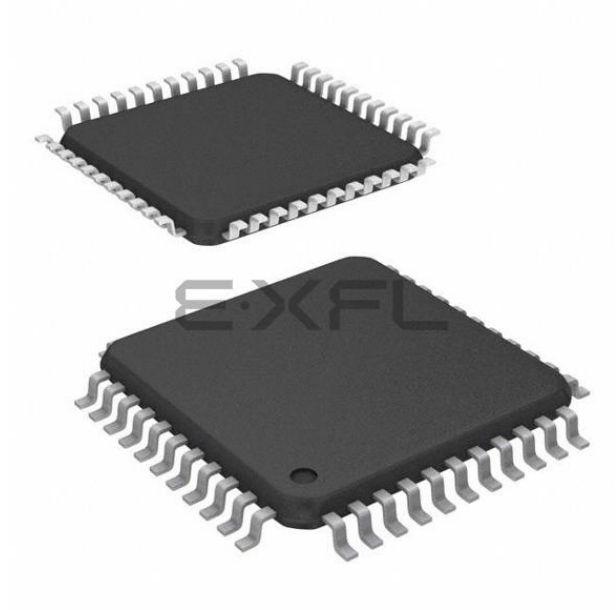


Welcome to E-XFL.COM

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	40MHz
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	1.2K 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-LQFP
Supplier Device Package	44-VQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/t89c51rc2-rltim

Table 3. Pin Description for 40 - 44 Pin Packages (Continued)

Mnemonic	Pin Number			Type	Name and Function
	DIL	LCC	VQFP44 1.4		
				I/O	**CEX4:** Capture/Compare External I/O for PCA module 4
P1.0 - P1.7				I/O	**MOSI**: SPI Master Output Slave Input line When SPI is in master mode, MOSI outputs data to the slave peripheral. When SPI is in slave mode, MOSI receives data from the master controller.
XTAL1	19	21	15	I	**Crystal 1:** Input to the inverting oscillator amplifier and input to the internal clock generator circuits.
XTAL2	18	20	14	O	**Crystal 2:** Output from the inverting oscillator amplifier
P2.0 - P2.7	21 - 28	24 - 31	18 - 25	I/O	**Port 2**: Port 2 is an 8 - bit bidirectional I/O port with internal pull - ups. Port 2 pins that have 1s written to them are pulled high by the internal pull - ups and can be used as inputs. As inputs, Port 2 pins that are externally pulled low will source current because of the internal pull - ups. Port 2 emits the high - order address byte during fetches from external program memory and during accesses to external data memory that use 16 - bit addresses (MOVX @DPTR). In this application, it uses strong internal pull - ups emitting 1s. During accesses to external data memory that use 8 - bit addresses (MOVX @Ri), port 2 emits the contents of the P2 SFR. Some Port 2 pins receive the high order address bits during EPROM programming and verification: P2.0 to P2.5 for 16 KB devices P2.0 to P2.6 for 32KB devices
P3.0 - P3.7	10 - 17	11, 13 - 19	5, 7 - 13	I/O	**Port 3:** Port 3 is an 8 - bit bidirectional I/O port with internal pull - ups. Port 3 pins that have 1s written to them are pulled high by the internal pull - ups and can be used as inputs. As inputs, Port 3 pins that are externally pulled low will source current because of the internal pull - ups. Port 3 also serves the special features of the 80C51 family, as listed below.
	10	11	5	I	**RXD (P3.0):** Serial input port
	11	13	7	O	**TXD (P3.1):** Serial output port
	12	14	8	I	**$\overline{\text{INT0}}$ (P3.2):** External interrupt 0
	13	15	9	I	**$\overline{\text{INT1}}$ (P3.3):** External interrupt 1
	14	16	10	I	**T0 (P3.4):** Timer 0 external input
	15	17	11	I	**T1 (P3.5):** Timer 1 external input
	16	18	12	O	**$\overline{\text{WR}}$ (P3.6):** External data memory write strobe
	17	19	13	O	**$\overline{\text{RD}}$ (P3.7):** External data memory read strobe
RST	9	10	4	I/O	**Reset:** A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to V_{SS} permits a power - on reset using only an external capacitor to V_{CC}. This pin is an output when the hardware watchdog forces a system reset.
ALE/$\overline{\text{PROG}}$	30	33	27	O (I)	**Address Latch Enable/Program Pulse:** Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 (1/3 in X2 mode) the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory. This pin is also the program pulse input ($\overline{\text{PROG}}$) during Flash programming. ALE can be disabled by setting SFR's AUXR. 0 bit. With this bit set, ALE will be inactive during internal fetches.

