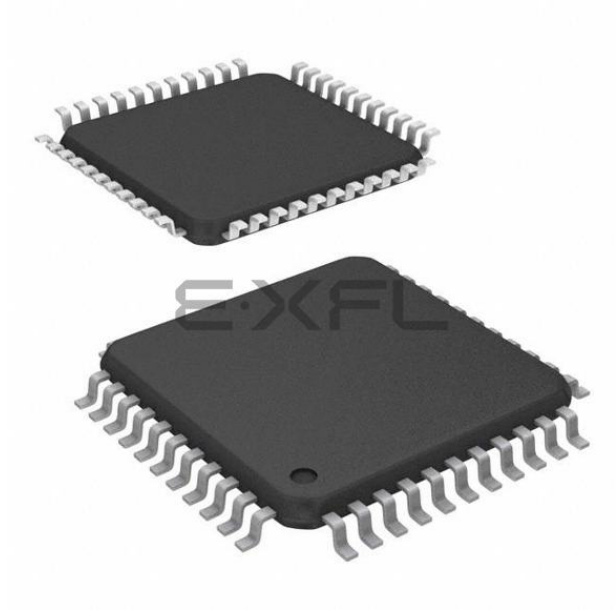


Welcome to E-XFL.COM

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	40MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	32
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K 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-TQFP
Supplier Device Package	44-TQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p89v51rd2fbc-557

