P89CV51RB2/RC2/RD2

8-bit 80C51 5 V low power 64 kB flash microcontroller with 1 kB RAM, SPI, 6-clock CPU with 6/12-clock peripherals

Rev. 03 — 25 August 2009 Product data sheet

1. General description

The P89CV51RB2/RC2/RD2 are three types of 80C51 microcontroller with respectively 16 kB/32 kB/64 kB flash and 1 kB of data RAM. These devices are designed to be drop-in and software-compatible replacements for the popular P89C51RB2/RC2/RD2 devices. Both the In-System Programming (ISP) and In-Application Programming (IAP) boot codes are upward compatible.

Additional features of the P89CV51RB2/RC2/RD2 devices compared to the P89C51RB2/RC2/RD2 are the inclusion of an SPI interface, larger RAM size, and the ability to erase code memory in 128-B page blocks.

The IAP capability combined with the 128-B page size allows for efficient use of the code memory for non-volatile data storage.

2. Features

2.1 Principal features

- Supports 12-clock (default) or 6-clock mode selection via ISP or parallel programmer
- 6-clock/12-clock mode programmable “on-the-fly” by an SFR bit
- Peripherals (PCA, timers, UART) may use either 6-clock or 12-clock mode while the CPU is in 6-clock mode
- 128-B page erase for efficient use of code memory as non-volatile data storage
- 0 MHz to 40 MHz operating frequency in 12× mode, 20 MHz in 6× mode
- 16/32/64 kB of on-chip flash user-code memory with ISP and IAP
- 1 kB RAM
- SPI (Serial Peripheral Interface) and enhanced UART
- PCA (Programmable Counter Array) with PWM and capture/compare functions
- Three 16-bit timers/counters

2.2 Additional features

- Four 8-bit I/O ports
- WatchDog Timer (WDT)
- 30 ms page erase, 150 ms block erase
- PLCC44 and TQFP44 packages
- Ten interrupt sources with four priority levels
- Second DPTR register
- Low EMI mode (ALE inhibit)

- Power-down mode with external interrupt wake-up
- Idle mode

2.3 Comparison to P89C51RB2/RC2/RD2 devices

- **SPI:** The P89CV51RB2/RC2/RD2 devices have an SPI interface that was not present on the P89C51RB2/RC2/RD2 devices.
- **Smaller block size:** The page size decreased from 4 kB to 128 B. These smaller pages can be erased and reprogrammed using IAP function calls, which makes practical use of code memory for non-volatile data storage. A page is erased in 30 ms or less. IAP and ISP code both support 128-B page operations. The IAP and ISP code uses multiple page-erase operations to emulate the erasing of larger block sizes (8 kB and 16 kB) to maintain firmware compatibility.
- **Status bit replaces Status byte:** Automatic entry into ISP mode following a reset is now controlled by one status bit. Its operation is almost identical to that used by the previous devices, which was based on the zero/non-zero value of the status byte.
- **Faster block erase:** The erase time for the entire user-code memory of the P89CV51RB2/RC2/RD2 devices is 150 ms, which is a significant improvement.
- **Larger RAM size:** RAM size increased from 512 B to 1 kB.

3. Ordering information

Table 1. Ordering information

Type number	Package		
	Name	Description	Version
P89CV51RB2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89CV51RB2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89CV51RC2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89CV51RC2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89CV51RD2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89CV51RD2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1

3.1 Ordering options

Table 2. Ordering options

Type number	Flash memory	RAM	Temperature range	Frequency
P89CV51RB2FA	16 kB	1 kB	–40 °C to +85 °C	0 MHz to 40 MHz
P89CV51RB2FBC				
P89CV51RC2FA	32 kB	1 kB		
P89CV51RC2FBC				
P89CV51RD2FA	64 kB	1 kB		
P89CV51RD2FBC				

4. Block diagram

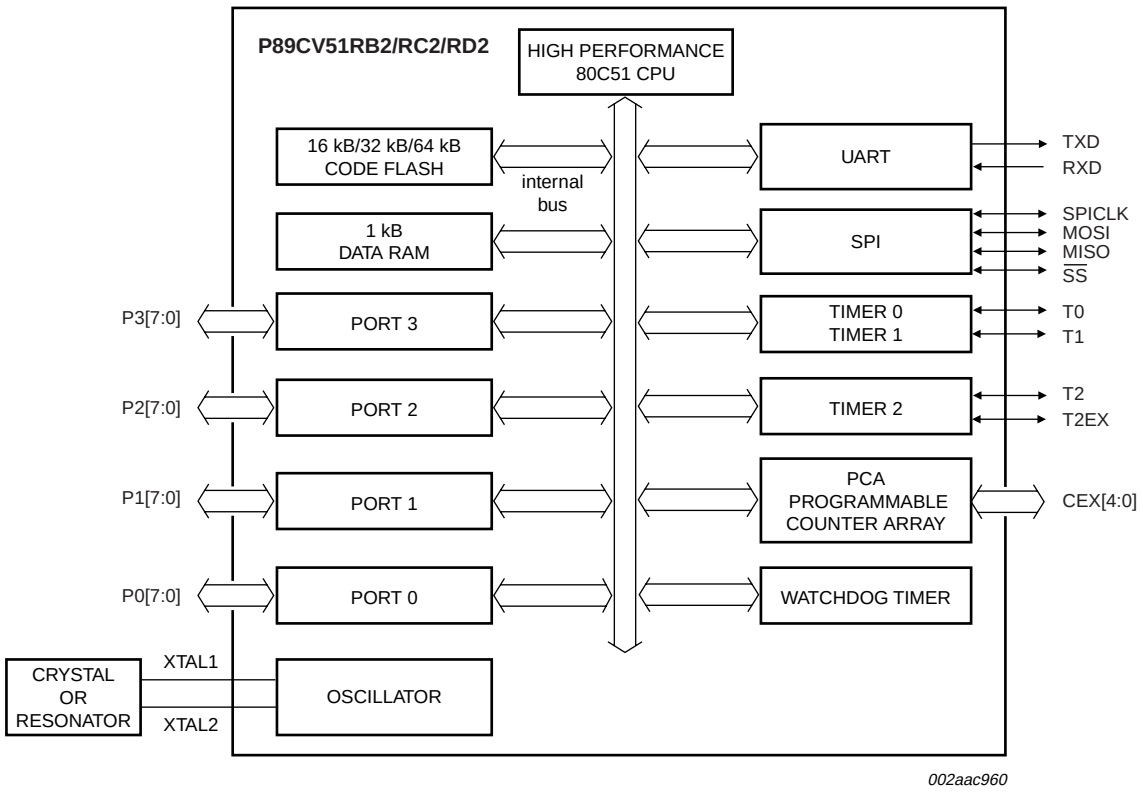


Fig 1. Block diagram

5. Pinning information

5.1 Pinning

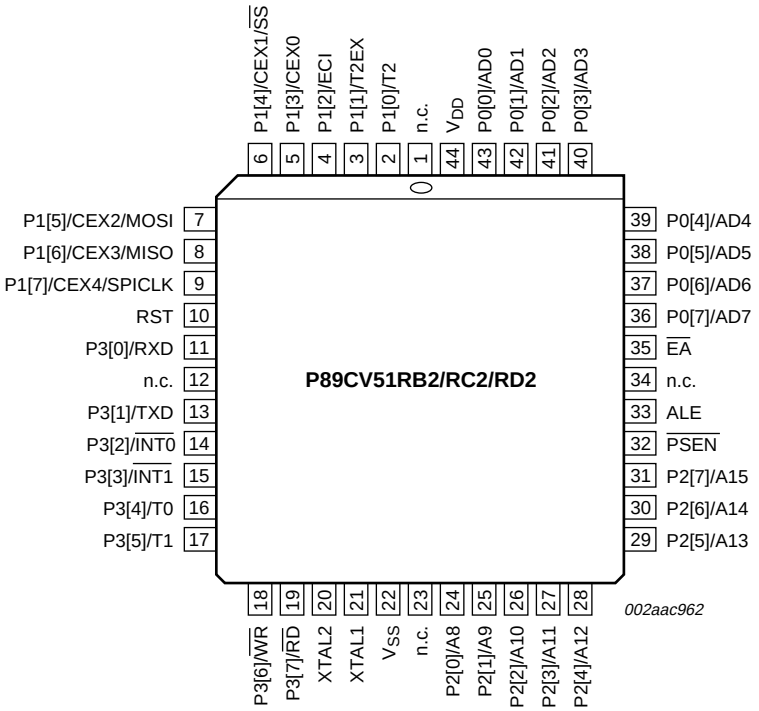


Fig 2. PLCC44 pin configuration

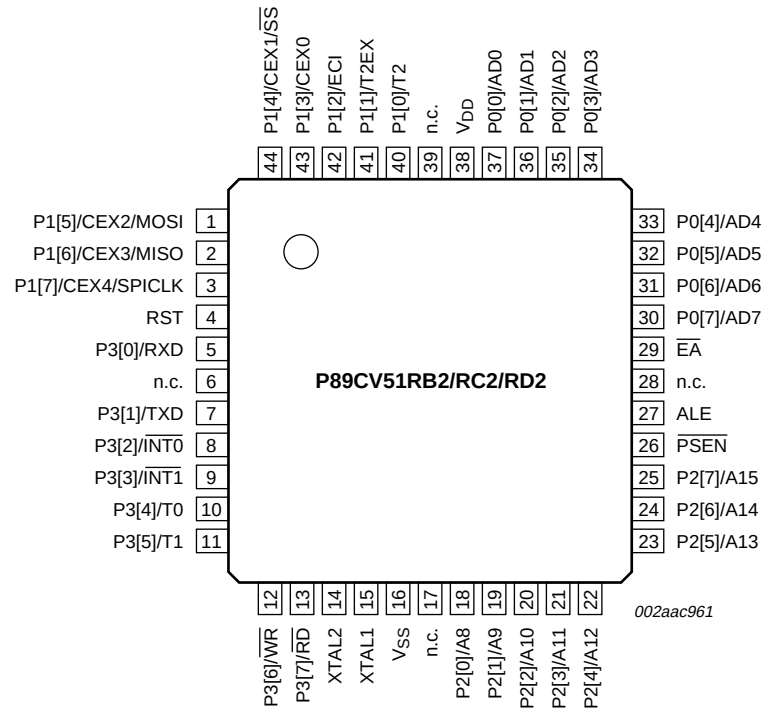


Fig 3. TQFP44 pin configuration

5.2 Pin description

Table 3. P89CV51RB2/RC2/RD2 Pin description

Symbol	Pin		Type	Description
	PLCC44	TQFP44		
P0[0] to P0[7]			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 in this state can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external code and data memory. In this application, it uses strong internal pull-ups when transitioning to 1s. External pull-ups are required when used as a general purpose I/O port.
P0[0]/AD0	43	37	I/O	**P0[0] —** Port 0 bit 0.
			I/O	**AD0 —** Address/data bit 0.
P0[1]/AD1	42	36	I/O	**P0[1] —** Port 0 bit 1.
			I/O	**AD1 —** Address/data bit 1.
P0[2]/AD2	41	35	I/O	**P0[2] —** Port 0 bit 2.
			I/O	**AD2 —** Address/data bit 2.
P0[3]/AD3	40	34	I/O	**P0[3] —** Port 0 bit 3.
			I/O	**AD3 —** Address/data bit 3.
P0[4]/AD4	39	33	I/O	**P0[4] —** Port 0 bit 4.
			I/O	**AD4 —** Address/data bit 4.
P0[5]/AD5	38	32	I/O	**P0[5] —** Port 0 bit 5.
			I/O	**AD5 —** Address/data bit 5.

Table 3. P89CV51RB2/RC2/RD2 Pin description *...continued*

Symbol	Pin		Type	Description
	PLCC44	TQFP44		
P0[6]/AD6	37	31	I/O	**P0[6] —** Port 0 bit 6.
			I/O	**AD6 —** Address/data bit 6.
P0[7]/AD7	36	30	I/O	**P0[7] —** Port 0 bit 7.
			I/O	**AD7 —** Address/data bit 7.
P1[0] to P1[7]			I/O with internal pull-up	**Port 1**: Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 1 pins are pulled HIGH by the internal pull-ups when 1s are written to them and can be used as inputs in this state. As inputs, Port 1 pins that are externally pulled LOW will source current (I_{IL}) because of the internal pull-ups. P1[5], P1[6], P1[7] have high current drive of 16 mA.
P1[0]/T2	2	40	I/O	**P1[0] —** Port 1 bit 0.
			I/O	**T2 —** External count input to timer/counter 2 or clock-out from timer/counter 2.
P1[1]/T2EX	3	41	I/O	**P1[1] —** Port 1 bit 1.
			I	**T2EX**: Timer/counter 2 capture/reload trigger and direction control.
P1[2]/ECI	4	42	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	5	43	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]/CEX1/ $\overline{SS}$	6	44	I/O	**P1[4] —** Port 1 bit 4.
			I/O	**CEX1 —** Capture/compare external I/O for PCA Module 1.
			I	**$\overline{SS}$ —** Slave Select input for SPI.
P1[5]/CEX2/ MOSI	7	1	I/O	**P1[5] —** Port 1 bit 5.
			I/O	**CEX2 —** Capture/compare external I/O for PCA Module 2.
			I/O	**MOSI —** Master output/slave input for SPI.
P1[6]/CEX3/ MISO	8	2	I/O	**P1[6] —** Port 1 bit 6.
			I/O	**CEX3 —** Capture/compare external I/O for PCA Module 3.
			I/O	**MISO —** Master input/slave output for SPI.
P1[7]/CEX4/ SPICLK	9	3	I/O	**P1[7] —** Port 1 bit 7.
			I/O	**CEX4 —** Capture/compare external I/O for PCA Module 4.
			I/O	**SPICLK —** Serial clock input/output for SPI.
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 1s 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 1s.
P2[0]/A8	24	18	I/O	**P2[0] —** Port 2 bit 0.
			O	**A8 —** Address bit 8.

Table 3. P89CV51RB2/RC2/RD2 Pin description *...continued*

Symbol	Pin		Type	Description
	PLCC44	TQFP44		
P2[1]/A9	25	19	I/O	**P2[1] —** Port 2 bit 1.
			O	**A9 —** Address bit 9.
P2[2]/A10	26	20	I/O	**P2[2] —** Port 2 bit 2.
			O	**A10 —** Address bit 10.
P2[3]/A11	27	21	I/O	**P2[3] —** Port 2 bit 3.
			O	**A11 —** Address bit 11.
P2[4]/A12	28	22	I/O	**P2[4] —** Port 2 bit 4.
			O	**A12 —** Address bit 12.
P2[5]/A13	29	23	I/O	**P2[5] —** Port 2 bit 5.
			O	**A13 —** Address bit 13.
P2[6]/A14	30	24	I/O	**P2[6] —** Port 2 bit 6.
			O	**A14 —** Address bit 14.
P2[7]/A15	31	25	I/O	**P2[7] —** Port 2 bit 7.
			O	**A15 —** Address bit 15.
P3[0] to P3[7]			I/O with internal pull-up	**Port 3**: Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins are pulled HIGH by the internal pull-ups when 1s are written to them and can be used as inputs in this state. As inputs, Port 3 pins that are externally pulled LOW will source current (I_{IL}) because of the internal pull-ups.
P3[0]/RXD	11	5	I/O	**P3[0] —** Port 3 bit 0.
			I	**RXD —** Serial input port.
P3[1]/TXD	13	7	I/O	**P3[1] —** Port 3 bit 1.
			O	**TXD —** Serial output port.
P3[2]/$\overline{\text{INT0}}$	14	8	I/O	**P3[2] —** Port 3 bit 2.
			I	**$\overline{\text{INT0}}$ —** External interrupt 0 input.
P3[3]/$\overline{\text{INT1}}$	15	9	I/O	**P3[3] —** Port 3 bit 3.
			I	**$\overline{\text{INT1}}$ —** External interrupt 1 input.
P3[4]/T0	16	10	I/O	**P3[4] —** Port 3 bit 4.
			I	**T0 —** External count input to timer/counter 0.
P3[5]/T1	17	11	I/O	**P3[5] —** Port 3 bit 5.
			I	**T1 —** External count input to timer/counter 1.
P3[6]/$\overline{\text{WR}}$	18	12	I/O	**P3[6] —** Port 3 bit 6.
			O	**$\overline{\text{WR}}$ —** External data memory write strobe.
P3[7]/$\overline{\text{RD}}$	19	13	I/O	**P3[7] —** Port 3 bit 7.
			O	**$\overline{\text{RD}}$ —** External data memory read strobe.
$\overline{\text{PSEN}}$	32	26	O	**Program Store Enable**: $\overline{\text{PSEN}}$ is the read strobe for external program memory. When the device is executing from internal program memory, $\overline{\text{PSEN}}$ is inactive (HIGH). When the device is executing code from 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.

Table 3. P89CV51RB2/RC2/RD2 Pin description *...continued*

Symbol	Pin		Type	Description
	PLCC44	TQFP44		
RST	10	4	I	**Reset**: While the oscillator is running, a HIGH logic state on this pin for two machine cycles will reset the device.
$\overline{EA}$	35	29	I	**External Access Enable**: $\overline{EA}$ must be connected to V_{SS} in order to enable the device to fetch code from the external program memory. $\overline{EA}$ must be strapped to V_{DD} for internal program execution.
ALE	33	27	I/O	**Address Latch Enable**: ALE is the output signal for latching the low byte of the address during an access to external memory. Normally the ALE[1] is emitted at a constant rate of $\frac{1}{6}$ the crystal frequency[2] and can be used for external timing and clocking. One ALE pulse is skipped during each access to external data memory. However, if bit AO is set to 1, ALE is disabled.
XTAL1	21	15	I	**Crystal 1**: Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL2	20	14	O	**Crystal 2**: Output from the inverting oscillator amplifier.
V_{DD}	44	38	supply	**Power supply**
V_{SS}	22	16	supply	**Ground**

[1] ALE loading issue: When ALE pin experiences higher loading (> 30 pF) during the reset, the microcontroller may accidentally enter into modes other than normal working mode. The solution is to add a pull-up resistor from 3 kΩ to 50 kΩ between e.g., ALE pin and V_{DD}.

[2] For 6-clock mode, ALE is emitted at $\frac{1}{3}$ of crystal frequency.

6. Functional description

6.1 Special function registers

Remark: SFR accesses are restricted in the following ways:

- Do **not** attempt to access any SFR locations that are undefined.
- Access to defined SFR locations must be strictly for the functions of the SFRs.
- SFR bits labeled '-', '0' or '1' can **only** be written and read as follows:
  - '-' Unless otherwise specified, **must** be written with '0', but can return any value when read (even if it was written with '0'). It is a reserved bit and may be used in future derivatives.
  - '0' **must** be written with '0', and will return a '0' when read.
  - '1' **must** be written with '1', and will return a '1' when read.

