

# **USER MANUAL**

## **P89LPC901/902/903**

8-bit microcontrollers with accelerated two-clock 80C51 core  
1KB 3V Low-Power byte-eraseable Flash with 128 Byte RAM

**2003 Dec 8**

**Table of Contents****P89LPC901/902/903****Table of Contents**

1. General Description.....	7
Pin Configurations .....	7
Product comparison.....	8
Pin Descriptions - P89LPC901 .....	12
Pin Descriptions - P89LPC902 .....	14
Pin Descriptions - P89LPC903 .....	15
Special Function Registers .....	16
Special Function Registers Table - P89LPC901 .....	16
Special Function Registers Table - P89LPC902 .....	19
Special Function Registers Table - P89LPC903 .....	22
Memory Organization .....	25
2. Clocks.....	27
Enhanced CPU.....	27
Clock Definitions.....	27
CPU Clock (OSCCLK).....	27
Low Speed Oscillator Option - P89LPC901.....	27
Medium Speed Oscillator Option - P89LPC901 .....	27
High Speed Oscillator Option - P89LPC901.....	27
Oscillator Option Selection- P89LPC901.....	28
Clock Output - P89LPC901 .....	28
On-Chip RC oscillator Option .....	28
Watchdog Oscillator Option .....	29
External Clock Input Option - P89LPC901 .....	29
CPU Clock (CCLK) Wakeup Delay.....	32
CPU Clock (CCLK) Modification: DIVM Register.....	32
Low Power Select (P89LPC901) .....	33
3. Interrupts .....	35
Interrupt Priority Structure.....	35
External Interrupt Inputs .....	37
External Interrupt Pin Glitch Suppression.....	37
4. I/O Ports .....	39
Port Configurations.....	39
Quasi-Bidirectional Output Configuration .....	39
Open Drain Output Configuration .....	40
Input-Only Configuration.....	41
Push-Pull Output Configuration .....	41
Port 0 Analog Functions .....	41
Additional Port Features .....	42
5. Timers 0 and 1.....	45
Mode 0.....	46
Mode 2.....	46
Mode 3.....	47

**Table of Contents****P89LPC901/902/903**

Mode 6 - P89LPC901 .....	47
Timer Overflow toggle output - P89LPC901 .....	49
6. Real-Time Clock/System Timer .....	51
Real-time Clock Source .....	51
Changing RTCS1-0 .....	53
Real-time Clock Interrupt/Wake Up .....	53
Reset Sources Affecting the Real-time Clock.....	53
7. Power Monitoring Functions .....	55
Brownout Detection .....	55
Power-On Detection .....	56
Power Reduction Modes.....	57
8. UART (P89LPC903) .....	61
Mode 0.....	61
Mode 1.....	61
Mode 2.....	61
Mode 3.....	61
SFR Space .....	62
Baud Rate Generator and Selection.....	62
Updating the BRGR1 and BRGR0 SFRs.....	62
Framing Error.....	63
Break Detect.....	63
More About UART Mode 0 .....	65
More About UART Mode 1 .....	66
More About UART Modes 2 and 3.....	67
Framing Error and RI in Modes 2 and 3 with SM2 = 1 .....	67
Break Detect.....	67
Double Buffering.....	68
Double Buffering in Different Modes.....	68
Transmit Interrupts with Double Buffering Enabled (Modes 1, 2 and 3)..	68
The 9th Bit (Bit 8) in Double Buffering (Modes 1, 2 and 3).....	69
Multiprocessor Communications.....	70
Automatic Address Recognition.....	70
9. Reset .....	73
Power-On reset code execution .....	73
10. Analog Comparators.....	75
Comparator Configuration .....	75
Internal Reference Voltage.....	77
Comparator Interrupt .....	77
Comparator and Power Reduction Modes.....	78
Comparator Configuration Example .....	78
11. Keypad Interrupt (KBI).....	79
12. Watchdog Timer .....	83
Watchdog Function.....	83

**Table of Contents****P89LPC901/902/903**

Feed Sequence .....	84
Watchdog Timer in Timer Mode .....	87
Power down operation .....	87
Watchdog Clock Source .....	87
Periodic wakeup from Power down without an external oscillator .....	89
13. Additional Features .....	91
Software Reset .....	91
Dual Data Pointers .....	91
14. Flash program memory .....	93
General description .....	93
Features .....	93
Introduction to IAP-Lite .....	93
Using Flash as data storage .....	93
Accessing additional flash elements .....	96
Erase-programming additional flash elements .....	97
Reading additional flash elements .....	97
User Configuration Bytes .....	99
User Security Bytes .....	101
Boot Vector .....	102
Boot Status .....	102
15. Instruction set .....	103
16. Revision History .....	107
17. Index .....	109

**List of Figures****P89LPC901/902/903****List of Figures**

P89LPC901/902/903 Memory Map . . . . .	25
Using the Crystal Oscillator - P89LPC901 . . . . .	28
On-Chip RC Oscillator TRIM Register. . . . .	29
Block Diagram of Oscillator Control - P89LPC901 . . . . .	30
Block Diagram of Oscillator Control - P89LPC902 . . . . .	31
Block Diagram of Oscillator Control- P89LPC903. . . . .	32
Interrupt priority level. . . . .	35
Summary of Interrupts - P89LPC901 . . . . .	36
Summary of Interrupts - P89LPC903 . . . . .	36
Summary of Interrupts - P89LPC902 . . . . .	36
Interrupt sources, enables, and Power down Wake-up sources - P89LPC901 . . . . .	37
Interrupt sources, enables, and Power down Wake-up sources - P89LPC902 . . . . .	38
Interrupt sources, enables, and Power down Wake-up sources - P89LPC903 . . . . .	38
Number of I/O Pins Available . . . . .	39
Port Output Configuration Settings. . . . .	39
Quasi-Bidirectional Output . . . . .	40
Open Drain Output . . . . .	40
Input Only . . . . .	41
Push-Pull Output . . . . .	41
Port Output Configuration - P89LPC901. . . . .	42
Port Output Configuration - P89LPC902. . . . .	42
Port Output Configuration - P89LPC903. . . . .	42
Timer/Counter Mode Control register (TMOD) . . . . .	45
Timer/Counter Auxiliary Mode Control register (TAMOD). . . . .	46
Timer/Counter Control register (TCON) . . . . .	47
Timer/Counter 0 or 1 in Mode 0 (13-bit counter). . . . .	48
Timer/Counter 0 or 1 in Mode 1 (16-bit counter). . . . .	48
Timer/Counter 0 or 1 in Mode 2 (8-bit auto-reload). . . . .	48
Timer/Counter 0 Mode 3 (two 8-bit counters) . . . . .	49
Timer/Counter 0 in Mode 6 (PWM auto-reload), P89LPC901. . . . .	49
Real-time clock/system timer Block Diagram . . . . .	51
Real-time Clock/System Timer Clock Source - P89LPC901 . . . . .	52
:Real-time Clock/System Timer Clock Source - P89LPC902/903 . . . . .	53
RTCCON Register. . . . .	54
Brownout Options. . . . .	56
Power Reduction Modes. . . . .	57
Power Control Register (PCON) . . . . .	58
Power Control Register (PCONA). . . . .	59
SFR Locations for UARTs. . . . .	62
Baud Rate Generation for UART. . . . .	62
BRGCON Register . . . . .	63
Baud Rate Generations for UART (Modes 1, 3) . . . . .	63
Serial Port Control Register (SCON). . . . .	64

**List of Figures****P89LPC901/902/903**

Serial Port Status Register (SSTAT) . . . . .	65
Serial Port Mode 0 (Double Buffering Must Be Disabled) . . . . .	66
Serial Port Mode 1 (Only Single Transmit Buffering Case Is Shown) . . . . .	66
Serial Port Mode 2 or 3 (Only Single Transmit Buffering Case Is Shown) . . . . .	67
FE and RI when SM2 = 1 in Modes 2 and 3. . . . .	67
Transmission with and without Double Buffering . . . . .	69
Block Diagram of Reset. . . . .	73
Reset Sources Register . . . . .	74
Comparator Control Registers (CMP1 and CMP2) . . . . .	75
Comparator Input and Output Connections - P89LPC901 . . . . .	76
Comparator Input and Output Connections - P89LPC902 . . . . .	76
Comparator Input and Output Connections - P89LPC903 . . . . .	77
Comparator Configurations . . . . .	77
Keypad Pattern Register- P89LPC901 . . . . .	79
Keypad Pattern Register - P89LPC902. . . . .	79
Keypad Pattern Register - P89LPC903. . . . .	80
Keypad Control Register. . . . .	80
Keypad Interrupt Mask Register (KBM) - P89LPC901 . . . . .	80
Keypad Interrupt Mask Register (KBM)) - P89LPC902. . . . .	81
Keypad Interrupt Mask Register (KBM)) - P89LPC903. . . . .	81
Watchdog timer configuration . . . . .	83
Watchdog Prescaler . . . . .	84
Watchdog Timer Control Register. . . . .	85
P89LPC901/902/903 Watchdog Timeout Values . . . . .	86
Watchdog Timer in Watchdog Mode (WDTE = 1). . . . .	86
Watchdog Timer in Timer Mode (WDTE = 0) . . . . .	87
AUXR1 Register . . . . .	91
Flash Memory Control Register. . . . .	95
Assembly language routine to erase/program all or part of a page. . . . .	95
C-language routine to erase/program all or part of a page . . . . .	96
Flash elements accesable through IAP-Lite . . . . .	97
Assembly language routine to erase/program a flash element . . . . .	98
C-language routine to erase/program a flash element . . . . .	98
C-language routine to read a flash element . . . . .	99
Flash User Configuration Byte 1 (UCFG1) . . . . .	100
User Sector Security Bytes (SEC0 ... SEC3) . . . . .	101
Effects of Security Bits . . . . .	101
Boot Vector (BOOTVEC). . . . .	102
Boot Status (BOOTSTAT). . . . .	102
Instruction set summary . . . . .	103

# GENERAL DESCRIPTION

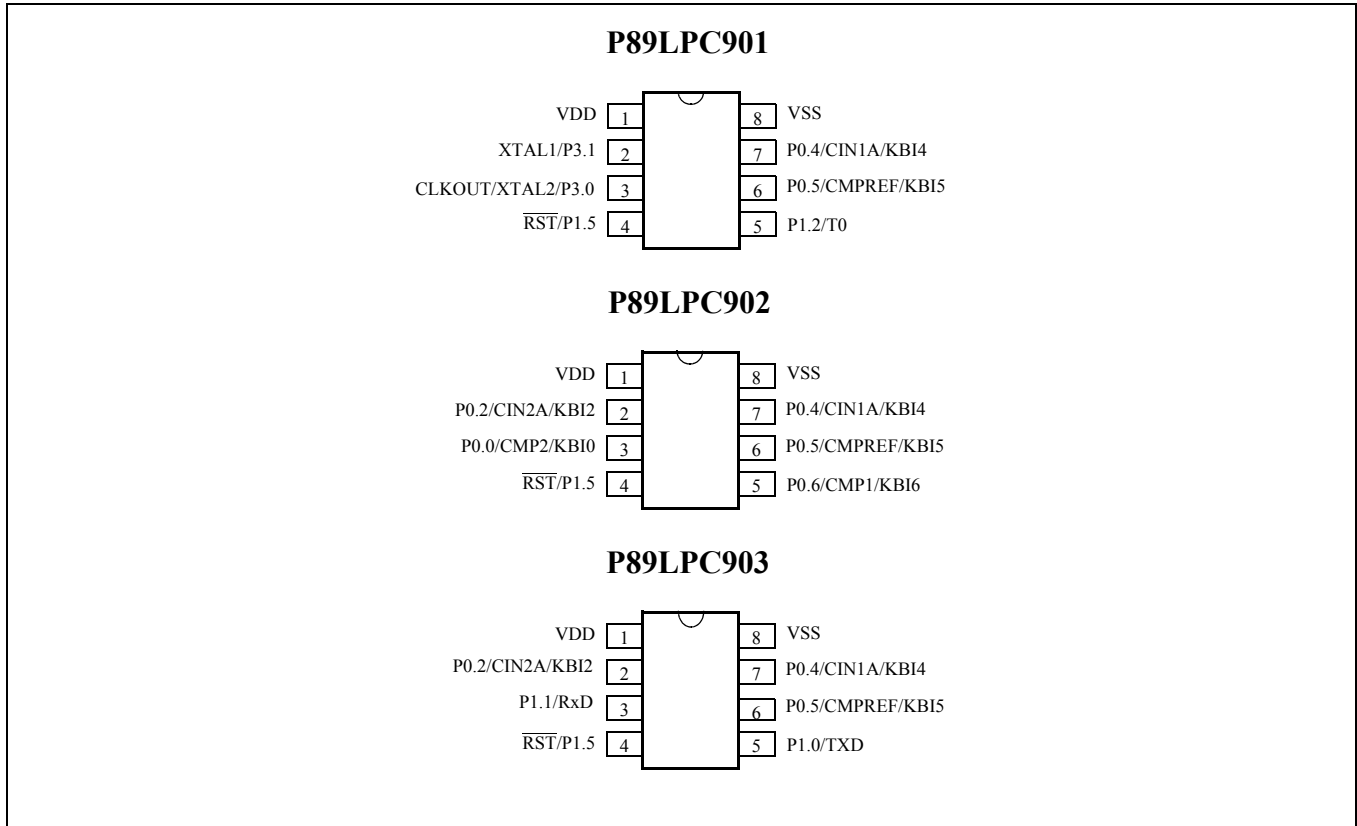
## P89LPC901/902/903

### 1. GENERAL DESCRIPTION

The P89LPC901/902/903 is a single-chip microcontroller designed for applications demanding high-integration, low cost solutions over a wide range of performance requirements. The P89LPC901/902/903 is based on a high performance processor architecture that executes instructions six times the rate of standard 80C51 devices. Many system level functions have been incorporated into the P89LPC901/902/903 in order to reduce component count, board space, and system cost.

#### Pin Configurations

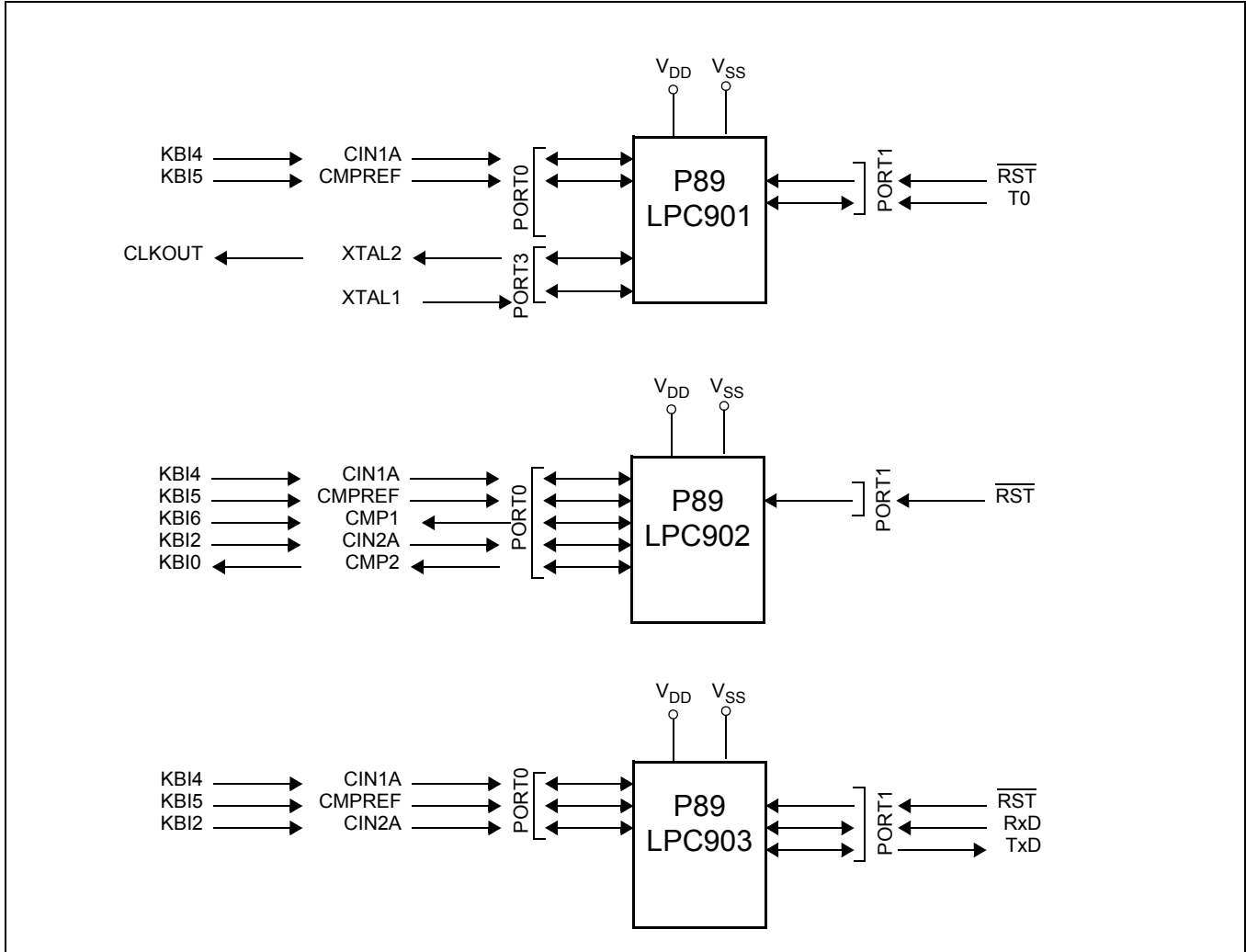
##### 8-Pin Packages



# GENERAL DESCRIPTION

## P89LPC901/902/903

### Logic Symbols



### Product comparison

The following table highlights differences between these three devices.

