



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	40MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	32
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	4.5V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	44-PLCC (16.59x16.59)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p89v51rc2fa-512

- Brownout detection
- Low power modes
 - ◆ Power-down mode with external interrupt wake-up
 - ◆ Idle mode
- DIP40, PLCC44 and TQFP44 packages

3. Ordering information

Table 1. Ordering information

Type number	Package		Version
	Name	Description	
P89V51RB2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89V51RB2FN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1
P89V51RB2BBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89V51RC2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89V51RC2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89V51RC2FN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1
P89V51RD2FA	PLCC44	plastic leaded chip carrier; 44 leads	SOT187-2
P89V51RD2FBC	TQFP44	plastic thin quad flat package; 44 leads; body 10 × 10 × 1.0 mm	SOT376-1
P89V51RD2BN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1
P89V51RD2FN	DIP40	plastic dual in-line package; 40 leads (600 mil)	SOT129-1

3.1 Ordering options

Table 2. Ordering options

Type number	Flash memory	Temperature range	Frequency
P89V51RB2FA	16 kB	–40 °C to +85 °C	0 MHz to 40 MHz
P89V51RB2FN	16 kB	–40 °C to +85 °C	
P89V51RB2BBC	16 kB	0 °C to +70 °C	
P89V51RC2FA	32 kB	–40 °C to +85 °C	
P89V51RC2FBC	32 kB	–40 °C to +85 °C	
P89V51RC2FN	32 kB	–40 °C to +85 °C	
P89V51RD2FA	64 kB	–40 °C to +85 °C	
P89V51RD2FBC	64 kB	–40 °C to +85 °C	
P89V51RD2BN	64 kB	0 °C to +70 °C	
P89V51RD2FN	64 kB	–40 °C to +85 °C	

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

Symbol	Pin			Type	Description
	DIP40	TQFP44	PLCC44		
P2.5/A13	26	23	29	I/O	P2.5 — Port 2 bit 5.
				O	A13 — Address bit 13.
P2.6/A14	27	24	30	I/O	P2.6 — Port 2 bit 6.
				O	A14 — Address bit 14.
P2.7/A15	28	25	31	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 '1's 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. Port 3 also receives some control signals and a partial of high-order address bits during the external host mode programming and verification.
P3.0/RXD	10	5	11	I	P3.0 — Port 3 bit 0.
				I	RXD — Serial input port.
P3.1/TXD	11	7	13	O	P3.1 — Port 3 bit 1.
				O	TXD — Serial output port.
P3.2/ $\overline{\text{INT0}}$	12	8	14	I	P3.2 — Port 3 bit 2.
				I	$\overline{\text{INT0}}$ — External interrupt 0 input.
P3.3/ $\overline{\text{INT1}}$	13	9	15	I	P3.3 — Port 3 bit 3.
				I	$\overline{\text{INT1}}$ — External interrupt 1 input.
P3.4/T0	14	10	16	I/O	P3.4 — Port 3 bit 4.
				I	T0 — External count input to Timer/counter 0.
P3.5/T1	15	11	17	I/O	P3.5 — Port 3 bit 5.
				I	T1 — External count input to Timer/counter 1.
P3.6/ $\overline{\text{WR}}$	16	12	18	O	P3.6 — Port 3 bit 6.
				O	$\overline{\text{WR}}$ — External data memory write strobe.
P3.7/ $\overline{\text{RD}}$	17	13	19	O	P3.7 — Port 3 bit 7.
				O	$\overline{\text{RD}}$ — External data memory read strobe.
$\overline{\text{PSEN}}$	29	26	32	I/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. A forced HIGH-to-LOW input transition on the $\overline{\text{PSEN}}$ pin while the RST input is continually held HIGH for more than 10 machine cycles will cause the device to enter external host mode programming.

Table 4. Special function registers ...continued

* indicates SFRs that are bit addressable

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

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

DPTR points to 0A0H and data in 'A' is written to address 0A0H of the expanded RAM rather than external memory. Access to external memory higher than 2FFH using the MOVX instruction will access external memory (0300H to FFFFH) 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.

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 the 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 9 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 9. External data memory \overline{RD} , \overline{WR} with EXTRAM bit^[1]

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

[1] Access limited to ERAM address within OSPI to 0FFH; cannot access 100H to 02FFH.