5. Pinning information

5.1 Pinning

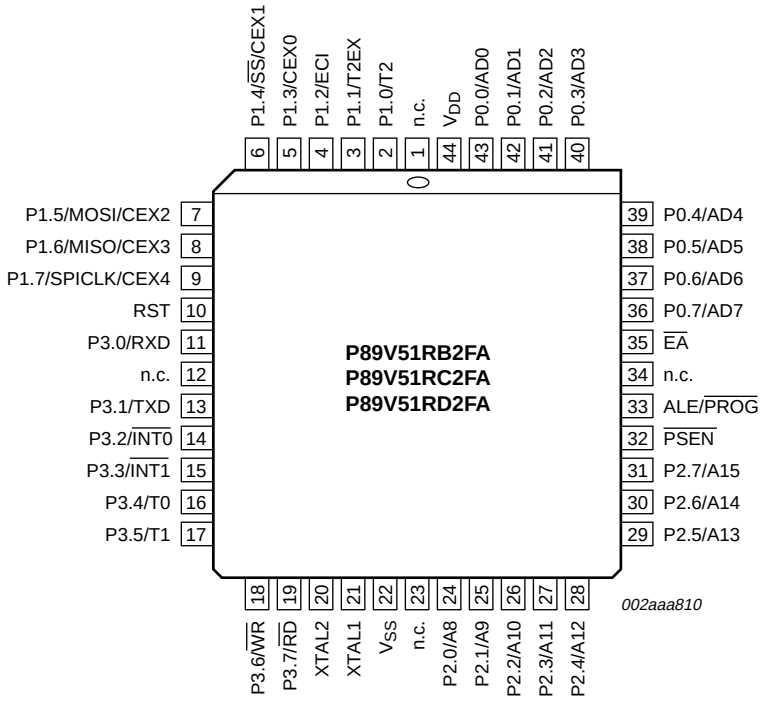


Fig 2. PLCC44 pin configuration

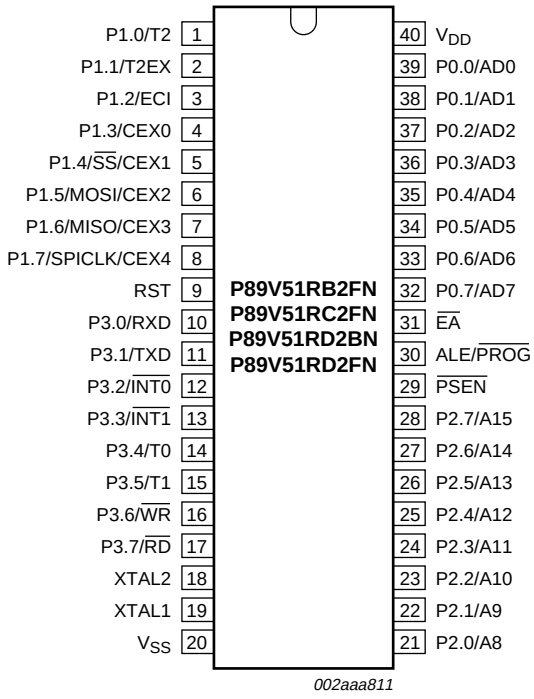


Fig 3. DIP40 pin configuration

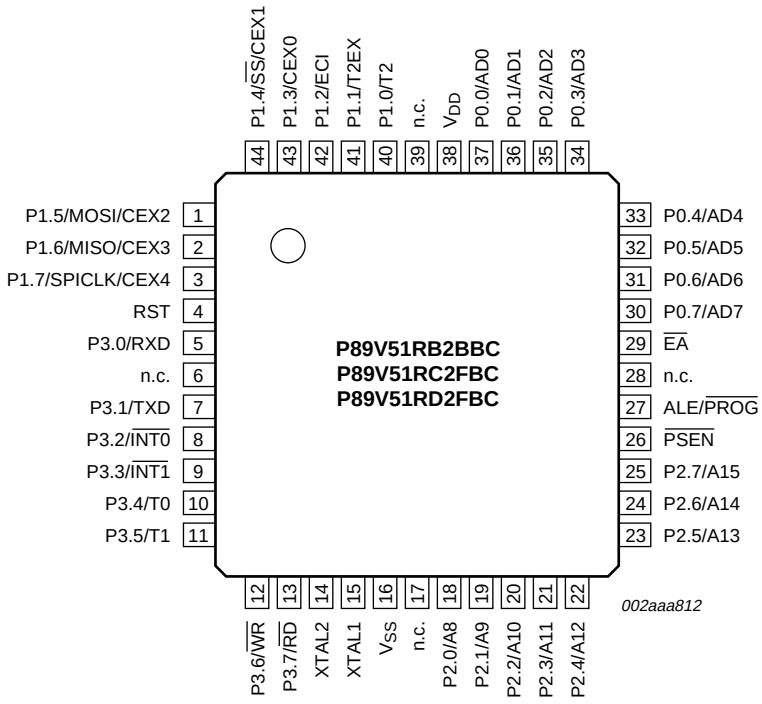


Fig 4. TQFP44 pin configuration

Table 3. P89V51RB2/RC2/RD2 pin description *…continued*

Symbol	Pin			Type	Description
	DIP40	TQFP44	PLCC44		
P1.2/ECI	3	42	4	I/O	**P1.2 —** Port 1 bit 2.
				I	**ECI —** External clock input. This signal is the external clock input for the PCA.
P1.3/CEX0	4	43	5	I/O	**P1.3 —** Port 1 bit 3.
				I/O	**CEX0 —** Capture/compare external I/O for PCA Module 0. Each capture/compare module connects to a Port 1 pin for external I/O. When not used by the PCA, this pin can handle standard I/O.
P1.4/$\overline{SS}$/CEX1	5	44	6	I/O	**P1.4 —** Port 1 bit 4.
				I	**$\overline{SS}$ —** Slave port select input for SPI.
				I/O	**CEX1 —** Capture/compare external I/O for PCA Module 1.
P1.5/MOSI/ CEX2	6	1	7	I/O	**P1.5 —** Port 1 bit 5.
				I/O	**MOSI —** Master Output Slave Input for SPI.
				I/O	**CEX2 —** Capture/compare external I/O for PCA Module 2.
P1.6/MISO/ CEX3	7	2	8	I/O	**P1.6 —** Port 1 bit 6.
				I/O	**MISO —** Master Input Slave Output for SPI.
				I/O	**CEX3 —** Capture/compare external I/O for PCA Module 3.
P1.7/SPICLK/ CEX4	8	3	9	I/O	**P1.7 —** Port 1 bit 7.
				I/O	**SPICLK —** Serial clock input/output for SPI.
				I/O	**CEX4 —** Capture/compare external I/O for PCA Module 4.
P2.0 to P2.7				I/O with internal pull-up	**Port 2**: Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins are pulled HIGH by the internal pull-ups when ‘1’s are written to them and can be used as inputs in this state. As inputs, Port 2 pins that are externally pulled LOW will source current (I_{IL}) because of the internal pull-ups. Port 2 sends the high-order address byte during fetches from external program memory and during accesses to external Data Memory that use 16-bit address (MOVX@DPTR). In this application, it uses strong internal pull-ups when transitioning to ‘1’s. Port 2 also receives some control signals and a partial of high-order address bits during the external host mode programming and verification.
P2.0/A8	21	18	24	I/O	**P2.0 —** Port 2 bit 0.
				O	**A8 —** Address bit 8.
P2.1/A9	22	19	25	I/O	**P2.1 —** Port 2 bit 1.
				O	**A9 —** Address bit 9.
P2.2/A10	23	20	26	I/O	**P2.2 —** Port 2 bit 2.
				O	**A10 —** Address bit 10.
P2.3/A11	24	21	27	I/O	**P2.3 —** Port 2 bit 3.
				O	**A11 —** Address bit 11.
P2.4/A12	25	22	28	I/O	**P2.4 —** Port 2 bit 4.
				O	**A12 —** Address bit 12.

Table 4. Special function registers *...continued*

** indicates SFRs that are bit addressable*

Name	Description	SFR address	Bit functions and addresses							
			MSB							LSB
FST	Flash Status Register	B6	-	SB	-	-	EDC	-	-	-
		Bit address	**AF**	**AE**	**AD**	**AC**	**AB**	**AA**	**A9**	**A8**
IEN0*	Interrupt Enable 0	A8H	EA	EC	ET2	ES0	ET1	EX1	ET0	EX0
		Bit address	**EF**	**EE**	**ED**	**EC**	**EB**	**EA**	**E9**	**E8**
IEN1*	Interrupt Enable 1	E8H	-	-	-	-	EBO			
		Bit address	**BF**	**BE**	**BD**	**BC**	**BB**	**BA**	**B9**	**B8**
IP0*	Interrupt Priority	B8H	-	PPC	PT2	PS	PT1	PX1	PT0	PX0
IP0H	Interrupt Priority 0 HIGH	B7H	-	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
		Bit address	**FF**	**FE**	**FD**	**FC**	**FB**	**FA**	**F9**	**F8**
IP1*	Interrupt Priority 1	F8H	-	-	-	-	PBO			
IP1H	Interrupt Priority 1 HIGH	F7H	-	-	-	-	PBOH			
FCF		B1H	-	-	-	-	-	-	SWR	BSEL
		Bit address	**87**	**86**	**85**	**84**	**83**	**82**	**81**	**80**
P0*	Port 0	80H	AD7	AD6	AD5	AD4	AD3	AD2	AD1	AD0
		Bit address	**97**	**96**	**95**	**94**	**93**	**92**	**91**	**90**
P1*	Port 1	90H	CEX4/ SPICLK	CEX3/ MISO	CEX2/ MOSI	CEX1/ $\overline{SS}$	CEX0	ECI	T2EX	T2
		Bit address	**A7**	**A6**	**A5**	**A4**	**A3**	**A2**	**A1**	**A0**
P2*	Port 2	A0H	A15	A14	A13	A12	A11	A10	A9	A8
		Bit address	**B7**	**B6**	**B5**	**B4**	**B3**	**B2**	**B1**	**B0**
P3*	Port 3	B0H	$\overline{RD}$	$\overline{WR}$	T1	T0	$\overline{INT1}$	$\overline{INT0}$	TXD	RXD
PCON	Power Control Register	87H	SMOD1	SMOD0	BOF	POF	GF1	GF0	PD	IDL
		Bit address	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
PSW*	Program Status Word	D0H	CY	AC	F0	RS1	RS0	OV	F1	P
RCAP2H	Timer2 Capture HIGH	CBH								
RCAP2L	Timer2 Capture LOW	CAH								
		Bit address	**9F**	**9E**	**9D**	**9C**	**9B**	**9A**	**99**	**98**
SCON*	Serial Port Control	98H	SM0/FE_	SM1	SM2	REN	TB8	RB8	TI	RI
SBUF	Serial Port Data Buffer Register	99H								

