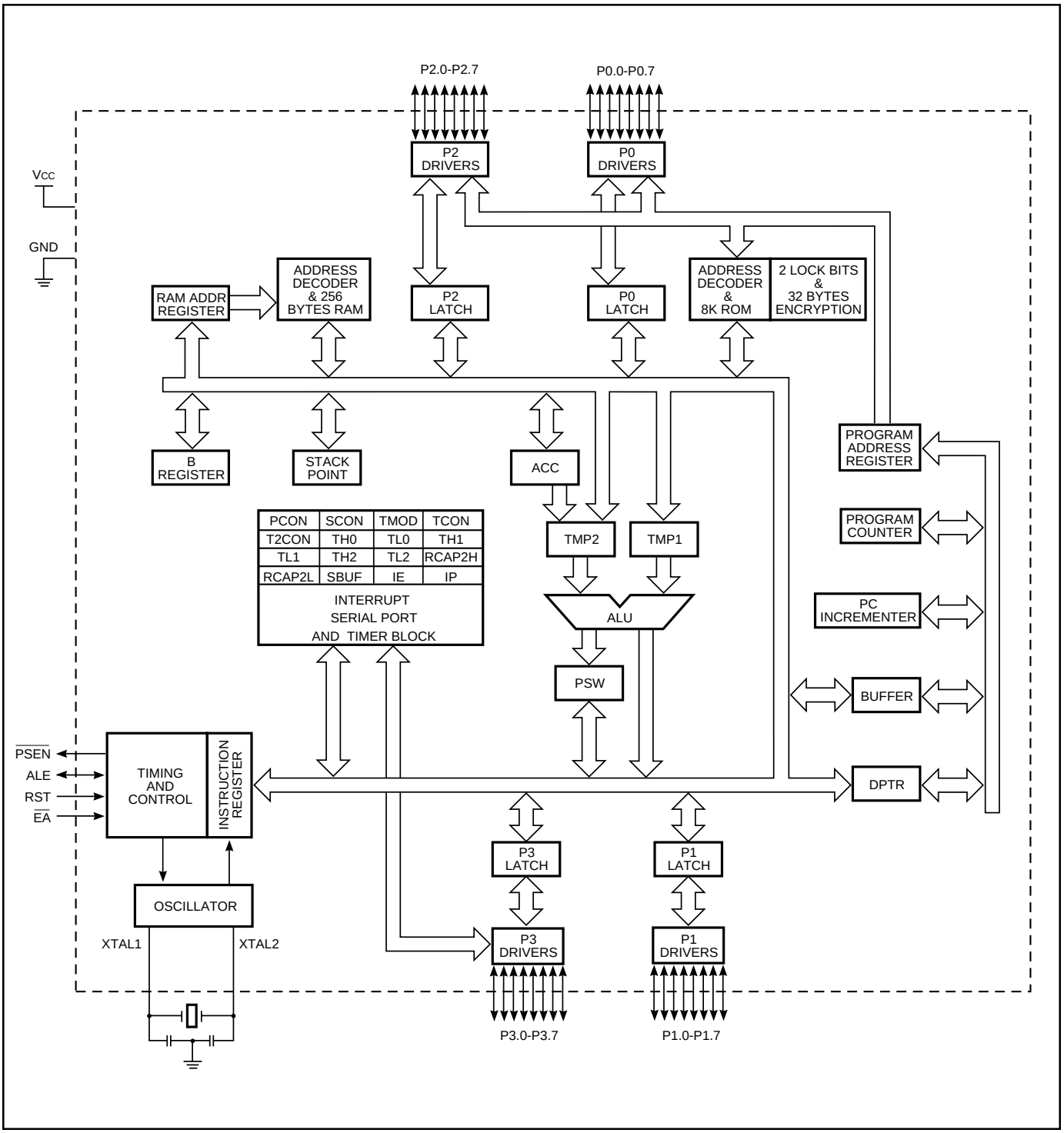

Figure 4. IS80C52/32 Block Diagram

Table 1. Detailed Pin Description

Symbol	PDIP	PLCC	PQFP	I/O	Name and Function
ALE	30	33	27	I/O	**Address Latch Enable:** Output pulse for latching the low byte of the address during an address to the external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory.
$\overline{EA}$	31	35	29	I	**External Access enable:** $\overline{EA}$ must be externally held low to enable the device to fetch code from external program memory locations 0000H to FFFFH. If $\overline{EA}$ is held high, the device executes from internal program memory unless the program counter contains an address greater than 0FFFH.
P0.0-P0.7	39-32	43-36	37-30	I/O	**Port 0:** Port 0 is an 8-bit open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and 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.
P1.0-P1.7	1-8	2-9	40-44 1-3	I/O	**Port 1:** Port 1 is an 8-bit bidirectional I/O port with internal pullups. Port 1 pins that have 1s written to them are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally pulled low will source current because of the internal pullups. (See DC Characteristics: I_{IL}). The Port 1 output buffers can sink/source four TTL inputs. Port 1 also receives the low-order address byte during ROM verification.
	1	2	40	I	**T2(P1.0):** Timer/Counter 2 external count input.
	2	3	41	I	**T2EX(P1.1):** Timer/Counter 2 trigger input.
P2.0-P2.7	21-28	24-31	18-25	I/O	**Port 2:** Port 2 is an 8-bit bidirectional I/O port with internal pullups. Port 2 pins that have 1s written to them are pulled high by the internal pullups and can be used as inputs. As inputs, Port 2 pins that are externally pulled low will source current because of the internal pullups. (See DC Characteristics: I_{IL}). Port 2 emits the high order address byte during fetches from external program memory and during accesses to external data memory that used 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pullups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ Ri [i = 0, 1]), Port 2 emits the contents of the P2 Special Function Register. Port 2 also receives the high-order bits and some control signals during ROM verification.

Table 1. Detailed Pin Description *(continued)*

Symbol	PDIP	PLCC	PQFP	I/O	Name and Function
P3.0-P3.7	10-17	11, 13-19	5, 7-13	I/O	**Port 3:** Port 3 is an 8-bit bidirectional I/O port with internal pullups. Port 3 pins that have 1s written to them are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally pulled low will source current because of the internal pullups. (See DC Characteristics: I_{IL}).
					Port 3 also serves the special features of the IS80C51/31, as listed below:
	10	11	5	I	**RxD (P3.0):** Serial input port.
	11	13	7	O	**TxD (P3.1):** Serial output port.
	12	14	8	I	**$\overline{\text{INT0}}$ (P3.2):** External interrupt 0.
	13	15	9	I	**$\overline{\text{INT1}}$ (P3.3):** External interrupt 1.
	14	16	10	I	**T0 (P3.4):** Timer 0 external input.
	15	17	11	I	**T1 (P3.5):** Timer 1 external input.
	16	18	12	O	**$\overline{\text{WR}}$ (P3.6):** External data memory write strobe.
	17	19	13	O	**$\overline{\text{RD}}$ (P3.7):** External data memory read strobe.
$\overline{\text{PSEN}}$	29	32	26	O	**Program Store Enable:** The read strobe to external program memory. When the device is executing code from the external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory. $\overline{\text{PSEN}}$ is not activated during fetches from internal program memory.
RST	9	10	4	I	**Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal MOS resistor to GND permits a power-on reset using only an external capacitor connected to Vcc.
XTAL 1	19	21	15	I	**Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL 2	18	20	14	O	**Crystal 2:** Output from the inverting oscillator amplifier.
GND	20	22	16	I	**Ground:** 0V reference.
Vcc	40	44	38	I	**Power Supply:** This is the power supply voltage for operation.

OPERATING DESCRIPTION

The detail description of the IS80C52/32 included in this description are:

•Memory Map and Registers

•Timer/Counters

•Serial Interface

•Interrupt System

•Other Information

MEMORY MAP AND REGISTERS

Memory

The IS80C52/32 has separate address spaces for program and data memory. The program and data memory can be up to 64K bytes long. The lower 8K program memory can reside on-chip. (IS80C52 only) Figure 5 shows a map of the IS80C52/32 program and data memory.

The IS80C52/32 has 256 bytes of on-chip RAM, plus numbers of special function registers. The lower 128 bytes can be accessed either by direct addressing or by indirect addressing. The upper 128 bytes can be accessed by indirect addressing only. Figure 6 shows internal data memory organization and SFR Memory Map.

The lower 128 bytes of RAM can be divided into three segments as listed below and shown in Figure 7.

1. ***Register Banks 0-3:*** locations 00H through 1FH (32 bytes). The device after reset defaults to register bank 0. To use the other register banks, the user must select them in software. Each register bank contains eight 1-byte registers R0-R7. Reset initializes the stack point to location 07H, and is incremented once to start from 08H, which is the first register of the second register bank.
2. ***Bit Addressable Area:*** 16 bytes have been assigned for this segment 20H-2FH. Each one of the 128 bits of this segment can be directly addressed (0-7FH). Each of the 16 bytes in this segment can also be addressed as a byte.
3. ***Scratch Pad Area:*** 30H-7FH are available to the user as data RAM. However, if the data pointer has been initialized to this area, enough bytes should be left aside to prevent SP data destruction.

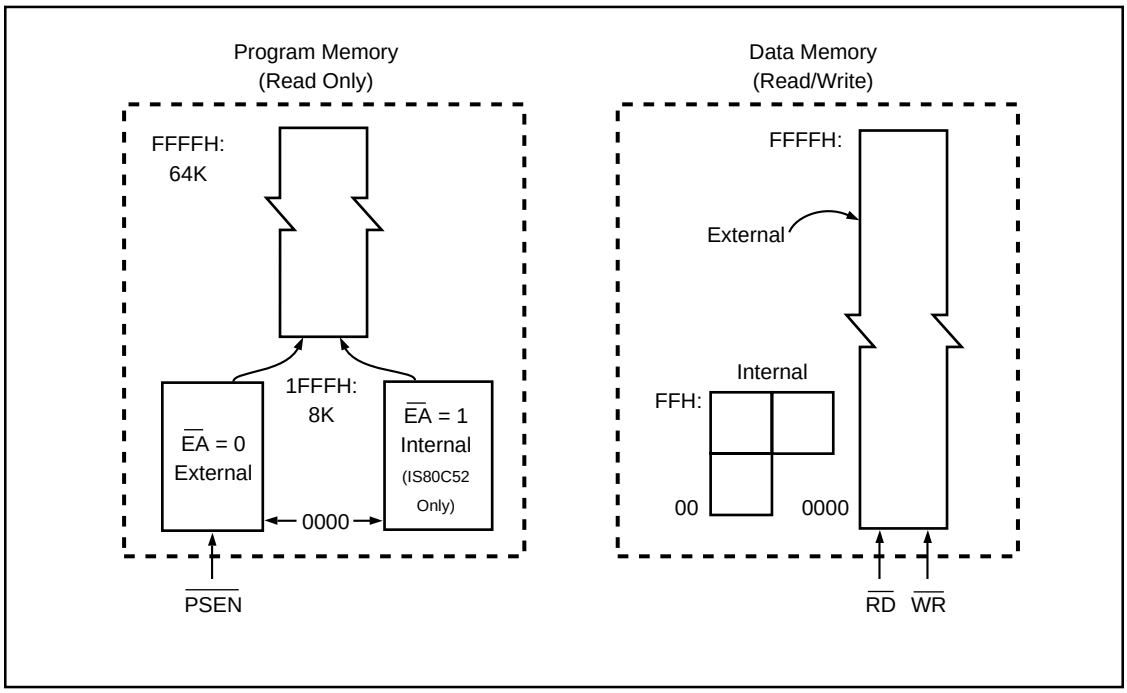

Figure 5. IS80C52/32 Program and Data Memory Structure

SPECIAL FUNCTION REGISTERS

The Special Function Registers (SFR's) are located in upper 128 Bytes direct addressing area. The SFR Memory Map in Figure 6 shows that.

Not all of the addresses are occupied. Unoccupied addresses are not implemented on the chip. Read accesses to these addresses in general return random data, and write accesses have no effect.

User software should not write 1s to these unimplemented locations, since they may be used in future microcontrollers to invoke new features. In that case, the reset or inactive values of the new bits will always be 0, and their active values will be 1.

The functions of the SFRs are outlined in the following sections, and detailed in Table 2.

Accumulator (ACC)
ACC is the Accumulator register. The mnemonics for Accumulator-specific instructions, however, refer to the Accumulator simply as A.

B Register (B)
The B register is used during multiply and divide operations. For other instructions it can be treated as another scratch pad register.

Program Status Word (PSW). The PSW register contains program status information.

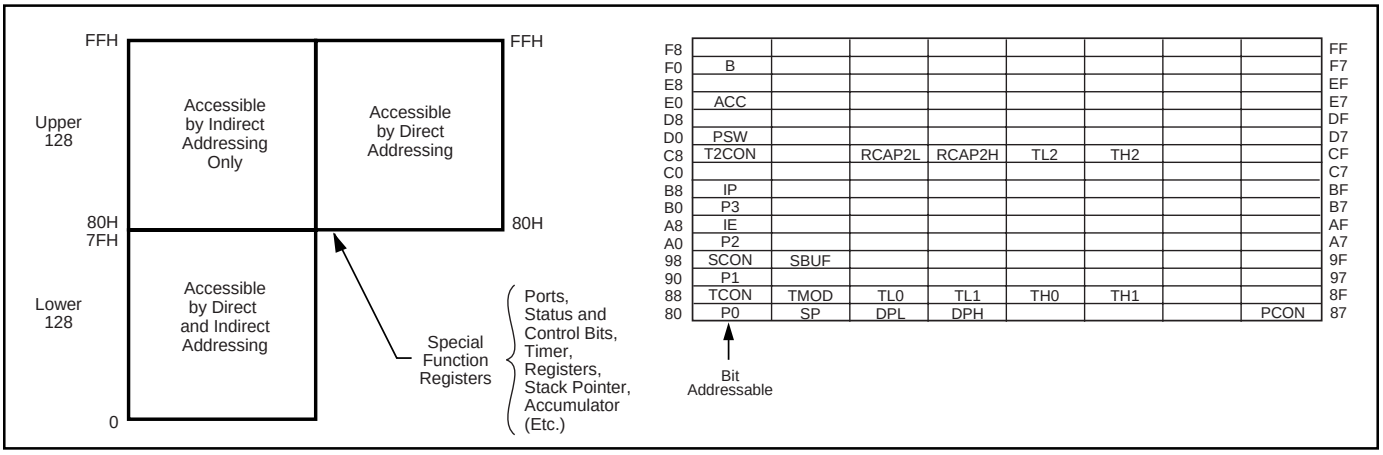

Figure 6. Internal Data Memory and SFR Memory Map

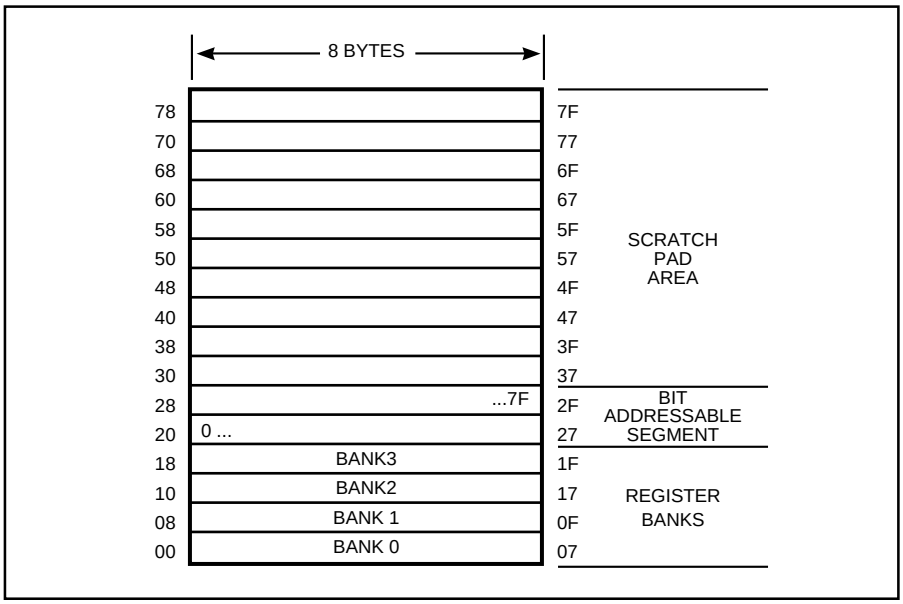

Figure 7. Lower 128 Bytes of Internal RAM

SPECIAL FUNCTION REGISTERS

(continued)

Stack Pointer (SP)
The Stack Pointer Register is eight bits wide. It is incremented before data is stored during PUSH and CALL executions. While the stack may reside anywhere in on-chip RAM, the Stack Pointer is initialized to 07H after a reset. This causes the stack to begin at location 08H.

Data Pointer (DPTR)
The Data Pointer consists of a high byte (DPH) and a low byte (DPL). Its function is to hold a 16-bit address. It may be manipulated as a 16-bit register or as two independent 8-bit registers.

Ports 0 To 3
P0, P1, P2, and P3 are the SFR latches of Ports 0, 1, 2, and 3, respectively.

Serial Data Buffer (SBUF)
The Serial Data Buffer is actually two separate registers, a transmit buffer and a receive buffer register. When data is moved to SBUF, it goes to the transmit buffer, where it is held for serial transmission. (Moving a byte to SBUF initiates the transmission.) When data is moved from SBUF, it comes from the receive buffer.

Timer Registers
Register pairs (TH0, TL0), (TH1, TL1), and (TH2, TL2) are the 16-bit Counter registers for Timer/Counters 0, 1, and 2, respectively.

Capture Registers
The register pair (RCAP2H, RCAP2L) are the Capture registers for the Timer 2 Capture Mode. In this mode, in response to a transition at the IS80C52/32's T2EX pin, TH2 and TL2 are copied into RCAP2H and RCAP2L. Timer 2 also has a 16-bit auto-reload mode, and RCAP2H and RCAP2L hold the reload value for this mode.

Control Registers
Special Function Registers IP, IE, TMOD, TCON, T2CON, SCON, and PCON contain control and status bits for the interrupt system, the Timer/Counters, and the serial port. They are described in later sections of this chapter.