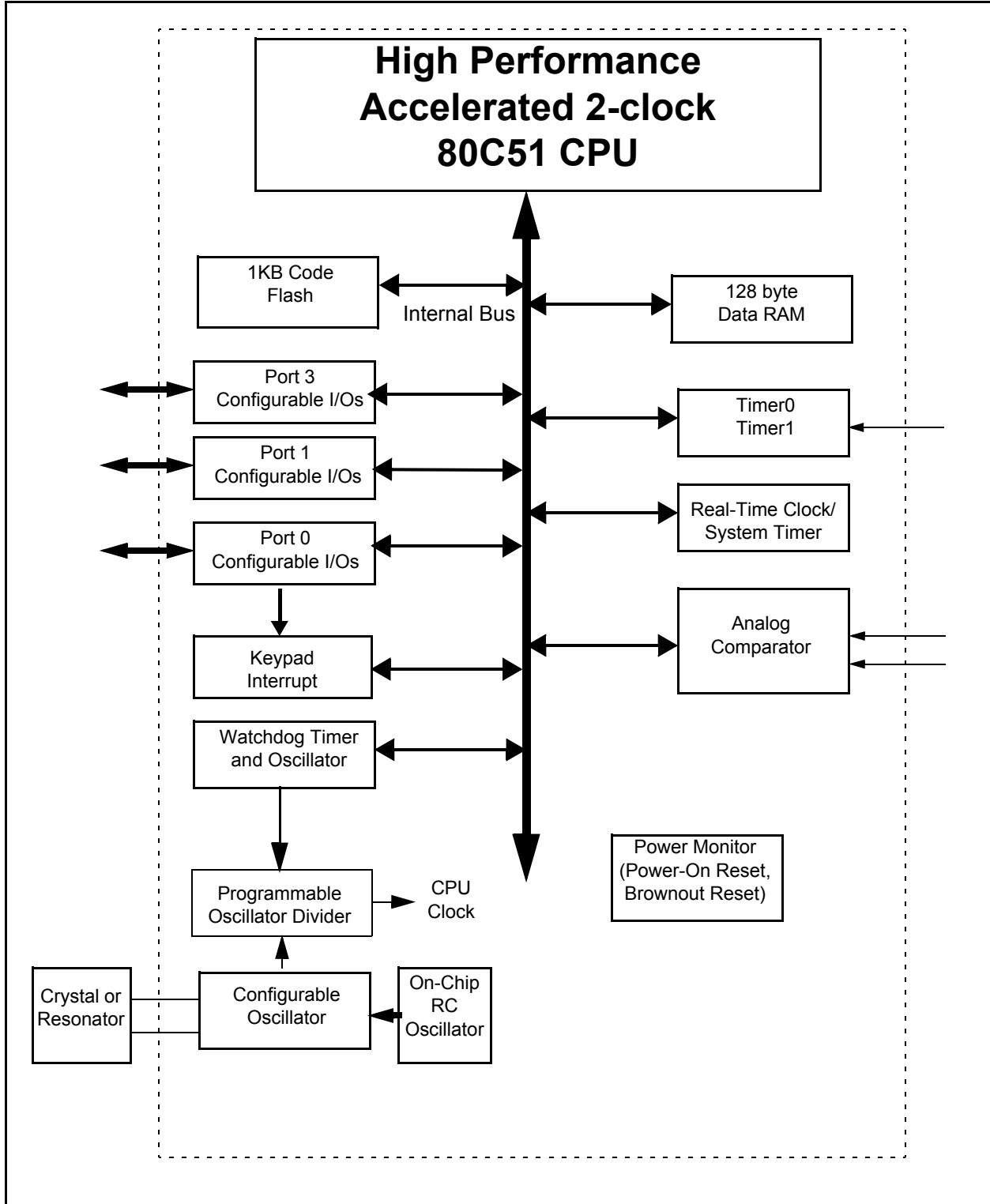
Part number	Ext crystal pins	X2 clkout	T0 PWM output	CMP1 input	CMP2 input	CMP Ref Input	CMP1 & CMP2 outputs	UART	
								TxD	RxD
P89LPC901	Y	Y	Y	Y		Y			
P89LPC902				Y	Y	Y	Y		
P89LPC903				Y	Y	Y		Y	Y



# GENERAL DESCRIPTION

P89LPC901/902/903

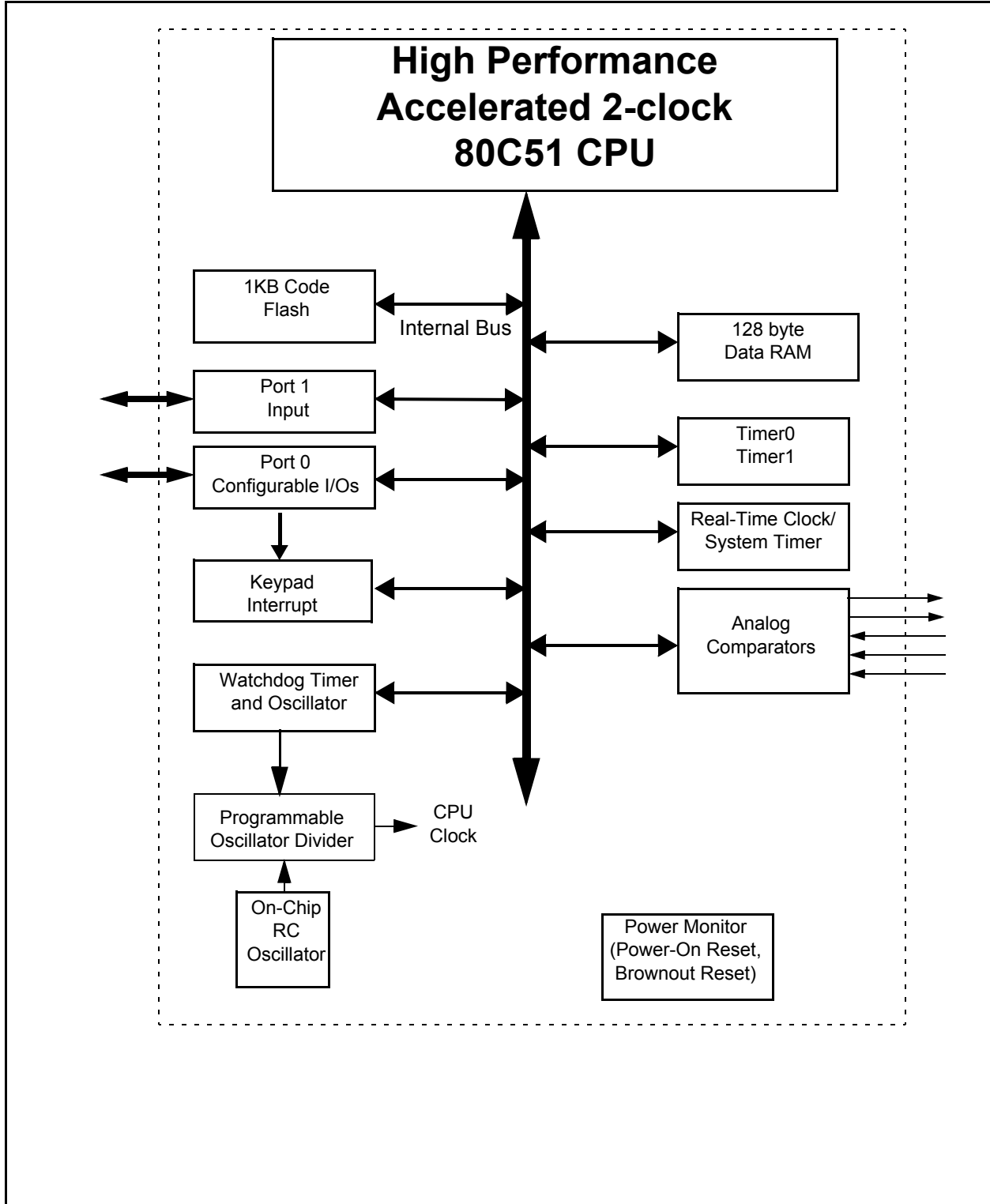
Block Diagram - P89LPC901



# GENERAL DESCRIPTION

P89LPC901/902/903

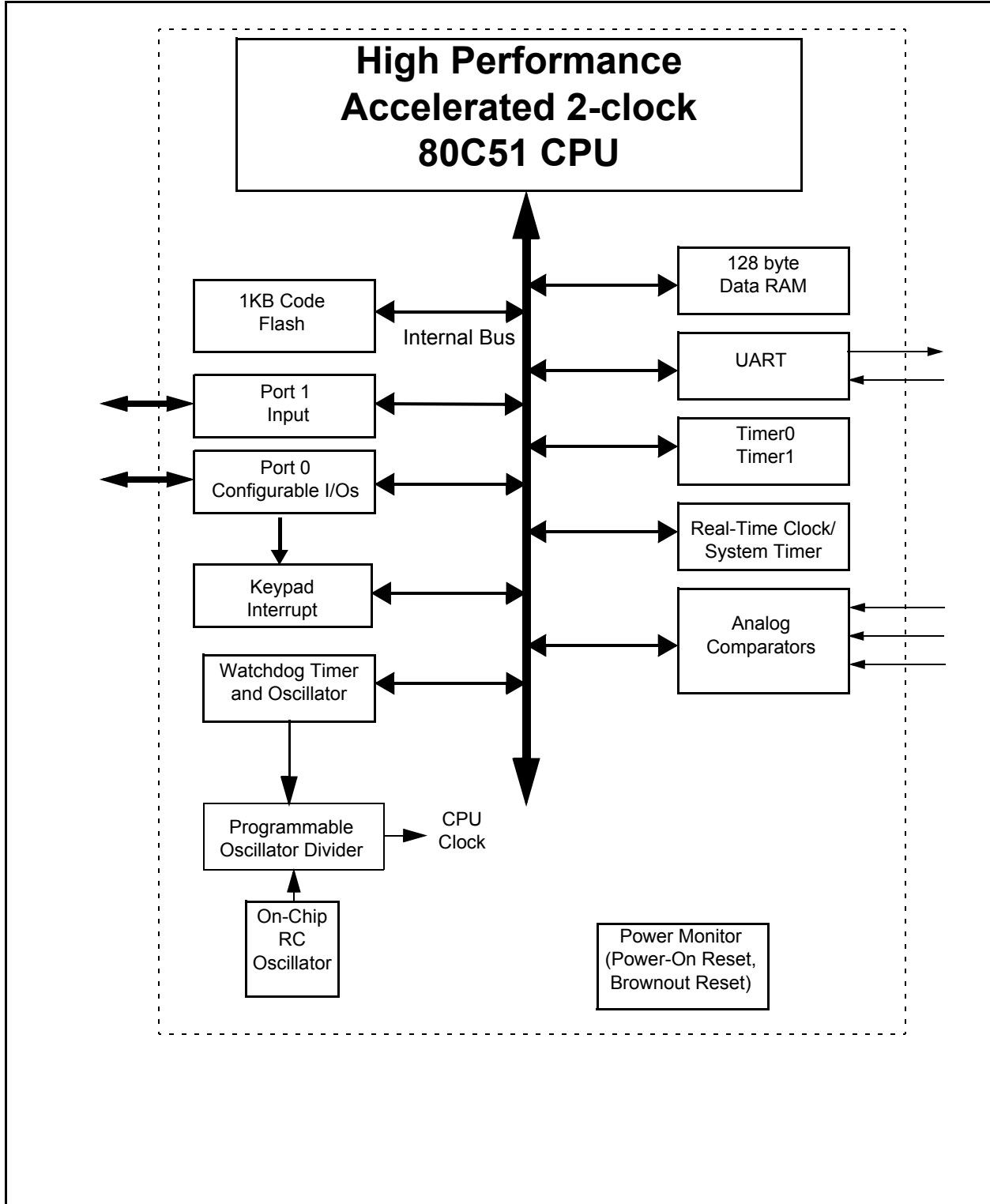
Block Diagram - P89LPC902



# GENERAL DESCRIPTION

P89LPC901/902/903

Block Diagram - P89LPC903



# GENERAL DESCRIPTION

**P89LPC901/902/903**

## PIN DESCRIPTIONS - P89LPC901

MNEMONIC	PIN NO.	TYPE	NAME AND FUNCTION
P0.0 - P0.6	6, 7	I/O	<p><b>Port 0:</b> Port 0 is an I/O port with a user-configurable output types. During reset Port 0 latches are configured in the input only mode with the internal pullup disabled. The operation of port 0 pins as inputs and outputs depends upon the port configuration selected. Each port pin is configured independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics in the Data Sheet for details.</p> <p>The Keypad Interrupt feature operates with port 0 pins.</p> <p>All pins have Schmitt triggered inputs.</p> <p>Port 0 also provides various special functions as described below.</p>
	7	I/O I I	<p><b>P0.4</b> Port 0 bit 4.</p> <p><b>CIN1A</b> Comparator positive input.</p> <p><b>KBI4</b> Keyboard Input 4.</p>
	6	I/O I I	<p><b>P0.5</b> Port 0 bit 5.</p> <p><b>CMPREF</b> Comparator reference (negative) input.</p> <p><b>KBI5</b> Keyboard Input 5.</p>
P1.0 - P1.5	4,5		<p><b>Port 1:</b> Port 1 is an I/O port with a user-configurable output types. During reset Port 1 latches are configured in the input only mode with the internal pull-up disabled. The operation of the configurable port 1 pins as inputs and outputs depends upon the port configuration selected. Each of the configurable port pins are programmed independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics in the Data Sheet for details. P1.5 is input only.</p> <p>All pins have Schmitt triggered inputs.</p> <p>Port 1 also provides various special functions as described below.</p>
	5	I/O I/O	<p><b>P1.2</b> Port 1 bit 2.</p> <p><b>T0</b> Timer/counter 0 external count input or overflow output.</p>
	4	I I	<p><b>P1.5</b> Port 1 bit 5. (Input only)</p> <p><b>RST</b> External Reset input during power-on or if selected via UCFG1. When functioning as a reset input a low on this pin resets the microcontroller, causing I/O ports and peripherals to take on their default states, and the processor begins execution at address 0. Also used during a power-on sequence to force In-Circuit Programming mode.</p>
P3.0 - P3.1	2, 3	I/O	<p><b>Port 3</b> Port 3 is an I/O port with a user-configurable output types. During reset Port 3 latches are configured in the input only mode with the internal pullup disabled. The operation of port 3 pins as inputs and outputs depends upon the port configuration selected. Each port pin is configured independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics in the Data Sheet for details.</p> <p>All pins have Schmitt triggered inputs.</p> <p>Port 3 also provides various special functions as described below:</p>
	3	I/O	<p><b>P3.0</b> Port 3 bit 0.</p>

# GENERAL DESCRIPTION

**P89LPC901/902/903**

MNEMONIC	PIN NO.	TYPE	NAME AND FUNCTION
		O	<b>XTAL2</b> Output from the oscillator amplifier (when a crystal oscillator option is selected via the FLASH configuration).
		O	<b>CLKOUT</b> CPU clock divided by 2 when enabled via SFR bit (ENCLK - TRIM.6). It can be used if the CPU clock is the internal RC oscillator, watchdog oscillator or external clock input, except when XTAL1/XTAL2 are used to generate clock source for the Real-Time clock/system timer.
	2	I/O I	<b>P3.1</b> Port 3 bit 1. <b>XTAL1</b> Input to the oscillator circuit and internal clock generator circuits (when selected via the FLASH configuration). It can be a port pin if internal RC oscillator or watchdog oscillator is used as the CPU clock source, AND if XTAL1/XTAL2 are not used to generate the clock for the Real-Time clock/system timer.
V <sub>SS</sub>	8	I	<b>Ground:</b> 0V reference.
V <sub>DD</sub>	1	I	<b>Power Supply:</b> This is the power supply voltage for normal operation as well as Idle and Power down modes.

# GENERAL DESCRIPTION

**P89LPC901/902/903**

## PIN DESCRIPTIONS - P89LPC902

MNEMONIC	PIN NO.	TYPE	NAME AND FUNCTION
P0.0 - P0.6	2,3,5,6,7	I/O	<p><b>Port 0:</b> Port 0 is an I/O port with a user-configurable output types. During reset Port 0 latches are configured in the input only mode with the internal pullup disabled. The operation of port 0 pins as inputs and outputs depends upon the port configuration selected. Each port pin is configured independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics for details.</p> <p>The Keypad Interrupt feature operates with port 0 pins. All pins have Schmitt triggered inputs. Port 0 also provides various special functions as described below.</p>
	3	I/O 0 I	<p><b>P0.0</b> Port 0 bit 0. <b>CMP2</b> Comparator 2 output. <b>KBI0</b> Keyboard Input 0.</p>
	2	I/O I I	<p><b>P0.2</b> Port 0 bit 2. <b>CIN2A</b> Comparator 2 positive input. <b>KBI2</b> Keyboard Input 2.</p>
	7	I/O I I	<p><b>P0.4</b> Port 0 bit 4. <b>CIN1A</b> Comparator 1 positive input. <b>KBI4</b> Keyboard Input 4.</p>
	6	I/O I I	<p><b>P0.5</b> Port 0 bit 5. <b>CMPREF</b> Comparator reference (negative) input. <b>KBI5</b> Keyboard Input 5.</p>
	5	I/O O I	<p><b>P0.6</b> Port 0 bit 6. <b>CMP1</b> Comparator 1 output. <b>KBI6</b> Keyboard Input 6.</p>
P1.5	4	I	<p><b>Port 1:</b> Port 1 is a single-bit input-only port. I/O port without a pull-up. This pin has a Schmitt trigger. Port 1 also provides the special function described below.</p>
	4	I I	<p><b>P1.5</b> Port 1 bit 5. (Input only) <b>RST</b> External Reset input during power-on or if selected via UCFG1. When functioning as a reset input a low on this pin resets the microcontroller, causing I/O ports and peripherals to take on their default states, and the processor begins execution at address 0. Also used during a power-on sequence to force In-System Programming mode.</p>
V <sub>SS</sub>	8	I	<b>Ground:</b> 0V reference.
V <sub>DD</sub>	1	I	<b>Power Supply:</b> This is the power supply voltage for normal operation as well as Idle and Power down modes.

# GENERAL DESCRIPTION

**P89LPC901/902/903**

## PIN DESCRIPTIONS - P89LPC903

MNEMONIC	PIN NO.	TYPE	NAME AND FUNCTION
P0.0 - P0.6	2,6,7	I/O	<p><b>Port 0:</b> Port 0 is an I/O port with a user-configurable output types. During reset Port 0 latches are configured in the input only mode with the internal pullup disabled. The operation of port 0 pins as inputs and outputs depends upon the port configuration selected. Each port pin is configured independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics in the Data Sheet for details.</p> <p>The Keypad Interrupt feature operates with port 0 pins.</p> <p>All pins have Schmitt triggered inputs.</p> <p>Port 0 also provides various special functions as described below.</p>
	2	I/O I I	<p><b>P0.2</b> Port 0 bit 2.</p> <p><b>CIN2A</b> Comparator 2 positive input.</p> <p><b>KBI2</b> Keyboard Input 2.</p>
	7	I/O I I	<p><b>P0.4</b> Port 0 bit 4.</p> <p><b>CIN1A</b> Comparator 1 positive input.</p> <p><b>KBI4</b> Keyboard Input 4.</p>
	6	I/O I I	<p><b>P0.5</b> Port 0 bit 5.</p> <p><b>CMPREF</b> Comparator reference (negative) input.</p> <p><b>KBI5</b> Keyboard Input 5.</p>
P1.0 - P1.5	3,4,5		<p><b>Port 1:</b> Port 1 is an I/O port with a user-configurable output types. During reset Port 1 latches are configured in the input only mode with the internal pull-up disabled. The operation of the configurable port 1 pins as inputs and outputs depends upon the port configuration selected. Each of the configurable port pins are programmed independently. Refer to the section on I/O port configuration and the DC Electrical Characteristics in the Data Sheet for details. P1.5 is input only.</p> <p>All pins have Schmitt triggered inputs.</p> <p>Port 1 also provides various special functions as described below.</p>
	5	I/O O	<p><b>P1.0</b> Port 1 bit 0.</p> <p><b>TxD</b> Serial port transmitter data.</p>
	3	I/O I	<p><b>P1.1</b> Port 1 bit 1.</p> <p><b>RxD</b> Serial port receiver data.</p>
	4	I I	<p><b>P1.5</b> Port 1 bit 5. (Input only)</p> <p><b>RST</b> External Reset input during power-on or if selected via UCFG1. When functioning as a reset input a low on this pin resets the microcontroller, causing I/O ports and peripherals to take on their default states, and the processor begins execution at address 0. Also used during a power-on sequence to force In-Circuit Programming mode.</p>
V <sub>SS</sub>	8	I	<b>Ground:</b> 0V reference.
V <sub>DD</sub>	1	I	<b>Power Supply:</b> This is the power supply voltage for normal operation as well as Idle and Power down modes.

# GENERAL DESCRIPTION

## P89LPC901/902/903

### SPECIAL FUNCTION REGISTERS

**Note: Special Function Registers (SFRs) accesses are restricted in the following ways:**

1. User must NOT attempt to access any SFR locations not defined.
2. Accesses to any defined SFR locations must be strictly for the functions for the SFRs.
3. SFR bits labeled '-', '0' or '1' can ONLY be written and read as follows:
  - '-' Unless otherwise specified, MUST be written with '0', but can return any value when read (even if it was written with '0'). It is a reserved bit and may be used in future derivatives.
  - '0' MUST be written with '0', and will return a '0' when read.
  - '1' MUST be written with '1', and will return a '1' when read.