Table 4. Special function registers *…continued*

** indicates SFRs that are bit addressable*

Name	Description	SFR address	Bit functions and addresses							
			MSB							LSB
SADDR	Serial Port Address Register	A9H								
SADEN	Serial Port Address Enable	B9H								
		Bit address	**87**[1]	**86**[1]	**85**[1]	**84**[1]	**83**[1]	**82**[1]	**81**[1]	**80**[1]
SPCTL	SPI Control Register	D5H	SPIE	SPEN	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
SPCFG	SPI Configuration Register	AAH	SPIF	SPWCOL	-	-	-	-	-	-
SPDAT	SPI Data	86H								
SP	Stack Pointer	81H								
		Bit address	**8F**	**8E**	**8D**	**8C**	**8B**	**8A**	**89**	**88**
TCON*	Timer Control Register	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
		Bit address	**CF**	**CE**	**CD**	**CC**	**CB**	**CA**	**C9**	**C8**
T2CON*	Timer2 Control Register	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
T2MOD	Timer2 Mode Control	C9H	-	-	ENT2				T2OE	DCEN
TH0	Timer 0 HIGH	8CH								
TH1	Timer 1 HIGH	8DH								
TH2	Timer 2 HIGH	CDH								
TL0	Timer 0 LOW	8AH								
TL1	Timer 1 LOW	8BH								
TL2	Timer 2 LOW	CCH								
TMOD	Timer 0 and 1 Mode	89H	GATE	C/T	M1	M0	GATE	C/T	M1	M0
WDTC	Watchdog Timer Control	C0H	-	-	-	WDOUT	WDRE	WDTS	WDT	SWDT
WDTD	Watchdog Timer Data/Reload	85H								

[1] Unimplemented bits in SFRs (labeled '-') are 'X's (unknown) at all times. Unless otherwise specified, '1's should not be written to these bits since they may be used for other purposes in future derivatives. The reset values shown for these bits are '0's although they are unknown when read.

A chip-erase operation can be performed using a commercially available parallel programer. This operation will erase the contents of this boot block and it will be necessary for the user to reprogram this boot block (block 1) with the NXP-provided ISP/IAP code in order to use the ISP or IAP capabilities of this device. Go to http://www.nxp.com/support for questions or to obtain the hex file for this device.

6.3.3 ISP

ISP is performed without removing the microcontroller from the system. The ISP facility consists of a series of internal hardware resources coupled with internal firmware to facilitate remote programming of the P89V51RB2/RC2/RD2 through the serial port. This firmware is provided by NXP and embedded within each P89V51RB2/RC2/RD2 device. The NXP ISP facility has made in-circuit programming in an embedded application possible with a minimum of additional expense in components and circuit board area. The ISP function uses five pins (V_{DD}, V_{SS}, TXD, RXD, and RST). Only a small connector needs to be available to interface your application to an external circuit in order to use this feature.

6.3.4 Using ISP

The ISP feature allows for a wide range of baud rates to be used in your application, independent of the oscillator frequency. It is also adaptable to a wide range of oscillator frequencies. This is accomplished by measuring the bit-time of a single bit in a received character. This information is then used to program the baud rate in terms of timer counts based on the oscillator frequency. The ISP feature requires that an initial character (an uppercase U) be sent to the P89V51RB2/RC2/RD2 to establish the baud rate. The ISP firmware provides auto-echo of received characters. Once baud rate initialization has been performed, the ISP firmware will only accept Intel Hex-type records. Intel Hex records consist of ASCII characters used to represent hexadecimal values and are summarized below:

:NNAAAARRDD..DDCC<crlf>

In the Intel Hex record, the ‘NN’ represents the number of data bytes in the record. The P89V51RB2/RC2/RD2 will accept up to 32 data bytes. The ‘AAAA’ string represents the address of the first byte in the record. If there are zero bytes in the record, this field is often set to 0000. The ‘RR’ string indicates the record type. A record type of ‘00’ is a data record. A record type of ‘01’ indicates the end-of-file mark. In this application, additional record types will be added to indicate either commands or data for the ISP facility.

The maximum number of data bytes in a record is limited to 32 (decimal). ISP commands are summarized in Table 12. As a record is received by the P89V51RB2/RC2/RD2, the information in the record is stored internally and a checksum calculation is performed. The operation indicated by the record type is not performed until the entire record has been received. Should an error occur in the checksum, the P89V51RB2/RC2/RD2 will send an ‘X’ out the serial port indicating a checksum error. If the checksum calculation is found to match the checksum in the record, then the command will be executed. In most cases, successful reception of the record will be indicated by transmitting a ‘.’ character out the serial port.

Table 13. IAP function calls *…continued*

IAP function	IAP call parameters
Program Security Bit, Double Clock	**Input parameters:** R1 = 05H DPL = 01H = security bit DPL = 05H = Double Clock **Return parameter(s):** ACC = 00 = pass ACC = !00 = fail
Read Security Bit, Double Clock, SoftICE	**Input parameters:** ACC = 07H **Return parameter(s):** ACC = 00 SoftICE S/N-match 0 SB 0 DBL_CLK
Erase sector	**Input parameters:** R1 = 08H DPH = sector address high byte DPL = sector address low byte **Return parameter(s):** ACC = 00 = pass ACC = !00 = fail

6.4 Timers/counters 0 and 1

The two 16-bit Timer/counter registers: Timer 0 and Timer 1 can be configured to operate either as timers or event counters (see Table 14 and Table 15).

In the ‘Timer’ function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of six oscillator periods, the count rate is $\frac{1}{6}$ of the oscillator frequency.

In the ‘Counter’ function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled once 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 in the machine cycle following the one in which the transition was detected. Since it takes two machine cycles (12 oscillator periods) for 1-to-0 transition to be recognized, the maximum count rate is $\frac{1}{12}$ of the oscillator frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle. In addition to the ‘Timer’ or ‘Counter’ selection, Timer 0 and Timer 1 have four operating modes from which to select.

The ‘Timer’ or ‘Counter’ function is selected by control bits C/T in the Special Function Register 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 Timers/counters. Mode 3 is different. The four operating modes are described in the following text.