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

6.3.3 ISP

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

6.3.4 Using ISP

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

```
:NNAAAARRDD..DDCC<crLf>
```

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

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

Table 12. ISP hex record formats

Record type	Command/data function
00	<p>Program User Code Memory</p> <p>:nnaaaa0dd..ddcc</p> <p>Where:</p> <p>nn = number of bytes to program</p> <p>aaaa = address</p> <p>dd..dd = data bytes</p> <p>cc = checksum</p> <p>Example:</p> <p>:100000000102030405006070809cc</p>
01	<p>End of File (EOF), no operation</p> <p>:xxxxxx01cc</p> <p>Where:</p> <p>xxxxxx = required field but value is a 'don't care'</p> <p>cc = checksum</p> <p>Example:</p> <p>:00000001FF</p>
02	<p>Set SoftICE mode</p> <p>Following the next reset the device will enter the SoftICE mode. Will erase user code memory, erase device serial number.</p> <p>:00000002cc</p> <p>Where:</p> <p>xxxxxx = required field but value is a 'don't care'</p> <p>cc = checksum</p> <p>Example:</p> <p>:00000002FE</p>

Table 12. ISP hex record formats ...continued

Record type	Command/data function
03	<p>Miscellaneous Write Functions</p> <p>:nnxxxx03ffssddcc</p> <p>Where:</p> <p>nn = number of bytes in the record</p> <p>xxxx = required field but value is a 'don't care'</p> <p>ff = subfunction code</p> <p>ss = selection code</p> <p>dd = data (if needed)</p> <p>cc = checksum</p> <p>Subfunction code = 01 (Erase block 0)</p> <p>ff = 01</p> <p>Subfunction code = 05 (Program security bit, Double Clock)</p> <p>ff = 05</p> <p>ss = 01 program security bit</p> <p>ss = 05 program double clock bit</p> <p>Subfunction code = 08 (Erase sector, 128 B)</p> <p>ff = 08</p> <p>ss = high byte of sector address (A15:8)</p> <p>dd = low byte of sector address (A7, A6:0 = 0)</p> <p>Example:</p> <p>:0300000308E000F2 (erase sector at E000H)</p>
04	<p>Display Device Data or Blank Check</p> <p>:05xxxx04ssssseeeffcc</p> <p>Where</p> <p>05 = number of bytes in the record</p> <p>xxxx = required field but value is a 'don't care'</p> <p>04 = function code for display or blank check</p> <p>ssss = starting address, MSB first</p> <p>eeee = ending address, MSB first</p> <p>ff = subfunction</p> <p>00 = display data</p> <p>01 = blank check</p> <p>cc = checksum</p> <p>Subfunction codes:</p> <p>Example:</p> <p>:0500000400001FFF00D9 (display from 0000H to 1FFFH)</p>

Table 12. ISP hex record formats ...continued

Record type	Command/data function
09	Write serial number :nnxxx09ss..sscc Where: xxxxxx = required field but value is a 'don't care' 09 = write serial number function ss..ss = serial number contents cc = checksum Example: :03000009010203EE (write s/n = 010203)
0A	Display serial number :xxxxxx0Acc Where: xxxxxx = required field but value is a 'don't care' 0A = display serial number function cc = checksum Example: :0000000AF6
0B	Reset and run user code :xxxxxx0Bcc Where: xxxxxx = required field but value is a 'don't care' 0B = Reset and run user code cc = checksum Example: :0000000BF5

6.3.5 Using the serial number

This device has the option of storing a 31 B serial number along with the length of the serial number (for a total of 32 B) in a non-volatile memory space. When ISP mode is entered, the serial number length is evaluated to determine if the serial number is in use. If the length of the serial number is programmed to either 00H or FFH, the serial number is considered not in use. If the serial number is in use, reading, programming, or erasing of the user code memory or the serial number is blocked until the user transmits a 'verify serial number' record containing a serial number and length that matches the serial number and length previously stored in the device. The user can reset the serial number to all zeros and set the length to zero by sending the 'reset serial number' record. In addition, the 'reset serial number' record will also erase all user code.

6.3.6 IAP method

Several IAP calls are available for use by an application program to permit selective erasing, reading and programming of flash sectors, security bit, configuration bytes, 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 1FF0H. The IAP calls are shown in [Table 13](#).

Table 13. IAP function calls ...continued

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

6.4 Timers/counters 0 and 1

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

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

In the 'Counter' function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled once every machine cycle.

