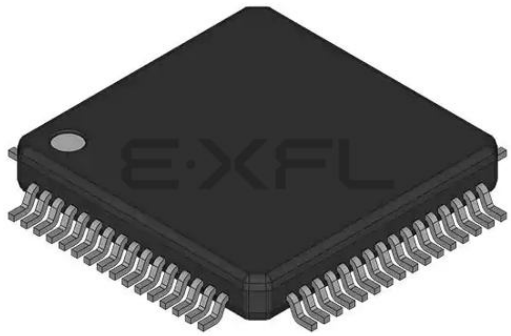


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

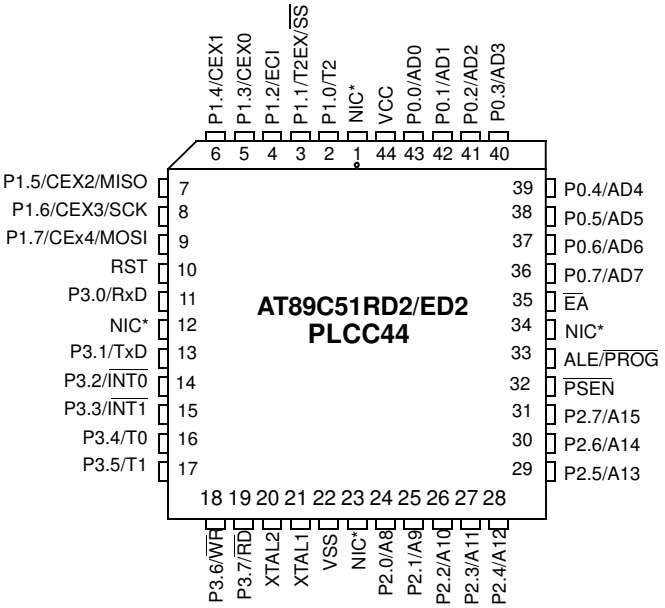
"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" 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.

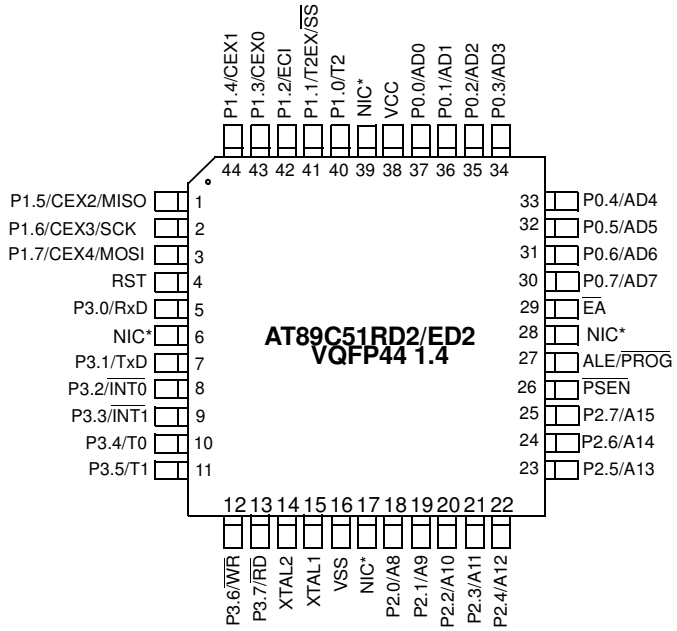
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	80C51
Core Size	8-Bit
Speed	60MHz
Connectivity	SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	50
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	2K x 8
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	-
Oscillator Type	External
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	64-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/atmel/at89c51ed2-rdrum

4. Pin Configurations

Figure 4-1. Pin Configurations





6. Oscillator

To optimize the power consumption and execution time needed for a specific task, an internal prescaler feature has been implemented between the oscillator and the CPU and peripherals.

6.1 Registers

Table 6-1. CKRL Register

CKRL – Clock Reload Register (97h)

7	6	5	4	3	2	1	0
CKRL7	CKRL6	CKRL5	CKRL4	CKRL3	CKRL2	CKRL1	CKRL0

Bit Number	Mnemonic	Description
7:0	CKRL	Clock Reload Register Prescaler value

Reset Value = 1111 1111b
Not bit addressable

Table 6-2. PCON Register

PCON – Power Control Register (87h)