### Special Function Registers Table - P89LPC901

Name	Description	SFR Address	Bit Functions and Addresses								Reset Value	
			MSB				LSB				Hex	Binary
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H	00000000
AUXR1#	Auxiliary Function Register	A2H	CLKLP	-	-	ENT0	SRST	0	-	DPS	00H <sup>1</sup>	000000x0
B*	B Register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H	00000000
CMP1#	Comparator 1 Control Register	ACH	-	-	CE1	-	CN1	-	CO1	CMF1	00H <sup>1</sup>	xx000000
DIVM#	CPU Clock Divide-by-M Control	95H									00H	00000000
DPTR	Data Pointer (2 bytes)											
DPH	Data Pointer High	83H									00H	00000000
DPL	Data Pointer Low	82H									00H	00000000
FMADR#	Program Flash Address High	E7H									00H	00000000
FMADRL#	Program Flash Address Low	E6H									00H	00000000
FMCON#	Program Flash Control (Read)	E4H	BUSY	-	-	-	HVA	HVE	SV	OI	70H	01110000
	Program Flash Control (Write)		FMCMD.	FMCMD.	FMCMD.	FMCMD.	FMCMD.	FMCMD.	FMCMD.	FMCMD.		
			7	6	5	4	3	2	1	0		
FMDATA#	Program Flash Data	E5H									00H	00000000
IEN0*	Interrupt Enable 0	A8H	AF	AE	AD	AC	AB	AA	A9	A8	00H	00000000
			EA	EWDRT	EBO	-	ET1	-	ET0	-		
IEN1*#	Interrupt Enable 1	E8H	EF	EE	ED	EC	EB	EA	E9	E8	00H <sup>1</sup>	00x000000
			-	-	-	-	-	EC	EKBI	-		
IP0*	Interrupt Priority 0	B8H	BF	BE	BD	BC	BB	BA	B9	B8	00H <sup>1</sup>	x00000000
			-	PWDRT	PBO	-	PT1	-	PT0	-		



# GENERAL DESCRIPTION

## P89LPC901/902/903

Name	Description	SFR Address	Bit Functions and Addresses								Reset Value	
			MSB				LSB				Hex	Binary
IP0H#	Interrupt Priority 0 High	B7H	-	PWDRT H	PBOH	-	PT1H	-	PT0H	-	00H <sup>1</sup>	x0000000
IP1*#	Interrupt Priority 1	F8H	FF	FE	FD	FC	FB	FA	F9	F8	00H <sup>1</sup>	00x000000
IP1H#	Interrupt Priority 1 High	F7H	-	-	-	-	-	PCH	PKBIH	-	00H <sup>1</sup>	00x000000
KBCON#	Keypad Control Register	94H	-	-	-	-	-	PATN_S EL	KBIF	-	00H <sup>1</sup>	xxxxxx00
KBMASK#	Keypad Interrupt Mask Register	86H									00H	00000000
KBPATN#	Keypad Pattern Register	93H									FFH	11111111
P0*	Port 0	80H	87	86	85	84	83	82	81	80	Note 1	
P1*	Port 1	90H	-	-	CMPREF/ KB5	CIN1A/ KB4	-	-	-	-		
P3*	Port 3	B0H	97	96	95	94	93	92	91	90		
P0M1#	Port 0 Output Mode 1	84H	-	-	RST	-	-	T0	-	-		
P0M2#	Port 0 Output Mode 2	85H	B7	B6	B5	B4	B3	B2	B1	B0		
P1M1#	Port 1 Output Mode 1	91H	-	-	-	-	-	-	XTAL1	XTAL2	Note 1	
P1M2#	Port 1 Output Mode 2	92H	-	-	(P0M1.5)	(P0M1.4)	-	-	-	-	FFH	11111111
P3M1#	Port 3 Output Mode 1	B1H	-	-	(P0M2.5)	(P0M2.4)	-	-	-	-	00H	00000000
P3M2#	Port 3 Output Mode 2	B2H	-	-	(P1M1.5)	-	-	(P1M1.2)	-	-	FFH <sup>1</sup>	11111111
PCON#	Power Control Register	87H	-	-	(P1M2.5)	-	-	(P1M2.2)	-	-	00H <sup>1</sup>	00000000
PCONA#	Power Control Register A	B5H	-	-	-	-	-	-	(P3M1.1)	(P3M1.0)	03H <sup>1</sup>	xxxxxx11
PSW*	Program Status Word	D0H	-	-	-	-	-	-	(P3M2.1)	(P3M2.0)	00H <sup>1</sup>	xxxxxx00
PT0AD#	Port 0 Digital Input Disable	F6H	-	-	BOPD	BOI	GF1	GF0	PMOD1	PMOD0	00H	00000000
RSTSRC#	Reset Source Register	DFH	RTCPD	-	VCPD	-	-	-	-	-	00H <sup>1</sup>	00000000
RTCCON#	Real-Time Clock Control	D1H	D7	D6	D5	D4	D3	D2	D1	D0		
RTCH#	Real-Time Clock Register High	D2H	CY	AC	F0	RS1	RS0	OV	F1	P	00H	00000000
RTCL#	Real-Time Clock Register Low	D3H	-	-	PT0AD.5	PT0AD.4	-	-	-	-	00H	xx00000x
SP	Stack Pointer	81H	-	-	BOF	POF	-	R_WD	R_SF	R_EX	Note 2	
TAMOD#	Timer 0 Auxiliary Mode	8FH	-	-	-	-	-	-	-	T0M2	00H	xxx0xxx0
											60H <sup>1,5</sup>	011xxx00
											00H <sup>5</sup>	00000000
											00H <sup>5</sup>	00000000

# GENERAL DESCRIPTION

## P89LPC901/902/903

Name	Description	SFR Address	Bit Functions and Addresses								Reset Value	
			MSB				LSB				Hex	Binary
TCON*	Timer 0 and 1 Control	88H	8F	8E	8D	8C	8B	8A	89	88	00H	00000000
			TF1	TR1	TF0	TR0	-	-	-	-		
TH0	Timer 0 High	8CH									00H	00000000
TH1	Timer 1 High	8DH									00H	00000000
TL0	Timer 0 Low	8AH									00H	00000000
TL1	Timer 1 Low	8BH									00H	00000000
TMOD	Timer 0 and 1 Mode	89H	-	-	T1M1	T1M0	-	T0C\T	T0M1	T0M0	00H	00000000
TRIM#	Internal Oscillator Trim Register	96H	-	ENCLK	TRIM.5	TRIM.4	TRIM.3	TRIM.2	TRIM.1	TRIM.0	Notes 4,5	
WDCON#	Watchdog Control Register	A7H	PRE2	PRE1	PRE0	-	-	WDRUN	WDTOF	WDCLK	Notes 3,5	
WDL#	Watchdog Load	C1H									FFH	11111111
WFEEED1#	Watchdog Feed 1	C2H										
WFEEED2#	Watchdog Feed 2	C3H										

# GENERAL DESCRIPTION

## P89LPC901/902/903

### Special Function Registers Table - P89LPC902

Name	Description	SFR Address	Bit Functions and Addresses								Reset Value									
			MSB				LSB				Hex	Binary								
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0			00H	00000000						
AUXR1#	Auxiliary Function Register	A2H	-	-	-	-	SRST	0	-	DPS			00H <sup>1</sup>	000000x0						
B*	B Register	F0H	F7	F6	F5	F4	F3	F2	F1	F0			00H	00000000						
CMP1#	Comparator 1 Control Register	ACH	-	-	CE1	-	CN1	OE1	CO1	CMF1			00H <sup>1</sup>	xx000000						
CMP2#	Comparator 2 Control Register	ADH	-	-	CE2	-	CN2	OE2	CO2	CMF2			00H <sup>1</sup>	xx000000						
DIVM#	CPU Clock Divide-by-M Control	95H											00H	00000000						
DPTR	Data Pointer (2 bytes)																			
DPH	Data Pointer High	83H											00H	00000000						
DPL	Data Pointer Low	82H											00H	00000000						
FMADRH#	Program Flash Address High	E7H											00H	00000000						
FMADRL#	Program Flash Address Low	E6H											00H	00000000						
FMCON#	Program Flash Control (Read)		BUSY	-	-	-	HVA	HVE	SV	OI			70H	01110000						
FMCON#	Program Flash Control (Write)	E4H	FMCMD. 7	FMCMD. 6	FMCMD. 5	FMCMD. 4	FMCMD. 3	FMCMD. 2	FMCMD. 1	FMCMD. 0										
FMDATA#	Program Flash Data	E5H											00H	00000000						
IEN0*	Interrupt Enable 0	A8H	AF	AE	AD	AC	AB	AA	A9	A8	EA	EWDRT	EBO	-	ET1	-	ET0	-	00H	00000000
IEN1#	Interrupt Enable 1	E8H	EF	EE	ED	EC	EB	EA	E9	E8	-	-	-	-	-	EC	EKBI	-	00H <sup>1</sup>	00x000000
IP0*	Interrupt Priority 0	B8H	BF	BE	BD	BC	BB	BA	B9	B8	-	PWDRT	PBO	-	PT1	-	PT0	-	00H <sup>1</sup>	x0000000
IP0H#	Interrupt Priority 0 High	B7H	-	PWDRT H	PBOH	-	PT1H	-	PT0H	-									00H <sup>1</sup>	x00000000
IP1#	Interrupt Priority 1	F8H	FF	FE	FD	FC	FB	FA	F9	F8	-	-	-	-	-	PC	PKBI	-	00H <sup>1</sup>	00x000000
IP1H#	Interrupt Priority 1 High	F7H	-	-	-	-	-	PCH	PKBIH	-									00H <sup>1</sup>	00x000000
KBCON#	Keypad Control Register	94H	-	-	-	-	-	-	PATN_S EL	KBIF									00H <sup>1</sup>	xxxxxx00
KBMASK#	Keypad Interrupt Mask Register	86H																	00H	00000000

# GENERAL DESCRIPTION

## P89LPC901/902/903

Name	Description	SFR Address	Bit Functions and Addresses						Reset Value				
			MSB			LSB			Hex	Binary			
KBPATN#	Keypad Pattern Register	93H							FFH	11111111			
P0*	Port 0	80H	87	86	85	84	83	82	81	80	Note 1		
			-	CMP1/ KB6	CMPREF/ KB5	CIN1A/ KB4	-	KB2	-	KB0			
P1*	Port 1	90H	97	96	95	94	93	92	91	90			
			-	-	RST	-	-	-	-	-			
P0M1#	Port 0 Output Mode 1	84H	B7	B6	B5	B4	B3	B2	B1	B0	FFH	11111111	
P0M2#	Port 0 Output Mode 2	85H	-	(P0M1.6)	(P0M1.5)	(P0M1.4)	-	(P0M1.2)	-	(P0M1.0)	00H	00000000	
P1M1#	Port 1 Output Mode 1	91H	-	-	(P0M2.6)	(P0M2.5)	(P0M2.4)	-	(P0M2.2)	-	(P0M2.0)	FFH <sup>1</sup>	11111111
P1M2#	Port 1 Output Mode 2	92H	-	-	(P1M1.5)	-	-	-	-	-	00H <sup>1</sup>	00000000	
PCON#	Power Control Register	87H	-	-	(P1M2.5)	-	-	-	-	-	00H	00000000	
PCONA#	Power Control Register A	B5H	RTCPD	-	BOPD	BOI	GF1	GF0	PMOD1	PMOD0	00H <sup>1</sup>	00000000	
PSW*	Program Status Word	D0H	D7	D6	D5	D4	D3	D2	D1	D0	00H	00000000	
PT0AD#	Port 0 Digital Input Disable	F6H	CY	AC	F0	RS1	RS0	OV	F1	P	00H	xx0000x	
RSTSRC#	Reset Source Register	DFH	-	-	PT0AD.5	PT0AD.4	-	PT0AD.2	-	-	00H	xx0000x	
RTCCON#	Real-Time Clock Control	D1H	-	-	BOF	POF	-	R_WD	R_SF	R_EX	Note 2		
RTCH#	Real-Time Clock Register High	D2H	RTCF	RTCS1	RTCS0	-	-	-	ERTC	RTCEN	60H <sup>1.5</sup>	011xx00	
RTCL#	Real-Time Clock Register Low	D3H							00H <sup>5</sup>	00000000			
SP	Stack Pointer	81H							00H <sup>5</sup>	00000000			
TCON*	Timer 0 and 1 Control	88H	8F	8E	8D	8C	8B	8A	89	88	07H	00000111	
TH0	Timer 0 High	8CH	TF1	TR1	TF0	TR0	-	-	-	-	00H	00000000	
TH1	Timer 1 High	8DH							00H	00000000			
TL0	Timer 0 Low	8AH							00H	00000000			
TL1	Timer 1 Low	8BH							00H	00000000			
TMOD	Timer 0 and 1 Mode	89H	-	-	T1M1	T1M0	-	-	T0M1	T0M0	00H	00000000	
TRIM#	Internal Oscillator Trim Register	96H	-	-	TRIM.5	TRIM.4	TRIM.3	TRIM.2	TRIM.1	TRIM.0	Notes 4,5		

# GENERAL DESCRIPTION

P89LPC901/902/903

Name	Description	SFR Address	Bit Functions and Addresses							Reset Value	
			MSB			LSB				Hex	Binary
WDCON#	Watchdog Control Register	A7H	PRE2	PRE1	PRE0	-	-	WDRUN	WDTOF	WDCLK	Notes 3,5
WDL#	Watchdog Load	C1H								FFH	11111111
WFEEED1#	Watchdog Feed 1	C2H									
WFEEED2#	Watchdog Feed 2	C3H									

# GENERAL DESCRIPTION

P89LPC901/902/903

## Special Function Registers Table - P89LPC903

Name	Description	SFR Address	Bit Functions and Addresses						Reset Value			
			MSB			LSB			Hex	Binary		
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H	00000000
AUXR1#	Auxiliary Function Register	A2H	-	EBRR	-	-	SRST	0	-	DPS	00H <sup>1</sup>	000000x0
B*	B Register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H	00000000
BRGR0#§	Baud Rate Generator Rate Low	BEH									00H	00000000
BRGR1#§	Baud Rate Generator Rate High	BFH									00H	00000000
BRGCON#	Baud Rate Generator Control	BDH	-	-	-	-	-	-	SBRGS	BRGEN	00H%	xxxxxx00
CMP1#	Comparator 1 Control Register	ACH	-	-	CE1	-	CN1	-	CO1	CMF1	00H <sup>1</sup>	xx000000
CMP2#	Comparator 2 Control Register	ADH	-	-	CE2	-	CN2	-	CO2	CMF2	00H <sup>1</sup>	xx000000
DIVM#	CPU Clock Divide-by-M Control	95H									00H	00000000
DPTR	Data Pointer (2 bytes)											
DPH	Data Pointer High	83H									00H	00000000
DPL	Data Pointer Low	82H									00H	00000000
FMADRH#	Program Flash Address High	E7H									00H	00000000
FMADRL#	Program Flash Address Low	E6H									00H	00000000
FMCON#	Program Flash Control (Read)		BUSY	-	-	-	HVA	HVE	SV	OI	70H	01110000
FMCON#	Program Flash Control (Write)	E4H	FMCMD. 7	FMCMD. 6	FMCMD. 5	FMCMD. 4	FMCMD. 3	FMCMD. 2	FMCMD. 1	FMCMD. 0		
FMDATA#	Program Flash Data	E5H									00H	00000000
IEN0*	Interrupt Enable 0	A8H	AF	AE	AD	AC	AB	AA	A9	A8		
			EA	EWDRT	EBO	ES/ESR	ET1	-	ET0	-	00H	00000000
IEN1#	Interrupt Enable 1	E8H	EF	EE	ED	EC	EB	EA	E9	E8		
			-	EST	-	-	-	EC	EKBI	-	00H <sup>1</sup>	00x00000
IP0*	Interrupt Priority 0	B8H	BF	BE	BD	BC	BB	BA	B9	B8		
			-	PWDRT	PBO	PS/PSR	PT1	-	PT0	-	00H <sup>1</sup>	x0000000
IP0H#	Interrupt Priority 0 High	B7H	-	PWDRT H	PBOH	PSH/ PSRH	PT1H	-	PT0H	-	00H <sup>1</sup>	x0000000
IP1#	Interrupt Priority 1	F8H	FF	FE	FD	FC	FB	FA	F9	F8		
			-	PST	-	-	-	PC	PKBI	-	00H <sup>1</sup>	00x00000

# GENERAL DESCRIPTION

## P89LPC901/902/903

Name	Description	SFR Address	Bit Functions and Addresses								Reset Value	
			MSB				LSB				Hex	Binary
IP1H#	Interrupt Priority 1 High	F7H	-	PSTH	-	-	-	PCH	PKBIH	-	00H <sup>1</sup>	00x00000
KBCON#	Keypad Control Register	94H	-	-	-	-	-	PATN_S EL	KBIF		00H <sup>1</sup>	xxxxxx00
KBMASK#	Keypad Interrupt Mask Register	86H									00H	00000000
KBPATN#	Keypad Pattern Register	93H									FFH	11111111
			87	86	85	84	83	82	81	80		
P0*	Port 0	80H	-	-	CMPREF/ KB5	CIN1A/ KB4	-	KB2	-	-		Note 1
			97	96	95	94	93	92	91	90		
P1*	Port 1	90H	-	-	RST	-	-	-	RxD	TxD		
P0M1#	Port 0 Output Mode 1	84H	-	-	(P0M1.5)	(P0M1.4)	-	(P0M1.2)	-	-	FFH	11111111
P0M2#	Port 0 Output Mode 2	85H	-	-	(P0M2.5)	(P0M2.4)	-	(P0M2.2)	-	-	00H	00000000
P1M1#	Port 1 Output Mode 1	91H	-	-	(P1M1.5)	-	-	-	(P1M1.1)	(P1M1.0)	FFH <sup>1</sup>	11111111
P1M2#	Port 1 Output Mode 2	92H	-	-	(P1M2.5)	-	-	-	(P1M2.1)	(P1M2.0)	00H <sup>1</sup>	00000000
PCON#	Power Control Register	87H	SMOD1	SMOD0	BOPD	BOI	GF1	GF0	PMOD1	PMOD0	00H	00000000
PCONA#	Power Control Register A	B5H	RTCPD	-	VCPD	-	-	-	SPD	-	00H <sup>1</sup>	00000000
			D7	D6	D5	D4	D3	D2	D1	D0		
PSW*	Program Status Word	D0H	CY	AC	F0	RS1	RS0	OV	F1	P	00H	00000000
PT0AD#	Port 0 Digital Input Disable	F6H	-	-	PT0AD.5	PT0AD.4	-	PT0AD.2	-	-	00H	xx00000x
RSTSRC#	Reset Source Register	DFH	-	-	BOF	POF	R_BK	R_WD	R_SF	R_EX		Note 2
RTCCON#	Real-Time Clock Control	D1H	RTCF	RTCS1	RTCS0	-	-	-	ERTC	RTCEN	60H <sup>1,5</sup>	011xxx00
RTCH#	Real-Time Clock Register High	D2H									00H <sup>5</sup>	00000000
RTCL#	Real-Time Clock Register Low	D3H									00H <sup>5</sup>	00000000
SADDR#	Serial Port Address Register	A9H									00H	00000000
SADEN#	Serial Port Address Enable	B9H									00H	00000000
SBUF	Serial Port Data Buffer Register	99H									xxH	xxxxxxxx
			9F	9E	9D	9C	9B	9A	99	98		
SCON*	Serial Port Control	98H	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI	00H	00000000
SSTAT#	Serial Port Extended Status Register	BAH	DBMOD	INTLO	CIDIS	DBISEL	FE	BR	OE	STINT	00H	00000000
SP	Stack Pointer	81H									07H	00001111
			8F	8E	8D	8C	8B	8A	89	88		
TCON*	Timer 0 and 1 Control	88H	TF1	TR1	TF0	TR0	-	-	-	-	00H	00000000

# GENERAL DESCRIPTION

## P89LPC901/902/903

Name	Description	SFR Address	Bit Functions and Addresses								Reset Value	
			MSB				LSB				Hex	Binary
TH0	Timer 0 High	8CH									00H	00000000
TH1	Timer 1 High	8DH									00H	00000000
TL0	Timer 0 Low	8AH									00H	00000000
TL1	Timer 1 Low	8BH									00H	00000000
TMOD	Timer 0 and 1 Mode	89H	-	-	T1M1	T1M0	-	-	T0M1	T0M0	00H	00000000
TRIM#	Internal Oscillator Trim Register	96H	-	-	TRIM.5	TRIM.4	TRIM.3	TRIM.2	TRIM.1	TRIM.0		Notes 4,5
WDCON#	Watchdog Control Register	A7H	PRE2	PRE1	PRE0	-	-	WDRUN	WDTOF	WDCLK		Notes 3,5
WDL#	Watchdog Load	C1H									FFH	11111111
WFEEED1#	Watchdog Feed 1	C2H										
WFEEED2#	Watchdog Feed 2	C3H										

**Notes:**

- \* SFRs are bit addressable.
- # SFRs are modified from or added to the 80C51 SFRs.
- Reserved bits, must be written with 0's.
- § BRGR1 and BRGR0 must only be written if BRGEN in BRGCON SFR is '0'. If any of them is written if BRGEN = 1, result is unpredictable.