When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register in the machine cycle following the one in which the transition was detected. Since it takes two machine cycles (12 oscillator periods) for 1-to-0 transition to be recognized, the maximum count rate is $\frac{1}{12}$ of the oscillator frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle. In addition to the 'Timer' or 'Counter' selection, Timer 0 and Timer 1 have four operating modes from which to select.

The 'Timer' or 'Counter' function is selected by control bits C/T in the Special Function Register TMOD. These two Timer/counters have four operating modes, which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timers/counters. Mode 3 is different. The four operating modes are described in the following text.

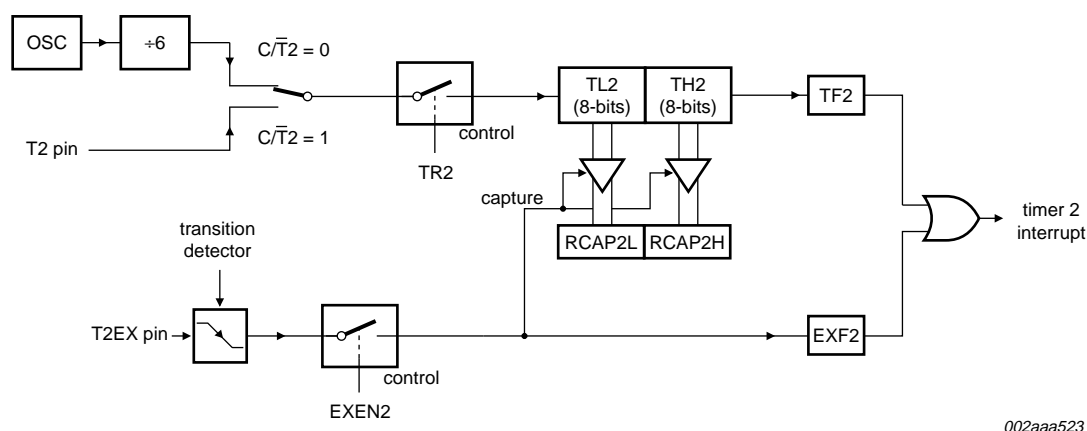


Fig 12. Timer 2 in Capture mode

This bit can be used to generate an interrupt (by enabling the Timer 2 interrupt bit in the IEN0 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.

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 Timer2 interrupt is signalled it has to be serviced before 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 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 $C/\bar{T}2$ 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 22](#) and [Table 23](#)). 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 13](#) shows Timer 2 counting up automatically (DCEN = 0).

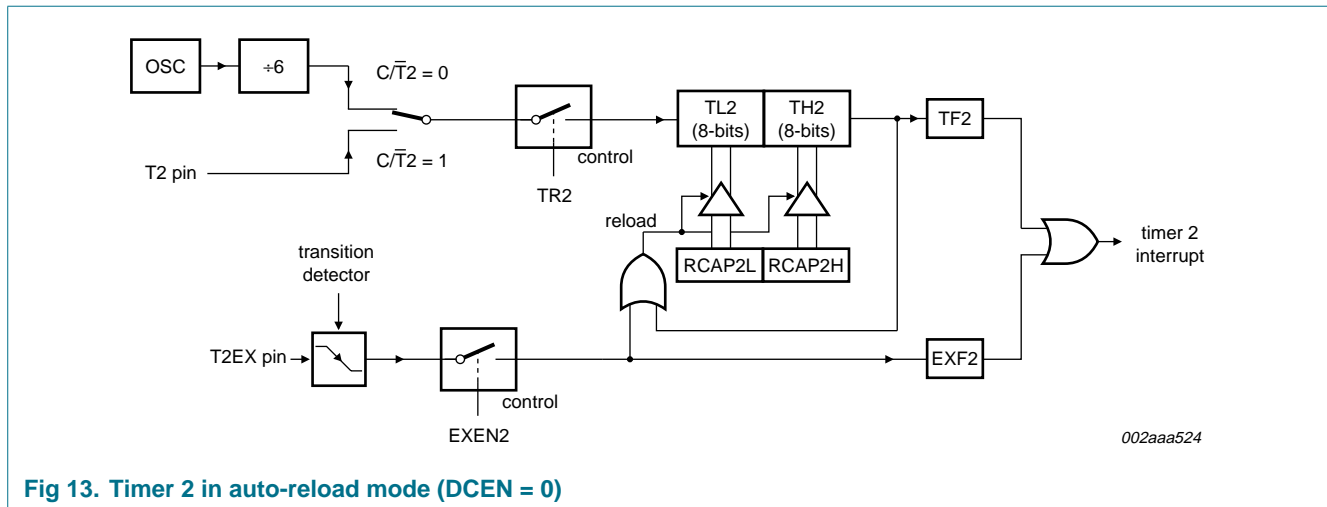


Fig 13. Timer 2 in auto-reload mode (DCEN = 0)

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 this formula:

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

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

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'.