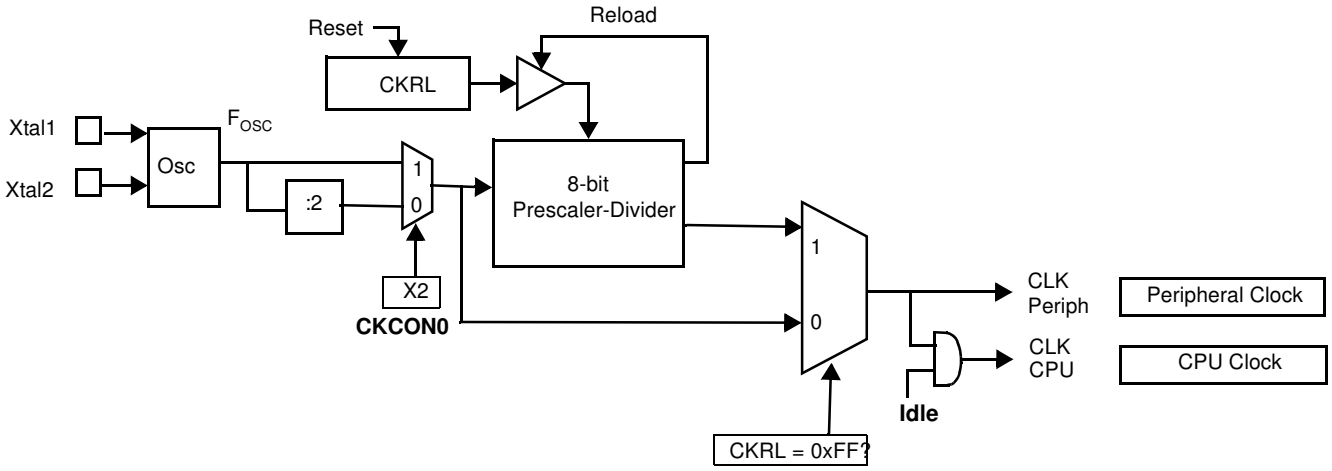
7	6	5	4	3	2	1	0
SMOD1	SMOD0	-	POF	GF1	GF0	PD	IDL

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

Reset Value = 00X1 0000b Not bit addressable

6.2 Functional Block Diagram

Figure 6-1. Functional Oscillator Block Diagram



6.2.1 Prescaler Divider

- A hardware RESET puts the prescaler divider in the following state:
  - CKRL = FFh: $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/2$ (Standard C51 feature)
- Any value between FFh down to 00h can be written by software into CKRL register in order to divide frequency of the selected oscillator:
  - CKRL = 00h: minimum frequency
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/1020$ (Standard Mode)
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/510$ (X2 Mode)
  - CKRL = FFh: maximum frequency
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}/2$ (Standard Mode)
    $F_{CLK\ CPU} = F_{CLK\ PERIPH} = F_{OSC}$ (X2 Mode)

$F_{CLK\ CPU}$ and $F_{CLK\ PERIPH}$

In X2 Mode, for CKRL<>0xFF:

$$F_{CPU} = F_{CLKPERIPH} = \frac{F_{OSC}}{2 \times (255 - CKRL)}$$

In X1 Mode, for CKRL<>0xFF then:

$$F_{CPU} = F_{CLKPERIPH} = \frac{F_{OSC}}{4 \times (255 - CKRL)}$$

part of the available XRAM as explained in Table 9-1. This can be useful if external peripherals are mapped at addresses already used by the internal XRAM.

- With EXTRAM = 0, the XRAM is indirectly addressed, using the MOVX instruction in combination with any of the registers R0, R1 of the selected bank or DPTR. An access to XRAM will not affect ports P0, P2, P3.6 (WR) and P3.7 (RD). For example, with EXTRAM = 0, MOVX @R0, # data where R0 contains 0A0H, accesses the XRAM at address 0A0H rather than external memory. An access to external data memory locations higher than the accessible size of the XRAM will be performed with the MOVX DPTR instructions in the same way as in the standard 80C51, with P0 and P2 as data/address busses, and P3.6 and P3.7 as write and read timing signals. Accesses to XRAM above 0FFH can only be done by the use of DPTR.
- With EXTRAM = 1, MOVX @Ri and MOVX @DPTR will be similar to the standard 80C51.MOVX @ Ri will provide an eight-bit address multiplexed with data on Port0 and any output port pins can be used to output higher order address bits. This is to provide the external paging capability. MOVX @DPTR will generate a sixteen-bit address. Port2 outputs the high-order eight address bits (the contents of DPH) while Port0 multiplexes the low-order eight address bits (DPL) with data. MOVX @ Ri and MOVX @DPTR will generate either read or write signals on P3.6 ($\overline{WR}$) and P3.7 ($\overline{RD}$).