Table 7. CKCON1 Register

CKCON1 - Clock Control Register (AFh)

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SPIX2

Bit Number	Bit Mnemonic	Description
7	-	**Reserved**
6	-	**Reserved**
5	-	**Reserved**
4	-	**Reserved**
3	-	**Reserved**
2	-	**Reserved**
1	-	**Reserved**
0	SPIX2	**SPI** (This control bit is validated when the CPU clock X2 is set; when X2 is low, this bit has no effect). Clear to select 6 clock periods per peripheral clock cycle. Set to select 12 clock periods per peripheral clock cycle.

Reset Value = XXXX XXX0b
Not bit addressable

Registers

Table 10. AUXR Register

AUXR - Auxiliary Register (8Eh)

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

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	M0	**Pulse length** Cleared to stretch MOVX control: the RD/ and the WR/ pulse length is 6 clock periods (default). Set to stretch MOVX control: the RD/ and the WR/ pulse length is 30 clock periods.
4	-	**Reserved** The value read from this bit is indeterminate. Do not set this bit.
3	XRS1	**XRAM Size**
2	XRS0	XRS1 XRS0 XRAM size 0 0 256 bytes (default) 0 1 512 bytes 1 0 768 bytes 1 1 1024 bytes
1	EXTRAM	**EXTRAM bit** Cleared to access internal XRAM using movx @ Ri/ @ DPTR. Set to access external memory. Programmed by hardware after Power-up regarding Hardware Security Byte (HSB), default setting, XRAM selected.
0	AO	**ALE Output bit** Cleared, ALE is emitted at a constant rate of 1/6 the oscillator frequency (or 1/3 if X2 mode is used). (default) Set, ALE is active only during a MOVX or MOVC instruction is used.

Reset Value = XX0X 00'HSB. XRAM'0b (See Table 69)
Not bit addressable

Timer 2

The Timer 2 in the T89C51RB2/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 11) and T2MOD (Table 12) registers. Timer 2 operation is similar to Timer 0 and Timer 1.C/$\overline{T2}$ 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).

Refer to 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 (refer to the Atmel C51 Microcontroller Hardware description). If DCEN bit is set, Timer 2 acts as an Up/down timer/counter as shown in Figure 8. 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.

Table 18. 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 19. 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 20. CL Register

CL - PCA Counter Register Low (0E9h)

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

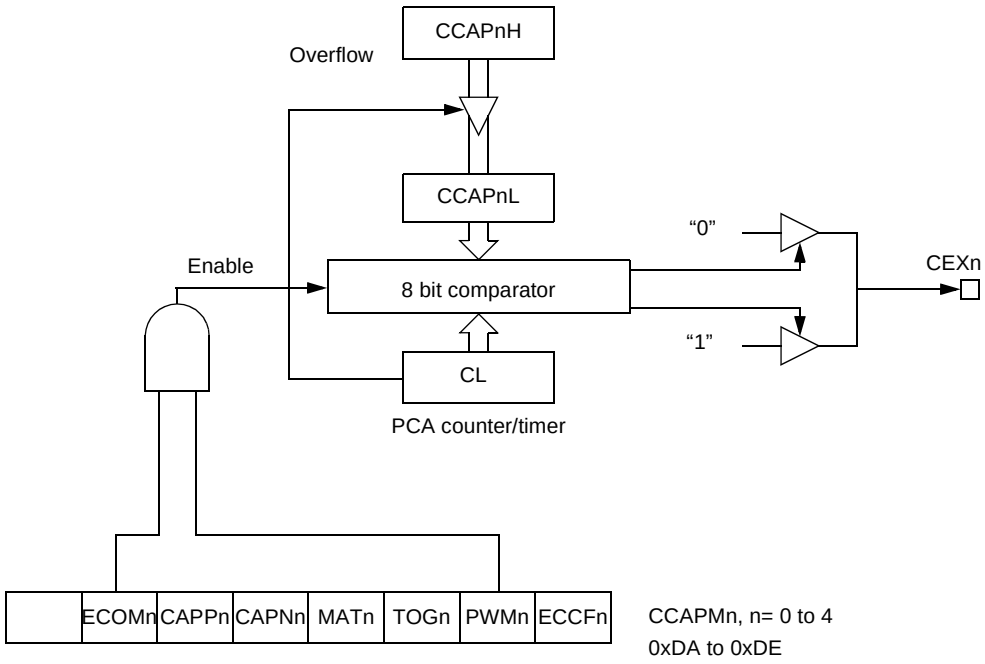
Bit Number	Bit Mnemonic	Description
7-0	-	**PCA Counter** CL Value