Table 14. TMOD - Timer/counter mode control register (address 89H) bit allocation

Not bit addressable; Reset value: 0000 0000B; Reset source(s): any source

Bit	7	6	5	4	3	2	1	0
Symbol	T1GATE	T1C/$\overline{T}$	T1M1	T1M0	T0GATE	T0C/$\overline{T}$	T0M1	T0M0

Table 15. TMOD - Timer/counter mode control register (address 89H) bit description

Bit	Symbol	Description
	T1/T0	Bits controlling Timer1/Timer0
	GATE	Gating control when set. Timer/counter 'x' is enabled only while 'INTx' pin is HIGH and 'TRx' control pin is set. When cleared, Timer 'x' is enabled whenever 'TRx' control bit is set.
	C/$\overline{T}$	Gating Timer or Counter Selector cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from 'Tx' input pin).

Table 16. TMOD - Timer/counter mode control register (address 89H) M1/M0 operating mode

M1	M0	Operating mode	
0	0	0	8048 timer 'TLx' serves as 5-bit prescaler
0	1	1	16-bit Timer/counter 'THx' and 'TLx' are cascaded; there is no prescaler.
1	0	2	8-bit auto-reload Timer/counter 'THx' holds a value which is to be reloaded into 'TLx' each time it overflows.
1	1	3	(Timer 0) TL0 is an 8-bit Timer/counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only controlled by Timer 1 control bits.
1	1	3	(Timer 1) Timer/counter 1 stopped.

Table 17. TCON - Timer/counter control register (address 88H) bit allocation

Bit addressable; Reset value: 0000 0000B; Reset source(s): any reset

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

Table 18. TCON - Timer/counter control register (address 88H) bit description

Bit	Symbol	Description
7	TF1	Timer 1 overflow flag. Set by hardware on Timer/counter overflow. Cleared by hardware when the processor vectors to Timer 1 Interrupt routine, or by software.
6	TR1	Timer 1 Run control bit. Set/cleared by software to turn Timer/counter 1 on/off.
5	TF0	Timer 0 overflow flag. Set by hardware on Timer/counter overflow. Cleared by hardware when the processor vectors to Timer 0 Interrupt routine, or by software.
4	TR0	Timer 0 Run control bit. Set/cleared by software to turn Timer/counter 0 on/off.
3	IE1	Interrupt 1 Edge flag. Set by hardware when external interrupt 1 edge/low level is detected. Cleared by hardware when the interrupt is processed, or by software.

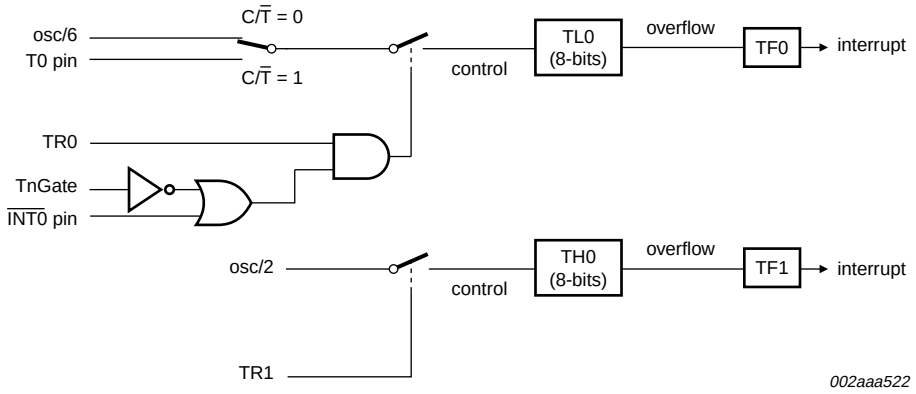


Fig 11. Timer/counter 0 mode 3 (two 8-bit counters)

6.5 Timer 2

Timer 2 is a 16-bit Timer/counter which can operate as either an event timer or an event counter, as selected by C/T2 in the special function register T2CON. Timer 2 has four operating modes: Capture, Auto-reload (up or down counting), Clock-out, and Baud Rate Generator which are selected according to Table 19 using T2CON (Table 20 and Table 21) and T2MOD (Table 22 and Table 23).

Table 19. Timer 2 operating mode

RCLK + TCLK	CP/$\overline{RL2}$	TR2	T2OE	Mode
0	0	1	0	16-bit auto reload
0	1	1	0	16-bit capture
0	0	1	1	programmable clock-out
1	X	1	0	baud rate generator
X	X	0	X	off

Table 20. T2CON - Timer/counter 2 control register (address C8H) bit allocation

Bit addressable; Reset value: 00H

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

Table 21. T2CON - Timer/counter 2 control register (address C8H) bit description

Bit	Symbol	Description
7	TF2	Timer 2 overflow flag set by a Timer 2 overflow and must be cleared by software. TF2 will not be set when either RCLK or TCLK = 1 or when Timer 2 is in Clock-out mode.
6	EXF2	Timer 2 external flag is set when Timer 2 is in capture, reload or baud-rate mode, EXEN2 = 1 and a negative transition on T2EX occurs. If Timer 2 interrupt is enabled EXF2 = 1 causes the CPU to vector to the Timer 2 interrupt routine. EXF2 must be cleared by software.
5	RCLK	Receive clock flag. When set, causes the UART 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.

TCLK = 1, Timer 2 is used as the UART transmit baud rate generator. RCLK has the same effect for the UART receive baud rate. With these two bits, the serial port can have different receive and transmit baud rates – Timer 1 or Timer 2.

Figure 15 shows Timer 2 in baud rate generator mode:

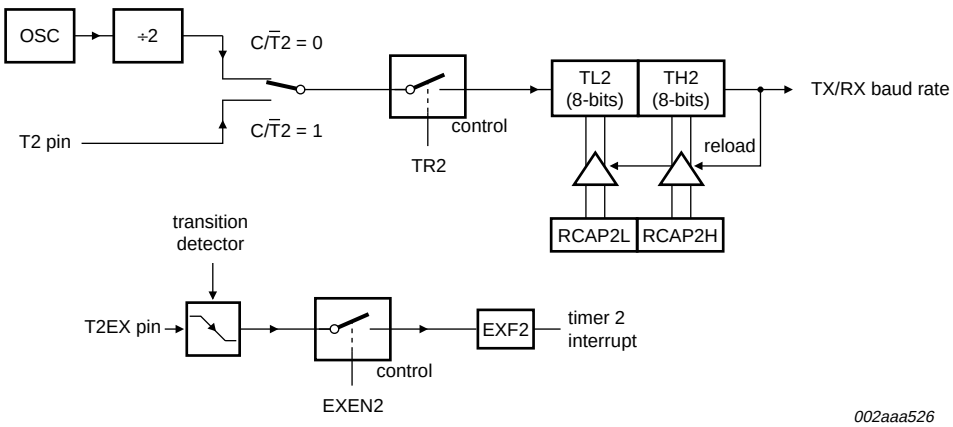


Fig 15. Timer 2 in Baud Rate Generator mode

The baud rate generation mode is like the auto-reload mode, when a rollover in TH2 causes the Timer 2 registers to be reloaded with the 16-bit value in registers RCAP2H and RCAP2L, which are preset by software.