The stack pointer (SP) may be located anywhere in the 256 bytes RAM (lower and upper RAM) internal data memory. The stack may not be located in the XRAM.

The M0 bit allows to stretch the XRAM timings; if M0 is set, the read and write pulses are extended from 6 to 30 clock periods. This is useful to access external slow peripherals.

9.1 Registers

Table 9-1. AUXR Register

AUXR - Auxiliary Register (8Eh)

7	6	5	4	3	2	1	0
DPU	**-**	**M0**	**XRS2**	**XRS1**	**XRS0**	**EXTRAM**	**AO**

Bit Number	Bit Mnemonic	Description
7	DPU	**Disable Weak Pull-up** Cleared by software to activate the permanent weak pull-up (default) Set by software to disable the weak pull-up (reduce power consumption)
6	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
5	M0	**Pulse length** Cleared to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 6 clock periods (default). Set to stretch MOVX control: the $\overline{RD}$ and the $\overline{WR}$ pulse length is 30 clock periods.
4	XRS2	**XRAM Size** XRS2 XRS1 XRS0 XRAM size 0 0 0 256 bytes 0 0 1 512 bytes 0 1 0 768 bytes (default) 0 1 1 1024 bytes 1 0 0 1792 bytes
3	XRS1	
2	XRS0	

Figure 10-3. Recommended Reset Output Schematic

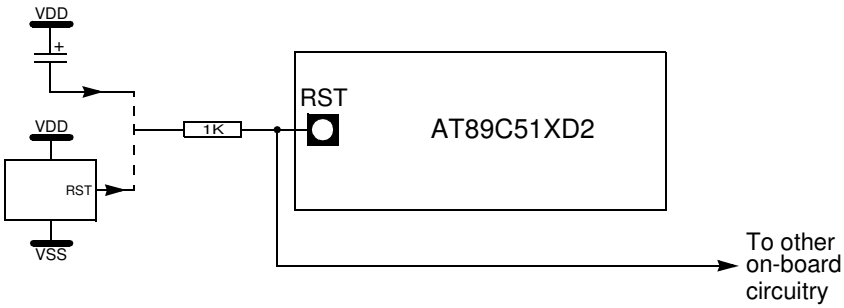
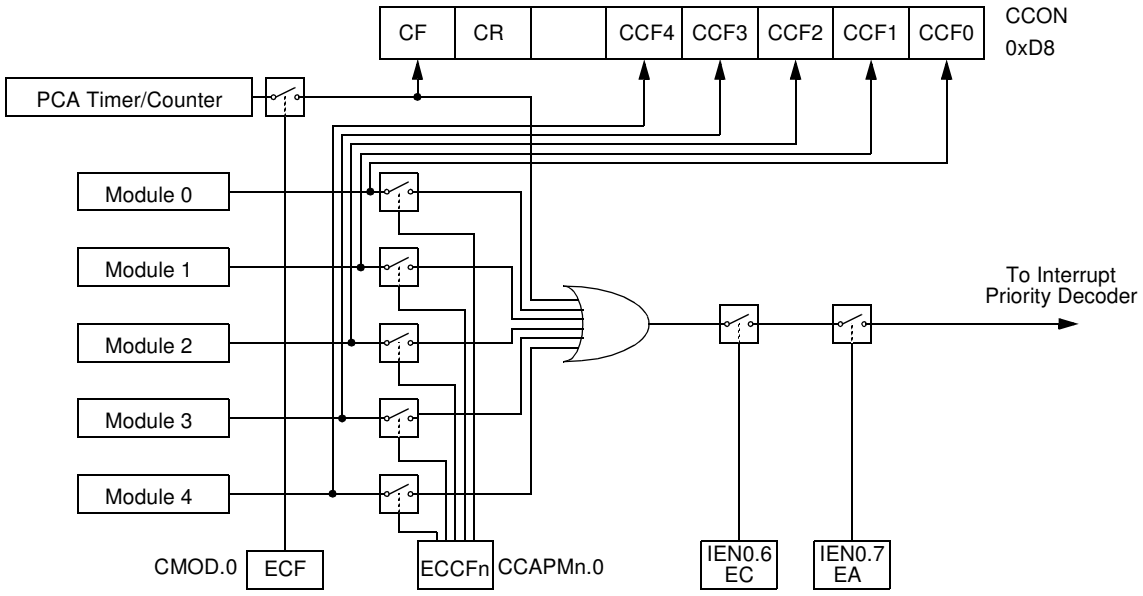


Figure 13-2. PCA Interrupt System



PCA Modules: each one of the five compare/capture modules has six possible functions. It can perform:

- 16-bit Capture, positive-edge triggered
- 16-bit Capture, negative-edge triggered
- 16-bit Capture, both positive and negative-edge triggered
- 16-bit Software Timer
- 16-bit High Speed Output
- 8-bit Pulse Width Modulator

In addition, Module 4 can be used as a Watchdog Timer.