Reset Value = 0000 0000b
Not bit addressable

Pulse Width Modulator Mode

All of the PCA modules can be used as PWM outputs. Figure 15 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 15. PCA PWM Mode



PCA Watchdog Timer

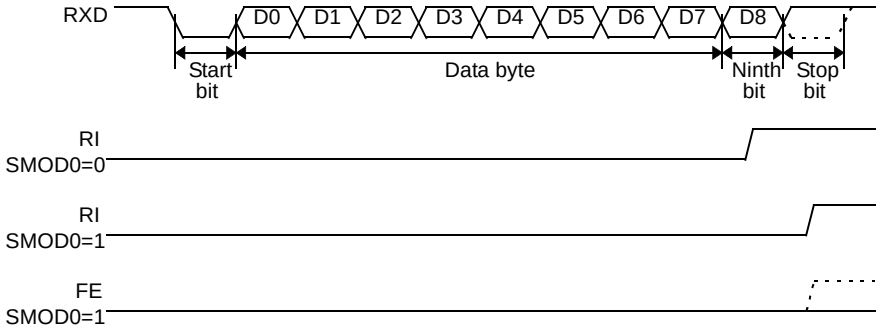
An on-board watchdog timer is available with the PCA to improve the reliability of the system without increasing chip count. Watchdog timers are useful for systems that are susceptible to noise, power glitches, or electrostatic discharge. Module 4 is the only PCA module that can be programmed as a watchdog. However, this module can still be used for other modes if the watchdog is not needed. Figure 13 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:

- 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, or
- 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

Figure 18. UART Timings in Modes 2 and 3



Automatic Address Recognition

The automatic address recognition feature is enabled when the multiprocessor communication feature is enabled (SM2 bit in SCON register is set).

Implemented in hardware, automatic address recognition enhances the multiprocessor communication feature by allowing the serial port to examine the address of each incoming command frame. Only when the serial port recognizes its own address, the receiver sets RI bit in SCON register to generate an interrupt. This ensures that the CPU is not interrupted by command frames addressed to other devices.

If desired, the user may enable the automatic address recognition feature in mode 1.In this configuration, the stop bit takes the place of the ninth data bit. Bit RI is set only when the received command frame address matches the device's address and is terminated by a valid stop bit.

To support automatic address recognition, a device is identified by a given address and a broadcast address.

Note: The multiprocessor communication and automatic address recognition features cannot be enabled in mode 0 (i. e. setting SM2 bit in SCON register in mode 0 has no effect).

Given Address

Each device has an individual address that is specified in SADDR register; the SADEN register is a mask byte that contains don't-care bits (defined by zeros) to form the device's given address. The don't-care bits provide the flexibility to address one or more slaves at a time. The following example illustrates how a given address is formed.

To address a device by its individual address, the SADEN mask byte must be 1111 1111b.

For example:

SADDR0101 0110b
SADEN1111 1100b
Given0101 01XXb

The following is an example of how to use given addresses to address different slaves:

Slave A:SADDR1111 0001b
SADEN1111 1010b
Given1111 0X0Xb

Slave B:SADDR1111 0011b
SADEN1111 1001b
Given1111 0XX1b

Slave C:SADDR1111 0010b
SADEN1111 1101b
Given1111 00X1b

Table 32. 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 for UART** Set to select double baud rate in mode 1, 2 or 3.
6	SMOD0	**Serial port Mode bit 0 for UART** 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 to recognize next reset type. Set by hardware when VCC rises from 0 to its nominal voltage. Can also be set by software.
3	GF1	**General purpose Flag** Cleared by user for general purpose usage. Set by user for general purpose usage.
2	GF0	**General purpose Flag** Cleared by user for general purpose usage. Set by user 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.

