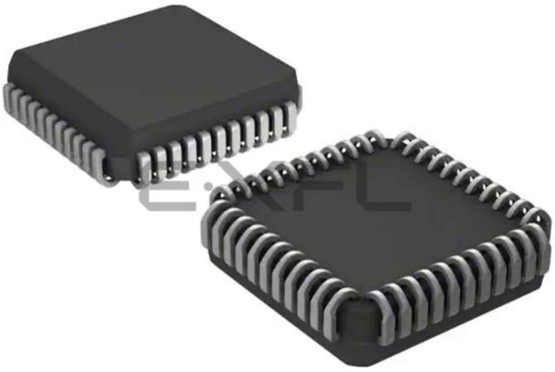
NXP USA Inc. - P89C51RC2FA/01,529 Datasheet



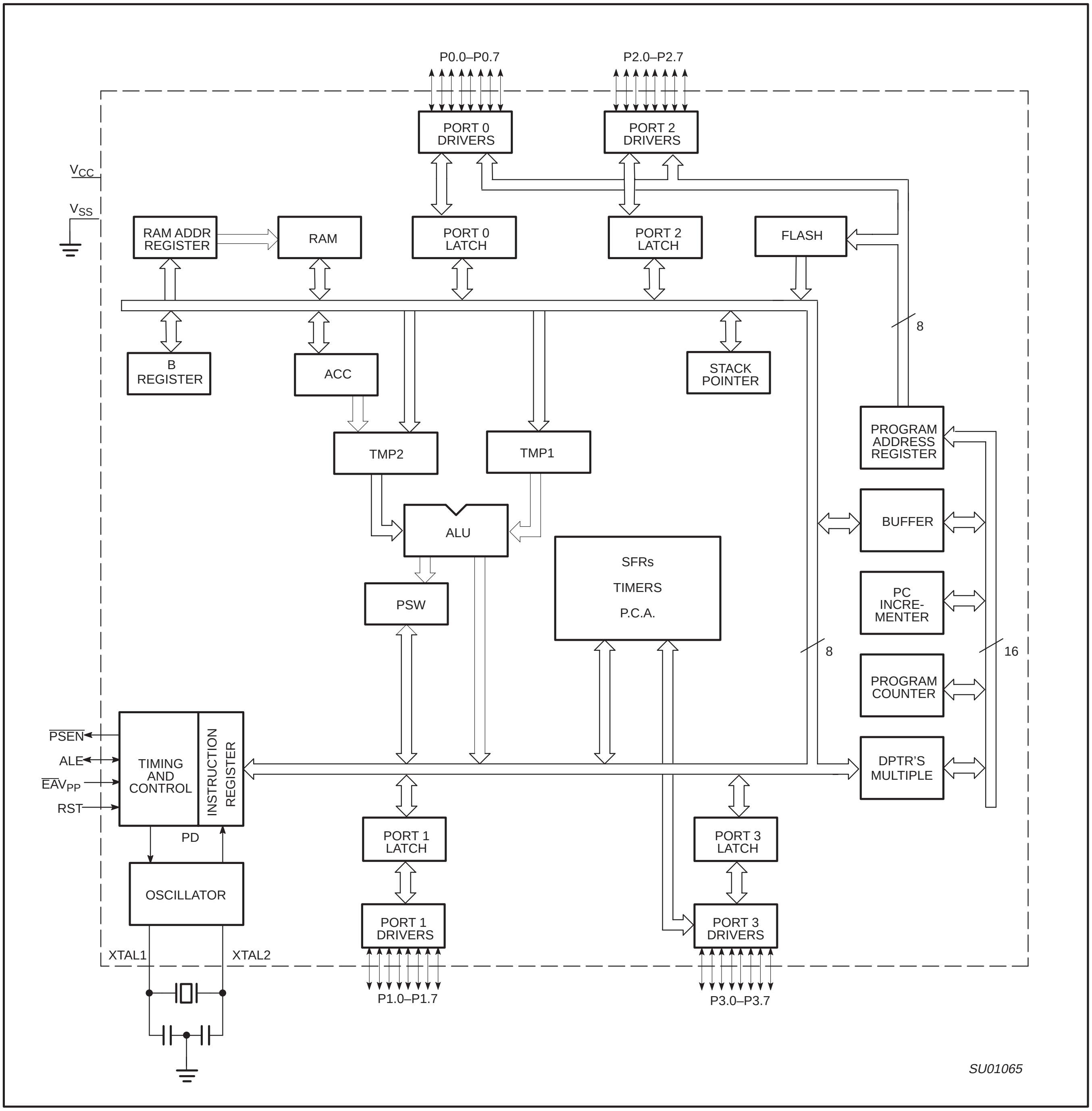
What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

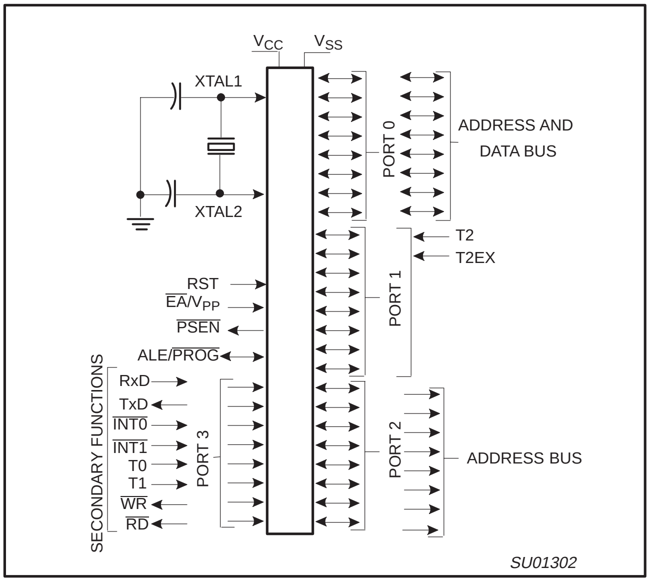
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	33MHz
Connectivity	UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	32
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p89c51rc2fa-01-529

BLOCK DIAGRAM – CPU ORIENTED



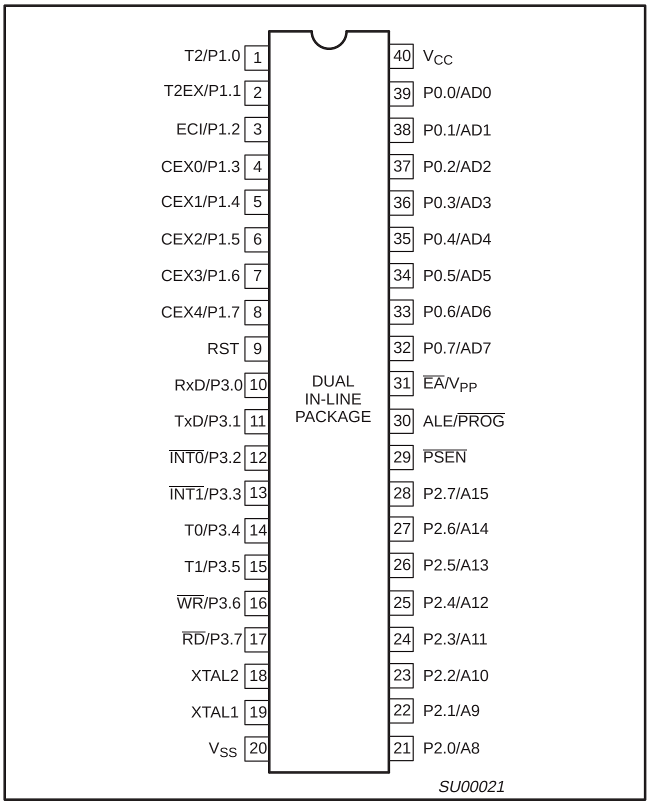
LOGIC SYMBOL



SU01302

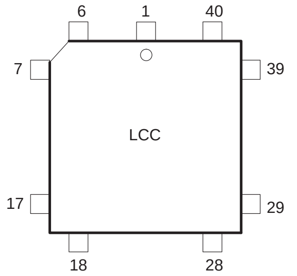
PINNING

Plastic Dual In-Line Package



SU00021

Plastic Leaded Chip Carrier

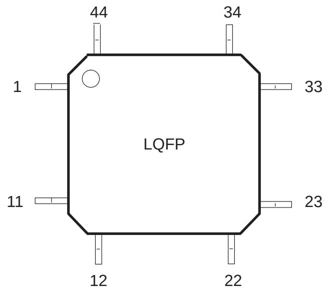


Pin	Function	Pin	Function	Pin	Function
1	NIC*	16	P3.4/T0	31	P2.7/A15
2	P1.0/T2	17	P3.5/T1	32	$\overline{\text{PSEN}}$
3	P1.1/T2EX	18	P3.6/$\overline{\text{WR}}$	33	ALE/$\overline{\text{PROG}}$
4	P1.2/ECI	19	P3.7/$\overline{\text{RD}}$	34	NIC*
5	P1.3/CEX0	20	XTAL2	35	$\overline{\text{EA}}/V_{PP}$
6	P1.4/CEX1	21	XTAL1	36	P0.7/AD7
7	P1.5/CEX2	22	V_{SS}	37	P0.6/AD6
8	P1.6/CEX3	23	NIC*	38	P0.5/AD5
9	P1.7/CEX4	24	P2.0/A8	39	P0.4/AD4
10	RST	25	P2.1/A9	40	P0.3/AD3
11	P3.0/RxD	26	P2.2/A10	41	P0.2/AD2
12	NIC*	27	P2.3/A11	42	P0.1/AD1
13	P3.1/TxD	28	P2.4/A12	43	P0.0/AD0
14	P3.2/$\overline{\text{INT0}}$	29	P2.5/A13	44	V_{CC}
15	P3.3/$\overline{\text{INT1}}$	30	P2.6/A14		