Table 2: Special Function Register

Symbol	Description	Direct Address	Bit Address, Symbol, or Alternative Port Function								Reset Value
ACC(1)	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H
B(1)	B register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H
DPH	Data pointer (DPTR) high	83H									00H
DPL	Data pointer (DPTR) low	82H									00H
IE(1)	Interrupt enable	A8H	**AF** EA	**AE** —	**AD** ET2	**AC** ES	**AB** ET1	**AA** EX1	**A9** ET0	**A8** EX0	0X000000B
IP(1)	Interrupt priority	B8H	**BF** —	**BE** —	**BD** PT2	**BC** PS	**BB** PT1	**BA** PX1	**B9** PT0	**B8** PX0	XX000000B
P0(1)	Port 0	80H	**87** P0.7 AD7	**86** P0.6 AD6	**85** P0.5 AD5	**84** P0.4 AD4	**83** P0.3 AD3	**82** P0.2 AD2	**81** P0.1 AD1	**80** P0.0 AD0	FFH
P1(1)	Port 1	90H	**97** P1.7 —	**96** P1.6 —	**95** P1.5 —	**94** P1.4 —	**93** P1.3 —	**92** P1.2 —	**91** P1.1 T2EX	**90** P1.0 T2	FFH
P2(1)	Port 2	A0H	**A7** P2.7 AD15	**A6** P2.6 AD14	**A5** P2.5 AD13	**A4** P2.4 AD12	**A3** P2.3 AD11	**A2** P2.2 AD10	**A1** P2.1 AD9	**A0** P2.0 AD8	FFH
P3(1)	Port 3	B0H	**B7** P3.7 $\overline{RD}$	**B6** P3.6 $\overline{WR}$	**B5** P3.5 T1	**B4** P3.4 T0	**B3** P3.3 $\overline{INT1}$	**B2** P3.2 $\overline{INT0}$	**B1** P3.1 TXD	**B0** P3.0 RXD	FFH
PCON	Power control	87H	SMOD	—	—	—	GF1	GF0	PD	IDL	0XXX0000B
PSW(1)	Program status word	D0H	**D7** CY	**D6** AC	**D5** F0	**D4** RS1	**D3** RS0	**D2** OV	**D1** —	**D0** P	00H
SBUF	Serial data buffer	99H									XXXXXXXXB
SCON(1)	Serial controller	98H	**9F** SM0	**9E** SM1	**9D** SM2	**9C** REN	**9B** TB8	**9A** RB8	**99** TI	**98** RI	00H
SP	Stack pointer	81H									07H
TCON(1)	Timer control	88H	**8F** TF1	**8E** TR1	**8D** TF0	**8C** TR0	**8B** IE1	**8A** IT1	**89** IE0	**88** IT0	00H
TMOD	Timer mode	89H	GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0	00H
T2CON(1,2)	Timer/Counter 2	C8H	**CF** TF2	**CE** EXF2	**CD** RCLK	**CC** TCLK	**CB** EXEN2	**CA** TR2	**C9** C/$\overline{T2}$	**C8** CP/$\overline{RL2}$	00H
TH0	Timer high 0	8CH									00H
TH1	Timer high 1	8DH									00H
TH2(2)	Timer high 2	CDH									00H
TL0	Timer low 0	8AH									00H
TL1	Timer low 1	8BH									00H
RCAP2H(2)	Capture high	CAH									00H
RCAP2L(2)	Capture low	CBH									00H
TL2(2)	Timer low 2	CCH									00H

Notes:
1. Denotes bit addressable.
2. SFRs are added to the 80C51 SFRs.

The detail description of each bit is as follows:

PSW:

Program Status Word. Bit Addressable.

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	—	P

Register Description:

CY	PSW.7	Carry flag.
AC	PSW.6	Auxiliary carry flag.
F0	PSW.5	Flag 0 available to the user for general purpose.
RS1	PSW.4	Register bank selector bit 1.(1)
RS0	PSW.3	Register bank selector bit 0.(1)
OV	PSW.2	Overflow flag.
—	PSW.1	Usable as a general purpose flag
P	PSW.0	Parity flag. Set/Clear by hardware each instruction cycle to indicate an odd/even number of "1" bits in the accumulator.

Note:

1. The value presented by RS0 and RS1 selects the corresponding register bank.

RS1	RS0	Register Bank	Address
0	0	0	00H-07H
0	1	1	08H-0FH
1	0	2	10H-17H
1	1	3	18H-1FH

PCON:

Power Control Register. Not Bit Addressable.

7	6	5	4	3	2	1	0
SMOD	—	—	—	GF1	GF0	PD	IDL

Register	Description:
SMOD	Double baud rate bit. If Timer 1 is used to generate baud rate and SMOD=1, the baud rate is doubled when the serial port is used in modes 1, 2, or 3.
—	Not implemented, reserve for future use.(1)
—	Not implemented, reserve for future use.(1)
—	Not implemented, reserve for future use.(1)
GF1	General purpose flag bit.
GF0	General purpose flag bit.
PD	Power-down bit. Setting this bit activates power-down operation in the IS80C52/32.
IDL	Idle mode bit. Setting this bit activates idle mode operation in the IS80C52/32. If 1s are written to PD and IDL at the same time, PD takes precedence.

Note:

1. User software should not write 1s to reserved bits. These bits may be used in future products to invoke new features.

IE:

Interrupt Enable Register. Bit Addressable.

7	6	5	4	3	2	1	0
EA	—	ET2	ES	ET1	EX1	ET0	EX0

Register Description:

EA	IE.7	Disable all interrupts. If EA=0, no interrupt will be acknowledged. If EA=1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
—	IE.6	Not implemented, reserve for future use.(5)
ET2	IE.5	Enables or disables timer 2 overflow interrupt.
ES	IE.4	Enable or disable the serial port interrupt.
ET1	IE.3	Enable or disable the timer 1 overflow interrupt.
EX1	IE.2	Enable or disable external interrupt 1.
ET0	IE.1	Enable or disable the timer 0 overflow interrupt.
EX0	IE.0	Enable or disable external interrupt 0.

Note:

To use any of the interrupts in the 80C51 Family, the following three steps must be taken:

1. Set the $\overline{EA}$ (enable all) bit in the IE register to 1.
2. Set the coresponding individual interrupt enable bit in the IE register to 1.
3. Begin the interrupt service routine at the corresponding Vector Address of that interrupt (see below).

Interrupt Source	Vector Address
IE0	0003H
TF0	000BH
IE1	0013H
TF1	001BH
RI & TI	0023H
TF2 and EXF2	002BH

4. In addition, for external interrupts, pins INT0 and INT1 (P3.2 and P3.3) must be set to 1, and depending on whether the interrupt is to be level or transition activated, bits IT0 or IT1 in the TCON register may need to be set to 0 or 1.
   ITX = 0 level activated (X = 0, 1)
   ITX = 1 transition activated
5. User software should not write 1s to reserved bits. These bits may be used in future products to invoke new features.

IP:

Interrupt Priority Register. Bit Addressable.

7	6	5	4	3	2	1	0
—	—	PT2	PS	PT1	PX1	PT0	PX0

Register Description:

—	IP.7	Not implemented, reserve for future use(3)
—	IP.6	Not implemented, reserve for future use(3)
PT2	IP.5	Defines Timer 2 interrupt priority level
PS	IP.4	Defines Serial Port interrupt priority level
PT1	IP.3	Defines Timer 1 interrupt priority level
PX1	IP.2	Defines External Interrupt 1 priority level
PT0	IP.1	Defines Timer 0 interrupt priority level
PX0	IP.0	Defines External Interrupt 0 priority level

Notes:

1. In order to assign higher priority to an interrupt the coresponding bit in the IP register must be set to 1. While an interrupt service is in progress, it cannot be interrupted by a lower or same level interrupt.
2. Priority within level is only to resolve simultaneous requests of the same priority level. From high to low, interrupt sources are listed below:
   IE0
   TF0
   IE1
   TF1
   RI or TI
   TF2 or EXF2
3. User software should not write 1s to reserved bits. These bits may be used in future products to invoke new features.

TCON:

Timer/Counter Control Register. Bit Addressable

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Register Description:

TF1	TCON.7	Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine.
TR1	TCON.6	Timer 1 run control bit. Set/Cleared by software to turn Timer/Counter 1 ON/OFF.
TF0	TCON.5	Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the interrupt service routine.
TR0	TCON.4	Timer 0 run control bit. Set/Cleared by software to turn Timer/Counter 0 ON/OFF.
IE1	TCON.3	External Interrupt 1 edge flag. Set by hardware when the External Interrupt edge is detected. Cleared by hardware when interrupt is processed.
IT1	TCON.2	Interrupt 1 type control bit. Set/Cleared by software specify falling edge/low level triggered External Interrupt.
IE0	TCON.1	External Interrupt 0 edge flag. Set by hardware when the External Interrupt edge is detected. Cleared by hardware when interrupt is processed.
IT0	TCON.0	Interrupt 0 type control bit. Set/Cleared by software specify falling edge/low level triggered External Interrupt.

TMOD:

Timer/Counter Mode Control Register.
Not Bit Addressable.

Timer 1				Timer 0			
GATE	**C/$\overline{T}$**	**M1**	**M0**	**GATE**	**C/$\overline{T}$**	**M1**	**M0**

GATE	When TRx (in TCON) is set and GATE=1, TIMER/COUNTERx will run only while INTx pin is high (hardware control). When GATE=0, TIMER/COUNTERx will run only while TRx=1 (software control).
C/$\overline{T}$	Timer or Counter selector. Cleared for Timer operation (input from internal system clock). Set for Counter operation (input from Tx input pin).
M1	Mode selector bit.(1)
M0	Mode selector bit.(1)

Note 1:

M1	M0	Operating mode
0	0	Mode 0. (13-bit Timer)
0	1	Mode 1. (16-bit Timer/Counter)
1	0	Mode 2. (8-bit auto-load Timer/Counter)
1	1	Mode 3. (Splits Timer 0 into TL0 and TH0. TL0 is an 8-bit Timer/Counter controller by the standard Timer 0 control bits. TH0 is an 8-bit Timer and is controlled by Timer 1 control bits.)
1	1	Mode 3. (Timer/Counter 1 stopped).

SCON:

Serial Port Control Register. Bit Addressable.

7	6	5	4	3	2	1	0
SM0	**SM1**	**SM2**	**REN**	**TB8**	**RB8**	**TI**	**RI**

Register Description:

SM0	SCON.7	Serial port mode specifier.(1)
SM1	SCON.6	Serial port mode specifier.(1)
SM2	SCON.5	Enable the multiprocessor communication feature in mode 2 and 3. In mode 2 or 3, if SM2 is set to 1 then RI will not be activated if the received 9th data bit (RB8) is 0. In mode 1, if SM2=1 then RI will not be activated if valid stop bit was not received. In mode 0, SM2 should be 0.
REN	SCON.4	Set/Cleared by software to Enable/Disable reception.
TB8	SCON.3	The 9th bit that will be transmitted in mode 2 and 3. Set/Cleared by software.
RB8	SCON.2	In modes 2 and 3, RB8 is the 9th data bit that was received. In mode 1, if SM2=0, RB8 is the stop bit that was received. In mode 0, RB8 is not used.
TI	SCON.1	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the beginning of the stop bit in the other modes. Must be cleared by software.
RI	SCON.0	Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or halfway through the stop bit time in the other modes (except see SM2). Must be cleared by software.

Note:

SM0	SM1	MODE	Description	Baud rate
0	0	0	Shift register	Fosc/12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	Fosc/64 or Fosc/32
1	1	3	9-bit UART	Variable

T2CON:

Timer/Counter 2 Control Register. Bit Addressable.

7	6	5	4	3	2	1	0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/$\overline{T2}$	CP/$\overline{RL2}$

Register Description:

TF2	T2CON.7	Timer 2 overflow flag set by hardware and cleared by software. TF2 cannot be set when either RCLK = 1 or TCLK = 1.
EXF2	T2CON.6	Timer 2 external flag set when either a capture or reload is caused by a negative transition on T2EX, and EXEN2 = 1. When Timer 2 interrupt is enabled, EXF2 = 1 causes the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software.
RCLK	T2CON.5	Receive clock flag. When set, causes the Serial Port to use Timer 2 overflow pulses for its receive clock in modes 1 and 3. RCLK = 0 causes Timer 1 overflow to be used for the receive clock.
TLCK	T2CON.4	Transmit clock flag. When set, causes the Serial Port to use Timer 2 overflow pulses for its transmit clock in modes 1 and 3. TCLK = 0 causes Timer 1 overflows to be used for the transmit clock.
EXEN2	T2CON.3	Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of negative transition on T2EX if Timer 2 is not being used to clock the Serial Port, EXEN2 = 0 causes Timer 2 to ignore events at T2EX.
TR2	T2CON.2	Software START/STOP control for Timer 2. A logic 1 starts the Timer.
C/T2	T2CON.1	Timer or Counter select. 0 = Internal Timer. 1 = External Event Counter (triggered by falling edge).
CP/RL2	T2CON.0	Capture/Reload flag. When set, captures occur on negative transitions at T2EX if EXEN2 = 1. When cleared, auto-reloads occur either with Timer 2 overflows or negative transitions at T2EX when EXEN2 = 1. When either RCLK = 1 or TCLK = 1, this bit is ignored and the Timer is forced to auto-reload on Timer 2 overflow.

Notes:

1. Timer 2 Operating Modes

RCLK + TCLK	CP/$\overline{RL2}$	TR2	MODE
0	0	1	16-Bit Auto-Reload
0	1	1	16-Bit Capture
1	X	1	Baud Rate Generator
X	X	0	(Off)

2. User software should not write 1s to reserved bits. These bits may be used in future products to invoke new features.

TIMER/COUNTERS

The IS80C52/32 has three 16-bit Timer/Counter registers: Timer 0, Timer 1, and in addition Timer 2. All three can be configured to operate either as Timers or event Counters.

As a Timer, the register is incremented every machine cycle. Thus, the register counts machine cycles. Since a machine cycle consists of 12 oscillator periods, the count rate is 1/12 of the oscillator frequency.

As a Counter, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0, T1, and T2. The external input is sampled during S5P2 of every machine cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during S3P1 of the cycle following the one in which the transition was detected. Since two machine cycles (24 oscillator periods) are required to recognize a 1-to-0 transition, the maximum count rate is 1/24 of the oscillator frequency. There are no restrictions on the duty cycle of the external input signal, but it should be held for at least one full machine cycle to ensure that a given level is sampled at least once before it changes.

In addition to the Timer or Counter functions, Timer 0 and Timer 1 have four operating modes: (13-bit timer, 16-bit timer, 8-bit auto-reload, split timer). Timer 2 in the IS80C52/32 has three modes of operation: Capture, Auto-Reoload, and Baud Rate Generator.

Timer 0 and Timer 1

Timer/Counters 0 and 1 are present in both the IS80C51/31 and IS80C52/32. The Timer or Counter function is selected by control bits C/T in the Special Function Regiser TMOD. These two Timer/Counters have four operating modes, which are selected by bit pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timer/Counters, but Mode 3 is different. The four modes are described in the following sections.

Mode 0:
Both Timers in Mode 0 are 8-bit Counters with a divide-by-32 prescaler. Figure 8 shows the Mode 0 operation as it applies to Timer 1.

In this mode, the Timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the Timer interrupt flag TF1. The counted input is enabled to the Timer when TR1 = 1 and either GATE = 0 or $\overline{INT1}$ = 1. Setting GATE = 1 allows the Timer to be controlled by external input $\overline{INT1}$, to facilitate pulse width measurements. TR1 is a control bit in the Special Function Register TCON. Gate is in TMOD.

The 13-bit register consists of all eight bits of TH1 and the lower five bits of TL1. The upper three bits of TL1 are indeterminate and should be ignored. Setting the run flag (TR1) does not clear the registers.

Mode 0 operation is the same for Timer 0 as for Timer 1, except that TR0, TF0 and $\overline{INT0}$ replace the corresponding Timer 1 signals in Figure 8. There are two different GATE bits, one for Timer 1 (TMOD.7) and one for Timer 0 (TMOD.3).

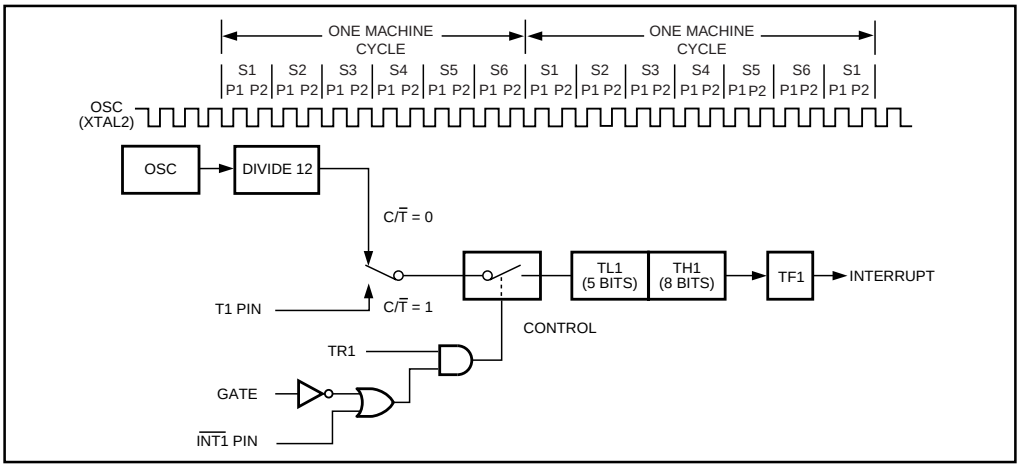

Figure 8. Timer/Counter 1 Mode 0: 13-Bit Counter

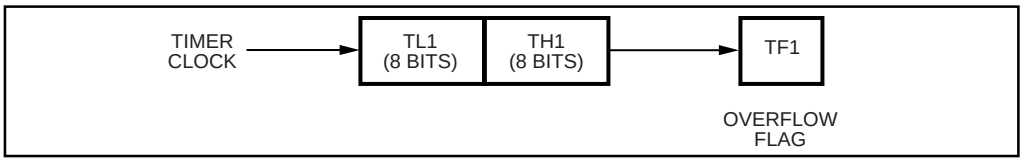

Figure 9. Timer/Counter 1 Mode 1: 16-Bit Counter

Mode 1:
Mode 1 is the same as Mode 0, except that the Timer register is run with all 16 bits. The clock is applied to the combined high and low timer registers (TL1/TH1). As clock pulses are received, the timer counts up: 0000H, 0001H, 0002H, etc. An overflow occurs on the FFFFH-to-0000H overflow flag. The timer continues to count. The overflow flag is the TF1 bit in TCON that is read or written by software (see Figure 9).

Mode 2:
Mode 2 configures the Timer register as an 8-bit Counter (TL1) with automatic reload, as shown in Figure 10. Overflow from TL1 not only sets TF1, but also reloads TL1 with the contents of TH1, which is preset by software. The reload leaves the TH1 unchanged. Mode 2 operation is the same for Timer/Counter 0.

Mode 3:
Timer 1 in Mode 3 simply holds its count. The effect is the same as setting TR1 = 0. Timer 0 in Mode 3 establishes TL0 and TH0 as two separate counters. The logic for Mode 3 on Timer 0 is shown in Figure 10. TL0 uses the Timer 0 control bits: C/T, GATE, TR0, INT0, and TF0. TH0 is locked into a timer function (counting machine cycles) and over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the Timer 1 interrupt.