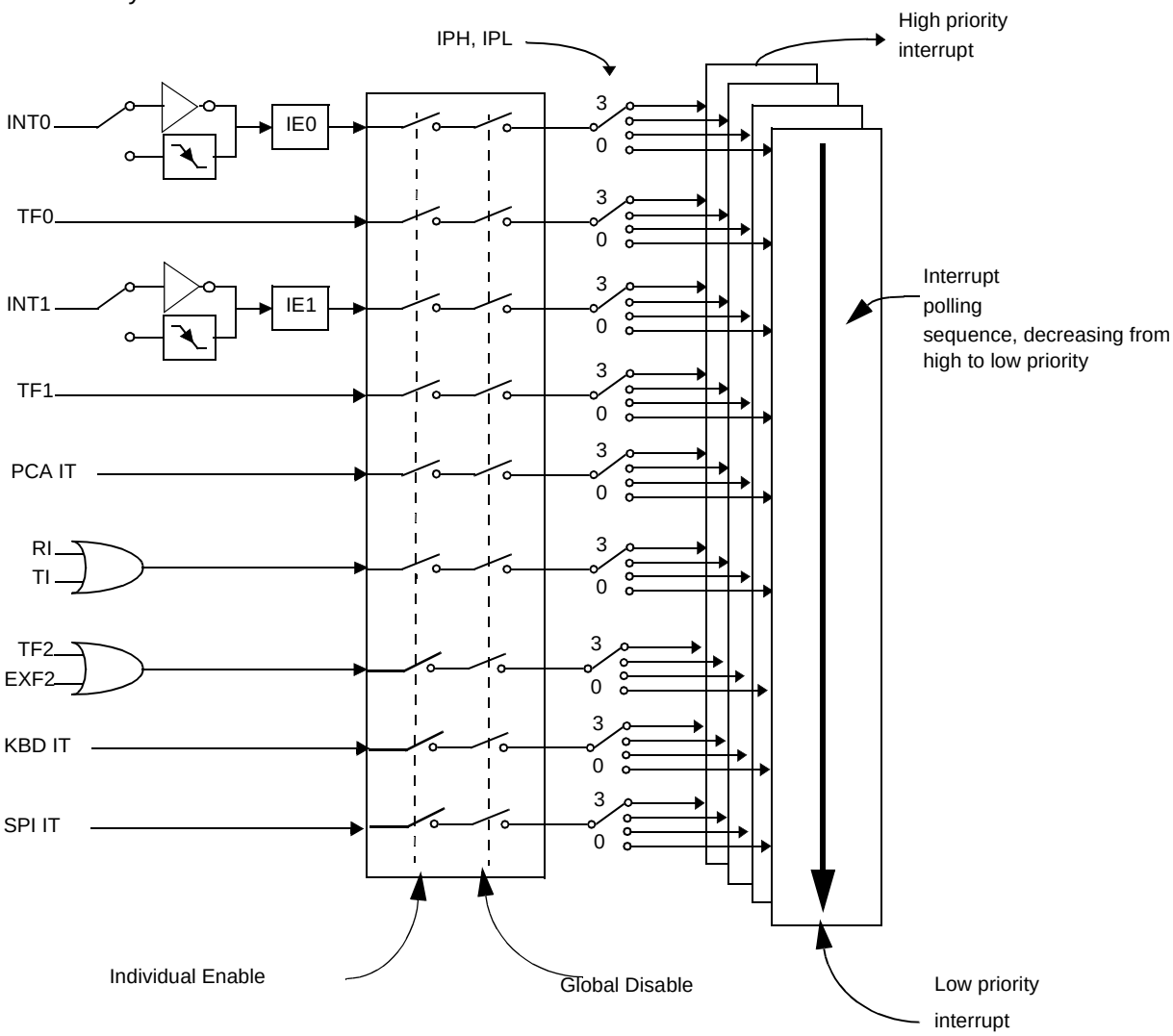
Reset Value = 00X1 0000b
Not bit addressable

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.

Interrupt System

The T89C51RB2/RC2 has a total of 10 interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), three timer interrupts (timers 0, 1 and 2), the serial port interrupt, SPI interrupt, Keyboard interrupt and the PCA global interrupt. These interrupts are shown in Figure 21.

Figure 21. Interrupt Control System



Each of the interrupt sources can be individually enabled or disabled by setting or clearing a bit in the Interrupt Enable register (Table 38 and Table 36). This register also contains a global disable bit, which must be cleared to disable all interrupts at once.