Unimplemented bits in SFRs (labeled '-') are X (unknown) at all times. Unless otherwise specified, ones should not be written to these bits since they may be used for other purposes in future derivatives. The reset values shown for these bits are '0's although they are unknown when read.

1. All ports are in input only (high impedance) state after power-up.
2. The RSTSRC register reflects the cause of the **P89LPC901/902/903** reset. Upon a power-up reset, all reset source flags are cleared except POF and BOF - the power-on reset value is xx110000.
3. After reset, the value is 111001x1, i.e., PRE2-PRE0 are all 1, WDRUN=1 and WDCLK=1. WDTOF bit is 1 after watchdog reset and is 0 after power-on reset. Other resets will not affect WDTOF.
4. On power-on reset, the TRIM SFR is initialized with a factory preprogrammed value. Other resets will not cause initialization of the TRIM register.
5. The only reset source that affects these SFRs is power-on reset.

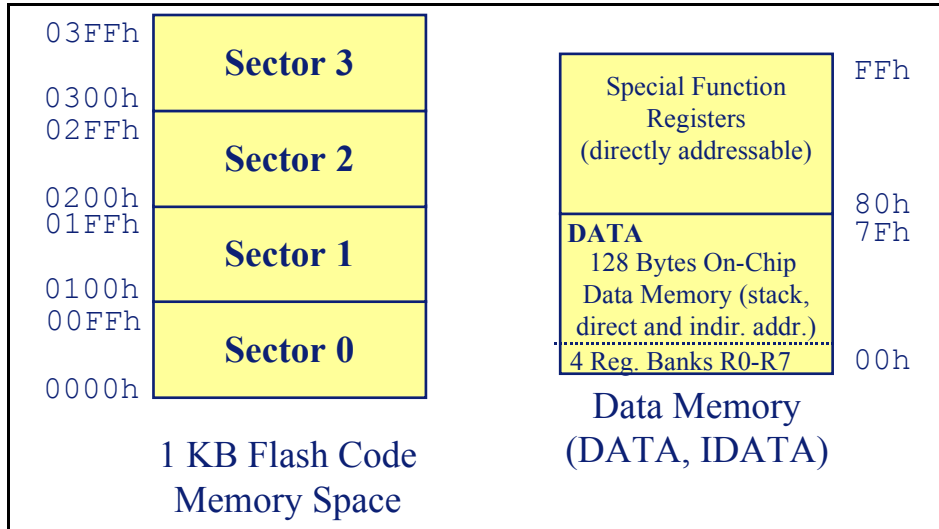


# GENERAL DESCRIPTION

## P89LPC901/902/903

### Memory Organization

The P89LPC901/902/903 memory map is shown in Figure 1-1.



**Figure 1-1: P89LPC901/902/903 Memory Map**

The various P89LPC901/902/903 memory spaces are as follows:

- DATA** 128 bytes of internal data memory space (00h..7Fh) accessed via direct or indirect addressing, using instructions other than MOVX and MOVC.
- SFR** Special Function Registers. Selected CPU registers and peripheral control and status registers, accessible only via direct addressing.
- CODE** 1KB of Code memory accessed as part of program execution and via the MOVC instruction.

---

# GENERAL DESCRIPTION

---

P89LPC901/902/903

# CLOCKS

P89LPC901/902/903

## 2. CLOCKS

### Enhanced CPU

The **P89LPC901/902/903** uses an enhanced 80C51 CPU which runs at 6 times the speed of standard 80C51 devices. A machine cycle consists of two CPU clock cycles, and most instructions execute in one or two machine cycles.

### Clock Definitions

The **P89LPC901/902/903** device has several internal clocks as defined below:

- OSCCLK - Input to the DIVM clock divider. OSCCLK is selected from one of the clock sources (see Figure 2-3, Figure 2-4, Figure 2-5) and can also be optionally divided to a slower frequency (see section "CPU Clock (CCLK) Modification: DIVM Register"). **Note:**  $f_{OSC}$  is defined as the OSCCLK frequency.
- XCLK - Output of the crystal oscillator (**P89LPC901**)
- CCLK - CPU clock.
- PCLK - Clock for the various peripheral devices and is CCLK/2

### CPU Clock (OSCCLK)

The **P89LPC901** provides several user-selectable oscillator options. This allows optimization for a range of needs from high precision to lowest possible cost. These options are configured when the FLASH is programmed and include an on-chip watchdog oscillator, an on-chip RC oscillator, an oscillator using an external crystal, or an external clock source. The crystal oscillator can be optimized for low, medium, or high frequency crystals covering a range from 20 kHz to 12 MHz.

The **P89LPC902** and **P89LPC903** devices allow the user to select between an on-chip watchdog oscillator and an on-chip RC oscillator as the CPU clock source.

### Low Speed Oscillator Option - P89LPC901

This option supports an external crystal in the range of 20 kHz to 100 kHz. Ceramic resonators are also supported in this configuration.

### Medium Speed Oscillator Option - P89LPC901

This option supports an external crystal in the range of 100 kHz to 4 MHz. Ceramic resonators are also supported in this configuration.

### High Speed Oscillator Option - P89LPC901

This option supports an external crystal in the range of 4MHz to 12 MHz. Ceramic resonators are also supported in this configuration. If CCLK is 8MHz or slower, the CLKLP SFR bit (AUXR1.7) can be set to '1' to reduce power consumption. On reset, CLKLP is '0' allowing highest performance access. This bit can then be set in software if CCLK is running at 8MHz or slower.

# CLOCKS

## P89LPC901/902/903

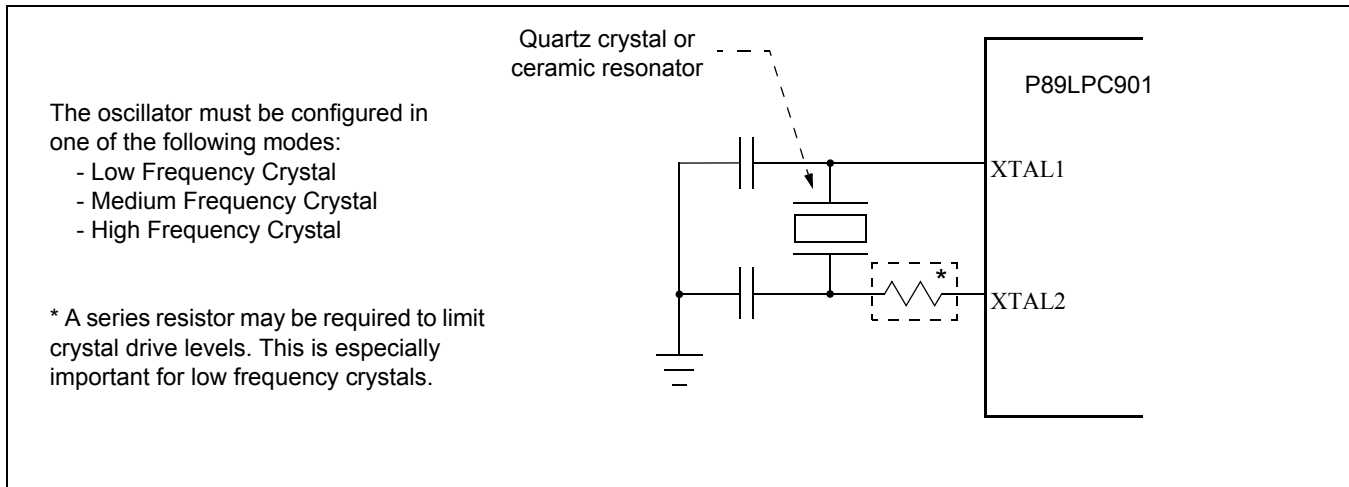


Figure 2-1: Using the Crystal Oscillator - P89LPC901

### Oscillator Option Selection- P89LPC901

The oscillator option is selectable either by the FOSC2:0 bits in UCFG1 or by the RTCS1:0 bits in RTCCON. If the FOSC2:0 bits select an OSCCLK source of either the internal RC oscillator or the WDT oscillator, then the RTCS1:0 bits will select the oscillator option for the crystal oscillator. Otherwise, the crystal oscillator option is selected by FOSC2:0. See Table 6-1 and Table 6-2.

### Clock Output - P89LPC901

The **P89LPC901** supports a user selectable clock output function on the XTAL2 / CLKOUT pin when no crystal oscillator is being used. This condition occurs if another clock source has been selected (on-chip RC oscillator, watchdog oscillator, external clock input on X1) and if the Real-Time clock is not using the crystal oscillator as its clock source. This allows external devices to synchronize to the **P89LPC901**. This output is enabled by the ENCLK bit in the TRIM register

The frequency of this clock output is 1/2 that of the CCLK. If the clock output is not needed in Idle mode, it may be turned off prior to entering Idle, saving additional power. Note: on reset, the TRIM SFR is initialized with a factory preprogrammed value. Therefore when setting or clearing the ENCLK bit, the user should retain the contents of bits 5:0 of the TRIM register. This can be done by reading the contents of the TRIM register (into the ACC for example), modifying bit 6, and writing this result back into the TRIM register. Alternatively, the "ANL direct" or "ORL direct" instructions can be used to clear or set bit 6 of the TRIM register.

### On-Chip RC oscillator Option

The **P89LPC901/902/903** has a 6-bit field within the TRIM register that can be used to tune the frequency of the RC oscillator. During reset, the TRIM value is initialized to a factory pre-programmed value to adjust the oscillator frequency to 7.373 MHz,  $\pm 1\%$ . (Note: the initial value is better than 1%; please refer to the datasheet for behavior over temperature). End user applications can write to the TRIM register to adjust the on-chip RC oscillator to other frequencies. Increasing the TRIM value will decrease the oscillator frequency.

# CLOCKS

## P89LPC901/902/903

If CCLK is 8MHz or slower, the CLKLP SFR bit (AUXR1.7) can be set to '1' to reduce power consumption. On reset, CLKLP is '0' allowing highest performance access. This bit can then be set in software if CCLK is running at 8MHz or slower.

TRIM	7	6	5	4	3	2	1	0
Address: 96h	-	ENCLK	TRIM.5	TRIM.4	TRIM.3	TRIM.2	TRIM.1	TRIM.0
Not bit addressable								
Reset Source(s): Power-up only								
Reset Value: On power-up reset, ENCLK = 0, and TRIM.5-0 are loaded with the factory programmed value.								
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>						
TRIM.7	-	Reserved.						
TRIM.6	ENCLK	When ENCLK =1, CCLK/ 2 is output on the XTAL2 pin (P3.0) provided that the crystal oscillator is not being used. When ENCLK=0, no clock output is enabled.(P89LPC901).						
TRIM.5-0		Trim value.						

Note: on reset, the TRIM SFR is initialized with a factory preprogrammed value. When setting or clearing either the ENCLK or RCCLK bits, the user should retain the contents of bits 5:0 of the TRIM register. This can be done by reading the contents of the TRIM register (into the ACC for example), modifying bits 6 or 7, and writing this result back into the TRIM register. Alternatively, the "ANL direct" or "ORL direct" instructions can be used to clear or set bit 6 or 7 of the TRIM register.

Figure 2-2: On-Chip RC Oscillator TRIM Register

### Watchdog Oscillator Option

The watchdog has a separate oscillator which has a nominal frequency of 400KHz. This oscillator can be used to save power when a high clock frequency is not needed.

### External Clock Input Option - P89LPC901

In this configuration, the processor clock is derived from an external source driving the XTAL1 / P3.1 pin. The rate may be from 0 Hz up to 12 MHz. The XTAL2 / P3.0 pin may be used as a standard port pin or a clock output.