Table 4. Special function registers

** indicates SFRs that are bit addressable.*

Name	Description	SFR address	Bit functions and addresses[1]					
			MSB					
		Bit address	E7	E6	E5	E4	E3	E2
ACC*	Accumulator	E0H	-	-	-	-	-	-
AUXR	Auxiliary function Register	8EH	-	-	-	-	-	-
AUXR1	Auxiliary function Register 1	A2H	-	-	ENBOOT	-	GF2	0
		Bit address	F7	F6	F5	F4	F3	F2
B*	B register	F0H	-	-	-	-	-	-
CCAP0H	Module 0 Capture High	FAH	-	-	-	-	-	-
CCAP1H	Module 1 Capture High	FBH	-	-	-	-	-	-
CCAP2H	Module 2 Capture High	FCH	-	-	-	-	-	-
CCAP3H	Module 3 Capture High	FDH	-	-	-	-	-	-
CCAP4H	Module 4 Capture High	FEH	-	-	-	-	-	-
CCAP0L	Module 0 Capture Low	EAH	-	-	-	-	-	-
CCAP1L	Module 1 Capture Low	EBH	-	-	-	-	-	-
CCAP2L	Module 2 Capture Low	ECH	-	-	-	-	-	-
CCAP3L	Module 3 Capture Low	EDH	-	-	-	-	-	-
CCAP4L	Module 4 Capture Low	EEH	-	-	-	-	-	-
CCAPM0	Module 0 mode	DAH	-	ECOM_0	CAPP_0	CAPN_0	MAT_0	TOG
CCAPM1	Module 1 mode	DBH	-	ECOM_1	CAPP_1	CAPN_1	MAT_1	TOG
CCAPM2	Module 2 mode	DCH	-	ECOM_2	CAPP_2	CAPN_2	MAT_2	TOG
CCAPM3	Module 3 mode	DDH	-	ECOM_3	CAPP_3	CAPN_3	MAT_3	TOG
CCAPM4	Module 4 mode	DEH	-	ECOM_4	CAPP_4	CAPN_4	MAT_4	TOG
		Bit address	DF	DE	DD	DC	DB	DA
CCON*	PCA Counter Control	D8H	CF	CR	-	CCF4	CCF3	CC
CH	PCA Counter High	F9H	-	-	-	-	-	-
CL	PCA Counter Low	E9H	-	-	-	-	-	-
CMOD	PCA Counter Mode	D9H	CIDL	WDTE	-	-	-	CPS
CKCON	Clock Control	8FH	SPIX2	WDX2	PCAX2	SIX2	T2X2	T1X

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

** indicates SFRs that are bit addressable.*

Name	Description	SFR address	Bit functions and addresses[1]					
			MSB					
DPTR	Data Pointer (2 B)							
DPH	Data Pointer High	83H	-	-	-	-	-	-
DPL	Data Pointer Low	82H	-	-	-	-	-	-
		Bit address	**AF**	**AE**	**AD**	**AC**	**AB**	**A**
IE*	Interrupt Enable	A8H	EA	EC	ET2	ES	ET1	EX
		Bit address	**BF**	**BE**	**BD**	**BC**	**BB**	**B**
IP*	Interrupt Priority Low	B8H	-	PPC	PT2	PS	PT1	PX
IPH	Interrupt Priority High	B7H	-	PPCH	PT2H	PSH	PT1H	PX
		Bit address	**87**	**86**	**85**	**84**	**83**	**8**
P0*	Port 0	80H	AD7	AD6	AD5	AD4	AD3	AD
		Bit address	**97**	**96**	**95**	**94**	**93**	**9**
P1*	Port 1	90H	CEX4/ SPICLK	CEX3/ MISO	CEX2/ MOSI	CEX1/$\overline{SS}$	CEX0	EC
		Bit address	**A7**	**A6**	**A5**	**A4**	**A3**	**A**
P2*	Port 2	A0H	A15	A14	A13	A12	A11	A1
		Bit address	**B7**	**B6**	**B5**	**B4**	**B3**	**B**
P3*	Port 3	B0H	$\overline{RD}$	$\overline{WR}$	T1	T0	$\overline{INT1}$	$\overline{IN}$
PCON	Power Control	87H	SMOD1	SMOD0	-	POF	GF1	GF
		Bit address	**D7**	**D6**	**D5**	**D4**	**D3**	**D**
PSW*	Program Status Word	D0H	CY	AC	F0	RS1	RS0	OV
RCAP2H	Timer 2 Capture High	CBH	-	-	-	-	-	-
RCAP2L	Timer 2 Capture Low	CAH	-	-	-	-	-	-
		Bit address	**9F**	**9E**	**9D**	**9C**	**9B**	**9**
SCON*	Serial port Control	98H	SM0/FE	SM1	SM2	REN	TB8	RB
SBUF	Serial port data Buffer	99H	-	-	-	-	-	-
SADDR	Serial port Address	A9H	-	-	-	-	-	-
SADEN	Serial port Address Enable	B9H	-	-	-	-	-	-

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

** indicates SFRs that are bit addressable.*

Name	Description	SFR address	Bit functions and addresses[1]					
			MSB					
		Bit address	**87**	**86**	**85**	**84**	**83**	**82**
SPCR	SPI Control Register	D5H	SPIE	SPEN	DORD	MSTR	CPOL	CPH
SPSR	SPI Status Register	AAH	SPIF	WCOL	-	-	-	-
SPDAT	SPI Data	86H	-	-	-	-	-	-
SP	Stack Pointer	81H	-	-	-	-	-	-
		Bit address	**8F**	**8E**	**8D**	**8C**	**8B**	**8A**
TCON*	Timer/counter Control	88H	TF1	TR1	TF0	TR0	IE1	IT
		Bit address	**CF**	**CE**	**CD**	**CC**	**CB**	**CA**
T2CON*	Timer/counter 2 Control	C8H	TF2	EXF2	RCLK	TCLK	EXEN2	TR
T2MOD	Timer 2 Mode control	C9H	-	-	-	-	-	-
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/counter 0 and 1 Mode	89H	T1GATE	T1C/$\overline{T}$	T1M1	T1M0	T0GATE	T0C
WDTRST	WatchDog Timer Reset	A6H	-	-	-	-	-	-

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

6.2 Memory organization

The various P89CV51RB2/RC2/RD2 memory spaces are as follows:

- DATA

  128 B of internal data memory space (00H:7FH) accessed via direct or indirect addressing, using instructions other than MOVX and MOVC. All or part of the stack may be in this area.

- IDATA

  Indirect Data. 256 B of internal data memory space (00H:FFH) accessed via indirect addressing using instructions other than MOVX and MOVC. All or part of the stack may be in this area. This area includes the DATA area and the 128 B immediately above it.

- SFR

  Special Function Registers. Selected CPU registers and peripheral control and status registers, accessible only via direct addressing.

- XDATA

  'External' Data or auxiliary RAM. Duplicates the classic 80C51 64 kB memory space addressed via the MOVX instruction using the DPTR, R0, or R1. The P89CV51RB2/RC2/RD2 have 768 B of on-chip XDATA memory.

- CODE

  64 kB of code memory space, accessed as part of program execution and via the MOVC instruction. The P89CV51RB2/RC2/RD2 have 16/32/64 kB of on-chip code memory.

6.2.1 Expanded data RAM addressing

The P89CV51RB2/RC2/RD2 have 1 kB of data RAM; see Figure 4.

To access the expanded RAM (XRAM), the EXTRAM bit must be set and MOVX instructions must be used. The expanded memory is physically located on the chip and logically occupies the first bytes of external memory (addresses 000H to 2FFH).

Table 5. AUXR - Auxiliary function register (address 8EH) bit allocation

Not bit addressable; reset value 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	EXTRAM	AO

When EXTRAM = 0, the expanded RAM is indirectly addressed using the MOVX instruction in combination with any of the registers R0, R1 of the selected bank or DPTR. Accessing the expanded RAM does not affect ports P0, P3[6] ($\overline{WR}$), P3[7] ($\overline{RD}$), or P2. With EXTRAM = 0, the expanded RAM can be accessed as in the following example.

Expanded RAM access (indirect addressing only):

```
MOVX @DPTR, A; DPTR contains 0A0H
```

The DPTR points to location 0A0H and the data in the accumulator is written to address 0A0H of the expanded RAM rather than off-chip external memory. Access to EXTRAM addresses that are not present on the device (above 2FFH) will access external off-chip memory and will perform in the same way as the standard 8051, with P0 and P2 as data/address bus, and P3[6] and P3[7] as write and read timing signals.

Table 6. AUXR - Auxiliary function register (address 8EH) bit description

Bit	Symbol	Description
7 to 2	-	Reserved for future use. Should be set to 0 by user programs.
1	EXTRAM	Internal/external RAM access using MOVX @Ri/@DPTR. When 0, accesses internal XRAM with address specified in MOVX instruction. If address supplied with this instruction exceeds on-chip available XRAM, off-chip RAM is accessed. When 1, every MOVX instruction targets external data memory by default.
0	AO	ALE off: disables/enables ALE. AO = 0 results in ALE emitted at a constant rate of $\frac{1}{2}$ the oscillator frequency. In case of AO = 1, ALE is active only during a MOVX or MOVC.

When EXTRAM = 1, MOVX @Ri and MOVX @DPTR will be similar to the standard 8051. Using MOVX @Ri provides an 8-bit address with multiplexed data on Port 0. Other output port pins can be used to output higher order address bits. This provides external paging capabilities. Using MOVX @DPTR generates a 16-bit address. This allows external addressing up to 64 kB. Port 2 provides the high-order eight address bits (DPH), and Port 0 multiplexes the low-order eight address bits (DPL) with data. Both MOVX @Ri and MOVX @DPTR generates the necessary read and write signals (P3[6] - $\overline{WR}$ and P3[7] - $\overline{RD}$) for external memory use. Table 7 shows external data memory $\overline{RD}$, $\overline{WR}$ operation with EXTRAM bit.

The stack pointer (SP) can be located anywhere within the 256 B of internal RAM (lower 128 B and upper 128 B). The stack pointer may not be located in any part of the expanded RAM.

Table 7. External data memory $\overline{RD}$, $\overline{WR}$ with EXTRAM bit

AUXR	MOVX @DPTR, A or MOVX A, @DPTR		MOVX @Ri, A or MOVX A, @Ri
	ADDR < 0300H	ADDR ≥ 0300H	ADDR = any
EXTRAM = 1	$\overline{RD}$/$\overline{WR}$ asserted	$\overline{RD}$/$\overline{WR}$ asserted	$\overline{RD}$/$\overline{WR}$ asserted
EXTRAM = 0	$\overline{RD}$/$\overline{WR}$ not asserted	$\overline{RD}$/$\overline{WR}$ asserted	$\overline{RD}$/$\overline{WR}$ not asserted

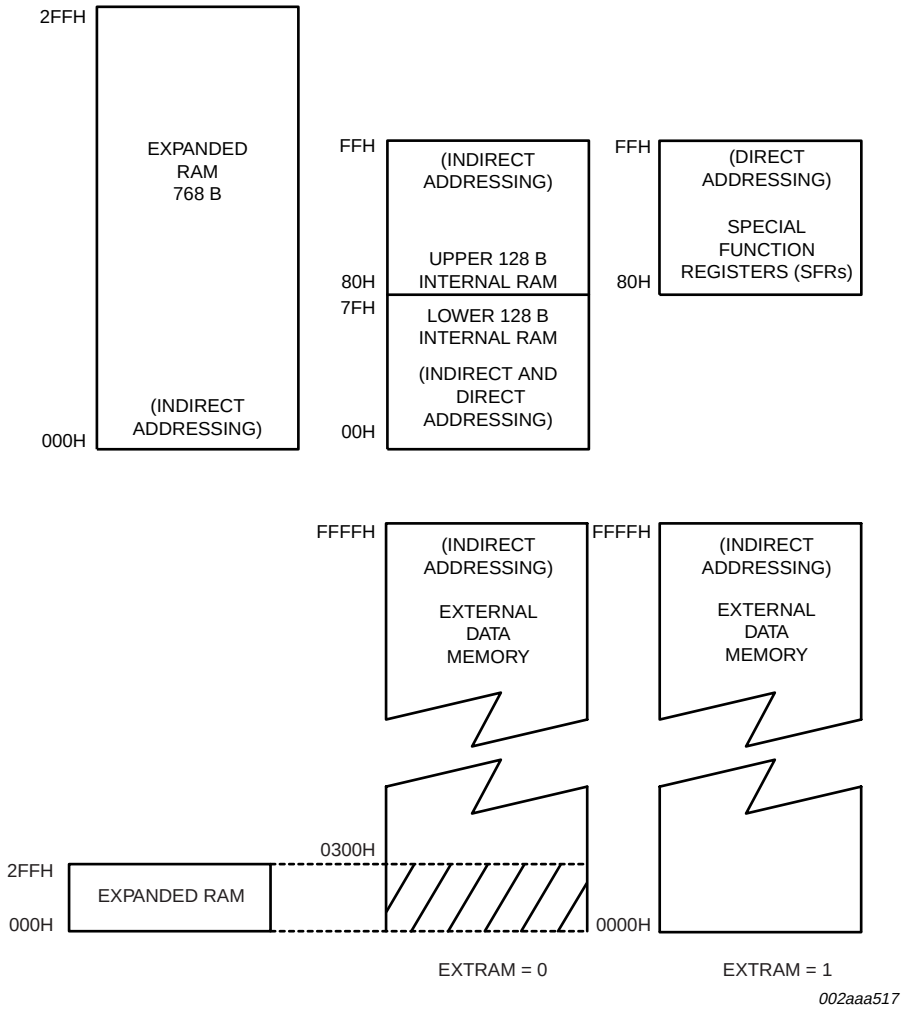


Fig 4. Internal and external data memory structure

6.2.2 Dual data pointers

The device has two 16-bit data pointers. The DPTR Select (DPS) bit in AUXR1 determines which of the two data pointers is accessed. When DPS = 0, DPTR0 is selected; when DPS = 1, DPTR1 is selected. Quickly switching between the two data pointers can be accomplished by a single INC instruction on AUXR1; see Figure 5.

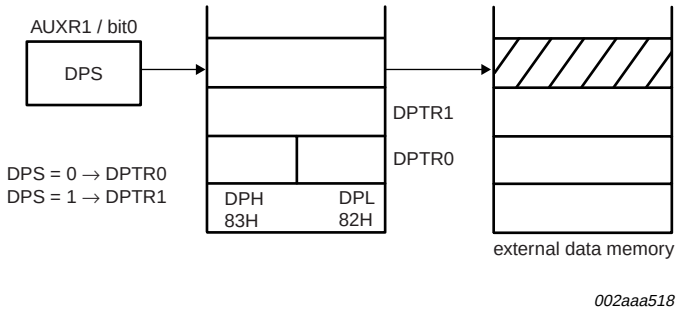


Fig 5. Dual data pointer organization

Table 8. AUXR1 - Auxiliary function register 1 (address A2H) bit allocation

Not bit addressable; reset value 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	-	-	ENBOOT	-	GF2	0	-	DPS

Table 9. AUXR1 - Auxiliary function register 1 (address A2H) bit description

Bit	Symbol	Description
7, 6, 4	-	Reserved for future use. Should be set to 0 by user programs.
5	ENBOOT	Enable BOOTROM
3	GF2	General purpose user-defined Flag.
2	0	This bit contains a hard-wired 0. Allows toggling of the DPS bit by incrementing AUXR1, without interfering with other bits in the register.
1	-	Reserved for future use. Should be set to 0 by user programs.
0	DPS	Data Pointer Select. Chooses one of two data pointers for use by the program. See text for details.

6.2.3 Reset

At initial power-up, the port pins will be in a random state until the oscillator has started and the internal reset algorithm has weakly pulled all pins HIGH. Powering up the device without a valid reset could cause the MCU to start executing instructions from an indeterminate location. Such undefined states may inadvertently corrupt the code in the flash. A system reset will not affect the on-chip RAM while the device is running, however, the contents of the on-chip RAM during power-up are indeterminate.

When power is applied to the device, the RST pin must be held HIGH long enough for the oscillator to start-up (usually several milliseconds for a low frequency crystal), in addition to two machine cycles for a valid power-on reset. An example of a method to extend the RST signal is to implement an RC circuit by connecting the RST pin to V_{DD} through a 10 µF capacitor and to V_{SS} through an 8.2 kΩ resistor as shown in Figure 6.

During initial power-up the POF flag in the PCON register is set to indicate an initial power-up condition. The POF flag will remain active until cleared by software.

Following a reset condition, under normal conditions, the MCU will start executing code from address 0000H in the user's code memory. However if either the $\overline{PSEN}$ pin was LOW when reset was exited, or the status bit = 1, the MCU will start executing code from the boot address. The boot address is formed using the value of the boot vector as the high byte of the address and 00H as the low byte.

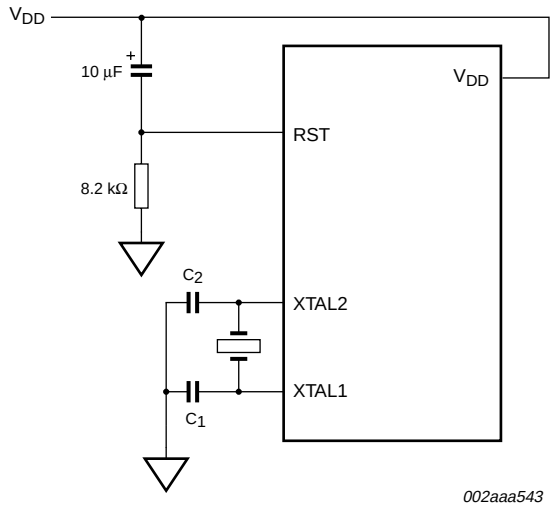


Fig 6. Power-on reset circuit

6.3 Flash memory

6.3.1 Flash organization

The P89CV51RB2/RC2/RD2 program memory consists of a 16/32/64 kB block for user code. The flash can be read or written in bytes and can be erased in 128-B pages. A chip erase function will erase the entire user code memory and its associated security bits. There are three methods for erasing or programming the flash memory that may be used. First, the flash may be programmed or erased in the end-user application by calling LOW-state routines through a common IAP entry point. Second, the on-chip ISP bootloader may be invoked. This ISP bootloader will, in turn, call LOW-state routines through the same common entry point that can be used by the end-user application. Third, the flash may be programmed or erased using the parallel method by using a commercially available EPROM programmer which supports this device.

6.3.2 Features

- Flash internal program memory with 128-B page erase.
- Internal boot block, containing LOW-state IAP routines available to user code.
- Boot vector allows user-provided flash loader code to reside anywhere in the flash memory space, providing flexibility to the user.
- Default loader providing ISP via the serial port, located in upper-end of program memory.
- Programming and erase over the full operating voltage range.
- Read/Programming/Erase using ISP/IAP.

- Programming with industry-standard commercial programmers.
- 10000 typical erase/program cycles for each byte.
- 100 year minimum data retention.

6.3.3 Boot block

When the microcontroller programs its own flash memory, all of the low-level details are handled by code (bootloader) that is contained in a boot block. A user program calls the common entry point in the boot block with appropriate parameters to accomplish the desired operation. Boot block operations include erase user code, program user code, program security bits, chip erase, etc. The boot block logically overlays the program memory space from FC00H to FFFFH, when it is enabled. The boot block may be disabled on-the-fly so that the upper 1 kB of user code is available to the user's program.

6.3.4 Power-on reset code execution

The P89CV51RB2/RC2/RD2 contains two special flash elements: the boot vector and the boot status bit. Following reset, the P89CV51RB2/RC2/RD2 examines the contents of the boot status bit. If the boot status bit is set to zero, power-up execution starts at location 0000H, which is the normal start address of the user's application code. When the boot status bit is set to a value other than zero, the contents of the boot vector are used as the high byte of the execution address and the low byte is set to 00H.

Table 10 shows the factory default boot vector setting for this device. A factory-provided bootloader is pre-programmed into the address space indicated and uses the indicated bootloader entry point to perform ISP functions.

Table 10. Default boot vector values and ISP entry points

Device	Default boot vector	Default bootloader entry point	Default bootloader code range
P89CV51RB2/RC2/RD2	FCH	FC00H	FC00H to FFFFH

6.3.5 Hardware activation of the bootloader

The bootloader can also be executed by forcing the device into ISP mode during a power-on sequence. This has the same effect as having a non-zero status byte. This allows an application to be built that will normally execute user code but can be manually forced into ISP operation. This occurs by holding $\overline{\text{PSEN}}$ LOW at the falling edge of reset. If the factory default setting for the boot vector (FCH) is changed, it will no longer point to the factory pre-programmed ISP bootloader code. After programming the flash, the status byte should be programmed to zero in order to allow execution of the user's application code beginning at address 0000H.

6.3.6 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 P89CV51RB2/RC2/RD2 through the serial port. This firmware is provided by NXP and embedded within each P89CV51RB2/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.