* NO INTERNAL CONNECTION

SU00023

Plastic Quad Flat Pack



Pin	Function	Pin	Function	Pin	Function
1	P1.5/CEX2	16	V_{SS}	31	P0.6/AD6
2	P1.6/CEX3	17	NIC*	32	P0.5/AD5
3	P1.7/CEX4	18	P2.0/A8	33	P0.4/AD4
4	RST	19	P2.1/A9	34	P0.3/AD3
5	P3.0/RxD	20	P2.2/A10	35	P0.2/AD2
6	NIC*	21	P2.3/A11	36	P0.1/AD1
7	P3.1/TxD	22	P2.4/A12	37	P0.0/AD0
8	P3.2/$\overline{\text{INT0}}$	23	P2.5/A13	38	V_{CC}
9	P3.3/$\overline{\text{INT1}}$	24	P2.6/A14	39	NIC*
10	P3.4/T0	25	P2.7/A15	40	P1.0/T2
11	P3.5/T1	26	$\overline{\text{PSEN}}$	41	P1.1/T2EX
12	P3.6/$\overline{\text{WR}}$	27	ALE/$\overline{\text{PROG}}$	42	P1.2/ECI
13	P3.7/$\overline{\text{RD}}$	28	NIC*	43	P1.3/CEX0
14	XTAL2	29	$\overline{\text{EA}}/V_{PP}$	44	P1.4/CEX1
15	XTAL1	30	P0.7/AD7		

* NO INTERNAL CONNECTION

SU01400

CLOCK CONTROL REGISTER (CKCON)

This device provides control of the 6-clock/12-clock mode by means of both an SFR bit (X2) and a Flash bit (FX2, located in the Security Block). The Flash clock control bit, FX2, when programmed (6-clock mode) supercedes the X2 bit (CKCON.0).

The CKCON register also provides individual control of the clock rates for the peripherals devices. When running in 6-clock mode each peripheral may be individually clocked from either fosc/6 or fosc/12. When in 12-clock mode, all peripheral devices will use fosc/12. The CKCON register is shown below.

CKCON Address = 8Fh Reset Value = x0000000B

Not Bit Addressable

7	6	5	4	3	2	1	0
–	WDX2	PCAX2	SIX2	T2X2	T1X2	T0X2	X2

BIT	**SYMBOL**	**FUNCTION**
CKCON.7	–	Reserved.
CKCON.6	WDX2	Watchdog clock; 0 = 6 clocks for each WDT clock, 1 = 12 clocks for each WDT clock
CKCON.5	PCAX2	PCA clock; 0 = 6 clocks for each PCA clock, 1 = 12 clocks for each PCA clock
CKCON.4	SIX2	UART clock; 0 = 6 clocks for each UART clock, 1 = 12 clocks for each UART clock
CKCON.3	T2X2	Timer2 clock; 0 = 6 clocks for each Timer2 clock, 1 = 12 clocks for each Timer2 clock
CKCON.2	T1X2	Timer1 clock; 0 = 6 clocks for each Timer1 clock, 1 = 12 clocks for each Timer1 clock
CKCON.1	T0X2	Timer0 clock; 0 = 6 clocks for each Timer0 clock, 1 = 12 clocks for each Timer0 clock
CKCON.0	X2	CPU clock; 1 = 6 clocks for each machine cycle, 0 = 12 clocks for each machine cycle

SU01607

Bits 1 through 6 only apply if 6 clocks per machine cycle is chosen (i.e.– Bit 0 = 1). If Bit 0 = 0 (12 clocks per machine cycle) then all peripherals will have 12 clocks per machine cycle as their clock source.

Also please note that the clock divider applies to the serial port for modes 0 & 2 (fixed baud rate modes). This is because modes 1 & 3 (variable baud rate modes) use either Timer 1 or Timer 2.

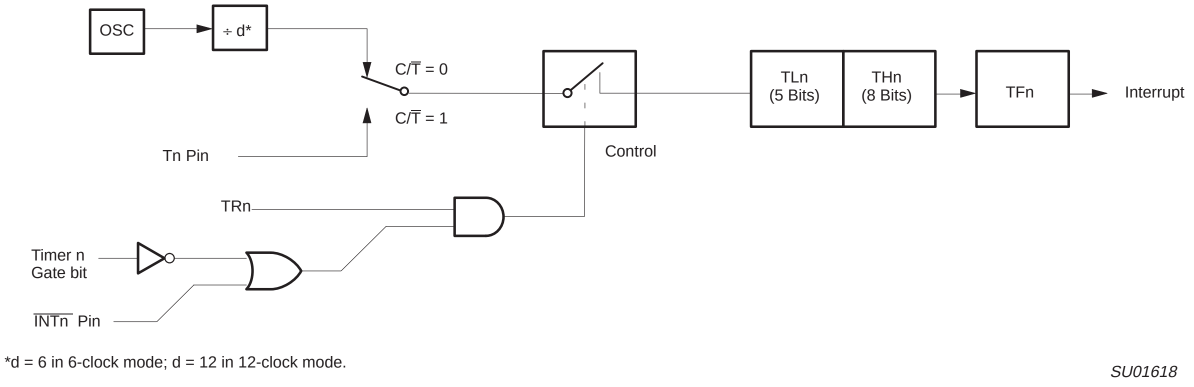
Below is the truth table for the peripheral input clock sources.

FX2 clock mode bit	X2	Peripheral clock mode bit (e.g., T0X2)	CPU MODE	Peripheral Clock Rate
erased	0	x	12-clock (default)	12-clock (default)
erased	1	0	6-clock	6-clock
erased	1	1	6-clock	12-clock
programmed	x	0	6-clock	6-clock
programmed	x	1	6-clock	12-clock

RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (12 oscillator periods in 6-clock mode, or 24 oscillator periods in 12-clock mode), while the oscillator is running. To ensure a good power-on reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-on, the voltage on V_{CC} and RST must come up at the same time for a proper start-up. Ports 1, 2, and 3 will asynchronously be driven to their reset condition when a voltage above V_{IH1} (min.) is applied to RST.

The value on the $\overline{EA}$ pin is latched when RST is deasserted and has no further effect.



*d = 6 in 6-clock mode; d = 12 in 12-clock mode.

Figure 2. Timer/Counter 0/1 Mode 0: 13-Bit Timer/Counter

TCON Address = 88H Reset Value = 00H

Bit Addressable

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

BIT	**SYMBOL**	**FUNCTION**
TCON.7	TF1	Timer 1 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine, or clearing the bit in software.
TCON.6	TR1	Timer 1 Run control bit. Set/cleared by software to turn Timer/Counter on/off.
TCON.5	TF0	Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine, or by clearing the bit in software.
TCON.4	TR0	Timer 0 Run control bit. Set/cleared by software to turn Timer/Counter on/off.
TCON.3	IE1	Interrupt 1 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TCON.2	IT1	Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TCON.1	IE0	Interrupt 0 Edge flag. Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
TCON.0	IT0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

SU01516

Figure 3. Timer/Counter 0/1 Control (TCON) Register

When Timer 2 is in the baud rate generator mode, one should not try to read or write TH2 and TL2. As a baud rate generator, Timer 2 is incremented every state time (osc/2) or asynchronously from pin T2; under these conditions, a read or write of TH2 or TL2 may not be accurate. The RCAP2 registers may be read, but should not be written to, because a write might overlap a reload and cause write and/or reload errors. The timer should be turned off (clear TR2) before accessing the Timer 2 or RCAP2 registers.

Table 4 shows commonly used baud rates and how they can be obtained from Timer 2.

Summary of Baud Rate Equations

Timer 2 is in baud rate generating mode. If Timer 2 is being clocked through pin T2 (P1.0) the baud rate is:

$$\text{Baud Rate} = \frac{\text{Timer 2 Overflow Rate}}{16}$$

If Timer 2 is being clocked internally, the baud rate is:

$$\text{Baud Rate} = \frac{f_{OSC}}{[\,n^* \times [65536 - (\text{RCAP2H}, \text{RCAP2L})]]}$$

* n = 16 in 6-clock mode
32 in 12-clock mode

Where f_{OSC}= Oscillator Frequency

To obtain the reload value for RCAP2H and RCAP2L, the above equation can be rewritten as:

$$\text{RCAP2H}, \text{RCAP2L} = 65536 - \left(\frac{f_{OSC}}{n^* \times \text{Baud Rate}}\right)$$

Timer/Counter 2 Set-up

Except for the baud rate generator mode, the values given for T2CON do not include the setting of the TR2 bit. Therefore, bit TR2 must be set, separately, to turn the timer on. see Table 5 for set-up of Timer 2 as a timer. Also see Table 6 for set-up of Timer 2 as a counter.

Table 5. Timer 2 as a Timer

MODE	T2CON	
	INTERNAL CONTROL (Note 1)	**EXTERNAL CONTROL (Note 2)**
16-bit Auto-Reload	00H	08H
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 6. Timer 2 as a Counter

MODE	TMOD	
	INTERNAL CONTROL (Note 1)	**EXTERNAL CONTROL (Note 2)**
16-bit	02H	0AH
Auto-Reload	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 generator mode.

FULL-DUPLEX ENHANCED UART

Standard UART operation

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive-buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the register. (However, if the first byte still hasn't been read by the time 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 4 modes:

Mode 0: Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted/received (LSB first). The baud rate is fixed at 1/12 the oscillator frequency in 12-clock mode or 1/6 the oscillator frequency in 6-clock mode.