The baud rates in modes 1 and 3 are determined by Timer 2's overflow rate given below:

Modes 1 and 3 baud rates = Timer 2 Overflow Rate / 16

The timer can be configured for either 'timer' or 'counter' operation. In many applications, it is configured for 'timer' operation (C/$\overline{T}$2 = 0). Timer operation is different for Timer 2 when it is being used as a baud rate generator.

Usually, as a timer it would increment every machine cycle (i.e., $\frac{1}{6}$ the oscillator frequency). As a baud rate generator, it increments at the oscillator frequency. Thus the baud rate formula is as follows:

Modes 1 and 3 baud rates =

$$\frac{OscillatorFrequency}{(n \times (65536 - (RCAP2H, RCAP2L)))} \quad (3)$$

n = 32 in X1 mode, 16 in X2 mode

Where: (RCAP2H, RCAP2L) = The content of RCAP2H and RCAP2L taken as a 16-bit unsigned integer.

The Timer 2 as a baud rate generator mode is valid only if RCLK and/or TCLK = 1 in T2CON register. Note that a rollover in TH2 does not set TF2, and will not generate an interrupt. Thus, the Timer 2 interrupt does not have to be disabled when Timer 2 is in the baud rate generator mode. Also if the EXEN2 (T2 external enable flag) is set, a 1-to-0 transition in T2EX (Timer/counter 2 trigger input) will set EXF2 (T2 external flag) but will

Table 26. SCON - Serial port control register (address 98H) bit description *...continued*

Bit	Symbol	Description
2	RB8	In modes 2 and 3, 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 undefined.
1	TI	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the stop bit in the other modes, in any serial transmission. Must be cleared by software.
0	RI	Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or approximately halfway through the stop bit time in all other modes. (See SM2 for exceptions). Must be cleared by software.

Table 27. SCON - Serial port control register (address 98H) SM0/SM1 mode definition

SM0, SM1	UART mode	Baud rate
0 0	0: shift register	CPU clock / 6
0 1	1: 8-bit UART	variable
1 0	2: 9-bit UART	CPU clock / 32 or CPU clock / 16
1 1	3: 9-bit UART	variable

6.6.5 Framing error

Framing error (FE) is reported in the SCON.7 bit if SMOD0 (PCON.6) = 1. If SMOD0 = 0, SCON.7 is the SM0 bit for the UART, it is recommended that SM0 is set up before SMOD0 is set to '1'.

6.6.6 More about UART mode 1

Reception is initiated by a detected 1-to-0 transition at RXD. For this purpose RXD is sampled at a rate of 16 times whatever baud rate has been established. When a transition is detected, the divide-by-16 counter is immediately reset to align its rollovers with the boundaries of the incoming bit times.

The 16 states of the counter divide each bit time into 16ths. At the 7th, 8th, and 9th 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 2 of the 3 samples. This is done for noise rejection. If the value accepted during the first bit time is not 0, the receive circuits are reset and the unit goes back to looking for another 1-to-0 transition. This is to provide rejection of false start bits. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated: (a) RI = 0, and (b) 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 8 data bits go into SBUF, and RI is activated.

6.6.7 More about UART modes 2 and 3

Reception is performed in the same manner as in mode 1.

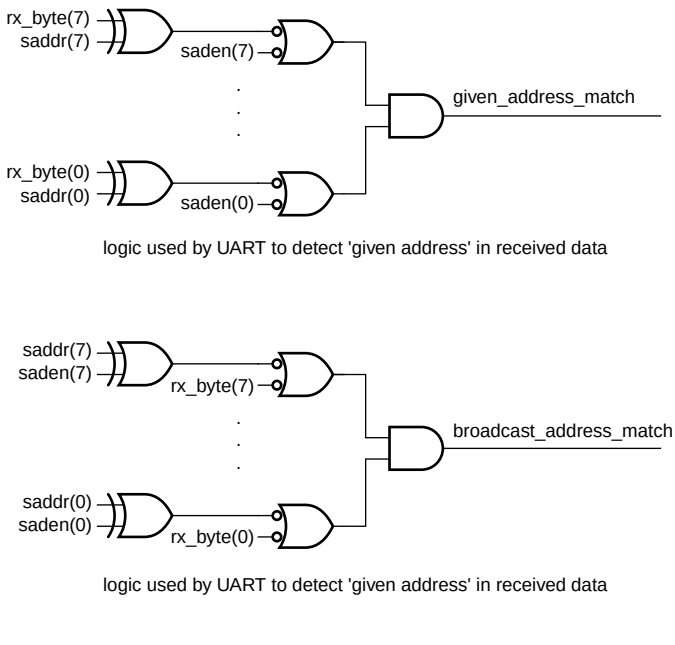


Fig 16. Schemes used by the UART to detect ‘given’ and ‘broadcast’ addresses when multiprocessor communications is enabled

The following examples will help to show the versatility of this scheme.

Example 1, slave 0:

$$\begin{array}{l} \text{SADDR} = 1100\ 0000 \\ \underline{\text{SADEN} = 1111\ 1101} \\ \text{Given} = 1100\ 00\text{X}0 \end{array} \quad (4)$$

Example 2, slave 1:

$$\begin{array}{l} \text{SADDR} = 1100\ 0000 \\ \underline{\text{SADEN} = 1111\ 1110} \\ \text{Given} = 1100\ 000\text{X} \end{array} \quad (5)$$

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.

clock output and input for the master and slave modes, respectively. The SPI clock generator will start following a write to the master devices SPI data register. The written data is then shifted out of the MOSI pin on the master device into the MOSI pin of the slave device. Following a complete transmission of one byte of data, the SPI clock generator is stopped and the SPIF flag is set. An SPI interrupt request will be generated if the SPI Interrupt Enable bit (SPIE) and the Serial Port Interrupt Enable bit (ES) are both set.