Mode 3 is for applications requiring an extra 8-bit timer or counter. With Timer 0 in Mode 3, the IS80C51/31 can appear to have three Timer/Counters, and an IS80C52/32 can appear to have four. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it out of and into its own Mode 3. In this case, Timer 1 can still be used by the serial port as a baud rate generator or in any application not requiring an interrupt.

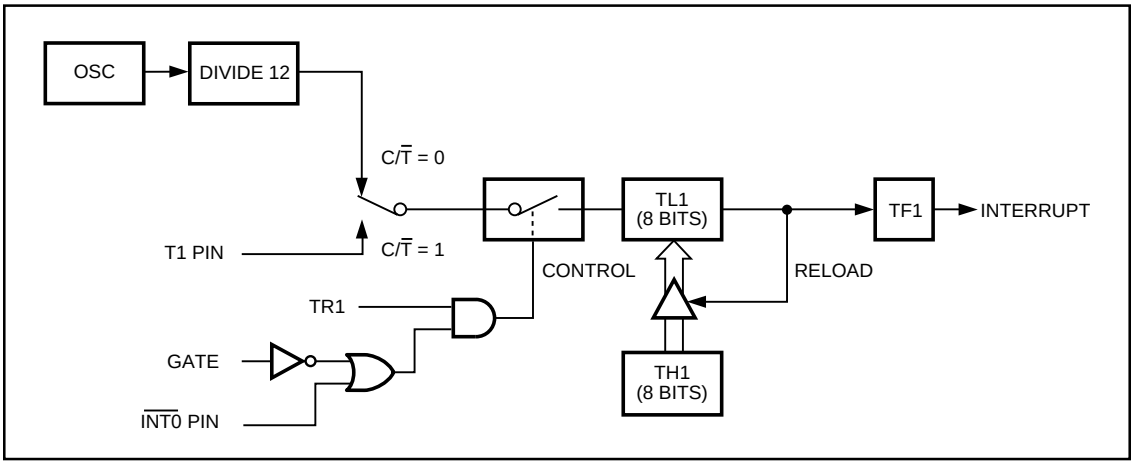

Figure 10. Timer/Counter 1 Mode 2: 8-Bit Auto-Reload

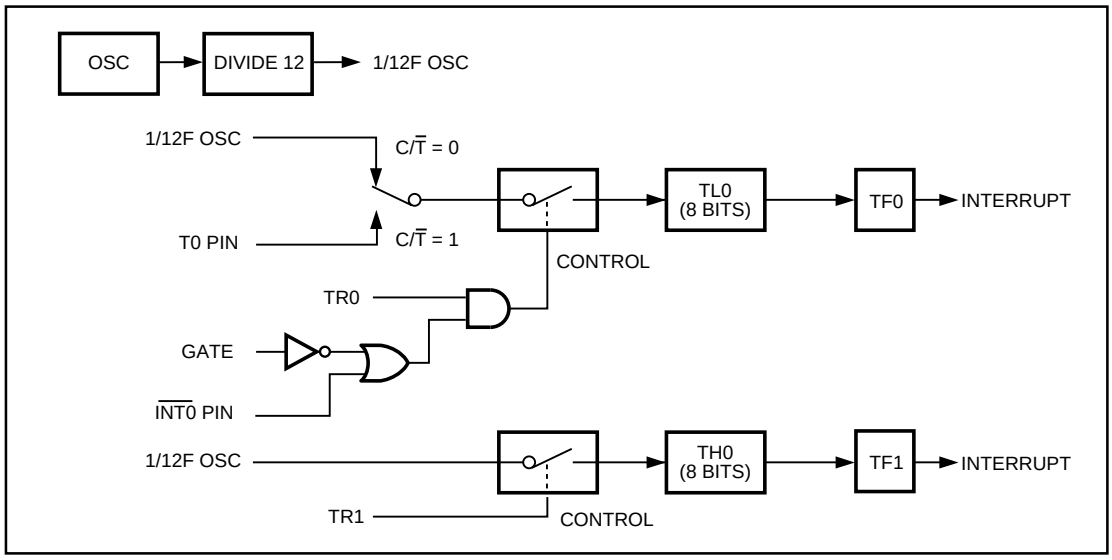

Figure 11. Timer/Counter 0 Mode 3: Two 8-Bit Counters

Timer 2

Timer 2 is a 16-bit Timer/Counter present only in the IS80C52/32. This is a powerful addition to the other two just discussed. Five extra special function registers are added to accommodate Timer 2 which are: the timer registers, TL2 and TH2, the timer control register, T2CON, and the capture registers, RCAP2L and RCAP2H. Like Timers 0 and 1, it can operate either as a timer or as an event counter, depending on the value of bit C/T2 in the Special Function Register T2CON. Timer 2 has three operating modes: capture, auto-reload, and baud rate generator, which are selected by RCLK, TCLK, CP/RL2 and TR2.

In the Capture Mode, the EXEN2 bit in T2CON selects two options. If EXEN2 = 0, then Timer 2 is a 16-bit timer or counter whose overflow sets bit TF2, the Timer 2 overflow bit, which can be used to generate an interrupt. If EXEN2 = 1, then Timer 2 performs the same way, but a 1-to-0 transition at external input T2EX also causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into the RCAP2L and RCAP2H registers, respectively. In addition, the transition at T2EX sets the EXF2 bit in T2CON, and EXF2, like TF2, can generate an interrupt.

The Capture Mode is illustrated in Figure 12.

In the auto-reload mode, the EXEN2 bit in T2CON also selects two options. If EXEN2 = 0, then when Timer 2 rolls over it sets TF2 and also reloads the Timer 2 registers with the 16-bit value in the RCAP2L and RCAP2H registers, which are preset by software. If EXEN2 = 1, then Timer 2 performs the same way, but a 1-to-0 transition at external input T2EX also triggers the 16-bit reload and sets EXF2.

The auto-reload mode is illustrated in Figure 13.

The baud rate generator mode is selected by RCLK = 1 and/or TCLK = 1. This mode is described in conjunction with the serial port (Figure 14).

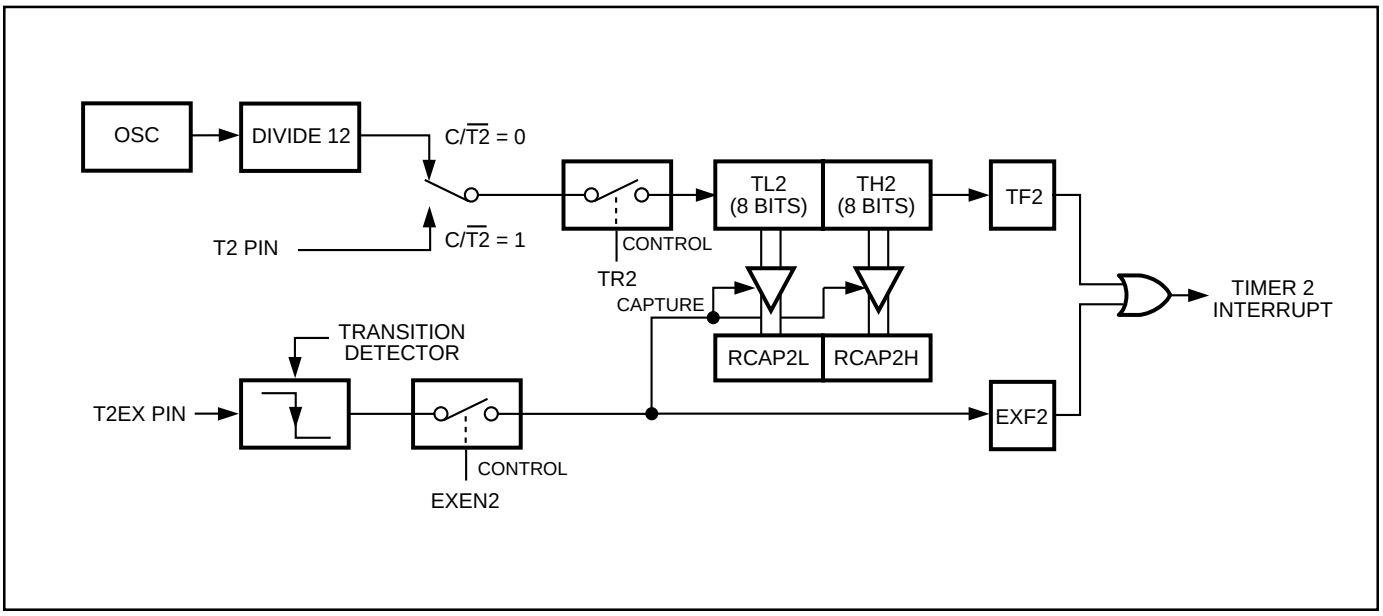

Figure 12. Timer 2 In Capture Mode

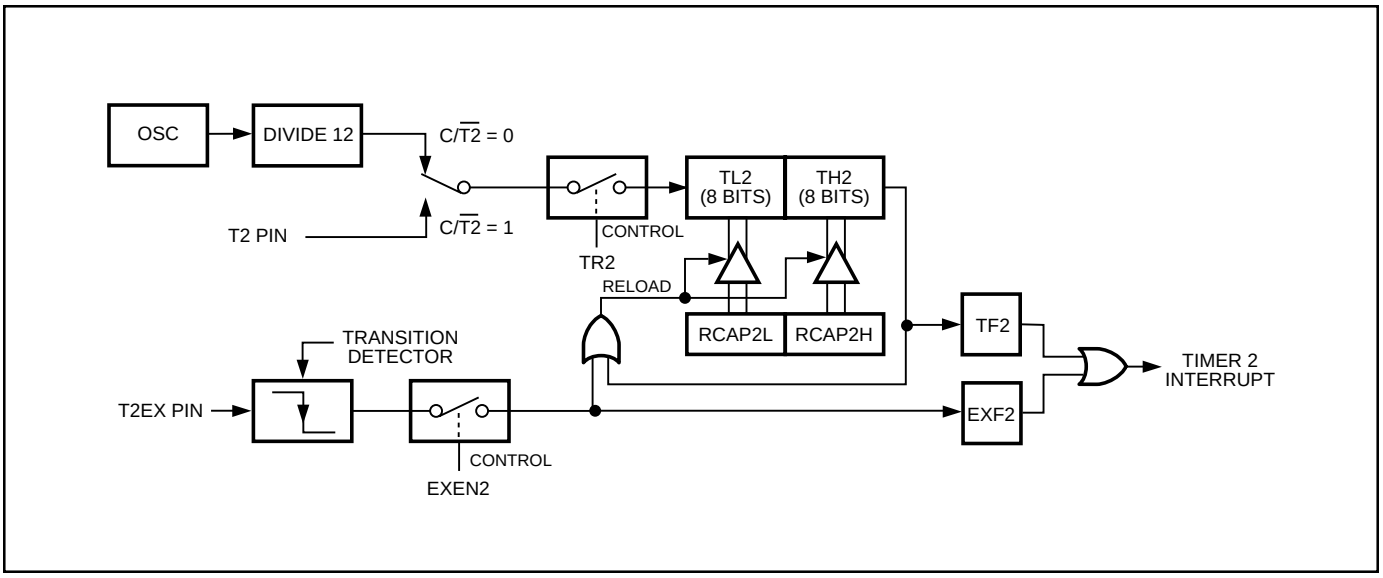

Figure 13. Timer 2 in Auto-Reload Mode

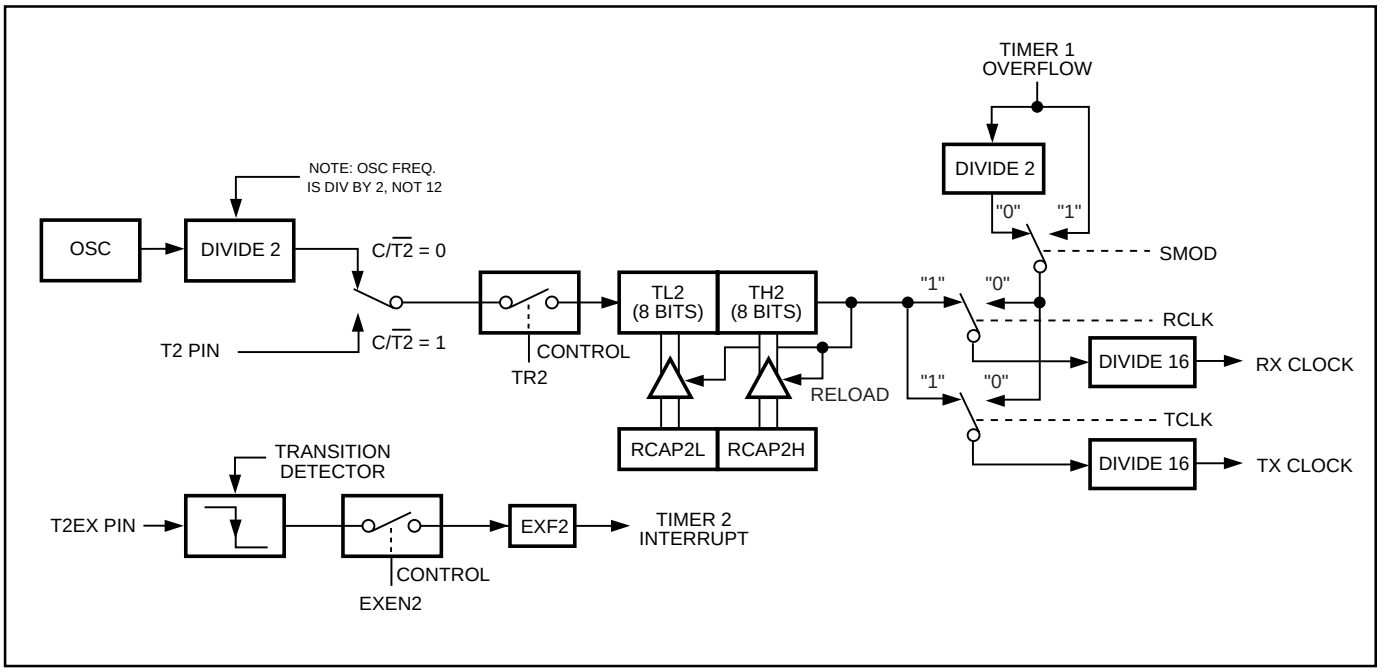

Figure 14. Timer 2 in Baud Rate Generator Mode

Note:
1. T2EX can be used as an additional external interrupt.

Timer Set-Up

Tables 3 through 6 give TMOD values that can be used to set up Timers in different modes.

It assumes that only one timer is used at a time. If Timers 0 and 1 must run simultaneously in any mode, the value in TMOD for Timer 0 must be ORed with the value shown for Timer 1 (Tables 5 and 6).

For example, if Timer 0 must run in Mode 1 GATE (external control), and Timer 1 must run in Mode 2 COUNTER, then the value that must be loaded into TMOD is 69H (09H from Table 3 ORed with 60H from Table 6).

Moreover, it is assumed that the user is not ready at this point to turn the timers on and will do so at another point in the program by setting bit TRx (in TCON) to 1.

Table 3. Timer/Counter 0 Used as a Timer

Mode	Timer 0 Function	TMOD Internal Control(1)	TMOD External Control(2)
0	13-Bit Timer	00H	08H
1	16-Bit Timer	01H	09H
2	8-Bit Auto-Reload	02H	0AH
3	Two 8-Bit Timers	03H	0BH

Table 4. Timer/Counter 0 Used as a Counter

Mode	Timer 0 Function	TMOD Internal Control(1)	TMOD External Control(2)
0	13-Bit Timer	04H	0CH
1	16-Bit Timer	05H	0DH
2	8-Bit Auto-Reload	06H	0EH
3	One 8-Bit Counter	07H	0FH

Notes:
1. The Timer is turned ON/OFF by setting/clearing bit TR0 in the software.

Table 5. Timer/Counter 1 Used as a Timer

Mode	Timer 1 Function	TMOD Internal Control(1)	TMOD External Control(2)
0	13-Bit Timer	00H	80H
1	16-Bit Timer	10H	90H
2	8-Bit Auto-Reload	20H	A0H
3	Does Not Run	30H	B0H

Table 6. Timer/Counter 1 Used as a Counter

Mode	Timer 1 Function	TMOD Internal Control(1)	TMOD External Control(2)
0	13-Bit Timer	40H	C0H
1	16-Bit Timer	50H	D0H
2	8-Bit Auto-Reload	60H	E0H
3	Not Available	—	—

Notes:
1. The Timer is turned ON/OFF by setting/clearing bit TR1 in the software.
2. The Timer is turned ON/OFF by the 1 to 0 transition on INT1 (P3.3) when TR1 = 1 (hardware control).

Timer/Counter 2 Set-Up

Except for the baud rate generator mode, the values given for T2C0N do not include the setting of the TR2 bit. Therefore, bit TR2 must be set separately to turn the Timer on.

Table 7. Timer/Counter 2 Used as a Timer

Mode	T2CON Internal Control(1)	T2CON External Control(2)
16-Bit Auto-Reload 08H		00H
16-Bit Capture	01H	09H
Baud Rate Generator Receive and Transmit Same Baud Rate	34H	36H
Receive Only	24H	26H
Transmit Only	14H	16H

Table 8. Timer/Counter 2 Used as a Counter

Mode	TMOD Internal Control(1)	TMOD External Control(2)
16-Bit Auto-Reload 0AH		02H
16-Bit Capture	03H	0BH

Notes:
1. Capture/Reload occurs only on Timer/Counter overflow.
2. Capture/Reload occurs on Timer/Counter overflow and a 1 to 0 transition on T2EX (P1.1) pin except when Timer 2 is used in the baud rate generating mode.

SERIAL INTERFACE

The Serial port is full duplex, which means it can transmit and receive simultaneously. It is also receive-buffered, which means it can begin receiving a second byte before a previously received byte has been read from the receive register. (However, if the first byte still has not been read when reception of the second byte is complete, one of the bytes will be lost.) The serial port receive and transmit registers are both accessed at Special Function Register SBUF. Writing to SBUF loads the transmit register, and reading SBUF accesses a physically separate receive register.

The serial port can operate in the following four modes:

Mode 0:
Serial data enters and exits through RXD. TXD outputs the shift clock. Eight data bits are transmitted/received, with the LSB first. The baud rate is fixed at 1/12 the oscillator frequency (see Figure 15).

Mode 1:
Ten bits are transmitted (through TXD) or received (through RXD): a start bit (0), eight data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in Special Function Register SCON. The baud rate is variable (see Figure 16).

Mode 2:
Eleven bits are transmitted (through TXD) or received (through RXD): a start bit (0), eight data bits (LSB first), a programmable ninth data bit, and a stop bit (1). On transmit, the ninth data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) can be moved into TB8. On receive, the ninth data bit goes into RB8 in Special Function Register SCON, while the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency (see Figure 17).

Mode 3:
Eleven bits are transmitted (through TXD) or received (through RXD): a start bit (0), eight data bits (LSB first), a programmable ninth data bit, and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except the baud rate, which is variable in Mode 3 (see Figure 18).

