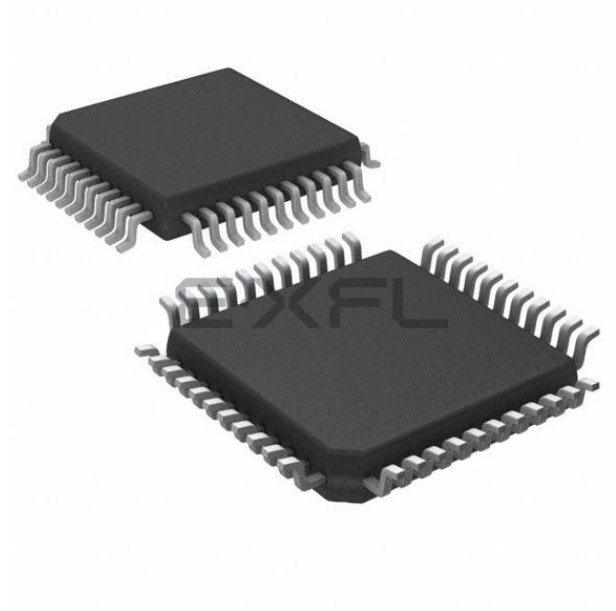


**Welcome to E-XFL.COM**

## 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.

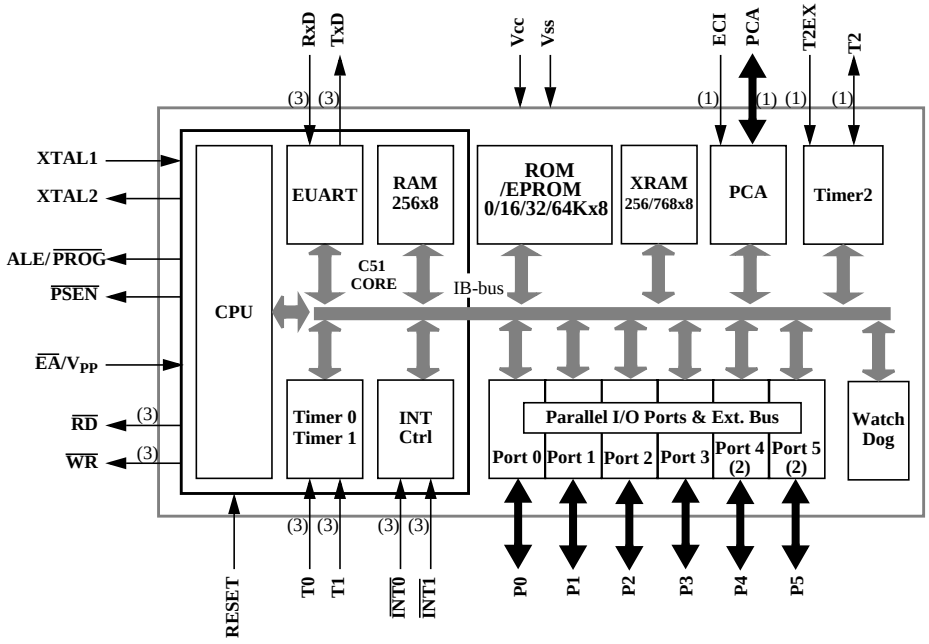
## Applications of "Embedded - Microcontrollers"

| Details | |
|---|---|
| Product Status | Obsolete |
| Core Processor | 80C51 |
| Core Size | 8-Bit |
| Speed | 40/20MHz |
| Connectivity | UART/USART |
| Peripherals | POR, PWM, WDT |
| Number of I/O | 32 |
| Program Memory Size | - |
| Program Memory Type | ROMless |
| EEPROM Size | - |
| RAM Size | 768 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-QFP |
| Supplier Device Package | 44-VQFP |
| Purchase URL | https://www.e-xfl.com/product-detail/microchip-technology/ts80c51rd2-mie |

| PDIL40<br>PLCC44<br>VQFP44 1.4 | ROM (bytes) | EPROM (bytes) | XRAM (bytes) | TOTAL RAM (bytes) | I/O |
|---|---|---|---|---|---|
| **TS80C51RA2** | 0 | 0 | 256 | 512 | 32 |
| **TS80C51RD2** | 0 | 0 | 768 | 1024 | 32 |
| **TS83C51RB2** | 16k | 0 | 256 | 512 | 32 |
| **TS83C51RC2** | 32k | 0 | 256 | 512 | 32 |
| **TS83C51RD2** | 64k | 0 | 768 | 1024 | 32 |
| **TS87C51RB2** | 0 | 16k | 256 | 512 | 32 |
| **TS87C51RC2** | 0 | 32k | 256 | 512 | 32 |
| **TS87C51RD2** | 0 | 64k | 768 | 1024 | 32 |

| PLCC68<br>VQFP64 1.4 | ROM (bytes) | EPROM (bytes) | XRAM (bytes) | TOTAL RAM (bytes) | I/O |
|---|---|---|---|---|---|
| **TS80C51RD2** | 0 | 0 | 768 | 1024 | 48 |
| **TS83C51RD2** | 64k | 0 | 768 | 1024 | 48 |
| **TS87C51RD2** | 0 | 64k | 768 | 1024 | 48 |

## 3. Block Diagram




(1): Alternate function of Port 1

(2): Only available on high pin count packages

(3): Alternate function of Port 3

# 4. SFR Mapping

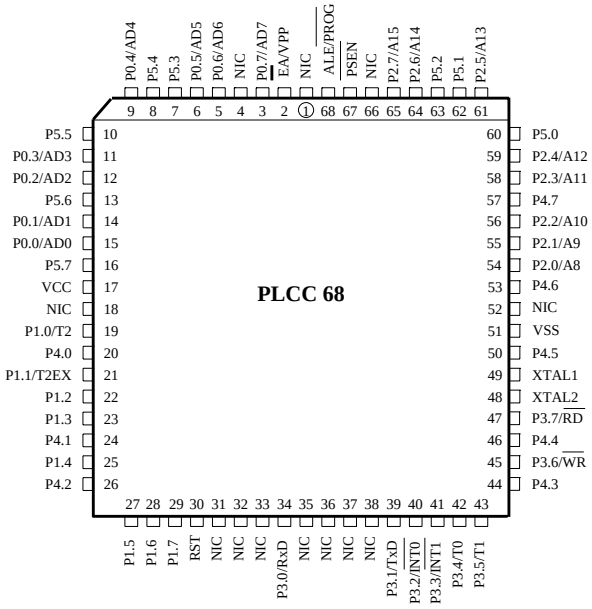
The Special Function Registers (SFRs) of the TS80C51Rx2 fall into the following categories:

