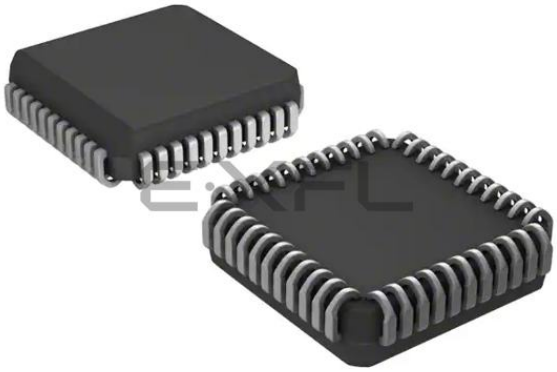


**Welcome to E-XFL.COM**

## What is "Embedded - Microcontrollers"?

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

## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Obsolete |
| Core Processor | 8051 |
| Core Size | 8-Bit |
| Speed | 33MHz |
| 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 | 512 x 8 |
| Voltage - Supply (Vcc/Vdd) | 2.7V ~ 3.6V |
| 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/p89lv51rc2fa-512 |

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

** Indicates SFRs that are bit addressable.*

| Name | Description | SFR addr. | 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 | PSC1 | PSC0 |
| SPCFG | SPI Configuration Register | AAH | SPIF | WCOL | - | - | - | - | - | - |
| 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/$\overline{T}$2 | 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 | T1GATE | T1C/T | T1M1 | T1M0 | T0GATE | T0C/T | T0M1 | T0M0 |
| 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.

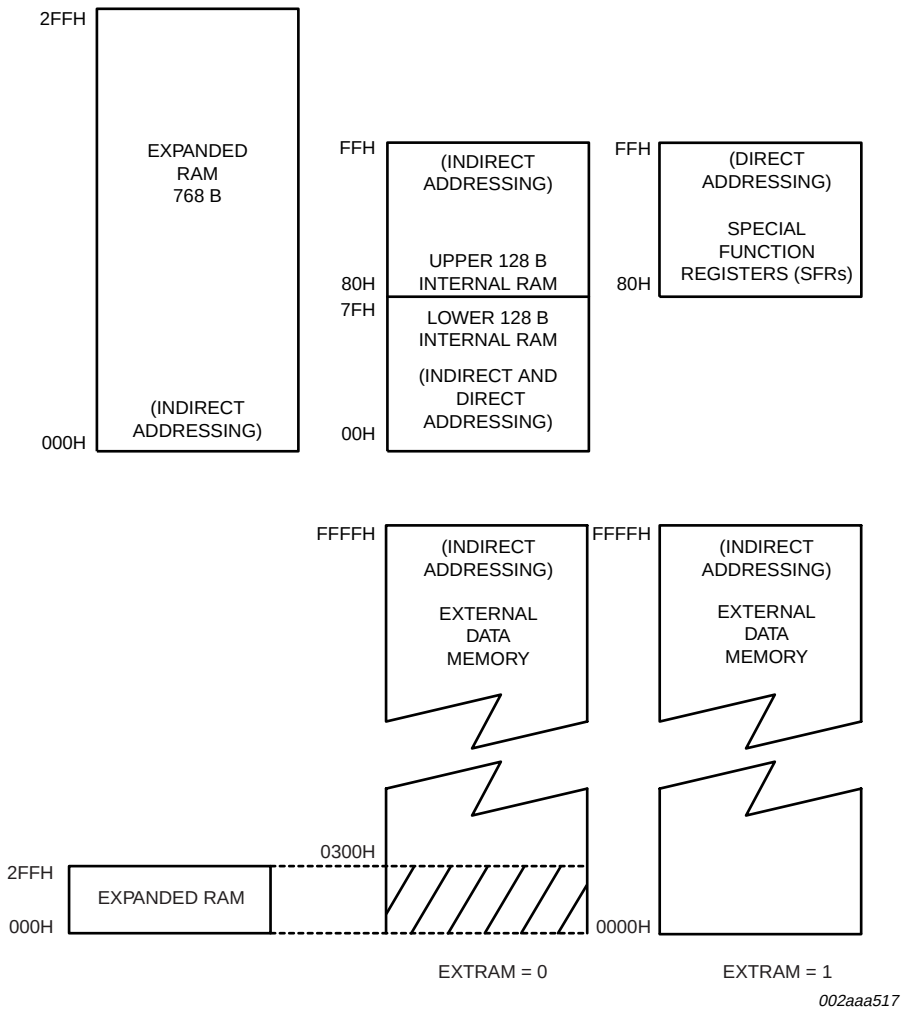



**Fig 5. Internal and external data memory structure**

### 6.2.8 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 6).

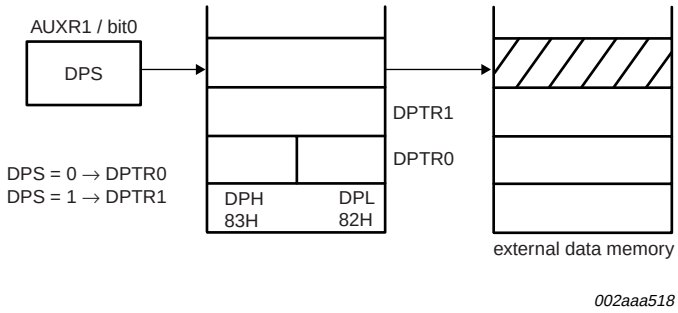



**Fig 6. Dual data pointer organization**

**Table 10. AUXR1 - Auxiliary register 1 (address A2H) bit allocation**

*Not bit addressable; reset value 00H.*

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

**Table 11. AUXR1 - Auxiliary register 1 (address A2H) bit descriptions**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 4 | - | Reserved for future use. Should be set to ‘0’ by user programs. |
| 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. |

## 6.3 Flash memory IAP

### 6.3.1 Flash organization

The P89LV51RB2/RC2/RD2 program memory consists of a 16/32/64 kB block. ISP capability, in a second 8 kB block, is provided to allow the user code to be programmed in-circuit through the serial port. There are three methods of erasing or programming of the flash memory that may be used. First, the flash may be programmed or erased in the end-user application by calling low-level routines through a common entry point (IAP). Second, the on-chip ISP bootloader may be invoked. This ISP bootloader will, in turn, call low-level 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 Boot block (block 1)

When the microcontroller programs its own flash memory, all of the low level details are handled by code that is contained in block 1. A user program calls the common entry point in the block 1 with appropriate parameters to accomplish the desired operation. Boot block operations include erase user code, program user code, program security bits, etc.