# CLOCKS

## P89LPC901/902/903

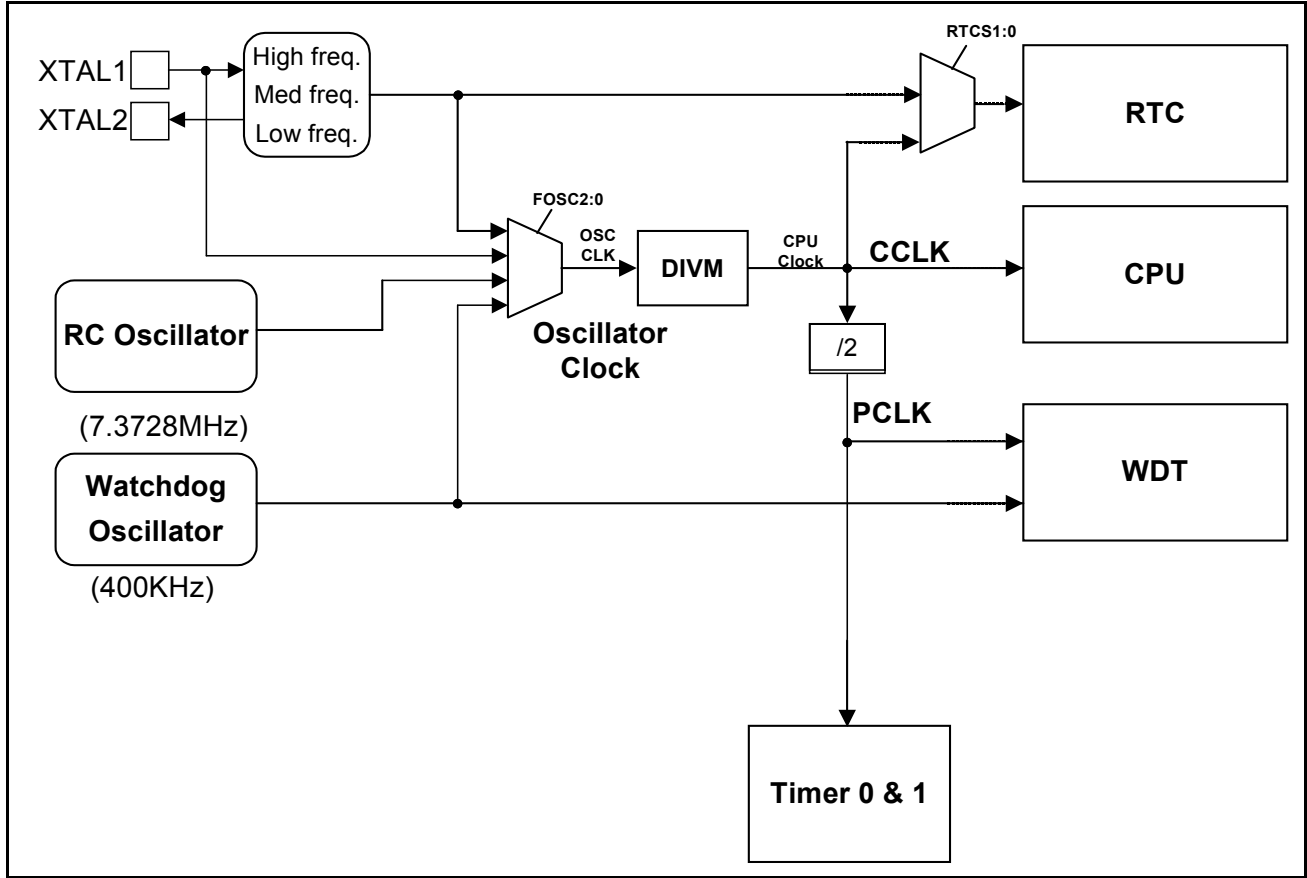


Figure 2-3: Block Diagram of Oscillator Control - P89LPC901

# CLOCKS

P89LPC901/902/903

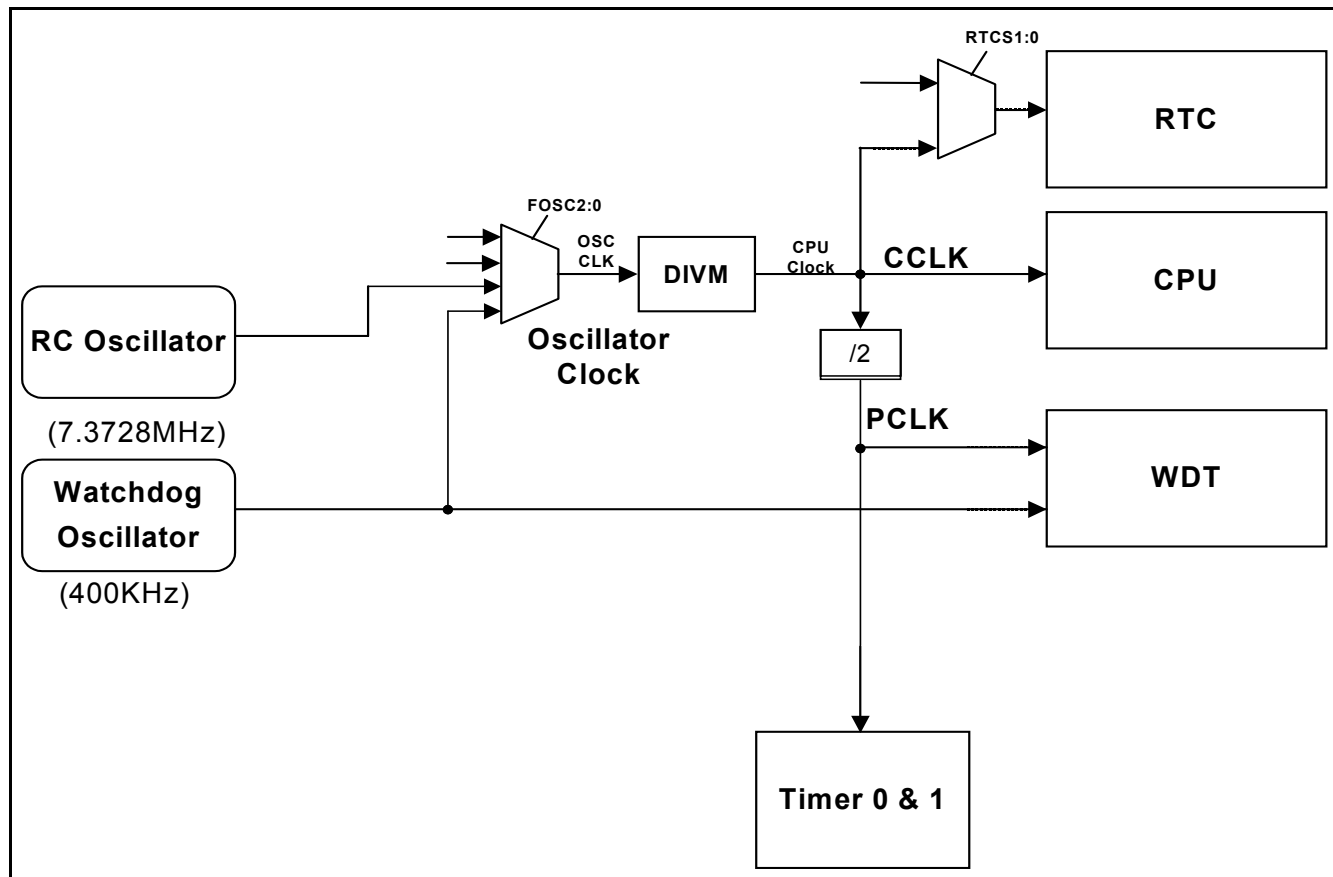


Figure 2-4: Block Diagram of Oscillator Control - P89LPC902

# CLOCKS

## P89LPC901/902/903

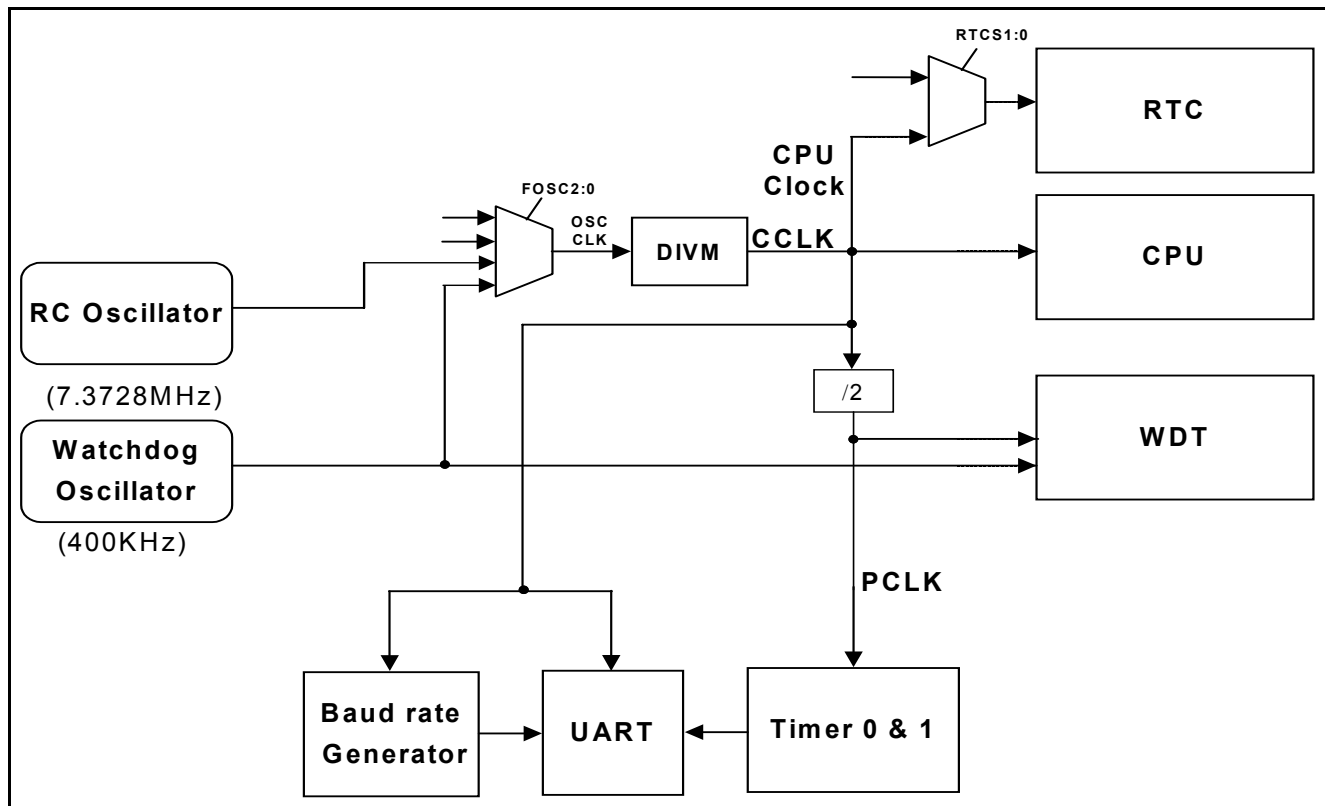


Figure 2-5: Block Diagram of Oscillator Control- P89LPC903

### CPU Clock (CCLK) Wakeup Delay

The **P89LPC901/902/903** has an internal wakeup timer that delays the clock until it stabilizes depending to the clock source used. If the clock source is any of the three crystal selections (**P89LPC901**), the delay is 992 OSCCLK cycles plus 60-100µs. If the clock source is either the internal RC oscillator or the Watchdog oscillator, the delay is 224 OSCCLK cycles plus 60-100µs.

### CPU Clock (CCLK) Modification: DIVM Register

The OSCCLK frequency can be divided down, by an integer, up to 510 times by configuring a dividing register, DIVM, to provide CCLK. This produces the CCLK frequency using the following formula:

$$\text{CCLK frequency} = f_{\text{OSC}} / (2N)$$

Where:  $f_{\text{OSC}}$  is the frequency of OSCCLK

N is the value of DIVM.

Since N ranges from 0 to 255, the CCLK frequency can be in the range of  $f_{\text{OSC}}$  to  $f_{\text{OSC}}/510$  (for N =0, CCLK =  $f_{\text{OSC}}$ ).

This feature makes it possible to temporarily run the CPU at a lower rate, reducing power consumption. By dividing the clock, the CPU can retain the ability to respond to events other than those that can cause interrupts (i.e. events that allow exiting the Idle mode) by executing its normal program at a lower rate. This can often result in lower power consumption than in Idle mode. This can allow bypassing the oscillator start-up time in cases where Power down mode would otherwise be used. The value of DIVM may be changed by the program at any time without interrupting code execution.



---

# CLOCKS

---

**P89LPC901/902/903**

## Low Power Select (P89LPC901)

The P89LPC901 is designed to run at 12MHz (CCLK) maximum. However, if CCLK is 8MHz or slower, the CLKLP SFR bit (AUXR1.7) can be set to a '1' to lower the power consumption further. On any reset, CLKLP is '0' allowing highest performance. This bit can then be set in software if CCLK is running at 8MHz or slower.

---

# CLOCKS

---

P89LPC901/902/903

# INTERRUPTS

**P89LPC901/902/903**

## 3. INTERRUPTS

The **P89LPC901/902/903** use a four priority level interrupt structure. This allows great flexibility in controlling the handling of the many interrupt sources. The **P89LPC901** supports 6 interrupt sources: timers 0 and 1, brownout detect, watchdog/ realtime clock, keyboard, and the comparator. The **P89LPC902** supports 6 interrupt sources: timers 0 and 1, brownout detect, watchdog/ realtime clock, keyboard, and comparators 1 and 2. The **P89LPC903** supports 9 interrupt sources: timers 0 and 1, serial port Tx, serial port Rx, combined serial port Rx/Tx, brownout detect, watchdog/ realtime clock, keyboard, and comparators 1 and 2.

Each interrupt source can be individually enabled or disabled by setting or clearing a bit in the interrupt enable registers IEN0 or IEN1. The IEN0 register also contains a global enable bit, EA, which enables all interrupts.

Each interrupt source can be individually programmed to one of four priority levels by setting or clearing bits in the interrupt priority registers IP0, IP0H, IP1, and IP1H. An interrupt service routine in progress can be interrupted by a higher priority interrupt, but not by another interrupt of the same or lower priority. The highest priority interrupt service cannot be interrupted by any other interrupt source. If two requests of different priority levels are pending at the start of an instruction, the request of higher priority level is serviced.

If requests of the same priority level are pending at the start of an instruction, an internal polling sequence determines which request is serviced. This is called the arbitration ranking. Note that the arbitration ranking is only used to resolve pending requests of the same priority level.

Table summarizes the interrupt sources, flag bits, vector addresses, enable bits, priority bits, arbitration ranking, and whether each interrupt may wake up the CPU from a Power down mode.

### Interrupt Priority Structure

There are four SFRs associated with the four interrupt levels: IP0, IP0H, IP1, IP1H. Every interrupt has two bits in IPx and IPxH (x = 0,1) and can therefore be assigned to one of four levels, as shown in Table .

**Table 3-1: Interrupt priority level**

Priority Bits		Interrupt Priority Level
IPxH	IPx	
0	0	Level 0 (lowest priority)
0	1	Level 1
1	0	Level 2
1	1	Level 3 (highest priority)

# INTERRUPTS

**P89LPC901/902/903****Table 3-2: Summary of Interrupts - P89LPC901**

Description	Interrupt Flag Bit(s)	Vector Address	Interrupt Enable Bit(s)	Interrupt Priority	Arbitration Ranking	Power down Wakeup
Timer 0 Interrupt	TF0	000Bh	ET0 (IEN0.1)	IP0H.1, IP0.1	3	No
Timer 1 Interrupt	TF1	001Bh	ET1 (IEN0.3)	IP0H.3, IP0.3	5	No
Brownout Detect	BOF	002Bh	EBO (IEN0.5)	IP0H.5, IP0.5	1	Yes
Watchdog Timer/Real-time Clock	WDOVF/ RTCF	0053h	EWDRT (IEN0.6)	IP0H.6, IP0.6	2	Yes
KBI Interrupt	KBIF	003Bh	EKBI (IEN1.1)	IP1H.1, IP1.1	4	Yes
Comparator interrupt	CMF	0043h	EC (IEN1.2)	IP1H.2, IP1.2	6	Yes

**Table 3-3: Summary of Interrupts - P89LPC902**

Description	Interrupt Flag Bit(s)	Vector Address	Interrupt Enable Bit(s)	Interrupt Priority	Arbitration Ranking	Power down Wakeup
Timer 0 Interrupt	TF0	000Bh	ET0 (IEN0.1)	IP0H.1, IP0.1	3	No
Timer 1 Interrupt	TF1	001Bh	ET1 (IEN0.3)	IP0H.3, IP0.3	5	No
Brownout Detect	BOF	002Bh	EBO (IEN0.5)	IP0H.5, IP0.5	1	Yes
Watchdog Timer/Real-time Clock	WDOVF/ RTCF	0053h	EWDRT (IEN0.6)	IP0H.6, IP0.6	2	Yes
KBI Interrupt	KBIF	003Bh	EKBI (IEN1.1)	IP1H.1, IP1.1	4	Yes
Comparator interrupt	CMF	0043h	EC (IEN1.2)	IP1H.2, IP1.2	6	Yes

**Table 3-4: Summary of Interrupts - P89LPC903**

Description	Interrupt Flag Bit(s)	Vector Address	Interrupt Enable Bit(s)	Interrupt Priority	Arbitration Ranking	Power down Wakeup
Timer 0 Interrupt	TF0	000Bh	ET0 (IEN0.1)	IP0H.1, IP0.1	3	No
Timer 1 Interrupt	TF1	001Bh	ET1 (IEN0.3)	IP0H.3, IP0.3	5	No
Serial Port Tx and Rx <sup>1,3</sup>	TI & RI	0023h	ES/ESR (IEN0.4)	IP0H.4, IP0.4	8	No
Serial Port Rx <sup>1,3</sup>	RI					
Brownout Detect	BOF	002Bh	EBO (IEN0.5)	IP0H.5, IP0.5	1	Yes
Watchdog Timer/Real-time Clock	WDOVF/ RTCF	0053h	EWDRT (IEN0.6)	IP0H.6, IP0.6	2	Yes
KBI Interrupt	KBIF	003Bh	EKBI (IEN1.1)	IP1H.1, IP1.1	4	Yes
Comparator interrupt	CMF	0043h	EC (IEN1.2)	IP1H.2, IP1.2	6	Yes
Serial Port Tx <sup>2</sup>	TI	006Bh	EST (IEN1.6)	P1H.6, IP1.6	7	No

1. SSTAT.5 = 0 selects combined Serial Port (UART) Tx and Rx interrupt; SSTAT.5 = 1 selects Serial Port Rx interrupt only (Tx interrupt will be different, see Note 3 below).

2. This interrupt is used as Serial Port (UART) Tx interrupt if and only if SSTAT.5 = 1, and is disabled otherwise.

3. If SSTAT.0 = 1, the following Serial Port additional flag bits can cause this interrupt: FE, BR, OE

# INTERRUPTS

## P89LPC901/902/903

### External Interrupt Inputs

The P89LPC901/902/903 have a Keypad Interrupt function (see Keypad Interrupt (KBI) on page 79). This can be used as an external interrupt input. If enabled when the P89LPC901/902/903 is put into Power down or Idle mode, the keypad interrupt will cause the processor to wake up and resume operation. Refer to the section on Power Reduction Modes for details.

### External Interrupt Pin Glitch Suppression

Most of the P89LPC901/902/903 pins have glitch suppression circuits to reject short glitches (please refer to the P89LPC901/902/903 datasheet, AC Electrical Characteristics for glitch filter specifications) .

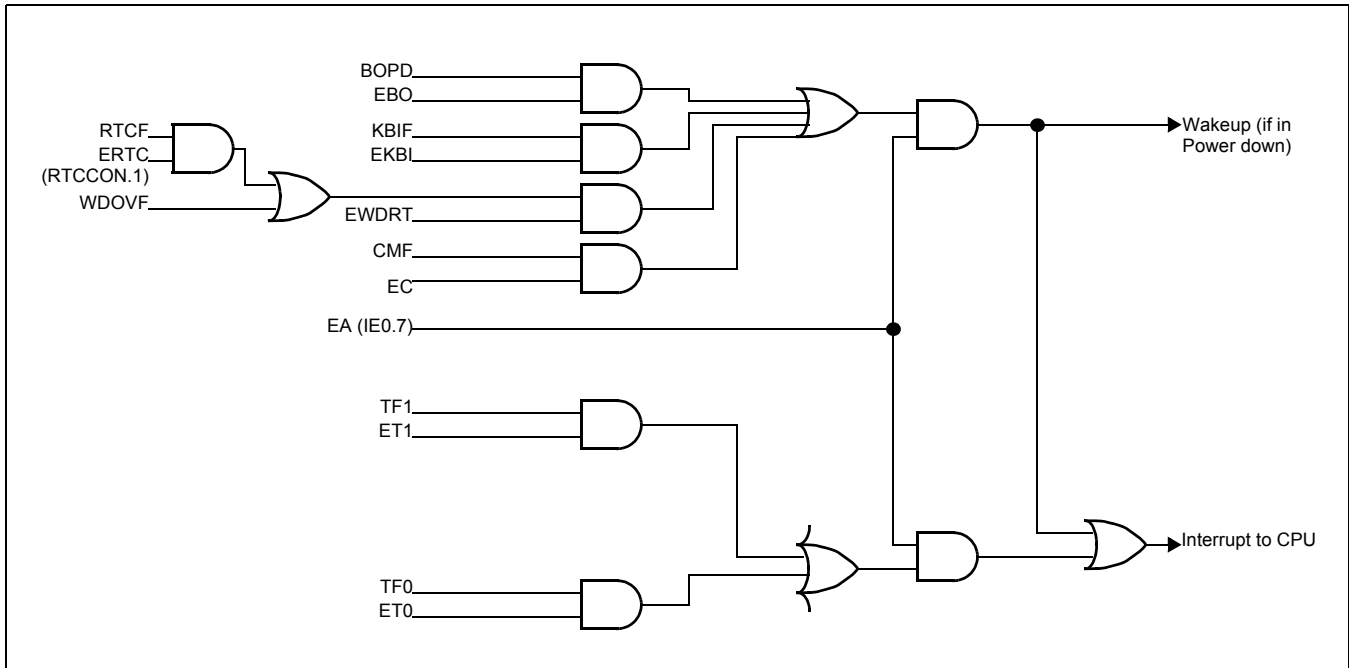


Figure 3-1: Interrupt sources, enables, and Power down Wake-up sources - P89LPC901

# INTERRUPTS

## P89LPC901/902/903

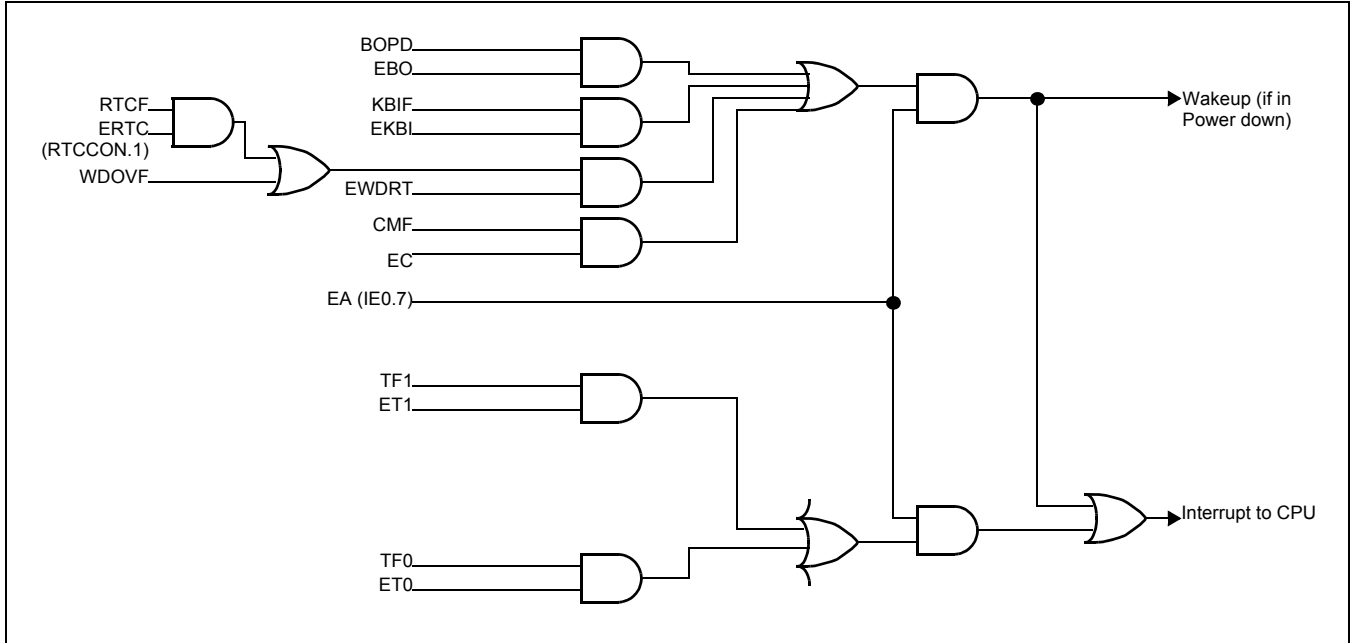


Figure 3-2: Interrupt sources, enables, and Power down Wake-up sources - P89LPC902

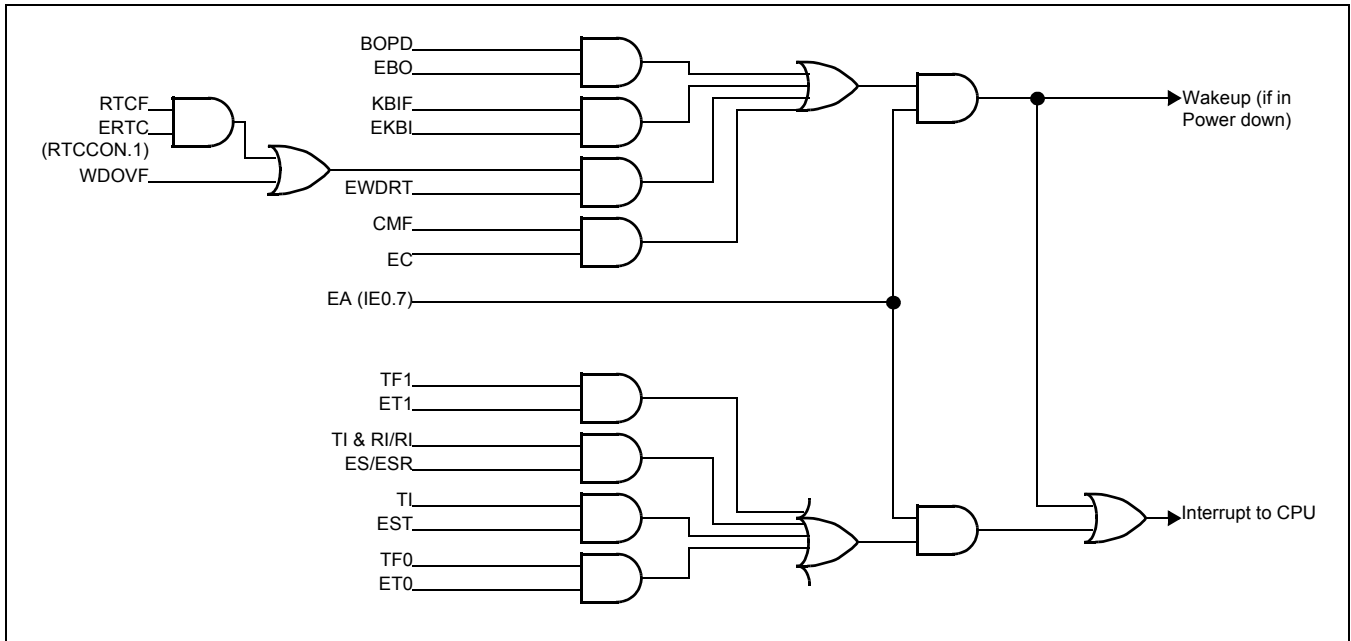


Figure 3-3: Interrupt sources, enables, and Power down Wake-up sources - P89LPC903

# I/O PORTS

## P89LPC901/902/903

### 4. I/O PORTS

The **P89LPC901/902/903** has between 3 and 6 I/O pins. The exact number of I/O pins available depends on the clock and reset options chosen:

**Table 4-1: Number of I/O Pins Available**

Clock Source	Reset Option	Number of I/O Pins
		8-Pin Package
On-chip oscillator or watchdog oscillator	No external reset(except during power-up)	6
	External $\overline{\text{RST}}$ pin supported	5
External clock input	No external reset(except during power-up)	5
	External $\overline{\text{RST}}$ pin supported	4
Low/medium/high speed oscillator (external crystal or resonator) (P89LPC901)	No external reset(except during power-up)	4
	External $\overline{\text{RST}}$ pin supported	3

### Port Configurations

All but one I/O port pin on the **P89LPC901/902/903** may be configured by software to one of four types on a pin-by-pin basis, as shown in Table . These are: quasi-bidirectional (standard 80C51 port outputs), push-pull, open drain, and input-only. Two configuration registers for each port select the output type for each port pin. P1.5 (RST) can only be an input and cannot be configured.