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 14](#), 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.

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

Bit	Symbol	Description
2	RB8	In modes 2 and 3, is the 9th data bit that was received. In mode 1, if SM2 = 0, RB8 is the stop bit that was received. In mode 0, RB8 is undefined.
1	TI	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or at the stop bit in the other modes, in any serial transmission. Must be cleared by software.
0	RI	Receive interrupt flag. Set by hardware at the end of the 8th bit time in mode 0, or approximately halfway through the stop bit time in all other modes. (See SM2 for exceptions). Must be cleared by software.

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

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

6.6.5 Framing error

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

6.6.6 More about UART mode 1

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

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

The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated: (a) RI = 0, and (b) either SM2 = 0, or the received stop bit = 1.

If either of these two conditions is not met, the received frame is irretrievably lost. If both conditions are met, the stop bit goes into RB8, the 8 data bits go into SBUF, and RI is activated.

6.6.7 More about UART modes 2 and 3

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

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

Bit	Symbol	Description
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 30 below.
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 30 below.

Table 30. SPCR - SPI control register (address D5H) clock rate selection

SPR1	SPR0	SPICLK = f_{osc} divided by
0	0	4
0	1	16
1	0	64
1	1	128

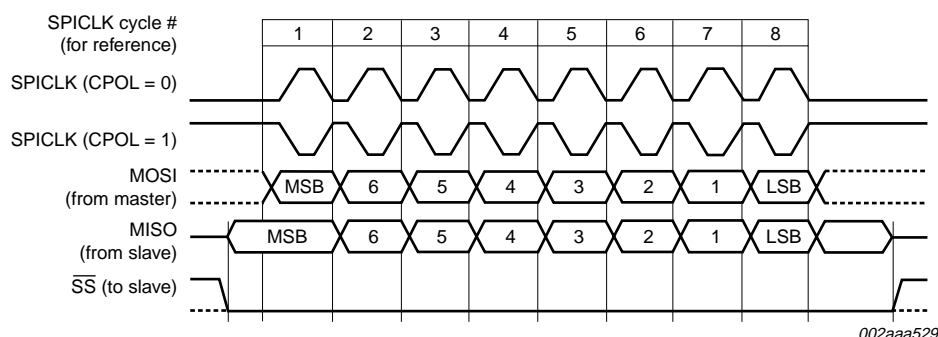
Table 31. SPSR - SPI status register (address AAH) bit allocation

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

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

Table 32. 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 18. SPI transfer format with CPHA = 0**

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 (CCF4:0). 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 22](#).

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 22](#)).

PWM (CCAPMn.1) enables the pulse width modulation mode.