**Table 12. ISP hex record formats**

| Record type | Command/data function |
|---|---|
| 00 | Program User Code Memory<br>:nnaaaa00dd..ddcc<br>Where:<br>nn = number of bytes to program<br>aaaa = address<br>dd..dd = data bytes<br>cc = checksum<br>Example:<br>:10000000010203040500607080 9cc |
| 01 | End of File (EOF), no operation<br>:xxxxxx01cc<br>Where:<br>xxxxxx = required field but value is a ‘don’t care’<br>cc = checksum<br>Example:<br>:00000001FF |
| 02 | Set SoftICE mode<br>Following the next reset the device will enter the SoftICE mode. Will erase user code memory, and erase device serial number.<br>:00000002cc<br>Where:<br>xxxxxx = required field but value is a ‘don’t care’<br>cc = checksum<br>Example:<br>:00000002FE |

**Table 13. IAP function calls**

| IAP function | IAP call parameters |
|---|---|
| Read ID | **Input parameters:**<br>R1 = 00H<br>DPH = 00H<br>DPL = 00H = mfgr id<br>DPL = 01H = device id 1<br>DPL = 02H = boot code version number<br>**Return parameter(s):**<br>ACC = requested parameter |
| Erase block 0 | **Input parameters:**<br>R1 = 01H<br>**Return parameter(s):**<br>ACC = 00 = pass<br>ACC = !00 = fail |
| Program User Code | **Input parameters:**<br>R1 = 02H<br>DPH = memory address MSB<br>DPL = memory address LSB<br>ACC = byte to program<br>**Return parameter(s):**<br>ACC = 00 = pass<br>ACC = !00 = fail |
| Read User Code | **Input parameters:**<br>R1 = 03H<br>DPH = memory address MSB<br>DPL = memory address LSB<br>**Return parameter(s):**<br>ACC = device data |
| Program Security Bit, Double Clock | **Input parameters:**<br>R1 = 05H<br>DPL = 01H = security bit<br>DPL = 05H = Double Clock<br>**Return parameter(s):**<br>ACC = 00 = pass<br>ACC = !00 = fail |

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.

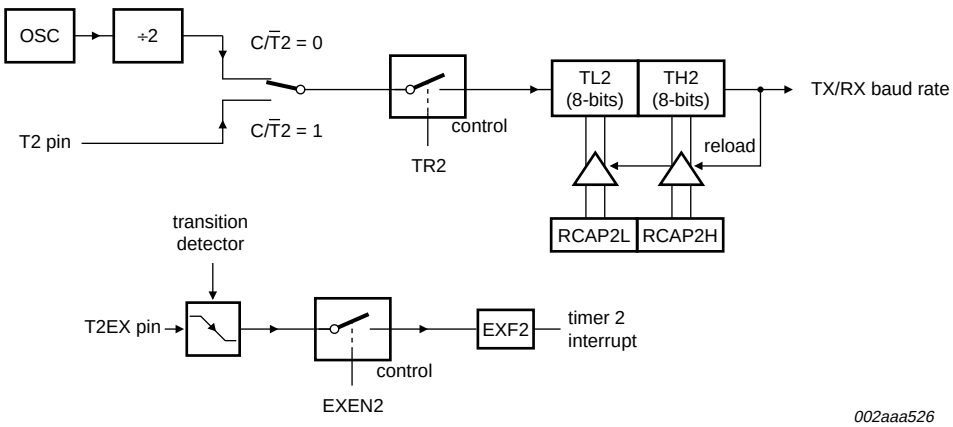



**Fig 14. Timer 2 in Baud Rate Generator mode**

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

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

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

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

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

Modes 1 and 3 baud rates =

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

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

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

**Table 26. SCON - Serial port control register (address 98H) bit descriptions** *…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 definitions**

| 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 37. CMOD - PCA counter mode register (address D9H) count pulse select**

| CPS1 | CPS0 | Select PCA input |
|---|---|---|
| 0 | 0 | 0 Internal clock, $f_{osc}$ / 6 |
| 0 | 1 | 1 Internal clock, $f_{osc}$ / 2 |
| 1 | 0 | 2 Timer 0 overflow |
| 1 | 1 | 3 External clock at ECI/P1.2 pin (maximum rate = $f_{osc}$ / 4) |

**Table 38. CCON - PCA counter control register (address 0D8H) 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 39. CCON - PCA counter control register (address 0D8H) bit descriptions**

| 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 Bit. 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 40. CCAPMn - PCA modules compare/capture register (address CCAPM0: 0DAH, CCAPM1: 0DBH, CCAPM2: 0DCH, CCAPM3: 0DDH, CCAPM4: 0DEH) 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 41. CCAPMn - PCA modules compare/capture register (address CCAPM0: 0DAH, CCAPM1: 0DBH, CCAPM2: 0DCH, CCAPM3: 0DDH, CCAPM4: 0DEH) bit descriptions**

| 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. |

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 PWMn and ECOMn 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 25 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 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 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 are still serviced and the Watchdog continues to 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.