6.3.7 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 P89CV51RB2/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 P89CV51RB2/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 11. As a record is received by the P89CV51RB2/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 P89CV51RB2/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 11. ISP Hex record formats

Record type	Command/data function
00	Program user code memory :nnaaaa00dd..ddcc Where: nn = number of bytes to program aaaa = address dd..dd = data bytes cc = checksum Example: :0900000001020304050607080 9CA
01	End of File (EOF), no operation :xxxxxx01cc Where: xxxxxx = required field but value is a don’t care cc = checksum Example: :00000001FF
02	not used

Table 11. ISP Hex record formats *…continued*

Record type	Command/data function
03	Miscellaneous write functions :nnxxxx03ffssddcc Where: nn = number of bytes in the record xxxx = required field but value is a don't care ff = subfunction code ss = selection code dd = data (if needed) cc = checksum Subfunction code = 0C (erase 4 kB blocks) ff = 0C ss = block code, as shown below: block 0, 0 kB to 4 kB, 00H block 1, 4 kB to 8 kB, 10H block 2, 8 kB to 12 kB, 20H block 3, 12 kB to 16 kB, 30H block 4, 16 kB to 20 kB, 40H (only available on P89CV51RC2/RD2) block 5, 20 kB to 24 kB, 50H (only available on P89CV51RC2/RD2) block 6, 24 kB to 28 kB, 60H (only available on P89CV51RC2/RD2) block 7, 28 kB to 32 kB, 70H (only available on P89CV51RC2/RD2) block 8, 32 kB to 36 kB, 80H (only available on P89CV51RD2) block 9, 36 kB to 40 kB, 90H (only available on P89CV51RD2) block 10, 40 kB to 44 kB, A0H (only available on P89CV51RD2) block 11, 44 kB to 48 kB, B0H (only available on P89CV51RD2) block 12, 48 kB to 52 kB, C0H (only available on P89CV51RD2) block 13, 52 kB to 56 kB, D0H (only available on P89CV51RD2) block 14, 56 kB to 60 kB, E0H (only available on P89CV51RD2) block 15, 60 kB to 64 kB, F0H (only available on P89CV51RD2) Example: :020000030C20CF (erase 4 kB block #2)

Table 11. ISP Hex record formats *…continued*

Record type	Command/data function
03 (continued)	Subfunction code = 01 (erase blocks) ff = 01 ss = block code, as shown below block 0, 0 kB to 8 kB, 00H block 1, 8 kB to 16 kB, 20H block 2, 16 kB to 32 kB, 40H block 3, 32 kB to 48 kB, 80H block 4, 48 kB to 64 kB, C0H Subfunction code = 04 (erase boot vector and status bit) ff = 04 ss = don't care Subfunction code = 05 (program security bits) ff = 05 ss = 00 program security bit 1 ss = 01 program security bit 2 ss = 02 program security bit 3 Subfunction code = 06 (program status bit, boot vector, 6×/12× bit) ff = 06 dd = data (for boot vector) ss = 00 program status bit ss = 01 program boot vector ss = 02 program 6×/12× bit Subfunction code = 07 (chip erase) Erases code memory and security bits, programs default boot vector and status bit ff = 07 Subfunction code = 08 (erase page, 128 B) ff = 08 ss = high byte of page address (A[15:8]) dd = low byte of page address (A[7:0]) Example: :0300000308E000F2 (erase page at E000H)

Table 11. ISP Hex record formats *…continued*

Record type	Command/data function
04	Display device data or blank check :05xxxx04sssseeeeffcc Where 05 = number of bytes in the record xxxx = required field but value is a don't care 04 = function code for display or blank check ssss = starting address, MSB first eeee = ending address, MSB first ff = subfunction 00 = display data 01 = blank check cc = checksum Subfunction codes: Example: :0500000400001FFF00D9 (display from 0000H to 1FFFH)
05	Miscellaneous read functions :02xxxx05ffsscc Where: 02 = number of bytes in the record xxxx = required field but value is a don't care 05 = function code for miscellaneous read ffss = subfunction and selection code 0000 = read manufacturer ID 0001 = read device ID 1 0002 = read device ID 2 0003 = read 6×/12× bit (bit 7 = 1 is 6×, bit 7 = 0 is 12×) 0080 = read boot code version 0700 = read security bits 0701 = read status bit 0702 = read boot vector cc = checksum Example: :020000050000F9 (display manufacturer ID)
06	Direct load of baud rate :02xxxx06HHLLcc Where: 02 = number of bytes in the record xxxx = required field but value is a don't care HH = high byte of timer T2 LL = low byte of timer T2 cc = checksum Example: :02000006FFFFcc (load T2 = FFFF)

6.3.8 IAP method

Several IAP calls are available for use by an application program to permit selective erasing, reading and programming of flash pages, security bits, status bit, and device ID. All calls are made through a common interface, PGM_MTP. The programming functions are selected by setting up the microcontroller's registers before making a call to PGM_MTP at FFF0H. The IAP calls are shown in Table 12.

Table 12. IAP function calls

IAP function	IAP call parameters
Read ID	**Input parameters:** R1 = 00H or 80H (WDT feed) DPH = 00H DPL = 00H = manufacturer ID DPL = 01H = device ID 1 DPL = 02H = device ID 2 DPL = 03H = 6×/12× bit (if bit 7 = 1: 6×) DPL = 80H = ISP version number **Return parameter(s):** ACC = requested parameter
Erase 4 kB code block (new function)	**Input parameters:** R0 = oscillator frequency (integer) R1 = 0CH or 8CH (WDT feed) DPH = address of 4 kB code block DPH = 00H, 4 kB block 0, 0 kB to 4 kB DPH = 10H, 4 kB block 1, 4 kB to 8 kB DPH = 20H, 4 kB block 2, 8 kB to 12 kB DPH = 30H, 4 kB block 3, 12 kB to 16 kB DPH = 40H, 4 kB block 4, 16 kB to 20 kB DPH = 50H, 4 kB block 5, 20 kB to 24 kB DPH = 60H, 4 kB block 6, 24 kB to 28 kB DPH = 70H, 4 kB block 7, 28 kB to 32 kB DPH = 80H, 4 kB block 8, 32 kB to 36 kB DPH = 90H, 4 kB block 9, 36 kB to 40 kB DPH = A0H, 4 kB block 10, 40 kB to 44 kB DPH = B0H, 4 kB block 11, 44 kB to 48 kB DPH = C0H, 4 kB block 12, 48 kB to 52 kB DPH = D0H, 4 kB block 13, 52 kB to 56 kB DPH = E0H, 4 kB block 14, 56 kB to 60 kB DPH = F0H, 4 kB block 15, 60 kB to 64 kB DPL = 00H **Return parameter(s):** ACC = 00: pass ACC is not 00: fail

Table 12. IAP function calls *…continued*

IAP function	IAP call parameters
Erase 8 kB/16 kB code block	**Input parameters:** R1 = 01H or 81H (WDT feed) DPH = 00H, block 0, 0 kB to 8 kB DPH = 20H, block 1, 8 kB to 16 kB DPH = 40H, block 2, 16 kB to 32 kB DPH = 80H, block 3, 32 kB to 48 kB DPH = C0H, block 4, 48 kB to 64 kB **Return parameter(s):** ACC = 00: pass ACC is not 00: fail
Program user code	**Input parameters:** R1 = 02H or 82H (WDT feed) DPH = memory address MSB DPL = memory address LSB ACC = byte to program **Return parameter(s):** ACC = 00: pass ACC is not 00: fail
Read user code	**Input parameters:** R1 = 03H or 83H (WDT feed) DPH = memory address MSB DPL = memory address LSB **Return parameter(s):** ACC = device data
Erase status bit and boot vector	**Input parameters:** R1 = 04H or 84H (WDT feed) DPL = don’t care DPH = don’t care **Return parameter(s):** ACC = 00: pass ACC is not 00: fail
Program security bits	**Input parameters:** R1 = 05H or 85H (WDT feed) DPL = 00H = security bit 1 DPL = 01H = security bit 2 DPL = 02H = security bit 3 **Return parameter(s):** ACC = 00: pass ACC is not 00: fail

Table 12. IAP function calls *…continued*

IAP function	IAP call parameters
Program status bit, boot vector, 6×/12× bit	**Input parameters:** R1 = 06H or 86H (WDT feed) DPL = 00H = program status bit DPL = 01H = program boot vector DPL = 02H = 6×/12× bit ACC = boot vector value to program **Return parameter(s):** ACC = 00: pass ACC is not 00: fail
Read security bits, status bit, boot vector	**Input parameters:** ACC = 07H or 87H (WDT feed) DPL = 00H = security bits DPL = 01H = status bit DPL = 02H = boot vector **Return parameter(s):** ACC = 00 SoftICE S/N-match 0 SB 0 DBL_CLK
Erase page	**Input parameters:** R1 = 08H or 88H (WDT feed) DPH = page address high byte DPL = page address low byte **Return parameter(s):** ACC = 00: pass ACC is not 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 13 and Table 14).

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 a 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 selectable operating modes.

The ‘Timer’ or ‘Counter’ function is selected by control bits C/$\overline{T}$ in the special function register TMOD. These two timers/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 13. 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 14. TMOD - Timer/Counter mode control register (address 89H) bit description

Bit	Symbol	Description
7	T1GATE	Gating control for Timer 1. When set, timer/counter is enabled only while the $\overline{INT1}$ pin is HIGH and the TR1 control bit is set. When cleared, Timer 1 is enabled when the TR1 control bit is set.
6	T1C/$\overline{T}$	Timer or counter select for Timer 1. Cleared for timer operation. Set for counter operation (input from T1 input pin).
5	T1M1	Mode select for Timer 1.
4	T1M0	
3	T0GATE	Gating control for Timer 0. When set, timer/counter is enabled only while the $\overline{INT0}$ pin is HIGH and the TR0 control bit is set. When cleared, Timer 0 is enabled when the TR0 control bit is set.
2	T0C/$\overline{T}$	Timer or counter select for Timer 0. Cleared for timer operation. Set for counter operation (input from T0 input pin).
1	T0M1	Mode select for Timer 0.
0	T0M0	

Table 15. 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 16. 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 17. 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-state is detected. Cleared by hardware when the interrupt is processed, or by software.
2	IT1	Interrupt 1 Type control bit. Set/cleared by software to specify falling edge/LOW-state that triggers external interrupt 1.
1	IE0	Interrupt 0 Edge flag. Set by hardware when external interrupt 0 edge/LOW-state is detected. Cleared by hardware when the interrupt is processed, or by software.
0	IT0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/LOW-state that triggers external interrupt 0.

6.4.1 Mode 0

Putting either timer into Mode 0 makes it look like an 8048 timer, which is an 8-bit counter with a fixed divide-by-32 prescaler. Figure 7 shows Mode 0 operation.

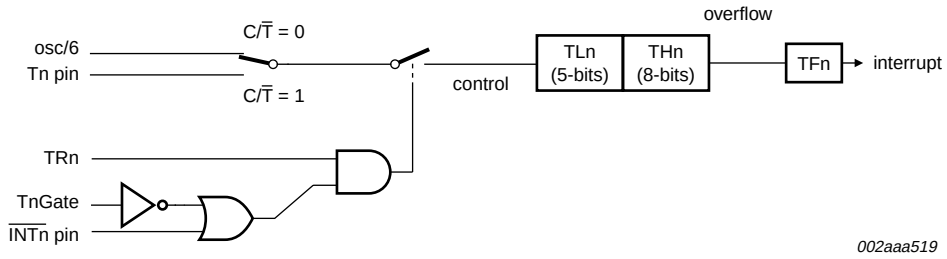


Fig 7. Timer/counter 0 or 1 in Mode 0 (13-bit counter)

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 TFn. The count input is enabled to the timer when TRn = 1 and either GATE = 0 or $\overline{INTn}$ = 1. (Setting GATE = 1 allows the timer to be controlled by external input $\overline{INTn}$, to facilitate pulse width measurements). TRn is a control bit in the special function register TCON (Table 17). The GATE bit is in the TMOD register.

The 13-bit register consists of all 8 bits of THn and the lower 5 bits of TLn. The upper 3 bits of TLn are indeterminate and should be ignored. Setting the run flag (TRn) does not clear the registers.

Mode 0 operation is the same for Timer 0 and Timer 1; see Figure 7. There are two different GATE bits, one for Timer 1 (TMOD.7) and one for Timer 0 (TMOD.3).

6.4.2 Mode 1

Mode 1 is the same as Mode 0, except that all 16 bits of the timer register (THn and TLn) are used; see Figure 8.

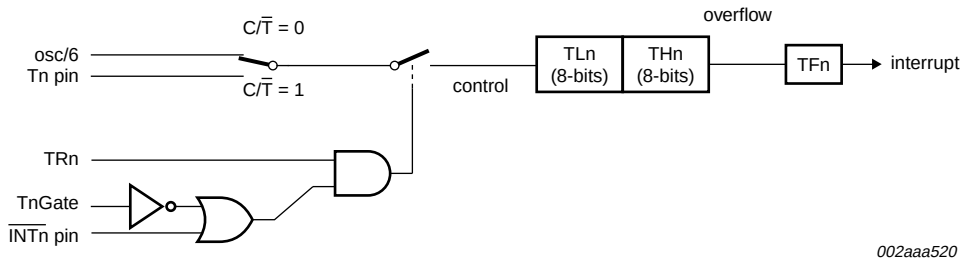


Fig 8. Timer/counter 0 or 1 in Mode 1 (16-bit counter)

6.4.3 Mode 2

Mode 2 configures the timer register as an 8-bit counter (TLn) with automatic reload, as shown in Figure 9. Overflow from TLn not only sets TFn, but also reloads TLn with the contents of THn, which must be preset by software. The reload leaves THn unchanged. Mode 2 operation is the same for Timer 0 and Timer 1.

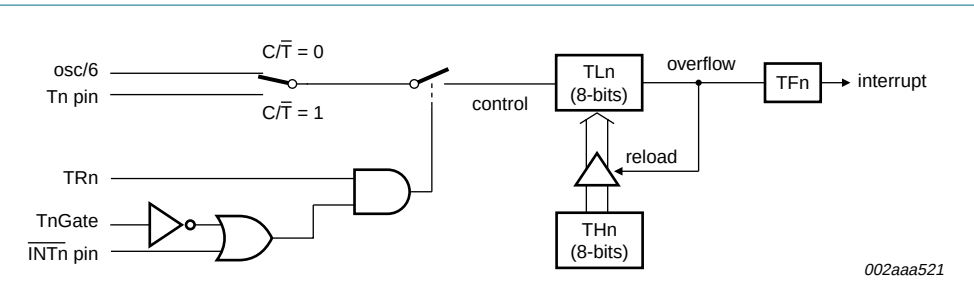


Fig 9. Timer/counter 0 or 1 in Mode 2 (8-bit auto-reload)

6.4.4 Mode 3

When Timer 1 is in Mode 3 it is stopped (holds its count). The effect is the same as setting TR1 = 0.

Timer 0 in Mode 3 establishes TL0 and TH0 as two separate 8-bit counters. The logic for Mode 3 and Timer 0 is shown in Figure 10. TL0 uses the Timer 0 control bits: T0C/$\overline{T}$, T0GATE, TR0, $\overline{INT0}$, and TF0. TH0 is locked into a timer function (counting machine cycles) and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the Timer 1 interrupt.

Mode 3 is provided for applications that require an extra 8-bit timer. With Timer 0 in mode 3, the P89CV51RB2/RC2/RD2 can look like it has an additional timer.

Note: When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it into and out of its own Mode 3. It can still be used by the serial port as a baud rate generator, or in any application not requiring an interrupt.

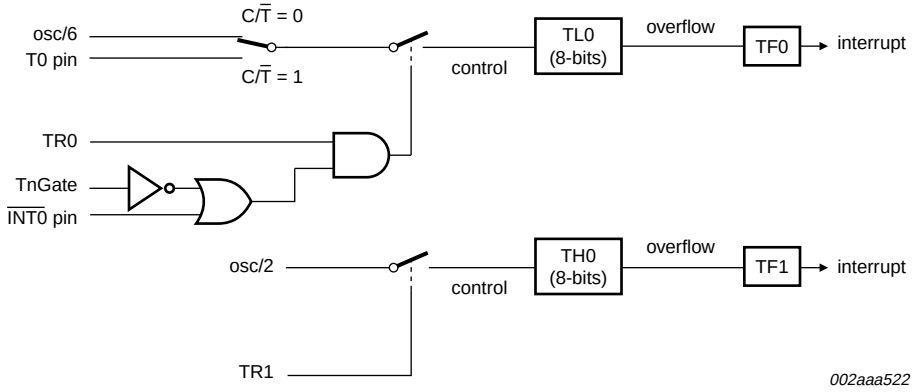


Fig 10. Timer/counter 0 Mode 3 (two 8-bit counters)

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/$\overline{T}$2 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 18 using T2CON (Table 19 and Table 20) and T2MOD (Table 21 and Table 22).

Table 18. Timer 2 operating mode

RCLK + TCLK	CP/$\overline{RL}$2	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 19. 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{T}$2	CP/$\overline{RL}$2

Table 20. 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.

Table 20. T2CON - Timer/Counter 2 control register (address C8H) bit description *...continued*

Bit	Symbol	Description
4	TCLK	Transmit clock flag. When set, causes the UART 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.
3	EXEN2	Timer 2 external enable flag. When set, allows a capture or reload to occur as a result of a 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.
2	TR2	Start/stop control for Timer 2. A logic 1 enables the timer to run.
1	C/$\overline{T}$2	Timer or counter select. (Timer 2) 0 = internal timer (f_{osc} / 6) 1 = external event counter (falling edge triggered; external clock's maximum rate = f_{osc} / 12)
0	CP/$\overline{RL}$2	Capture/Reload flag. When set, captures will occur on negative transitions at T2EX if EXEN2 = 1. When cleared, auto-reloads will 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.

Table 21. T2MOD - Timer 2 mode control register (address C9H) bit allocation

Not bit addressable; reset value: XX00 0000B.

Bit	7	6	5	4	3	2	1	0
Symbol	-	-	-	-	-	-	T2OE	DCEN

Table 22. T2MOD - Timer 2 mode control register (address C9H) bit description

Bit	Symbol	Description
7 to 2	-	Reserved for future use. Should be set to 0 by user programs.
1	T2OE	Timer 2 Output Enable bit. Used in programmable Clock-out mode only.
0	DCEN	Down Count Enable bit. When set, this allows Timer 2 to be configured as an up/down counter.

6.5.1 Capture mode

In the Capture mode there are two options which are selected by bit EXEN2 in T2CON. If EXEN2 = 0 Timer 2 is a 16-bit timer or counter (as selected by C/$\overline{T}$2 in T2CON) which upon overflowing sets bit TF2, the Timer 2 overflow bit. The Capture mode is illustrated in Figure 11.

This bit can be used to generate an interrupt (by enabling the Timer 2 interrupt bit ET2 in the IE register). If EXEN2 = 1, Timer 2 operates as described above, but with the added feature that a 1-to-0 transition at external input T2EX causes the current value in the Timer 2 registers, TL2 and TH2, to be captured into registers RCAP2L and RCAP2H, respectively.

In addition, the transition at T2EX causes bit EXF2 in T2CON to be set, and EXF2 like TF2 can generate an interrupt (which vectors to the same location as Timer 2 overflow interrupt). The Timer 2 interrupt service routine can interrogate TF2 and EXF2 to determine which event caused the interrupt.

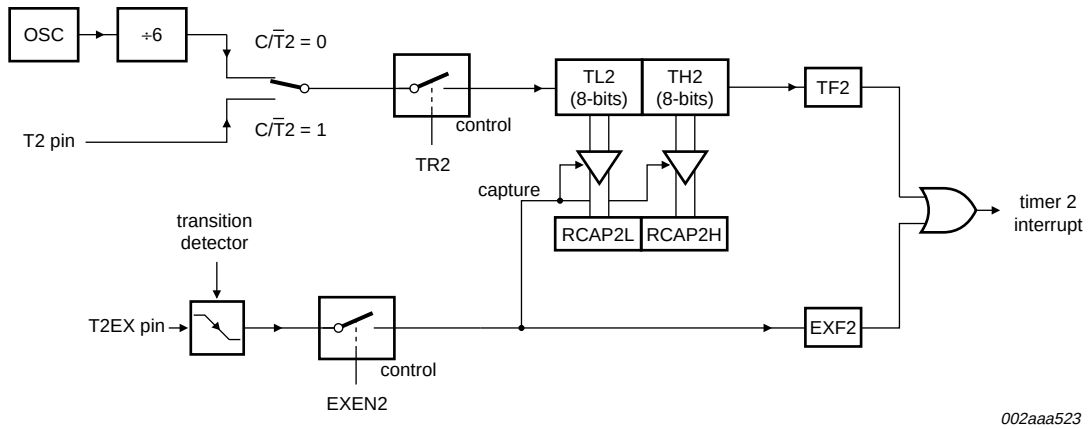


Fig 11. Timer 2 in Capture mode

There is no reload value for TL2 and TH2 in this mode. Even when a capture event occurs from T2EX, the counter keeps on counting T2 pin transitions or f_{osc} / 6 pulses. Since once loaded contents of RCAP2L and RCAP2H registers are not protected, once Timer 2 interrupt is signalled it has to be serviced before a new capture event on T2EX pin occurs. Otherwise, the next falling edge on T2EX pin will initiate reload of the current value from TL2 and TH2 to RCAP2L and RCAP2H and consequently corrupt their content related to the previously reported interrupt.

6.5.2 Auto-reload mode (up or down counter)

In the 16-bit auto-reload mode, Timer 2 can be configured as either a timer or counter (via C/T2 in T2CON), then programmed to count up or down. The counting direction is determined by bit DCEN (Down Counter Enable) which is located in the T2MOD register (see Table 21 and Table 22). When reset is applied, DCEN = 0 and Timer 2 will default to counting up. If the DCEN bit is set, Timer 2 can count up or down depending on the value of the T2EX pin.

Figure 12 shows Timer 2 counting up automatically (DCEN = 0).

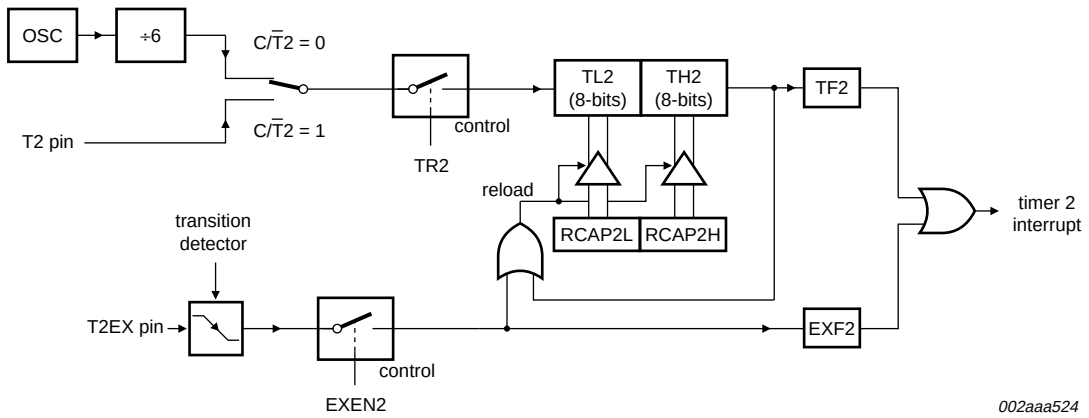


Fig 12. Timer 2 in Auto-reload mode (DCEN = 0)

In this mode, there are two options selected by bit EXEN2 in T2CON register. If EXEN2 = 0, then Timer 2 counts up to 0FFFFH and sets the TF2 (overflow flag) bit upon overflow. This causes the Timer 2 registers to be reloaded with the 16-bit value in RCAP2L and RCAP2H. The values in RCAP2L and RCAP2H are preset by software means.