Mode 1: 10 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 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.

Mode 2: 11 bits are transmitted (through TxD) or received (through RxD): start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). On Transmit, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. On receive, the 9th 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 in 12-clock mode or 1/16 or 1/32 the oscillator frequency in 6-clock mode.

Mode 3: 11 bits are transmitted (through TxD) or received (through RxD): a start bit (0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (1). In fact, Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable.

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, 9 data bits are received. The 9th one 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 will be activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON. A way to use this feature in multiprocessor systems is as follows:

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 0 in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that will be coming. The slaves that weren't being addressed leave their SM2s set and go on about their business, ignoring the coming data bytes.

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

Serial Port Control Register

The serial port control and status register is the Special Function Register SCON, shown in Figure 12. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

Baud Rates

The baud rate in Mode 0 is fixed: Mode 0 Baud Rate = Oscillator Frequency / 12 (12-clock mode) or / 6 (6-clock mode). The baud rate in Mode 2 depends on the value of bit SMOD in Special Function Register PCON. If SMOD = 0 (which is the value on reset), and the port pins in 12-clock mode, the baud rate is 1/64 the oscillator frequency. If SMOD = 1, the baud rate is 1/32 the oscillator frequency. In 6-clock mode, the baud rate is 1/32 or 1/16 the oscillator frequency, respectively.

Mode 2 Baud Rate =

$$\frac{2^{SMOD}}{n} \times (\text{Oscillator Frequency})$$

Where:

n = 64 in 12-clock mode, 32 in 6-clock mode

The baud rates in Modes 1 and 3 are determined by the Timer 1 or Timer 2 overflow rate.

Using Timer 1 to Generate Baud Rates

When Timer 1 is used as the baud rate generator (T2CON.RCLK = 0, T2CON.TCLK = 0), the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD as follows:

Mode 1, 3 Baud Rate =

$$\frac{2^{SMOD}}{n} \times (\text{Timer 1 Overflow Rate})$$

Where:

n = 32 in 12-clock mode, 16 in 6-clock mode

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

Mode 1, 3 Baud Rate =

$$\frac{2^{SMOD}}{n} \times \frac{\text{Oscillator Frequency}}{12 \times [256-(\text{TH1})]}$$

Where:

n = 32 in 12-clock mode, 16 in 6-clock mode

One can achieve very low baud rates with Timer 1 by leaving the Timer 1 interrupt enabled, and 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. Figure 13 lists various commonly used baud rates and how they can be obtained from Timer 1.

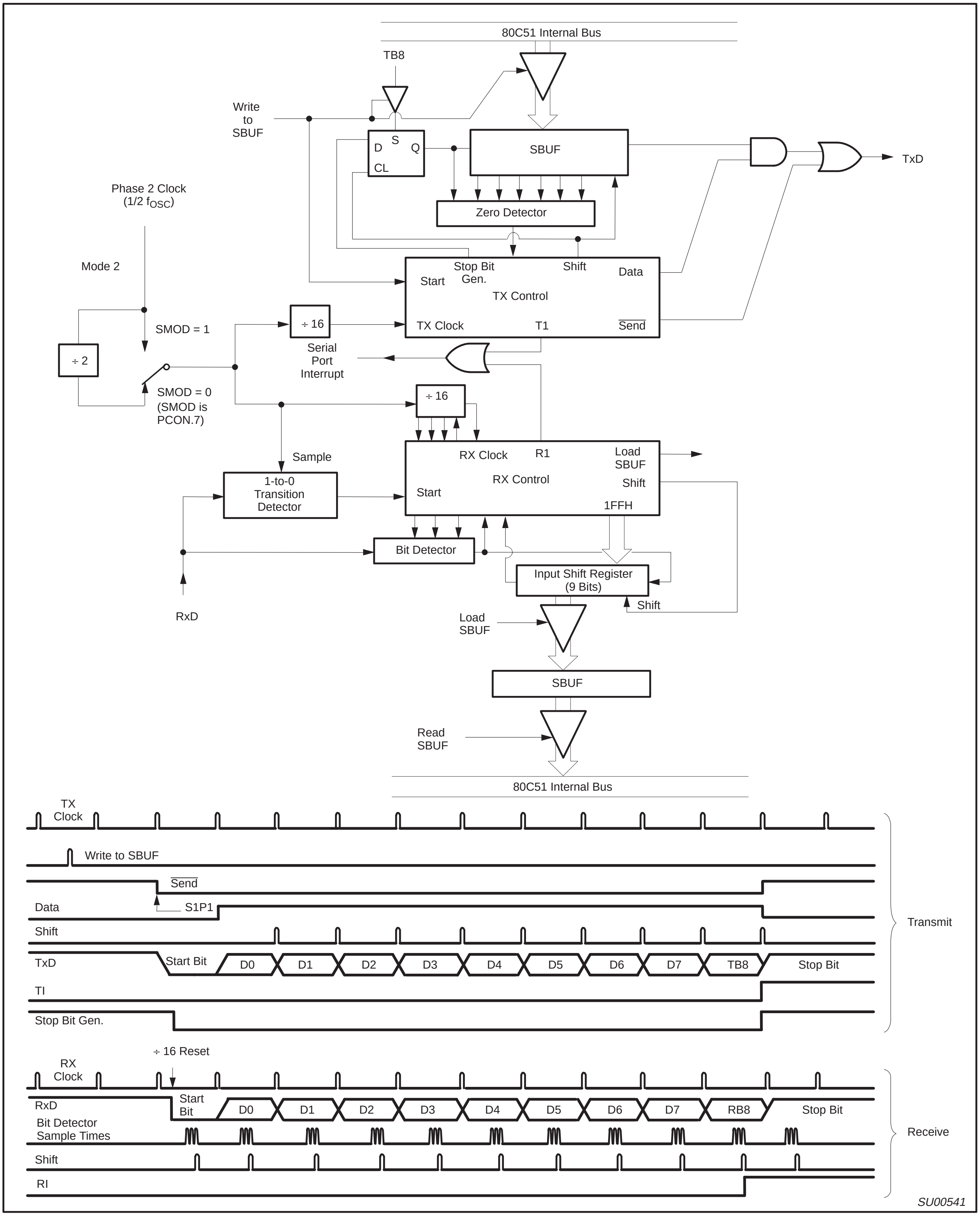


Figure 16. Serial Port Mode 2

Enhanced UART

In addition to the standard operation the UART can perform framing error detect by looking for missing stop bits, and automatic address recognition. The UART also fully supports multiprocessor communication as does the standard 80C51 UART.

When used for framing error detect the UART looks for missing stop bits in the communication. A missing bit will set the FE bit in the SCON register. The FE bit shares the SCON.7 bit with SM0 and the function of SCON.7 is determined by PCON.6 (SMOD0) (see Figure 18). If SMOD0 is set then SCON.7 functions as FE. SCON.7 functions as SM0 when SMOD0 is cleared. When used as FE SCON.7 can only be cleared by software. Refer to Figure 19.

Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the “Given” address or the “Broadcast” address. The 9-bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data. Automatic address recognition is shown in Figure 20.

The 8 bit mode is called Mode 1. In this mode the RI flag will be set if SM2 is enabled and the information received has a valid stop bit following the 8 address bits and the information is either a Given or Broadcast address.

Mode 0 is the Shift Register mode and SM2 is ignored.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave’s address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to b used and which bits are “don’t care”. The SADEN mask can be logically ANDed with the SADDR to create the “Given” address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

Slave 0	SADDR	=	1100 0000
	SADEN	=	1111 1101
	Given	=	1100 00X0

Slave 1	SADDR	=	1100 0000
	SADEN	=	1111 1110
	Given	=	1100 000X

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Slave 0	SADDR	=	1100 0000
	SADEN	=	1111 1001
	Given	=	1100 0XX0
Slave 1	SADDR	=	1110 0000
	SADEN	=	1111 1010
	Given	=	1110 0X0X
Slave 2	SADDR	=	1110 0000
	SADEN	=	1111 1100
	Given	=	1110 00XX

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2.

The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are trended as don’t-cares. In most cases, interpreting the don’t-cares as ones, the broadcast address will be FF hexadecimal.

Upon reset SADDR (SFR address 0A9H) and SADEN (SFR address 0B9H) are leaded with 0s. This produces a given address of all “don’t cares” as well as a Broadcast address of all “don’t cares”. This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard 80C51 type UART drivers which do not make use of this feature.

	7	6	5	4	3	2	1	0
IP (0B8H)	–	PPC	PT2	PS	PT1	PX1	PT0	PX0

Priority Bit = 1 assigns high priority
Priority Bit = 0 assigns low priority

BIT	SYMBOL	FUNCTION
IP.7	–	–
IP.6	PPC	PCA interrupt priority bit
IP.5	PT2	Timer 2 interrupt priority bit.
IP.4	PS	Serial Port interrupt priority bit.
IP.3	PT1	Timer 1 interrupt priority bit.
IP.2	PX1	External interrupt 1 priority bit.
IP.1	PT0	Timer 0 interrupt priority bit.
IP.0	PX0	External interrupt 0 priority bit.

SU01291

Figure 22. IP Registers

	7	6	5	4	3	2	1	0
IPH (B7H)	–	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H

Priority Bit = 1 assigns higher priority
Priority Bit = 0 assigns lower priority

BIT	SYMBOL	FUNCTION
IPH.7	–	–
IPH.6	PPCH	PCA interrupt priority bit
IPH.5	PT2H	Timer 2 interrupt priority bit high.
IPH.4	PSH	Serial Port interrupt priority bit high.
IPH.3	PT1H	Timer 1 interrupt priority bit high.
IPH.2	PX1H	External interrupt 1 priority bit high.
IPH.1	PT0H	Timer 0 interrupt priority bit high.
IPH.0	PX0H	External interrupt 0 priority bit high.