## 6.10 Security bit

The Security Bit protects against software piracy and prevents the contents of the flash from being read by unauthorized parties in Parallel Programmer mode. It also protects against code corruption resulting from accidental erasing and programming to the internal flash memory.

When the Security Bit is activated, all parallel programming commands except for Chip-Erase are ignored (thus the device cannot be read). However, ISP reading, writing, or erasing of the user's code can still be performed if the serial number and length has not been programmed. **Therefore, when a user requests to program the Security Bit, the programmer should prompt the user and program a serial number into the device.**

## 6.11 Interrupt priority and polling sequence

The device supports eight interrupt sources under a four level priority scheme. Table 43 summarizes the polling sequence of the supported interrupts. Note that the SPI serial interface and the UART share the same interrupt vector. (See Figure 26).

**Table 43. Interrupt polling sequence**

| Description | Interrupt flag | Vector address | Interrupt enable bit | Interrupt priority bit | Service priority | Wake-up power-down |
|---|---|---|---|---|---|---|
| External interrupt 0 | IE0 | 0003H | EX0 | PX0/H | 1 (highest) | yes |
| Brownout | - | 004BH | EBO | PBO/H | 2 | no |
| T0 | TF0 | 000BH | ET0 | PT0/H | 3 | no |
| External interrupt 1 | IE1 | 0013H | EX1 | PX1/H | 4 | yes |
| T1 | TF1 | 001BH | ET1 | PT1/H | 5 | no |
| PCA | CF/CCFn | 0033H | EC | PPCH | 6 | no |
| UART/SPI | TI/RI/SPIF | 0023H | ES | PS/H | 7 | no |
| T2 | TF2, EXF2 | 002BH | ET2 | PT2/H | 8 | no |

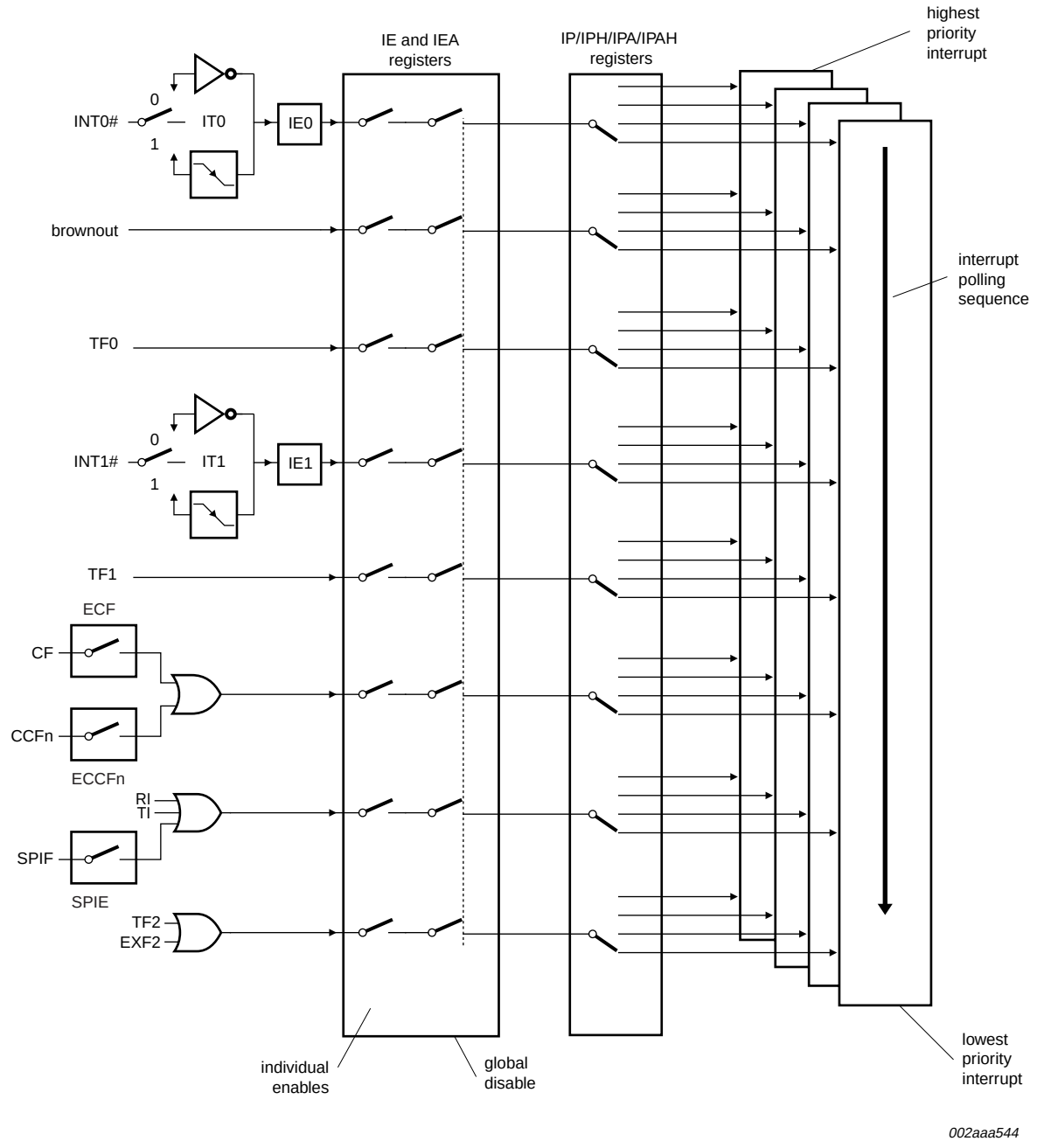



**Fig 26. Interrupt structure**

**Table 44. IEN0 - 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 45. IEN0 - Interrupt enable register 0 (address A8H) bit descriptions**

| Bit | Symbol | Description |
|---|---|---|
| 7 | EA | Interrupt Enable Bit: EA = 1 interrupt(s) can be serviced, EA = 0 interrupt servicing disabled. |
| 6 | EC | PCA Interrupt Enable bit. |
| 5 | ET2 | Timer 2 Interrupt Enable. |
| 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 46. IEN1 - Interrupt enable register 1 (address E8H) bit allocation**