An external master drives the Slave Select input pin, $\overline{SS}$/P1[4], low to select the SPI module as a slave. If $\overline{SS}$/P1[4] has not been driven low, then the slave SPI unit is not active and the MOSI/P1[5] port can also be used as an input port pin.

CPHA and CPOL control the phase and polarity of the SPI clock. Figure 18 and Figure 19 show the four possible combinations of these two bits.

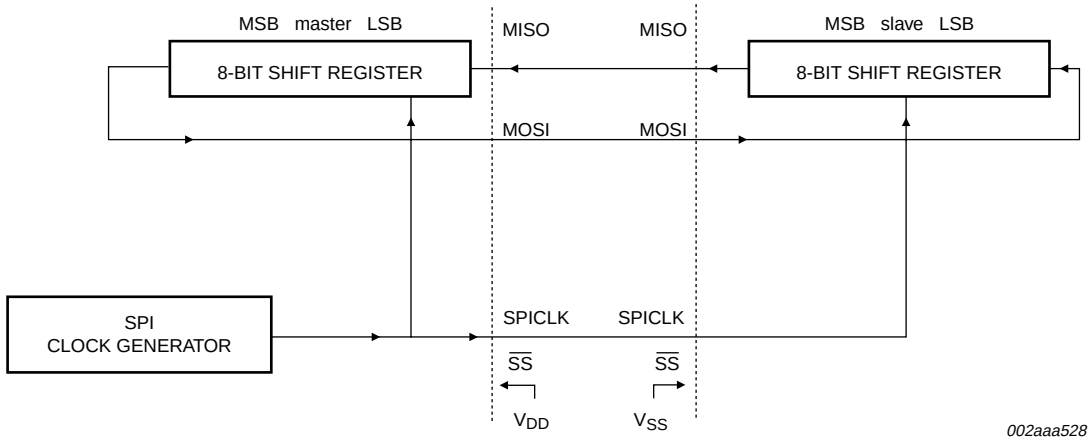


Fig 17. SPI master-slave interconnection

Table 28. SPCR - SPI control register (address D5H) bit allocation

Bit addressable; Reset source(s): any reset; Reset value: 0000 0000B

Bit	7	6	5	4	3	2	1	0
Symbol	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0

Table 29. SPCR - SPI control register (address D5H) bit description

Bit	Symbol	Description
7	SPIE	If both SPIE and ES are set to one, SPI interrupts are enabled.
6	SPE	SPI enable bit. When set enables SPI.
5	DORD	Data transmission order. 0 = MSB first; 1 = LSB first in data transmission.
4	MSTR	Master/slave select. 1 = master mode, 0 = slave mode.
3	CPOL	Clock polarity. 1 = SPICLK is high when idle (active LOW), 0 = SPICLK is low when idle (active HIGH).

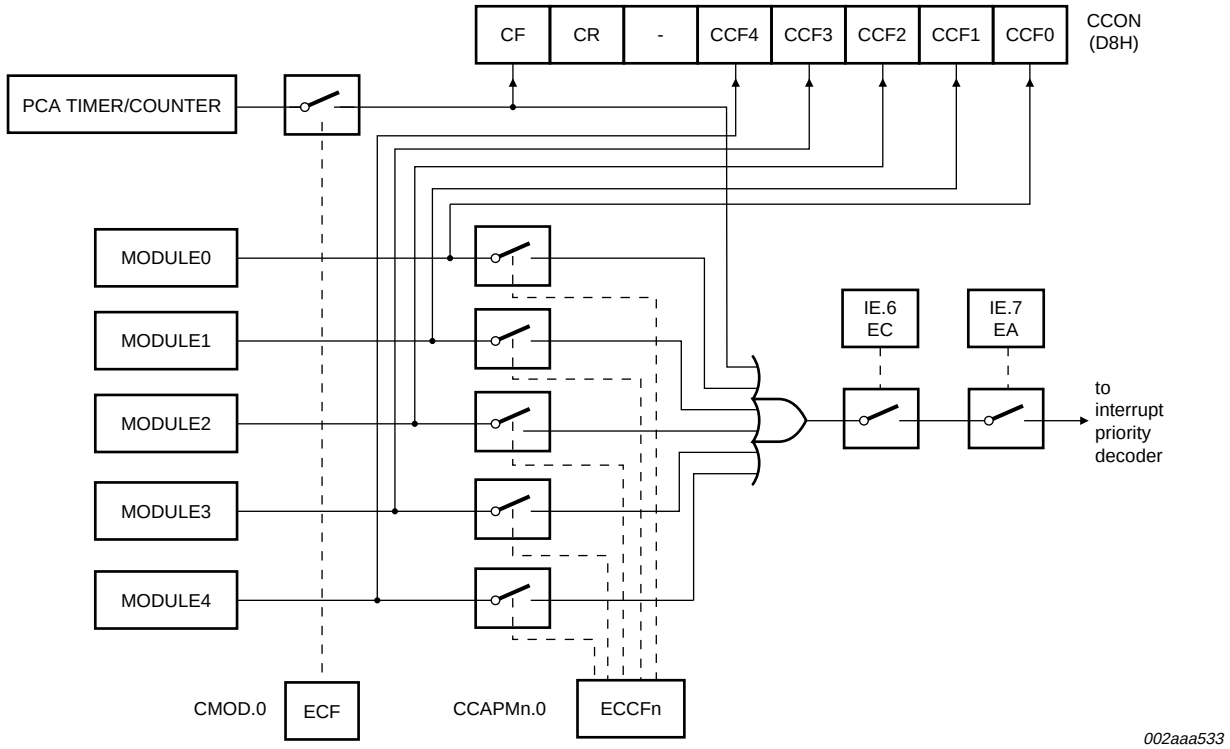


Fig 22. PCA interrupt system

Table 35. CMOD - PCA counter mode register (address D9H) bit allocation

Not bit addressable; Reset value: 00H

Bit	7	6	5	4	3	2	1	0
Symbol	CIDL	WDTE	-	-	-	CPS1	CPS0	ECF

Table 36. CMOD - PCA counter mode register (address D9H) bit description

Bit	Symbol	Description
7	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.
6	WDTE	Watchdog Timer Enable: WDTE = 0 disables watchdog timer function on module 4. WDTE = 1 enables it.
5 to 3	-	Reserved for future use. Should be set to '0' by user programs.
2 to 1	CPS1, CPS0	PCA Count Pulse Select (see Table 37 below).
0	ECF	PCA Enable Counter Overflow Interrupt: ECF = 1 enables CF bit in CCON to generate an interrupt. ECF = 0 disables that function.