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

MULTIPROCESSOR COMMUNICATIONS

Modes 2 and 3 have a special provision for multiprocessor communications. In these modes, nine data bits are received, followed by a stop bit. The ninth bit goes into RB8; then comes a stop bit. The port can be programmed such that when the stop bit is received, the serial port interrupt is activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON.

The following example shows how to use the serial interrupt for multiprocessor communications. When the master processor must transmit a block of data to one of several slaves, it first sends out an address byte that identifies the target slave. An address byte differs from a data byte in that the ninth bit is 1 in an address byte and 0 in a data byte. With SM2 = 1, no slave is interrupted by a data byte. An address byte, however, interrupts all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave clears its SM2 bit and prepares to receive the data bytes that follows. The slaves that are not addressed set their SM2 bits and ignore the data bytes.

SM2 has no effect in Mode 0 but can be used to check the validity of the stop bit in Mode 1. In a Mode 1 reception, if SM2 = 1, the receive interrupt is not activated unless a valid stop bit is received.

Baud Rates

The baud rate in Mode 0 is fixed as shown in the following equation.

$$\text{Mode 0 Baud Rate} = \frac{\text{Oscillator Frequency}}{12}$$

The baud rate in Mode 2 depends on the value of the SMOD bit in Special Function Register PCON. If SMOD = 0 (the value on reset), the baud rate is 1/64 of the oscillator frequency. If SMOD = 1, the baud rate is 1/32 of the oscillator frequency, as shown in the following equation.

$$\text{Mode 2 Baud Rate} = \frac{2^{SMOD}}{64} \times \text{(Oscillator Frequency)}$$

In the IS80C51/31, the Timer 1 overflow rate determines the baud rates in Modes 1 and 3. In the IS80C52/32, these baud rates can be determined by Timer 1, Timer 2, or both (one for transmit and the other for receive).

Using the Timer 1 to Generate Baud Rates

When Timer 1 is the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD according to the following equation.

$$\text{Mode 1, 3 Baud Rate} = \frac{2^{SMOD}}{32} \times \text{(Timer 1 Overflow Rate)}$$

The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either timer or counter operation in any of its 3 running modes. In the most typical applications, it is configured for timer operation in auto-reload mode (high nibble of TMOD = 0010B). In this case, the baud rate is given by the following formula.

$$\text{Mode 1,3 Baud Rate} = \frac{2^{SMOD}}{32} \times \frac{\text{Oscillator Frequency}}{12\text{x [256-(TH1)]}}$$

Programmers can achieve very low baud rates with Timer 1 by leaving the Timer 1 interrupt enabled, configuring the Timer to run as a 16-bit timer (high nibble of TMOD = 0001B), and using the Timer 1 interrupt to do a 16-bit software reload.

Table 9 lists commonly used baud rates and how they can be obtained from Timer 1.

Using Timer 2 to Generate Baud Rates

In the IS80C52/32, setting TCLK and/or RCLK in T2CON selects Timer 2 as the baud rate generator. Under these conditions, the baud rates for transmit and receive can be simultaneously different. Setting RCLK and/or TCLK puts Timer 2 into its baud rate generator mode, as shown in Figure 14.

The baud rate generator mode is similar to the auto-reload mode, in that a rollover in TH2 reloads the Timer 2 registers with the 16-bit value in the RCAP2H and RCAP2L registers, which are preset by software.

In this case, the baud rates in Mode 1 and 3 are determined by the Timer 2 overflow rate according to the following equation.

$$\text{Modes 1, 3 Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

Timer 2 can be configured for either timer or counter operation. In the most typical applications, it is configured for timer operation (C/T2 = 0). Normally, a timer increments every machine cycle (thus at 1/12 the oscillator frequency), but timer operation is a different for Timer 2 when it is used as a baud rate generator. As a baud rate generator, Timer 2 increments every state time (thus at 1/2 the oscillator frequency). In this case, the baud rate is given by the following formula.

$$\text{Modes 1, 3 Baud Rate} = \frac{\text{Oscillator Frequency}}{32\text{x [65536} - \text{(RCAP2H, RCAP2L)]}}$$

Where (RCAP2H, RCAP2L) is the content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

Figure 13 shows Timer 2 as a baud rate generator. This figure is valid only if RCLK + TCLK = 1 in T2CON. A rollover in TH2 does not set TF2 and does no generate an interrupt. Therefore, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. If EXEN2 is set, a 1-to-0 transition in T2EX sets EXF2 but does not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Thus, when Timer 2 is used as a baud rate generator, T2EX can be used as an extra external interrupt.

When Timer 2 is running (TR2 = 1) as a timer in the baud rate generator mode, programmers should not read from or write to TH2 or TL2. Under these conditions, Timer 2 is incremented every state time, and the results of a read or write may not be accurate. The RCAP registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. Turn Timer 2 off (clear TR2) before accessing the Timer 2 or RCAP registers, in this case.

Table 9. Commonly Used Baud Rates Generated by Timer 1

Baud Rate	fosc	SMOD	C/T	Timer 1 Mode	Reload Value
Mode 0 Max: 1 MHz	12 MHz	X	X	X	X
Mode 2 Max: 375K	12 MHz	1	X	X	X
Modes 1, 3: 62.5K	12 MHz	1	0	2	FFH
19.2K	11.059 MHz	1	0	2	FDH
9.6K	11.059 MHz	0	0	2	FDH
4.8K	11.059 MHz	0	0	2	FAH
2.4K	11.059 MHz	0	0	2	F4H
1.2K	11.059 MHz	0	0	2	E8H
137.5	11.986 MHz	0	0	2	1DH
110	6 MHz	0	0	2	72H
110	12 MHz	0	0	1	FEEBH

More About Mode 0

Serial data enters and exits through RXD. TXD outputs the shift clock. Eight data bits are transmitted/received, with the LSB first. The baud rate is fixed at 1/12 the oscillator frequency.

Figure 15 shows a simplified functional diagram of the serial port in Mode 0 and associated timing.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal at S6P2 also loads a 1 into the ninth position of the transmit shift register and tells the TX Control block to begin a transmission. The internal timing is such that one full machine cycle will elapse between "write to SBUF" and activation of SEND.

SEND transfer the output of the shift register to the alternate output function line of P3.0, and also transfers SHIFT CLOCK to the alternate output function line of P3.1. SHIFT CLOCK is low during S3, S4, and S5 of every machine cycle, and high during S6, S1, and S2. At S6P2 of every machine cycle in which SEND is active, the contents of the transmit shift register are shifted one position to the right.

As data bits shift out to the right, 0s come in from the left. When the MSB of the data byte is at the output position of the shift register, the 1 that was initially loaded into the ninth position is just to the left of the MSB, and all positions to the left of that contain 0s. This condition flags the TX Control block to do one last shift, then deactivate SEND and set TI. Both of these actions occur at S1P1 of the tenth machine cycle after "write to SBUF."

Reception is initiated by the condition REN = 1 and RI = 0. At S6P2 of the next machine cycle, the RX Control unit writes the bits 11111110 to the receive shift register and activates RECEIVE in the next clock phase.

RECEIVE enables SHIFT CLOCK to the alternate output function line of P3.1. SHIFT CLOCK makes transitions at S3P1 and S6P1 of every machine cycle. At S6P2 of every machine cycle in which RECEIVE is active, the contents of the receive shift register are shifted on position to the left. The value that comes in from the right is the value that was sampled at the P3.0 pin at S5P2 of the same machine cycle.

As data bits come in from the right, 1s shift out to the left. When the 0 that was initially loaded into the right-most position arrives at the left-most position in the shift register, it flags the RX Control block to do one last shift and load SBUF. At S1P1 of the 10th machine cycle after the write to SCON that cleared RI, RECEIVE is cleared and RI is set.

More About Mode 1

Ten bits are transmitted (through TXD), or received (through RXD): a start bit (0), eight data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in SCON. In the IS80C52/32 the baud rate is determined either by the Timer 1 overflow rate, the Timer 2 overflow rate, or both. In this case, one Timer is for transmit, and the other is for receive.

Figure 16 shows a simplified functional diagram of the serial port in Mode 1 and associated timings for transmit and receive.

Transmission is initiated by any instruction that uses SBUF as a destination register.

The "write to =SBUF" signal also loads a 1 into the ninth bit position of the transmit shift register and flags the TX control unit that a transmission is requested. Transmission actually commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.

The transmission begins when SEND is activated, which puts the start bit at TXD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TXD. The first shift pulse occurs one bit time after that.

As data bits shift out to the right, 0s are clocked in from the left. When the MSB of the data byte is at the output position of the shift register, the 1 that was initially loaded into the ninth position is just to the left of the MSB, and all positions to the left of that contain 0s. This condition flags the TX Control unit to do one last shift, then deactivate SEND and set TI. This occurs at the tenth divide-by-16 rollover after "write to SBUF".

Reception is initiated by a 1-to-0 transition detected at RXD. For this purpose, RXD is sampled at a rate of 16 times the established baud rate. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written into the input shift register. Resetting the divide-by-16 counter aligns its rollovers with the boundaries of the incoming bit times.

The 16 states of the counter divide each bit time into 16th. At the seventh, eighth, and ninth counter states of each bit time, the bit detector samples the value of RXD. The value accepted is the value that was seen in at least two of the three samples. This is done to reject noise. In order to reject false bits, if the value accepted during the first bit time is not 0, the receive circuits are reset and the unit continues looking for another 1-to-0 transition. If the start bit is valid, it is shifted into the input shift register, and reception of the rest of the frame proceeds.

As data bits come in from the right, 1s shift to the left. When the start bit arrives at the leftmost position in the shift register, (which is a 9-bit register in Mode 1), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI. The signal to load SBUF and RB8 and to set RI is generated if, and only if, the following conditions are met at the time the final shift pulse is generated.

1. RI = 0 and
2. Either SM2 = 0, or the received stop bit =1

If either of these two conditions is not met, the received frame is irretrievably lost. If both conditions are met, the stop bit goes into RB8, the eight data bits go into SBUF, and RI is activated. At this time, whether or not the above conditions are met, the unit continues looking for a 1-to-0 transition in RXD.

More About Modes 2 and 3

Eleven bits are transmitted (through TXD), or received (through RXD): a start bit (0), 8 data bits (LSB first), a programmable ninth data bit, and a stop bit (1). On transmit, the ninth data bit (TB8) can be assigned the value of 0 or 1. On receive, the ninth data bit goes into RB8 in SCON. The baud rate is programmable to either 1/32 or 1/64 of the oscillator frequency in Mode 2. Mode 3 may have a variable baud rate generated from either Timer 1 or 2, depending on the state of TCLK and RCLK.

Figures 17 and 18 show a functional diagram of the serial port in Modes 2 and 3. The receive portion is exactly the same as in Mode 1. The transmit portion differs from Mode 1 only in the ninth bit of the transmit shift register.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal also loads TB8 into the ninth bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.

The transmission begins when SEND is activated, which puts the start bit at TXD. One bit timer later, DATA is activated, which enables the output bit of the transmit shift register to TXD. The first shift pulse occurs one bit time after that. The first shift clocks a 1 (the stop bit) into the ninth bit position of the shift register. Thereafter, only 0s are clocked in. Thus, as data bits shift out to the right, 0s are clocked in from the left. When TB8 is at the output position of the shift register, then the stop bit is just to the left of TB8, and all positions to the left of that contain 0s. This condition flags the TX Control unit to do one last shift, then deactivate SEND and set TI. This occurs at the 11th divide-by-16 rollover after "write to SBUF".

Reception is initiated by a 1-to-0 transition detected at RXD. For this purpose, RXD is sampled at a rate of 16 times the established baud rate. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written to the input shift register.

At the seventh, eighth, and ninth counter states of each bit time, the bit detector samples the value of RXD. The value accepted is the value that was seen in at least two of the three samples. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit continues looking for another 1-to-0 transition. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame proceeds.

As data bits come in from the right, Is shift out to the left. When the start bit arrives at the leftmost position in the shift register (which in Modes 2 and 3 is a 9-bit register), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI. The signal to load SBUF and RB8 and to set RI is generated if, and only if, the following conditions are met at the time the final shift pulse is generated:

1. RI = 0, and
2. Either SM2 = 0 or the received 9th data bit = 1

If either of these conditions is not met, the received frame is irretrievably lost, and RI is not set. If both conditions are met, the received ninth data bit goes into RB8, and the first eight data bits go into SBUF. One bit time later, whether the above conditions were met or not, the unit continues looking for a 1-to-0 transition at the RXD input.

Note that the value of the received stop bit is irrelevant to SBUF, RB8, or RI.

Table 10. Serial Port Setup

Mode	SCON	SM2Variation
0	10H	Single Processor Environment (SM2 = 0)
1	50H	
2	90H	
3	D0H	
0	NA	Multiprocessor Environment (SM2 = 1)
1	70H	
2	B0H	
3	F0H	