**Table 4-2: Port Output Configuration Settings**

PxM1.y	PxM2.y	Port Output Mode
0	0	Quasi-bidirectional
0	1	Push-Pull
1	0	Input Only (High Impedance)
1	1	Open Drain

### Quasi-Bidirectional Output Configuration

Quasi-bidirectional outputs can be used both as an input and output without the need to reconfigure the port. This is possible because when the port outputs a logic high, it is weakly driven, allowing an external device to pull the pin low. When the pin is driven low, it is driven strongly and able to sink a large current. There are three pullup transistors in the quasi-bidirectional output that serve different purposes.

One of these pullups, called the "very weak" pullup, is turned on whenever the port latch for the pin contains a logic 1. This very weak pullup sources a very small current that will pull the pin high if it is left floating.

A second pullup, called the "weak" pullup, is turned on when the port latch for the pin contains a logic 1 and the pin itself is also at a logic 1 level. This pullup provides the primary source current for a quasi-bidirectional pin that is outputting a 1. If this pin is pulled low by an external device, this weak pullup turns off, and only the very weak pullup remains on. In order to pull the pin low under these conditions, the external device has to sink enough current to overpower the weak pullup and pull the port pin below its input threshold voltage.

# I/O PORTS

## P89LPC901/902/903

The third pullup is referred to as the "strong" pullup. This pullup is used to speed up low-to-high transitions on a quasi-bidirectional port pin when the port latch changes from a logic 0 to a logic 1. When this occurs, the strong pullup turns on for two CPU clocks quickly pulling the port pin high .

The quasi-bidirectional port configuration is shown in Figure 4-1.

Although the **P89LPC901/902/903** is a 3V device the pins are 5V-tolerant (except for XTAL1 and XTAL2). If 5V is applied to a pin configured in quasi-bidirectional mode, there will be a current flowing from the pin to  $V_{DD}$  causing extra power consumption. Therefore, applying 5V to pins configured in quasi-bidirectional mode is discouraged.

A quasi-bidirectional port pin has a Schmitt-triggered input that also has a glitch suppression circuit. (Please refer to the **P89LPC901/902/903** datasheet, AC Characteristics for glitch filter specifications)

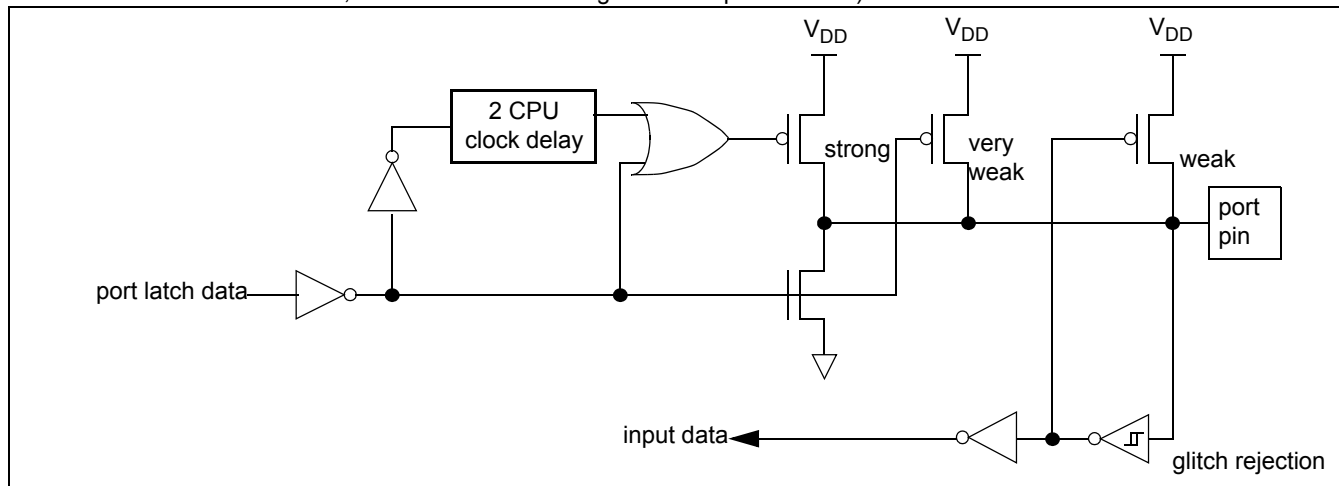


Figure 4-1: Quasi-Bidirectional Output

### Open Drain Output Configuration

The open drain output configuration turns off all pullups and only drives the pull-down transistor of the port pin when the port latch contains a logic 0. To be used as a logic output, a port configured in this manner must have an external pullup, typically a resistor tied to  $V_{DD}$ . The pull-down for this mode is the same as for the quasi-bidirectional mode.

The open drain port configuration is shown in Figure 4-2.

An open drain port pin has a Schmitt-triggered input that also has a glitch suppression circuit .(Please refer to the **P89LPC901/902/903** datasheet, AC Characteristics for glitch filter specifications).

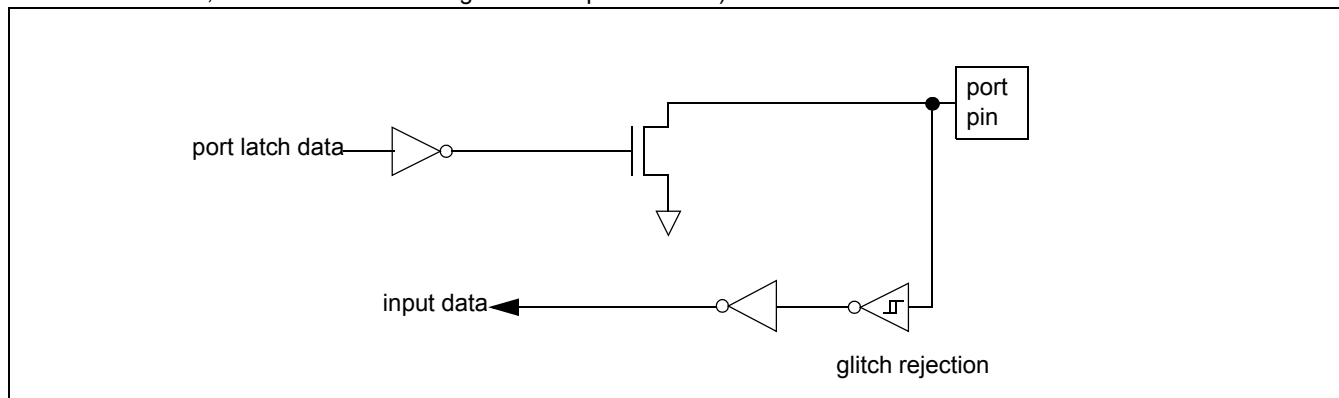


Figure 4-2: Open Drain Output



# I/O PORTS

## P89LPC901/902/903

### Input-Only Configuration

The input port configuration is shown in Figure 4-3. It is a Schmitt-triggered input that also has a glitch suppression circuit. (Please refer to the **P89LPC901/902/903** datasheet, AC Characteristics for glitch filter specifications)

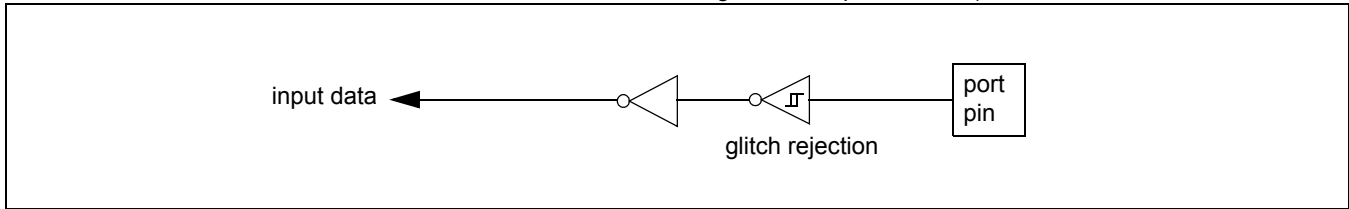


Figure 4-3: Input Only

### Push-Pull Output Configuration

The push-pull output configuration has the same pulldown structure as both the open drain and the quasi-bidirectional output modes, but provides a continuous strong pullup when the port latch contains a logic 1. The push-pull mode may be used when more source current is needed from a port output.

The push-pull port configuration is shown in Figure 4-4.

A push-pull port pin has a Schmitt-triggered input that also has a glitch suppression circuit. (Please refer to the **P89LPC901/902/903** datasheet, AC Characteristics for glitch filter specifications)

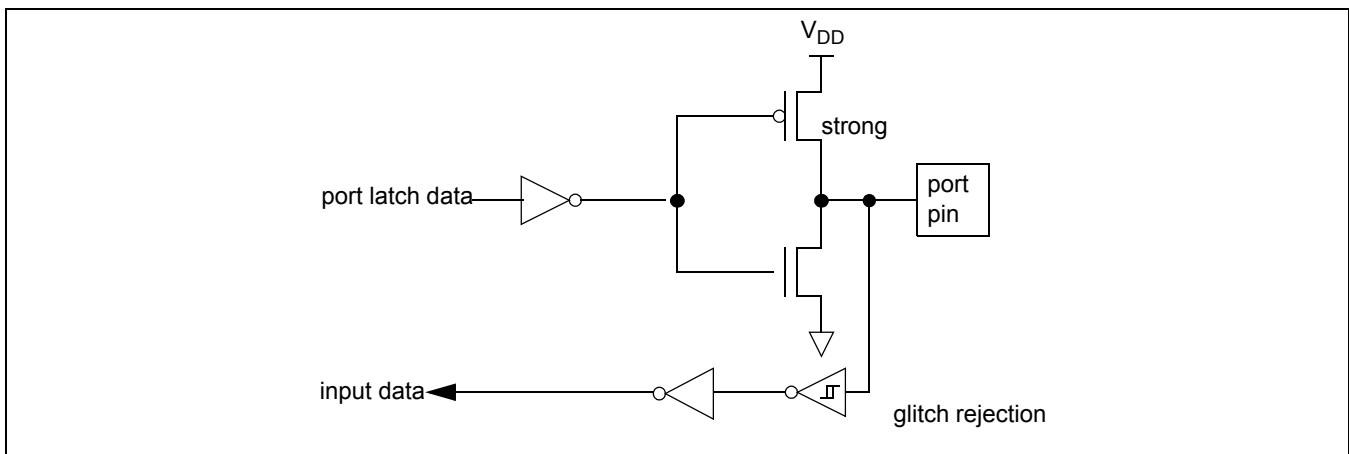


Figure 4-4: Push-Pull Output

### Port 0 Analog Functions

The **P89LPC901/902/903** incorporates up to two analog comparators. In order to give the best analog performance and minimize power consumption, pins that are being used for analog functions must have both the digital outputs and digital inputs disabled.

Digital outputs are disabled by putting the port pins into the input-only mode as described in the Port Configurations section (see Table ).

Digital inputs on Port 0 may be disabled through the use of the PT0AD register. On any reset, the PT0AD bits default to '0's to enable digital functions.

# I/O PORTS

## P89LPC901/902/903

**Table 4-3: Port Output Configuration - P89LPC901**

Port Pin	Configuration SFR Bits		Alternate Usage	Notes
	PxM1.y	PxM2.y		
P0.4	P0M1.4	P0M2.4	KBI4,CIN1A	Refer to section "Port 0 Analog Functions" for usage as analog inputs CINxA and CMPREF.
P0.5	P0M1.5	P0M2.5	KBI5,CMPREF	
P1.2	P1M1.2	P1M2.2	T0	
P1.5	not configurable		$\overline{\text{RST}}$	Input only. Usage as general purpose input or $\overline{\text{RST}}$ is determined by User Configuration Bit RPD (UCFG1.6). Always a reset input during a power-on sequence.
P3.0	P3M1.0	P3M2.0	XTAL2,CLKOUT	
P3.1	P3M1.1	P3M2.1	XTAL1	

**Table 4-4: Port Output Configuration - P89LPC902**

Port Pin	Configuration SFR Bits		Alternate Usage	Notes
	PxM1.y	PxM2.y		
P0.0	P0M1.0	P0M2.0	KBI0,CMP2	Refer to section "Port 0 Analog Functions" for usage as analog inputs CINxA and CMPREF.
P0.2	P0M1.2	P0M2.2	KBI2,CIN2A	
P0.4	P0M1.4	P0M2.4	KBI4,CIN1A	
P0.5	P0M1.5	P0M2.5	KBI5,CMPREF	
P0.6	P0M1.6	P0M2.6	KBI6,CMP1	
P1.5	not configurable		$\overline{\text{RST}}$	Input only. Usage as general purpose input or $\overline{\text{RST}}$ is determined by User Configuration Bit RPD (UCFG1.6). Always a reset input during a power-on sequence.

**Table 4-5: Port Output Configuration - P89LPC903**

Port Pin	Configuration SFR Bits		Alternate Usage	Notes
	PxM1.y	PxM2.y		
P0.2	P0M1.2	P0M2.2	KBI2,CIN2A	Refer to section "Port 0 Analog Functions" for usage as analog inputs CINxA and CMPREF.
P0.4	P0M1.4	P0M2.4	KBI4,CIN1A	
P0.5	P0M1.5	P0M2.5	KBI5,CMPREF	
P1.0	P1M1.0	P1M2.0	TxD	
P1.1	P1M1.1	P1M2.1	RxD	
P1.5	not configurable		$\overline{\text{RST}}$	Input only. Usage as general purpose input or $\overline{\text{RST}}$ is determined by User Configuration Bit RPD (UCFG1.6). Always a reset input during a power-on sequence.

## Additional Port Features

After power-up, all pins are in Input-Only mode. **Please note that this is different from the LPC76x series of devices.**

- After power-up, all I/O pins except P1.5, may be configured by software.
- Pin P1.5 is input only.
- Every output on the **P89LPC901/902/903** has been designed to sink typical LED drive current. However, there is a maximum total output current for all ports which must not be exceeded. Please refer to the **P89LPC901/902/903** datasheet for detailed specifications.

---

# I/O PORTS

## P89LPC901/902/903

---

All ports pins that can function as an output have slew rate controlled outputs to limit noise generated by quickly switching output signals. The slew rate is factory-set to approximately 10 ns rise and fall times.

---

# I/O PORTS

P89LPC901/902/903

---

# TIMERS 0 AND 1

# P89LPC901/902/903

## 5. TIMERS 0 AND 1

The **P89LPC901/902/903** has two general-purpose counter/timers which are similar to the 80C51 Timer 0 and Timer 1. Timer 0 of the **P89LPC901** can be configured to operate either as a timer or event counter (see Figure 5-1). An option to automatically toggle the T0 pin upon timer overflow has been added. Timer 1 of the **P89LPC901** and both Timer 0 and Timer 1 of the **P89LPC902** and **P89LPC903** devices may only function as timers.

In the "Timer" function, the timer is incremented every PCLK.

In the "Counter" function, the Timer 0 register is incremented in response to a 1-to-0 transition on the external input pin, T0, which is sampled once during every machine cycle. When the pin is high during one cycle and low in the next cycle, the count is incremented. The new count value appears in the register during the cycle following the one in which the transition was detected. Since it takes 2 machine cycles (4 CPU clocks) to recognize a 1-to-0 transition, the maximum count rate is 1/4 of the CPU clock frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle.

The "Timer" or "Counter" function is selected by control bit  $T0C/\bar{T}$  in the Special Function Register TMOD. Timer 0 and Timer 1 of the **P89LPC902** and **P89LPC903**, and Timer 1 of the **P89LPC901** have four operating modes (modes 0, 1, 2, and 3), which are selected by bit-pairs (TnM1, TnM0) in TMOD. Modes 0, 1, 2 and 3 are the same for both Timers. Mode 3 is different. The operating modes are described later in this section. In addition to these modes, Timer 0 of the **P89LPC901** has mode 6. Additionally the T0M2 mode bit in TAMOD is used to specify modes with Timer 0 of the **P89LPC901**.

TMOD																		
Address: 89h																		
Not bit addressable																		
Reset Source(s): Any source																		
Reset Value: 0000000B																		
		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">T1M1</td> <td style="text-align: center;">T1M0</td> <td style="text-align: center;">-</td> <td style="text-align: center;"><math>T0C/\bar{T}</math></td> <td style="text-align: center;">T0M1</td> <td style="text-align: center;">T0M0</td> </tr> </table>	7	6	5	4	3	2	1	0	-	-	T1M1	T1M0	-	$T0C/\bar{T}$	T0M1	T0M0
7	6	5	4	3	2	1	0											
-	-	T1M1	T1M0	-	$T0C/\bar{T}$	T0M1	T0M0											
BIT	SYMBOL	FUNCTION																
TMOD.7	-	Reserved.																
TMOD.6	-	Reserved.																
TMOD.5, 4	T1M1,T1M0	Mode Select for Timer 1. These bits are used to determine the Timer 1 mode (see Figure 5-2).																
TMOD.3	-	Reserved.																
TMOD.2	$T0C/\bar{T}$	Timer or Counter Selector for Timer 0. Cleared for Timer operation (input from CCLK). Set for Counter operation (input from T0 input pin). <b>P89LPC901</b> . When writing to this register on the <b>P89LPC902</b> or <b>P89LPC903</b> devices, this bit position should be written with a zero.																
TMOD.1, 0	T0M1,T0M0	Mode Select for Timer 0. These bits are used to determine the Timer 0 mode (see Figure 5-2). On the <b>P89LPC901</b> these bits are used with the T0M2 bit in the TAMOD register to determine the Timer 0 mode (see Figure 5-2).																

Figure 5-1: Timer/Counter Mode Control register (TMOD)

# TIMERS 0 AND 1

P89LPC901/902/903

<b>TAMOD - P89LPC901</b>		7	6	5	4	3	2	1	0
Address: 8Fh		-	-	-	-	-	-	-	T0M2
Not bit addressable									
Reset Source(s): Any reset									
Reset Value: xxx0xxx0B									
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>							
TAMOD.7-1	-	Reserved for future use. Should not be set to 1 by user programs.							
TAMOD.0	T0M2	Mode Select bit 2 for Timer 0. Used with T0M1 and T0M0 in the TMOD register to determine Timer 0 mode ( <b>P89LPC901</b> ).							
	<u>TnM2-TnM0</u>	<u>Timer Mode</u>							
	0 0 0	8048 Timer "TLn" serves as 5-bit prescaler. (Mode 0)							
	0 0 1	16-bit Timer/Counter "THn" and "TLn" are cascaded; there is no prescaler. (Mode 1)							
	0 1 0	8-bit auto-reload Timer/Counter. THn holds a value which is loaded into TLn when it overflows. (Mode 2)							
	0 1 1	Timer 0 is a dual 8-bit Timer/Counter in this mode. TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only, controlled by the Timer 1 control bits (see text). Timer 1 in this mode is stopped. (Mode 3)							
	1 0 0	Reserved. User must not configure to this mode.							
	1 0 1	Reserved. User must not configure to this mode.							
	1 1 0	PWM mode (see section "Mode 6 - P89LPC901").							
	1 1 1	Reserved. User must not configure to this mode.							