SU01292

Figure 23. IPH Registers

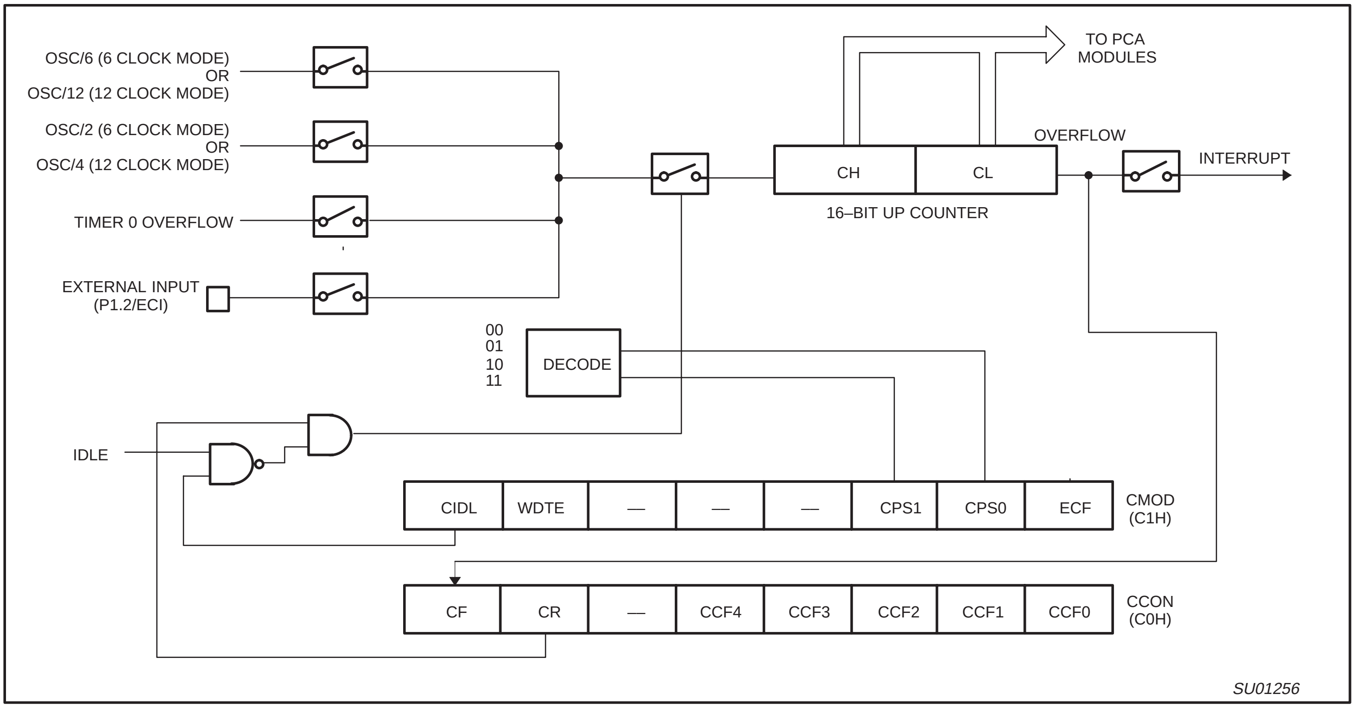


Figure 26. PCA Timer/Counter

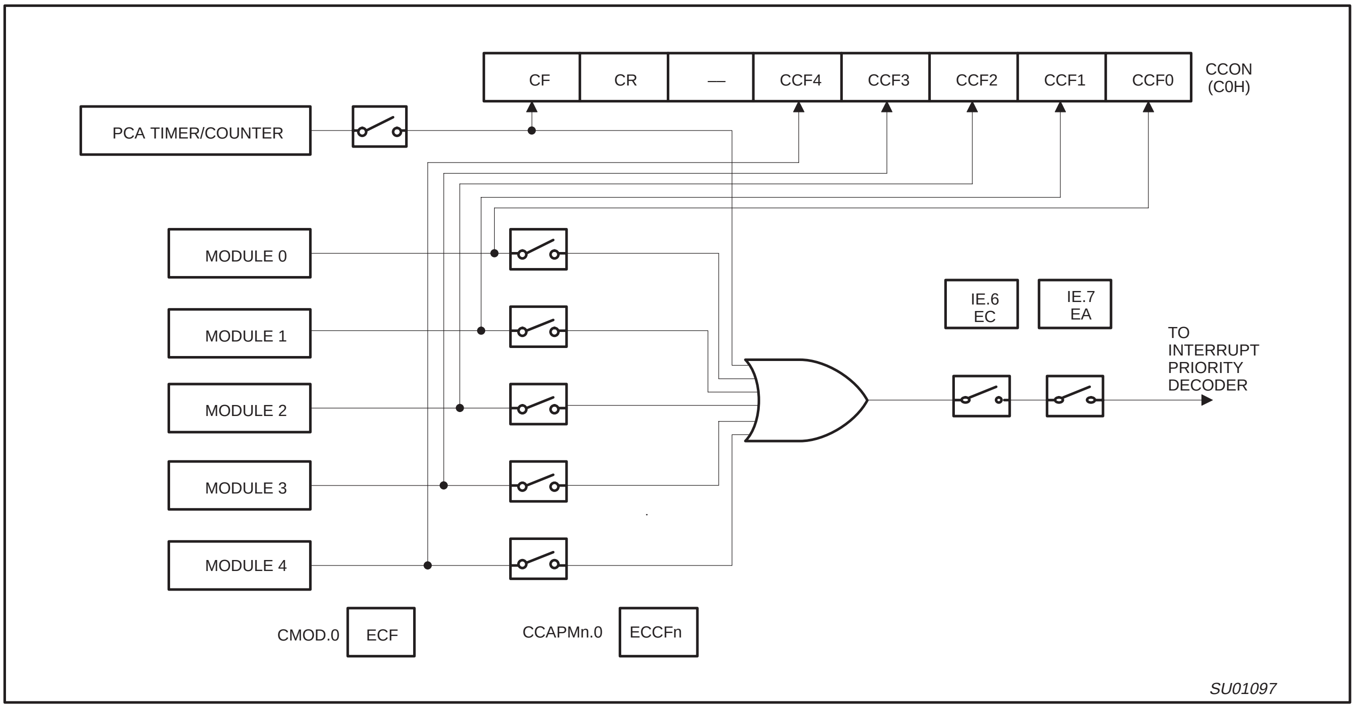


Figure 27. PCA Interrupt System

CMOD Address = D9H Reset Value = 00XX X000B

	CIDL	**WDTE**	**–**	**–**	**–**	**CPS1**	**CPS0**	**ECF**
Bit:	7	6	5	4	3	2	1	0

Symbol	Function
CIDL	Counter Idle control: CIDL = 0 programs the PCA Counter to continue functioning during idle Mode. CIDL = 1 programs it to be gated off during idle.
WDTE	Watchdog Timer Enable: WDTE = 0 disables Watchdog Timer function on PCA Module 4. WDTE = 1 enables it.
–	Not implemented, reserved for future use.*
CPS1	PCA Count Pulse Select bit 1.
CPS0	PCA Count Pulse Select bit 0.

CPS1	CPS0	Selected PCA Input**	
0	0	0	Internal clock, f_{OSC}/6 in 6-clock mode (f_{OSC}/12 in 12-clock mode)
0	1	1	Internal clock, f_{OSC}/2 in 6-clock mode (f_{OSC}/4 in 12-clock mode)
1	0	2	Timer 0 overflow
1	1	3	External clock at ECI/P1.2 pin (max. rate = f_{OSC}/4 in 6-clock mode, f_{OCS}/8 in 12-clock mode)

Symbol	Function
ECF	PCA Enable Counter Overflow interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function of CF.

NOTE:

* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

** f_{OSC} = oscillator frequency

SU01318

Figure 28. CMOD: PCA Counter Mode Register

CCON Address = D8H Reset Value = 00X0 0000B

Bit Addressable

	CF	**CR**	**–**	**CCF4**	**CCF3**	**CCF2**	**CCF1**	**CCF0**
Bit:	7	6	5	4	3	2	1	0

Symbol	Function
CF	PCA Counter Overflow flag. Set by hardware when the counter rolls over. CF flags an interrupt if bit ECF in CMOD is set. CF may be set by either hardware or software but can only be cleared by software.
CR	PCA Counter Run control bit. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off.
–	Not implemented, reserved for future use*.
CCF4	PCA Module 4 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software.
CCF3	PCA Module 3 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software.
CCF2	PCA Module 2 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software.
CCF1	PCA Module 1 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software.
CCF0	PCA Module 0 interrupt flag. Set by hardware when a match or capture occurs. Must be cleared by software.