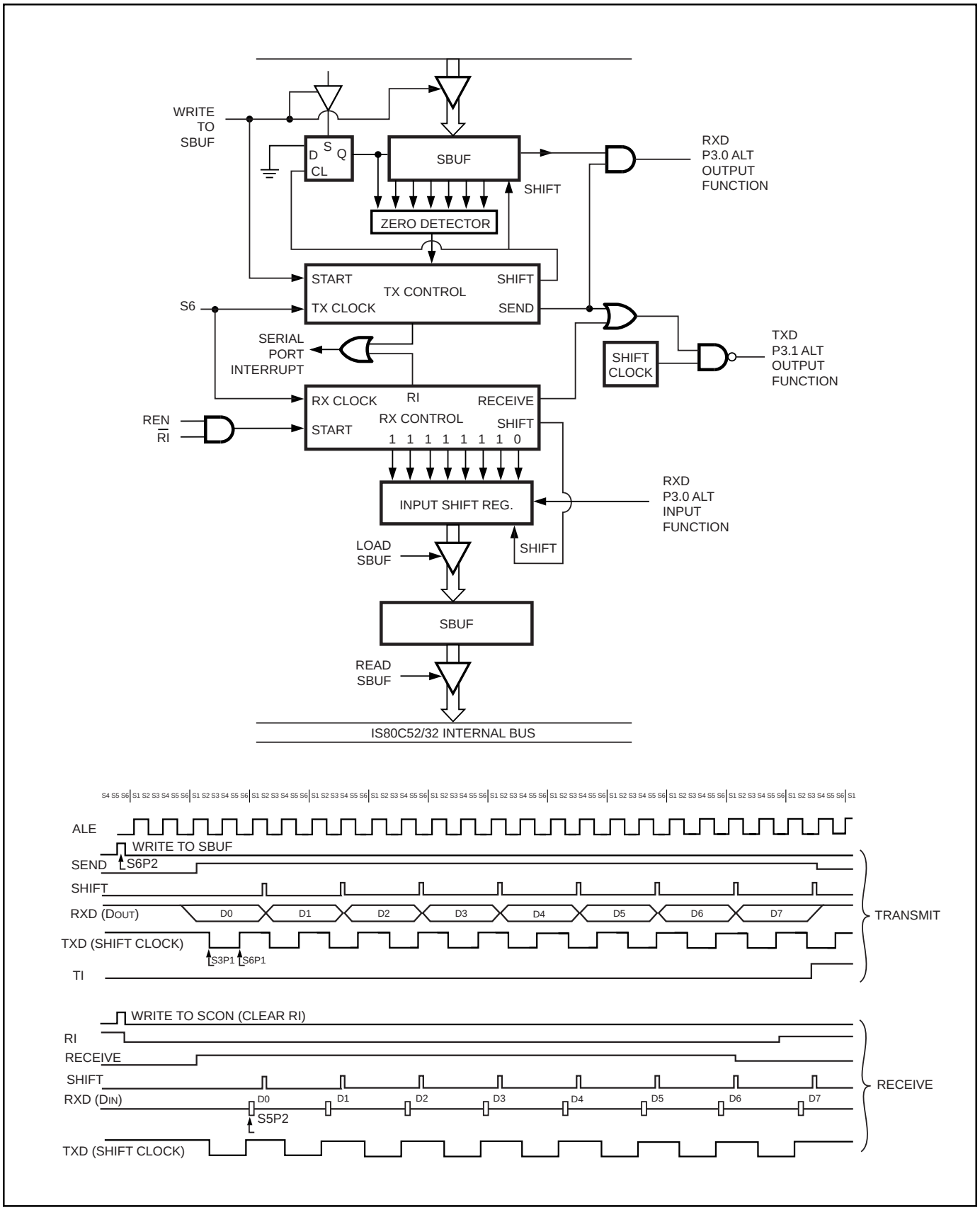

Figure 15. Serial Port Mode 0

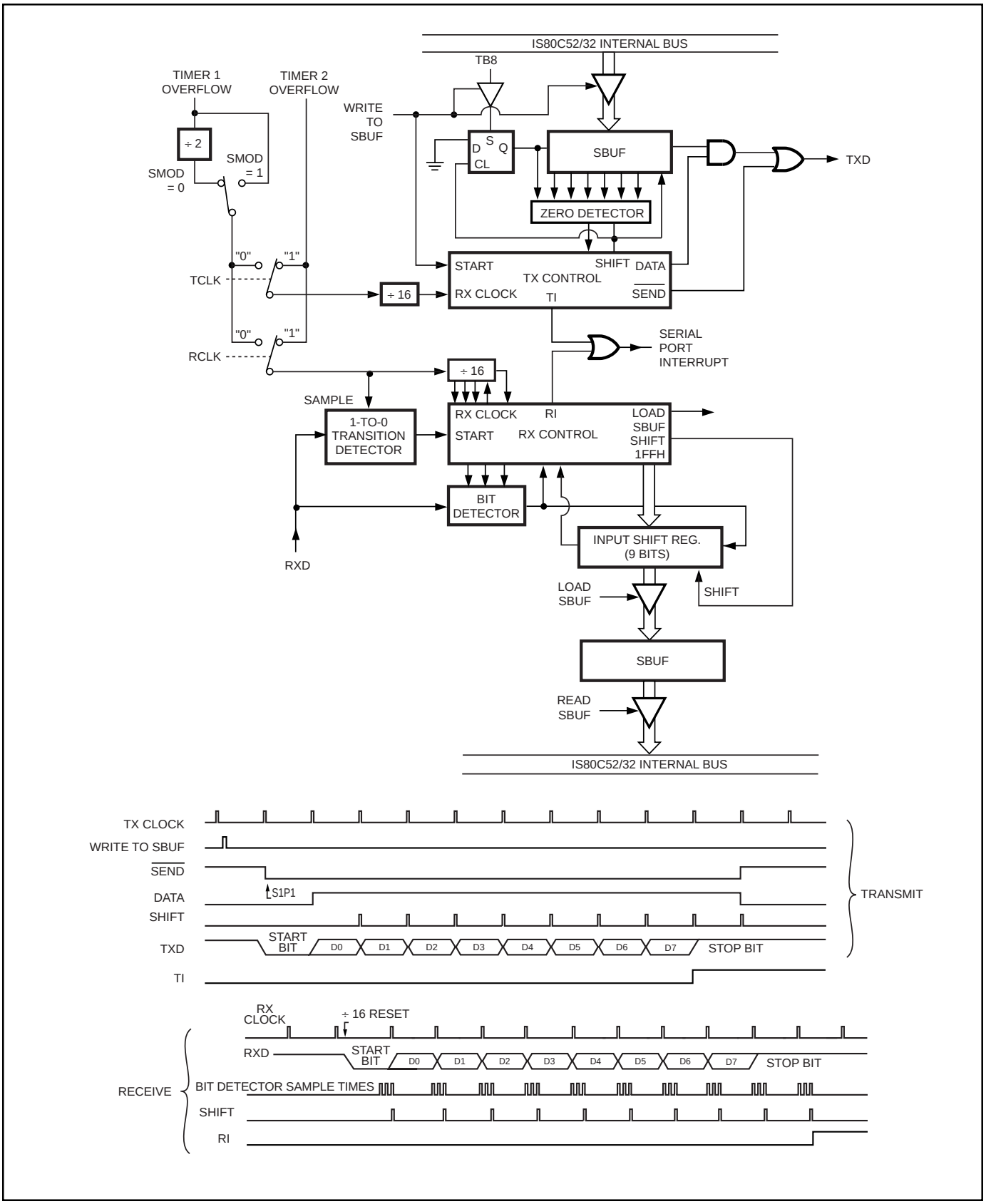

Figure 16. Serial Port Mode 1

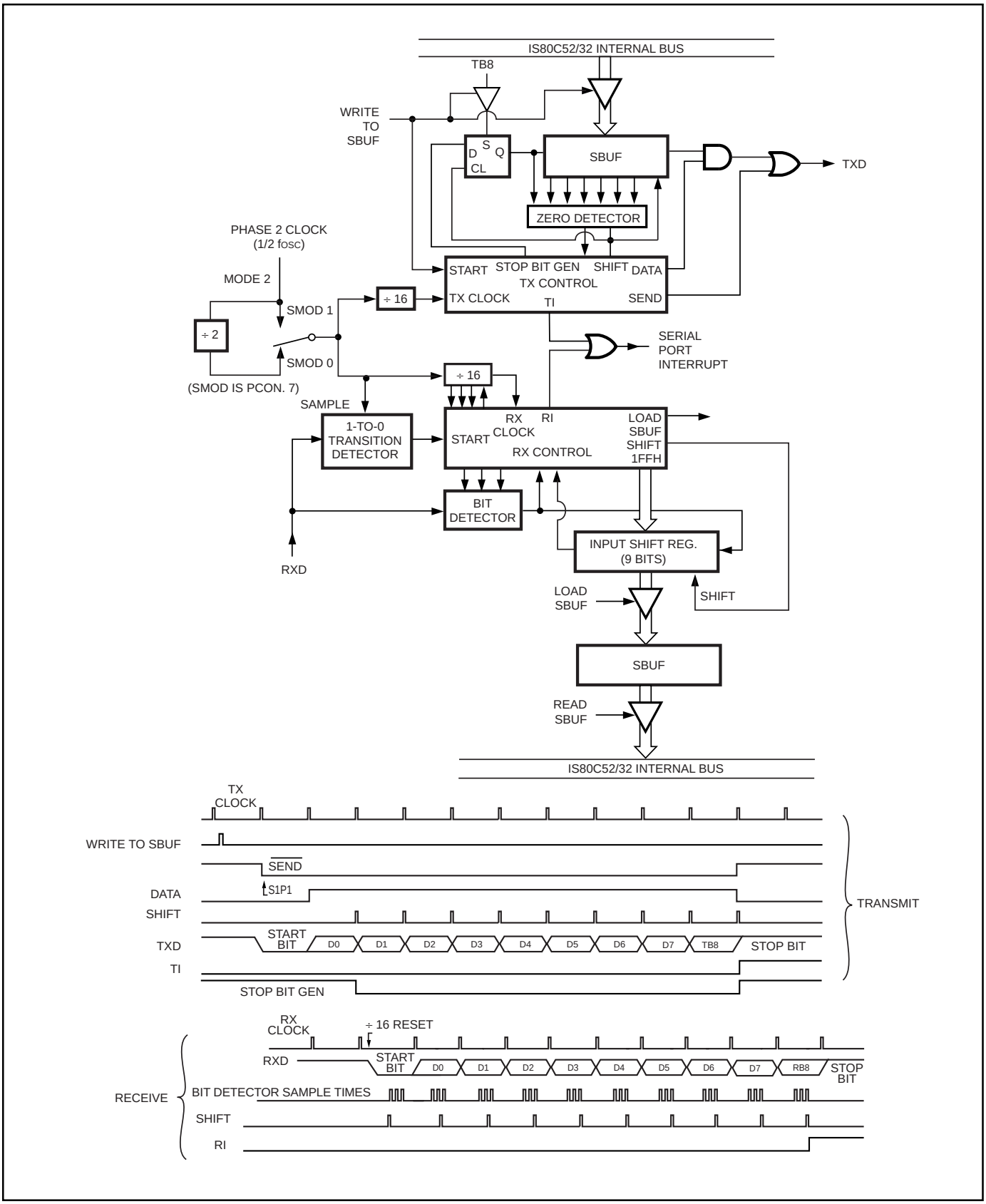

Figure 17. Serial Port Mode 2

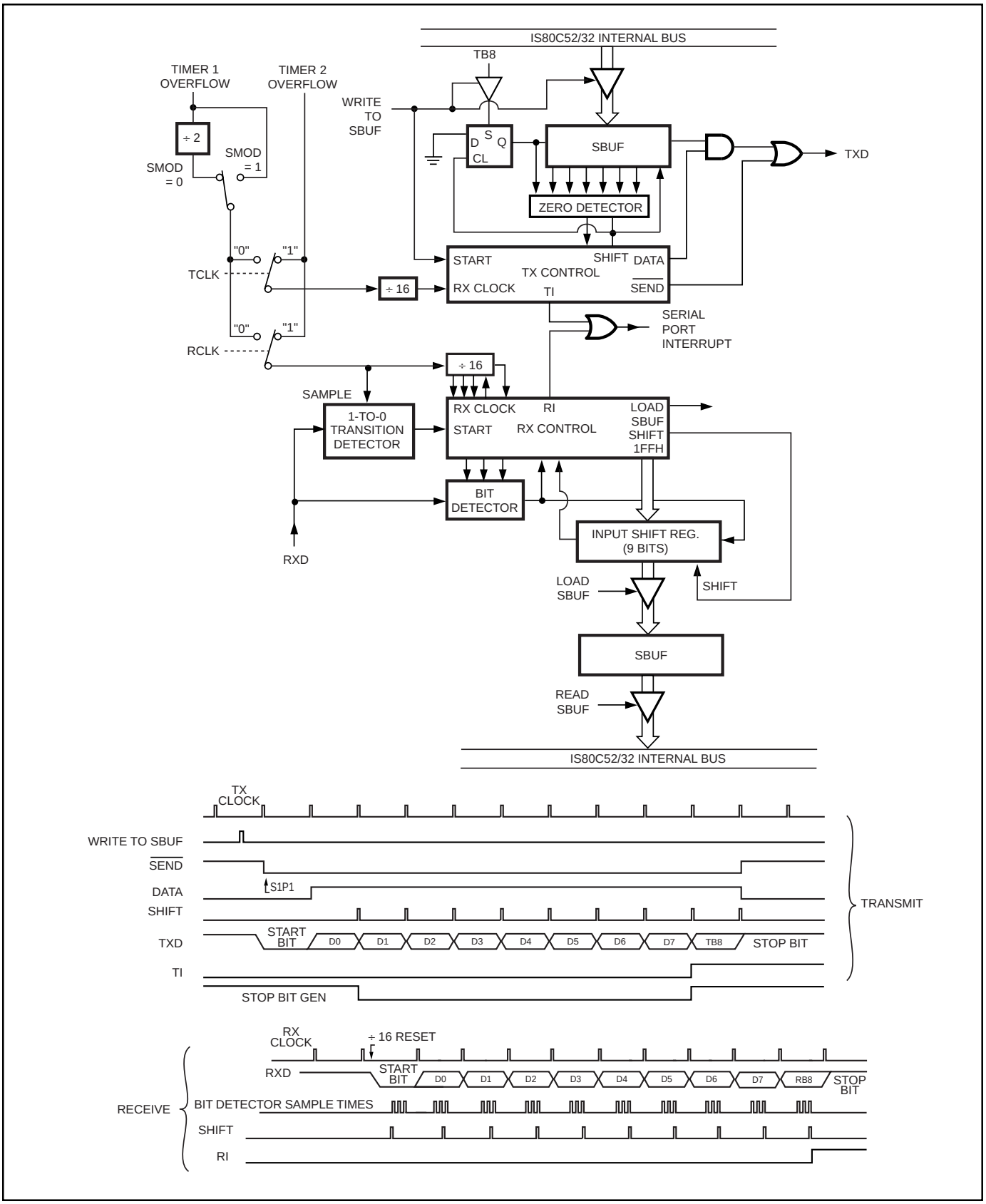

Figure 18. Serial Port Mode 3

INTERRUPT SYSTEM

The IS80C52/32 provides six interrupt sources: two external interrupts, three timer interrupts, and a serial port interrupt. These are shown in Figure 19.

The External Interrupts INT0 and INT1 can each be either level-activated or transition-activated, depending on bits IT0 and IT1 in Register TCON. The flags that actually generate these interrupts are the IE0 and IE1 bits in TCON. When the service routine is vectored to, hardware clears the flag that generated an external interrupt only if the interrupt was transition-activated. If the interrupt was level-activated, then the external requesting source (rather than the on-chip hardware) controls the request flag.

The Timer 0 and Timer 1 Interrupts are generated by TF0 and TF1, which are set by a rollover in their respective Timer/Counter registers (except for Timer 0 in Mode 3). When a timer interrupt is generated, the on-chip hardware clears the flag that generated it when the service routine is vectored to.

The Serial Port Interrupt is generated by the logical OR of RI and TI. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine normally must determine whether RI or TI generated the interrupt, and the bit must be cleared in software.

In the IS80C52/32, the Timer 2 Interrupt is generated by the logical OR of TF2 and EXF2. Neither of these flags is cleared by hardware when the service routine is vectored to. In fact, the service routine may have to determine whether TF2 or EXF2 generated the interrupt, and the bit must be cleared in software.

All of the bits that generate interrupts can be set or cleared by software, with the same result as though they had been set or cleared by hardware. That is, interrupts can be generated and pending interrupts can be canceled in software.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE (interrupt enable) at address 0A8H. As well as individual enable bits for each interrupt source, there is a global enable/disable bit that is cleared to disable all interrupts or set to turn on interrupts (see SFR IE).

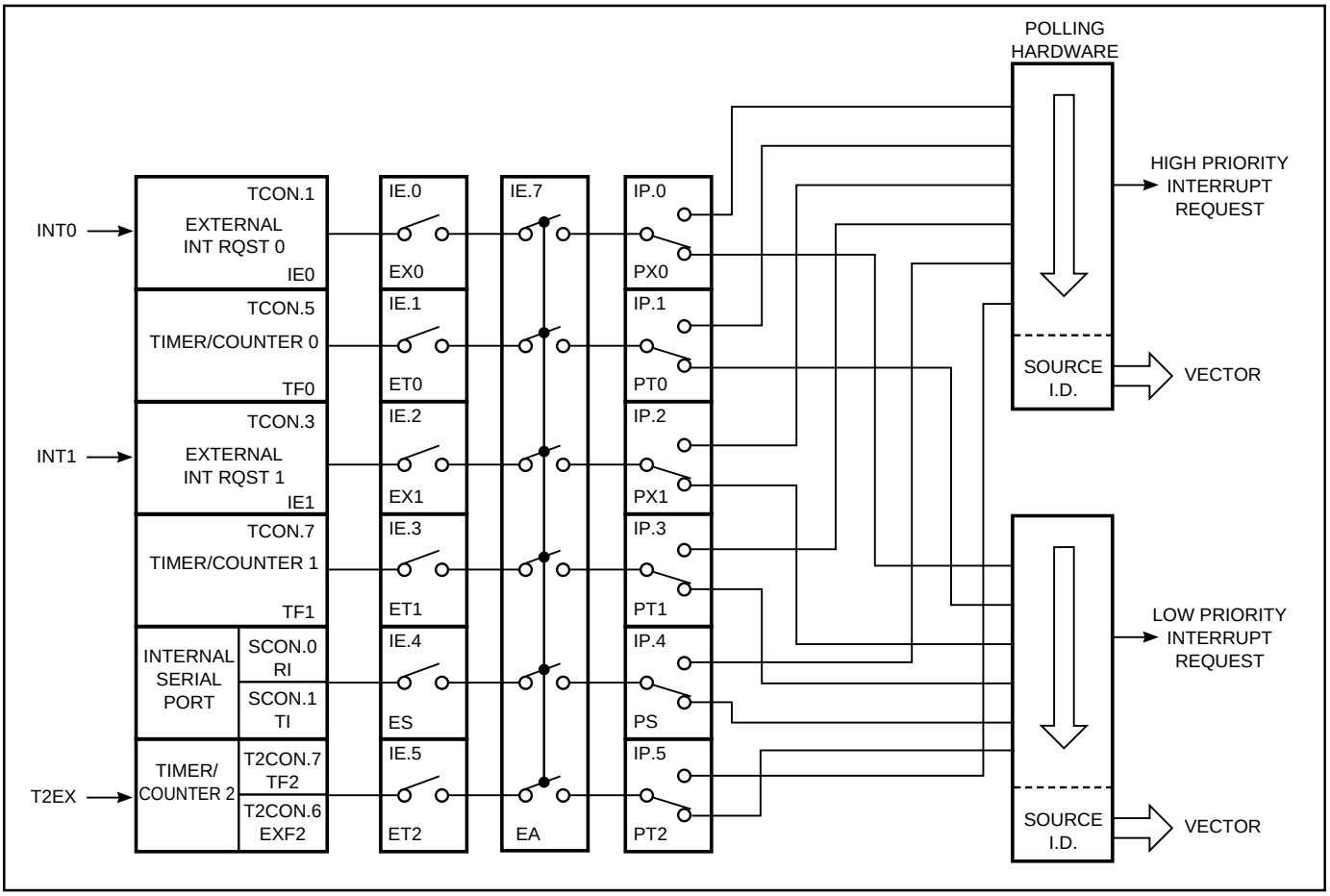

Figure 19. Interrupt System

Priority Level Structure

Each interrupt source can also be individually programmed to one of two priority levels by setting or clearing a bit in Special Function Register IP (interrupt priority) at address 0B8H. IP is cleared after a system reset to place all interrupts at the lower priority level by default. A low-priority interrupt can be interrupted by a high-priority interrupt but not by another low-priority interrupt. A high-priority interrupt can not be interrupted by any other interrupt source.

If two requests of different priority levels are received simultaneously, the request of higher priority level is serviced. If requests of the same priority level are received simultaneously, an internal polling sequence determines which request is serviced. Thus, within each priority level there is a second priority structure determined by the polling sequence, as follows:

	Source	Priority Within Level
1.	IE0	(Highest)
2.	TF0	
3.	IE1	
4.	TF1	
5.	RI + TI	
6.	TF2 + EXF2	(Lowest)

Note that the "priority within level" structure is only used to resolve *simultaneous requests of the same priority level*.

How Interrupts Are Handled

The interrupt flags are sampled at S5P2 of every machine cycle. The samples are polled during the following machine cycle (the Timer 2 interrupt cycle is different, as described in the Response Timer Section). If one of the flags was in a set condition at S5P2 of the preceding cycle, the polling cycle will find it and the interrupt system will generate an LCALL to the appropriate service routine, provided this hardware generated LCALL is not blocked by any of the following conditions:

1. An interrupt of equal or higher priority level is already in progress.
2. The current (polling) cycle is not the final cycle in the execution of the instruction in progress.
3. The instruction in progress is RETI or any write to the IE or IP registers.

Any of these three conditions will block the generation of the LCALL to the interrupt service routine. Condition 2 ensures that the instruction in progress will be completed before vectoring to any service routine. Condition 3 ensures that if the instruction in progress is RETI or any access to IE or IP, then at least one more instruction will be executed before any interrupt is vectored to.

The polling cycle is repeated with each machine cycle, and the values polled are the values that were present at S5P2 of the previous machine cycle. If an active interrupt flag is not being serviced because of one of the above conditions and is not still active when the blocking condition is removed, the denied interrupt will not be serviced. In other words, the fact that the interrupt flag was once active but not serviced is not remembered. Every polling cycle is new. The polling cycle/LCALL sequence is illustrated in Figure 20.

Note that if an interrupt of higher priority level goes active prior to S5P2 of the machine cycle labeled C3 in Figure 20, then in accordance with the above rules it will be serviced during C5 and C6, without any instruction of the lower priority routine having been executed.

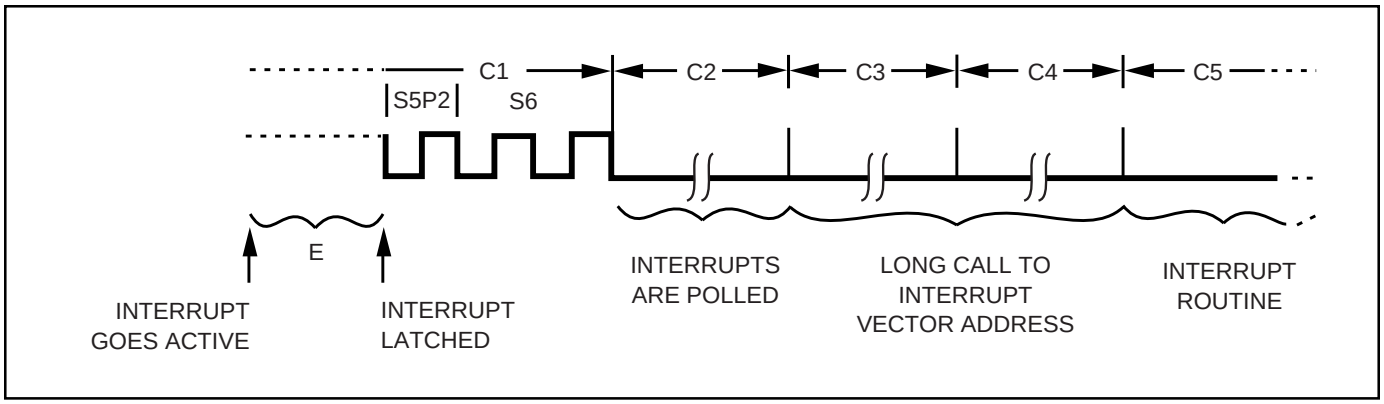

Figure 20. Interrupt Response Timing Diagram

Thus, the processor acknowledges an interrupt request by executing a hardware-generated LCALL to the appropriate servicing routine. In some cases it also clears the flag that generated the interrupt, and in other cases it does not. It never clears the Serial Port or Timer 2 flags. This must be done in the user's software. The processor clears an external interrupt flag (IE0 or IE1) only if it was transition-activated. The hardware-generated LCALL pushes the contents of the Program Counter onto the stack (but it does not save the PSW) and reloads the PC with an address that depends on the source of the interrupt being serviced, as follows:

Interrupt Source	Interrupt Request Bits	Cleared by Hardware	Vector Address
$\overline{\text{INT0}}$	IE0	No (level) Yes (trans.)	0003H
Timer 0	TF0	Yes	000BH
$\overline{\text{INT1}}$	IE1	No (level) Yes (trans.)	0013H
Timer 1	TF1	Yes	001BH
Serial Port	RI, TI	No	0023H
Timer 2	TF2, EXF2	No	002BH
System Reset	RST		0000H

Execution proceeds from that location until the RETI instruction is encountered. The RETI instruction informs the processor that this interrupt routine is no longer in progress, then pops the top two bytes from the stack and reloads the Program Counter. Execution of the interrupted program continues from where it left off.