Auto reload frequency when Timer 2 is counting up can be determined from Equation 1:

$$\frac{SupplyFrequency}{65536 - (RCAP2H, RCAP2L)} \quad (1)$$

Where SupplyFrequency is either f_{osc} ($C/\overline{T}2 = 0$) or frequency of signal on T2 pin ($C/\overline{T}2 = 1$).

If EXEN2 = 1, a 16-bit reload can be triggered either by an overflow or by a 1-to-0 transition at input T2EX. This transition also sets the EXF2 bit. The Timer 2 interrupt, if enabled, can be generated when either TF2 or EXF2 is 1.

The microcontroller's hardware will need three consecutive machine cycles in order to recognize falling edge on T2EX and set EXF2 = 1: in the first machine cycle pin T2EX has to be sampled as 1; in the second machine cycle it has to be sampled as 0, and in the third machine cycle EXF2 will be set to 1.

In Figure 13, DCEN = 1 and Timer 2 is enabled to count up or down. This mode allows pin T2EX to control the direction of count. When a logic 1 is applied at pin T2EX Timer 2 will count up. Timer 2 will overflow at 0FFFFH and set the TF2 flag, which can then generate an interrupt, if the interrupt is enabled. This timer overflow also causes the 16-bit value in RCAP2L and RCAP2H to be reloaded into the timer registers TL2 and TH2.

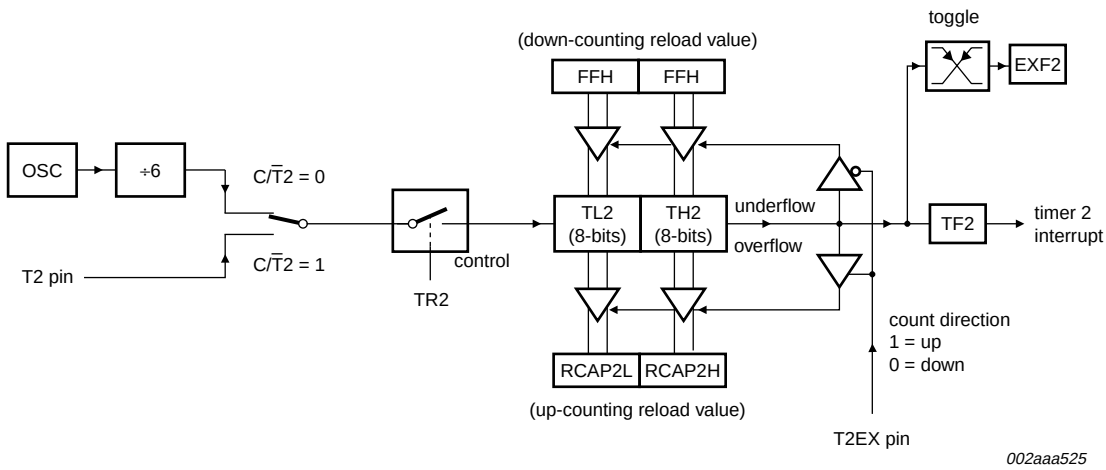


Fig 13. Timer 2 in Auto reload mode (DCEN = 1)

When a logic 0 is applied at pin T2EX this causes Timer 2 to count down. The timer will underflow when TL2 and TH2 become equal to the value stored in RCAP2L and RCAP2H. Timer 2 underflow sets the TF2 flag and causes 0FFFFH to be reloaded into the timer registers TL2 and TH2. The external flag EXF2 toggles when Timer 2 underflows or overflows. This EXF2 bit can be used as a 17th bit of resolution if needed.

6.5.3 Programmable clock-out

A 50 % duty cycle clock can be programmed to come out on pin T2 (P1[0], Clock-out mode). This pin, besides being a regular I/O pin, has two additional functions. It can be programmed:

1. To input the external clock for timer/counter 2, or
2. To output a 50 % duty cycle clock ranging from 122 Hz to 8 MHz at a 16 MHz operating frequency.

To configure the timer/counter 2 as a clock generator, bit C/$\overline{T}$2 (in T2CON) must be cleared and bit T2OE in T2MOD must be set. Bit TR2 (T2CON.2) also must be set to start the timer.

The clock-out frequency depends on the oscillator frequency and the reload value of Timer 2 capture registers (RCAP2H, RCAP2L) as shown in Equation 2:

$$\frac{OscillatorFrequency}{2 \times (65536 - (RCAP2H, RCAP2L))} \quad (2)$$

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

In the Clock-out mode, Timer 2 roll-overs will not generate an interrupt. This is similar to when it is used as a baud rate generator.

6.5.4 Baud rate generator mode

Bits TCLK and/or RCLK in T2CON allow the UART transmit and receive baud rates to be derived from either Timer 1 or Timer 2; see Section 6.6 for details. When TCLK = 0, Timer 1 is used as the UART transmit baud rate generator. When 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 14 shows Timer 2 in Baud rate generator mode:

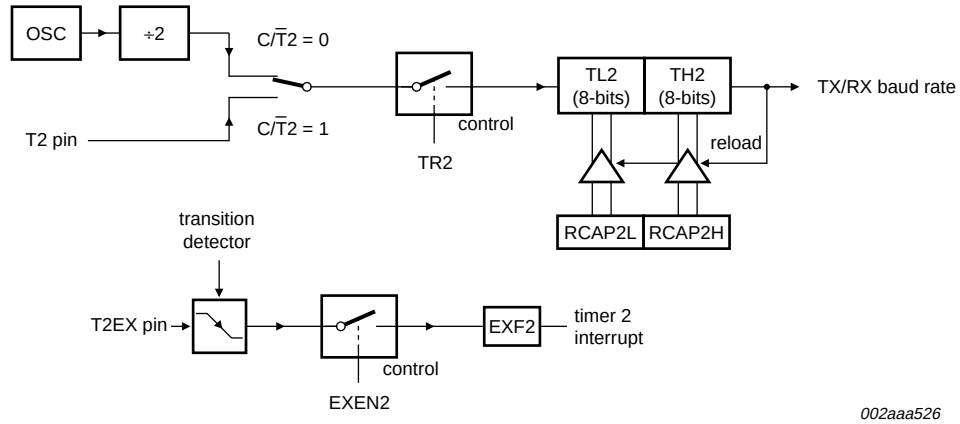


Fig 14. Timer 2 in Baud rate generator mode

The Baud rate generator mode is like the Auto-reload mode, when a roll-over 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 at 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 shown in Equation 3:

Modes 1 and 3 baud rates =

$$\frac{OscillatorFrequency}{16 \times (65536 - (RCAP2H, RCAP2L))} \quad (3)$$

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

The Timer 2 in Baud rate generator mode is valid only if RCLK and/or TCLK = 1 in T2CON register. Note that a roll-over 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 not cause a reload from (RCAP2H, RCAP2L) to (TH2, TL2). Therefore when Timer 2 is used as a baud rate generator, T2EX can be used as an additional external interrupt, if needed.

When Timer 2 is in the Baud rate generator mode, one should not try to read or write TH2 and TL2. 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 23 shows commonly used baud rates and how they can be obtained from Timer 2.

6.5.5 Summary of baud rate equations

Timer 2 is in Baud rate generator mode: if Timer 2 is being clocked through pin T2 (P1[0]) the baud rate is:

Baud rate = Timer 2 overflow rate / 16

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

Baud rate = f_{osc} / (16 × (65536 – (RCAP2H, RCAP2L)))

Where f_{osc} = oscillator frequency

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

RCAP2H, RCAP2L = 65536 – f_{osc} / (16 × baud rate)

Table 23. Timer 2-generated commonly used baud rates

Rate	Oscillator frequency	Timer 2	
		RCAP2H	RCAP2L
750 kBd	12 MHz	FF	FF
19.2 kBd	12 MHz	FF	D9
9.6 kBd	12 MHz	FF	B2
4.8 kBd	12 MHz	FF	64
2.4 kBd	12 MHz	FE	C8
600 Bd	12 MHz	FB	1E
220 Bd	12 MHz	F2	AF
600 Bd	6 MHz	FD	8F
220 Bd	6 MHz	F9	57

6.6 UART

The UART operates in all standard modes. Enhancements over the standard 80C51 UART include framing error detection, and automatic address recognition.

6.6.1 Mode 0

Serial data enters and exits through RXD, and TXD outputs the shift clock. Only 8 bits are transmitted or received, LSB first. The baud rate is fixed at $1/6$ of the CPU clock frequency. The UART configured to operate in this mode outputs serial clock on the TXD line no matter whether it sends or receives data on the RXD line.

6.6.2 Mode 1

10 bits are transmitted (through TXD) or received (through RXD): a start bit (logical 0), 8 data bits (LSB first), and a stop bit (logical 1). When data is received, the stop bit is stored in RB8 in special function register SCON. The baud rate is variable and is determined by the Timer 1/Timer 2 overflow rate.

6.6.3 Mode 2

11 bits are transmitted (through TXD) or received (through RXD): start bit (logical 0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (logical 1). When data is transmitted, the 9th data bit (TB8 in special function register SCON) can be assigned the value of 0 or (e.g. the parity bit (P in special function register PSW) could be moved into bit TB8). When data is received, 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 $1/16$ or $1/32$ of the CPU clock frequency, as determined by the SMOD1 bit in PCON.

6.6.4 Mode 3

11 bits are transmitted (through TXD) or received (through RXD): a start bit (logical 0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (logical 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 and is determined by the Timer 1/Timer 2 overflow rate.

Table 24. SCON - Serial port control register (address 98H) bit allocation

Bit addressable; reset value: 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI

Table 25. SCON - Serial port control register (address 98H) bit description

Bit	Symbol	Description
7	SM0/FE	The usage of this bit is determined by SMOD0 in the PCON register. If SMOD0 = 0, this bit is SM0, which with SM1, defines the serial port mode. If SMOD0 = 1, this bit is FE (Framing Error). FE is set by the receiver when an invalid stop bit is detected. Once set, this bit cannot be cleared by valid frames but can only be cleared by software. (Note: It is recommended to set up UART mode bits SM0 and SM1 before setting SMOD0 to 1.)
6	SM1	With SM0, defines the serial port mode; see Table 26.
5	SM2	Enables the multiprocessor communication feature in modes 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 a valid stop bit was not received. In Mode 0, SM2 should be 0.
4	REN	Enables serial Reception. Set by software to enable reception. Clear by software to disable reception.
3	TB8	The 9th data bit that will be transmitted in modes 2 and 3. Set or clear by software as desired.
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 26. 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 roll-overs 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.

The signal to load special function register 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 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 9th data bit goes into RB8, and the first 8 data bits go into SBUF.

6.6.8 Multiprocessor communications

UART modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received or transmitted. When data is received, the 9th bit is stored in RB8. The UART can be programmed so 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. One 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: 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, i.e. the received 9th bit is 0. However, an address byte having the 9th bit set to 1 will interrupt all slaves, so that each slave can examine the received byte to see if it is being addressed or not. The addressed slave will clear its SM2 bit and prepare to receive the data (still 9 bits long) that follow. The slaves that were not addressed leave their SM2 bits set and ignore the subsequent data bytes.

SM2 has no effect in Mode 0, and in Mode 1 can be used to check the validity of the stop bit, although it is preferable to use the Framing Error flag (FE). When the UART receives data in Mode 1 and SM2 = 1, the receive interrupt will not be activated unless a valid stop bit is received.

6.6.9 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 for the UART 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.

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 be 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.

This device uses the methods presented in Figure 15 to determine if a given or broadcast address has been received or not.

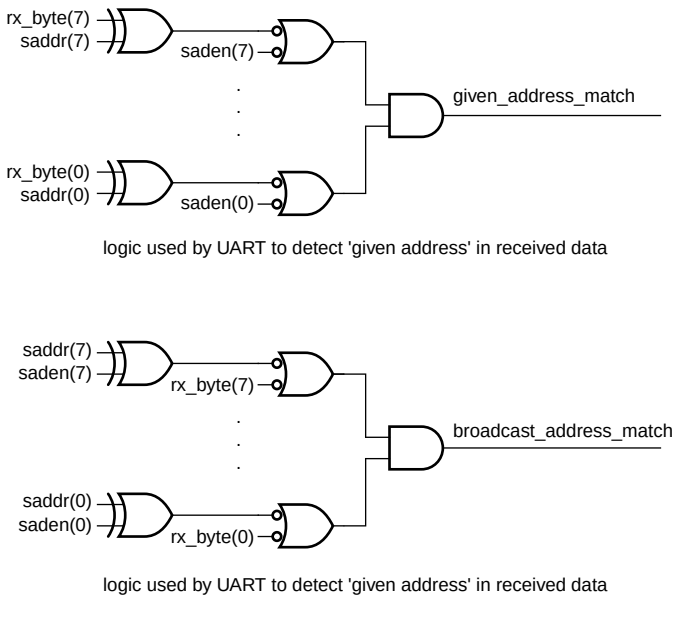


Fig 15. Schemes used by the UART to detect 'given' and 'broadcast' addresses when multiprocessor communications is enabled

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

Example 1, slave 0:

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

Example 2, 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.

Example 1, slave 0:

SADDR = 1100 0000
SADEN = 1111 1001
Given = 1100 0XX0

Example 2, slave 1:

SADDR = 1110 0000
SADEN = 1111 1010
Given = 1110 0X0X

Example 3, slave 2:

SADDR = 1100 0000
SADEN = 1111 1100
Given = 1100 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 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 treated as don't cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FFH. Upon reset SADDR and SADEN are loaded 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 UART drivers which do not make use of this feature.

6.7 Serial Peripheral Interface (SPI)

6.7.1 SPI features

- Master or slave operation
- 10 MHz bit frequency (max)
- LSB first or MSB first data transfer
- Four programmable bit rates
- End of transmission (SPIF)
- Write-collision flag protection (WCOL)
- Wake-up from Idle mode (Slave mode only)

6.7.2 SPI description

The serial peripheral interface allows high-speed synchronous data transfer between the P89CV51RB2/RC2/RD2 and peripheral devices or between several P89CV51RB2/RC2/RD2 devices. Figure 16 shows the correspondence between master and slave SPI devices. The SPICLK pin is the 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 of 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 SPI interrupt Flag (SPIF) is set. An SPI interrupt request will be generated if the SPI Interrupt Enable bit (SPIE) and the SPI interrupt enable bit, ES, are both set.

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

CPHA and CPOL control the phase and polarity of the SPI clock (SCK). Figure 17 and Figure 18 show the four possible combinations of these two bits.

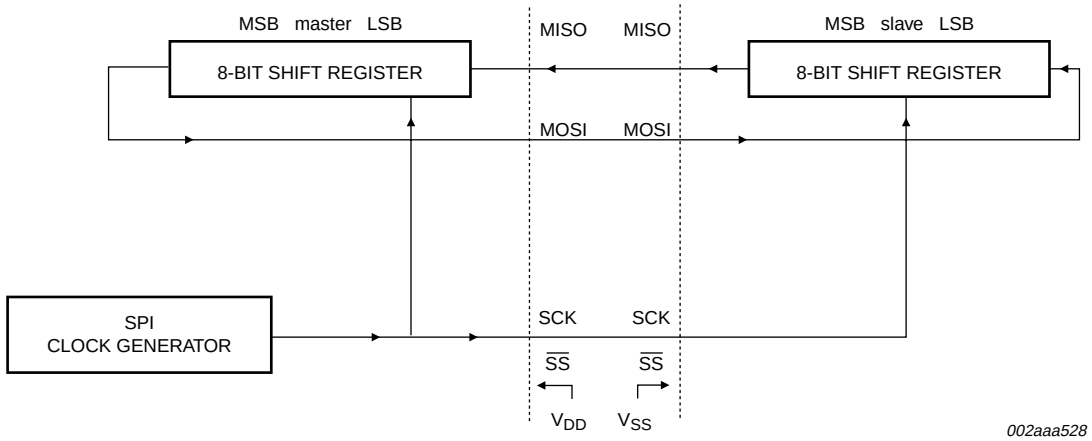


Fig 16. SPI master-slave interconnection

Table 27. SPCR - SPI control register (address D5H) bit allocation

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

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

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

Bit	Symbol	Description
7	SPIE	SPI interrupt enable. If both SPIE = 1 and ES = 1, SPI interrupts are enabled.
6	SPEN	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).
2	CPHA	Clock Phase control bit. 1 = shift-triggered on the trailing edge of the clock; 0 = shift-triggered on the leading edge of the clock.
1	SPR1	SPI clock Rate select bit 1. Along with SPR0 controls the SPICLK rate of the device when a master. SPR1 and SPR0 have no effect on the slave; see Table 29.
0	SPR0	SPI clock Rate select bit 0. Along with SPR1 controls the SPICLK rate of the device when a master. SPR1 and SPR0 have no effect on the slave; see Table 29.

Table 29. SPCR - SPI control register (address D5H) clock rate selection

SPR1	SPR0	SPICLK = f_{osc} divided by	
		6-clock mode	12-clock mode
0	0	2	4
0	1	8	16
1	0	32	64
1	1	64	128

Table 30. SPSR - SPI Status Register (address AAH) bit allocation

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

Bit	7	6	5	4	3	2	1	0
Symbol	SPIF	WCOL	-	-	-	-	-	-

Table 31. SPSR - SPI Status Register (address AAH) bit description

Bit	Symbol	Description
7	SPIF	SPI interrupt flag. Upon completion of data transfer, this bit is set to 1. If SPIE = 1 and ES = 1, an interrupt is then generated. This bit is cleared by software.
6	WCOL	Write Collision flag. Set if the SPI data register is written to during data transfer. This bit is cleared by software.
5 to 0	-	Reserved for future use. Should be set to 0 by user programs.

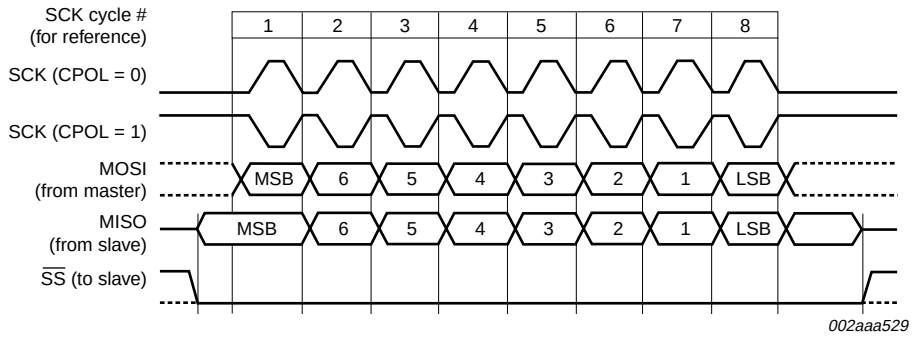


Fig 17. SPI transfer format with CPHA = 0

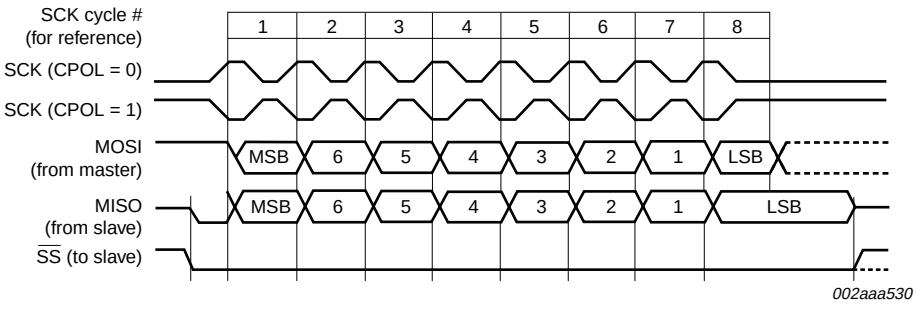


Fig 18. SPI transfer format with CPHA = 1

6.8 Watchdog timer

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upset. The WDT consists of a 14-bit counter and the WatchDog Timer Reset (WDTRST) SFR. The WDT is disabled at reset. To enable the WDT, the user must write 01EH and 0E1H, in sequence, to the WDTRST SFR. When the WDT is enabled, it will increment every machine cycle while the oscillator is running. There is no

way to disable the WDT, except through a reset (either a hardware reset or a WDT overflow reset). When the WDT overflows, it will drive an output reset HIGH pulse at the RST pin.

When the WDT is enabled (and thus running) the user needs to reset it by writing 01EH and 0E1H, in sequence, to the WDTRST SFR to avoid WDT overflow. The 14-bit counter reaches overflow when it reaches 16383 (3FFFH) and this will reset the device.