NOTE:

* User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU01319

Figure 29. CCON: PCA Counter Control Register

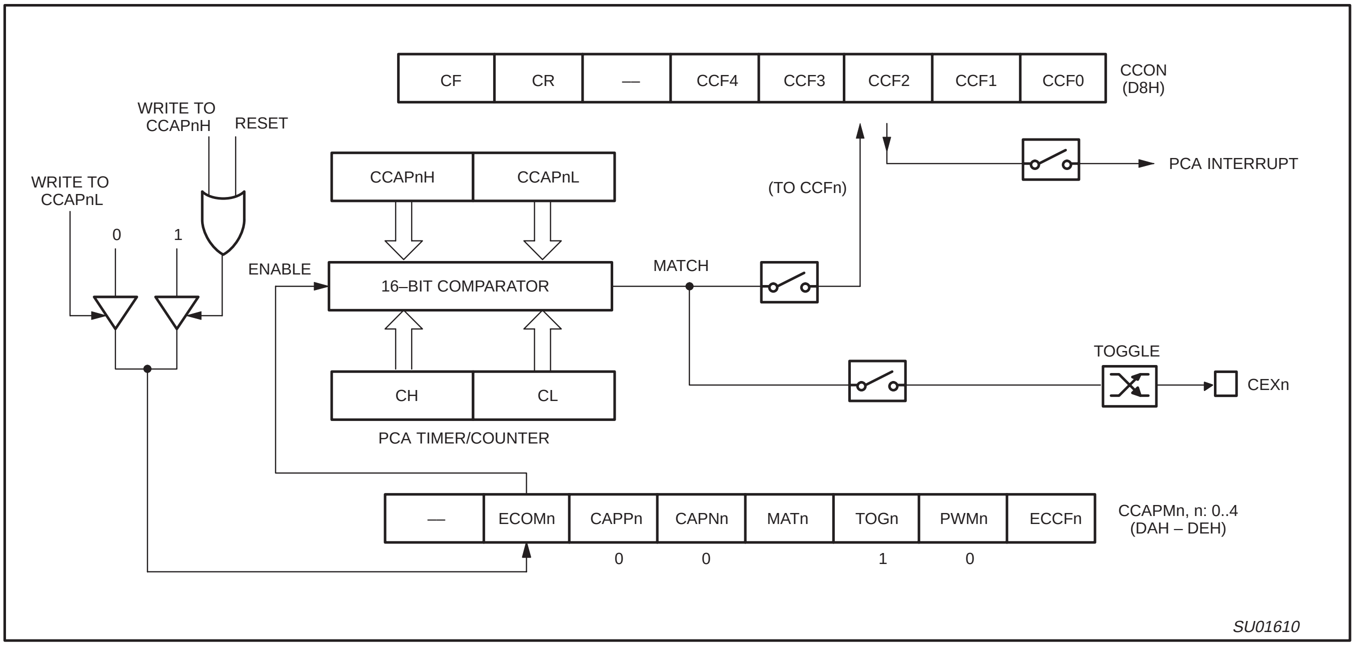


Figure 34. PCA High Speed Output Mode

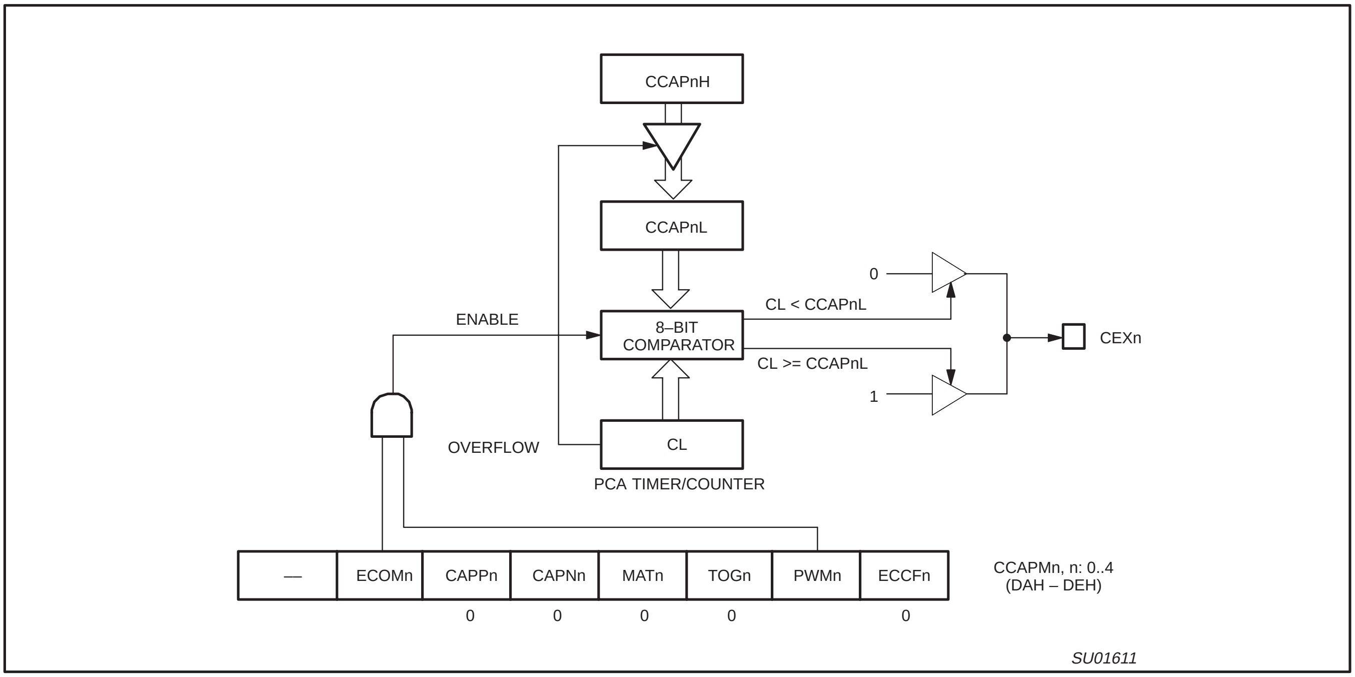


Figure 35. PCA PWM Mode

Expanded Data RAM Addressing

The P89C51RA2/RB2/RC2/RD2xx has internal data memory that is mapped into four separate segments: the lower 128 bytes of RAM, upper 128 bytes of RAM, 128 bytes Special Function Register (SFR), and 256 bytes expanded RAM (ERAM) (768 bytes for the RD2xx).

The four segments are:

1. The Lower 128 bytes of RAM (addresses 00H to 7FH) are directly and indirectly addressable.
2. The Upper 128 bytes of RAM (addresses 80H to FFH) are indirectly addressable only.
3. The Special Function Registers, SFRs, (addresses 80H to FFH) are directly addressable only.
4. The 256/768-bytes expanded RAM (ERAM, 00H – 1FFH/2FFH) are indirectly accessed by move external instruction, MOVX, and with the EXTRAM bit cleared, see Figure 38.

The Lower 128 bytes can be accessed by either direct or indirect addressing. The Upper 128 bytes can be accessed by indirect addressing only. The Upper 128 bytes occupy the same address space as the SFR. That means they have the same address, but are physically separate from SFR space.

When an instruction accesses an internal location above address 7FH, the CPU knows whether the access is to the upper 128 bytes of data RAM or to SFR space by the addressing mode used in the instruction. Instructions that use direct addressing access SFR space. For example:

```
MOV 0A0H,#data
```

accesses the SFR at location 0A0H (which is P2). Instructions that use indirect addressing access the Upper 128 bytes of data RAM.

For example:

```
MOV @R0,acc
```

where R0 contains 0A0H, accesses the data byte at address 0A0H, rather than P2 (whose address is 0A0H).

The ERAM can be accessed by indirect addressing, with EXTRAM bit cleared and MOVX instructions. This part of memory is physically located on-chip, logically occupies the first 256/768 bytes of external data memory in the P89C51RA2/RB2/RC2/89C51RD2.

With EXTRAM = 0, the ERAM is indirectly addressed, using the MOVX instruction in combination with any of the registers R0, R1 of the selected bank or DPTR. An access to ERAM will not affect ports P0, P3.6 (WR#) and P3.7 (RD#). P2 SFR is output during external addressing. For example, with EXTRAM = 0,

```
MOVX @R0,acc
```

where R0 contains 0A0H, accesses the ERAM at address 0A0H rather than external memory. An access to external data memory locations higher than the ERAM will be performed with the MOVX DPTR instructions in the same way as in the standard 80C51, so with P0 and P2 as data/address bus, and P3.6 and P3.7 as write and read timing signals. Refer to Figure 39.

With EXTRAM = 1, MOVX @Ri and MOVX @DPTR will be similar to the standard 80C51. MOVX @ Ri will provide an 8-bit address multiplexed with data on Port 0 and any output port pins can be used to output higher order address bits. This is to provide the external paging capability. MOVX @DPTR will generate a 16-bit address. Port 2 outputs the high-order eight address bits (the contents of DPH) while Port 0 multiplexes the low-order eight address bits (DPL) with data. MOVX @Ri and MOVX @DPTR will generate either read or write signals on P3.6 ($\overline{WR}$) and P3.7 ($\overline{RD}$).

The stack pointer (SP) may be located anywhere in the 256 bytes RAM (lower and upper RAM) internal data memory. The stack may not be located in the ERAM.

AUXR Address = 8EH Reset Value = xxxx xx00B

Not Bit Addressable

	—	—	—	—	—	—	**EXTRAM**	**AO**
Bit:	7	6	5	4	3	2	1	0

Symbol **Function**

AO Disable/Enable ALE

AO	**Operating Mode**
0	ALE is emitted at a constant rate of $^{1}/_{6}$ the oscillator frequency (12-clock mode; $^{1}/_{3}$ f_{OSC} in 6-clock mode).
1	ALE is active only during off-chip memory access.

EXTRAM Internal/External RAM access using MOVX @Ri/@DPTR

EXTRAM	**Operating Mode**
0	Internal ERAM access using MOVX @Ri/@DPTR
1	External data memory access.

— Not implemented, reserved for future use*.

NOTE:
*User software should not write 1s to reserved bits. These bits may be used in future 8051 family products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1. The value read from a reserved bit is indeterminate.