Note that a simple RET instruction would also have returned execution to the interrupted program, but it would have left the interrupt control system thinking an interrupt was still in progress.

Interrupt	Flag	SFR Register and Bit Position
External 0	IE0	TCON.1
External 1	IE1	TCON.3
Timer 1	TF1	TCON.7
Timer 0	TF0	TCON.5
Serial Port	TI	SCON.1
Serial Port TF2	RI T2CON.7	SCON.0Timer 2
Timer 2	EXF2	T2CON.6

When an interrupt is accepted the following action occurs:

1. The current instruction completes operation.
2. The PC is saved on the stack.
3. The current interrupt status is saved internally.
4. Interrupts are blocked at the level of the interrupts.
5. The PC is loaded with the vector address of the ISR (interrupts service routine).
6. The ISR executes.

The ISR executes and takes action in response to the interrupt. The ISR finishes with RETI (return from interrupt) instruction. This retrieves the old value of the PC from the stack and restores the old interrupt status. Execution of the main program continues where it left off.

External Interrupts

The external sources can be programmed to be level-activated or transition-activated by setting or clearing bit IT1 or IT0 in Register TCON. If ITx= 0, external interrupt x is triggered by a detected low at the INTx pin. If ITx = 1, external interrupt x is edge-triggered. In this mode if successive samples of the INTx pin show a high in one cycle and a low in the next cycle, interrupt request flag IEx in TCON is set. Flag bit IEx then requests the interrupt.

Since the external interrupt pins are sampled once each machine cycle, an input high or low should hold for at least 12 oscillator periods to ensure sampling. If the external interrupt is transition-activated, the external source has to hold the request pin high for at least one machine cycle, and then hold it low for at least one machine cycle to ensure that the transition is seen so that interrupt request flag IEx will be set. IEx will be automatically cleared by the CPU when the service routine is called.

If the external interrupt is level-activated, the external source has to hold the request active until the requested interrupt is actually generated. Then the external source must deactivate the request before the interrupt service routine is completed, or else another interrupt will be generated.

Response Time

The INT0 and INT1 levels are inverted and latched into the interrupt flags IE0 and IE1 at S5P2 of every machine cycle. Similarly, the Timer 2 flag EXF2 and the Serial Port flags RI and TI are set at S5P2. The values are not actually polled by the circuitry until the next machine cycle.

The Timer 0 and Timer 1 flags, TF0 and TF1, are set at S5P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle. However, the Timer 2 flag TF2 is set at S2P2 and is polled in the same cycle in which the timer overflows.

If a request is active and conditions are right for it to be acknowledged, a hardware subroutine call to the requested service routine will be the next instruction executed. The call itself takes two cycles. Thus, a minimum of three complete machine cycles elapsed between activation of an external interrupt request and the beginning of execution of the first instruction of the service routine. Figure 19 shows response timings.

A longer response time results if the request is blocked by one of the three previously listed conditions. If an interrupt of equal or higher priority level is already in progress, the additional wait time depends on the nature of the other interrupt's service routine. If the instruction in progress is not in its final cycle, the additional wait time cannot be more than three cycles, since the longest instructions (MUL and DIV) are only four cycles long. If the instruction in progress is RETI or an access to IE or IP, the additional wait time cannot be more than five cycles (a maximum of one more cycle to complete the instruction in progress, plus four cycles to complete the next instruction if the instruction is MUL or DIV).

Thus, in a single-interrupt system, the response time is always more than three cycles and less than nine cycles.

Single-Step Operation

The IS80C52/32 interrupt structure allows single-step execution with very little software overhead. As previously noted, an interrupt request will not be serviced while an interrupt of equal priority level is still in progress, nor will it be serviced after RETI until at least one other instruction has been executed. Thus, once an interrupt routine has been entered, it cannot be reentered until at least one instruction of the interrupted program is executed. One way to use this feature for single-step operation is to program one of the external interrupts (for example, $\overline{INT0}$) to be level-activated. The service routine for the interrupt will terminate with the following code:

```
JNB     P3.2,$      ;Wait Here Till INT0 Goes High
JB      P3.2,$      ;Now Wait Here Till it Goes Low
RETI                ;Go Back and Execute One
                     Instruction
```

If the $\overline{INT0}$ pin, which is also the P3.2 pin, is held normally low, the CPU will go right into the External Interrupt 0 routine and stay there until $\overline{INT0}$ is pulsed (from low-to-high-to-low). Then it will execute RETI, go back to the task program, execute one instruction, and immediately reenter the External Interrupt 0 routine to await the next pulsing of P3.2. One step of the task program is executed each time P3.2 is pulsed.

OTHER INFORMATION

Reset

The reset input is the RST pin, which is the input to a Schmitt Trigger.

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), *while the oscillator is running*. The CPU responds by generating an internal reset, with the timing shown in Figure 21.

The external reset signal is asynchronous to the internal clock. The RST pin is sampled during State 5 Phase 2 of every machine cycle. The port pins will maintain their current activities for 19 oscillator periods after a logic 1 has been sampled at the RST pin; that is, for 19 to 31 oscillator periods after the external reset signal has been applied to the RST pin.

The internal reset algorithm writes 0s to all the SFRs except the port latches, the Stack Pointer, and SBUF. The port latches are initialized to FFH, the Stack Pointer to 07H, and SBUF is indeterminate. Table 11 lists the SFRs and their reset values.

Then internal RAM is not affected by reset. On power-up the RAM content is indeterminate.

Table 11. Reset Values of the SFR's

SFR Name	Reset Value
PC	0000H
ACC	00H
B	00H
PSW	00H
SP	07H
DPTR	0000H
P0-P3	FFH
IP	XX000000B
IE	0X000000B
TMOD	00H
TCON	00H
T2CON	00H
TH0	00H
TL0	00H
TH1	00H
TL1	00H
TH2	00H
TL2	00H
RCAP2H	00H
RCAP2L	00H
SCON	00H
SBUF	Indeterminate
PCON	0XXX0000B

Power-on Reset

An automatic reset can be obtained when Vcc goes through a 10µF capacitor and GND through an 8.2K resistor, providing the Vcc rise time does not exceed 1 msec and the oscillator start-up time does not exceed 10 msec. This power-on reset circuit is shown in Figure 21. The CMOS devices do not require the 8.2K pulldown resistor, although its presence does no harm.

When power is turned on, the circuit holds the RST pin high for an amount of time that depends on the value of the capacitor and the rate at which it charges. To ensure a good reset, the RST pin must be high long enough to allow the oscillator time to start-up (normally a few msec) plus two machine cycles.

Note that the port pins will be in a random state until the oscillator has start and the internal reset algorithm has written 1s to them.

With this circuit, reducing Vcc quickly to 0 causes the RST pin voltage to momentarily fall below 0V. However, this voltage is internally limited, and will not harm the device.

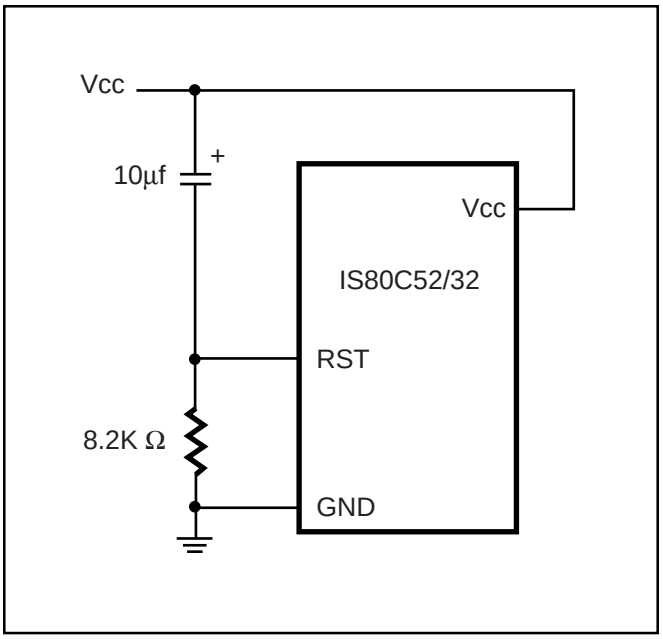

Figure 21. Power-On Reset Circuit

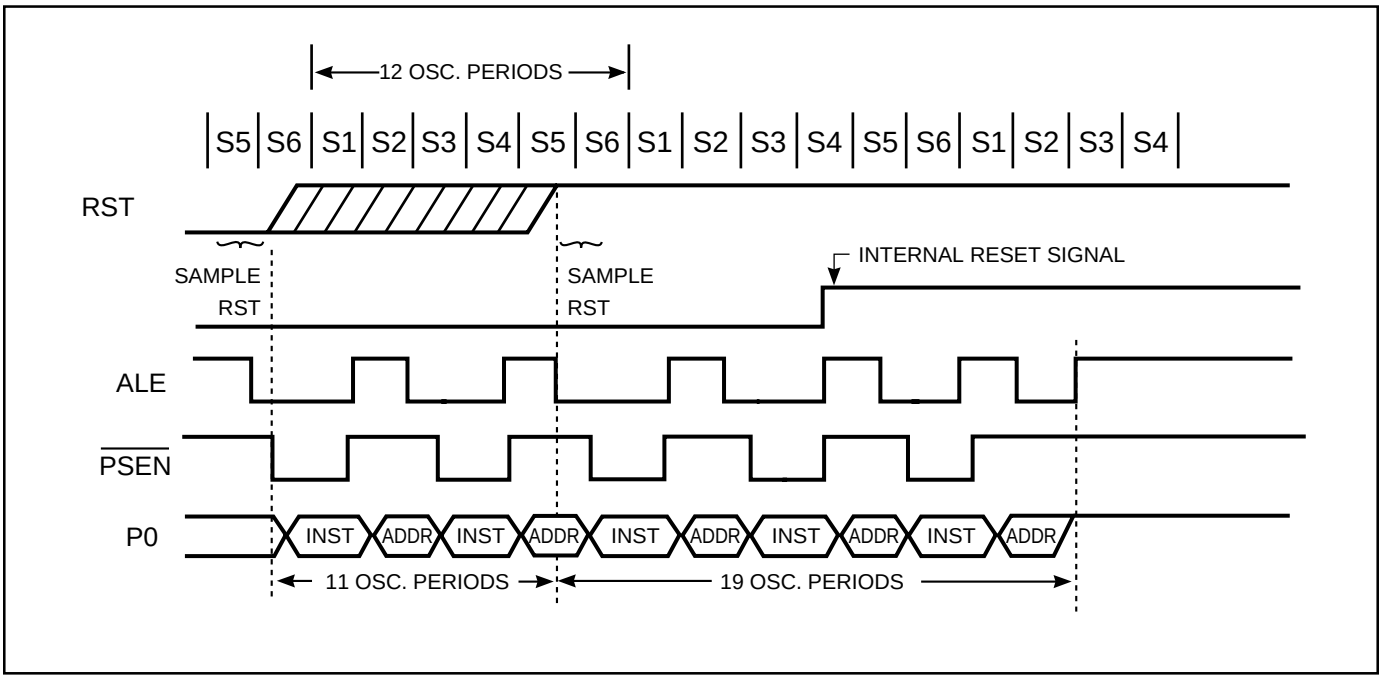

Figure 22. Reset Timing

Power-Saving Modes of Operation

The IS80C52/32 has two power-reducing modes. Idle and Power-down. The input through which backup power is supplied during these operations is Vcc. Figure 23 shows the internal circuitry which implements these features. In the Idle mode (IDL = 1), the oscillator continues to run and the Interrupt, Serial Port, and Timer blocks continue to be clocked, but the clock signal is gated off to the CPU. In Power-down (PD = 1), the oscillator is frozen. The Idle and Power-down modes are activated by setting bits in Special Function Register PCON.

Idle Mode

An instruction that sets PCON.0 is the last instruction executed before the Idle mode begins. In the Idle mode, the internal clock signal is gated off to the CPU, but not to the Interrupt, Timer, and Serial Port functions. The CPU status is preserved in its entirety: the Stack Pointer, Program Counter, Program Status Word, Accumulator, and all other registers maintain their data during Idle. The port pins hold the logical states they had at the time Idle was activated. ALE and PSEN hold at logic high levels.

There are two ways to terminate the Idle. Activation of any enabled interrupt will cause PCON.0 to be cleared by hardware, terminating the Idle mode. The interrupt will be serviced, and following RETI the next instruction to be executed will be the one following the instruction that put the device into Idle.

The flag bits GF0 and GF1 can be used to indicate whether an interrupt occurred during normal operation or during an Idle. For example, an instruction that activates Idle can also set one or both flag bits. When Idle is terminated by an interrupt, the interrupt service routine can examine the flag bits.

The other way of terminating the Idle mode is with a hardware reset. Since the clock oscillator is still running, the hardware reset must be held active for only two machine cycles (24 oscillator periods) to complete the reset.

The signal at the RST pin clears the IDL bit directly and asynchronously. At this time, the CPU resumes program execution from where it left off; that is, at the instruction following the one that invoked the Idle Mode. As shown in Figure 22, two or three machine cycles of program execution may take place before the internal reset algorithm takes control. On-chip hardware inhibits access to the internal RAM during his time, but access to the port pins is not inhibited. To eliminate the possibility of unexpected outputs at the port pins, the instruction following the one that invokes Idle should not write to a port pin or to external data RAM.

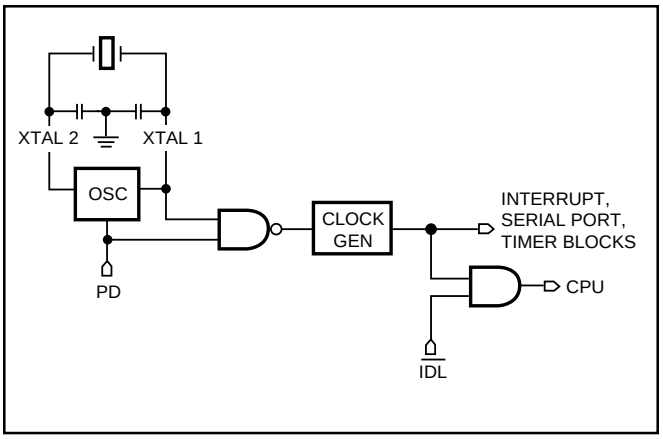

Figure 23. Idle and Power-Down Hardware

Power-down Mode

An instruction that sets PCON.1 is the last instruction executed before Power-down mode begins. In the Power-down mode, the on-chip oscillator stops. With the clock frozen, all functions are stopped, but the on-chip RAM and Special function Registers are held. The port pins output the values held by their respective SFRs. ALE and $\overline{PSEN}$ output lows.

In the Power-down mode of operation, Vcc can be reduced to as low as 2V. However, Vcc must not be reduced before the Power-down mode is invoked, and Vcc must be restored to its normal operating level before the Power-down mode is terminated. The reset that terminates Power-down also frees the oscillator. The reset should not be activated before Vcc is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize (normally less than 10 msec).

The only exit from power-down is a hardware reset. Reset redefines all the SFRs but does not change the on-chip RAM.

Table 12. Status of the External Pins During Idle and Power-down Modes.

Mode	Memory	ALE	$\overline{PSEN}$	PORT 0	PORT 1	PORT 2	PORT 3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

On-Chip Oscillators

The on-chip oscillator circuitry of the IS80C52/32 is a single stage linear inverter, intended for use as a crystal-controlled, positive reactance oscillator (Figure 24). In this application the crystal is operated in its fundamental response mode as an inductive reactance in parallel resonance with capacitance external to the crystal (Figure 24). Examples of how to drive the clock with external oscillator are shown in Figure 25.

The crystal specifications and capacitance values (C1 and C2 in Figure 24) are not critical. 20 pF to 30 pF can be used in these positions at a12 MHz to 24 MHz frequency with good quality crystals. (For ranges greater than 24 MHz refer to Figure 26.) A ceramic resonator can be used in place of the crystal in cost-sensitive applications. When a ceramic resonator is used, C1 and C2 are normally selected to be of somewhat higher values. The manufacturer of the ceramic resonator should be consulted for recommendation on the values of these capacitors.