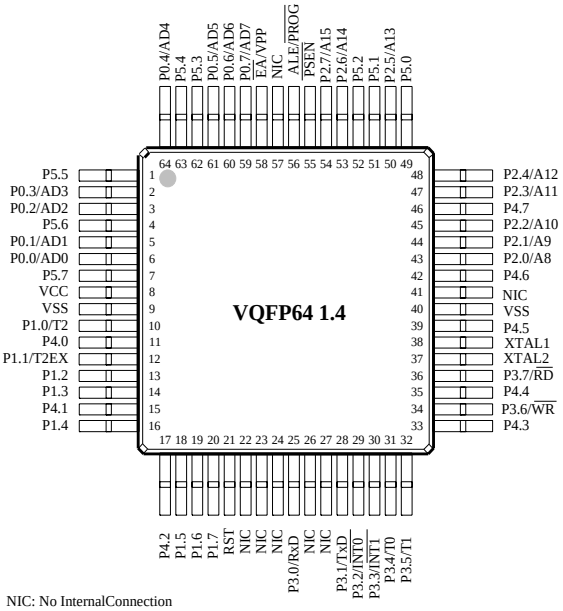
- C51 core registers: ACC, B, DPH, DPL, PSW, SP, AUXR1
- I/O port registers: P0, P1, P2, P3, P4, P5
- Timer registers: T2CON, T2MOD, TCON, TH0, TH1, TH2, TMOD, TL0, TL1, TL2, RCAP2L, RCAP2H
- Serial I/O port registers: SADDR, SADEN, SBUF, SCON
- Power and clock control registers: PCON
- HDW Watchdog Timer Reset: WDTRST, WDTPRG
- PCA registers: CL, CH, CCAPiL, CCAPiH, CCON, CMOD, CCAPMi
- Interrupt system registers: IE, IP, IPH
- Others: AUXR, CKCON

**Table 1. All SFRs with their address and their reset value**

| | **Bit addressable** | **Non Bit addressable** | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 0/8 | 1/9 | 2/A | 3/B | 4/C | 5/D | 6/E | 7/F | |
| F8h | | CH<br>0000 0000 | CCAP0H<br>XXXX XXXX | CCAP1H<br>XXXX XXXX | CCAPL2H<br>XXXX XXXX | CCAPL3H<br>XXXX XXXX | CCAPL4H<br>XXXX XXXX | | FFh |
| F0h | B<br>0000 0000 | | | | | | | | F7h |
| E8h | P5 bit addressable<br>1111 1111 | CL<br>0000 0000 | CCAP0L<br>XXXX XXXX | CCAP1L<br>XXXX XXXX | CCAPL2L<br>XXXX XXXX | CCAPL3L<br>XXXX XXXX | CCAPL4L<br>XXXX XXXX | | EFh |
| E0h | ACC<br>0000 0000 | | | | | | | | E7h |
| D8h | CCON<br>00X0 0000 | CMOD<br>00XX X000 | CCAPM0<br>X000 0000 | CCAPM1<br>X000 0000 | CCAPM2<br>X000 0000 | CCAPM3<br>X000 0000 | CCAPM4<br>X000 0000 | | DFh |
| D0h | PSW<br>0000 0000 | | | | | | | | D7h |
| C8h | T2CON<br>0000 0000 | T2MOD<br>XXXX XX00 | RCAP2L<br>0000 0000 | RCAP2H<br>0000 0000 | TL2<br>0000 0000 | TH2<br>0000 0000 | | | CFh |
| C0h | P4 bit addressable<br>1111 1111 | | | | | | | P5 byte addressable<br>1111 1111 | C7h |
| B8h | IP<br>X000 000 | SADEN<br>0000 0000 | | | | | | | BFh |
| B0h | P3<br>1111 1111 | | | | | | | IPH<br>X000 0000 | B7h |
| A8h | IE<br>0000 0000 | SADDR<br>0000 0000 | | | | | | | AFh |
| A0h | P2<br>1111 1111 | | AUXR1<br>XXXX0XX0 | | | | WDTRST<br>XXXX XXXX | WDTPRG<br>XXXX X000 | A7h |
| 98h | SCON<br>0000 0000 | SBUF<br>XXXX XXXX | | | | | | | 9Fh |
| 90h | P1<br>1111 1111 | | | | | | | | 97h |
| 88h | TCON<br>0000 0000 | TMOD<br>0000 0000 | TL0<br>0000 0000 | TL1<br>0000 0000 | TH0<br>0000 0000 | TH1<br>0000 0000 | AUXR<br>XXXXXX00 | CKCON<br>XXXX XXX0 | 8Fh |
| 80h | P0<br>1111 1111 | SP<br>0000 0111 | DPL<br>0000 0000 | DPH<br>0000 0000 | | | | PCON<br>00X1 0000 | 87h |
| | 0/8 | 1/9 | 2/A | 3/B | 4/C | 5/D | 6/E | 7/F | |