Each interrupt source can also be individually programmed to one out of four priority levels by setting or clearing a bit in the Interrupt Priority register (Table 39) and in the Interrupt Priority High register (Table 37 and Table 38) shows the bit values and priority levels associated with each combination.

Interrupt Sources and Vector Addresses

Table 41. 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

Keyboard Interface

The T89C51RB2/RC2 implements a keyboard interface allowing the connection of a 8 x n matrix keyboard. It is based on 8 inputs with programmable interrupt capability on both high or low level. These inputs are available as alternate function of P1 and allow to exit from idle and power down modes.

The keyboard interface interfaces with the C51 core through 3 special function registers: KBLS, the Keyboard Level Selection register (Table 44), KBE, The Keyboard interrupt Enable register (Table 43), and KBF, the Keyboard Flag register (Table 42).

Interrupt

The keyboard inputs are considered as 8 independent interrupt sources sharing the same interrupt vector. An interrupt enable bit (KBD in IE1) allows global enable or disable of the keyboard interrupt (see Figure 22). As detailed in Figure 23 each keyboard input has the capability to detect a programmable level according to KBLS. x bit value. Level detection is then reported in interrupt flags KBF. x that can be masked by software using KBE. x bits.
This structure allow keyboard arrangement from 1 by n to 8 by n matrix and allow usage of P1 inputs for other purpose.

Figure 22. Keyboard Interface Block Diagram

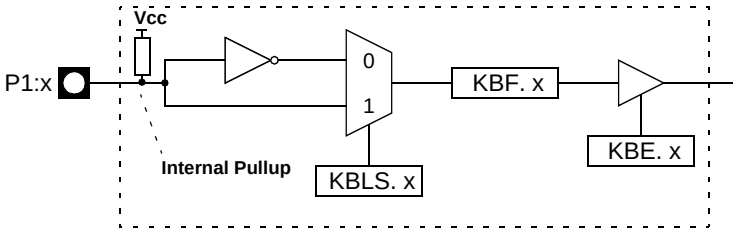
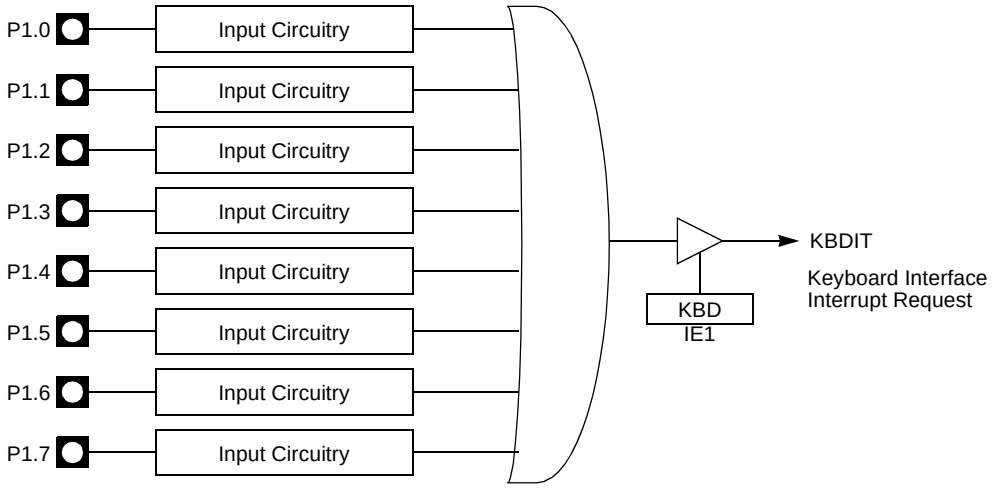


Figure 23. Keyboard Input Circuitry



Power Reduction Mode

P1 inputs allow exit from idle and power down modes as detailed in Section “Power-down Mode”, page 75.

Table 43. KBE Register

KBE-Keyboard Input Enable Register (9Dh)

7	6	5	4	3	2	1	0
KBE7	KBE6	KBE5	KBE4	KBE3	KBE2	KBE1	KBE0

Bit Number	Bit Mnemonic	Description
7	KBE7	**Keyboard line 7 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 7 bit in KBF register to generate an interrupt request.
6	KBE6	**Keyboard line 6 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 6 bit in KBF register to generate an interrupt request.
5	KBE5	**Keyboard line 5 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 5 bit in KBF register to generate an interrupt request.
4	KBE4	**Keyboard line 4 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 4 bit in KBF register to generate an interrupt request.
3	KBE3	**Keyboard line 3 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 3 bit in KBF register to generate an interrupt request.
2	KBE2	**Keyboard line 2 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 2 bit in KBF register to generate an interrupt request.
1	KBE1	**Keyboard line 1 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 1 bit in KBF register to generate an interrupt request.
0	KBE0	**Keyboard line 0 Enable bit** Cleared to enable standard I/O pin. Set to enable KBF. 0 bit in KBF register to generate an interrupt request.