Table 41. CCAPMn - PCA modules compare/capture register (address CCAPM0 0DAH, CCAPM1 0DBH, CCAPM2 0DCH, CCAPM3 0DDH, CCAPM4 0DEH) bit description

Bit	Symbol	Description
2	TOGn	Toggle. When TOGn = 1, a match of the PCA counter with this module's compare/capture register causes the CEXn pin to toggle.
1	PWMn	Pulse Width Modulation mode. PWMn = 1 enables the CEXn pin to be used as a pulse width modulated output.
0	ECCFn	Enable CCF Interrupt. Enables compare/capture flag CCFn in the CCON register to generate an interrupt.

Table 42. PCA module modes (CCAPMn register)

ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	Module function
0	0	0	0	0	0	0	no operation
x	1	0	0	0	0	x	16-bit capture by a positive-edge trigger on CEXn
x	0	1	0	0	0	x	16-bit capture by a negative-edge trigger on CEXn
x	1	1	0	0	0	x	16-bit capture by any transition on CEXn
1	0	0	1	0	0	x	16-bit software timer
1	0	0	1	1	0	x	16-bit high-speed output
1	0	0	0	0	1	0	8-bit PWM
1	0	0	1	x	0	x	Watchdog timer

6.9.1 PCA capture mode

To use one of the PCA modules in the capture mode (Figure 23) either one or both of the CCAPM bits CAPN and CAPP for that module must be set. The external CEX input for the module (on port 1) is sampled for a transition. When a valid transition occurs the PCA hardware loads the value of the PCA counter registers (CH and CL) into the module's capture registers (CCAPnL and CCAPnH).

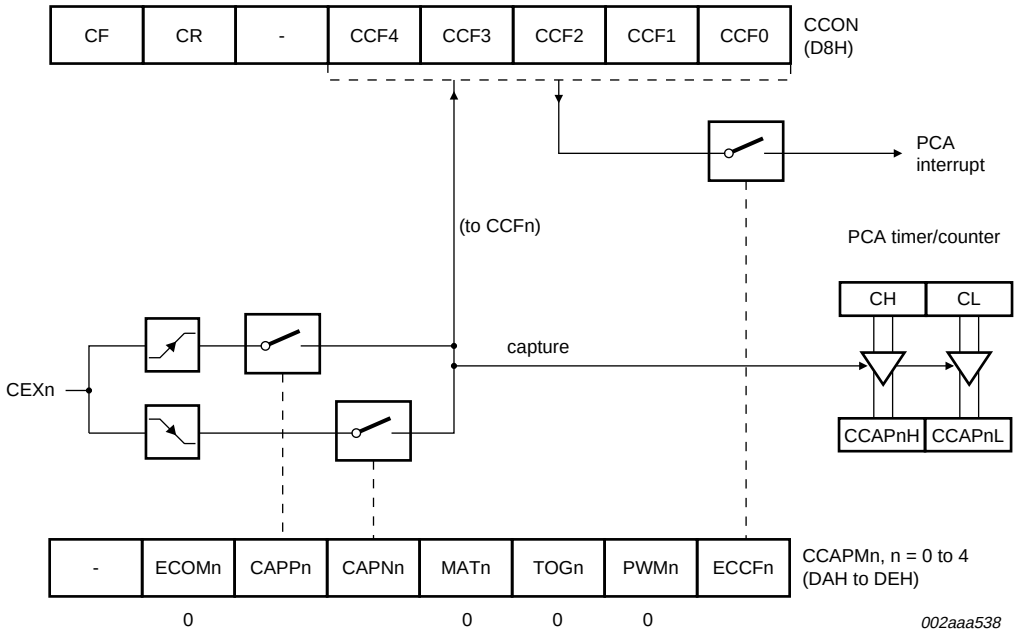


Fig 23. PCA capture mode

If the CCFn bit for the module in the CCON SFR and the ECCFn bit in the CCAPMn SFR are set then an interrupt will be generated.

6.9.2 16-bit software timer mode

The PCA modules can be used as software timers (Figure 24) by setting both the ECOM and MAT bits in the modules CCAPMn register. The PCA timer will be compared to the module's capture registers and when a match occurs an interrupt will occur if the CCFn (CCON SFR) and the ECCFn (CCAPMn SFR) bits for the module are both set.

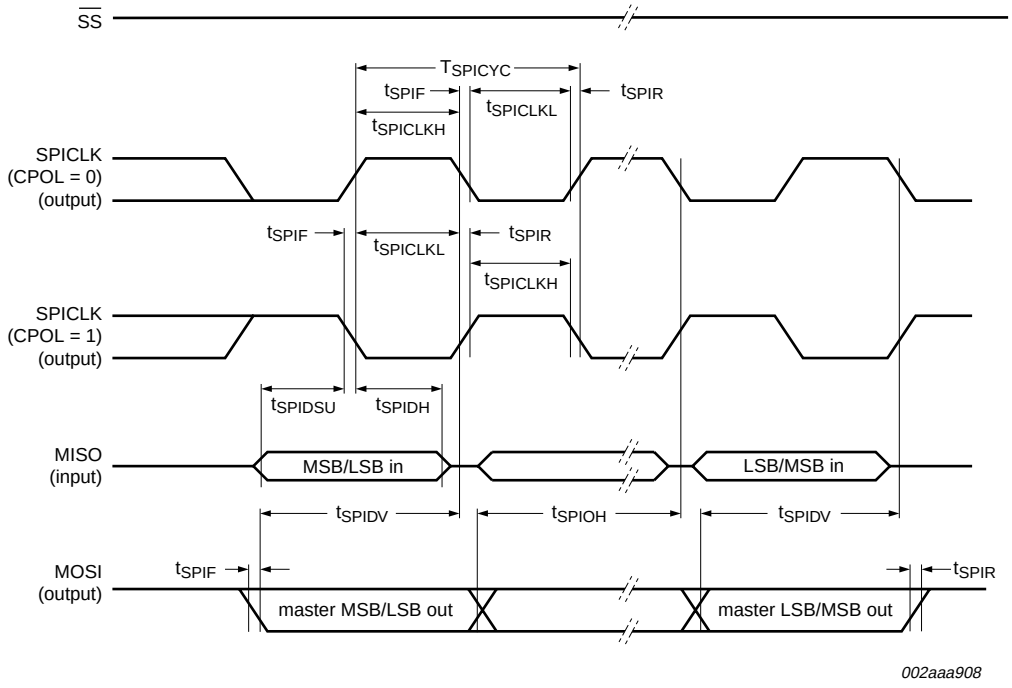


Fig 36. SPI master timing (CPHA = 0)