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. (See Table 13-3). The registers contain the bits that control the mode that each module will operate in.

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

Table 13-6. CCAPnL Registers (n = 0 - 4)

CCAP0L - PCA Module 0 Compare/Capture Control Register Low (0EAh)

CCAP1L - PCA Module 1 Compare/Capture Control Register Low (0EBh)

CCAP2L - PCA Module 2 Compare/Capture Control Register Low (0ECh)

CCAP3L - PCA Module 3 Compare/Capture Control Register Low (0EDh)

CCAP4L - PCA Module 4 Compare/Capture Control Register Low (0EEh)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 0	-	**PCA Module n Compare/Capture Control** CCAPnL Value

Reset Value = 0000 0000b

Not bit addressable

Table 13-7. CH Register

CH - PCA Counter Register High (0F9h)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 0	-	**PCA counter** CH Value

Reset Value = 0000 0000b

Not bit addressable

Table 13-8. CL Register

CL - PCA Counter Register Low (0E9h)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

Bit Number	Bit Mnemonic	Description
7 - 0	-	**PCA Counter** CL Value

Reset Value = 0000 0000b

Not bit addressable

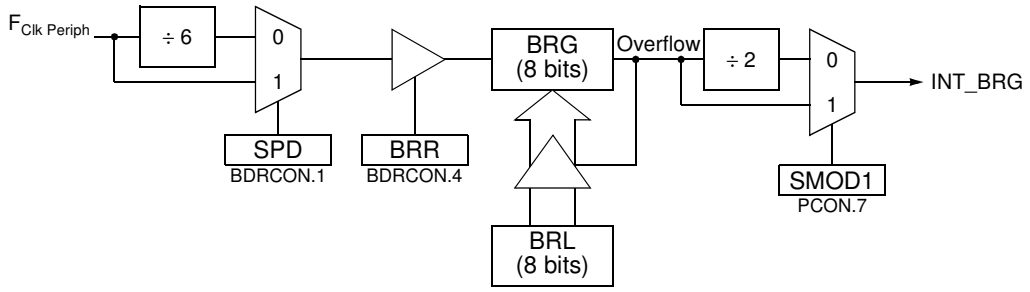
Table 14-3. Baud Rate Selection Table UART

TCLK (T2CON)	RCLK (T2CON)	TBCK (BDRCON)	RBCK (BDRCON)	Clock Source UART Tx	Clock Source UART Rx
0	0	0	0	Timer 1	Timer 1
1	0	0	0	Timer 2	Timer 1
0	1	0	0	Timer 1	Timer 2
1	1	0	0	Timer 2	Timer 2
X	0	1	0	INT_BRG	Timer 1
X	1	1	0	INT_BRG	Timer 2
0	X	0	1	Timer 1	INT_BRG
1	X	0	1	Timer 2	INT_BRG
X	X	1	1	INT_BRG	INT_BRG

14.4.1 Internal Baud Rate Generator (BRG)

When the internal Baud Rate Generator is used, the Baud Rates are determined by the BRG overflow depending on the BRL reload value, the value of SPD bit (Speed Mode) in BDRCON register and the value of the SMOD1 bit in PCON register.

Figure 14-5. Internal Baud Rate



- The baud rate for UART is token by formula:

$$\text{Baud\_Rate} = \frac{2^{SMOD1} \cdot F_{PER}}{6^{(1-SPD)} \cdot 32 \cdot (256 - BRL)}$$

$$BRL = 256 - \frac{2^{SMOD1} \cdot F_{PER}}{6^{(1-SPD)} \cdot 32 \cdot \text{Baud\_Rate}}$$

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

Reset Value = 0000 0000b

Bit addressable

Table 14-9. SBUF Register

SBUF - Serial Buffer Register for UART (99h)

7	6	5	4	3	2	1	0

Reset Value = XXXX XXXXb

Table 14-10. BRL Register

BRL - Baud Rate Reload Register for the internal baud rate generator, UART (9Ah)