*Bit addressable; reset value: 00H.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | - | - | - | - | EBO | - | - | - |

**Table 47. IEN1 - Interrupt enable register 1 (address E8H) bit descriptions**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 4 | - | Reserved for future use. Should be set to ‘0’ by user programs. |
| 3 | EBO | Brownout Interrupt Enable. 1 = enable, 0 = disable. |
| 2 to 0 | - | Reserved for future use. Should be set to ‘0’ by user programs. |

**Table 48. IP0 - Interrupt priority 0 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 49. IP0 - Interrupt priority 0 low register (address B8H) bit descriptions**

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

**Table 50. IP0H - Interrupt priority 0 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 51. IP0H - Interrupt priority 0 high register (address B7H) bit descriptions**

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

**Table 52. IP1 - Interrupt priority 1 register (address F8H) bit allocation**

*Bit addressable; reset value: 00H.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | - | - | - | PBO | - | - | - | - |

**Table 53. IP1 - Interrupt priority 1 register (address F8H) bit descriptions**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 5 | - | Reserved for future use. Should be set to ‘0’ by user programs. |
| 4 | PBO | Brownout interrupt priority bit. |
| 3 to 0 | - | Reserved for future use. Should be set to ‘0’ by user programs. |

**Table 54. IP1H - Interrupt priority 1 high register (address F7H) bit allocation**

*Not bit addressable; reset value: 00H.*

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Symbol | - | - | - | PBOH | - | - | - | - |

**Table 55. IP1H - Interrupt priority 1 high register (address F7H) bit descriptions**

| Bit | Symbol | Description |
|---|---|---|
| 7 to 5 | - | Reserved for future use. Should be set to ‘0’ by user programs. |
| 4 | PBOH | Brownout interrupt priority bit. |
| 3 to 0 | - | Reserved for future use. Should be set to ‘0’ by user programs. |

## 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 56.

### 6.12.1 Idle mode

Idle mode is entered by setting the IDL bit in the PCON register. In Idle mode, the Program Counter (PC) 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.

# 7. Limiting values

**Table 61. 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 | pins P1.5, P1.6, P1.7 | - | 20 | mA |
| | | all other pins | - | 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 62. Static characteristics**

*$T_{amb}$ = 0 °C to +70 °C or –40 °C to +85 °C; $V_{DD}$ = 2.7 V to 3.6 V; $V_{SS}$ = 0 V.*

| Symbol | Parameter | Conditions | | Min | 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 |
| $V_{IL}$ | LOW-level input voltage | 2.7 V < $V_{DD}$ < 3.6 V | | –0.5 | +0.7 | V |
| $V_{IH}$ | HIGH-level input voltage | 2.7 V < $V_{DD}$ < 3.6 V | | 0.2$V_{DD}$ + 0.9 | $V_{DD}$ + 0.5 | V |
| | | XTAL1, RST | | 0.7$V_{DD}$ | $V_{DD}$ + 0.5 | V |
| $V_{OL}$ | LOW-level output voltage | $V_{DD}$ = 2.7 V; ports 1.5, 1.6, 1.7 | | | | |
| | | $I_{OL}$ = 16 mA | | - | 1.0 | V |
| | | $V_{DD}$ = 2.7 V; ports 1, 2, 3, except $\overline{PSEN}$, ALE | [2][3][4] | | | |
| | | $I_{OL}$ = 100 µA | | - | 0.3 | V |
| | | $I_{OL}$ = 1.6 mA | | - | 0.45 | V |
| | | $I_{OL}$ = 3.5 mA | | - | 1.0 | V |
| | | $V_{DD}$ = 2.7 V; port 0, $\overline{PSEN}$, ALE | | | | |
| | | $I_{OL}$ = 200 µA | | - | 0.3 | V |
| | | $I_{OL}$ = 3.2 mA | | - | 0.45 | V |

# 9. Dynamic characteristics

**Table 63. Dynamic characteristics**

*Over operating conditions: load capacitance for Port 0, ALE, and $\overline{PSEN}$ = 100 pF; load capacitance for all other outputs = 80 pF; $T_{amb}$ = 0 °C to +70 °C or −40 °C to +85 °C; $V_{DD}$ = 2.7 V to 3.6 V at 33 MHz; $V_{SS}$ = 0 V.*[1][2]