Reset Value= 0000 0000b

Table 44. 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

Table 51. WDTPRG Register

WDTPRG - Watchdog Timer Out Register (0A7h)

7	6	5	4	3	2	1	0
-	-	-	-	-	S2	S1	S0

Bit Number	Bit Mnemonic	Description
7	-	**Reserved** The value read from this bit is undetermined. Do not try to set this bit.
6	-	
5	-	
4	-	
3	-	
2	S2	**WDT Time-out select bit 2**
1	S1	**WDT Time-out select bit 1**
0	S0	**WDT Time-out select bit 0**
		S2 S1 S0 Selected Time-out 0 0 0 (2^{14} - 1) machine cycles, 16. 3 ms @ F_{OSCA}=12 MHz 0 0 1 (2^{15} - 1) machine cycles, 32.7 ms @ F_{OSCA}=12 MHz 0 1 0 (2^{16} - 1) machine cycles, 65. 5 ms @ F_{OSCA}=12 MHz 0 1 1 (2^{17} - 1) machine cycles, 131 ms @ F_{OSCA}=12 MHz 1 0 0 (2^{18} - 1) machine cycles, 262 ms @ F_{OSCA}=12 MHz 1 0 1 (2^{19} - 1) machine cycles, 542 ms @ F_{OSCA}=12 MHz 1 1 0 (2^{20} - 1) machine cycles, 1.05 s @ F_{OSCA}=12 MHz 1 1 1 (2^{21} - 1) machine cycles, 2.09 s @ F_{OSCA}=12 MHz

Reset value XXXX X000

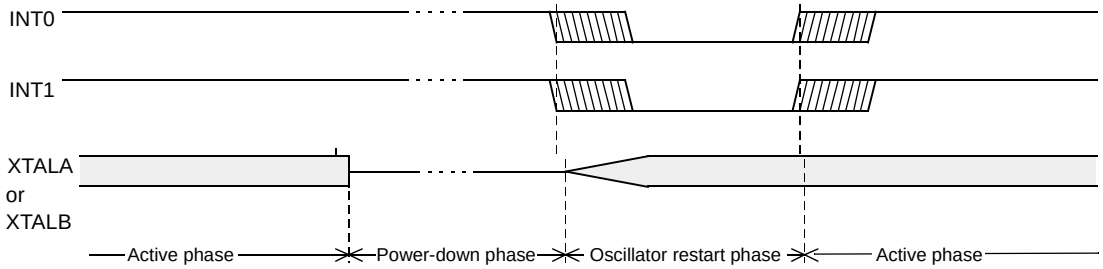
WDT During Power Down and Idle

In Power Down mode the oscillator stops, which means the WDT also stops. While in Power Down mode the user does not need to service the WDT. There are 2 methods of exiting Power Down mode: by a hardware reset or via a level activated external interrupt which is enabled prior to entering Power Down mode. When Power Down is exited with hardware reset, servicing the WDT should occur as it normally should whenever the T89C51RB2/RC2 is reset. Exiting Power Down with an interrupt is significantly different. The interrupt is held low long enough for the oscillator to stabilize. When the interrupt is brought high, the interrupt is serviced. To prevent the WDT from resetting the device while the interrupt pin is held low, the WDT is not started until the interrupt is pulled high. It is suggested that the WDT be reset during the interrupt service routine.

To ensure that the WDT does not overflow within a few states of exiting of powerdown, it is better to reset the WDT just before entering powerdown.

In the Idle mode, the oscillator continues to run. To prevent the WDT from resetting the T89C51RB2/RC2 while in Idle mode, the user should always set up a timer that will periodically exit Idle, service the WDT, and re-enter Idle mode.

Figure 32. Power-down Exit Waveform



Exit from power-down by reset redefines all the SFRs, exit from power-down by external interrupt does no affect the SFRs.

Exit from power-down by either reset or external interrupt does not affect the internal RAM content.