7	6	5	4	3	2	1	0

Reset Value = 0000 0000b

Table 14-13. BDRCON Register

BDRCON - Baud Rate Control Register (9Bh)

7	6	5	4	3	2	1	0
-	-	-	BRR	TBCK	RBCK	SPD	SRC

Bit Number	Bit Mnemonic	Description
7	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit
6	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit
5	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
4	BRR	**Baud Rate Run Control bit** Cleared to stop the internal Baud Rate Generator. Set to start the internal Baud Rate Generator.
3	TBCK	**Transmission Baud rate Generator Selection bit for UART** Cleared to select Timer 1 or Timer 2 for the Baud Rate Generator. Set to select internal Baud Rate Generator.
2	RBCK	**Reception Baud Rate Generator Selection bit for UART** Cleared to select Timer 1 or Timer 2 for the Baud Rate Generator. Set to select internal Baud Rate Generator.
1	SPD	**Baud Rate Speed Control bit for UART** Cleared to select the SLOW Baud Rate Generator. Set to select the FAST Baud Rate Generator.
0	SRC	**Baud Rate Source select bit in Mode 0 for UART** Cleared to select $F_{OSC}/12$ as the Baud Rate Generator ($F_{CLK\ PERIPH}/6$ in X2 mode). Set to select the internal Baud Rate Generator for UARTs in mode 0.

Reset Value = XXX0 0000b

Not bit addressable

Table 15-3. KBLS Register

KBLS-Keyboard Level Selector Register (9Ch)

7	6	5	4	3	2	1	0
KBLS7	**KBLS6**	**KBLS5**	**KBLS4**	**KBLS3**	**KBLS2**	**KBLS1**	**KBLS0**

Bit Number	Bit Mnemonic	Description
7	KBLS7	**Keyboard line 7 Level Selection bit** Cleared to enable a low level detection on Port line 7. Set to enable a high level detection on Port line 7.
6	KBLS6	**Keyboard line 6 Level Selection bit** Cleared to enable a low level detection on Port line 6. Set to enable a high level detection on Port line 6.
5	KBLS5	**Keyboard line 5 Level Selection bit** Cleared to enable a low level detection on Port line 5. Set to enable a high level detection on Port line 5.
4	KBLS4	**Keyboard line 4 Level Selection bit** Cleared to enable a low level detection on Port line 4. Set to enable a high level detection on Port line 4.
3	KBLS3	**Keyboard line 3 Level Selection bit** Cleared to enable a low level detection on Port line 3. Set to enable a high level detection on Port line 3.
2	KBLS2	**Keyboard line 2 Level Selection bit** Cleared to enable a low level detection on Port line 2. Set to enable a high level detection on Port line 2.
1	KBLS1	**Keyboard line 1 Level Selection bit** Cleared to enable a low level detection on Port line 1. Set to enable a high level detection on Port line 1.
0	KBLS0	**Keyboard line 0 Level Selection bit** Cleared to enable a low level detection on Port line 0. Set to enable a high level detection on Port line 0.

Reset Value = 0000 0000b

16.2.3 SPI Serial Clock (SCK)

This signal is used to synchronize the data movement both in and out of the devices through their MOSI and MISO lines. It is driven by the Master for eight clock cycles which allows to exchange one Byte on the serial lines.

16.2.4 Slave Select ($\overline{SS}$)

Each Slave peripheral is selected by one Slave Select pin ($\overline{SS}$). This signal must stay low for any message for a Slave. It is obvious that only one Master ($\overline{SS}$ high level) can drive the network. The Master may select each Slave device by software through port pins (Figure 16-2). To prevent bus conflicts on the MISO line, only one slave should be selected at a time by the Master for a transmission.

In a Master configuration, the $\overline{SS}$ line can be used in conjunction with the MODF flag in the SPI Status register (SPSTA) to prevent multiple masters from driving MOSI and SCK (see Error conditions).

A high level on the $\overline{SS}$ pin puts the MISO line of a Slave SPI in a high-impedance state.

The $\overline{SS}$ pin could be used as a general-purpose if the following conditions are met:

- The device is configured as a Master and the SSDIS control bit in SPCON is set. This kind of configuration can be found when only one Master is driving the network and there is no way that the $\overline{SS}$ pin could be pulled low. Therefore, the MODF flag in the SPSTA will never be set(1).
- The Device is configured as a Slave with CPHA and SSDIS control bits set(2). This kind of configuration can happen when the system comprises one Master and one Slave only. Therefore, the device should always be selected and there is no reason that the Master uses the $\overline{SS}$ pin to select the communicating Slave device.