SU01613

Figure 38. AUXR: Auxiliary Register

FLASH EPROM MEMORY

GENERAL DESCRIPTION

The P89C51RA2/RB2/RC2/RD2xx Flash memory augments EPROM functionality with in-circuit electrical erasure and programming. The Flash can be read and written as bytes. The Chip Erase operation will erase the entire program memory. The Block Erase function can erase any Flash block. In-system programming and standard parallel programming are both available. On-chip erase and write timing generation contribute to a user friendly programming interface.

The P89C51RA2/RB2/RC2/RD2xx Flash reliably stores memory contents even after 10,000 erase and program cycles. The cell is designed to optimize the erase and programming mechanisms. In addition, the combination of advanced tunnel oxide processing and low internal electric fields for erase and programming operations produces reliable cycling. The P89C51RA2/RB2/RC2/RD2xx uses a +5 V V_{PP} supply to perform the Program/Erase algorithms.

FEATURES – IN-SYSTEM PROGRAMMING (ISP) AND IN-APPLICATION PROGRAMMING (IAP)

- Flash EPROM internal program memory with Block Erase.
- Internal 1-kbyte fixed BootROM, containing low-level in-system programming routines and a default serial loader. User program can call these routines to perform In-Application Programming (IAP). The BootROM can be turned off to provide access to the full 64-kbyte Flash memory.
- Boot Vector allows user provided Flash loader code to reside anywhere in the Flash memory space. This configuration provides flexibility to the user.
- Default loader in BootROM allows programming via the serial port without the need for a user provided loader.
- Up to 64-kbyte external program memory if the internal program memory is disabled ($\overline{EA}$ = 0).
- Programming and erase voltage +5 V (+12 V tolerant).
- Read/Programming/Erase using ISP/IAP:
  - Byte Programming (8 µs).
  - Typical quick erase times:

    Block Erase (4 kbyte) in 3 seconds.

    Full Chip Erase:
    – RD2xx (64K) in 11 seconds
    – RC2 (32K) in 7 seconds
    – RB2 (16K) in 5 seconds
    – RA2 (4K) in 4 seconds
- Parallel programming with 87C51 compatible hardware interface to programmer.
- In-system programming (ISP).
- In-application programming (IAP).
- Programmable security for the code in the Flash.
- 10,000 minimum erase/program cycles for each byte.
- 10-year minimum data retention.

FLASH PROGRAMMING AND ERASURE

In general, there are three methods of erasing or programming of the Flash memory that may be used. First, the Flash may be programmed or erased in the end-user application by calling low-level routines through entry point in the BootROM. The end-user application, though, must be executing code from a different block than the block that is being erased or programmed. Second, the on-chip ISP boot loader may be invoked. This ISP boot loader will, in turn, call low-level routines through the common entry point in the BootROM that can be used by end-user applications. Third, the Flash may be programmed or erased using parallel method by using a commercially available EPROM programmer. The parallel programming method used by these devices is similar to that used by EPROM 87C51, but it is not identical, and the commercially available programmer will need to have support for these devices.

FLASH MEMORY SPACES

Flash User Code Memory Organization

The P89C51RA2/RB2/RC2/RD2xx contains 8KB/16KB/32KB/64KB Flash user code program memory organized into 4-kbyte blocks. ISP and IAP BootROM routines will support the new 4-kbyte block sizes through additional block number assignments while maintaining compatibility with previous 8-kbyte and 16-kbyte block assignments. This memory space is programmable via IAP, ISP, and parallel modes.

Status Byte/Boot Vector Block

This device includes a 4-kbyte block which contains the Status Byte and Boot Vector (Status Byte Block) . The Status Byte and Boot Vector are programmable via IAP, ISP, and parallel modes. Note that erasing of either the Status Byte and Boot Vector will erase the entire contents of this block. Thus the Status Byte and Boot Vector are erased together but are programmable separately.

Security & User Configuration Block

This device includes a 4-kbyte block (Security Block) which contains the Security Bits, the 6-clock/12-clock Flash-based clock mode bit FX2, and 4095 user programmable bytes. This block is programmable via IAP, ISP, and parallel modes. Security bits will prevent, as required, parallel programmers from reading or writing, however, IAP or ISP inhibitions will be software controlled. This block may only be erased using full-chip erase functions in ISP, IAP, or parallel mode. This security feature protects against software piracy and prevents the contents of the Flash from being read. The Security bits are located in the Flash. There are three programmable security bits that will provide different levels of protection for the on-chip code and data (See Table 11). The 4095 user programmable bytes are not part of user code memory are intended to be programmed or read through IAP, ISP, or parallel programmer functions.

The 6-clock/12-clock Flash-based clock mode bit FX2 will be latched at power-on. This allows the bit to be changed via IAP or ISP and delay taking effect until the next reset. This avoids changing baud rates during ISP operations.

Boot ROM

When the microcontroller programs its Flash memory, all of the low level details are handled by code that is contained in a 1-kbyte

RECORD TYPE	COMMAND/DATA FUNCTION
03 (Cont.)	Subfunction Code = 07 (Full Chip Erase) Erases all blocks, security bits, and sets status byte and boot vector to default values ff = 07 ss = don’t care dd = don’t care Example: :0100000307F5 full chip erase Subfunction Code = 0C (Erase 4K Blocks) ff = 0C ss = block code as shown below: Block 0 , 0k~4k , 00H Block 1 , 4k~8k , 10H Block 2 , 8k~12k , 20H (only available on RD2 / RC2 / RB2) Block 3 , 12k~16k , 30H (only available on RD2 / RC2 / RB2) Block 4 , 16k~20k , 40H (only available on RD2 / RC2) Block 5 , 20k~24k , 50H (only available on RD2 / RC2) Block 6 , 24k~28k , 60H (only available on RD2 / RC2) Block 7 , 28k~32k , 70H (only available on RD2 / RC2) Block 8 , 32k~36k , 80H (only available on RD2) Block 9 , 36k~40k , 90H (only available on RD2) Block 10, 40k~44k , A0H (only available on RD2) Block 11, 44k~48k , B0H (only available on RD2) Block 12, 48k~52k , C0H (only available on RD2) Block 13, 52k~56k , D0H (only available on RD2) Block 14, 56k~60k , E0H (only available on RD2) Block 15, 60k~64k , F0H (only available on RD2) Example: :020000030C20CF (Erase 4k block #2)
04	Display Device Data or Blank Check – Record type 04 causes the contents of the entire Flash array to be sent out the serial port in a formatted display. This display consists of an address and the contents of 16 bytes starting with that address. No display of the device contents will occur if security bit 2 has been programmed. Data to the serial port is initiated by the reception of any character and terminated by the reception of any character. General Format of Function 04 :05xxxx04sssseeeeffcc Where: 05 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 04 = “Display Device Data or Blank Check” function code ssss = starting address eeee = ending address ff = subfunction 00 = display data 01 = blank check 02 = display data in data block (valid addresses: 0001~0FFFH) cc = checksum Example 1: :0500000440004FFF0069 display 4000–4FFF Example 2: :0500000400000FFF02E7 display data in data block (the data at address 0000 is invalid)

RECORD TYPE	COMMAND/DATA FUNCTION
05	Miscellaneous Read Functions (Selection) General Format of Function 05 :02xxxx05ffsscc Where: 02 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 05 = “Miscellaneous Read” function code ffss = subfunction and selection code 0000 = read signature byte – manufacturer id (15H) 0001 = read signature byte – device id # 1 (C2H) 0002 = read signature byte – device id # 2 0003 = read FX2 bit 0080 = read ROM Code Revision 0700 = read security bits 0701 = read status byte 0702 = read boot vector cc = checksum Example 1: :020000050001F8 read signature byte – device id # 1 Example 2: :020000050003F6 read FX2 bit (bit7=0 represent 12–clock mode, bit7=1 represent 6–clock mode) Example 3: :02000005008079 read ROM Code Revision (0A: Rev. A, 0B:Rev. B)
06	Direct Load of Baud Rate General Format of Function 06 :02xxxx06hhllcc Where: 02 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 06 = ”Direct Load of Baud Rate” function code hh = high byte of Timer 2 ll = low byte of Timer 2 cc = checksum Example: :02000006F500F3
07	Program Data in Data Block :nnaaaa07dd....ddcc Where: nn = number of bytes (hex) in record aaaa = memory address of first byte in record (the valid address:0001~0FFFH) dd....dd = data bytes cc = checksum Example: :10008007AF5F67F0602703E0322CFA92007780C3F6