Note: If idle mode is activated with power-down mode (IDL and PD bits set), the exit sequence is unchanged, when execution is vectored to interrupt, PD and IDL bits are cleared and idle mode is not entered.

Table shows the state of ports during idle and power-down modes.

Table 54. State of Ports

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Port Data*	Port Data	Port Data	Port Data
Idle	External	1	1	Floating	Port Data	Address	Port Data
Power Down	Internal	0	0	Port Dat*	Port Data	Port Data	Port Data
Power Down	External	0	0	Floating	Port Data	Port Data	Port Data

* Port 0 can force a 0 level. A "one" will leave port floating.

External Data Memory Characteristics

Table 61. Symbol Description

Symbol	Parameter
T_{RLRH}	$\overline{RD}$ Pulse Width
T_{WLWH}	$\overline{WR}$ Pulse Width
T_{RLDV}	$\overline{RD}$ to Valid Data In
T_{RHDX}	Data Hold After $\overline{RD}$
T_{RHDZ}	Data Float After $\overline{RD}$
T_{LLDV}	ALE to Valid Data In
T_{AVDV}	Address to Valid Data In
T_{LLWL}	ALE to $\overline{WR}$ or $\overline{RD}$
T_{AVWL}	Address to $\overline{WR}$ or $\overline{RD}$
T_{QVWX}	Data Valid to $\overline{WR}$ Transition
T_{QVWH}	Data set-up to $\overline{WR}$ High
T_{WHQX}	Data Hold After $\overline{WR}$
T_{RLAZ}	$\overline{RD}$ Low to Address Float
T_{WHLH}	$\overline{RD}$ or $\overline{WR}$ High to ALE high

Table 62. AC Parameters for a Fix Clock

	-M		-L		
Symbol	Min	Max	Min	Max	Units
T_{RLRH}	125		125		ns
T_{WLWH}	125		125		ns
T_{RLDV}		95		95	ns
T_{RHDX}	0		0		ns
T_{RHDZ}		25		25	ns
T_{LLDV}		155		155	ns
T_{AVDV}		160		160	ns
T_{LLWL}	45	105	45	105	ns
T_{AVWL}	70		70		ns
T_{QVWX}	5		5		ns
T_{QVWH}	155		155		ns
T_{WHQX}	10		10		ns
T_{RLAZ}	0		0		ns
T_{WHLH}	5	45	5	45	ns

In-System Programming (ISP)

The In-System Programming (ISP) is performed without removing the microcontroller from the system. The ISP facility consists of a series of internal hardware resources coupled with internal firmware to facilitate remote programming of theT89C51RB2/RC2 through the serial port.

The Atmel ISP facility has made in-circuit programming in an embedded application possible with a minimum of additional expense in components and circuit board area.

The ISP function through UART uses four pins: TxD, RxD, V_{SS}, V_{CC}. Only a small connector needs to be available to interface the application to an external circuit in order to use this feature.

Using In-System Programming (ISP)

The ISP feature allows a wide range of baud rates in the user application. It is also adaptable to a wide range of oscillator frequencies. This is accomplished by measuring the bit-time of a single bit in a received character. This information is then used to program the baud rate in terms of timer counts based on the oscillator frequency. The ISP feature requires that an initial character (an uppercase U) be sent to the T89C51RB2/RC2 to establish the baud rate. The ISP firmware provides auto-echo of received characters.

Once baud rate initialization has been performed, the ISP firmware will only accept Intel Hex-type records. Intel Hex records consist of ASCII characters used to represent hexadecimal values and are summarized below:

:NNAAAARRDD. DDCC

T89C51RB2/RC2 will accept up to 16 (10h) data bytes. The “AAAA” string represents the address of the first byte in the record. If there are zero bytes in the record, this field is often set to ‘‘0000’’. The “RR” string indicates the record type. A record type of “00” is a data record. A record type of “01” indicates the end-of-file mark. In this application, additional record types will be added to indicate either commands or data for the ISP facility. The “DD” string represents the data bytes. The maximum number of data bytes in a record is limited to 16 (decimal). The “CC” string represents the checksum byte. ISP commands are summarized in Table 74.

As a record is received by the T89C51RB2/RC2, the information in the record is stored internally and a checksum calculation is performed and compared to ‘‘CC’’.