Note: 1. Clearing SSDIS control bit does not clear MODF.
2. Special care should be taken not to set SSDIS control bit when CPHA = '0' because in this mode, the $\overline{SS}$ is used to start the transmission.

16.2.5 Baud Rate

In Master mode, the baud rate can be selected from a baud rate generator which is controlled by three bits in the SPCON register: SPR2, SPR1 and SPR0.The Master clock is selected from one of seven clock rates resulting from the division of the internal clock by 2, 4, 8, 16, 32, 64 or 128.

Table 16-1 gives the different clock rates selected by SPR2:SPR1:SPR0.

Table 16-1. SPI Master Baud Rate Selection

SPR2	SPR1	SPR0	Clock Rate	Baud Rate Divisor (BD)
0	0	0	$F_{CLK\ PERIPH}$ /2	2
0	0	1	$F_{CLK\ PERIPH}$ /4	4
0	1	0	$F_{CLK\ PERIPH}$/8	8
0	1	1	$F_{CLK\ PERIPH}$ /16	16
1	0	0	$F_{CLK\ PERIPH}$ /32	32
1	0	1	$F_{CLK\ PERIPH}$ /64	64
1	1	0	$F_{CLK\ PERIPH}$ /128	128
1	1	1	Don't Use	No BRG

16.3.3.1 *Mode Fault (MODF)*

Mode Fault error in Master mode SPI indicates that the level on the Slave Select ($\overline{SS}$) pin is inconsistent with the actual mode of the device. MODF is set to warn that there may be a multi-master conflict for system control. In this case, the SPI system is affected in the following ways:

- An SPI receiver/error CPU interrupt request is generated
- The SPEN bit in SPCON is cleared. This disables the SPI
- The MSTR bit in SPCON is cleared

When $\overline{SS}$ Disable (SSDIS) bit in the SPCON register is cleared, the MODF flag is set when the $\overline{SS}$ signal becomes '0'.

However, as stated before, for a system with one Master, if the $\overline{SS}$ pin of the Master device is pulled low, there is no way that another Master attempts to drive the network. In this case, to prevent the MODF flag from being set, software can set the SSDIS bit in the SPCON register and therefore making the $\overline{SS}$ pin as a general-purpose I/O pin.

Clearing the MODF bit is accomplished by a read of SPSTA register with MODF bit set, followed by a write to the SPCON register. SPEN Control bit may be restored to its original set state after the MODF bit has been cleared.

16.3.3.2 *Write Collision (WCOL)*

A Write Collision (WCOL) flag in the SPSTA is set when a write to the SPDAT register is done during a transmit sequence.

WCOL does not cause an interruption, and the transfer continues uninterrupted.

Clearing the WCOL bit is done through a software sequence of an access to SPSTA and an access to SPDAT.

16.3.3.3 *Overrun Condition*

An overrun condition occurs when the Master device tries to send several data Bytes and the Slave devise has not cleared the SPIF bit issuing from the previous data Byte transmitted. In this case, the receiver buffer contains the Byte sent after the SPIF bit was last cleared. A read of the SPDAT returns this Byte. All others Bytes are lost.

This condition is not detected by the SPI peripheral.

16.3.3.4 *SS Error Flag (SSERR)*

A Synchronous Serial Slave Error occurs when $\overline{SS}$ goes high before the end of a received data in slave mode. SSERR does not cause in interruption, this bit is cleared by writing 0 to SPEN bit (reset of the SPI state machine).