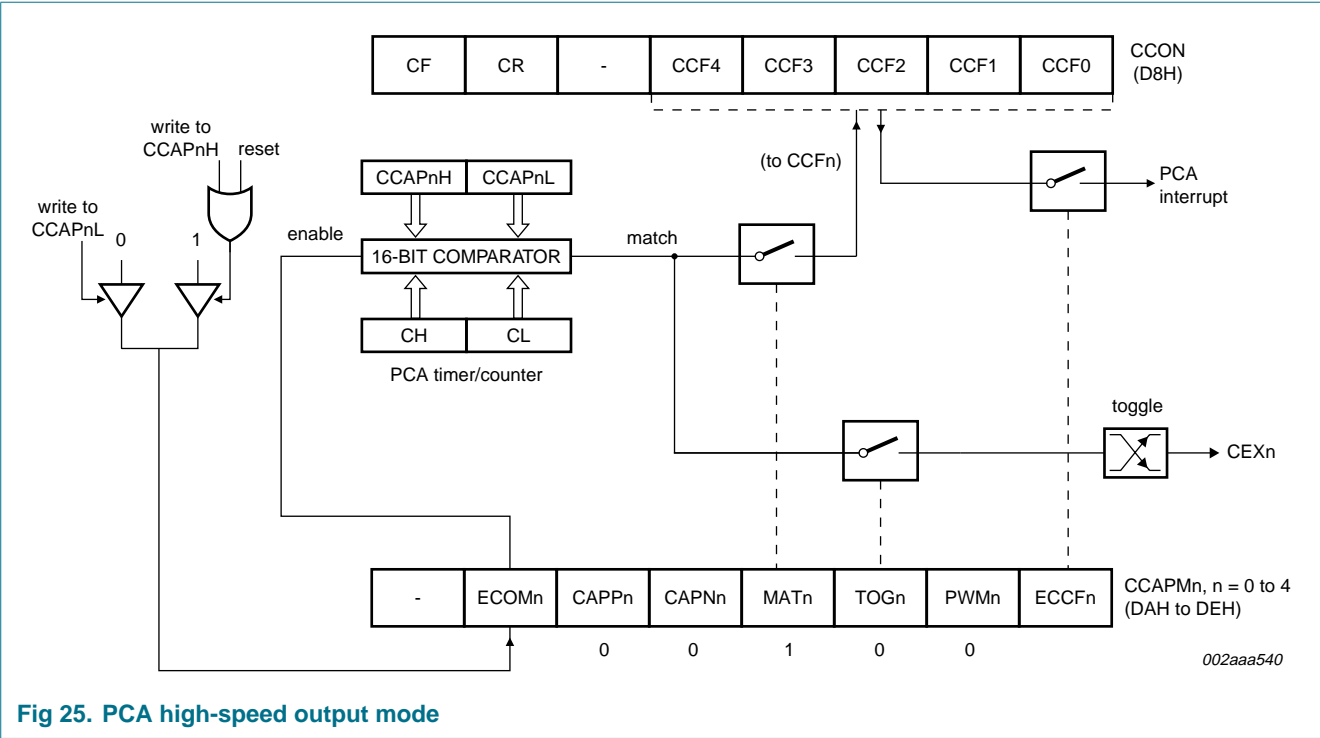
The TOG bit (CCAPMn.2) when set causes the CEX output associated with the module to toggle when there is a match between the PCA counter and the module's capture/compare register.

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.

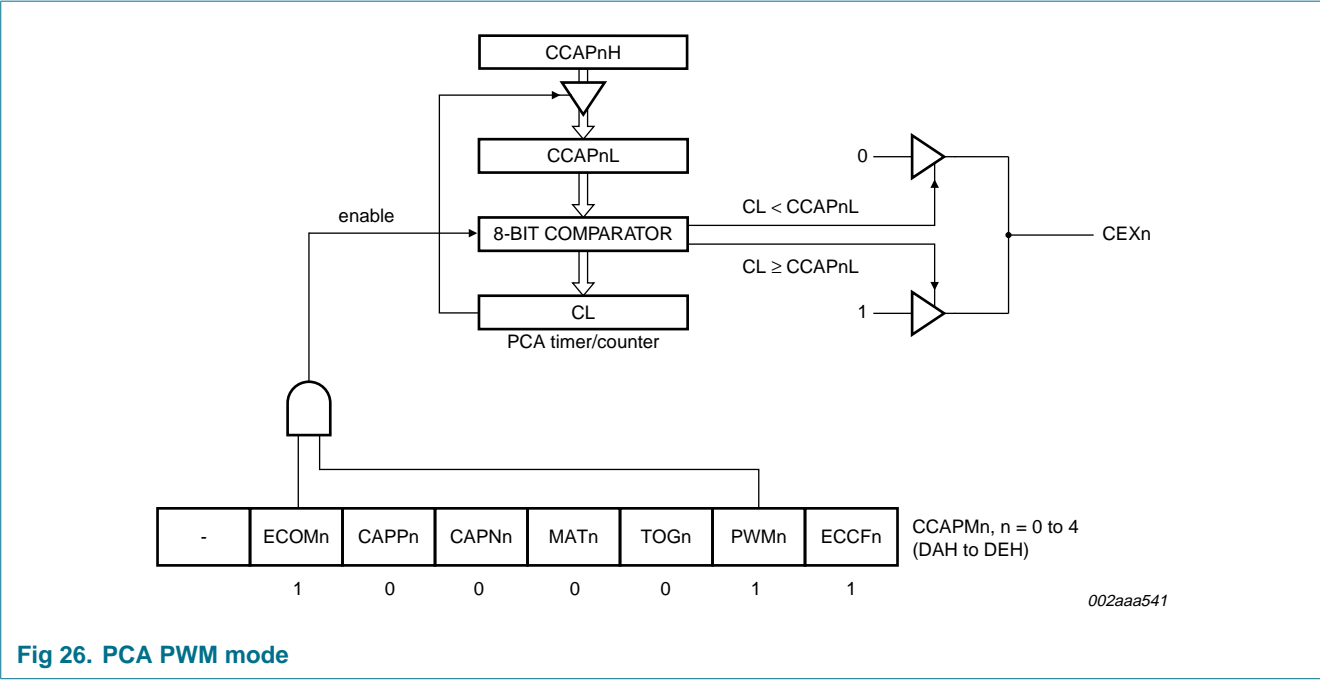
The last bit in the register ECOM (CCAPMn.6) when set enables the comparator function.