The operation indicated by the record type is not performed until the entire record has been received. Should an error occur in the checksum, the T89C51RB2/RC2 will send an “X” out the serial port indicating a checksum error. If the checksum calculation is found to match the checksum in the record, then the command will be executed. In most cases, successful reception of the record will be indicated by transmitting a “. ” character out the serial port (displaying the contents of the internal program memory is an exception). In the case of a Data Record (record type ‘‘00’’), an additional check is made. A “. ” character will NOT be sent unless the record checksum matched the calculated checksum and all of the bytes in the record were successfully programmed. For a data record, an “X” indicates that the checksum failed to match, and an “R” character indicates that one of the bytes did not properly program.

FLIP, a software utility to implement ISP programming with a PC, is available from the Atmel the web site.

Table 74. Intel-Hex Records Used by In-system Programming

RECORD TYPE	COMMAND/DATA FUNCTION
00	Data Record :nnaaaa00dd. . . . ddcc Where: Nn = number of bytes (hex) in record aaaa = memory address of first byte in record dd. . . . dd = data bytes cc = checksum Example: :05008000AF5F67F060B6
01	End of File (EOF), no operation :xxxxxx01cc Where: xxxxxx = required field, but value is a “don’t care” cc = checksum Example: :00000001FF
02	Specify Oscillator Frequency (Not required, left for Philips compatibility) :01xxxx02ddcc Where: xxxx = required field, but value is a “don’t care” dd = required field, but value is a “don’t care” cc = checksum Example: :0100000210ED

RECORD TYPE	COMMAND/DATA FUNCTION
04	Display Device Data or Blank Check Record type 04 causes the contents of the entire Flash array to be sent out the serial port in a formatted display. This display consists of an address and the contents of 16 bytes starting with that address. No display of the device contents will occur if security bit 2 has been programmed. The dumping of the device data to the serial port is terminated by the reception of any character. General Format of Function 04 :05xxxx04ssssseeeeffcc Where: 05 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 04 = “Display Device Data or Blank Check” function code ssss = starting address eeee = ending address ff = subfunction 00 = display data 01 = blank check cc = checksum Example: :0500000440004FFF0069 (display 4000–4FFF)
05	Miscellaneous Read Functions General Format of Function 05 :02xxxx05ffsscc Where: 02 = number of bytes (hex) in record xxxx = required field, but value is a “don’t care” 05= “Miscellaneous Read” function code ffss = subfunction and selection code 0000 = read copy of the signature byte – manufacturer id (58H) 0001 = read copy of the signature byte – device ID# 1 (Family code) 0002 = read copy of the signature byte – device ID # 2 (Memories size and type) 0003 = read copy of the signature byte – device ID # 3 (Product name and revision) 0700 = read the software security bits 0701 = read BSB 0702 = read SBV 0704 = read HSB cc = checksum Example: :020000050001F0 read copy of the signature byte – device id # 1

In-application Programming Method

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. The API calls are shown in Table .

A set of Philips® compatible API calls is provided.

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.

Table 75. API Calls (Continued)

API Call	Parameter
PROGRAM SOFTWARE SECURITY BIT	Input Parameters: R0 = osc freq (integer Not required, left for Philips compatibility) R1 = 05h DPH = 00h DPL = 00h – security bit # 1 (inhibit writing to Flash) 01h – security bit # 2 (inhibit Flash verify) 10h - allows ISP writing to Flash (see Note 1) 11h - allows ISP Flash verify (see Note 1) Return Parameter none
PROGRAM BOOT STATUS BYTE	Input Parameters: R0 = osc freq (integer Not required, left for Philips compatibility) R1 = 06h DPH = 00h DPL = 00h ACC = status byte Return Parameter ACC = status byte
PROGRAM BOOT VECTOR	Input Parameters: R0 = osc freq (integer Not required, left for Philips compatibility) R1 = 06h DPH = 00h DPL = 01h ACC = boot vector Return Parameter ACC = boot vector
PROGRAM X2 MODE	Input Parameters: R0 = osc freq (integer Not required, left for Philips compatibility) R1 = 0Ah DPH = 00h DPL = 08h ACC = value (00 or 01h) Return Parameter ACC = boot vector
PROGRAM BLJB	Input Parameters: R0 = osc freq (integer Not required, left for Philips compatibility) R1 = 0Ah DPH = 00h DPL = 04h ACC = value (00 or 01h) Return Parameter ACC = boot vector
READ DEVICE DATA	Input Parameters: R1 = 03h DPTR = address of byte to read Return Parameter ACC = value of byte read