The WDT's counter cannot be read or written. When the WDT overflows it will generate an output pulse at the RST pin with a duration of 98 oscillator periods in 6-clock mode or 196 oscillator periods in 12-clock mode.

6.9 PCA

The PCA includes a special 16-bit timer that has five 16-bit capture/compare modules associated with it. Each of the modules can be programmed to operate in one of four modes: rising and/or falling edge capture, software timer, high-speed output, or pulse-width modulator. Each module has a pin associated with it: Module 0 is connected to CEX0, module 1 to CEX1, etc. Registers CH and CL contain the current value of the free-running up-counting 16-bit PCA timer. The PCA timer is a common time base for all five modules and can be programmed to run at: $\frac{1}{6}$ the oscillator frequency, $\frac{1}{2}$ the oscillator frequency, the Timer 0 overflow, or the input on the ECI pin (P1[2]). The timer count source is determined from the CPS1 and CPS0 bits in the CMOD SFR; see Table 32 and Table 33.

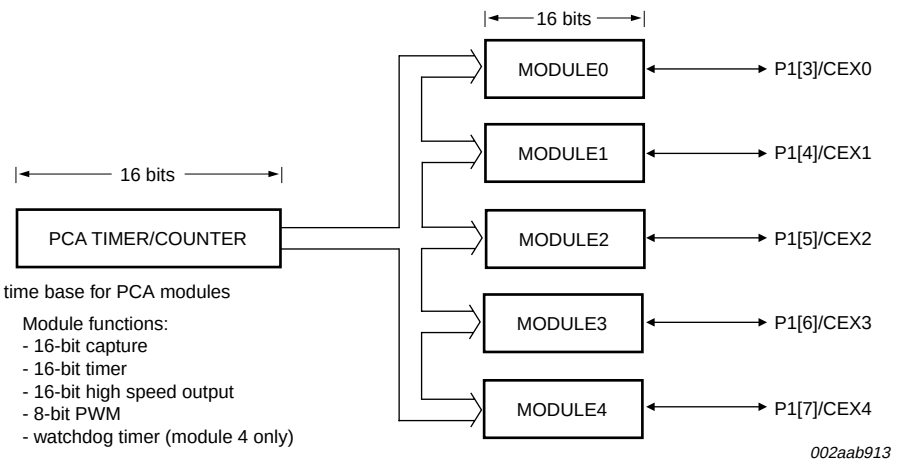


Fig 19. PCA

In the CMOD SFR there are three additional bits associated with the PCA. They are CIDL which allows the PCA to stop during Idle mode, WDTE which enables or disables the watchdog function on module 4, and ECF which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows.

The watchdog timer function is implemented in module 4 of PCA.

The CCON SFR contains the run control bit for the PCA (CR) and the flags for the PCA timer (CF) and each module (CCF[4:0]). To run the PCA the CR bit (CCON.6) must be set by software. The PCA is shut off by clearing this bit. The CF bit (CCON.7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD

register is set. The CF bit can only be cleared by software. Bits 0 through 4 of the CCON register are the flags for the modules (bit 0 for module 0, bit 1 for module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags can only be cleared by software. All the modules share one interrupt vector. The PCA interrupt system is shown in Figure 20.

Each module in the PCA has a special function register associated with it. These registers are: CCAPM0 for module 0, CCAPM1 for module 1, etc. The registers contain the bits that control the mode that each module will operate in.

The ECCF bit (from CCAPMn.0 where n = 0, 1, 2, 3, or 4 depending on the module) enables the CCFn flag in the CCON SFR to generate an interrupt when a match or compare occurs in the associated module; see Figure 20.

PWM (CCAPMn.1) enables the PWM mode.

The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register.

The match bit MAT (CCAPMn.3) when set will cause the CCFn bit in the CCON register to be set when there is a match between the PCA counter and the module's capture/compare register.

The next two bits CAPN (CCAPMn.4) and CAPP (CCAPMn.5) determine the edge that a capture input will be active on. The CAPN bit enables the negative edge, and the CAPP bit enables the positive edge. If both bits are set both edges will be enabled and a capture will occur for either transition.

The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function.

There are two additional registers associated with each of the PCA modules. They are CCAPnH and CCAPnL and these are the registers that store the 16-bit count when a capture occurs or a compare should occur. When a module is used in the PWM mode, these registers are used to control the duty cycle of the output.

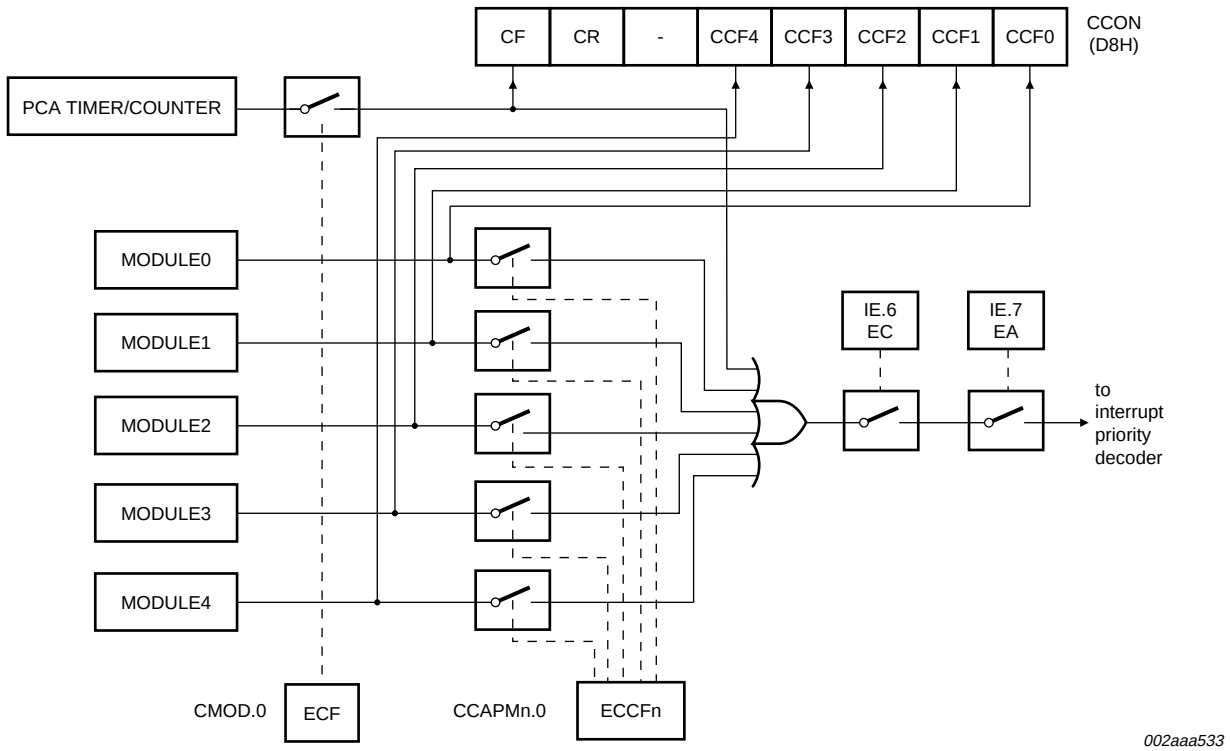


Fig 20. PCA interrupt system

Table 32. 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 33. 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 mode.
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 34.
0	ECF	PCA Enable Counter overflow interrupt Flag. ECF = 1 enables CF bit in CCON to generate an interrupt; ECF = 0 disables that function.

Table 34. CMOD - PCA counter mode register (address D9H) count pulse select

CPS1	CPS0	Select PCA input
0	0	0 internal clock, f_{osc} / 6

Table 34. CMOD - PCA counter mode register (address D9H) count pulse select *...continued*

CPS1	CPS0	Select PCA input
0	1	1 internal clock, f_{osc} / 6
1	0	2 Timer 0 overflow
1	1	3 external clock at pin P1[2]/ECI (maximum rate = f_{osc} / 4)

Table 35. CCON - PCA counter control register (address D8H) bit allocation

Bit addressable; reset value: 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	CF	CR	-	CCF4	CCF3	CCF2	CCF1	CCF0

Table 36. CCON - PCA counter control register (address D8H) bit description

Bit	Symbol	Description
7	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.
6	CR	PCA Counter Run control. Set by software to turn the PCA counter on. Must be cleared by software to turn the PCA counter off.
5	-	Reserved for future use. Should be set to 0 by user programs.
4	CCF4	PCA Module 4 interrupt Flag. Set by hardware when a match or capture occurs. Must be cleared by software.
3	CCF3	PCA Module 3 interrupt Flag. Set by hardware when a match or capture occurs. Must be cleared by software.
2	CCF2	PCA Module 2 interrupt Flag. Set by hardware when a match or capture occurs. Must be cleared by software.
1	CCF1	PCA Module 1 interrupt Flag. Set by hardware when a match or capture occurs. Must be cleared by software.
0	CCF0	PCA Module 0 interrupt Flag. Set by hardware when a match or capture occurs. Must be cleared by software.

Table 37. CCAPMn - PCA modules compare/capture register (address CCAPM0 DAH, CCAPM1 DBH, CCAPM2 DCH, CCAPM3 DDH, CCAPM4 DEH) bit allocation

Not bit addressable; reset value: 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	-	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn

Table 38. CCAPMn - PCA modules compare/capture register (address CCAPM0 DAH, CCAPM1 DBH, CCAPM2 DCH, CCAPM3 DDH, CCAPM4 DEH) bit description

Bit	Symbol	Description
7	-	Reserved for future use. Should be set to 0 by user programs.
6	ECOMn	Enable Comparator. ECOMn = 1 enables the comparator function.
5	CAPPn	Capture Positive, CAPPn = 1 enables positive edge capture.
4	CAPNn	Capture Negative, CAPNn = 1 enables negative edge capture.
3	MATn	Match. When MATn = 1 a match of the PCA counter with this module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt.

Table 38. CCAPMn - PCA modules compare/capture register (address CCAPM0 DAH, CCAPM1 DBH, CCAPM2 DCH, CCAPM3 DDH, CCAPM4 DEH) 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 39. 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 21), 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).

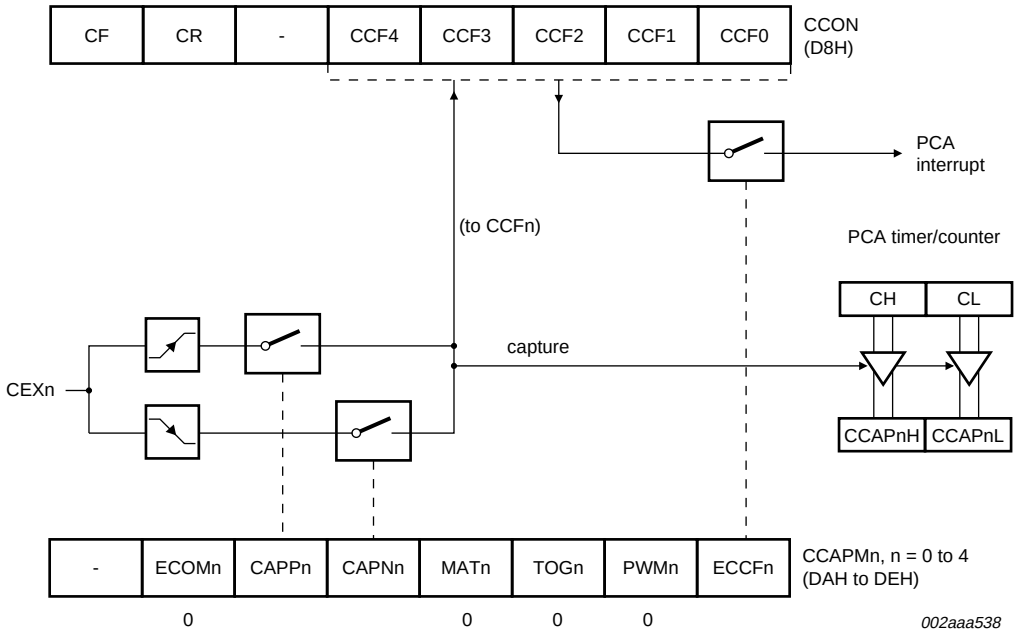


Fig 21. PCA Capture mode

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 22) 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.

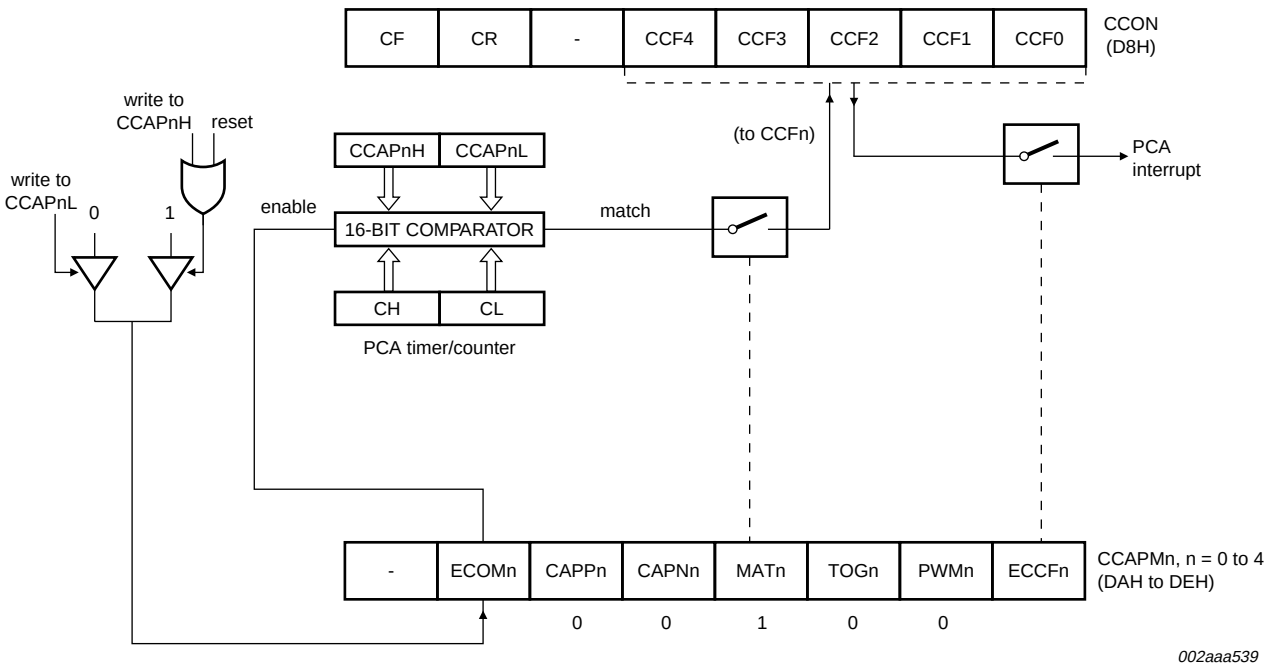


Fig 22. PCA Compare mode

6.9.3 High-speed output mode

In this mode (Figure 23) the CEX output (on port 1) associated with the PCA module will toggle each time a match occurs between the PCA counter and the module's capture registers. To activate this mode the TOG, MAT, and ECOM bits in the module's CCAPMn SFR must be set.

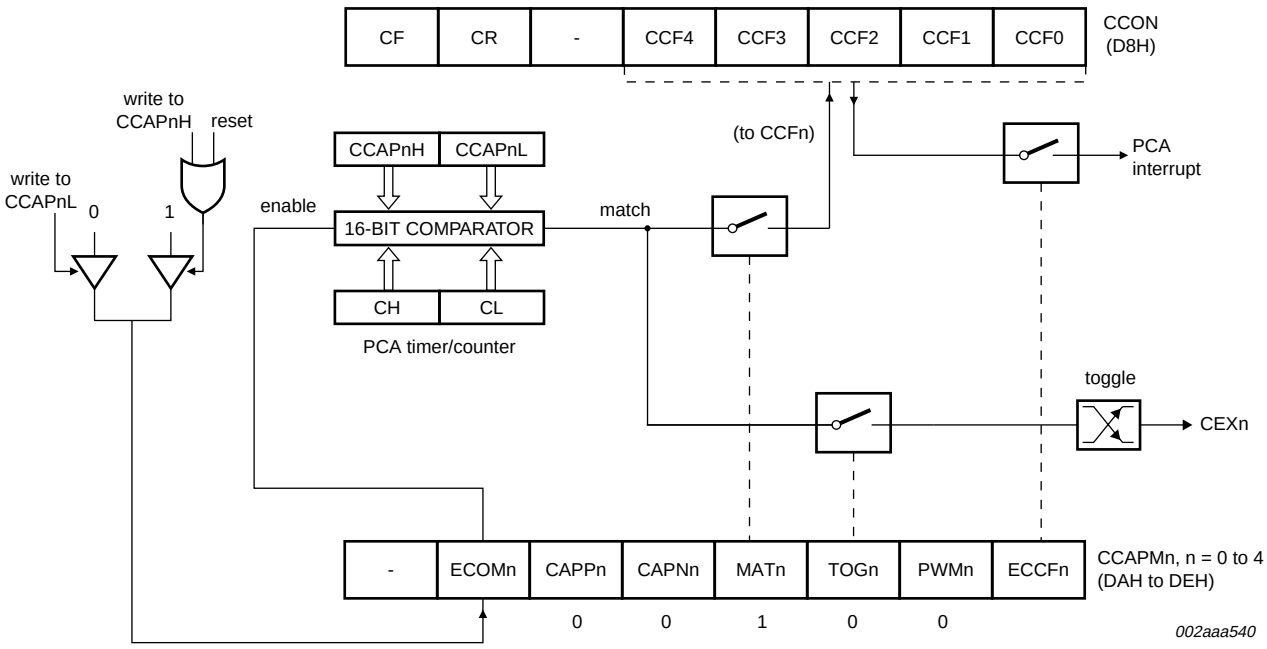


Fig 23. PCA High-speed output mode

6.9.4 Pulse width modulator mode

All of the PCA modules can be used as PWM outputs (Figure 24). Output frequency depends on the source for the PCA timer.

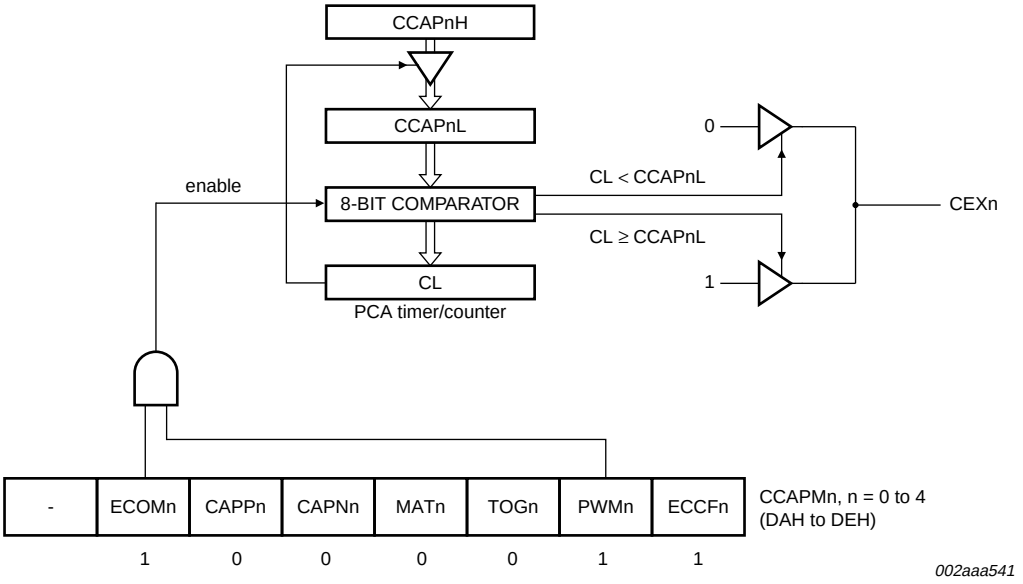


Fig 24. PCA PWM mode

All of the modules will have the same output frequency because they all share only one PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPnL. When the value of the PCA CL SFR is less than the value in the

module's CCAPnL SFR, the output will be LOW; when it is equal to, or greater, the output will be HIGH. When CL overflows from FFH to 00H, CCAPnL is reloaded with the value in CCAPnH. This allows the PWM to be updated without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable PWM mode.

6.9.5 PCA watchdog timer

An on-board watchdog timer is available with the PCA to improve the reliability of the system without increasing chip count. Watchdog timers are useful for systems that are susceptible to noise, power glitches, or electrostatic discharge. Module 4 is the only PCA module that can be programmed as a watchdog. However, this module can still be used for other modes if the watchdog is not needed. Figure 24 shows a diagram of how the watchdog works. The user pre-loads a 16-bit value in the compare registers. Just like the other compare modes, this 16-bit value is compared to the PCA timer value. If a match is allowed to occur, an internal reset will be generated. This will not cause the RST pin to be driven HIGH.