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $f_{osc}$ | oscillator frequency | X1 mode | 0 | - | 33 | MHz |
| | | X2 mode | 0 | - | 16 | MHz |
| | | IAP | 0.25 | - | 33 | MHz |
| $t_{LHLL}$ | ALE pulse width | | $2T_{cy(clk)} - 15$ | - | - | ns |
| $t_{AVLL}$ | address valid to ALE LOW time | | $T_{cy(clk)} - 25$ | - | - | ns |
| $t_{LLAX}$ | address hold after ALE LOW time | | $T_{cy(clk)} - 25$ | - | - | ns |
| $t_{LLIV}$ | ALE LOW to valid instruction in time | | - | - | $4T_{cy(clk)} - 65$ | ns |
| $t_{LLPL}$ | ALE LOW to $\overline{PSEN}$ LOW time | | $T_{cy(clk)} - 25$ | - | - | ns |
| $t_{PLPH}$ | $\overline{PSEN}$ pulse width | | $T_{cy(clk)} - 25$ | - | - | ns |
| $t_{PLIV}$ | $\overline{PSEN}$ LOW to valid instruction in time | | - | - | $3T_{cy(clk)} - 55$ | ns |
| $t_{PXIX}$ | input instruction hold after $\overline{PSEN}$ time | | 0 | - | - | ns |
| $t_{PXIZ}$ | input instruction float after $\overline{PSEN}$ time | | - | - | $T_{cy(clk)} - 5$ | ns |
| $t_{PXAV}$ | $\overline{PSEN}$ to address valid time | | $T_{cy(clk)} - 8$ | - | - | ns |
| $t_{AVIV}$ | address to valid instruction in time | | - | - | $5T_{cy(clk)} - 80$ | ns |
| $t_{PLAZ}$ | $\overline{PSEN}$ LOW to address float time | | - | - | 10 | ns |
| $t_{RLRH}$ | $\overline{RD}$ LOW pulse width | | $6T_{cy(clk)} - 40$ | - | - | ns |
| $t_{WLWH}$ | $\overline{WR}$ LOW pulse width | | $6T_{cy(clk)} - 40$ | - | - | ns |
| $t_{RLDV}$ | $\overline{RD}$ LOW to valid data in time | | - | - | $5T_{cy(clk)} - 90$ | ns |
| $t_{RHDX}$ | data hold after $\overline{RD}$ time | | 0 | - | - | ns |
| $t_{RHDZ}$ | data float after $\overline{RD}$ time | | - | - | $2T_{cy(clk)} - 25$ | ns |
| $t_{LLDV}$ | ALE LOW to valid data in time | | - | - | $8T_{cy(clk)} - 90$ | ns |
| $t_{AVDV}$ | address to valid data in time | | - | - | $9T_{cy(clk)} - 90$ | ns |
| $t_{LLWL}$ | ALE LOW to $\overline{RD}$ or $\overline{WR}$ LOW time | | $3T_{cy(clk)} - 25$ | - | $3T_{cy(clk)} + 25$ | ns |
| $t_{AVWL}$ | address to $\overline{RD}$ or $\overline{WR}$ LOW time | | $4T_{cy(clk)} - 75$ | - | - | ns |
| $t_{WHQX}$ | data hold after $\overline{WR}$ time | | $T_{cy(clk)} - 27$ | - | - | ns |
| $t_{QVWH}$ | data output valid to $\overline{WR}$ HIGH time | | $7T_{cy(clk)} - 70$ | - | - | ns |
| $t_{RLAZ}$ | $\overline{RD}$ LOW to address float time | | - | - | 0 | ns |
| $t_{WHLH}$ | $\overline{RD}$ or $\overline{WR}$ HIGH to ALE HIGH time | | $T_{cy(clk)} - 25$ | - | $T_{cy(clk)} + 25$ | ns |