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.



6.9.4 PWM mode

All of the PCA modules can be used as PWM outputs (Figure 26). Output frequency depends on the source for the PCA timer.



All of the modules will have the same frequency of output because they all share one and only 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 than the output will be high. When CL overflows from FF to 00, CCAPnL is reloaded with the value in CCAPnH. This allows updating the PWM without glitches. The PWM and ECOM bits in the module's CCAPMn register must be set to enable the 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 26](#) 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 WATCHDOG routine shown above.

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

1. Periodically change the compare value so it will never match the PCA timer.
2. Periodically change the PCA timer value so it will never match the compare values.
3. 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 option #3. If the program counter ever goes astray, 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, the PCA timer is the time base for **all** modules; changing the time base for other modules would not be a good idea. Thus, in most applications the first solution is the best option.

```
;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
```

This routine should not be part of an interrupt service routine, because if the program counter goes astray and gets stuck in an infinite loop, interrupts will still be serviced and the Watchdog will keep getting reset. Thus, the purpose of the Watchdog would be defeated. Instead, call this subroutine from the main program within 2^{16} count of the PCA timer.

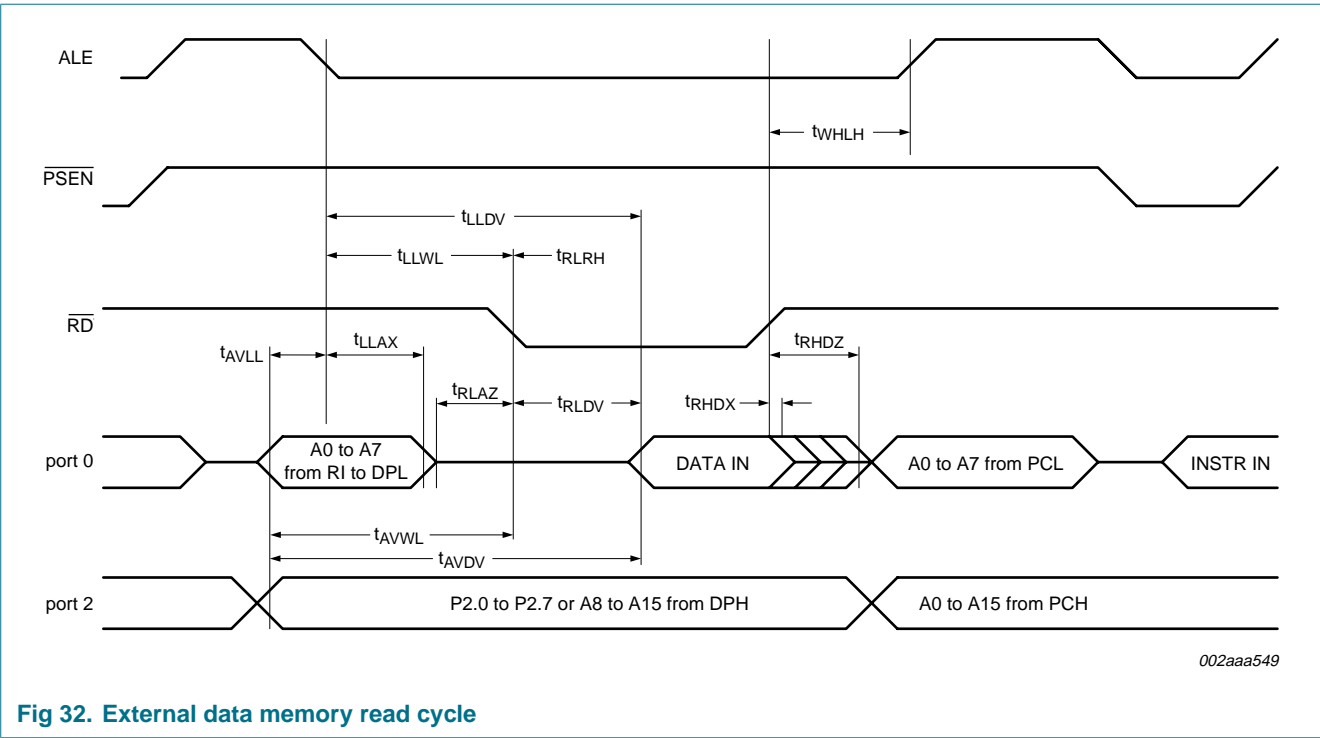


Fig 32. External data memory read cycle

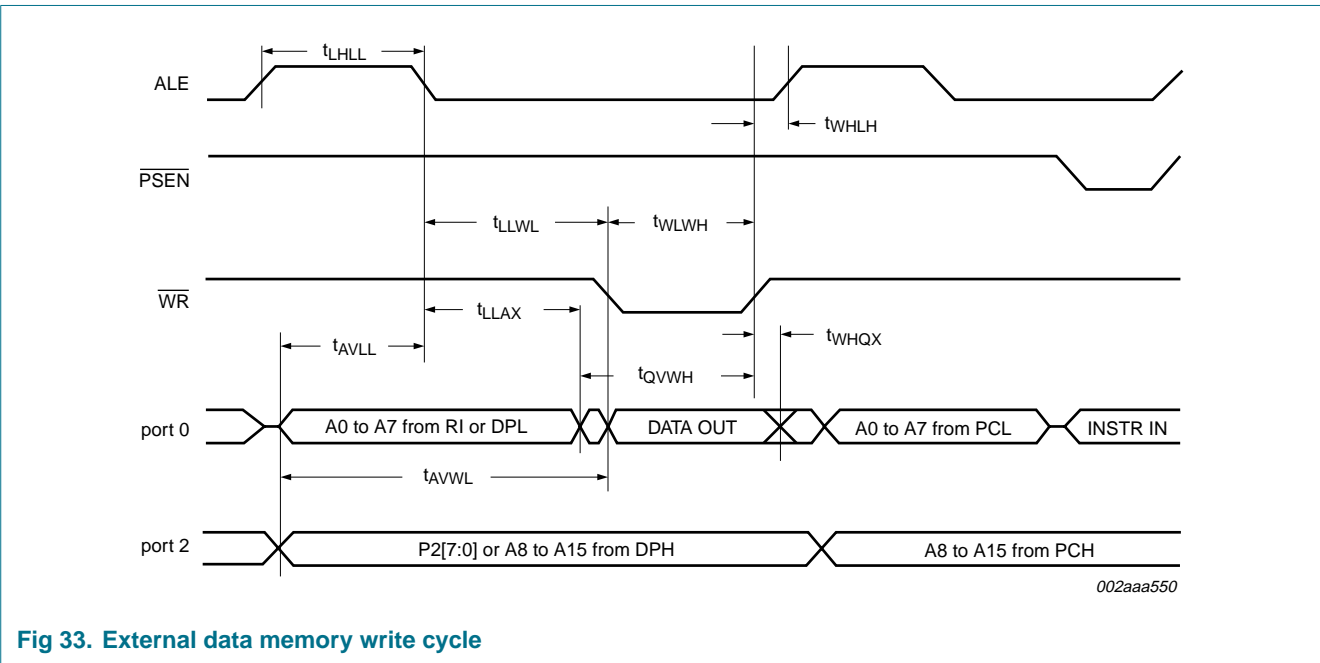
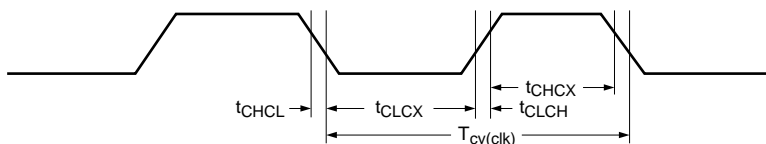


Fig 33. External data memory write cycle

Table 64. 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



002aaa907

Fig 34. External clock drive waveform (with an amplitude of at least $V_{i(RMS)} = 200$ mV)