DC ELECTRICAL CHARACTERISTICS

T_{amb} = 0 °C to +70 °C or –40 °C to +85 °C; V_{CC} = 5 V ± 10%; V_{SS} = 0 V

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS			UNIT
			MIN	TYP[1]	MAX	
V_{IL}	Input low voltage	4.5 V < V_{CC} < 5.5 V	–0.5		$0.2V_{CC}–0.1$	V
V_{IH}	Input high voltage (ports 0, 1, 2, 3, $\overline{\text{EA}}$)		$0.2V_{CC}+0.9$		$V_{CC}+0.5$	V
V_{IH1}	Input high voltage, XTAL1, RST		$0.7V_{CC}$		$V_{CC}+0.5$	V
V_{OL}	Output low voltage, ports 1, 2, 3[8]	V_{CC} = 4.5 V I_{OL} = 1.6 mA[2]			0.4	V
V_{OL1}	Output low voltage, port 0, ALE, $\overline{\text{PSEN}}$ [7, 8]	V_{CC} = 4.5 V I_{OL} = 3.2 mA[2]			0.45	V
V_{OH}	Output high voltage, ports 1, 2, 3 [3]	V_{CC} = 4.5 V I_{OH} = –30 µA	V_{CC} – 0.7			V
V_{OH1}	Output high voltage (port 0 in external bus mode), ALE[9], $\overline{\text{PSEN}}$[3]	V_{CC} = 4.5 V I_{OH} = –3.2 mA	V_{CC} – 0.7			V
I_{IL}	Logical 0 input current, ports 1, 2, 3	V_{IN} = 0.4 V	–1		–75	µA
I_{TL}	Logical 1-to-0 transition current, ports 1, 2, 3[6]	V_{IN} = 2.0 V See Note 4			–650	µA
I_{LI}	Input leakage current, port 0	0.45 < V_{IN} < V_{CC} – 0.3			±10	µA
I_{CC}	Power supply current (see Figure 49):	See Note 5				
	Active mode (see Note 5)					
	Idle mode (see Note 5)					
	Power-down mode or clock stopped (see Figure 55 for conditions)	T_{amb} = 0 °C to 70 °C		< 30	100	µA
		T_{amb} = –40 °C to +85 °C		< 40	125	µA
	Programming and erase mode	f_{osc} = 20 MHz		60		mA
R_{RST}	Internal reset pull-down resistor		40		225	kΩ
C_{IO}	Pin capacitance[10] (except $\overline{\text{EA}}$)				15	pF

NOTES:

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5 V.
2. Capacitive loading on ports 0 and 2 may cause spurious noise to be superimposed on the V_{OL}s of ALE and ports 1 and 3. The noise is due to external bus capacitance discharging into the port 0 and port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 100 pF), the noise pulse on the ALE pin may exceed 0.8 V. In such cases, it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input. I_{OL} can exceed these conditions provided that no single output sinks more than 5 mA and no more than two outputs exceed the test conditions.
3. Capacitive loading on ports 0 and 2 may cause the V_{OH} on ALE and $\overline{\text{PSEN}}$ to momentarily fall below the V_{CC}–0.7 specification when the address bits are stabilizing.
4. Pins of ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when V_{IN} is approximately 2 V.
5. See Figures 52 through 55 for I_{CC} test conditions and Figure 49 for I_{CC} vs Freq.
   Active mode: $I_{CC(MAX)}$ = (10.5 + 0.9 × FREQ.[MHz])mA in 12-clock mode
   Idle mode: $I_{CC(MAX)}$ = (2.5 + 0.33 × FREQ.[MHz])mA in 12-clock mode
6. This value applies to T_{amb} = 0 °C to +70 °C.
7. Load capacitance for port 0, ALE, and $\overline{\text{PSEN}}$ = 100 pF, load capacitance for all other outputs = 80 pF.
8. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 15 mA (*NOTE: This is 85 °C specification.)
   Maximum I_{OL} per 8-bit port: 26 mA
   Maximum total I_{OL} for all outputs: 71 mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
9. ALE is tested to V_{OH1}, except when ALE is off then V_{OH} is the voltage specification.
10. Pin capacitance is characterized but not tested. Pin capacitance is less than 25 pF. Pin capacitance of ceramic package is less than 15 pF (except $\overline{\text{EA}}$ is 25 pF).

EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always ‘t’ (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{\text{PSEN}}$
Q – Output data
R – $\overline{\text{RD}}$ signal
t – Time
V – Valid
W – $\overline{\text{WR}}$ signal
X – No longer a valid logic level
Z – Float

Examples: t_{AVLL} = Time for address valid to ALE low.
t_{LLPL} = Time for ALE low to $\overline{\text{PSEN}}$ low.

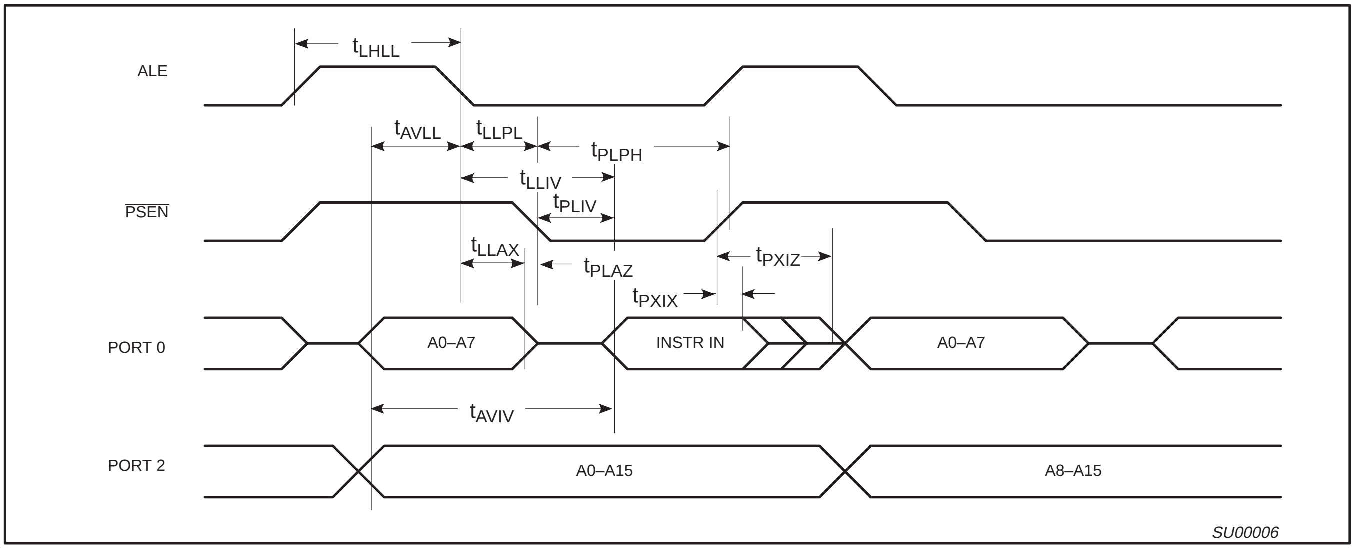


Figure 42. External Program Memory Read Cycle

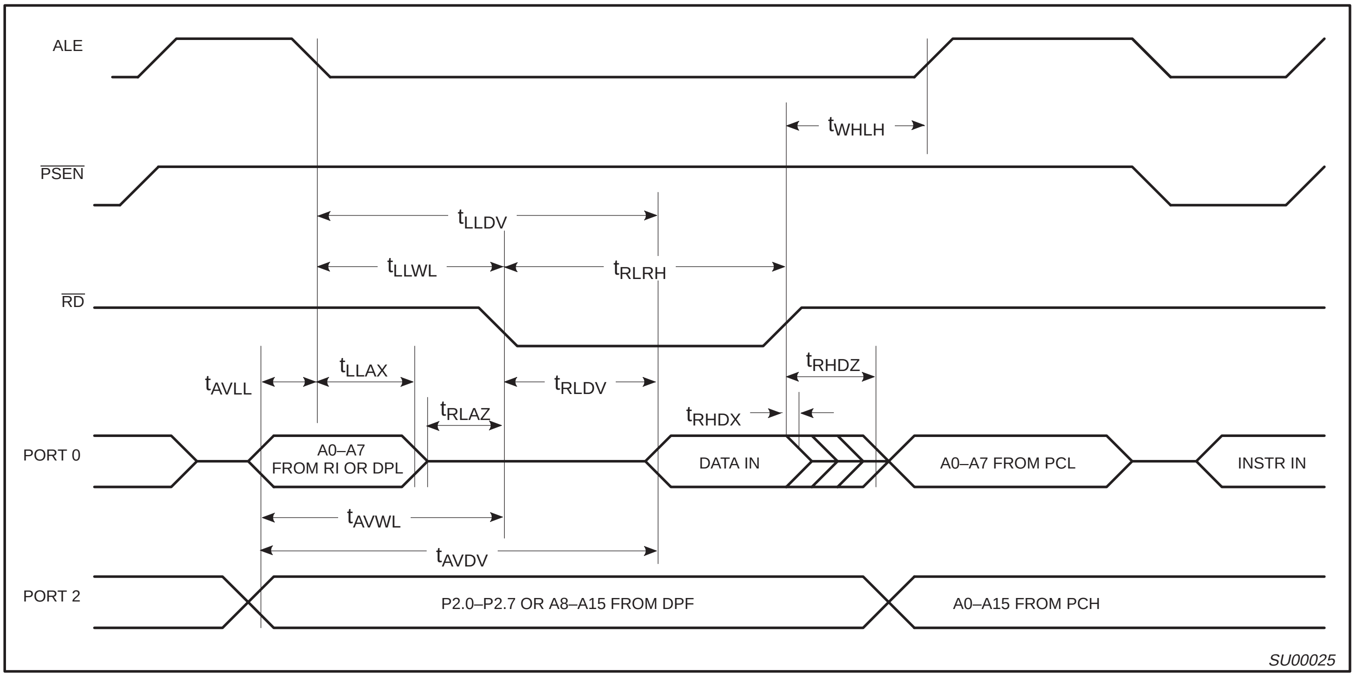
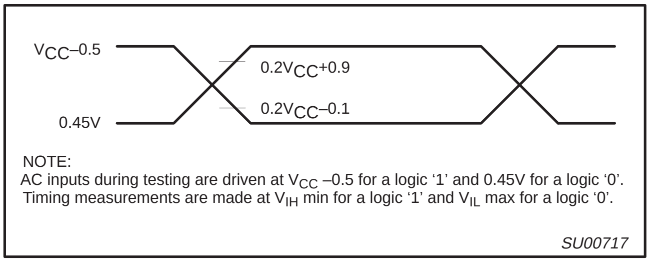


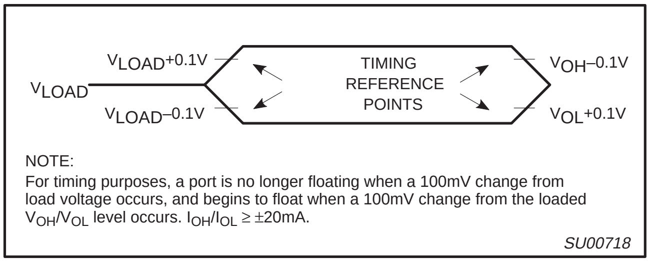
Figure 43. External Data Memory Read Cycle



NOTE:
AC inputs during testing are driven at V_{CC} –0.5 for a logic '1' and 0.45V for a logic '0'. Timing measurements are made at V_{IH} min for a logic '1' and V_{IL} max for a logic '0'.