User's software then must periodically change (CCAP4H, CCAP4L) to keep a match from occurring with the PCA timer (CH, CL). This code is given in the subroutine WATCHDOG shown below.

In order to hold off the reset, the user has three options:

- Periodically change the compare value so it will never match the PCA timer.
- Periodically change the PCA timer value so it will never match the compare values.
- Disable the watchdog by clearing the WDTE bit before a match occurs and then re-enable it.

The first two options are more reliable because the watchdog timer is never disabled as in the third option. If the program counter ever reaches an undesired value, a match will eventually occur and cause an internal reset. The second option is also not recommended if other PCA modules are being used. Remember that the PCA timer is the time base for **all** modules; changing the time base for other modules is not recommended. Thus, in most applications the first option is best.

```
;CALL the following WATCHDOG subroutine periodically.
   CLR    EA           ;Hold off interrupts
   MOV    CCAP4L,#00   ;Next compare value is within 255 counts of
                        current PCA timer value
   MOV    CCAP4H,CH
   SETB   EA           ;Re-enable interrupts
   RET
```

Do not use this routine as part of an interrupt service routine, because if the program counter would enter an infinite loop, still interrupts will be serviced and the watchdog will continually keep getting reset. Because this would defeat the purpose of the watchdog, it is recommended that this subroutine is called from the main program within 2^{16} PCA timer counts.

6.10 Security bits

The security bits protect against software piracy and prevent the contents of the flash from being read by unauthorized parties in Parallel programmer mode and ISP mode. Since the end application might need to erase pages and read from the code memory, the security bits have no effect in IAP mode. However, the security bits' programmed/erased state may be read using IAP function calls allowing the end-user code to limit access, if desired. The security bits and their effects are shown in Table 40.

Note: On this device, MOVC instructions executed from external code memory are prevented from fetching code bytes from internal code memory.

Table 40. Security bit functions

Security bit	Description
1	Write protect. When programmed, prohibits further erasing or programming, except to program other security bits or a chip erase.
2	Read protect. When programmed, inhibits reading of user code memory.
3	External execution inhibit. When programmed, prevents any execution of instructions from external code memory.

6.11 Interrupt priority and polling sequence

The device supports eight interrupt sources under a four-level priority scheme. Table 41 summarizes the polling sequence of the supported interrupts. Note that the SPI serial interface and the UART share the same interrupt vector; see Figure 25.

Table 41. Interrupt polling sequence

Description	Interrupt flag	Vector address	Interrupt enable	Interrupt priority	Service priority	Wake-up Power-down
External Interrupt 0	IE0	0003H	EX0	PX0/PX0H	1 (highest)	yes
T0	TF0	000BH	ET0	PT0/PT0H	2	no
External Interrupt 1	IE1	0013H	EX1	PX1/PX1H	3	yes
T1	TF1	001BH	ET1	PT1/PT1H	4	no
UART	TI/RI	0023H	ES	PS/PSH	5	no
SPI	SPIF	0023H	ES	PS/PSH	5	no
PCA	CF/CCFn	0033H	EC	PPC/PPCH	7	no
T2	TF2, EXF2	003BH	ET2	PT2/PT2H	6	no

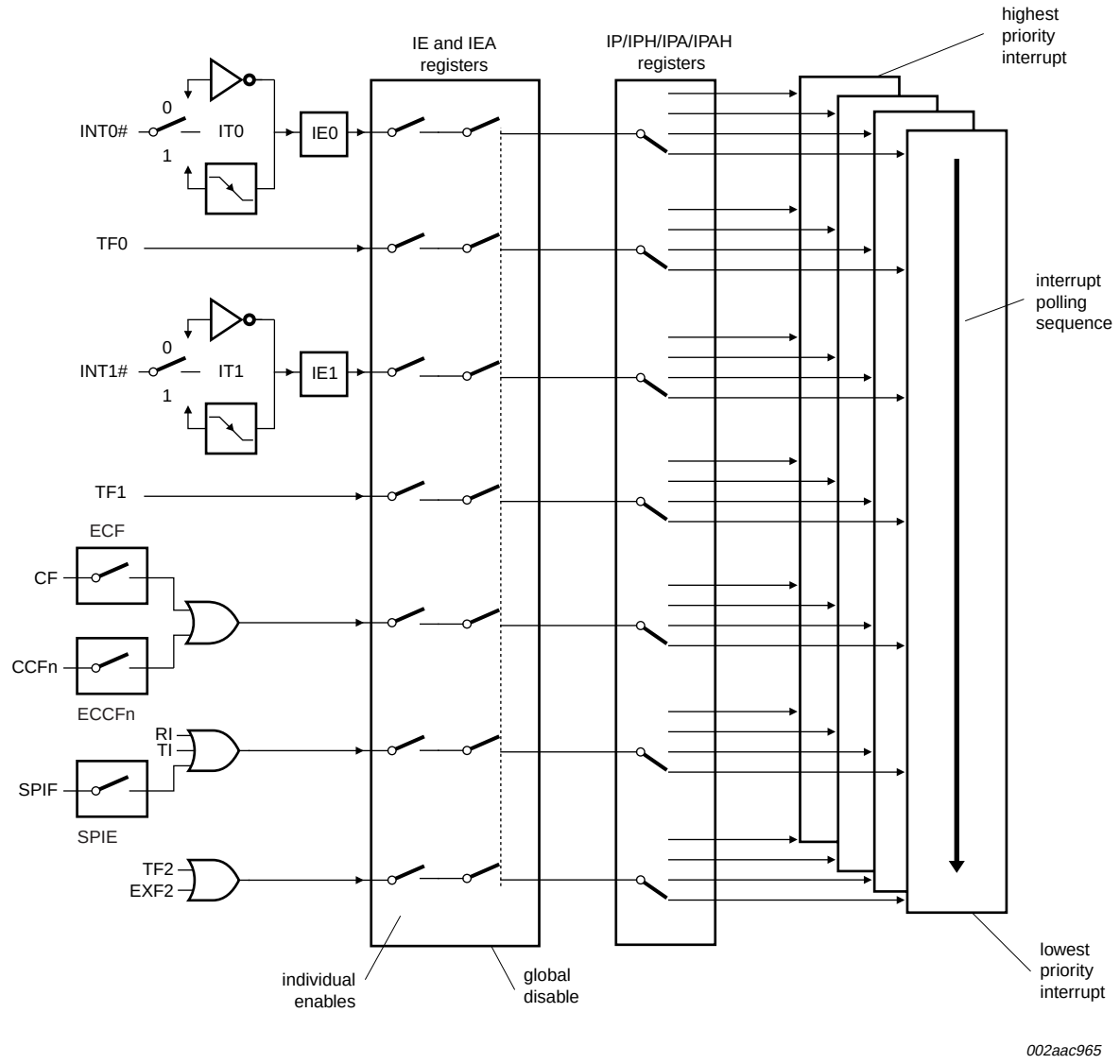


Fig 25. Interrupt structure

Table 42. IE - Interrupt enable register 0 (address A8H) bit allocation

Bit addressable; reset value: 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	EA	EC	ET2	ES	ET1	EX1	ET0	EX0

Table 43. IE - Interrupt enable register 0 (address A8H) bit description

Bit	Symbol	Description
7	EA	Interrupt Enable. EA = 1: interrupt(s) can be serviced; EA = 0: interrupt servicing disabled.
6	EC	PCA interrupt Enable.
5	ET2	Timer 2 interrupt Enable.

Table 43. IE - Interrupt enable register 0 (address A8H) bit description *...continued*

Bit	Symbol	Description
4	ES	Serial port interrupt Enable.
3	ET1	Timer 1 overflow interrupt Enable.
2	EX1	External interrupt 1 Enable.
1	ET0	Timer 0 overflow interrupt Enable.
0	EX0	External interrupt 0 Enable.

Table 44. IP - Interrupt priority low register (address B8H) bit allocation

Bit addressable; reset value: 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	-	PPC	PT2	PS	PT1	PX1	PT0	PX0

Table 45. IP - Interrupt priority low register (address B8H) bit description

Bit	Symbol	Description
7	-	Reserved for future use. Should be set to 0 by user programs.
6	PPC	PCA interrupt Priority Low.
5	PT2	Timer 2 interrupt Priority Low.
4	PS	Serial Port interrupt Priority Low.
3	PT1	Timer 1 interrupt Priority Low.
2	PX1	External interrupt 1 Priority Low.
1	PT0	Timer 0 interrupt Priority Low.
0	PX0	External interrupt 0 Priority Low.

Table 46. IPH - Interrupt priority high register (address B7H) bit allocation

Not bit addressable; reset value: 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	-	PPCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H

Table 47. IPH - Interrupt priority high register (address B7H) bit description

Bit	Symbol	Description
7	-	Reserved for future use. Should be set to 0 by user programs.
6	PPCH	PCA interrupt Priority High.
5	PT2H	Timer 2 interrupt Priority High.
4	PSH	Serial Port interrupt Priority High.
3	PT1H	Timer 1 interrupt Priority High.
2	PX1H	External interrupt 1 Priority High.
1	PT0H	Timer 0 interrupt Priority High.
0	PX0H	External interrupt 0 Priority High.

6.12 Power-saving modes

The device provides two power-saving modes of operation for applications where power consumption is critical. The two modes are Idle and Power-down; see Table 48.

6.12.1 Idle mode

Idle mode is entered by setting the IDL bit in the PCON register. In Idle mode, the program counter is stopped. The system clock continues to run and all interrupts and peripherals remain active. The on-chip RAM and the special function registers hold their data during this mode.

The device exits Idle mode through either a system interrupt or a hardware reset. When exiting Idle mode via system interrupt, the start of the interrupt clears the IDL bit and exits Idle mode. After exiting the Interrupt Service Routine (ISR), the interrupted program resumes execution at the instruction immediately following the instruction which invoked the Idle mode. A hardware reset starts the device similar to a power-on reset.

6.12.2 Power-down mode

The Power-down mode is entered by setting the PD bit in the PCON register. In the Power-down mode, the clock is stopped and external interrupts are active for level-sensitive interrupts only. SRAM contents are retained during power-down, the minimum V_{DD} level is 4.5 V.

The device exits Power-down mode through either an enabled external level-sensitive interrupt or a hardware reset. The start of the interrupt clears the PD bit and exits power-down. Holding an external interrupt pin LOW restarts the oscillator, the signal must hold LOW at least 1024 clock cycles before bringing back HIGH to complete the exit. When the interrupt signal is restored to logic V_{IH}, the interrupt service routine program execution resumes at the instruction immediately following the instruction which invoked Power-down mode. A hardware reset starts the device similar to a power-on reset.

To exit properly out of power-down, the reset or external interrupt should not be executed before the V_{DD} line is restored to its normal operating voltage. Be sure to hold V_{DD} voltage long enough at its normal operating level for the oscillator to restart and stabilize (normally less than 10 ms).

Table 48. Power-saving modes

Mode	Initiated by	State of MCU	Exited by
Idle	Software (Set IDL bit in PCON) MOV PCON, #01H	CLK is running. Interrupts, serial port and timers/counters are active. Program counter is stopped. ALE and $\overline{PSEN}$ signals at a HIGH-state during Idle. All registers remain unchanged.	Enabled interrupt or hardware reset. Start of interrupt clears IDL bit and exits Idle mode, after the ISR RETI instruction, program resumes execution beginning at the instruction following the one that invoked Idle mode. A hardware reset restarts the device similar to a power-on reset.
Power-down	Software (Set PD bit in PCON) MOV PCON, #02H	CLK is stopped. On-chip SRAM and SFR data is maintained. ALE and $\overline{PSEN}$ signals at a LOW-state during power-down. External interrupts are only active for level-sensitive interrupts, if enabled.	Enabled external level-sensitive interrupt or hardware reset. Start of interrupt clears PD bit and exits Power-down mode, after the ISR RETI instruction program resumes execution beginning at the instruction following the one that invoked Power-down mode. A hardware reset restarts the device similar to a power-on reset.

6.13 System clock and clock options

6.13.1 Clock input options and recommended capacitor values for oscillator

Shown in Figure 26 and Figure 27 are the input and output of an internal inverting amplifier (XTAL1, XTAL2), which can be configured for use as an on-chip oscillator.

When driving the device from an external clock source, XTAL2 should be left disconnected and XTAL1 should be driven.

At start-up, the external oscillator may encounter a higher capacitive load at XTAL1 due to interaction between the amplifier and its feedback capacitance. However, the capacitance will not exceed 15 pF once the external signal meets the V_{IL} and V_{IH} specifications.

Resonator manufacturer, supply voltage, and other factors may cause circuit performance to differ from one application to another. C_1 and C_2 should be adjusted appropriately for each design. Table 49 shows the typical values for C_1 and C_2 versus resonator type for various frequencies.

Table 49. Recommended values for C_1 and C_2 by crystal type

Resonator	$C_1 = C_2$
Quartz	20 pF to 30 pF
Ceramic	40 pF to 50 pF

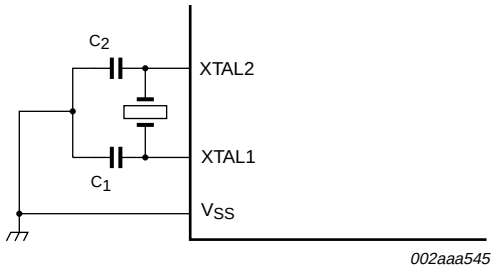


Fig 26. Oscillator characteristics (using the on-chip oscillator)

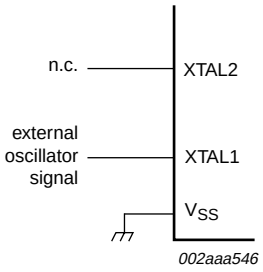


Fig 27. Oscillator characteristics (external clock drive)

6.13.1.1 Clock control register (CKCON)

By default, the device runs at twelve clock cycles per machine cycle (12-clock mode). The device may be run at 6 clock cycles per machine cycle (6-clock mode) by programming of either a non-volatile bit (FX2) or an SFR bit (X2); see Table 52 “Clock modes”. If the FX2 non-volatile bit is programmed, the device will run in 6-clock mode and the X2 SFR bit has no effect. If the FX2 bit is erased, then the clock mode is controlled by the X2 SFR bit.

Table 50. CKCON - Clock control register (address 8FH) bit allocation

Not bit addressable; reset value 00H.

Bit	7	6	5	4	3	2	1	0
Symbol	SPIX2	WDX2	PCAX2	SIX2	T2X2	T1X2	T0X2	X2

Table 51. CKCON - Clock control register (address 8FH) bit description

Bit	Symbol	Description
7	SPIX2	SPI clock; 0 = 6 clock cycles for each SPI clock cycle; 1 = 12 clock cycles
6	WDX2	Watchdog clock; 0 = 6 clock cycles for each WDT clock cycle; 1 = 12 clock cycles
5	PCAX2	PCA clock; 0 = 6 clock cycles for each PCA clock cycle; 1 = 12 clock cycles
4	SIX2	UART clock; 0 = 6 clock cycles for each UART clock cycle; 1 = 12 clock cycles
3	T2X2	Timer 2 clock; 0 = 6 clock cycles for each Timer 2 clock cycle; 1 = 12 clock cycles
2	T1X2	Timer 1 clock; 0 = 6 clock cycles for each Timer 1 clock cycle; 1 = 12 clock cycles
1	T0X2	Timer 0 clock; 0 = 6 clock cycles for each Timer 0 clock cycle; 1 = 12 clock cycles
0	X2	CPU clock; 0 = 12 clock cycles for each machine cycle; 1 = 6 clock cycles

Table 52. Clock modes

FX2 clock mode bit	X2 bit	CPU clock mode	Peripheral clock mode bit (e.g. T0X2)	Mode
erased	0	12-clock (default)	X	12-clock (default)
	1	6-clock	0	6-clock
			1	12-clock
programmed	X	6-clock	0	6-clock
			1	12-clock

7. Limiting values

Table 53. Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134). Parameters are valid over operating temperature range unless otherwise specified; all voltages are with respect to V_{SS} unless otherwise noted.

Symbol	Parameter	Conditions	Min	Max	Unit
$T_{amb(bias)}$	bias ambient temperature		–55	+125	°C
T_{stg}	storage temperature		–65	+150	°C
V_I	input voltage	on $\overline{EA}$ pin to V_{SS}	–0.5	+14	V
V_n	voltage on any other pin	except V_{SS}; with respect to V_{DD}	–0.5	V_{DD} + 0.5	V
$I_{OL(I/O)}$	LOW-level output current per input/output pin		-	15	mA
$P_{tot(pack)}$	total power dissipation (per package)	based on package heat transfer, not device power consumption	-	1.5	W

8. Static characteristics

Table 54. Static characteristics

T_{amb} = −40 °C to +85 °C; V_{DD} = 4.5 V to 5.5 V; V_{SS} = 0 V.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
$n_{endu(fl)}$	endurance of flash memory	JEDEC Standard A117	[1]	10000	-	-	cycles
$t_{ret(fl)}$	flash memory retention time	JEDEC Standard A103	[1]	100	-	-	years
I_{latch}	I/O latch-up current	JEDEC Standard 78	[1]	100 + I_{DD}	-	-	mA
$V_{th(HL)}$	HIGH-LOW threshold voltage			–0.5	-	+0.2V_{DD} – 0.1	V
$V_{th(LH)}$	LOW-HIGH threshold voltage	except XTAL1, RST		0.2V_{DD} + 0.9	-	V_{DD} + 0.5	V
V_{OL}	LOW-level output voltage	V_{DD} = 4.5 V; except $\overline{PSEN}$, ALE	[2][3][4]				
		I_{OL} = 1.6 mA		-	-	0.4	V
		V_{DD} = 4.5 V; ALE, $\overline{PSEN}$					
		I_{OL} = 3.2 mA		-	-	0.45	V
V_{OH}	HIGH-level output voltage	V_{DD} = 4.5 V; ports 1, 2, 3, 4	[5]				
		I_{OH} = –30 µA		V_{DD} – 0.7	-	-	V
		V_{DD} = 4.5 V; port 0 in External bus mode, ALE, $\overline{PSEN}$					
		I_{OH} = –3.2 mA		V_{DD} – 0.7	-	-	V
I_{IL}	LOW-level input current	V_I = 0.4 V; ports 1, 2, 3, 4		–1	-	–75	µA
I_{THL}	HIGH-LOW transition current	V_I = 2 V; ports 1, 2, 3, 4	[6]	-	-	–650	µA
I_{LI}	input leakage current	0.45 V < V_I < V_{DD} – 0.3 V; port 0		-	-	±10	µA
		0 V < V_I < 6 V		-	-	10	µA

Table 54. Static characteristics *…continued*

T_{amb} = −40 °C to +85 °C; V_{DD} = 4.5 V to 5.5 V; V_{SS} = 0 V.

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
R_{pd}	pull-down resistance	on pin RST		40	-	225	kΩ
C_{iss}	input capacitance	1 MHz; T_{amb} = 25 °C; V_I = 0 V	[7]	-	-	15	pF
$I_{DD(oper)}$	operating supply current	f_{osc} = 12 MHz		-	-	11.5	mA
		f_{osc} = 40 MHz		-	-	50	mA
		Programming and erase mode		-	-	70	mA
$I_{DD(idle)}$	Idle mode supply current	f_{osc} = 12 MHz		-	-	8.5	mA
		f_{osc} = 40 MHz		-	-	42	mA
$I_{DD(pd)}$	Power-down mode supply current	minimum V_{DD} = 4.5 V		-	-	90	µA

[1] This parameter is measured only for initial qualification and after a design or process change that could affect this parameter.

[2] Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:

a) Maximum I_{OL} per 8-bit port: 26 mA

b) Maximum I_{OL} total for all outputs: 71 mA

c) If I_{OL} exceeds the test condition, V_{OH} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

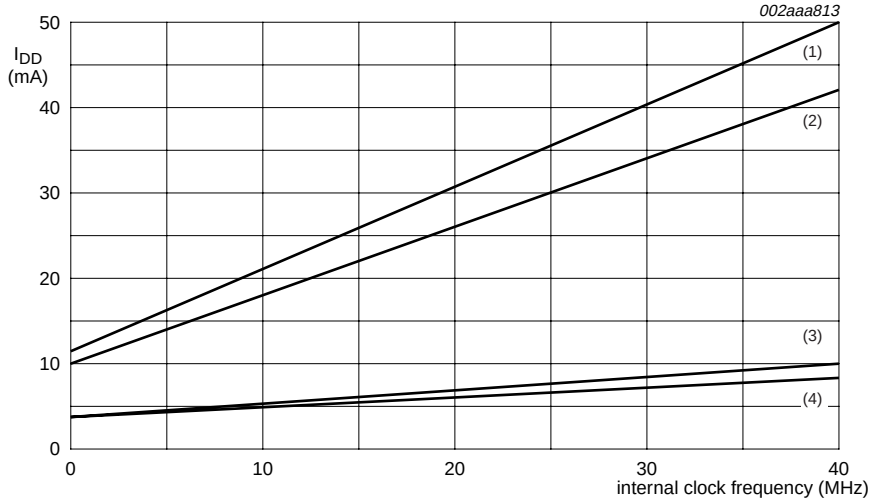
[3] Capacitive loading on Ports 0 and 2 may cause spurious noise to be superimposed on the V_{OL} of ALE and Ports 1 and 3. The noise is due to external bus capacitance discharging into the Port 0 and 2 pins when the pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading > 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.