Table 65. 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

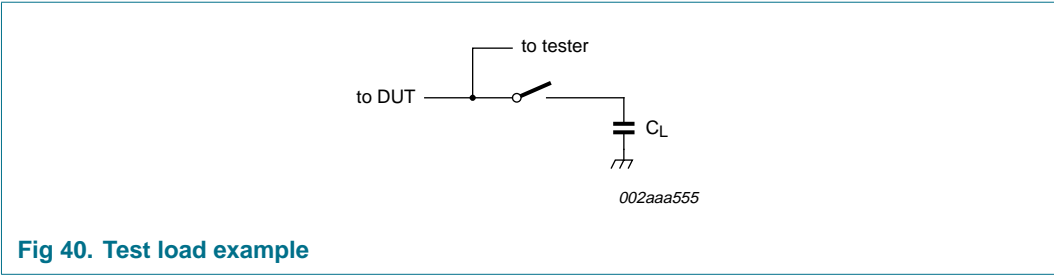
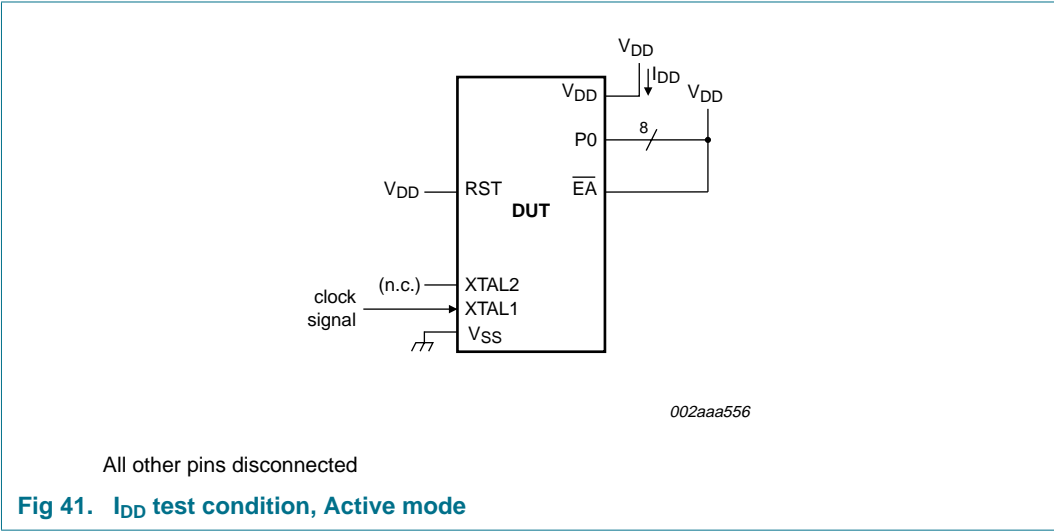
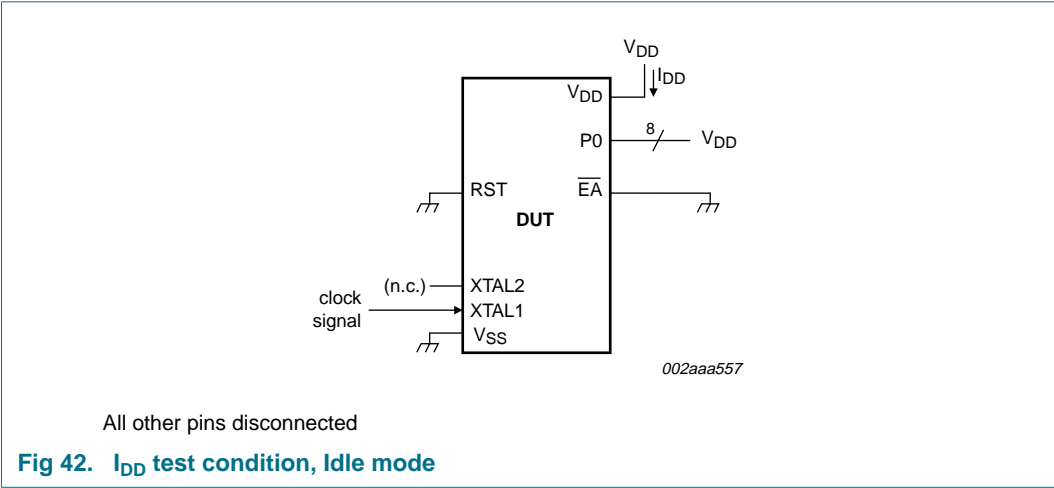


Fig 40. Test load example



All other pins disconnected

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



All other pins disconnected

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

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