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

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

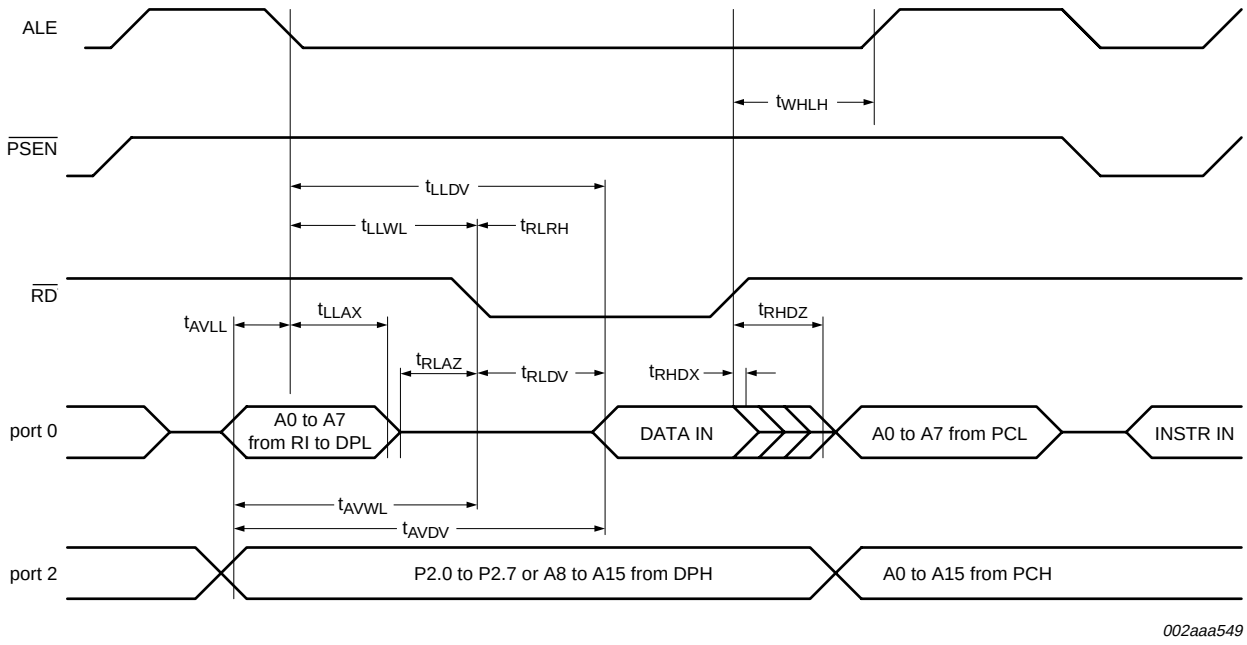



**Fig 30. External data memory read cycle**

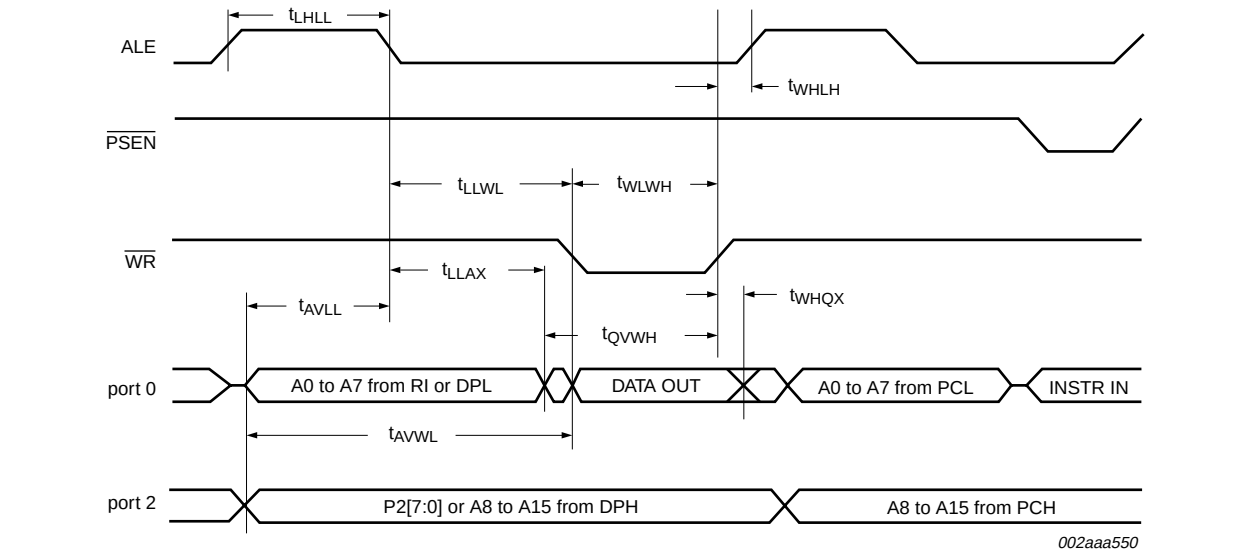
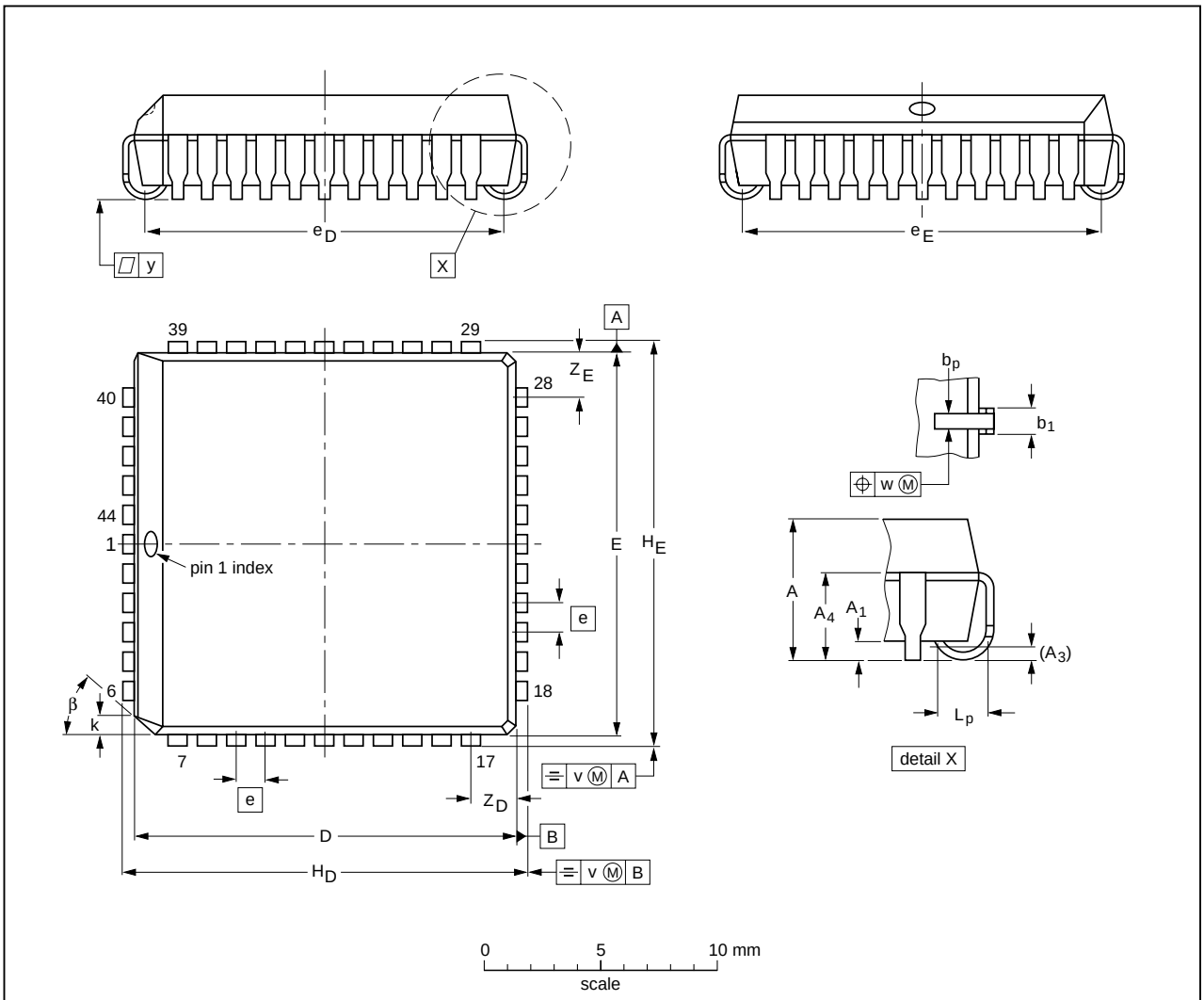



**Fig 31. External data memory write cycle**

# 10. Package outline

**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<br>4.19 | 0.51 | 0.25 | 3.05 | 0.53<br>0.33 | 0.81<br>0.66 | 16.66<br>16.51 | 16.66<br>16.51 | 1.27 | 16.00<br>14.99 | 16.00<br>14.99 | 17.65<br>17.40 | 17.65<br>17.40 | 1.22<br>1.07 | 1.44<br>1.02 | 0.18 | 0.18 | 0.1 | 2.16 | 2.16 | 45° |
| inches | 0.180<br>0.165 | 0.02 | 0.01 | 0.12 | 0.021<br>0.013 | 0.032<br>0.026 | 0.656<br>0.650 | 0.656<br>0.650 | 0.05 | 0.63<br>0.59 | 0.63<br>0.59 | 0.695<br>0.685 | 0.695<br>0.685 | 0.048<br>0.042 | 0.057<br>0.040 | 0.007 | 0.007 | 0.004 | 0.085 | 0.085 | |

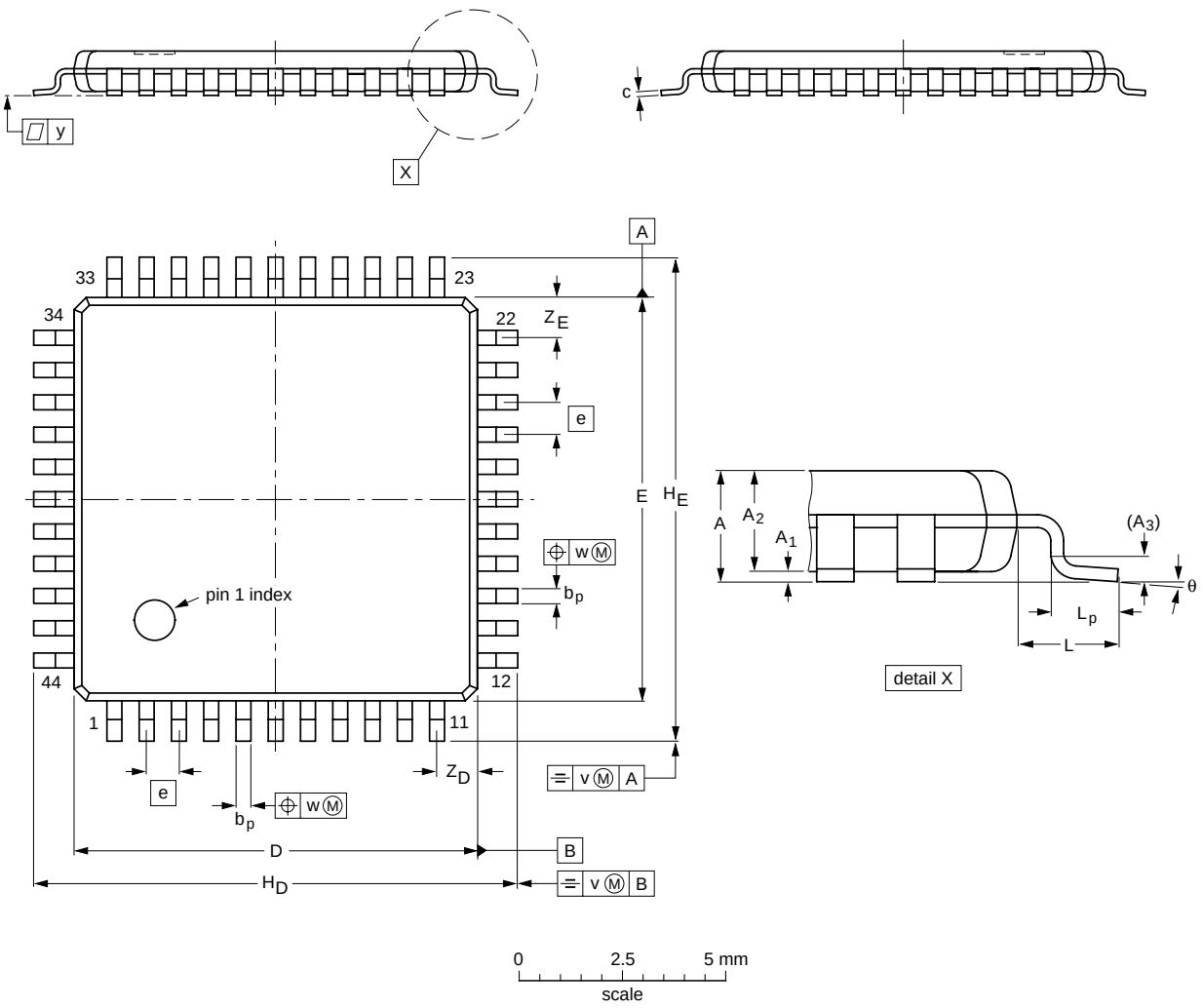
**Note**

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~~<br>01-11-14 |

**Fig 38. Package outline SOT187-2 (PLCC44)**

**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<br>0.05 | 1.05<br>0.95 | 0.25 | 0.45<br>0.30 | 0.18<br>0.12 | 10.1<br>9.9 | 10.1<br>9.9 | 0.8 | 12.15<br>11.85 | 12.15<br>11.85 | 1 | 0.75<br>0.45 | 0.2 | 0.2 | 0.1 | 1.2<br>0.8 | 1.2<br>0.8 | $7^o$<br>$0^o$ |

**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~~<br>02-03-14 |

**Fig 39. Package outline SOT376-1 (TQFP44)**

## 12. Revision history

**Table 67. Revision history**

| Document ID | Release date | Data sheet status | Change notice | Supersedes |
|---|---|---|---|---|
| P89LV51RB2_RC2_RD2_5 | 20091215 | Product data sheet | - | P89LV51RB2_RC2_RD2-04 |
| Modifications: | • The format of this data sheet has been redesigned to comply with the new identity guidelines of NXP Semiconductors.<br>• Legal texts have been adapted to the new company name where appropriate.<br>• Table 3: Removed sentence “However, Security lock level 4 will disable $\overline{EA}$...” for $\overline{EA}$ description.<br>• Table 37: Second row, changed “$f_{osc}$ / 6” to “$f_{osc}$ / 2”.<br>• Figure 30: Updated figure.<br>• Figure 31: Updated figure. | | | |
| P89LV51RB2_RC2_RD2-04 (9397 750 14342) | 20041202 | Product data | - | P89LV51RB2_RC2_RD2-03 |
| P89LV51RB2_RC2_RD2-03 (9397 750 14101) | 20041011 | Product data | - | P89LV51RB2_RC2_RD2-02 |
| P89LV51RB2_RC2_RD2-02 (9397 750 11783) | 20031113 | Product data | - | P89LV51RB2_RC2_RD2-01 |
| P89LV51RB2_RC2_RD2-01 (9397 750 11669) | 20030630 | Product data | ECN 853-2432 30075 dated 27 June 2003 | - |

## 15. Contents


**1 General description . . . . . 1**
**2 Features . . . . . 1**
**3 Ordering information . . . . . 2**
3.1 Ordering options . . . . . 2
**4 Block diagram . . . . . 3**
**5 Pinning information . . . . . 4**
5.1 Pinning . . . . . 4
5.2 Pin description . . . . . 6
**6 Functional description . . . . . 10**
6.1 Special function registers . . . . . 10
6.2 Memory organization . . . . . 14
6.2.1 Flash program memory bank selection . . . . . 14
6.2.2 Power-on reset code execution . . . . . 14
6.2.3 Software reset . . . . . 15
6.2.4 Brownout detect reset . . . . . 15
6.2.5 Watchdog reset . . . . . 16
6.2.6 Data RAM memory . . . . . 16
6.2.7 Expanded data RAM addressing . . . . . 16
6.2.8 Dual data pointers . . . . . 19
6.3 Flash memory IAP . . . . . 20
6.3.1 Flash organization . . . . . 20
6.3.2 Boot block (block 1) . . . . . 20
6.3.3 ISP . . . . . 21
6.3.4 Using ISP . . . . . 21
6.3.5 Using the serial number . . . . . 25
6.3.6 IAP method . . . . . 25
6.4 Timers/counters 0 and 1 . . . . . 27
6.4.1 Mode 0 . . . . . 29
6.4.2 Mode 1 . . . . . 29
6.4.3 Mode 2 . . . . . 30
6.4.4 Mode 3 . . . . . 30
6.5 Timer 2 . . . . . 31
6.5.1 Capture mode . . . . . 32
6.5.2 Auto-reload mode (up or down counter) . . . . . 33
6.5.3 Programmable clock-out . . . . . 35
6.5.4 Baud rate generator mode . . . . . 35
6.5.5 Summary of baud rate equations . . . . . 37
6.6 UART . . . . . 37
6.6.1 Mode 0 . . . . . 37
6.6.2 Mode 1 . . . . . 38
6.6.3 Mode 2 . . . . . 38
6.6.4 Mode 3 . . . . . 38
6.6.5 Framing error . . . . . 39
6.6.6 More about UART mode 1 . . . . . 39
6.6.7 More about UART modes 2 and 3 . . . . . 39
6.6.8 Multiprocessor communications . . . . . 40
6.6.9 Automatic address recognition . . . . . 40
6.7 SPI . . . . . 42
6.7.1 SPI features . . . . . 42
6.7.2 SPI description . . . . . 42
6.8 Watchdog timer . . . . . 45
6.9 PCA . . . . . 46
6.9.1 PCA capture mode . . . . . 50
6.9.2 16-bit software timer mode . . . . . 51
6.9.3 High-speed output mode . . . . . 52
6.9.4 PWM mode . . . . . 53
6.9.5 PCA watchdog timer . . . . . 54
6.10 Security bit . . . . . 55
6.11 Interrupt priority and polling sequence . . . . . 55
6.12 Power-saving modes . . . . . 58
6.12.1 Idle mode . . . . . 58
6.12.2 Power-down mode . . . . . 59
6.13 System clock and clock options . . . . . 60
6.13.1 Clock input options and recommended capacitor values for oscillator . . . . . 60
6.13.2 Clock doubling option . . . . . 60
**7 Limiting values . . . . . 62**
**8 Static characteristics . . . . . 62**
**9 Dynamic characteristics . . . . . 65**
9.1 Explanation of symbols . . . . . 66
**10 Package outline . . . . . 71**
**11 Abbreviations . . . . . 73**
**12 Revision history . . . . . 74**
**13 Legal information . . . . . 75**
13.1 Data sheet status . . . . . 75
13.2 Definitions . . . . . 75
13.3 Disclaimers . . . . . 75
13.4 Trademarks . . . . . 75
**14 Contact information . . . . . 75**
**15 Contents . . . . . 76**




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

**Date of release: 15 December 2009**
**Document identifier: P89LV51RB2_RC2_RD2_5**