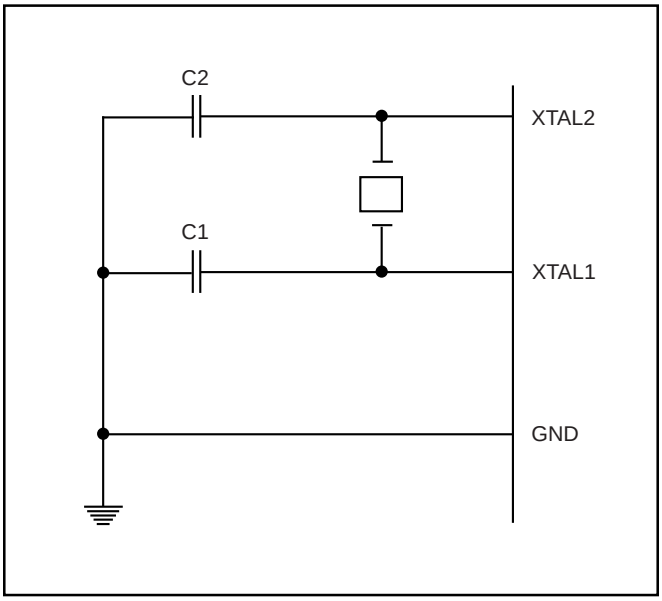

Figure 24. Oscillator Connections

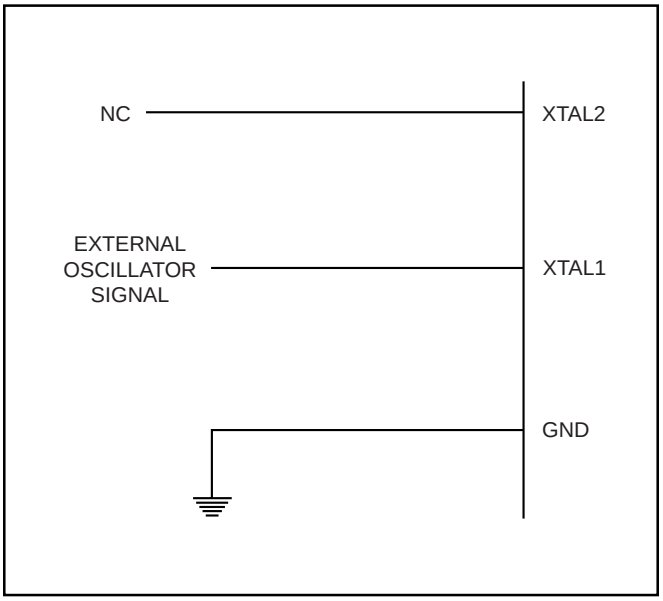

Figure 25. External Clock Drive Configuration

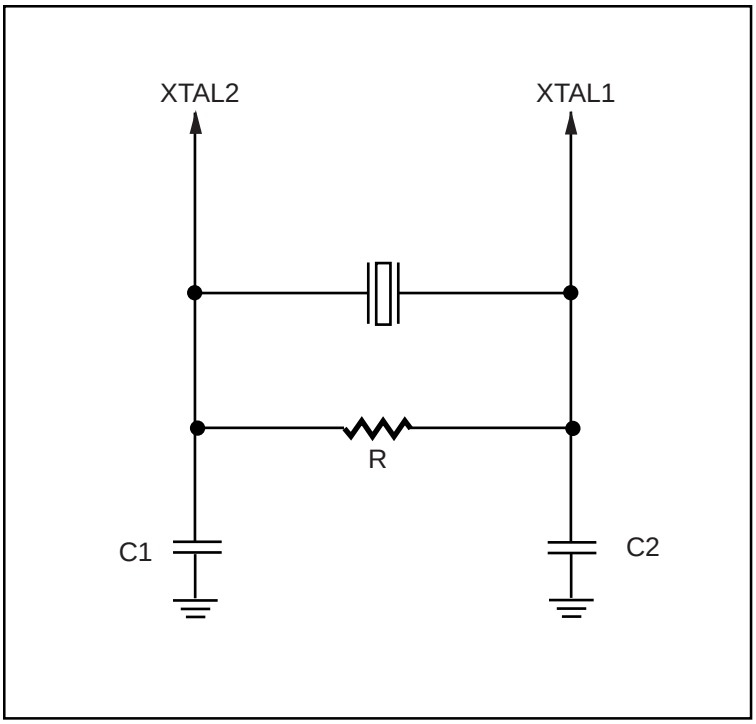

Figure 26. For High Speed (> 24 MHz)

Note:
When the frequency is higher than 24 MHz, please refer to Table 13 for recommended value of C1, C2, and R.

Table 13. Recommended Value for C1, C2, R

	Frequency Range	
	3.5 MHz-24 MHz	**30 MHz-40 MHz**
C1	20 pF-30 pF	3 pF-10 pF
C2	20 pF-30 pF	3 pF-10 pF
R	Not Apply	6.2K-10K

ROM Verification

The address of the program memory location to be read is applied to Port 1 and pins P2.4-P2.0. The other pins should be held at the “Verify” level are indicated in Figure 26. The contents of the addressed locations exits on Port 0. External pullups are required on Port 0 for this operation. Figure 26 shows the setup to verify the program memory.

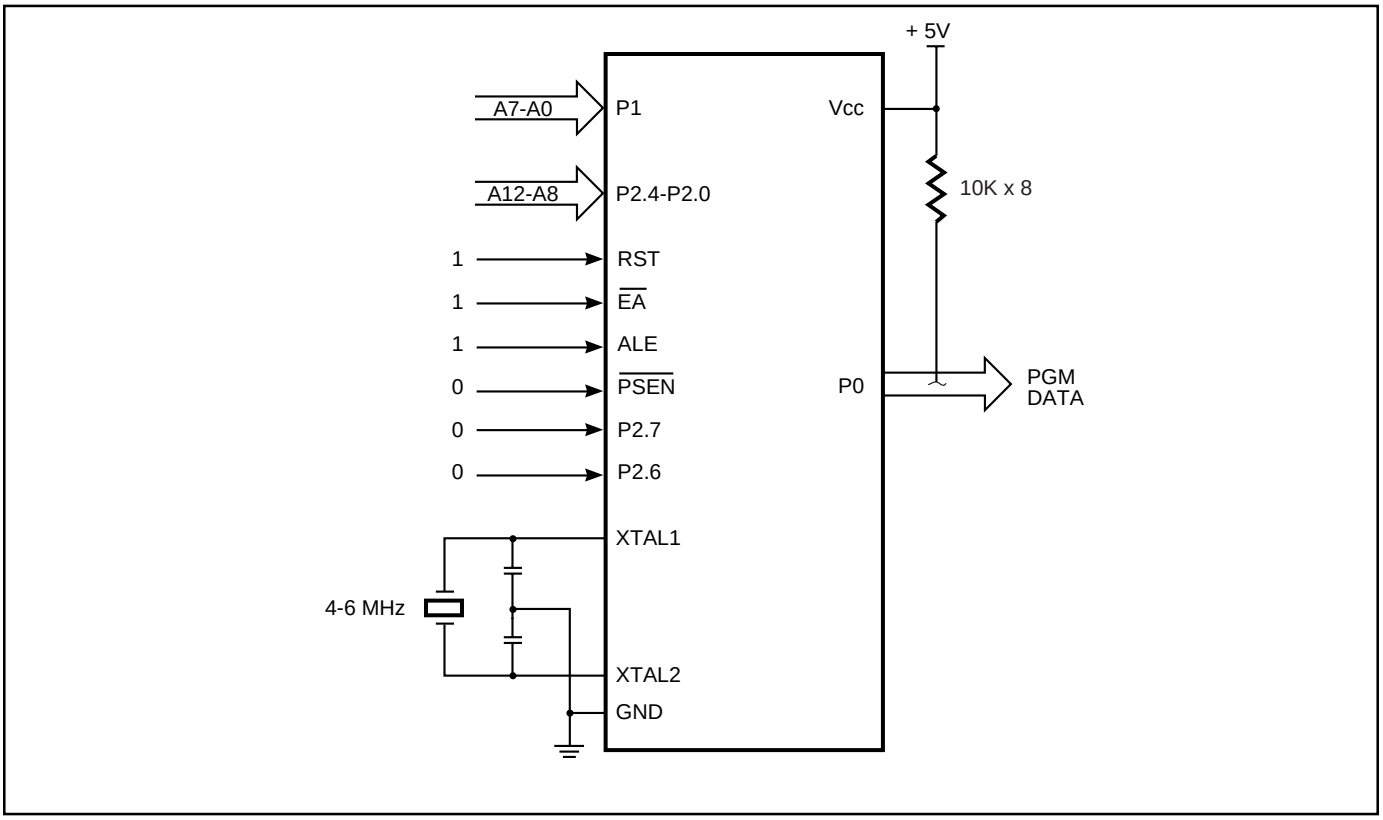

Figure 27. ROM Verification

ROM Lock System

The program lock system, when programmed, protects the ROM code against software piracy. The IS80C52/32 has a two-level program lock system (see Table 14) and a 32-byte encryption table. No matter what lock bit is, the user submits the encryption table with his or her code in verify ROM mode. Both the lock-bit and encryption array

Encryption Array

Within the ROM array are 32 bytes of Encryption Array that are initially unprogrammed (all 1's). Every time that a byte is addressed during verify, five address lines are used to select a byte of the Encryption Array.This byte is then exclusive-NOR'ed (XNOR) with the code byte, creating an Encryption verify byte. The algorithm, with the array in the unprogrammed state (all 1's), will return the code in its original, unmodified form.

When using the encryption array, one important factor needs to be considered. If a code byte has the value 0FFH, verifying the byte will produce the encryption byte value. If a large block (> 32 bytes) of code is left unprogrammed, a verification routine will display the contents of the encryption array. For this reason, all unused code bytes should be programmed with some value other than 0FFH, and not all of them the same value.

Table 14. Program Lock Bits

	LB1	LB2	Protection Type
1	U	U	No Program Lock Features enabled. (Code verify will still be encrypted by the Encryption Array if Programmed)
2	P	U	MOVC instructions executed from external program memory are diabled form fetching code bytes from internal memory, EA is sampled and latched on Reset.
3	P	P	Same as 2, also ROM verify is disabled.

ABSOLUTE MAXIMUM RATINGS[1]

Symbol	Parameter	Value	Unit
V_{TERM}	Terminal Voltage with Respect to GND[2]	–2.0 to +7.0	V
T_{BIAS}	Temperature Under Bias[3]	–40 to +85	°C
T_{STG}	Storage Temperature	–65 to +125	°C
P_T	Power Dissipation	1.5	W

Note:
1. Stress greater than 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 reliability.
2. Minimum DC input voltage is –0.5V. During transitions, inputs may undershoot to –2.0V for periods less than 20 ns. Maximum DC voltage on output pins is Vcc + 0.5V which may overshoot to Vcc + 2.0V for periods less than 20 ns.
3. Operating temperature is for commercial products only defined by this specification.

OPERATING RANGE[1]

Range	Ambient Temperature	V_{CC}	Oscillator Frequency
Commercial	0°C to +70°C	5V ± 10%	3.5 to 40 MHz
Industrial	–40°C to +85°C	5V ± 10%	3.5 to 40 MHz

Note:
1. Operating ranges define those limits between which the functionality of the device is guaranteed.

DC CHARACTERISTICS

(T_A = 0°C to 70°C; Vcc = 3.3V ± 10%; GND = 0V)

Symbol	Parameter	Test conditions	Min	Max	Unit
V_{IL}	Input low voltage (All except $\overline{EA}$)		–0.5	0.2Vcc – 0.1	V
$V_{IL}1$	Input low voltage ($\overline{EA}$)		–0.5	0.2Vcc – 0.3	V
V_{IH}	Input high voltage (All except XTAL 1, RST)		0.2Vcc + 0.9	Vcc + 0.5	V
$V_{IH}1$	Input high voltage (XTAL 1)		0.7Vcc	Vcc + 0.5	V
V_{SCH}+	RST positive schmitt-trigger threshold voltage		0.7Vcc	Vcc + 0.5	V
V_{SCH}–	RST negative schmitt-trigger threshold voltage		0	0.2Vcc	V
Vol(1)	Output low voltage	Iol = 100 µA	—	0.3	V
	(Ports 1, 2, 3)	I_{OL} = 1.6 mA	—	0.45	V
		I_{OL} = 3.5 mA	—	1.0	V
$V_{OL}1$(1)	Output low voltage	I_{OL} = 200 µA	—	0.3	V
	(Port 0, ALE, $\overline{PSEN}$)	I_{OL} = 3.2 mA	—	0.45	V
		I_{OL} = 7.0 mA	—	1.0	V
V_{OH}	Output high voltage (Ports 1, 2, 3, ALE, $\overline{PSEN}$)	I_{OH} = –10 µA Vcc = 4.5V-5.5V	0.9Vcc	—	V
		I_{OL} = –25 µA	0.75Vcc	—	V
		I_{OL} = –60 µA	2.4	—	V
$V_{OH}1$	Output high voltage (Port 0, ALE, $\overline{PSEN}$)	I_{OH} = –80 µA Vcc = 4.5V-5.5V	0.9Vcc	—	V
		I_{OH} = –300 µA	0.75Vcc	—	V
		I_{OH} = –800 µA	2.4	—	V
I_{IL}	Logical 0 input current (Ports 1, 2, 3)	V_{IN} = 0.45V	—	–110	µA
I_{LI}	Input leakage current (Port 0)	0.45V < V_{IN} < Vcc	–10	10	µA
I_{TL}	Logical 1-to-0 transition current (Ports 1, 2, 3)	V_{IN} = 2.0V	—	–650	µA
R_{RST}	RST pulldown resister		50	300	KΩ

Note:

1. Under steady state (non-transient) conditions, Iol must be externally limited as follows:
   Maximum I_{OL} per port pin: 10 mA
   Maximum I_{OL} per 8-bit port
   Port 0: 26 mA
   Ports 1, 2, 3: 15 mA
   Maximum total I_{OL} for all output pins: 71 mA

   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification.

POWER SUPPLY CHARACTERISTICS

Symbol	Parameter	Test conditions	Min	Max	Unit
Icc	Power supply current(1)	Vcc = 5.0V			
	Active mode	12 MHz	—	20	mA
		16 MHz	—	26	mA
		20 MHz	—	32	mA
		24 MHz	—	38	mA
		32 MHz	—	50	mA
		40 MHz	—	62	mA
	Idle mode	12 MHz	—	5	mA
		16 MHz	—	6	mA
		20 MHz	—	7.6	mA
		24 MHz	—	9	mA
		32 MHz	—	12	mA
		40 MHz	—	15	mA
	Power-down mode	Vcc = 5V	—	50	µA

Note:
1. See Figures 28, 29, 30, and 31 for Icc test conditiions.

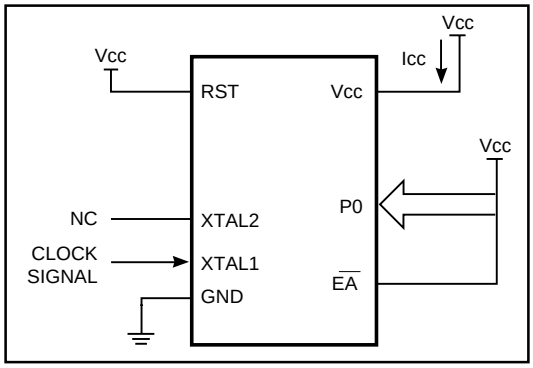

Figure 28. Active Mode

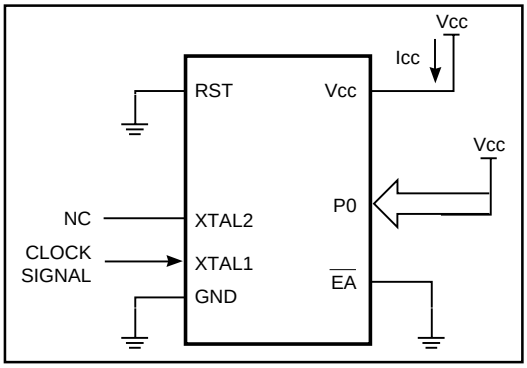

Figure 29. Idle Mode

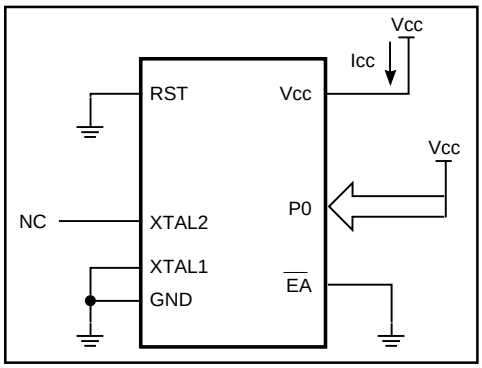

Figure 30. Power-down Mode

AC CHARACTERISTICS

(T_A = 0°C to 70°C; Vcc = 3.3V ± 10%; GND = 0V; Cl for Port 0, ALE and PSEN Outputs = 100 pF; Cl for other outputs = 80 pF)

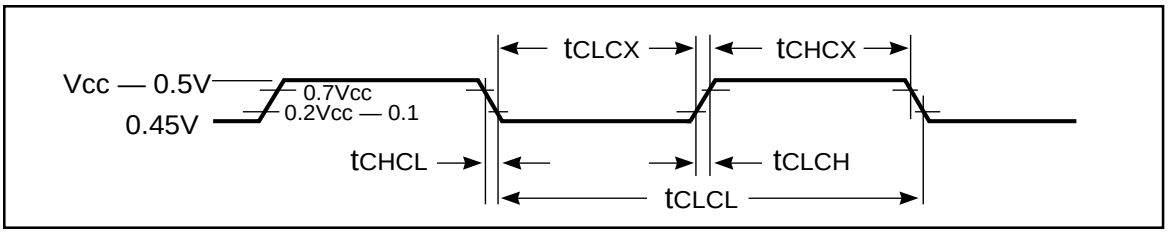

Figure 31. Icc Test Conditions

Note:
1. Clock signal waveform for Icc tests in active and idle mode (t_{CLCH} = t_{CHCL} = 5 ns)

EXTERNAL MEMORY CHARACTERISTICS

		24 MHz Clock		40 MHz Clock		Variable Oscillator (3.5-24 MHz)		
Symbol	**Parameter**	**Min**	**Max**	**Min**	**Max**	**Min**	**Max**	**Unit**
$1/t_{CLCL}$	Oscillator frequency	—	—	—	—	3.5	24	MHz
t_{LHLL}	ALE pulse width	43	—	40	—	$2t_{CLCL}$–40	—	ns
t_{AVLL}	Address valid to ALE low	2	—	9	—	t_{CLCL}–40	—	ns
t_{LLAX}	Address hold after ALE low	7	—	30	—	t_{CLCL}–35	—	ns
t_{LLIV}	ALE low to valid instr in	—	105	—	70	—	$3t_{CLCL}$–20	ns
t_{LLPL}	ALE low to $\overline{PSEN}$ low	2	—	15	—	t_{CLCL}–40	—	ns
t_{PLPH}	$\overline{PSEN}$ pulse width	80	—	65	—	$3t_{CLCL}$–45	—	ns
t_{PLIV}	$\overline{PSEN}$ low to valid instr in	—	73	—	45	—	$2t_{CLCL}$–10	ns
t_{PXIX}	Input instr hold after $\overline{PSEN}$	0	—	0	—	0	—	ns
t_{PXIZ}	Input instr float after $\overline{PSEN}$	—	73	—	25	—	$2t_{CLCL}$–10	ns
t_{AVIV}	Address to valid instr in	—	147	—	80	—	$4t_{CLCL}$–20	ns
t_{PLAZ}	$\overline{PSEN}$ low to address float	—	10	—	5	—	10	ns
t_{RLRH}	$\overline{RD}$ pulse width	150	—	100	—	$6t_{CLCL}$–100	—	ns
t_{WLWH}	WR pulse width	150	—	100	—	$6t_{CLCL}$–100	—	ns
t_{RLDV}	$\overline{RD}$ low to valid data in	—	114	—	90	—	$5t_{CLCL}$–95	ns
t_{RHDX}	Data hold after $\overline{RD}$	0	—	0	—	0	—	ns
t_{RHDZ}	Data float after $\overline{RD}$	—	63	—	50	—	$2t_{CLCL}$–70	ns
t_{LLDV}	ALE low to valid data in	—	244	—	150	—	$8t_{CLCL}$–90	ns
t_{AVDV}	Address to valid data in	—	285	—	180	—	$9t_{CLCL}$–90	ns
t_{LLWL}	ALE low to $\overline{RD}$ or $\overline{WR}$ low	75	175	60	95	$3t_{CLCL}$–50	$3t_{CLCL}$+50	ns
t_{AVWL}	Address to $\overline{RD}$ or $\overline{WR}$ low	77	—	65	—	$4t_{CLCL}$–90	—	ns
t_{QVWX}	Data valid to $\overline{WR}$ transition	2	—	10	—	t_{CLCL}–40	—	ns
t_{WHQX}	Data hold after $\overline{WR}$	2	—	10	—	t_{CLCL}–40	—	ns
t_{QVWH}	Data valid to $\overline{WR}$ high	219	—	165	—	$7t_{CLCL}$–70	—	ns
t_{RLAZ}	$\overline{RD}$ low to address float	—	63	—	0	—	$2t_{CLCL}$–20	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ high to ALE high	2	82	15	35	t_{CLCL}–40	t_{CLCL}+40	ns