16.3.4 Interrupts

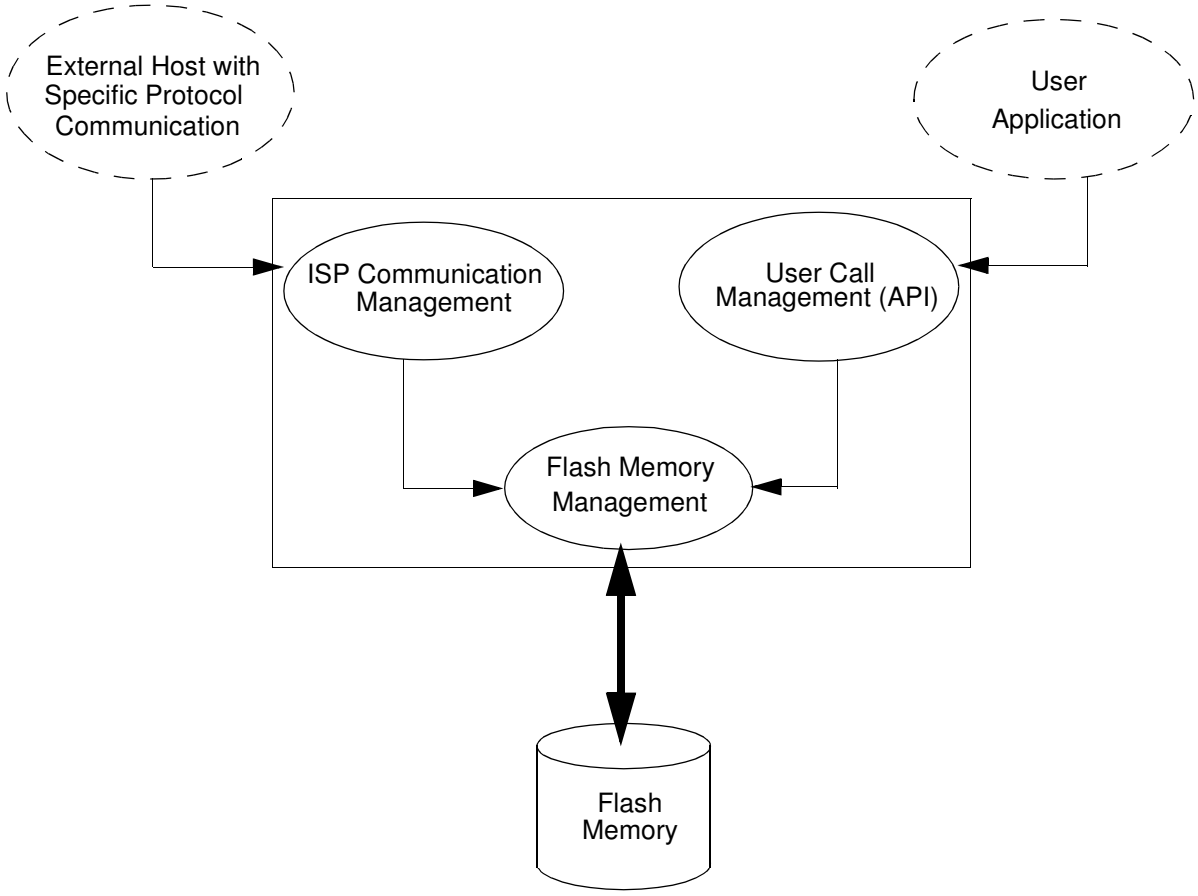
Two SPI status flags can generate a CPU interrupt requests:

Table 16-2. SPI Interrupts

Flag	Request
SPIF (SP data transfer)	SPI Transmitter Interrupt request
MODF (Mode Fault)	SPI Receiver/Error Interrupt Request (if SSDIS = '0')

24.6.3 Functional Description

Figure 24-3. Bootloader Functional Description



On the above diagram, the on-chip bootloader processes are:

- ISP Communication Management

The purpose of this process is to manage the communication and its protocol between the on-chip bootloader and a external device. The on-chip ROM implements a serial protocol (see section “Bootloader Protocol”). This process translate serial communication frame (UART) into Flash memory access (read, write, erase, etc.).

- User Call Management

Several Application Program Interface (API) calls are available for use by an application program to permit selective erasing and programming of Flash pages. All calls are made through a common interface (API calls), included in the ROM bootloader. The programming functions are selected by setting up the microcontroller’s registers before making a call to a common entry point (0xFFF0). Results are returned in the registers. The purpose on this process is to translate the registers values into internal Flash Memory Management.

- Flash Memory Management

This process manages low level access to Flash memory (performs read and write access).

24.6.4 Bootloader Functionality

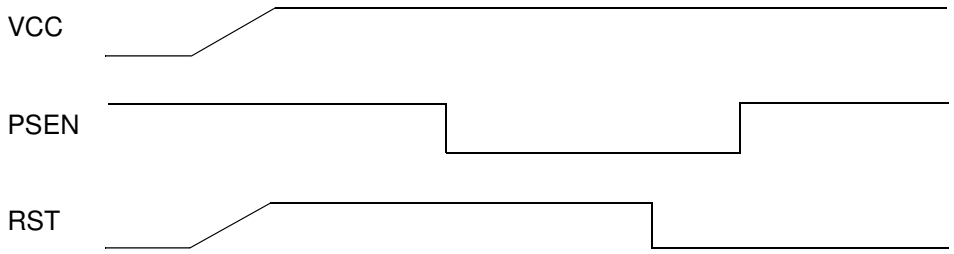
The bootloader can be activated by two means: Hardware conditions or regular boot process.