Figure 5-2: Timer/Counter Auxiliary Mode Control register (TAMOD)

## Mode 0

Putting either Timer into Mode 0 makes it look like an 8048 Timer, which is an 8-bit Counter with a divide-by-32 prescaler. Figure 5-4 shows Mode 0 operation.

In this mode, the Timer register is configured as a 13-bit register. As the count rolls over from all 1s to all 0s, it sets the Timer interrupt flag TF<sub>n</sub>. The count input is enabled to the Timer when TR<sub>n</sub> = 1. TR<sub>n</sub> is a control bit in the Special Function Register TCON (Figure 5-3).

The 13-bit register consists of all 8 bits of TH<sub>n</sub> and the lower 5 bits of TL<sub>n</sub>. The upper 3 bits of TL<sub>n</sub> are indeterminate and should be ignored. Setting the run flag (TR<sub>n</sub>) does not clear the registers.

Mode 0 operation is the same for Timer 0 and Timer 1. See Figure 5-4.

Mode 1 is the same as Mode 0, except that all 16 bits of the timer register (TH<sub>n</sub> and TL<sub>n</sub>) are used. See Figure 5-5.

## Mode 2

Mode 2 configures the Timer register as an 8-bit Counter (TL<sub>n</sub>) with automatic reload, as shown in Figure 5-6. Overflow from TL<sub>n</sub> not only sets TF<sub>n</sub>, but also reloads TL<sub>n</sub> with the contents of TH<sub>n</sub>, which must be preset by software. The reload leaves TH<sub>n</sub> unchanged. Mode 2 operation is the same for Timer 0 and Timer 1.

# TIMERS 0 AND 1

P89LPC901/902/903

## Mode 3

When Timer 1 is in Mode 3 it is stopped. The effect is the same as setting TR1 = 0.

Timer 0 in Mode 3 establishes TL0 and TH0 as two separate 8-bit counters. The logic for Mode 3 on Timer 0 is shown in Figure 5-7. TL0 uses the Timer 0 control bits: TR0 and TF0. TH0 is locked into a timer function (counting machine cycles) and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the "Timer 1" interrupt.

Mode 3 is provided for applications that require an extra 8-bit timer.

Note: When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it into and out of its own Mode 3. It can still be used by the serial port as a baud rate generator (**P89LPC903** only), or in any application not requiring an interrupt.

## Mode 6 - P89LPC901

In this mode, Timer 0 can be changed to a PWM with a full period of 256 timer clocks (see Figure 5-8). Its structure is similar to mode 2, except that:

- TF0 is set and cleared in hardware;
- The low period of the TF0 is in TH0, and should be between 1 and 254, and;
- The high period of the TF0 is always 256-TH0.
- Loading TH0 with 00h will force the T0 pin high, loading TH0 with FFh will force the T0 pin low.

Note that an interrupt can still be enabled on the low to high transition of TF0, and that TF0 can still be cleared in software as in any other modes.

<b>TCON</b>		7	6	5	4	3	2	1	0
Address: 88h		TF1	TR1	TF0	TR0	-	-	-	-
Bit addressable									
Reset Source(s): Any reset									
Reset Value: 00000000B									
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>							
TCON.7	TF1	Timer 1 overflow flag. Set by hardware on Timer overflow. Cleared by hardware when the interrupt is processed, or by software.							
TCON.6	TR1	Timer 1 Run control bit. Set/cleared by software to turn Timer 1 on/off.							
TCON.5	TF0	Timer 0 overflow flag. Set by hardware on Timer/Counter overflow. Cleared by hardware when the processor vectors to the interrupt routine, or by software. (except in mode 6, see above, when it is cleared in hardware)							
TCON.4	TR0	Timer 0 Run control bit. Set/cleared by software to turn Timer/Counter 0 on/off.							
TCON.3	-	Reserved for future use. Should not be set to 1 by user programs.							
TCON.2	-	Reserved for future use. Should not be set to 1 by user programs.							
TCON.1	-	Reserved for future use. Should not be set to 1 by user programs.							
TCON.0	-	Reserved for future use. Should not be set to 1 by user programs.							

Figure 5-3: Timer/Counter Control register (TCON)

# TIMERS 0 AND 1

## P89LPC901/902/903

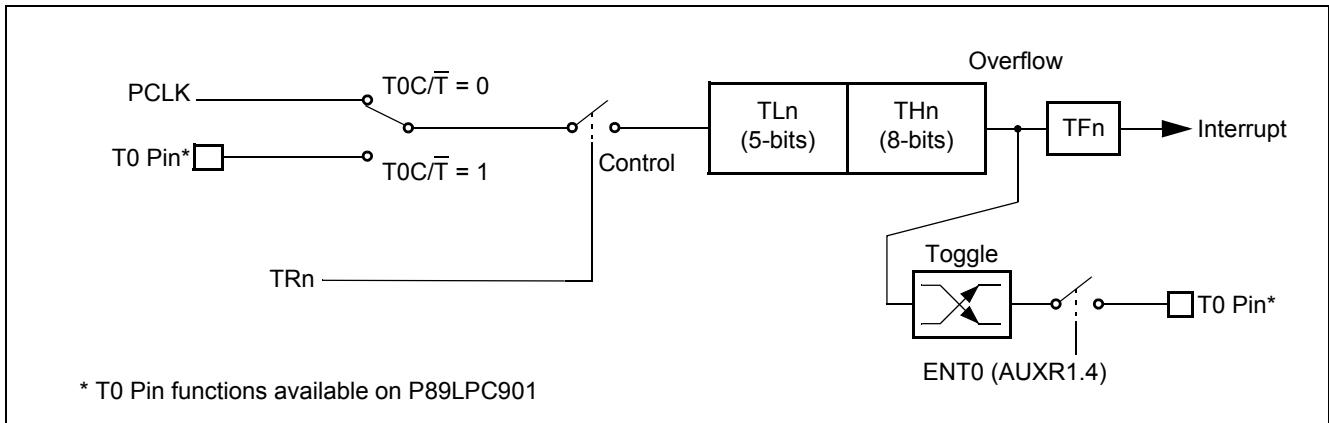


Figure 5-4: Timer/Counter 0 or 1 in Mode 0 (13-bit counter)

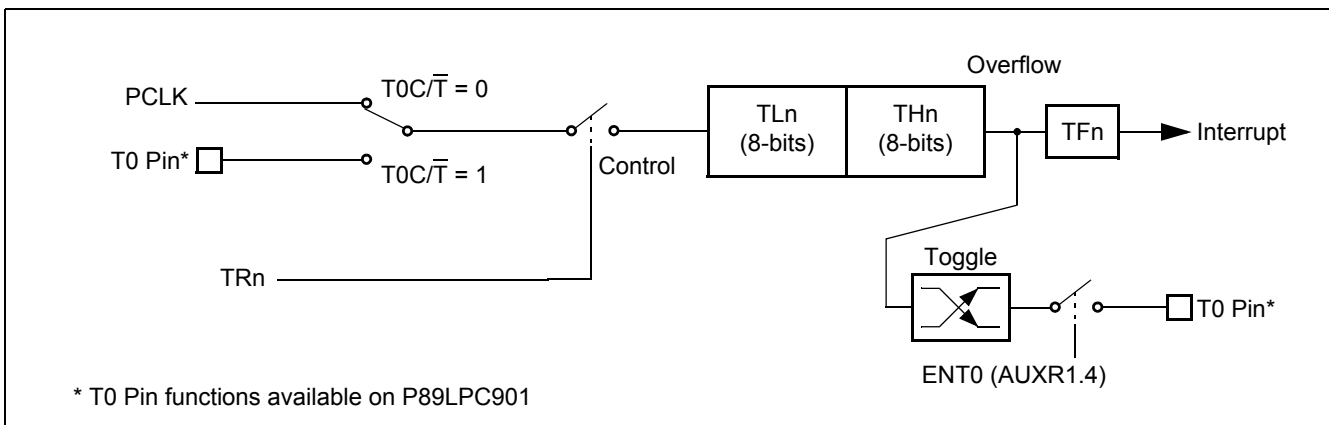


Figure 5-5: Timer/Counter 0 or 1 in Mode 1 (16-bit counter)

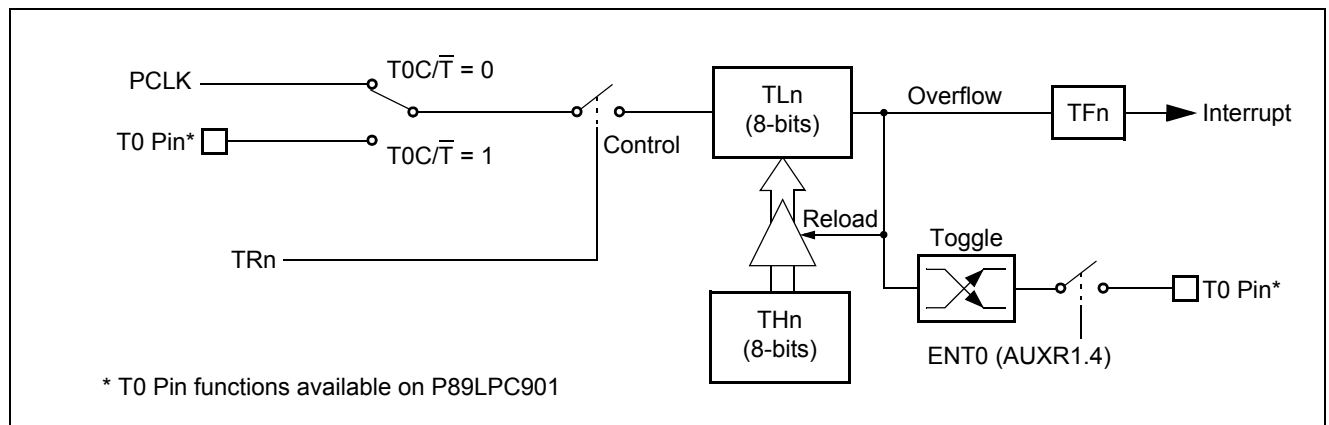


Figure 5-6: Timer/Counter 0 or 1 in Mode 2 (8-bit auto-reload)



# TIMERS 0 AND 1

## P89LPC901/902/903

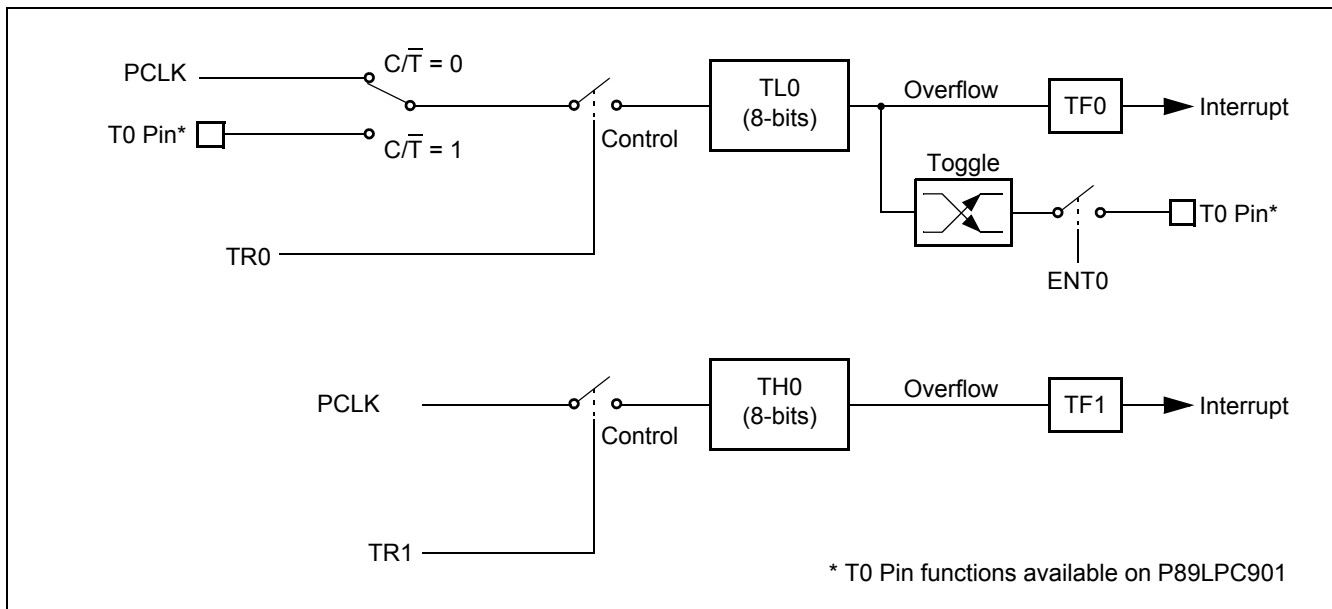


Figure 5-7: Timer/Counter 0 Mode 3 (two 8-bit counters)

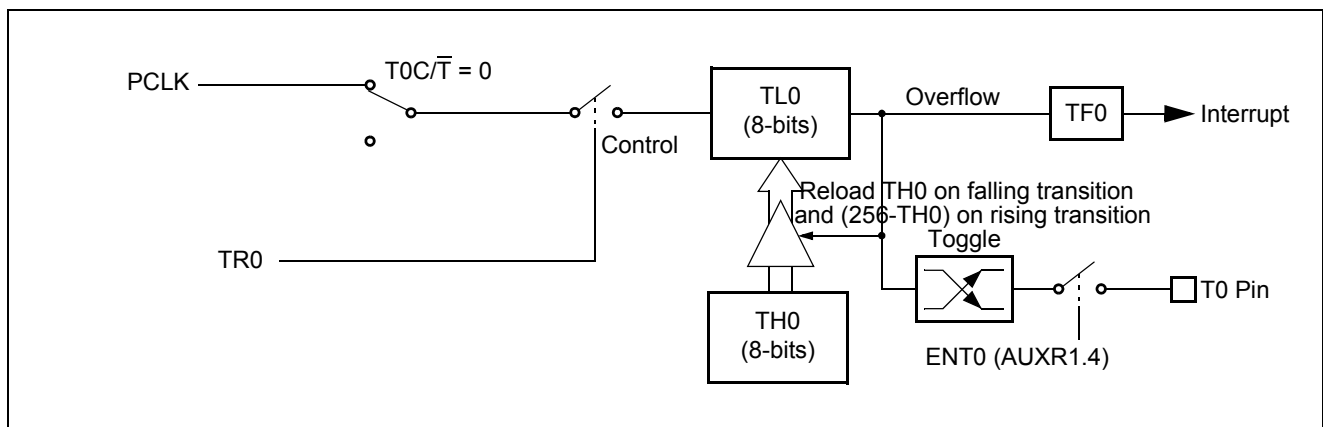


Figure 5-8: Timer/Counter 0 in Mode 6 (PWM auto-reload), P89LPC901.

### Timer Overflow toggle output - P89LPC901

Timer 0 can be configured to automatically toggle the T0 pin whenever the timer overflow occurs. This function is enabled by control bit ENT0 in the AUXR1 register. The port output will be a logic 1 prior to the first timer overflow when this mode is turned on. In order for this mode to function, the T0C/T-bar bit must be cleared selecting PCLK as the clock source for the timer.

---

# TIMERS 0 AND 1

P89LPC901/902/903

---

# REAL-TIME CLOCK/SYSTEM TIMER

P89LPC901/902/903

## 6. REAL-TIME CLOCK/SYSTEM TIMER

The **P89LPC901/902/903** has a simple Real-time clock/system timer that allows a user to continue running an accurate timer while the rest of the device is powered down. The Real-time clock can be an interrupt or a wake-up source (see Figure 6-1). The Real-time clock is a 23-bit down counter.

### Real-time Clock Source

On the **P89LPC901** the clock source for this counter can be either CCLK or the XTAL1-2 oscillator (XCLK). On the **P89LPC902** and **P89LPC903** devices the clock source for this counter is CCLK. Please refer to Figure 2-3 "Block Diagram of Oscillator Control - P89LPC901" in section "Clocks" on page 27. CCLK can have either the XTAL1-2 oscillator, the internal RC oscillator, or the Watchdog oscillator as its clock source. If the XTAL1-2 oscillator is used for producing CCLK, the RTC will use either the XTAL1-2 oscillator's output or CCLK as its clock source. The possible clocking combinations are shown in Table , below.

There are three SFRs used for the RTC:

- RTCCON - Real-time clock control.
- RTCH - Real-time clock counter reload high (bits 22-15).
- RTCL - Real-time clock counter reload low (bits 14-7).

The Real-time clock/system timer can be enabled by setting the RTCEN (RTCCON.0) bit. The Real-time clock is a 23-bit down counter (initialized to all 0's when RTCEN = 0) that is comprised of a 7-bit prescaler and a 16-bit loadable down counter. When RTCEN is written with '1', the counter is first loaded with (RTCH,RTCL,'1111111') and will count down. When it reaches all 0's, the counter will be reloaded again with (RTCH,RTCL,'1111111') and a flag - RTCF (RTCCON.7) - will be set.

Any write to RTCH and RTCL in-between the Real-time clock reloading will not cause reloading of the counter. When the current count terminates, the contents of RTCH and RTCL will be loaded into the counter and the new count will begin. An immediate reload of the counter can be forced by clearing the RTCEN bit to '0' and then setting it back to '1' .