[4] Load capacitance for Port 0, ALE and $\overline{PSEN}$ = 100 pF, load capacitance for all other outputs = 80 pF.

[5] Capacitive loading on Ports 0 and 2 may cause the V_{OH} on ALE and $\overline{PSEN}$ to momentarily fall below the V_{DD} − 0.7 V specification when the address bits are stabilizing.

[6] 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_I is approximately 2 V.

[7] Pin capacitance is characterized but not tested. Capacitance on pin $\overline{EA}$ = 25 pF (max.).



(1) Maximum active I_{DD}

(2) Maximum idle I_{DD}

(3) Typical active I_{DD}

(4) Typical idle I_{DD}

Fig 28. I_{DD} as a function of frequency

9. Dynamic characteristics

Table 55. Dynamic characteristics

Over operating conditions: load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF.

T_{amb} = −40 °C to +85 °C; V_{DD} = 4.5 V to 5.5 V; V_{SS} = 0 V.[1][2]

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{osc}	oscillator frequency	12-clock mode	0	-	40	MHz
		6-clock mode	0	-	20	MHz
		IAP	0.25	-	40	MHz
t_{LHLL}	ALE pulse width		$2T_{cy(clk)} - 15$	-	-	ns
t_{AVLL}	address valid to ALE LOW time		$T_{cy(clk)} - 15$	-	-	ns
t_{LLAX}	address hold after ALE LOW time		$T_{cy(clk)} - 15$	-	-	ns
t_{LLIV}	ALE LOW to valid instruction in time		-	-	$4T_{cy(clk)} - 45$	ns
t_{LLPL}	ALE LOW to $\overline{PSEN}$ LOW time		$T_{cy(clk)} - 15$	-	-	ns
t_{PLPH}	$\overline{PSEN}$ pulse width		$3T_{cy(clk)} - 15$	-	-	ns
t_{PLIV}	$\overline{PSEN}$ LOW to valid instruction in time		-	-	$3T_{cy(clk)} - 50$	ns
t_{PXIX}	input instruction hold after $\overline{PSEN}$ time		0	-	-	ns
t_{PXIZ}	input instruction float after $\overline{PSEN}$ time		-	-	$T_{cy(clk)} - 15$	ns
t_{PXAV}	$\overline{PSEN}$ to address valid time		$T_{cy(clk)} - 8$	-	-	ns
t_{AVIV}	address to valid instruction in time		-	-	$5T_{cy(clk)} - 60$	ns
t_{PLAZ}	$\overline{PSEN}$ LOW to address float time		-	-	10	ns
t_{RLRH}	$\overline{RD}$ LOW pulse width		$6T_{cy(clk)} - 30$	-	-	ns
t_{WLWH}	$\overline{WR}$ LOW pulse width		$6T_{cy(clk)} - 30$	-	-	ns
t_{RLDV}	$\overline{RD}$ LOW to valid data in time		-	-	$5T_{cy(clk)} - 50$	ns
t_{RHDX}	data hold after $\overline{RD}$ time		0	-	-	ns
t_{RHDZ}	data float after $\overline{RD}$ time		-	-	$2T_{cy(clk)} - 12$	ns
t_{LLDV}	ALE LOW to valid data in time		-	-	$8T_{cy(clk)} - 50$	ns
t_{AVDV}	address to valid data in time		-	-	$9T_{cy(clk)} - 75$	ns
t_{LLWL}	ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW time		$3T_{cy(clk)} - 15$	-	$3T_{cy(clk)} + 15$	ns
t_{AVWL}	address to $\overline{RD}$ or $\overline{WR}$ LOW time		$4T_{cy(clk)} - 30$	-	-	ns
t_{WHQX}	data hold after $\overline{WR}$ time		$T_{cy(clk)} - 20$	-	-	ns
t_{QVWH}	data output valid to $\overline{WR}$ HIGH time		$7T_{cy(clk)} - 50$	-	-	ns
t_{RLAZ}	$\overline{RD}$ LOW to address float time		-	-	0	ns
t_{WHLH}	$\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH time		$T_{cy(clk)} - 15$	-	$T_{cy(clk)} + 15$	ns

[1] $T_{cy(clk)} = 1 / f_{osc}$.

[2] Calculated values are for 6-clock mode only.

9.1 Explanation of symbols

Each timing symbol used in Figure 29 to Figure 33 has 5 characters. The first character is always a 't' (stands for time). The other characters, depending on their positions, stand for the name of a signal or the logical status of that signal. The following is a list of all the characters and what they stand for.

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 — cycle Time

V — Valid

W — $\overline{\text{WR}}$ signal

X — No longer a valid logic level

Z — High impedance (float)

Example:

t_{AVLL} = Address valid to ALE LOW time

t_{LLPL} = ALE LOW to $\overline{\text{PSEN}}$ LOW time

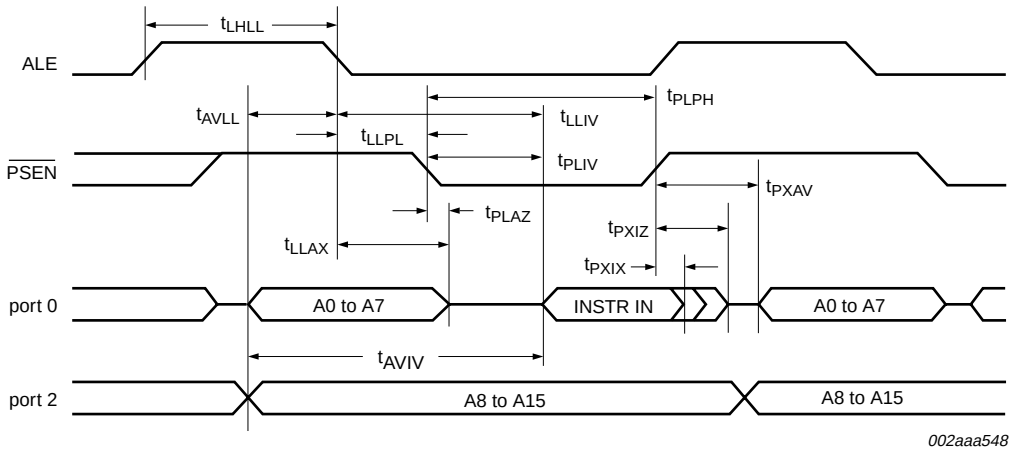


Fig 29. External program memory read cycle

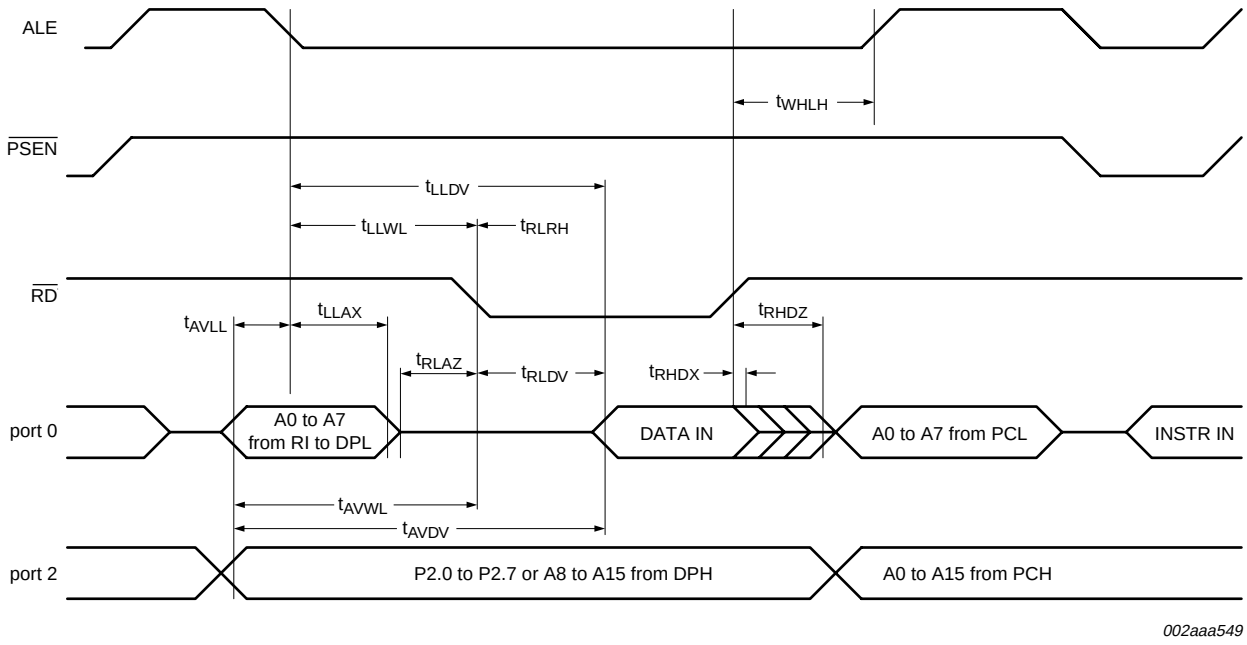


Fig 30. External data memory read cycle

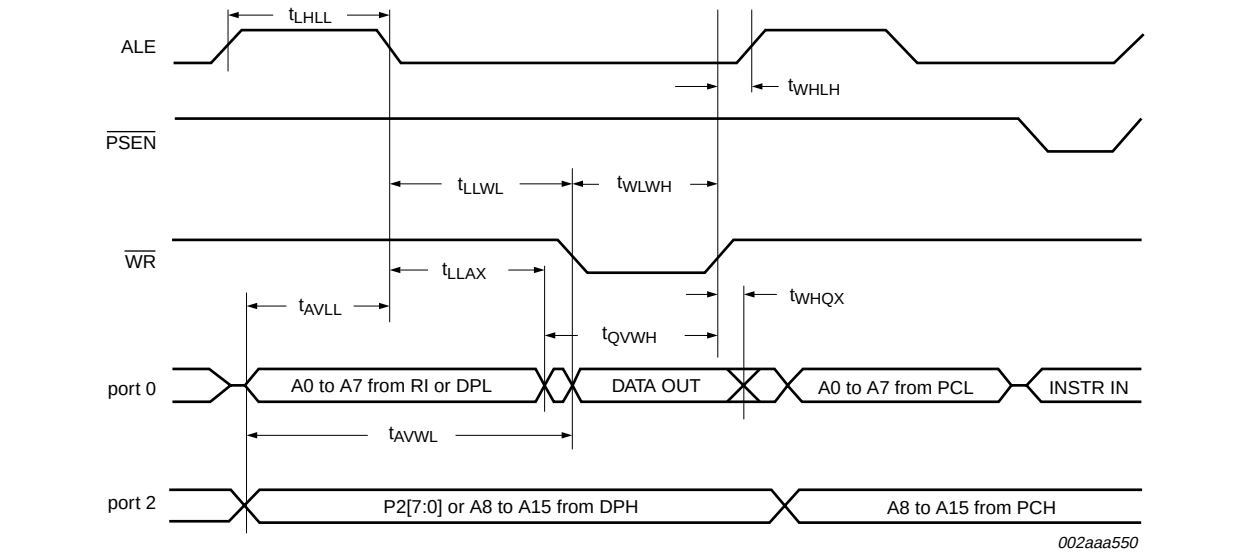


Fig 31. External data memory write cycle

Table 56. External clock drive

Symbol	Parameter	Oscillator				Unit
		40 MHz		Variable		
		Min	Max	Min	Max	
f_{osc}	oscillator frequency	-	-	0	40	MHz
$T_{cy(clk)}$	clock cycle time	25	-	-	-	ns
t_{CHCX}	clock HIGH time	8.75	-	$0.35T_{cy(clk)}$	$0.65T_{cy(clk)}$	ns
t_{CLCX}	clock LOW time	8.75	-	$0.35T_{cy(clk)}$	$0.65T_{cy(clk)}$	ns
t_{CLCH}	clock rise time	-	10	-	-	ns
t_{CHCL}	clock fall time	-	10	-	-	ns

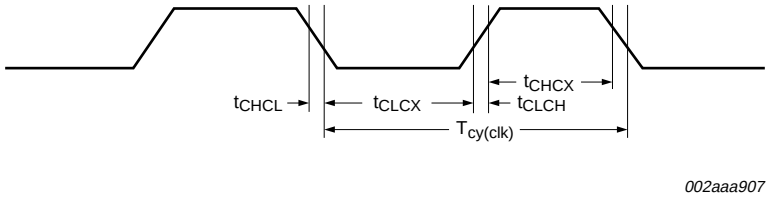


Fig 32. External clock timing (with an amplitude of at least $V_{i(RMS)}$ = 200 mV)

Table 57. Serial port timing

Symbol	Parameter	Oscillator				Unit
		40 MHz		Variable		
		Min	Max	Min	Max	
T_{XLXL}	serial port clock cycle time	0.3	-	$12T_{cy(clk)}$	-	µs
t_{QVXH}	output data set-up to clock rising edge time	117	-	$10T_{cy(clk)} - 133$	-	ns
t_{XHQX}	output data hold after clock rising edge time	0	-	$2T_{cy(clk)} - 50$	-	ns
t_{XHDX}	input data hold after clock rising edge time	0	-	0	-	ns
t_{XHDV}	input data valid to clock rising edge time	-	117	-	$10T_{cy(clk)} - 133$	ns

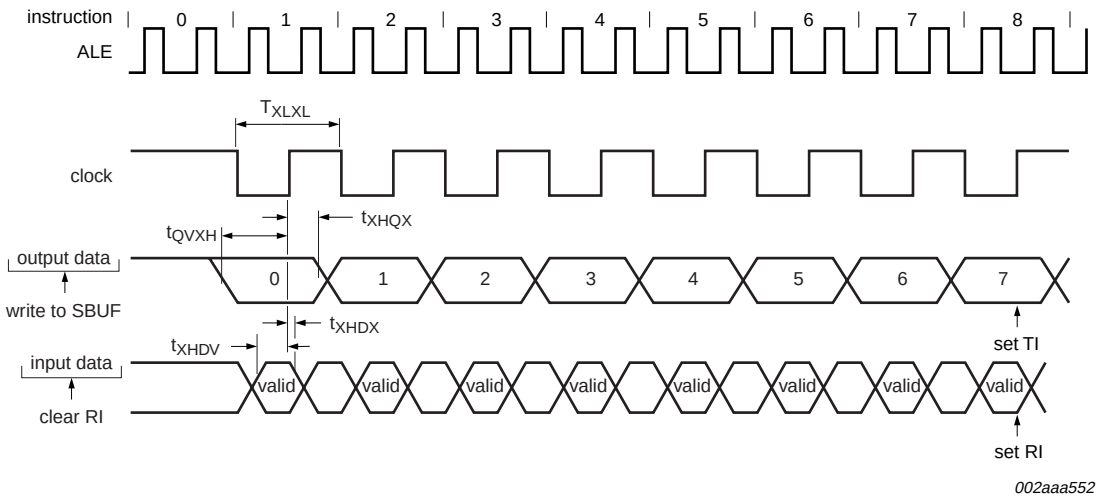


Fig 33. Shift register mode timing waveforms

Table 58. SPI interface timing

Symbol	Parameter	Conditions	Variable clock		f_{osc} = 18 MHz		Unit
			Min	Max	Min	Max	
f_{SPI}	SPI operating frequency		0	$T_{cy(clk)}$ / 4	0	10	MHz
T_{SPICYC}	SPI cycle time	see Figure 34, 35, 36, 37	$4T_{cy(clk)}$	-	222	-	ns
$t_{SPILEAD}$	SPI enable lead time	see Figure 36, 37	250	-	250	-	ns
t_{SPILAG}	SPI enable lag time	see Figure 36, 37	250	-	250	-	ns
$t_{SPICLKH}$	SPICLK HIGH time	see Figure 34, 35, 36, 37	$2T_{cy(clk)}$	-	111	-	ns
$t_{SPICLKL}$	SPICLK LOW time	see Figure 34, 35, 36, 37	$2T_{cy(clk)}$	-	111	-	ns
t_{SPIDSU}	SPI data set-up time	master or slave; see Figure 34, 35, 36, 37	100	-	100	-	ns
t_{SPIDH}	SPI data hold time	master or slave; see Figure 34, 35, 36, 37	100	-	100	-	ns
t_{SPIA}	SPI access time	see Figure 36, 37	0	80	0	80	ns
t_{SPIDIS}	SPI disable time	see Figure 36, 37	0	160	-	160	ns
t_{SPIDV}	SPI enable to output data valid time	see Figure 34, 35, 36, 37	-	111	-	111	ns
t_{SPIOH}	SPI output data hold time	see Figure 34, 35, 36, 37	0	-	0	-	ns
t_{SPIR}	SPI rise time	see Figure 34, 35, 36, 37					
		SPI outputs (SPICLK, MOSI, MISO)	-	100	-	100	ns
		SPI inputs (SPICLK, MOSI, MISO, $\overline{SS}$)	-	2000	-	2000	ns
t_{SPIF}	SPI fall time	see Figure 34, 35, 36, 37					
		SPI outputs (SPICLK, MOSI, MISO)	-	100	-	100	ns
		SPI inputs (SPICLK, MOSI, MISO, $\overline{SS}$)	-	2000	-	2000	ns

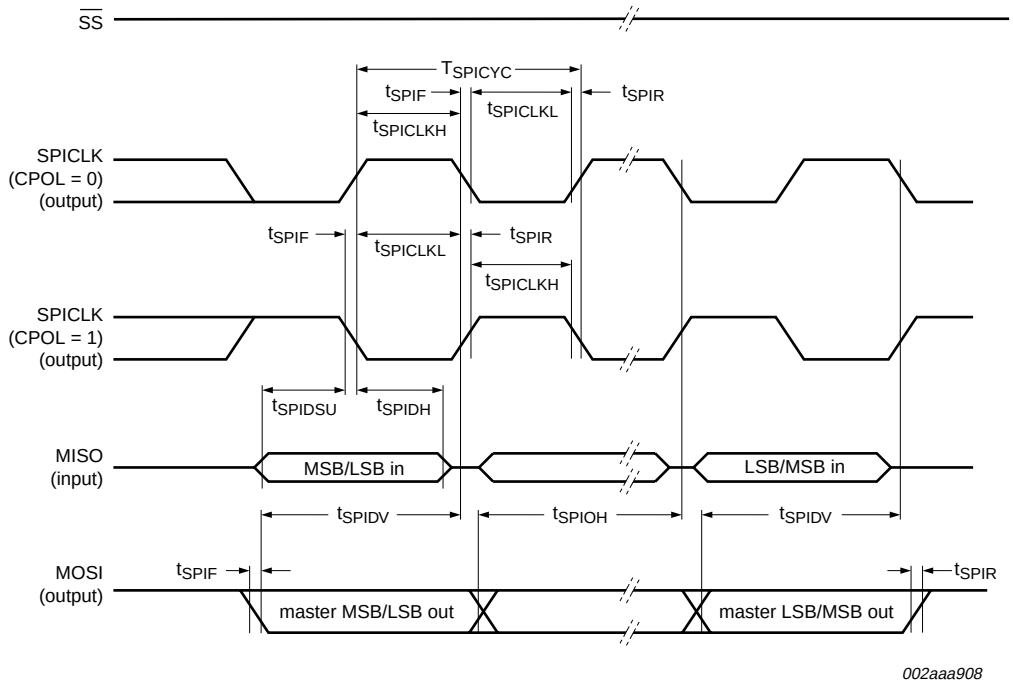


Fig 34. SPI master timing (CPHA = 0)

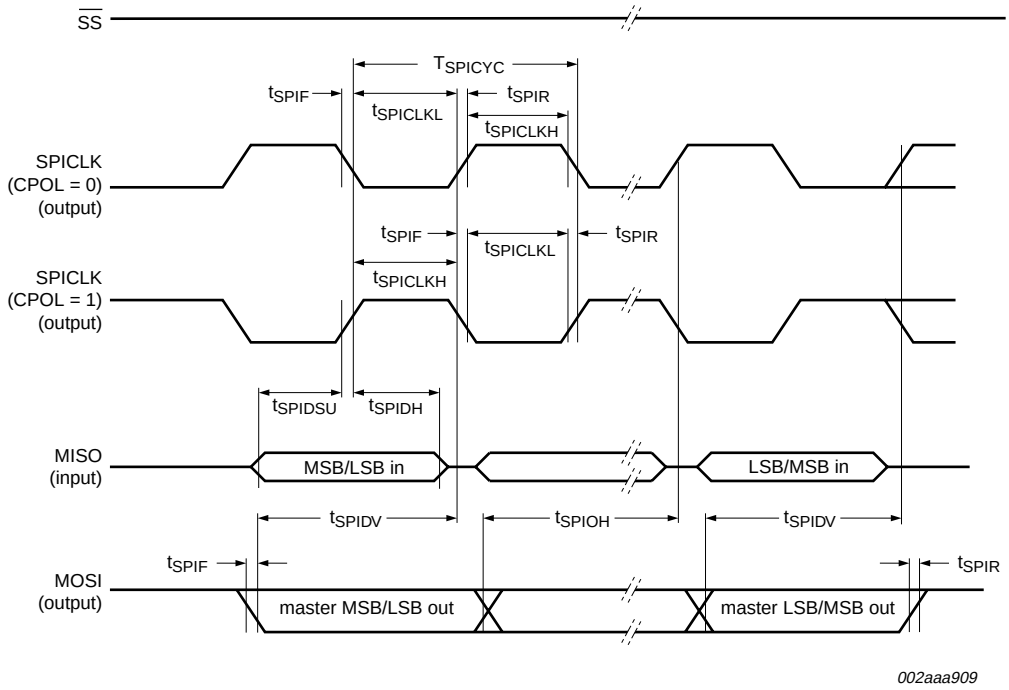


Fig 35. SPI master timing (CPHA = 1)