SU00717

Figure 47. AC Testing Input/Output



NOTE:
For timing purposes, a port is no longer floating when a 100mV change from load voltage occurs, and begins to float when a 100mV change from the loaded V_{OH}/V_{OL} level occurs. I_{OH}/I_{OL} ≥ ±20mA.

SU00718

Figure 48. Float Waveform

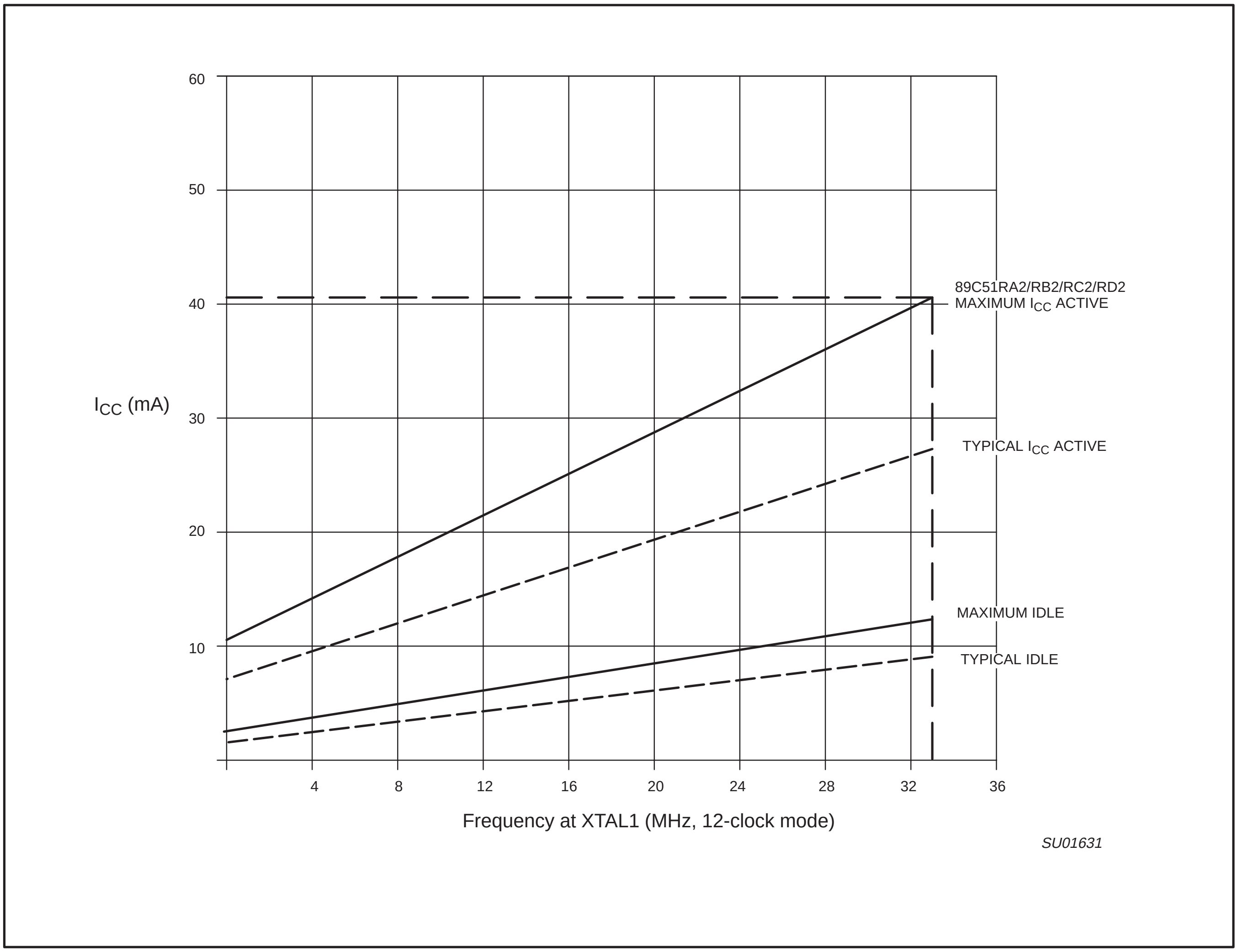
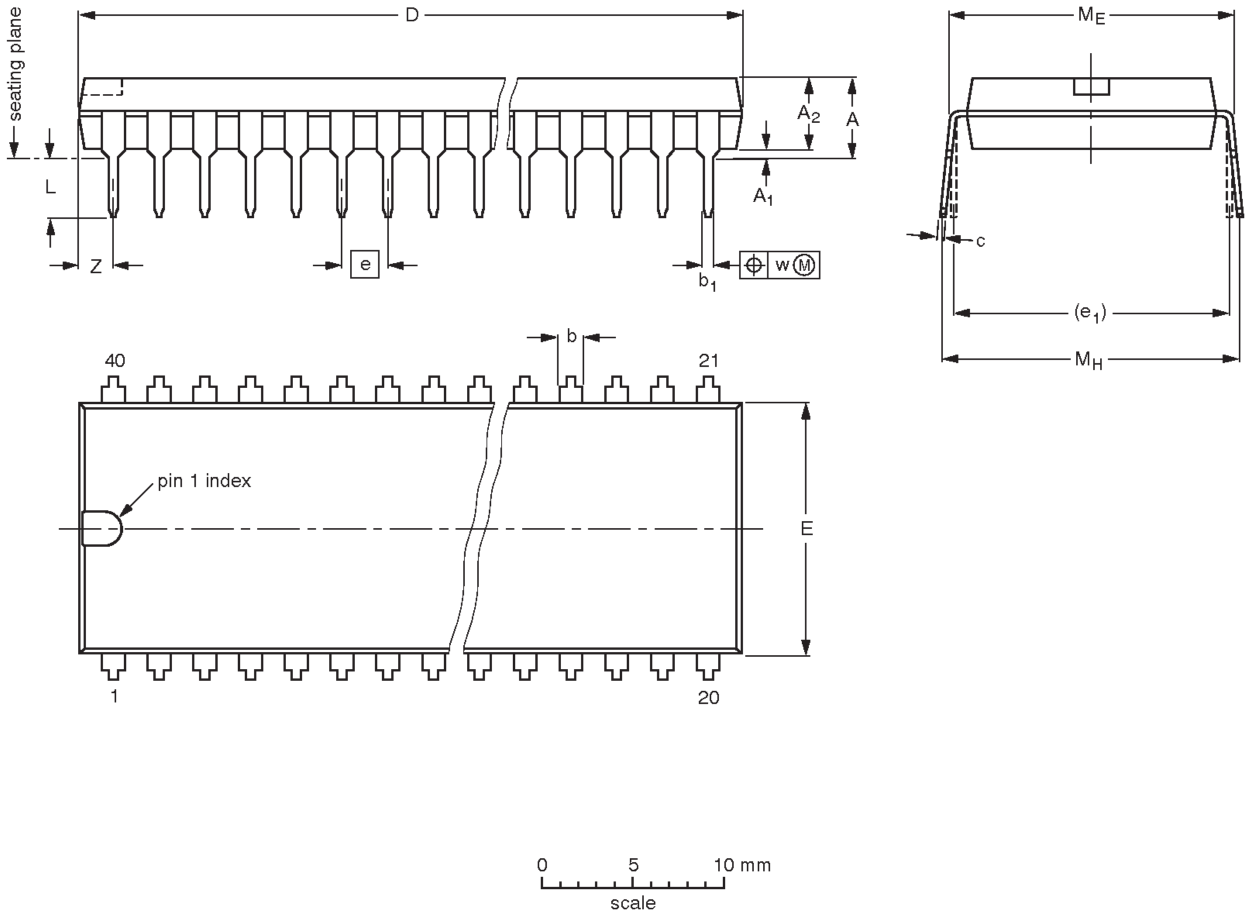


Figure 49. I_{CC} vs. FREQ
Valid only within frequency specifications of the device under test

DIP40: plastic dual in-line package; 40 leads (600 mil) SOT129-1



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A_1 min.	A_2 max.	b	b_1	c	D(1)	E(1)	e	e_1	L	M_E	M_H	w	Z(1) max.
mm	4.7	0.51	4.0	1.70 1.14	0.53 0.38	0.36 0.23	52.50 51.50	14.1 13.7	2.54	15.24	3.60 3.05	15.80 15.24	17.42 15.90	0.254	2.25
inches	0.19	0.020	0.16	0.067 0.045	0.021 0.015	0.014 0.009	2.067 2.028	0.56 0.54	0.10	0.60	0.14 0.12	0.62 0.60	0.69 0.63	0.01	0.089

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT129-1	051G08	MO-015	SC-511-40			~~95-01-14~~ 99-12-27