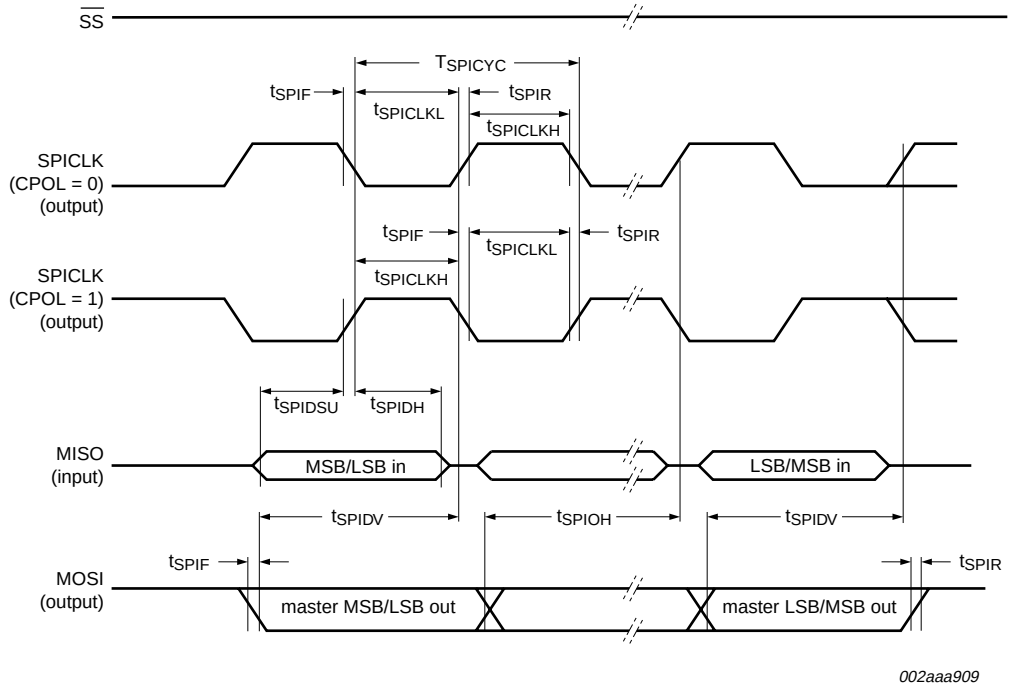


Fig 37. SPI master timing (CPHA = 1)

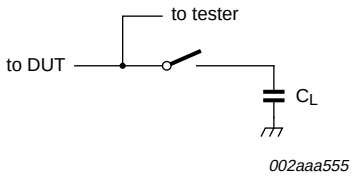
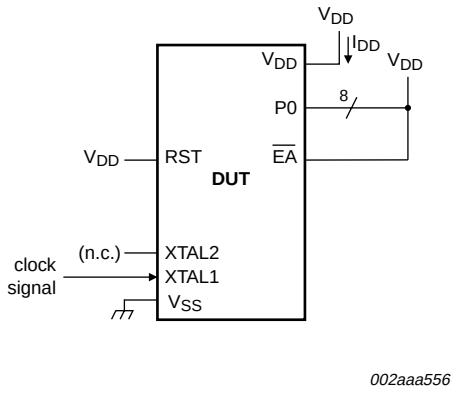
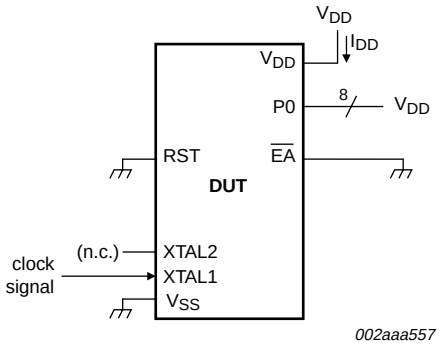


Fig 40. Test load example



All other pins disconnected

Fig 41. I_{DD} test condition, Active mode



All other pins disconnected

Fig 42. I_{DD} test condition, Idle mode

11. Abbreviations

Table 67. Abbreviations

Acronym	Description
DUT	Device Under Test
EMI	Electro-Magnetic Interference
IAP	In-Application Programming
ISP	In-System Programming
MCU	Microcontroller Unit
PCA	Programmable Counter Array
PWM	Pulse Width Modulator
RC	Resistance-Capacitance
SFR	Special Function Register
SPI	Serial Peripheral Interface
TTL	Transistor-Transistor Logic
UART	Universal Asynchronous Receiver/Transmitter

12. Revision history

Table 68. Revision history

Document ID	Release date	Data sheet status	Change notice	Supersedes
P89V51RB2_RC2_RD2_5	20091112	Product data sheet	-	P89V51RB2_RC2_RD2_4
Modifications:	• Table 37: Changed 2nd row, f_{osc} / 6 to f_{osc} / 2. • Table 62: Changed 12 MHz max values for $I_{DD(oper)}$ and $I_{DD(idle)}$. • Table 3: Removed sentence “However, Security lock level 4 will disable $\overline{EA}$...” from $\overline{EA}$ pin description. • Changed SCK to SPICLK throughout data sheet. • Table 3: Changed SCK to SPICLK and updated pin description.			
P89V51RB2_RC2_RD2_4	20070501	Product data sheet	-	P89V51RB2_RC2_RD2-03
P89V51RB2_RC2_RD2-03	20041202	Product data	-	P89V51RB2_RC2_RD2-02
P89V51RD2-02	20041011	Product data	-	P89V51RD2-01
P89V51RD2-01	20040301	Product data	-	-