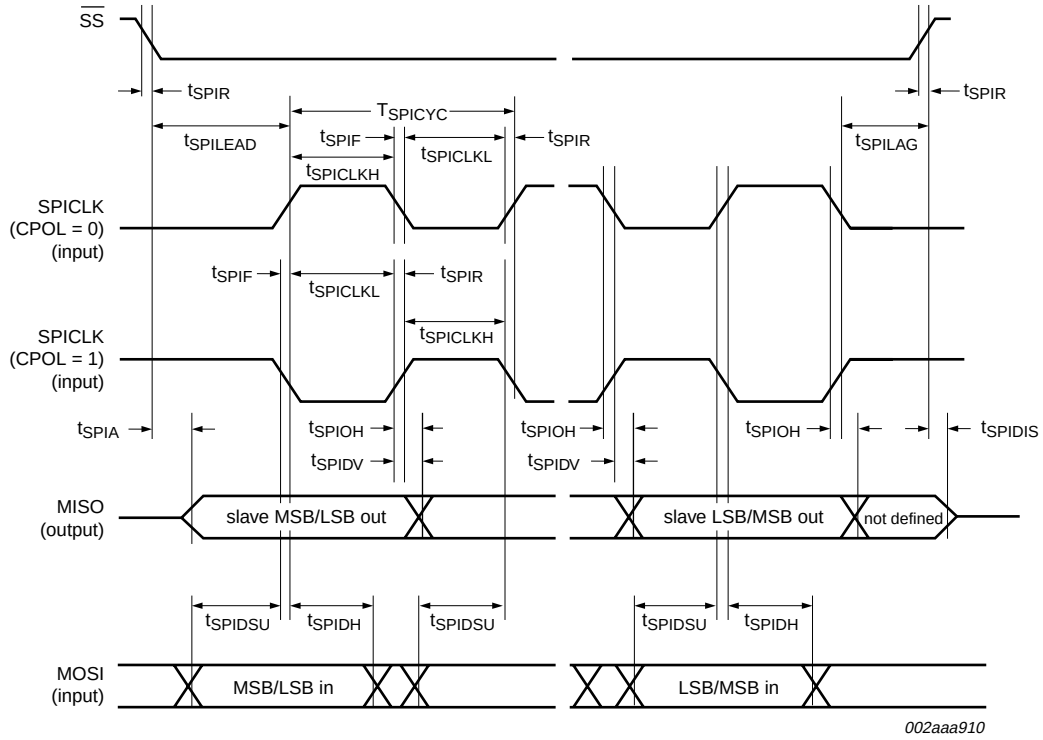


Fig 36. SPI slave timing (CPHA = 0)

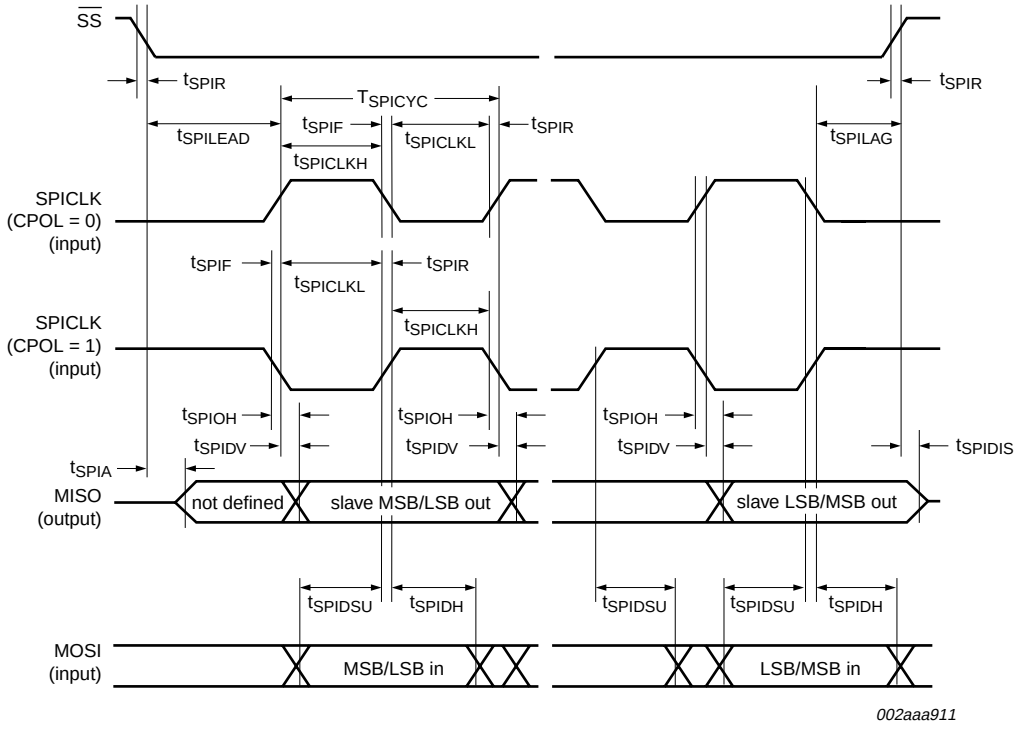


Fig 37. SPI slave timing (CPHA = 1)

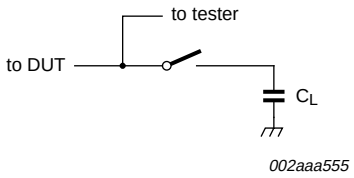
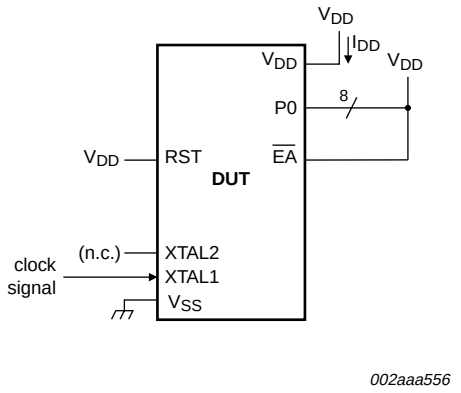
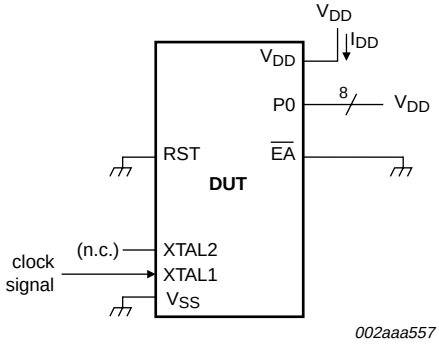


Fig 38. Test load example



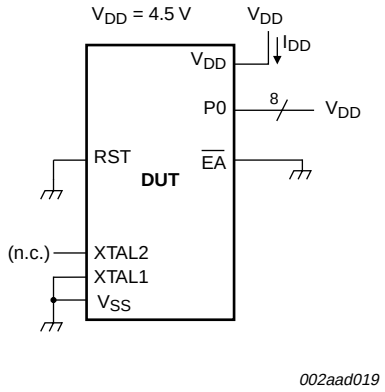
All other pins disconnected

Fig 39. I_{DD} test condition, Active mode



All other pins disconnected

Fig 40. I_{DD} test condition, Idle mode

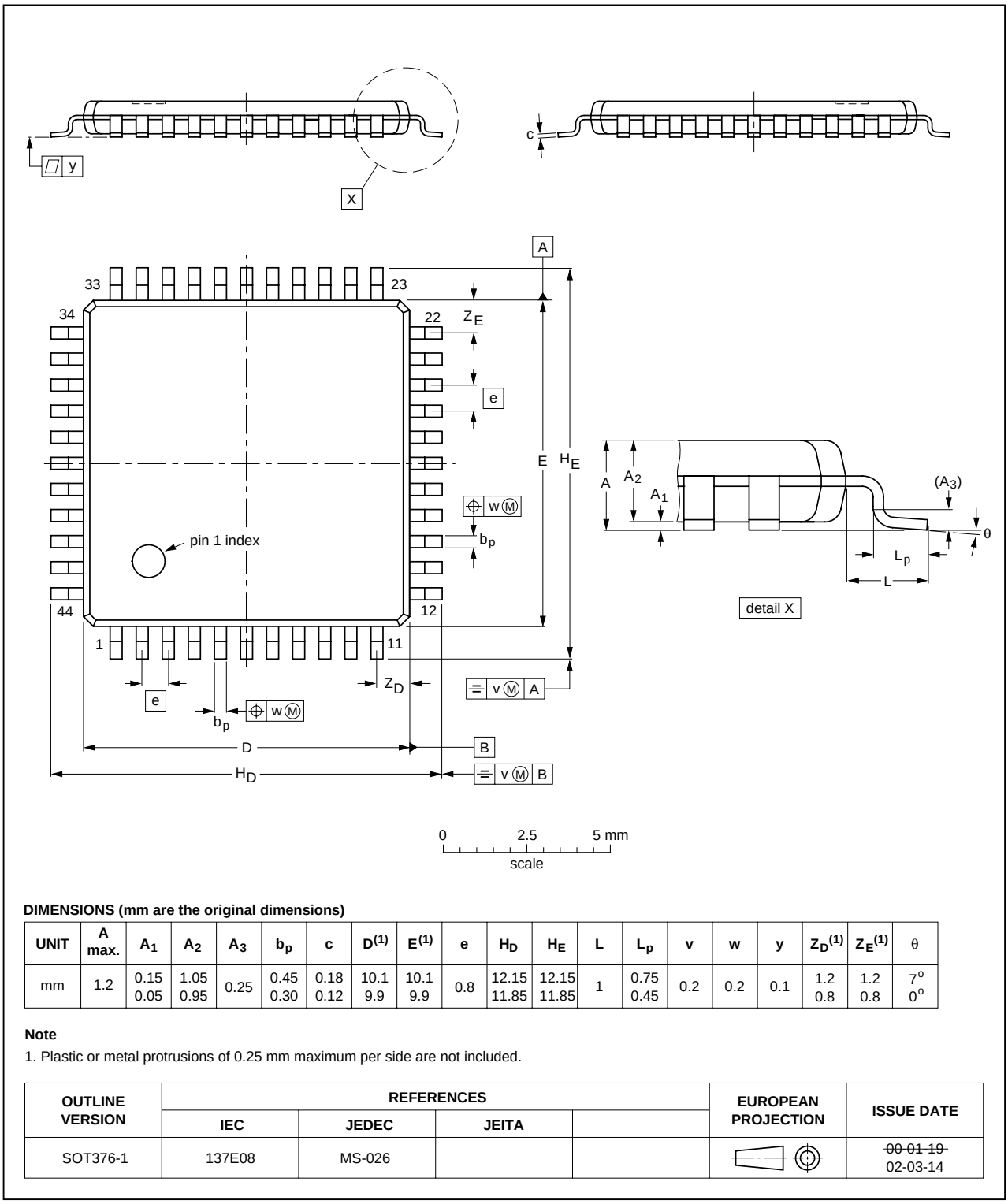


All other pins disconnected

Fig 41. I_{DD} test condition, Power-down mode

10. Package outline

TQFP44: plastic thin quad flat package; 44 leads; body 10 x 10 x 1.0 mm **SOT376-1**



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	1.2	0.15 0.05	1.05 0.95	0.25	0.45 0.30	0.18 0.12	10.1 9.9	10.1 9.9	0.8	12.15 11.85	12.15 11.85	1	0.75 0.45	0.2	0.2	0.1	1.2 0.8	1.2 0.8	7^o 0^o

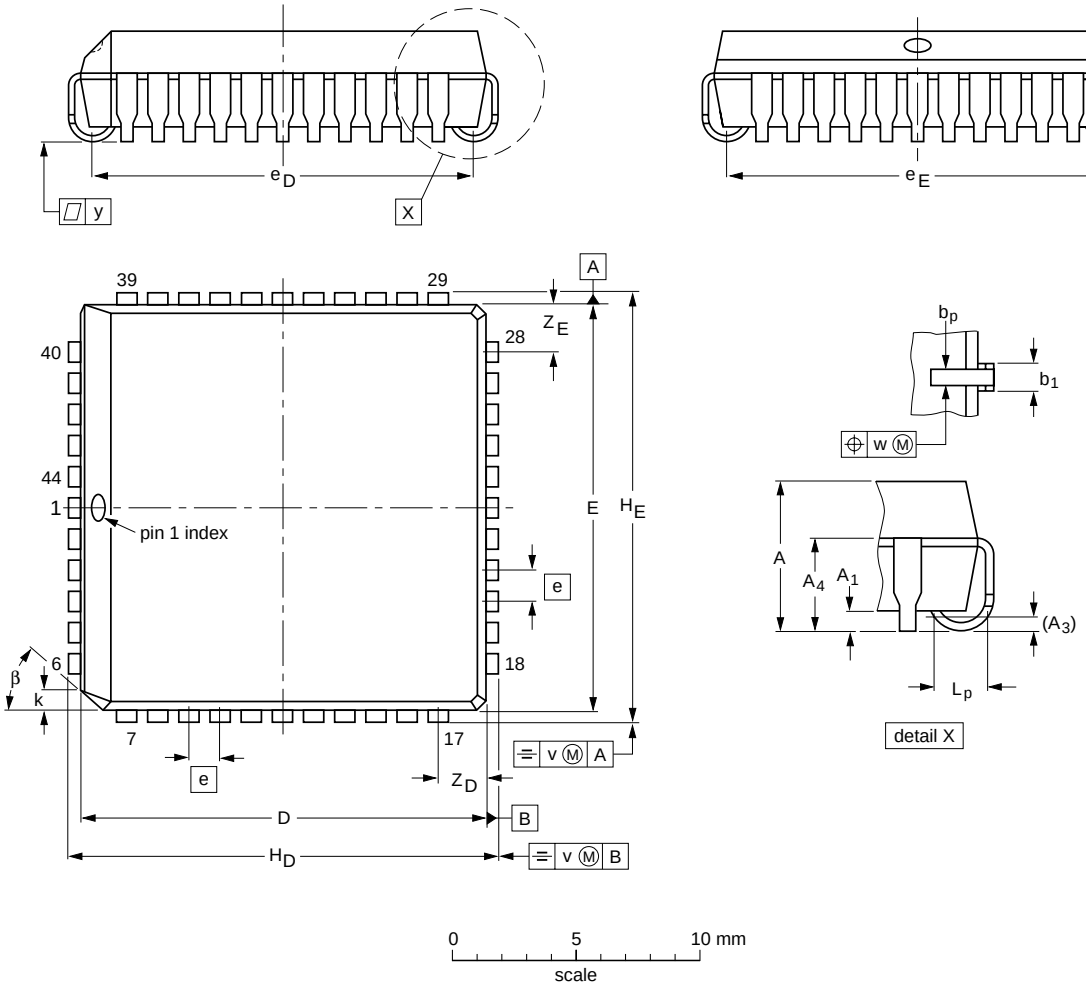
Note

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

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT376-1	137E08	MS-026				~~00-01-19~~ 02-03-14

Fig 42. Package outline SOT376-1 (TQFP44)

PLCC44: plastic leaded chip carrier; 44 leads **SOT187-2**



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

UNIT	A	A_1 min.	A_3	A_4 max.	b_p	b_1	$D^{(1)}$	$E^{(1)}$	e	e_D	e_E	H_D	H_E	k	L_p	v	w	y	$Z_D^{(1)}$ max.	$Z_E^{(1)}$ max.	β
mm	4.57 4.19	0.51	0.25	3.05	0.53 0.33	0.81 0.66	16.66 16.51	16.66 16.51	1.27	16.00 14.99	16.00 14.99	17.65 17.40	17.65 17.40	1.22 1.07	1.44 1.02	0.18	0.18	0.1	2.16	2.16	45°
inches	0.180 0.165	0.02	0.01	0.12	0.021 0.013	0.032 0.026	0.656 0.650	0.656 0.650	0.05	0.63 0.59	0.63 0.59	0.695 0.685	0.695 0.685	0.048 0.042	0.057 0.040	0.007	0.007	0.004	0.085	0.085	

Note

1. Plastic or metal protrusions of 0.25 mm (0.01 inch) maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	JEITA			
SOT187-2	112E10	MS-018	EDR-7319			~~99-12-27~~ 01-11-14

Fig 43. Package outline SOT187-2 (PLCC44)

11. Abbreviations

Table 59. Abbreviations

Acronym	Description
ALE	Address Latch Enable
CPU	Central Processing Unit
DPTR	Data PoinTeR
DUT	Device Under Test
EPROM	Erasable Programmable Read-Only Memory
EMI	Electro-Magnetic Interference
ID	IDentifier
IAP	In-Application Programming
ISP	In-System Programming
LSB	Least Significant Bit
MCU	MicroController Unit
MSB	Most Significant Bit
PCA	Programmable Counter Array
PCH	Programmable Counter High
PCL	Programmable Counter Low
PWM	Pulse-Width Modulator
RAM	Random Access Memory
RC	Resistance-Capacitance
SFR	Special Function Register
SPI	Serial Peripheral Interface
SRAM	Static Random Access Memory
UART	Universal Asynchronous Receiver/Transmitter
WDT	WatchDog Timer

12. Revision history

Table 60. Revision history

Document ID	Release date	Data sheet status	Change notice	Supersedes
P89CV51RB2_RC2_RD2_3	20090825	Product data sheet	-	P89CV51RB2_RC2_RD2_2
Modifications:	• Table 4: AUXR1, replaced '-' with 'ENBOOT'. • Table 4: CMOD, replaced 'C1H' with 'D9H'. • Table 8: Replaced '-' with 'ENBOOT'. • Table 32: Replaced 'C1H' with 'D9H'. • Table 33: Replaced 'C1H' with 'D9H'. • Table 34: Replaced 'C1H' with 'D9H'.			
P89CV51RB2_RC2_RD2_2	20090422	Product data sheet	-	P89CV51RB2_RC2_RD2_1
Modifications:	• Section 6.2.1: Corrected value for EXTRAM bit setting.			
P89CV51RB2_RC2_RD2_1	20071005	Product data sheet	-	-

13. Legal information

13.1 Data sheet status

Document status[1][2]	Product status[3]	Definition
Objective [short] data sheet	Development	This document contains data from the objective specification for product development.
Preliminary [short] data sheet	Qualification	This document contains data from the preliminary specification.
Product [short] data sheet	Production	This document contains the product specification.

[1] Please consult the most recently issued document before initiating or completing a design.

[2] The term 'short data sheet' is explained in section "Definitions".

[3] The product status of device(s) described in this document may have changed since this document was published and may differ in case of multiple devices. The latest product status information is available on the Internet at URL http://www.nxp.com.

13.2 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

Short data sheet — A short data sheet is an extract from a full data sheet with the same product type number(s) and title. A short data sheet is intended for quick reference only and should not be relied upon to contain detailed and full information. For detailed and full information see the relevant full data sheet, which is available on request via the local NXP Semiconductors sales office. In case of any inconsistency or conflict with the short data sheet, the full data sheet shall prevail.

13.3 Disclaimers

General — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) may cause permanent damage to the device. Limiting values are stress ratings only and operation of the device at these or any other conditions above those given in the Characteristics sections of this document is not implied. Exposure to limiting values for extended periods may affect device reliability.

Terms and conditions of sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, including those pertaining to warranty, intellectual property rights infringement and limitation of liability, unless explicitly otherwise agreed to in writing by NXP Semiconductors. In case of any inconsistency or conflict between information in this document and such terms and conditions, the latter will prevail.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from national authorities.

13.4 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

14. Contact information

For more information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

15. Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 25 August 2009
Document identifier: P89CV51RB2_RC2_RD2_3

Looking for pricing, stock, or lifecycle information?

Click below to explore more details on WIN SOURCE:

- View P89CV51RB2FA,512 on WIN SOURCE
- NXP / Nexperia Information

Optimize Your Supply Chain with WIN SOURCE Solutions

- Global Sourcing Solution
- Obsolete Management
- Cost Control Management
- Shortage Management
- Alternative Solution
- Excess Inventory Management