EXTERNAL MEMORY CHARACTERISTICS

(Continued)

Symbol	Parameter	24 MHz Clock Min	24 MHz Clock Max	40 MHz Clock Min	40 MHz Clock Max	Variable Oscillator (3.5-24 MHz) Min	Variable Oscillator (3.5-24 MHz) Max	Unit
t_{XLXL}	Serial port clock cycle time	500	—	250	—	$12t_{CLCL}$–10	—	ns
t_{QVXH}	Output data setup to clock rising edge	284	—	170	—	$10t_{CLCL}$–133	—	ns
t_{XHQX}	Output data hold after clock rising edge	33	—	33	—	$2t_{CLCL}$–50	—	ns
t_{XHDX}	Input data hold after clock rising edge	0	—	0	—	0	—	ns
t_{XHDV}	Clock rising edge to input data valid	—	284	—	117	—	$10t_{CLCL}$–133	ns

EXTERNAL CLOCK DRIVE

Symbol	wParameter	Min	Max	Unit
$1/t_{CLCL}$	Oscillator Frequency	3.5	40	MHz
t_{CHCX}	High time	10	—	ns
t_{CLCX}	Low time	10	—	ns
t_{CLCH}	Rise time	—	10	ns
t_{CHCL}	Fall time	—	10	ns

ROM VERIFICATION CHARACTERISTICS

Symbol	Parameter	Min	Max	Unit
$1/t_{CLCL}$	Oscillator Frequency	2.5	40	MHz
t_{AVQV}	Address to data valid	—	$48t_{CLCL}$	
t_{ELQV}	ENABLE low to data valid	—	$48t_{CLCL}$	
t_{EHQZ}	Data float after ENABLE	0	$48t_{CLCL}$	

TIMING WAVEFORMS

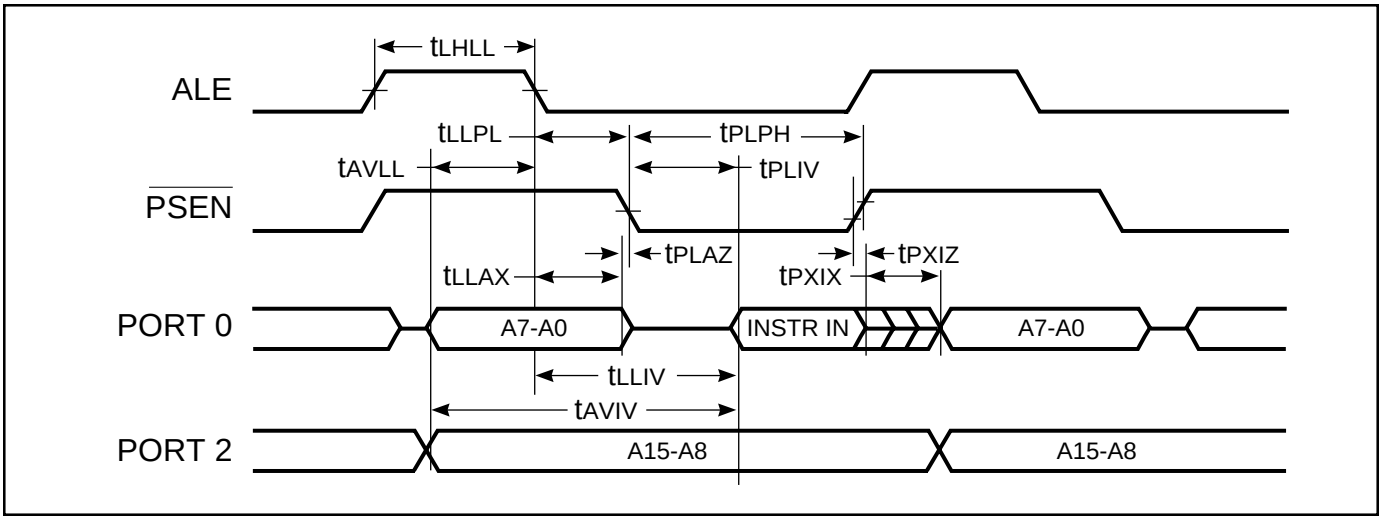

Figure 32. External Program Memory Read Cycle

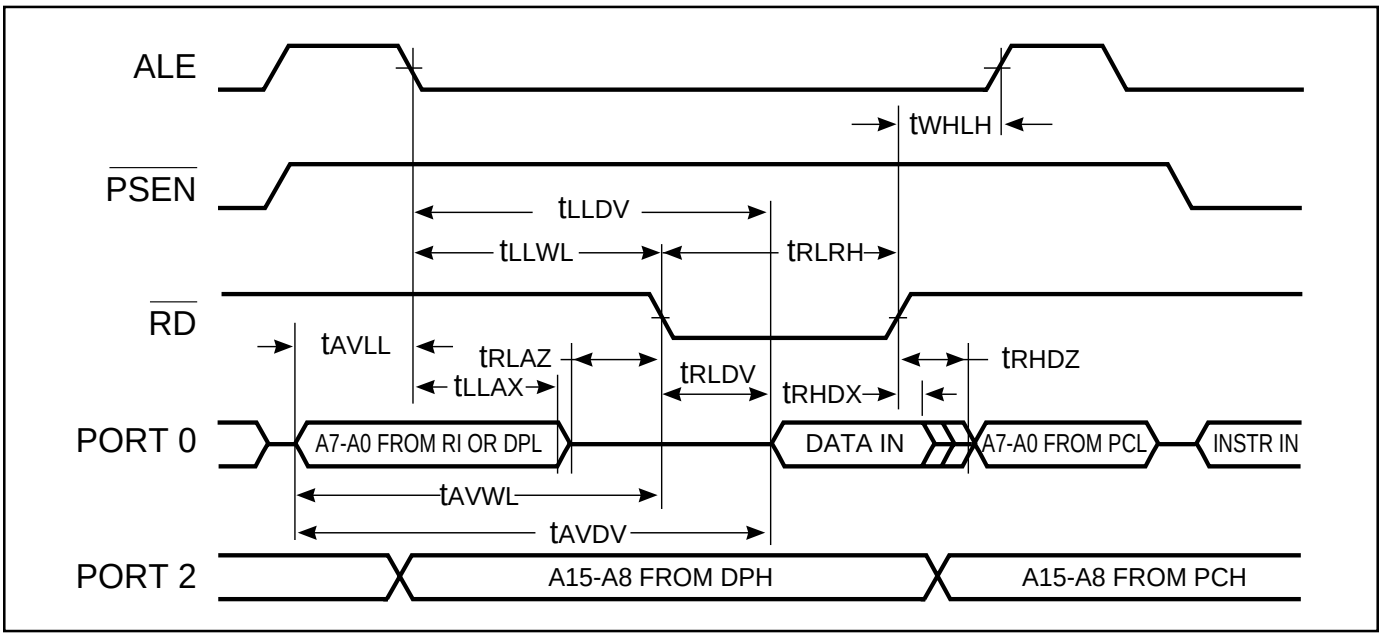

Figure 33. External Data Memory Read Cycle

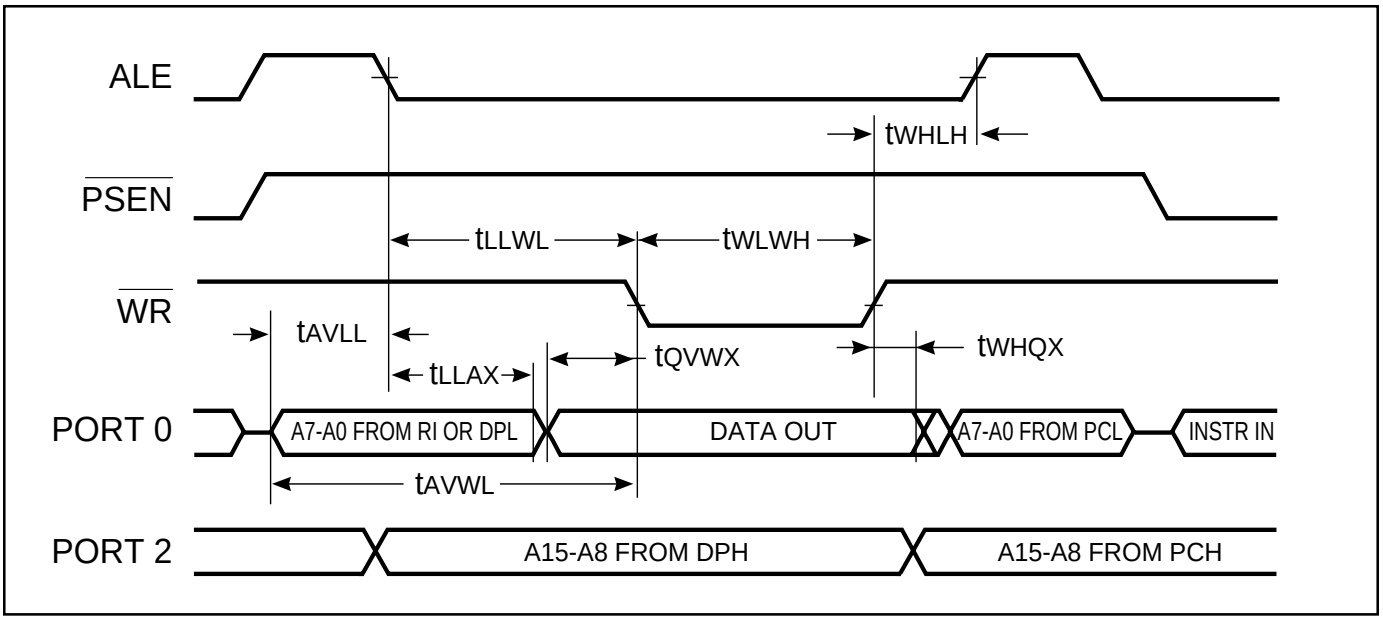

Figure 34. External Data Memory Write Cycle

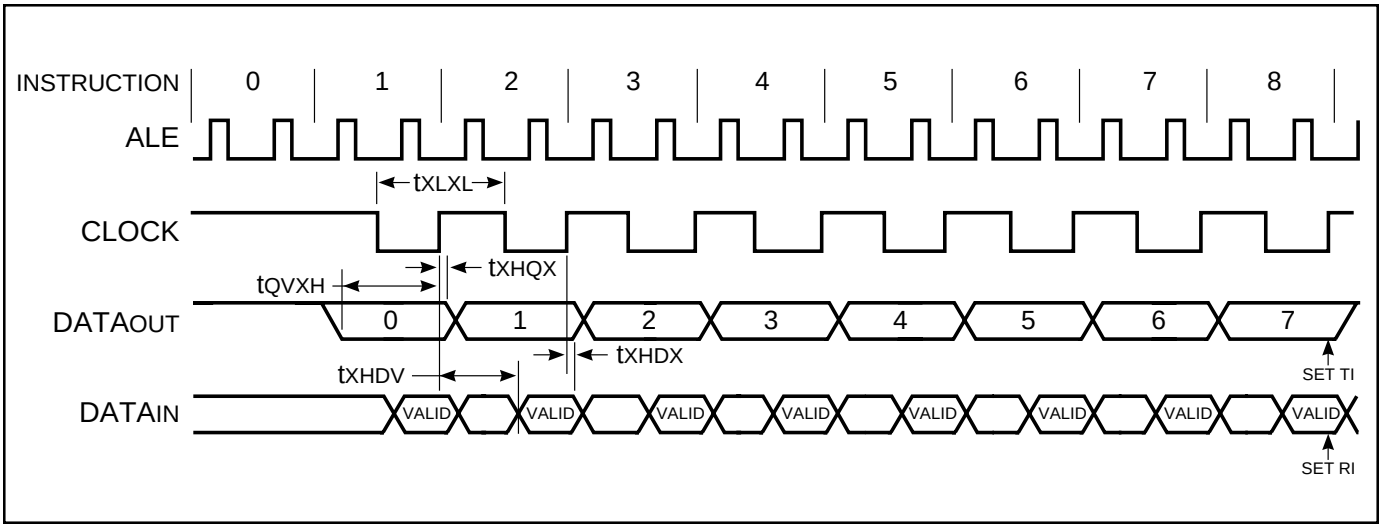

Figure 35. Shift Register Mode Timing Waveform

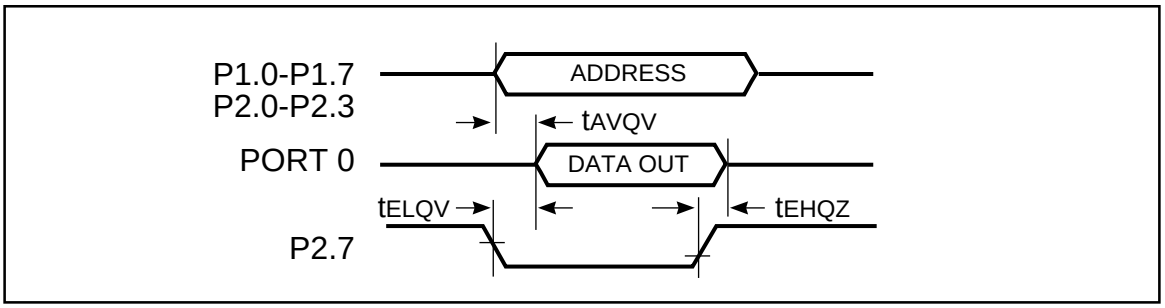

Figure 36. External Clock Drive Waveform

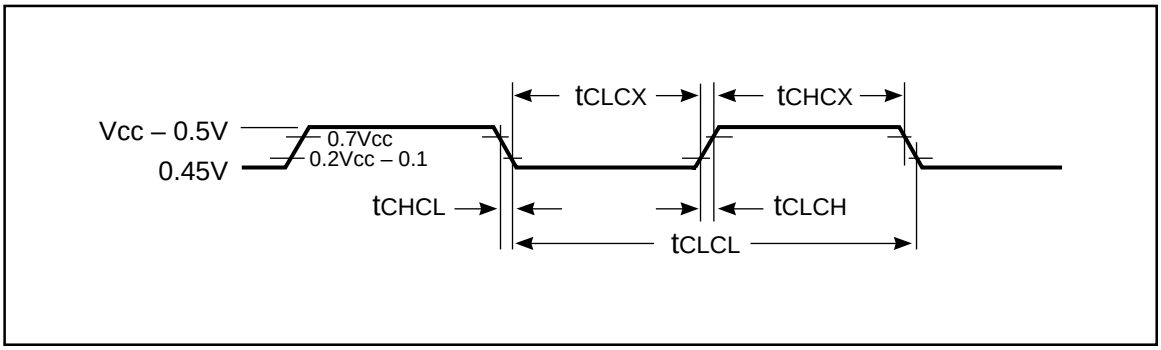

Figure 37. ROM Verification Waveforms

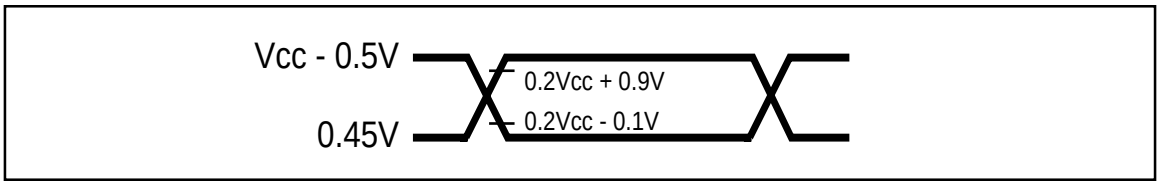

Figure 38. AC Test Point

Note:
1. AC inputs during testing are driven at VCC – 0.5V for logic “1” and 0.45V for logic “0”. Timing measurements are made at V_{IH} min for logic “1” and max for logic “0”.

ORDERING INFORMATION

COMMERCIAL TEMPERATURE:
0°C to +70°C

Speed	Order Part Number	Package
12 MHz	IS80C52-12PL	PLCC
	IS80C52-12PQ	PQFP
	IS80C52-12W	600-mil Plastic DIP
24 MHz	IS80C52-24PL	PLCC
	IS80C52-24PQ	PQFP
	IS80C52-24W	600-mil Plastic DIP
40 MHz	IS80C52-40PL	PLCC
	IS80C52-40PQ	PQFP
	IS80C52-40W	600-mil Plastic DIP
12 MHz	IS80C32-12PL	PLCC
	IS80C32-12PQ	PQFP
	IS80C32-12W	600-mil Plastic DIP
24 MHz	IS80C32-24PL	PLCC
	IS80C32-24PQ	PQFP
	IS80C32-24W	600-mil Plastic DIP
40 MHz	IS80C32-40PL	PLCC
	IS80C32-40PQ	PQFP
	IS80C32-40W	600-mil Plastic DIP

ORDERING INFORMATION

INDUSTRIAL TEMPERATURE:
–40°C to +85°C

Speed	Order Part Number	Package
12 MHz	IS80C52-12PLI	PLCC
	IS80C52-12PQI	PQFP
	IS80C52-12WI	600-mil Plastic DIP
24 MHz	IS80C52-24PLI	PLCC
	IS80C52-24PQI	PQFP
	IS80C52-24WI	600-mil Plastic DIP
40 MHz	IS80C52-40PLI	PLCC
	IS80C52-40PQI	PQFP
	IS80C52-40WI	600-mil Plastic DIP
12 MHz	IS80C32-12PLI	PLCC
	IS80C32-12PQI	PQFP
	IS80C32-12WI	600-mil Plastic DIP
24 MHz	IS80C32-24PLI	PLCC
	IS80C32-24PQI	PQFP
	IS80C32-24WI	600-mil Plastic DIP
40 MHz	IS80C32-40PLI	PLCC
	IS80C32-40PQI	PQFP
	IS80C32-40WI	600-mil Plastic DIP

Integrated Silicon Solution, Inc.
2231 Lawson Lane
Santa Clara, CA 95054
Fax: (408) 588-0806
Toll Free: 1-800-379-4774
http://www.issiusa.com