The Hardware conditions (EA = 1, PSEN = 0) during the Reset# falling edge force the on-chip bootloader execution. This allows an application to be built that will normally execute the end user's code but can be manually forced into default ISP operation.

As PSEN is a an output port in normal operating mode after reset, user application should take care to release PSEN after falling edge of reset signal. The hardware conditions are sampled at reset signal falling edge, thus they can be released at any time when reset input is low.

To ensure correct microcontroller startup, the PSEN pin should not be tied to ground during power-on (See Figure 24-4).

Figure 24-4. Hardware conditions typical sequence during power-on.

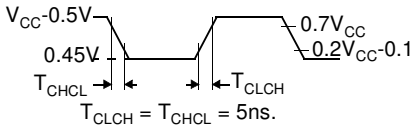


The on-chip bootloader boot process is shown Figure 24-5.

Table 24-6. Bootloader Process Description

	Purpose
Hardware Conditions	The Hardware Conditions force the bootloader execution whatever BLJB, BSB and SBV values.
BLJB	The Boot Loader Jump Bit forces the application execution. BLJB = 0 => Bootloader execution BLJB = 1 => Application execution The BLJB is a fuse bit in the Hardware Byte. It can be modified by hardware (programmer) or by software (API). Note: The BLJB test is performed by hardware to prevent any program execution.
SBV	The Software Boot Vector contains the high address of customer bootloader stored in the application. SBV = FCh (default value) if no customer bootloader in user Flash. Note: The customer bootloader is called by JMP [SBV]00h instruction.

Figure 25-4. Clock Signal Waveform for I_{CC} Tests in Active and Idle Modes



25.3 AC Parameters

25.3.1 Explanation of the AC Symbols

Each timing symbol has 5 characters. The first character is always a “T” (stands for time). The other characters, depending on their positions, stand for the name of a signal or the logical status of that signal. The following is a list of all the characters and what they stand for.

Example:T_{AVLL} = Time for Address Valid to ALE Low.
T_{LLPL} = Time for ALE Low to $\overline{PSEN}$ Low.

(Load Capacitance for port 0, ALE and PSEN = 100 pF; Load Capacitance for all other outputs = 80 pF.)

Table 25-1 Table 25-4, and Table 25-7 give the description of each AC symbols.

Table 25-2, Table 25-3, Table 25-5 and Table 25-8 gives the range for each AC parameter.

Table 25-2, Table 25-3 and Table 25-9 give the frequency derating formula of the AC parameter for each speed range description. To calculate each AC symbols. take the x value in the corresponding column (-M) and use this value in the formula.

Example: T_{LLIU} for -M and 20 MHz, Standard clock.
x = 35 ns
T 50 ns
T_{CCIV} = 4T - x = 165 ns

STANDARD NOTES FOR PLCC

1/ CONTROLLING DIMENSIONS : INCHES

2/ DIMENSIONING AND TOLERANCING PER ANSI Y 14.5M - 1982.

3/ "D" AND "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTUSIONS. MOLD FLASH OR PROTUSIONS SHALL NOT EXCEED 0.20 mm (.008 INCH) PER SIDE.

STANDARD NOTES FOR PQFP/ VQFP / TQFP / DQFP

1/ CONTROLLING DIMENSIONS : INCHES

2/ ALL DIMENSIONING AND TOLERANCING CONFORM TO ANSI Y 14.5M - 1982.

3/ "D1 AND E1" DIMENSIONS DO NOT INCLUDE MOLD PROTUSIONS. MOLD PROTUSIONS SHALL NOT EXCEED 0.25 mm (0.010 INCH). THE TOP PACKAGE BODY SIZE MAY BE SMALLER THAN THE BOTTOM PACKAGE BODY SIZE BY AS MUCH AS 0.15 mm.

4/ DATUM PLANE "H" LOCATED AT MOLD PARTING LINE AND COINCIDENT WITH LEAD, WHERE LEAD EXITS PLASTIC BODY AT BOTTOM OF PARTING LINE.

5/ DATUM "A" AND "D" TO BE DETERMINED AT DATUM PLANE H.

6/ DIMENSION " f " DOES NOT INCLUDE DAMBAR PROTUSION ALLOWABLE DAMBAR PROTUSION SHALL BE 0.08mm/.003" TOTAL IN EXCESS OF THE " f " DIMENSION AT MAXIMUM MATERIAL CONDITION . DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT.