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	60MHz
Connectivity	SPI, UART/USART
Peripherals	POR, PWM, WDT
Number of I/O	32
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1.25K 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	44-LQFP
Supplier Device Package	44-VQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/at89c51rc2-rltum

Table 2. C51 Core SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
ACC	E0h	Accumulator								
B	F0h	B Register								
PSW	D0h	Program Status Word	CY	AC	F0	RS1	RS0	OV	F1	P
SP	81h	Stack Pointer								
DPL	82h	Data Pointer Low Byte								
DPH	83h	Data Pointer High Byte								

Table 3. System Management SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
PCON	87h	Power Control	SMOD1	SMOD0	-	POF	GF1	GF0	PD	IDL
AUXR	8Eh	Auxiliary Register 0	DPU	-	M0	XRS2	XRS1	XRS0	EXTRAM	AO
AUXR1	A2h	Auxiliary Register 1	-	-	ENBOOT	-	GF3	0	-	DPS
CKRL	97h	Clock Reload Register	CKRL7	CKRL6	CKRL5	CKRL4	CKRL3	CKRL2	CKRL1	CKRL0
CKCKON0	8Fh	Clock Control Register 0	-	WDTX2	PCAX2	SIX2	T2X2	T1X2	T0X2	X2
CKCKON1	AFh	Clock Control Register 1	-	-	-	-	-	-	-	SPIX2

Table 4. Interrupt SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
IEN0	A8h	Interrupt Enable Control 0	EA	EC	ET2	ES	ET1	EX1	ET0	EX0
IEN1	B1h	Interrupt Enable Control 1	-	-	-	-	-	ESPI	EI2C	KBD
IPH0	B7h	Interrupt Priority Control High 0	-	PPCH	PT2H	PHS	PT1H	PX1H	PT0H	PX0H
IPL0	B8h	Interrupt Priority Control Low 0	-	PPCL	PT2L	PLS	PT1L	PX1L	PT0L	PX0L
IPH1	B3h	Interrupt Priority Control High 1	-	-	-	-	-	SPIH	IE2CH	KBDH
IPL1	B2h	Interrupt Priority Control Low 1	-	-	-	-	-	SPIL	IE2CL	KBDL

Table 5. Port SFRs

Mnemonic	Add	Name	7	6	5	4	3	2	1	0
P0	80h	8-bit Port 0								
P1	90h	8-bit Port 1								
P2	A0h	8-bit Port 2								
P3	B0h	8-bit Port 3								

- Instructions that use indirect addressing access the Upper 128 Bytes of data RAM. For example: MOV @R0, # data where R0 contains 0A0h, accesses the data Byte at address 0A0h, rather than P2 (whose address is 0A0h).
- The XRAM Bytes can be accessed by indirect addressing, with EXTRAM bit cleared and MOVX instructions. This part of memory that is physically located on-chip, logically occupies the first Bytes of external data memory. The bits XRS0 and XRS1 are used to hide a part of the available XRAM as explained in Table 18. 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.

Timer 2

The Timer 2 in the AT89C51RB2/RC2 is the standard C52 Timer 2.

It is a 16-bit timer/counter: the count is maintained by two eight-bit timer registers, TH2 and TL2 are cascaded. It is controlled by T2CON (Table 20) and T2MOD (Table 21) registers. Timer 2 operation is similar to Timer 0 and Timer 1C/$\overline{T2}$ selects F_{OSC}/12 (timer operation) or external pin T2 (counter operation) as the timer clock input. Setting TR2 allows TL2 to increment by the selected input.

Timer 2 has 3 operating modes: capture, autoreload and Baud Rate Generator. These modes are selected by the combination of RCLK, TCLK and CP/$\overline{RL2}$ (T2CON).

see the Atmel 8-bit Microcontroller Hardware description for the description of Capture and Baud Rate Generator Modes.

Timer 2 includes the following enhancements:

- Auto-reload mode with up or down counter
- Programmable clock-output

Auto-reload Mode

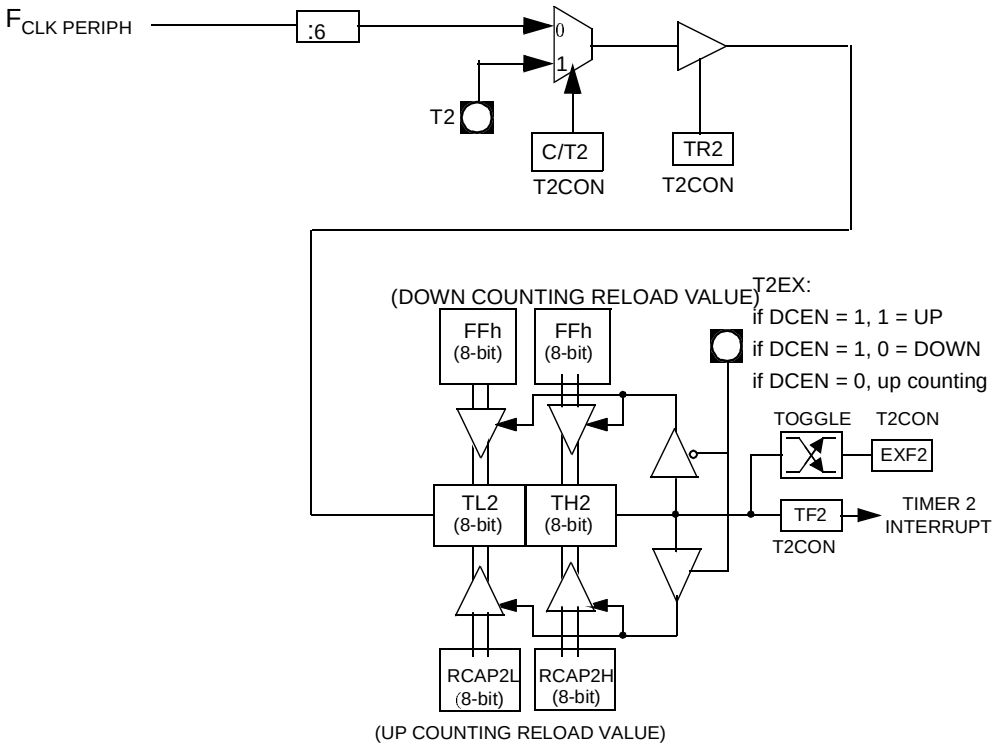
The auto-reload mode configures Timer 2 as a 16-bit timer or event counter with automatic reload. If DCEN bit in T2MOD is cleared, Timer 2 behaves as in 80C52 (see the Atmel C51 Microcontroller Hardware description). If DCEN bit is set, Timer 2 acts as an Up/down timer/counter as shown in Figure 9. In this mode the T2EX pin controls the direction of count.

When T2EX is high, Timer 2 counts up. Timer overflow occurs at FFFFh which sets the TF2 flag and generates an interrupt request. The overflow also causes the 16-bit value in RCAP2H and RCAP2L registers to be loaded into the timer registers TH2 and TL2.

When T2EX is low, Timer 2 counts down. Timer underflow occurs when the count in the timer registers TH2 and TL2 equals the value stored in RCAP2H and RCAP2L registers. The underflow sets TF2 flag and reloads FFFFh into the timer registers.

The EXF2 bit toggles when Timer 2 overflows or underflows according to the direction of the count. EXF2 does not generate any interrupt. This bit can be used to provide 17-bit resolution.

Figure 9. Auto-Reload Mode Up/Down Counter (DCEN = 1)



Programmable Clock-out Mode

In the clock-out mode, Timer 2 operates as a 50% duty-cycle, programmable clock generator (see Figure 10). The input clock increments TL2 at frequency $F_{CLK\ PERIPH}/2$. The timer repeatedly counts to overflow from a loaded value. At overflow, the contents of RCAP2H and RCAP2L registers are loaded into TH2 and TL2. In this mode, Timer 2 overflows do not generate interrupts. The formula gives the clock-out frequency as a function of the system oscillator frequency and the value in the RCAP2H and RCAP2L registers:

$$Clock\text{–}OutFrequency = \frac{F_{CLKPERIPH}}{4 \times (65536 - RCAP2H/RCAP2L)}$$

For a 16 MHz system clock, Timer 2 has a programmable frequency range of 61 Hz ($F_{CLK\ PERIPH}/2^{16}$) to 4 MHz ($F_{CLK\ PERIPH}/4$). The generated clock signal is brought out to T2 pin (P1.0).

Timer 2 is programmed for the clock-out mode as follows:

- Set T2OE bit in T2MOD register.
- Clear C/$\overline{T2}$ bit in T2CON register.
- Determine the 16-bit reload value from the formula and enter it in RCAP2H/RCAP2L registers.
- Enter a 16-bit initial value in timer registers TH2/TL2. It can be the same as the reload value or a different one depending on the application.
- To start the timer, set TR2 run control bit in T2CON register.

It is possible to use Timer 2 as a baud rate generator and a clock generator simultaneously. For this configuration, the baud rates and clock frequencies are not independent since both functions use the values in the RCAP2H and RCAP2L registers.

Registers

Table 20. T2CON Register

T2CON – Timer 2 Control Register (C8h)

7	6	5	4	3	2	1	0
TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2#	CP/RL2#

Bit Number	Bit Mnemonic	Description
7	TF2	**Timer 2 Overflow Flag** Must be cleared by software. Set by hardware on Timer 2 overflow, if RCLK = 0 and TCLK = 0.
6	EXF2	**Timer 2 External Flag** Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2 = 1. When set, causes the CPU to vector to Timer 2 interrupt routine when Timer 2 interrupt is enabled. Must be cleared by software. EXF2 doesn't cause an interrupt in Up/down counter mode (DCEN = 1).
5	RCLK	**Receive Clock Bit** Cleared to use timer 1 overflow as receive clock for serial port in mode 1 or 3. Set to use Timer 2 overflow as receive clock for serial port in mode 1 or 3.
4	TCLK	**Transmit Clock Bit** Cleared to use timer 1 overflow as transmit clock for serial port in mode 1 or 3. Set to use Timer 2 overflow as transmit clock for serial port in mode 1 or 3.
3	EXEN2	**Timer 2 External Enable Bit** Cleared to ignore events on T2EX pin for Timer 2 operation. Set to cause a capture or reload when a negative transition on T2EX pin is detected, if Timer 2 is not used to clock the serial port.
2	TR2	**Timer 2 Run Control Bit** Cleared to turn off Timer 2. Set to turn on Timer 2.
1	C/T2#	**Timer/Counter 2 Select Bit** Cleared for timer operation (input from internal clock system: $F_{CLK\ PERIPH}$). Set for counter operation (input from T2 input pin, falling edge trigger). Must be 0 for clock out mode.
0	CP/RL2#	**Timer 2 Capture/Reload Bit** If RCLK = 1 or TCLK = 1, CP/RL2# is ignored and timer is forced to auto-reload on Timer 2 overflow. Cleared to auto-reload on Timer 2 overflows or negative transitions on T2EX pin if EXEN2 = 1. Set to capture on negative transitions on T2EX pin if EXEN2 = 1.

Reset Value = 0000 0000b
Bit addressable

Table 21. T2MOD Register

T2MOD – Timer 2 Mode Control Register (C9h)

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

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	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
3	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
2	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
1	T2OE	**Timer 2 Output Enable Bitt** Cleared to program P1.0/T2 as clock input or I/O port. Set to program P1.0/T2 as clock output.
0	DCEN	**Down Counter Enable Bit** Cleared to disable Timer 2 as up/down counter. Set to enable Timer 2 as up/down counter.

Reset Value = XXXX XX00b
Not bit addressable

Registers

Table 22. CMOD Register

CMOD – PCA Counter Mode Register (D9h)

7	6	5	4	3	2	1	0
CIDL	**WDTE**	**-**	**-**	**-**	**CPS1**	**CPS0**	**ECF**

Bit Number	Bit Mnemonic	Description
7	CIDL	**Counter Idle Control** Cleared to program the PCA Counter to continue functioning during idle Mode. Set to program PCA to be gated off during idle.
6	WDTE	**Watchdog Timer Enable** Cleared to disable Watchdog Timer function on PCA Module 4. Set to enable Watchdog Timer function on PCA Module 4.
5	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
4	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
3	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
2	CPS1	**PCA Count Pulse Select** CPS1 CPS0 Selected PCA input 0 0 Internal clock $F_{CLK\ PERIPH}/6$ 0 1 Internal clock $F_{LK\ PERIPH}/2$ 1 0 Timer 0 Overflow 1 1 External clock at ECI/P1.2 pin (max rate = fCLK PERIPH/ 4)
1	CPS0	
0	ECF	**PCA Enable Counter Overflow Interrupt** Cleared to disable CF bit in CCON to inhibit an interrupt. Set to enable CF bit in CCON to generate an interrupt.

Reset Value = 00XX X000b
Not bit addressable

The CMOD register includes three additional bits associated with the PCA.

- The CIDL bit which allows the PCA to stop during idle mode.
- The WDTE bit which enables or disables the watchdog function on Module 4.
- The ECF bit which when set causes an interrupt and the PCA overflow flag CF (in the CCON SFR) to be set when the PCA timer overflows.

The CCON register contains the run control bit for the PCA and the flags for the PCA timer (CF) and each Module (see Table 23).

- Bit CR (CCON. 6) must be set by software to run the PCA. The PCA is shut off by clearing this bit.
- Bit CF: The CF bit (CCON. 7) is set when the PCA counter overflows and an interrupt will be generated if the ECF bit in the CMOD register is set. The CF bit can only be cleared by software.
- Bits 0 through 4 are the flags for the Modules (bit 0 for Module 0, bit 1 for Module 1, etc.) and are set by hardware when either a match or a capture occurs. These flags also can only be cleared by software.

Table 23. CCON Register

CCON – PCA Counter Control Register (D8h)

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

Bit Number	Bit Mnemonic	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** Must be cleared by software to turn the PCA counter off. Set by software to turn the PCA counter on.
5	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
4	CCF4	**PCA Module 4 Interrupt Flag** Must be cleared by software. Set by hardware when a match or capture occurs.
3	CCF3	**PCA Module 3 Interrupt Flag** Must be cleared by software. Set by hardware when a match or capture occurs.
2	CCF2	**PCA Module 2 Interrupt Flag** Must be cleared by software. Set by hardware when a match or capture occurs.
1	CCF1	**PCA Module 1 Interrupt Flag** Must be cleared by software. Set by hardware when a match or capture occurs.
0	CCF0	**PCA Module 0 Interrupt Flag** Must be cleared by software. Set by hardware when a match or capture occurs.

Reset Value = 000X 0000b
Bit addressable

The watchdog timer function is implemented in Module 4 (see Figure 14).

The PCA interrupt system is shown in Figure 12.

Table 24. CCAPMn Registers (n = 0-4)

CCAPM0 – PCA Module 0 Compare/Capture Control Register (0DAh)

CCAPM1 – PCA Module 1 Compare/Capture Control Register (0DBh)

CCAPM2 – PCA Module 2 Compare/Capture Control Register (0DCh)

CCAPM3 – PCA Module 3 Compare/Capture Control Register (0DDh)

CCAPM4 – PCA Module 4 Compare/Capture Control Register (0DEh)

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

Bit Number	Bit Mnemonic	Description
7	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
6	ECOMn	**Enable Comparator** Cleared to disable the comparator function. Set to enable the comparator function.
5	CAPPn	**Capture Positive** Cleared to disable positive edge capture. Set to enable positive edge capture.
4	CAPNn	**Capture Negative** Cleared to disable negative edge capture. Set to enable negative edge capture.
3	MATn	**Match** When MATn = 1, a match of the PCA counter with this Module's compare/capture register causes the CCFn bit in CCON to be set, flagging an interrupt.
2	TOGn	**Toggle** When TOGn = 1, a match of the PCA counter with this Module's compare/capture register causes theCEXn pin to toggle.
1	PWMn	**Pulse Width Modulation Mode** Cleared to disable the CEXn pin to be used as a pulse width modulated output. Set to enable the CEXn pin to be used as a pulse width modulated output.
0	CCF0	**Enable CCF Interrupt** Cleared to disable compare/capture flag CCFn in the CCON register to generate an interrupt. Set to enable compare/capture flag CCFn in the CCON register to generate an interrupt.

Reset Value = X000 0000b
Not bit addressable

Table 27. 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 28. 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 29. 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

A low-priority interrupt can be interrupted by a high-priority interrupt, but not by another low-priority interrupt. A high-priority interrupt can't be interrupted by any other interrupt source.

Table 43. Priority Level Bit Values

IPH. x	IPL. x	Interrupt Level Priority
0	0	0 (Lowest)
0	1	1
1	0	2
1	1	3 (Highest)

If two interrupt requests of different priority levels are received simultaneously, the request of higher-priority level is serviced. If interrupt requests of the same priority level are received simultaneously, an internal polling sequence determines which request is serviced. Thus within each priority level there is a second priority structure determined by the polling sequence.

Table 47. IEN1 Register

IEN1 - Interrupt Enable Register (B1h)

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

Bit Number	Bit Mnemonic	Description
7	-	**Reserved**
6	-	**Reserved**
5	-	**Reserved**
4	-	**Reserved**
3	-	**Reserved**
2	ESPI	**SPI Interrupt Enable Bit** Cleared to disable SPI interrupt. Set to enable SPI interrupt.
1	-	Reserved
0	KBD	**Keyboard Interrupt Enable Bit** Cleared to disable keyboard interrupt. Set to enable keyboard interrupt.

Reset Value = XXXX X000b
Bit addressable

Interrupt Sources and Vector Addresses

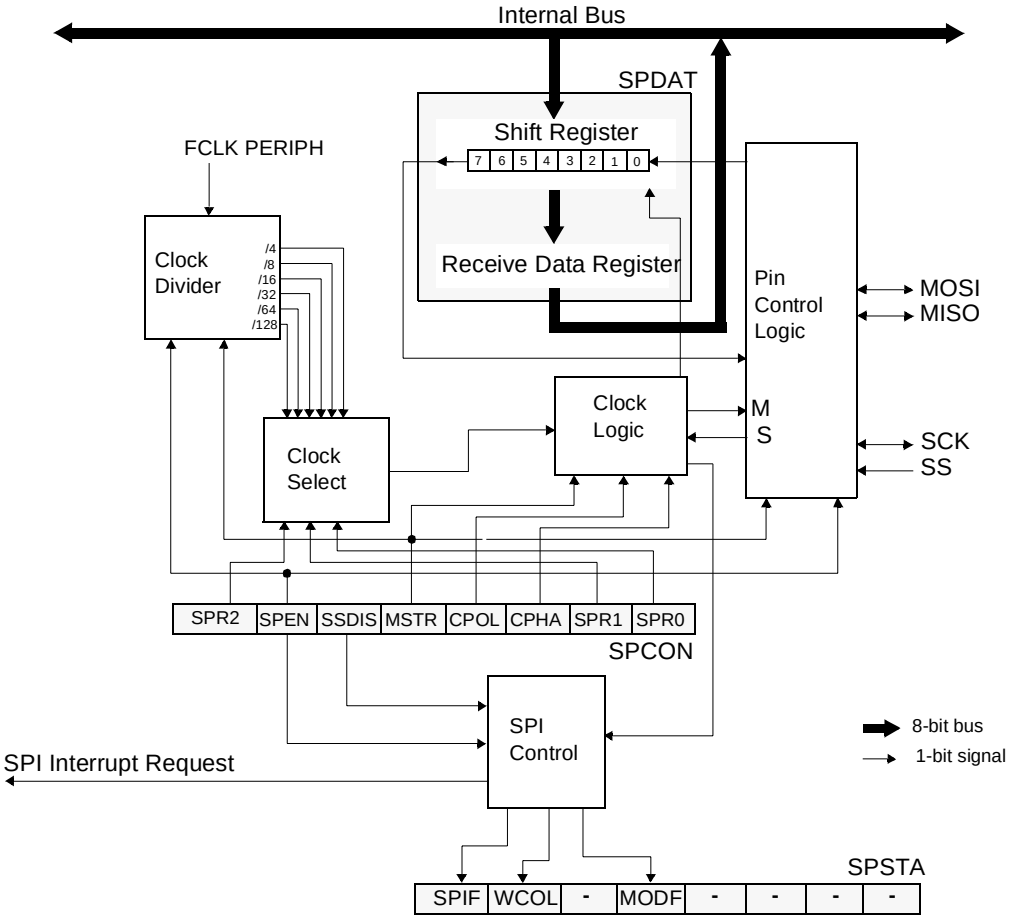
Table 50. Interrupt Sources and Vector Addresses

Number	Polling Priority	Interrupt Source	Interrupt Request	Vector Address
0	0	Reset		0000h
1	1	INT0	IE0	0003h
2	2	Timer 0	TF0	000Bh
3	3	INT1	IE1	0013h
4	4	Timer 1	IF1	001Bh
5	6	UART	RI+TI	0023h
6	7	Timer 2	TF2+EXF2	002Bh
7	5	PCA	CF + CCFn (n = 0-4)	0033h
8	8	Keyboard	KBDIT	003Bh
9	9	SPI	SPIIT	004Bh

Functional Description

Figure 26 shows a detailed structure of the SPI Module.

Figure 26. SPI Module Block Diagram



Operating Modes

The Serial Peripheral Interface can be configured in one of the two modes: Master mode or Slave mode. The configuration and initialization of the SPI Module is made through one register:

- The Serial Peripheral Control register (SPCON)

Once the SPI is configured, the data exchange is made using:

- SPCON
- The Serial Peripheral STAtus register (SPSTA)
- The Serial Peripheral DATa register (SPDAT)

During an SPI transmission, data is simultaneously transmitted (shifted out serially) and received (shifted in serially). A serial clock line (SCK) synchronizes shifting and sampling on the two serial data lines (MOSI and MISO). A Slave Select line ($\overline{SS}$) allows individual selection of a Slave SPI device; Slave devices that are not selected do not interfere with SPI bus activities.

When the Master device transmits data to the Slave device via the MOSI line, the Slave device responds by sending data to the Master device via the MISO line. This implies full-duplex transmission with both data out and data in synchronized with the same clock (Figure 27).

Table 69. Program Lock Bits of the SSB

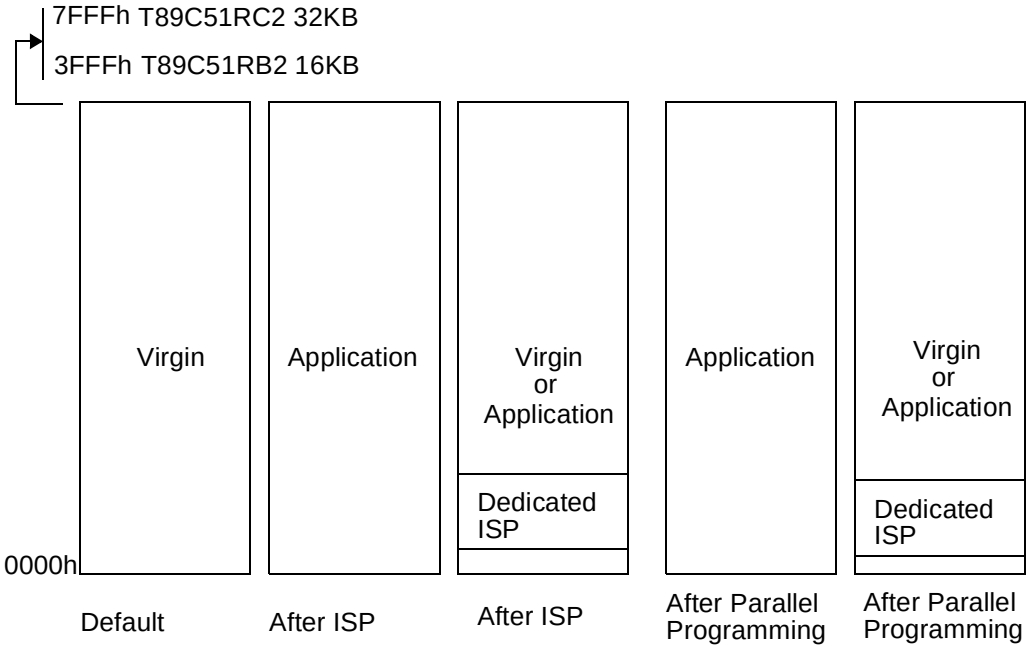
Program Lock Bits			
Security level	**LB0**	**LB1**	**Protection Description**
1	U	U	No program lock features enabled.
2	P	U	ISP programming of the Flash is disabled.
3	X	P	Same as 2, also verify through ISP programming interface is disabled.

Note: U: unprogrammed or "one" level.
P: programmed or "zero" level.
X: don't care
WARNING: Security level 2 and 3 should only be programmed after Flash and code verification.

Flash Memory Status

AT89C51RB2/RC2 parts are delivered in standard with the ISP boot in the Flash memory. After ISP or parallel programming, the possible contents of the Flash memory are summarized on Figure 35.

Figure 35. Flash Memory Possible Contents

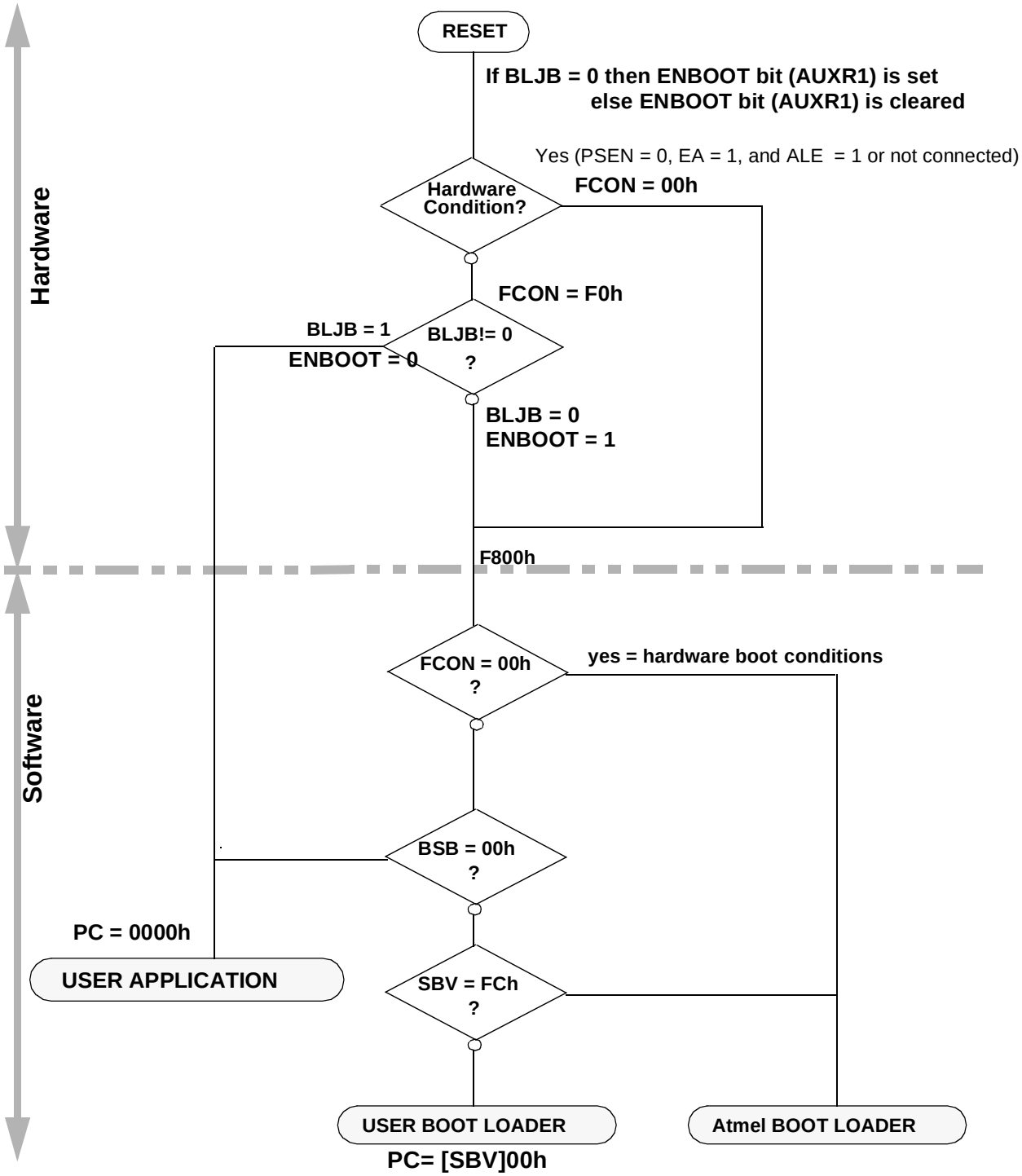


Memory Organization

In the AT89C51RB2/RC2, the lowest 16K or 32K of the 64 KB program memory address space is filled by internal Flash.

When the $\overline{EA}$ pin is high, the processor fetches instructions from internal program Flash. Bus expansion for accessing program memory from 16K or 32K upward automatic since external instruction fetches occur automatically when the program counter exceeds 3FFFh (16K) or 7FFFh (32K). If the $\overline{EA}$ pin is tied low, all program memory fetches are from external memory.

Figure 39. Bootloader process



Functional Description

Software Security Bits (SSB)

The SSB protects any Flash access from ISP command.
The command "Program Software Security bit" can only write a higher priority level.

There are three levels of security:

- level 0: ***NO_SECURITY*** (FFh)

This is the default level.
From level 0, one can write level 1 or level 2.

- level 1: ***WRITE_SECURITY*** (FEh)

For this level it is impossible to write in the Flash memory, BSB and SBV.
The Bootloader returns 'P' on write access.
From level 1, one can write only level 2.

- level 2: ***RD_WR_SECURITY*** (FCh

The level 2 forbids all read and write accesses to/from the Flash/EEPROM memory.
The Bootloader returns 'L' on read or write access.
Only a full chip erase in parallel mode (using a programmer) or ISP command can reset the software security bits.
From level 2, one cannot read and write anything.

Table 71. Software Security Byte Behavior

	Level 0	Level 1	Level 2
Flash/EEPROM	Any access allowed	Read only access allowed	Any access not allowed
Fuse Bit	Any access allowed	Read only access allowed	Any access not allowed
BSB & SBV	Any access allowed	Read only access allowed	Any access not allowed
SSB	Any access allowed	Write level 2 allowed	Read only access allowed
Manufacturer Info	Read only access allowed	Read only access allowed	Read only access allowed
Bootloader Info	Read only access allowed	Read only access allowed	Read only access allowed
Erase Block	Allowed	Not allowed	Not allowed
Full-chip Erase	Allowed	Allowed	Allowed
Blank Check	Allowed	Allowed	Allowed

API Call Description

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, PGM_MTP. The programming functions are selected by setting up the microcontroller's registers before making a call to PGM_MTP at FFF0h. Results are returned in the registers.

When several Bytes have to be programmed, it is highly recommended to use the Atmel API "PROGRAM DATA PAGE" call. Indeed, this API call writes up to 128 Bytes in a single command.

All routines for software access are provided in the C Flash driver available at Atmel's web site.

The API calls description and arguments are shown in Table 74.

Table 74. API Call Summary

Command	R1	A	DPTR0	DPTR1	Returned Value	Command Effect
READ MANUF ID	00h	XXh	0000h	XXh	ACC = Manufacturer Id	Read Manufacturer identifier
READ DEVICE ID1	00h	XXh	0001h	XXh	ACC = Device Id 1	Read Device identifier 1
READ DEVICE ID2	00h	XXh	0002h	XXh	ACC = Device Id 2	Read Device identifier 2
READ DEVICE ID3	00h	XXh	0003h	XXh	ACC = Device Id 3	Read Device identifier 3
ERASE BLOCK	01h	XXh	DPH = 00h	00h	ACC = DPH	Erase block 0
			DPH = 20h			Erase block 1
			DPH = 40h			Erase block 2
PROGRAM DATA BYTE	02h	Vaue to write	Address of byte to program	XXh	ACC = 0: DONE	Program up one data byte in the on-chip flash memory.
PROGRAM SSB	05h	XXh	DPH = 00h DPL = 00h	00h	ACC = SSB value	Set SSB level 1
			DPH = 00h DPL = 01h			Set SSB level 2
			DPH = 00h DPL = 10h			Set SSB level 0
			DPH = 00h DPL = 11h			Set SSB level 1
PROGRAM BSB	06h	New BSB value	0000h	XXh	none	Program boot status byte
PROGRAM SBV	06h	New SBV value	0001h	XXh	none	Program software boot vector
READ SSB	07h	XXh	0000h	XXh	ACC = SSB	Read Software Security Byte
READ BSB	07h	XXh	0001h	XXh	ACC = BSB	Read Boot Status Byte
READ SBV	07h	XXh	0002h	XXh	ACC = SBV	Read Software Boot Vector
PROGRAM DATA PAGE	09h	Number of byte to program	Address of the first byte to program in the Flash memory	Address in XRAM of the first data to program	ACC = 0: DONE	Program up to 128 bytes in user Flash. Remark: number of bytes to program is limited such as the Flash write remains in a single 128 bytes page. Hence, when ACC is 128, valid values of DPL are 00h, or, 80h.

AC Parameters

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 75 Table 78, and Table 80 give the description of each AC symbols.

Table 77, Table 79 and Table 81 give the AC parameterfor each range.

Table 76, Table 77 and Table 82 gives the frequency derating formula of the AC parameter for each speed range description. To calculate each AC symbols, take the x value in the correponding column (-M or -L) and use this value in the formula.

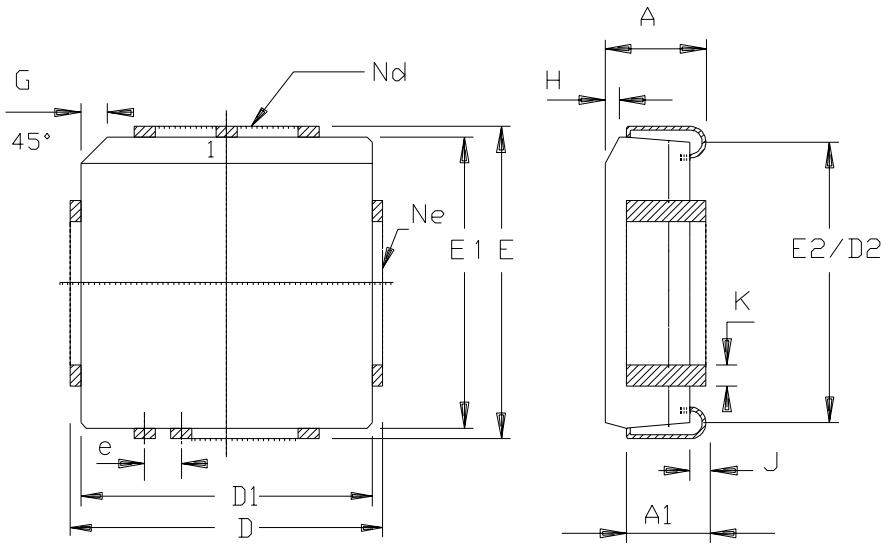
Example: T_{LLIU} for -M and 20 MHz, Standard clock.
x = 35 ns
T 50 ns
T_{CCIV} = 4T - x = 165 ns

External Program Memory Characteristics

Table 75. Symbol Description

Symbol	Parameter
T	Oscillator clock period
T_{LHLL}	ALE pulse width
T_{AVLL}	Address Valid to ALE
T_{LLAX}	Address Hold after ALE
T_{LLIV}	ALE to Valid Instruction In
T_{LLPL}	ALE to $\overline{PSEN}$
T_{PLPH}	$\overline{PSEN}$ Pulse Width
T_{PLIV}	$\overline{PSEN}$ to Valid Instruction In
T_{PXIX}	Input Instruction Hold after $\overline{PSEN}$
T_{PXIZ}	Input Instruction Float after $\overline{PSEN}$
T_{AVIV}	Address to Valid Instruction In
T_{PLAZ}	$\overline{PSEN}$ Low to Address Float

PLC44



	MM		INCH	
A	4.20	4.57	.165	.180
A1	2.29	3.04	.090	.120
D	17.40	17.65	.685	.695
D1	16.44	16.66	.647	.656
D2	14.99	16.00	.590	.630
E	17.40	17.65	.685	.695
E1	16.44	16.66	.647	.656
E2	14.99	16.00	.590	.630
e	1.27	BSC	.050	BSC
G	1.07	1.22	.042	.048
H	1.07	1.42	.042	.056
J	0.51	–	.020	–
K	0.33	0.53	.013	.021
Nd	11		11	
Ne	11		11	
PKG STD	00			