reserved




NIC: No InternalConnection

| | PLCC68 | SQUARE VQFP64 1.4 |
|---|---|---|
| P3.2 | 40 | 29 |
| P3.3 | 41 | 30 |
| P3.4 | 42 | 31 |
| P3.5 | 43 | 32 |
| P3.6 | 45 | 34 |
| P3.7 | 47 | 36 |
| RESET | 30 | 21 |
| ALE/$\overline{PROG}$ | 68 | 56 |
| $\overline{PSEN}$ | 67 | 55 |
| $\overline{EA}$/VPP | 2 | 58 |
| XTAL1 | 49 | 38 |
| XTAL2 | 48 | 37 |
| P4.0 | 20 | 11 |
| P4.1 | 24 | 15 |
| P4.2 | 26 | 17 |
| P4.3 | 44 | 33 |
| P4.4 | 46 | 35 |
| P4.5 | 50 | 39 |
| P4.6 | 53 | 42 |
| P4.7 | 57 | 46 |
| P5.0 | 60 | 49 |
| P5.1 | 62 | 51 |
| P5.2 | 63 | 52 |
| P5.3 | 7 | 62 |
| P5.4 | 8 | 63 |
| P5.5 | 10 | 1 |
| P5.6 | 13 | 4 |
| P5.7 | 16 | 7 |

**ASSEMBLY LANGUAGE**

```
; Block move using dual data pointers
; Destroys DPTR0, DPTR1, A and PSW
; note: DPS exits opposite of entry state
; unless an extra INC AUXR1 is added
;
00A2              AUXR1 EQU 0A2H
;
0000  909000      MOV  DPTR,#SOURCE     ; address of SOURCE
0003  05A2        INC   AUXR1           ; switch data pointers
0005  90A000      MOV  DPTR,#DEST       ; address of DEST
0008              LOOP:
0008  05A2        INC   AUXR1           ; switch data pointers
000A  E0          MOVX A,@DPTR          ; get a byte from SOURCE
000B  A3          INC   DPTR            ; increment SOURCE address
000C  05A2        INC   AUXR1           ; switch data pointers
000E  F0          MOVX @DPTR,A          ; write the byte to DEST
000F  A3          INC   DPTR            ; increment DEST address
0010  70F6        JNZ   LOOP            ; check for 0 terminator
0012  05A2        INC   AUXR1           ; (optional) restore DPS
```

INC is a short (2 bytes) and fast (12 clocks) way to manipulate the DPS bit in the AUXR1 SFR. However, note that the INC instruction does not directly force the DPS bit to a particular state, but simply toggles it. In simple routines, such as the block move example, only the fact that DPS is toggled in the proper sequence matters, not its actual value. In other words, the block move routine works the same whether DPS is '0' or '1' on entry. Observe that without the last instruction (INC AUXR1), the routine will exit with DPS in the opposite state.

**Table 12. CCAPnH: PCA Modules Capture/Compare Registers High**

| CCAPnH Address n = 0 - 4 | CCAP0H=0FAH<br>CCAP1H=0FBH<br>CCAP2H=0FCH<br>CCAP3H=0FDH<br>CCAP4H=0FEH | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reset value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 13. CCAPnL: PCA Modules Capture/Compare Registers Low**

| CCAPnL Address n = 0 - 4 | CCAP0L=0EAH<br>CCAP1L=0EBH<br>CCAP2L=0ECH<br>CCAP3L=0EDH<br>CCAP4L=0EEH | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reset value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 14. CH: PCA Counter High**

| CH Address 0F9H | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reset value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 15. CL: PCA Counter Low**

| CL Address 0E9H | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Reset value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### 6.5.4. Pulse Width Modulator Mode

All of the PCA modules can be used as PWM outputs. Figure 12 shows the PWM function. The frequency of the output depends on the source for the PCA timer. All of the modules will have the same frequency of output because they all share the PCA timer. The duty cycle of each module is independently variable using the module's capture register CCAPLn. When the value of the PCA CL SFR is less than the value in the module's CCAPLn 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, CCAPLn is reloaded with the value in CCAPHn. 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.

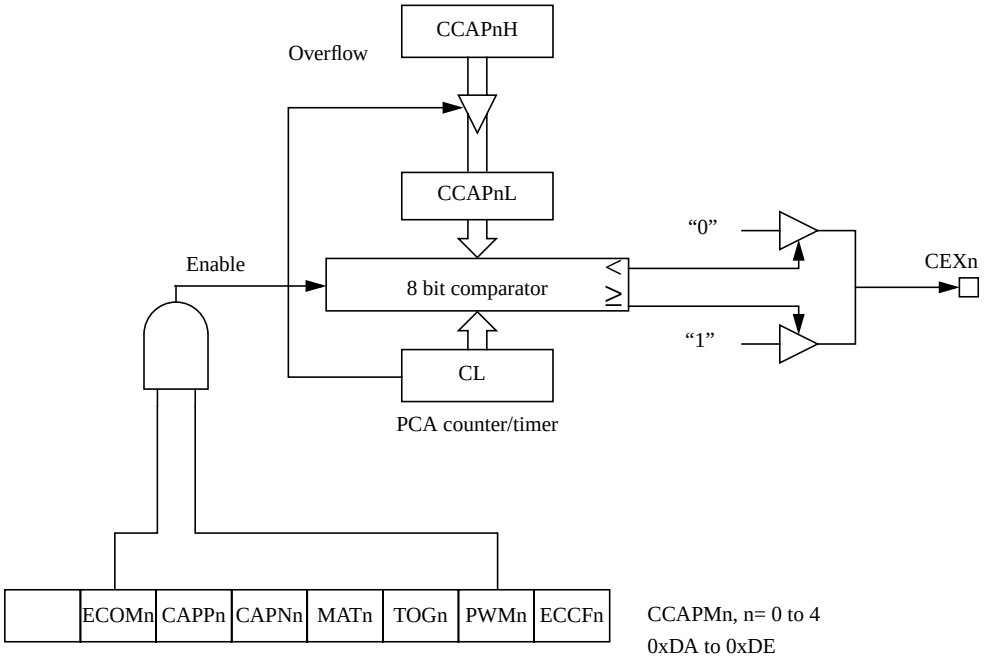



**Figure 12. PCA PWM Mode**

### 6.5.5. 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 10 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.

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, or
- 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.

This watchdog timer won't generate a reset out on the reset pin.

**Table 16. SCON Register**

**SCON - Serial Control Register (98h)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| **FE/SM0** | **SM1** | **SM2** | **REN** | **TB8** | **RB8** | **TI** | **RI** |

| Bit Number | Bit Mnemonic | Description |
|---|---|---|
| 7 | FE | **Framing Error bit (SMOD0=1)**<br>Clear to reset the error state, not cleared by a valid stop bit.<br>Set by hardware when an invalid stop bit is detected.<br>SMOD0 must be set to enable access to the FE bit |
| | SM0 | **Serial port Mode bit 0**<br>Refer to SM1 for serial port mode selection.<br>SMOD0 must be cleared to enable access to the SM0 bit |
| 6 | SM1 | **Serial port Mode bit 1**<br>SM0 SM1 Mode Description Baud Rate<br>0 0 0 Shift Register $F_{XTAL}$/12 (/6 in X2 mode)<br>0 1 1 8-bit UART Variable<br>1 0 2 9-bit UART $F_{XTAL}$/64 or $F_{XTAL}$/32 (/32, /16 in X2 mode)<br>1 1 3 9-bit UART Variable |
| 5 | SM2 | **Serial port Mode 2 bit / Multiprocessor Communication Enable bit**<br>Clear to disable multiprocessor communication feature.<br>Set to enable multiprocessor communication feature in mode 2 and 3, and eventually mode 1. This bit should be cleared in mode 0. |
| 4 | REN | **Reception Enable bit**<br>Clear to disable serial reception.<br>Set to enable serial reception. |
| 3 | TB8 | **Transmitter Bit 8 / Ninth bit to transmit in modes 2 and 3.**<br>Clear to transmit a logic 0 in the 9th bit.<br>Set to transmit a logic 1 in the 9th bit. |
| 2 | RB8 | **Receiver Bit 8 / Ninth bit received in modes 2 and 3**<br>Cleared by hardware if 9th bit received is a logic 0.<br>Set by hardware if 9th bit received is a logic 1.<br>In mode 1, if SM2 = 0, RB8 is the received stop bit. In mode 0 RB8 is not used. |
| 1 | TI | **Transmit Interrupt flag**<br>Clear to acknowledge interrupt.<br>Set by hardware at the end of the 8th bit time in mode 0 or at the beginning of the stop bit in the other modes. |
| 0 | RI | **Receive Interrupt flag**<br>Clear to acknowledge interrupt.<br>Set by hardware at the end of the 8th bit time in mode 0, see Figure 14. and Figure 15. in the other modes. |

Reset Value = 0000 0000b
Bit addressable

**Table 17. PCON Register**

**PCON - Power Control Register (87h)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| **SMOD1** | **SMOD0** | **-** | **POF** | **GF1** | **GF0** | **PD** | **IDL** |

| Bit Number | Bit Mnemonic | Description |
|---|---|---|
| 7 | SMOD1 | **Serial port Mode bit 1**<br>Set to select double baud rate in mode 1, 2 or 3. |
| 6 | SMOD0 | **Serial port Mode bit 0**<br>Clear to select SM0 bit in SCON register.<br>Set to to select FE bit in SCON register. |
| 5 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 4 | POF | **Power-Off Flag**<br>Clear to recognize next reset type.<br>Set by hardware when VCC rises from 0 to its nominal voltage. Can also be set by software. |
| 3 | GF1 | **General purpose Flag**<br>Cleared by user for general purpose usage.<br>Set by user for general purpose usage. |
| 2 | GF0 | **General purpose Flag**<br>Cleared by user for general purpose usage.<br>Set by user for general purpose usage. |
| 1 | PD | **Power-Down mode bit**<br>Cleared by hardware when reset occurs.<br>Set to enter power-down mode. |
| 0 | IDL | **Idle mode bit**<br>Clear by hardware when interrupt or reset occurs.<br>Set to enter idle mode. |

Reset Value = 00X1 0000b
Not bit addressable

Power-off flag reset value will be 1 only after a power on (cold reset). A warm reset doesn't affect the value of this bit.

**Table 21. IPH Register**

**IPH - Interrupt Priority High Register (B7h)**

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

| Bit Number | Bit Mnemonic | Description |
|---|---|---|
| 7 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 6 | PPCH | **PCA interrupt priority bit high.**<br>PPCH PPC Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |
| 5 | PT2H | **Timer 2 overflow interrupt Priority High bit**<br>PT2H PT2 Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |
| 4 | PSH | **Serial port Priority High bit**<br>PSH PS Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |
| 3 | PT1H | **Timer 1 overflow interrupt Priority High bit**<br>PT1H PT1 Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |
| 2 | PX1H | **External interrupt 1 Priority High bit**<br>PX1H PX1 Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |
| 1 | PT0H | **Timer 0 overflow interrupt Priority High bit**<br>PT0H PT0 Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |
| 0 | PX0H | **External interrupt 0 Priority High bit**<br>PX0H PX0 Priority Level<br>0 0 Lowest<br>0 1<br>1 0<br>1 1 Highest |

Reset Value = X000 0000b
Not bit addressable

## 6.12. Power-Off Flag

The power-off flag allows the user to distinguish between a “cold start” reset and a “warm start” reset.

A cold start reset is the one induced by $V_{CC}$ switch-on. A warm start reset occurs while $V_{CC}$ is still applied to the device and could be generated for example by an exit from power-down.

The power-off flag (POF) is located in PCON register (See Table 26.). POF is set by hardware when $V_{CC}$ rises from 0 to its nominal voltage. The POF can be set or cleared by software allowing the user to determine the type of reset.

The POF value is only relevant with a Vcc range from 4.5V to 5.5V. For lower Vcc value, reading POF bit will return indeterminate value.

**Table 26. PCON Register**

**PCON - Power Control Register (87h)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| **SMOD1** | **SMOD0** | **-** | **POF** | **GF1** | **GF0** | **PD** | **IDL** |

| Bit Number | Bit Mnemonic | Description |
|---|---|---|
| 7 | SMOD1 | **Serial port Mode bit 1**<br>Set to select double baud rate in mode 1, 2 or 3. |
| 6 | SMOD0 | **Serial port Mode bit 0**<br>Clear to select SM0 bit in SCON register.<br>Set to to select FE bit in SCON register. |
| 5 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 4 | POF | **Power-Off Flag**<br>Clear to recognize next reset type.<br>Set by hardware when $V_{CC}$ rises from 0 to its nominal voltage. Can also be set by software. |
| 3 | GF1 | **General purpose Flag**<br>Cleared by user for general purpose usage.<br>Set by user for general purpose usage. |
| 2 | GF0 | **General purpose Flag**<br>Cleared by user for general purpose usage.<br>Set by user for general purpose usage. |
| 1 | PD | **Power-Down mode bit**<br>Cleared by hardware when reset occurs.<br>Set to enter power-down mode. |
| 0 | IDL | **Idle mode bit**<br>Clear by hardware when interrupt or reset occurs.<br>Set to enter idle mode. |

Reset Value = 00X1 0000b
Not bit addressable

## 6.13. Reduced EMI Mode

The ALE signal is used to demultiplex address and data buses on port 0 when used with external program or data memory. Nevertheless, during internal code execution, ALE signal is still generated. In order to reduce EMI, ALE signal can be disabled by setting AO bit.

The AO bit is located in AUXR register at bit location 0. As soon as AO is set, ALE is no longer output but remains active during MOVX and MOVC instructions and external fetches. During ALE disabling, ALE pin is weakly pulled high.

**Table 27. AUXR Register**

**AUXR - Auxiliary Register (8Eh)**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| - | - | - | - | - | - | EXTRAM | AO |

| Bit Number | Bit Mnemonic | Description |
|---|---|---|
| 7 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 6 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 5 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 4 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 3 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 2 | - | **Reserved**<br>The value read from this bit is indeterminate. Do not set this bit. |
| 1 | EXTRAM | **EXTRAM bit**<br>See Table 5. |
| 0 | AO | **ALE Output bit**<br>Clear to restore ALE operation during internal fetches.<br>Set to disable ALE operation during internal fetches. |

Reset Value = XXXX XX00b
Not bit addressable

# 7. TS83C51RB2/RC2/RD2 ROM

## 7.1. ROM Structure

The TS83C51RB2/RC2/RD2 ROM memory is divided in three different arrays:

- the code array: . . . . . 16/32/64 Kbytes.
- the encryption array: . . . . . 64 bytes.
- the signature array:. . . . . 4 bytes.

## 7.2. ROM Lock System

The program Lock system, when programmed, protects the on-chip program against software piracy.

### 7.2.1. 7.2.1. Encryption Array

Within the ROM array are 64 bytes of encryption array that are initially unprogrammed (all FF's). Every time a byte is addressed during program verify, 6 address lines are used to select a byte of the encryption array. This byte is then exclusive-NOR'ed (XNOR) with the code byte, creating an encrypted verify byte. The algorithm, with the encryption array in the unprogrammed state, will return the code in its original, unmodified form.

When using the encryption array, one important factor needs to be considered. If a byte has the value FFh, verifying the byte will produce the encryption byte value. If a large block (>64 bytes) of code is left unprogrammed, a verification routine will display the content of the encryption array. For this reason all the unused code bytes should be programmed with random values. This will ensure program protection.

### 7.2.2. Program Lock Bits

The lock bits when programmed according to Table 28. will provide different level of protection for the on-chip code and data.

**Table 28. Program Lock bits**

| Program Lock Bits | | | | Protection description |
|---|---|---|---|---|
| **Security level** | **LB1** | **LB2** | **LB3** | |
| 1 | U | U | U | No program lock features enabled. Code verify will still be encrypted by the encryption array if programmed. MOVC instruction executed from external program memory returns non encrypted data. |
| 2 | P | U | U | MOVC instruction executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset. |
| 3 | U | P | U | Same as level 1+ Verify disable.<br>This security level is only available for 51RDX2 devices. |

U: unprogrammed
P: programmed

### 7.2.3. Signature bytes

The TS83C51RB2/RC2/RD2 contains 4 factory programmed signatures bytes. To read these bytes, perform the process described in section 8.3.

### 7.2.4. Verify Algorithm

Refer to 8.3.4.

# 8. TS87C51RB2/RC2/RD2 EPROM

## 8.1. EPROM Structure

The TS87C51RB2/RC2/RD2 EPROM is divided in two different arrays:

- the code array: . . . . . 16/32/64 Kbytes.
- the encryption array: . . . . . 64 bytes.

In addition a third non programmable array is implemented:

- the signature array: . . . . . 4 bytes.

## 8.2. EPROM Lock System

The program Lock system, when programmed, protects the on-chip program against software piracy.

### 8.2.1. Encryption Array

Within the EPROM array are 64 bytes of encryption array that are initially unprogrammed (all FF's). Every time a byte is addressed during program verify, 6 address lines are used to select a byte of the encryption array. This byte is then exclusive-NOR'ed (XNOR) with the code byte, creating an encrypted verify byte. The algorithm, with the encryption array in the unprogrammed state, will return the code in its original, unmodified form.

When using the encryption array, one important factor needs to be considered. If a byte has the value FFh, verifying the byte will produce the encryption byte value. If a large block (>64 bytes) of code is left unprogrammed, a verification routine will display the content of the encryption array. For this reason all the unused code bytes should be programmed with random values. This will ensure program protection.

### 8.2.2. Program Lock Bits

The three lock bits, when programmed according to Table 29.8.2.3. , will provide different level of protection for the on-chip code and data.

**Table 29. Program Lock bits**

| Program Lock Bits | | | | Protection description |
|---|---|---|---|---|
| Security level | LB1 | LB2 | LB3 | |
| 1 | U | U | U | No program lock features enabled. Code verify will still be encrypted by the encryption array if programmed. MOVC instruction executed from external program memory returns non encrypted data. |
| 2 | P | U | U | MOVC instruction executed from external program memory are disabled from fetching code bytes from internal memory, $\overline{EA}$ is sampled and latched on reset, and further programming of the EPROM is disabled. |
| 3 | U | P | U | Same as 2, also verify is disabled. |
| 4 | U | U | P | Same as 3, also external execution is disabled. |

U: unprogrammed,
P: programmed

WARNING: Security level 2 and 3 should only be programmed after EPROM and Core verification.

### 8.2.3. Signature bytes

The TS87C51RB2/RC2/RD2 contains 4 factory programmed signatures bytes. To read these bytes, perform the process described in section 8.3.

## 8.3. EPROM Programming

### 8.3.1. Set-up modes

In order to program and verify the EPROM or to read the signature bytes, the TS87C51RB2/RC2/RD2 is placed in specific set-up modes (See Figure 18.).

Control and program signals must be held at the levels indicated in Table 30.

### 8.3.2. Definition of terms

**Address Lines:** P1.0-P1.7, P2.0-P2.5, P3.4, P3.5 respectively for A0-A15 (P2.5 (A13) for RB, P3.4 (A14) for RC, P3.5 (A15) for RD)

**Data Lines:** P0.0-P0.7 for D0-D7

**Control Signals:** RST, $\overline{PSEN}$, P2.6, P2.7, P3.3, P3.6, P3.7.

**Program Signals:** ALE/$\overline{PROG}$, $\overline{EA}$/VPP.

**Table 30. EPROM Set-Up Modes**

| Mode | RST | PSEN | ALE/$\overline{PROG}$ | $\overline{EA}$/VPP | P2.6 | P2.7 | P3.3 | P3.6 | P3.7 |
|---|---|---|---|---|---|---|---|---|---|
| Program Code data | 1 | 0 | ⊓ | 12.75V | 0 | 1 | 1 | 1 | 1 |
| Verify Code data | 1 | 0 | 1 | 1 | 0 | ⊓ | 0 | 1 | 1 |
| Program Encryption Array Address 0-3Fh | 1 | 0 | ⊓ | 12.75V | 0 | 1 | 1 | 0 | 1 |
| Read Signature Bytes | 1 | 0 | 1 | 1 | 0 | ⊓ | 0 | 0 | 0 |
| Program Lock bit 1 | 1 | 0 | ⊓ | 12.75V | 1 | 1 | 1 | 1 | 1 |
| Program Lock bit 2 | 1 | 0 | ⊓ | 12.75V | 1 | 1 | 1 | 0 | 0 |
| Program Lock bit 3 | 1 | 0 | ⊓ | 12.75V | 1 | 0 | 1 | 1 | 0 |

# 9. Signature Bytes

The TS83/87C51RB2/RC2/RD2 has four signature bytes in location 30h, 31h, 60h and 61h. To read these bytes follow the procedure for EPROM verify but activate the control lines provided in Table 31. for Read Signature Bytes. Table 31. shows the content of the signature byte for the TS87C51RB2/RC2/RD2.

**Table 31. Signature Bytes Content**

| Location | Contents | Comment |
|---|---|---|
| 30h | 58h | Manufacturer Code: Atmel Wireless & Microcontrollers |
| 31h | 57h | Family Code: C51 X2 |
| 60h | 7Ch | Product name: TS83C51RD2 |
| 60h | FCh | Product name: TS87C51RD2 |
| 60h | 37h | Product name: TS83C51RC2 |
| 60h | B7h | Product name: TS87C51RC2 |
| 60h | 3Bh | Product name: TS83C51RB2 |
| 60h | BBh | Product name: TS87C51RB2 |
| 61h | FFh | Product revision number |

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| $I_{CC}$ operating | Power Supply Current Maximum values, X1 mode: (7) | | | 3 + 0.6 Freq (MHz) @12MHz 10.2 @16MHz 12.6 | mA | $V_{CC}$ = 5.5 V(8) |
| $I_{CC}$ idle | Power Supply Current Maximum values, X1 mode: (7) | | | 0.25+0.3 Freq (MHz) @12MHz 3.9 @16MHz 5.1 | mA | $V_{CC}$ = 5.5 V(2) |

## 10.4. DC Parameters for Low Voltage

$T_A$ = 0°C to +70°C; $V_{SS}$ = 0 V; $V_{CC}$ = 2.7 V to 5.5 V ± 10%; F = 0 to 30 MHz.
$T_A$ = -40°C to +85°C; $V_{SS}$ = 0 V; $V_{CC}$ = 2.7 V to 5.5 V ± 10%; F = 0 to 30 MHz.

**Table 33. DC Parameters for Low Voltage**

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| $V_{IL}$ | Input Low Voltage | -0.5 | | 0.2 $V_{CC}$ - 0.1 | V | |
| $V_{IH}$ | Input High Voltage except XTAL1, RST | 0.2 $V_{CC}$ + 0.9 | | $V_{CC}$ + 0.5 | V | |
| $V_{IH1}$ | Input High Voltage, XTAL1, RST | 0.7 $V_{CC}$ | | $V_{CC}$ + 0.5 | V | |
| $V_{OL}$ | Output Low Voltage, ports 1, 2, 3, 4, 5 (6) | | | 0.45 | V | $I_{OL}$ = 0.8 mA(4) |
| $V_{OL1}$ | Output Low Voltage, port 0, ALE, $\overline{PSEN}$ (6) | | | 0.45 | V | $I_{OL}$ = 1.6 mA(4) |
| $V_{OH}$ | Output High Voltage, ports 1, 2, 3, 4, 5 | 0.9 $V_{CC}$ | | | V | $I_{OH}$ = -10 µA |
| $V_{OH1}$ | Output High Voltage, port 0, ALE, $\overline{PSEN}$ | 0.9 $V_{CC}$ | | | V | $I_{OH}$ = -40 µA |
| $I_{IL}$ | Logical 0 Input Current ports 1, 2, 3, 4, 5 | | | -50 | µA | Vin = 0.45 V |
| $I_{LI}$ | Input Leakage Current | | | ±10 | µA | 0.45 V < Vin < $V_{CC}$ |
| $I_{TL}$ | Logical 1 to 0 Transition Current, ports 1, 2, 3, 4, 5 | | | -650 | µA | Vin = 2.0 V |
| $R_{RST}$ | RST Pulldown Resistor | 50 | 90 (5) | 200 | kΩ | |
| CIO | Capacitance of I/O Buffer | | | 10 | pF | Fc = 1 MHz $T_A$ = 25°C |
| $I_{PD}$ | Power Down Current | | 20 (5)<br>10 (5) | 50<br>30 | µA | $V_{CC}$ = 2.0 V to 5.5 V(3)<br>$V_{CC}$ = 2.0 V to 3.3 V(3) |
| $I_{CC}$ under RESET | Power Supply Current Maximum values, X1 mode: (7) | | | 1 + 0.2 Freq (MHz) @12MHz 3.4 @16MHz 4.2 | mA | $V_{CC}$ = 3.3 V(1) |
| $I_{CC}$ operating | Power Supply Current Maximum values, X1 mode: (7) | | | 1 + 0.3 Freq (MHz) @12MHz 4.6 @16MHz 5.8 | mA | $V_{CC}$ = 3.3 V(8) |

| Symbol | Parameter | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| $I_{CC}$ idle | Power Supply Current Maximum values, X1 mode: (7) | | | 0.15 Freq (MHz) + 0.2 @12MHz 2 @16MHz 2.6 | mA | $V_{CC}$ = 3.3 V(2) |

*NOTES*

*1. $I_{CC}$ under reset is measured with all output pins disconnected; XTAL1 driven with $T_{CLCH}$, $T_{CHCL}$ = 5 ns (see Figure 24.), $V_{IL}$ = $V_{SS}$ + 0.5 V, $V_{IH}$ = $V_{CC}$ - 0.5V; XTAL2 N.C.; $\overline{EA}$ = RST = Port 0 = $V_{CC}$. $I_{CC}$ would be slightly higher if a crystal oscillator used..*

*2. Idle $I_{CC}$ is measured with all output pins disconnected; XTAL1 driven with $T_{CLCH}$, $T_{CHCL}$ = 5 ns, $V_{IL}$ = $V_{SS}$ + 0.5 V, $V_{IH}$ = $V_{CC}$ - 0.5 V; XTAL2 N.C; Port 0 = $V_{CC}$; $\overline{EA}$ = RST = $V_{SS}$ (see Figure 22.).*

*3. Power Down $I_{CC}$ is measured with all output pins disconnected; $\overline{EA}$ = $V_{SS}$, PORT 0 = $V_{CC}$; XTAL2 NC.; RST = $V_{SS}$ (see Figure 23.).*

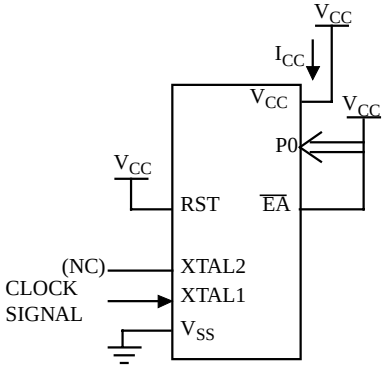
*4. Capacitance loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the $V_{OL}$s of ALE and Ports 1 and 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make 1 to 0 transitions during bus operation. In the worst cases (capacitive loading 100pF), the noise pulse on the ALE line may exceed 0.45V with maxi $V_{OL}$ peak 0.6V. A Schmitt Trigger use is not necessary.*

*5. Typicals are based on a limited number of samples and are not guaranteed. The values listed are at room temperature and 5V.*

*6. Under steady state (non-transient) conditions, $I_{OL}$ must be externally limited as follows:*
*Maximum $I_{OL}$ per port pin: 10 mA*
*Maximum $I_{OL}$ per 8-bit port:*
*Port 0: 26 mA*
*Ports 1, 2, 3 and 4 and 5 when available: 15 mA*
*Maximum total $I_{OL}$ for all output pins: 71 mA*
*If $I_{OL}$ exceeds the test condition, $V_{OL}$ may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.*

*7. For other values, please contact your sales office.*

*8. Operating $I_{CC}$ is measured with all output pins disconnected; XTAL1 driven with $T_{CLCH}$, $T_{CHCL}$ = 5 ns (see Figure 24.), $V_{IL}$ = $V_{SS}$ + 0.5 V, $V_{IH}$ = $V_{CC}$ - 0.5V; XTAL2 N.C.; $\overline{EA}$ = Port 0 = $V_{CC}$; RST = $V_{SS}$. The internal ROM runs the code 80 FE (label: SJMP label). $I_{CC}$ would be slightly higher if a crystal oscillator is used. Measurements are made with OTP products when possible, which is the worst case.*




All other pins are disconnected.

**Figure 20. $I_{CC}$ Test Condition, under reset**

### 10.5.6. External Data Memory Read Cycle

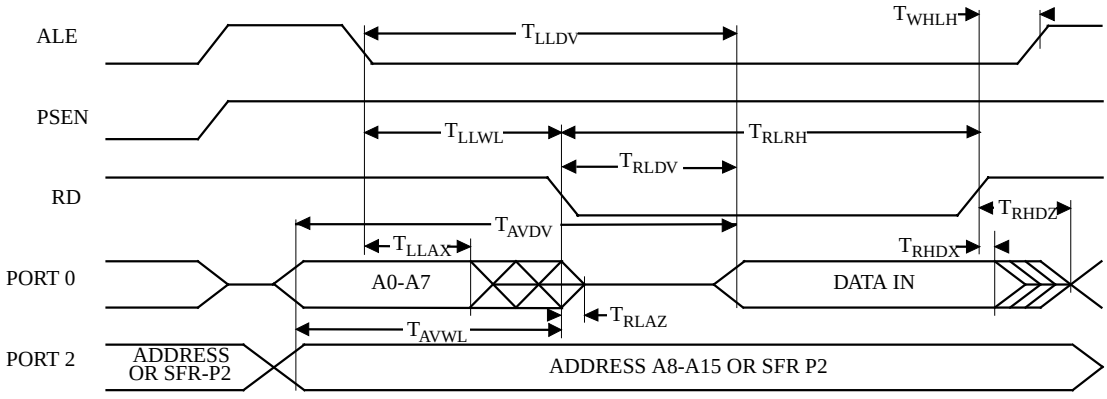



**Figure 27. External Data Memory Read Cycle**

### 10.5.7. Serial Port Timing - Shift Register Mode

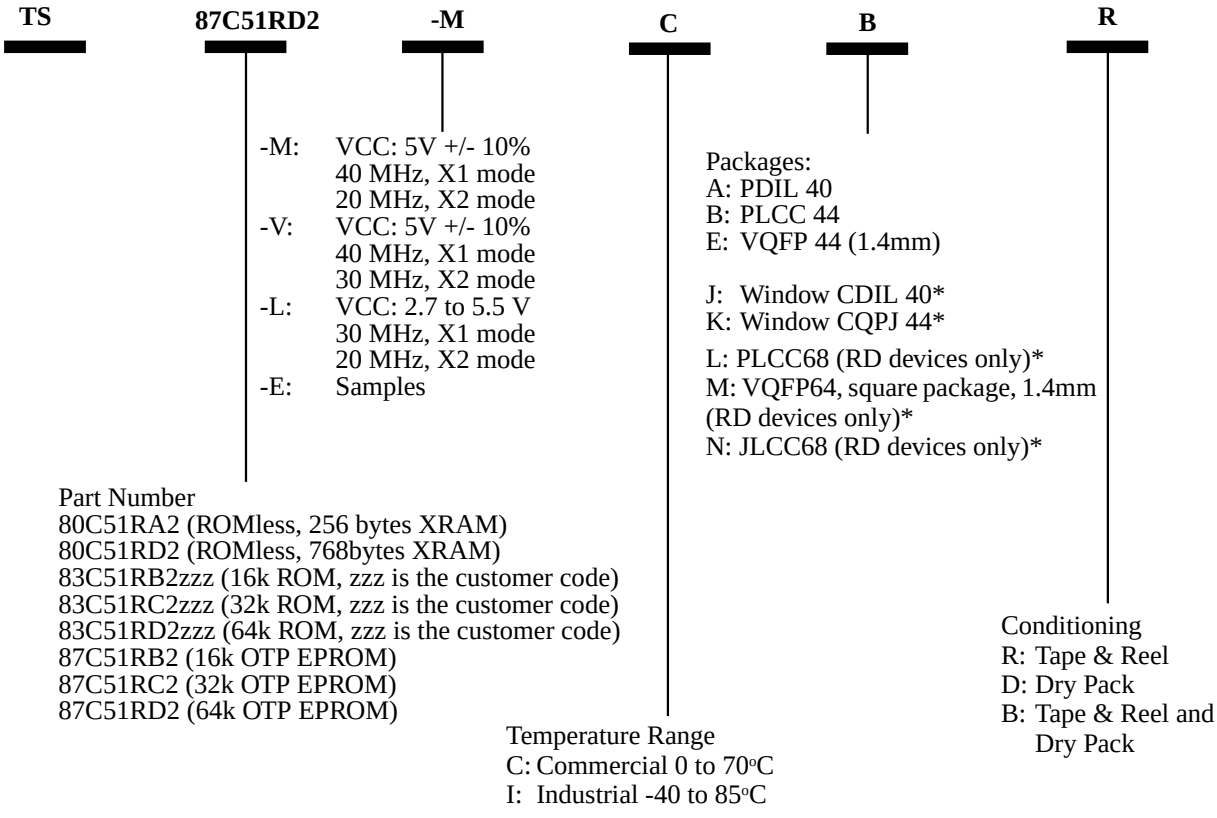
**Table 42. Symbol Description**

| Symbol | Parameter |
|---|---|
| $T_{XLXL}$ | Serial port clock cycle time |
| $T_{QVHX}$ | Output data set-up to clock rising edge |
| $T_{XHQX}$ | Output data hold after clock rising edge |
| $T_{XHDX}$ | Input data hold after clock rising edge |
| $T_{XHDV}$ | Clock rising edge to input data valid |

**Table 43. AC Parameters for a Fix Clock**

| Speed | -M 40 MHz | | -V X2 mode 30 MHz 60 MHz equiv. | | -V standard mode 40 MHz | | -L X2 mode 20 MHz 40 MHz equiv. | | -L standard mode 30 MHz | | Units |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **Symbol** | **Min** | **Max** | **Min** | **Max** | **Min** | **Max** | **Min** | **Max** | **Min** | **Max** | |
| $T_{XLXL}$ | 300 | | 200 | | 300 | | 300 | | 400 | | ns |
| $T_{QVHX}$ | 200 | | 117 | | 200 | | 200 | | 283 | | ns |
| $T_{XHQX}$ | 30 | | 13 | | 30 | | 30 | | 47 | | ns |
| $T_{XHDX}$ | 0 | | 0 | | 0 | | 0 | | 0 | | ns |
| $T_{XHDV}$ | | 117 | | 34 | | 117 | | 117 | | 200 | ns |

# 11. Ordering Information




(*) Check with Atmel Wireless & Microcontrollers Sales Office for availability. Ceramic packages (J, K, N) are available for proto typing, not for volume production. Ceramic packages are available for OTP only.

**Table 47. Maximum Clock Frequency**

| Code | -M | -V | -L | Unit |
|---|---|---|---|---|
| Standard Mode, oscillator frequency<br>Standard Mode, internal frequency | 40<br>40 | 40<br>40 | 30<br>30 | MHz |
| X2 Mode, oscillator frequency<br>X2 Mode, internal equivalent frequency | 20<br>40 | 30<br>**60** | 20<br>**40** | MHz |