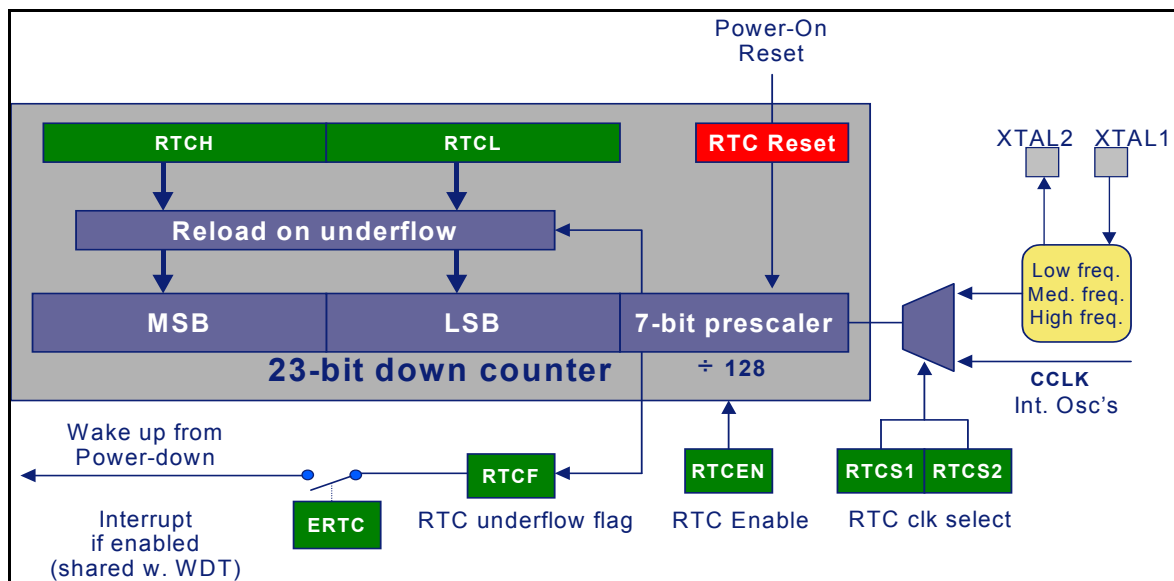


Figure 6-1: Real-time clock/system timer Block Diagram

**REAL-TIME CLOCK/SYSTEM TIMER****P89LPC901/902/903****Table 6-1: Real-time Clock/System Timer Clock Source - P89LPC901**

FOSC2 (UCFG1.2)	FOSC1 (UCFG1.1)	FOSC0 (UCFG1.0)	RTCS1:0	CCLK Frequency	RTC Clock Frequency
0	0	0	00	High frequency crystal/DIVM	High frequency crystal (XCLK)
			01		
			10		High frequency crystal/DIVM (CCLK)
			11		
0	0	1	00	Medium frequency crystal/DIVM	Medium frequency crystal (XCLK)
			01		
			10		Medium frequency crystal/ DIVM (CCLK)
			11		
0	1	0	00	Low frequency crystal/DIVM	Low frequency crystal (XCLK)
			01		
			10		Low frequency crystal/DIVM (CCLK)
			11		
0	1	1	00	RC Oscillator/DIVM	High frequency crystal (XCLK)
			01		Medium frequency crystal (XCLK)
			10		Low frequency crystal (XCLK)
			11		RC Oscillator/DIVM (CCLK)
1	0	0	00	WDT Oscillator/DIVM	High frequency crystal (XCLK)
			01		Medium frequency crystal (XCLK)
			10		Low frequency crystal (XCLK)
			11		WDT Oscillator/DIVM (CCLK)
1	0	1	xx	undefined	
1	1	0			
1	1	1	00	external clock/DIVM	external clock (XCLK)
			01		
			10		external clock/DIVM (CCLK)
			11		

**REAL-TIME CLOCK/SYSTEM TIMER****P89LPC901/902/903****Table 6-2: :Real-time Clock/System Timer Clock Source - P89LPC902/903**

FOSC2 (UCFG1.2)	FOSC1 (UCFG1.1)	FOSC0 (UCFG1.0)	RTCS1:0	CCLK Frequency	RTC Clock Frequency
0	0	0	x	undefined	
0	0	1			
0	1	0			
0	1	1	00	RC Oscillator/DIVM	undefined
			01		
			10		
			11		RC Oscillator/DIVM (CCLK)
1	0	0	00	WDT Oscillator/DIVM	undefined
			01		
			10		
			11		WDT Oscillator/DIVM (CCLK)
1	0	1	xx	undefined	
1	1	0			
1	1	1			

**Changing RTCS1-0**

RTCS1-0 cannot be changed if the RTC is currently enabled (RTCCON.0 =1). Setting RTCEN and updating RTCS1-0 may be done in a single write to RTCCON. However, if RTCEN = 1, this bit must first be cleared before updating RTCS1-0

**Real-time Clock Interrupt/Wake Up**

If ERTC (RTCCON.1), EWDRT (IEN0.6) and EA (IEN0.7) are set to '1', RTCF can be used as an interrupt source. This interrupt vector is shared with the watchdog timer. It can also be a source to wake up the device.

**Reset Sources Affecting the Real-time Clock**

Only power-on reset will reset the Real-time Clock and its associated SFRs to their default state.

# REAL-TIME CLOCK/SYSTEM TIMER

P89LPC901/902/903

<b>RTCCON</b>																		
Address: D1h																		
Not bit addressable																		
Reset Source(s): Power-up only																		
Reset Value: 011xxx00B																		
		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">RTCF</td> <td style="text-align: center;">RTCS1</td> <td style="text-align: center;">RTCS0</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">ERTC</td> <td style="text-align: center;">RTCEN</td> </tr> </table>	7	6	5	4	3	2	1	0	RTCF	RTCS1	RTCS0	-	-	-	ERTC	RTCEN
7	6	5	4	3	2	1	0											
RTCF	RTCS1	RTCS0	-	-	-	ERTC	RTCEN											
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>																
RTCCON.7	RTCF	Real-time Clock Flag. This bit is set to '1' when the 23-bit Real-time clock reaches a count of '0'. It can be cleared in software.																
RTCCON.6-5	RTCS1-0	Real-time Clock source select (see Table ,Table ).																
RTCCON.4-2	-	Reserved for future use. Should not be set to 1 by user programs.																
RTCCON.1	ERTC	Real-time Clock interrupt enable. The Real-time clock shares the same interrupt as the watchdog timer. Note that if the user configuration bit WDTE (UCFG1.7) is '0', the watchdog timer can be enabled to generate an interrupt. Users can read the RTCF (RTCCON.7) bit to determine whether the Real-time clock caused the interrupt.																
RTCCON.0	RTCEN	Real-time Clock enable. The Real-time clock will be enabled if this bit is '1'. Note that this bit will not Power down the Real-time Clock. The RTCPD bit (PCONA.7) if set, will Power down and disable this block regardless of RTCEN.																

**Figure 6-2: RTCCON Register**

# POWER MONITORING FUNCTIONS

P89LPC901/902/903

## 7. POWER MONITORING FUNCTIONS

The **P89LPC901/902/903** incorporates power monitoring functions designed to prevent incorrect operation during initial power-on and power loss or reduction during operation. This is accomplished with two hardware functions: Power-on Detect and Brownout Detect.

### Brownout Detection

The Brownout Detect function determines if the power supply voltage drops below a certain level. The default operation for a Brownout Detection is to cause a processor reset. However, it may alternatively be configured to generate an interrupt by setting the BOI (PCON.4) bit and the EBO (IEN0.5) bit.

Enabling and disabling of Brownout Detection is done via the BOPD (PCON.5) bit, bit field PMOD1-0 (PCON.1-0) and user configuration bit BOE (UCFG1.5). If BOE is in an unprogrammed state, brownout is disabled regardless of PMOD1-0 and BOPD. If BOE is in a programmed state, PMOD1-0 and BOPD will be used to determine whether Brownout Detect will be disabled or enabled. PMOD1-0 is used to select the power reduction mode. If PMOD1-0 = '11', the circuitry for the Brownout Detection is disabled for lowest power consumption. BOPD defaults to '0', indicating brownout detection is enabled on power-on if BOE is programmed.

If Brownout Detection is enabled, the operating voltage range for  $V_{DD}$  is 2.7V-3.6V, and the brownout condition occurs when  $V_{DD}$  falls below the Brownout trip voltage,  $V_{BO}$  (see D.C. Electrical Characteristics), and is negated when  $V_{DD}$  rises above  $V_{BO}$ . If Brownout Detection is disabled, the operating voltage range for  $V_{DD}$  is 2.4V-3.6V. If the **P89LPC901/902/903** device is to operate with a power supply that can be below 2.7V, BOE should be left in the unprogrammed state so that the device can operate at 2.4V, otherwise continuous brownout reset may prevent the device from operating.

If Brownout Detect is enabled (BOE programmed, PMOD1-0  $\neq$  '11', BOPD = 0), BOF (RSTSRC.5) will be set when a brownout is detected, regardless of whether a reset or an interrupt is enabled. BOF will stay set until it is cleared in software by writing '0' to the bit. Note that if BOE is unprogrammed, BOF is meaningless. If BOE is programmed, and a initial power-on occurs, BOF will be set in addition to the power-on flag (POF - RSTSRC.4).

For correct activation of Brownout Detect, certain  $V_{DD}$  rise and fall times must be observed. Please see the datasheet for specifications.

# POWER MONITORING FUNCTIONS

**P89LPC901/902/903**

BOE (UCFG1.5)	PMOD1-0 (PCON.1-0)	BOPD (PCON.5)	BOI (PCON.4)	EBO (IEN0.5)	EA (IEN0.7)	Description		
0 (erased)	XX	X	X	X	X			
1 (programmed)	11 (total power down)	X	X	X	X	Brownout disabled. $V_{DD}$ operating range is 2.4V-3.6V.		
	≠ 11 (any mode other than total power down)	1 (brownout detect powered down)	X	X	X	Brownout disabled. $V_{DD}$ operating range is 2.4V-3.6V. However, BOPD is default to '0' upon power-up.		
		0 (brownout detect active)	0 (brownout detect generates reset)		X	X	Brownout reset enabled. $V_{DD}$ operating range is 2.7V-3.6V. Upon a brownout reset, BOF (RSTSRC.5) will be set to indicate the reset source. BOF can be cleared by writing '0' to the bit.	
			1 (brownout detect generates an interrupt)	1 (enable brownout interrupt)	1 (global interrupt enable)	1 (global interrupt enable)	1 (global interrupt enable)	Brownout interrupt enabled. $V_{DD}$ operating range is 2.7V-3.6V. Upon a brownout interrupt, BOF (RSTSRC.5) will be set. BOF can be cleared by writing '0' to the bit.
				0	X	X	X	Both brownout reset and interrupt disabled. $V_{DD}$ operating range is 2.4V-3.6V. However, BOF (RSTSRC.5) will be set when $V_{DD}$ falls to the Brownout Detection trip point. BOF can be cleared by writing '0' to the bit.
				0	0			

**Table 7-1: Brownout Options.**

## Power-On Detection

The Power-On Detect has a function similar to the Brownout Detect, but is designed to work as power initially comes up, before the power supply voltage reaches a level where the Brownout Detect can function. The POF flag (RSTSRC.4) is set to indicate an initial power-on condition. The POF flag will remain set until cleared by software by writing '0' to the bit. Note that if BOE (UCFG1.5) is programmed, BOF (RSTSRC.5) will be set when POF is set. If BOE is unprogrammed, BOF is meaningless.



# POWER MONITORING FUNCTIONS

**P89LPC901/902/903**

## Power Reduction Modes

The **P89LPC901/902/903** supports three different power reduction modes as determined by SFR bits PCON.1-0 (see Table 7-2):

PMOD1 (PCON.1)	PMOD0 (PCON.0)	Description
0	0	Normal Mode (Default) - no power reduction.
0	1	Idle Mode. The Idle mode leaves peripherals running in order to allow them to activate the processor when an interrupt is generated. Any enabled interrupt source or reset may terminate Idle mode.
1	0	<p>Power down mode: The Power down mode stops the oscillator in order to minimize power consumption. The <b>P89LPC901/902/903</b> exits Power down mode via any reset, or certain interrupts - brownout Interrupt, keyboard, Real-time clock (system timer), watchdog, and comparator trips. Waking up by reset is only enabled if the corresponding reset is enabled, and waking up by interrupt is only enabled if the corresponding interrupt is enabled and the EA SFR bit (IEN0.7) is set.</p> <p>In Power down mode the internal RC oscillator is disabled unless both the RC oscillator has been selected as the system clock AND the RTC is enabled</p> <p>In Power down mode, the power supply voltage may be reduced to the RAM keep-alive voltage <math>V_{RAM}</math>. This retains the RAM contents at the point where Power down mode was entered. SFR contents are not guaranteed after <math>V_{DD}</math> has been lowered to <math>V_{RAM}</math>, therefore it is recommended to wake up the processor via Reset in this situation. <math>V_{DD}</math> must be raised to within the operating range before the Power down mode is exited.</p> <p>When the processor wakes up from Power down mode, it will start the oscillator immediately and begin execution when the oscillator is stable. Oscillator stability is determined by counting 1024 CPU clocks after start-up when one of the crystal oscillator configurations is used, or 256 clocks after start-up for the internal RC or external clock input configurations.</p> <p>Some chip functions continue to operate and draw power during Power down mode, increasing the total power used during Power down. These include:</p> <ul style="list-style-type: none"> <li>• Brownout Detect</li> <li>• Watchdog Timer if WDCLK (WDCON.0) is '1'.</li> <li>• Comparator (Note: Comparator can be powered down separately with PCONA.5 set to '1' and comparator disabled);</li> <li>• Real-time Clock/System Timer (and the crystal oscillator circuitry if this block is using it, unless RTCPD, i.e., PCONA.7 is '1').</li> </ul>
1	1	<p>Total Power down mode: This is the same as Power down mode except that the Brownout Detection circuitry and the voltage comparators are also disabled to conserve additional power. Note that a brownout reset or interrupt will not occur. Voltage comparator interrupts and Brownout interrupt cannot be used as a wakeup source. The internal RC oscillator is disabled unless both the RC oscillator has been selected as the system clock AND the RTC is enabled.</p> <p>The following are the wakeup options supported:</p> <ul style="list-style-type: none"> <li>• Watchdog Timer if WDCLK (WDCON.0) is '1'. Could generate Interrupt or Reset, either one can wake up the device</li> <li>• Keyboard Interrupt</li> <li>• Real-time Clock/System Timer (and the crystal oscillator circuitry if this block is using it, unless RTCPD, i.e., PCONA.7 is '1').</li> <li>• Note: Using the internal RC-oscillator to clock the RTC during Power down may result in relatively high power consumption. Lower power consumption can be achieved by using an external low frequency clock when the Real-time Clock is running during Power down.</li> </ul>

**Table 7-2: Power Reduction Modes.**

# POWER MONITORING FUNCTIONS

P89LPC901/902/903

<b>PCON</b>		7	6	5	4	3	2	1	0
Address: 87h		SMOD1	SMOD0	BOPD	BOI	GF1	GF0	PMOD1	PMOD0
Not bit addressable									
Reset Source(s): Any reset									
Reset Value: 00000000B									
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>							
PCON.7	SMOD1	Double Baud Rate bit for the serial port (UART) when Timer 1 is used as the baud rate source. When 1, the Timer 1 overflow rate is supplied to the UART. When 0, the Timer 1 overflow rate is divided by 2 before being supplied to the UART. (See Figure 8-2).							
PCON.6	SMOD0	Framing Error Location: -When 0, bit 7 of SCON is accessed as SM0 for the UART. -When 1, bit 7 of SCON is accessed as the framing error status (FE) for the UART. This bit also determines the location of the UART receiver interrupt RI (see description on RI in Figure 8-3).							
PCON.5	BOPD	Brownout Detect Power down. When 1, Brownout Detect is powered down and therefore disabled. When 0, Brownout Detect is enabled. (Note: BOPD must be '0' before any programming or erasing commands can be issued. Otherwise these commands will be aborted.)							
PCON.4	BOI	Brownout Detect Interrupt Enable. When 1, Brownout Detection will generate a interrupt . When 0, Brownout Detection will cause a reset.							
PCON.3	GF1	General Purpose Flag 1. May be read or written by user software, but has no effect on operation.							
PCON.2	GF0	General Purpose Flag 0. May be read or written by user software, but has no effect on operation.							
PCON.1-0	PMOD1-PMOD0	Power Reduction Mode (see section "Power Reduction Modes").							

**Figure 7-1: Power Control Register (PCON)**

# POWER MONITORING FUNCTIONS

P89LPC901/902/903

<b>PCONA</b>																		
Address: B5H																		
Not bit addressable																		
Reset Source(s): Any reset																		
Reset Value: 0000000B																		
		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">RTCPD</td> <td style="text-align: center;">-</td> <td style="text-align: center;">VCPD</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">SPD</td> <td style="text-align: center;">-</td> </tr> </table>	7	6	5	4	3	2	1	0	RTCPD	-	VCPD	-	-	-	SPD	-
7	6	5	4	3	2	1	0											
RTCPD	-	VCPD	-	-	-	SPD	-											
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>																
PCONA.7	RTCPD	Real-time Clock Power down: When '1', the internal clock to the Real-time Clock is disabled.																
PCONA.6	-	Not used. Reserved for future use.																
PCONA.5	VCPD	Analog Voltage Comparator Power down: When '1', the voltage comparator is powered down. User must disable the voltage comparator prior to setting this bit.																
PCONA.4	-	Not used. Reserved for future use.																
PCONA.3	-	Not used. Reserved for future use.																
PCONA.2	-	Not used. Reserved for future use.																
PCONA.1	SPD	Serial Port (UART) Power down: When '1', the internal clock to the UART is disabled. Note that in either Power down mode or Total Power down mode, the UART clock will be disabled regardless of this bit.																
PCONA.0	-	Not used. Reserved for future use.																

Figure 7-2: Power Control Register (PCONA)

---

# POWER MONITORING FUNCTIONS

---

P89LPC901/902/903

## UART (P89LPC903)

P89LPC901/902/903

### 8. UART (P89LPC903)

The **P89LPC903** has an enhanced UART that is compatible with the conventional 80C51 UART except that Timer 2 overflow cannot be used as a baud rate source. The **P89LPC903** does include an independent Baud Rate Generator. The baud rate can be selected from the oscillator (divided by a constant), Timer 1 overflow, or the independent Baud Rate Generator. In addition to the baud rate generation, enhancements over the standard 80C51 UART include Framing Error detection, break detect, automatic address recognition, selectable double buffering and several interrupt options.

The UART can be operated in 4 modes:

#### Mode 0

Serial data enters and exits through RxD. TxD outputs the shift clock. 8 bits are transmitted or received, LSB first. The baud rate is fixed at 1/16 of the CPU clock frequency.

#### Mode 1

10 bits are transmitted (through TxD) or received (through RxD): a start bit (logical 0), 8 data bits (LSB first), and a stop bit (logical 1). When data is received, the stop bit is stored in RB8 in Special Function Register SCON. The baud rate is variable and is determined by the Timer 1 overflow rate or the Baud Rate Generator (see "Baud Rate Generator and Selection" section).

#### Mode 2

11 bits are transmitted (through TxD) or received (through RxD): start bit (logical 0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (logical 1). When data is transmitted, the 9th data bit (TB8 in SCON) can be assigned the value of 0 or 1. Or, for example, the parity bit (P, in the PSW) could be moved into TB8. When data is received, the 9th data bit goes into RB8 in Special Function Register SCON and the stop bit is not saved. The baud rate is programmable to either 1/16 or 1/32 of the CCLK frequency, as determined by the SMOD1 bit in PCON.

#### Mode 3

11 bits are transmitted (through TxD) or received (through RxD): a start bit (logical 0), 8 data bits (LSB first), a programmable 9th data bit, and a stop bit (logical 1). Mode 3 is the same as Mode 2 in all respects except baud rate. The baud rate in Mode 3 is variable and is determined by the Timer 1 overflow rate or the Baud Rate Generator (see "Baud Rate Generator and Selection" section).

In all four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode 0 by the condition RI = 0 and REN = 1. Reception is initiated in the other modes by the incoming start bit if REN = 1.

# UART (P89LPC903)

P89LPC901/902/903

## SFR Space

The UART SFRs are at the following locations:

**Table 8-1: SFR Locations for UARTs**

Register	Description	SFR Location
PCON	Power Control	87H
SCON	Serial Port (UART) Control	98H
SBUF	Serial Port (UART) Data Buffer	99H
SADDR	Serial Port (UART) Address	A9H
SADEN	Serial Port (UART) Address Enable	B9H
SSTAT	Serial Port (UART) Status	BAH
BRGR1	Baud Rate Generator Rate High Byte	BFH
BRGR0	Baud Rate Generator Rate Low Byte	BEH
BRGCON	Baud Rate Generator Control	BDH

## Baud Rate Generator and Selection

The **P89LPC903** enhanced UART has an independent Baud Rate Generator. The baud rate is determined by a value programmed into the BRGR1 and BRGR0 SFRs. The UART can use either Timer 1 or the baud rate generator output as determined by BRGCON.2-1 (see Figure 8-2). Note that Timer T1 is further divided by 2 if the SMOD1 bit (PCON.7) is cleared. The independent Baud Rate Generator uses CCLK.

## Updating the BRGR1 and BRGR0 SFRs

The baud rate SFRs, BRGR1 and BRGR0 must only be loaded when the Baud Rate Generator is disabled (the BRGEN bit in the BRGCON register is '0'). This avoids the loading of an interim value to the baud rate generator. **(CAUTION: If either BRGR0 or BRGR1 is written when BRGEN = 1, the result is unpredictable.)**

**Table 8-2: Baud Rate Generation for UART**

SCON.7 (SM0)	SCON.6 (SM1)	PCON.7 (SMOD1)	BRGCON.1 (SBRGS)	Receive/Transmit Baud Rate for UART
0	0	X	X	CCLK/16
0	1	0	0	CCLK/(256-TH1)64
		1	0	CCLK/(256-TH1)32
		X	1	CCLK/((BRGR1,BRGR0)+16)
1	0	0	X	CCLK/32
		1	X	CCLK/16
1	1	0	0	CCLK/(256-TH1)64
		1	0	CCLK/(256-TH1)32
		X	1	CCLK/((BRGR1,BRGR0)+16)

# UART (P89LPC903)

P89LPC901/902/903

BRGCON																		
Address: BDh																		
Not bit addressable																		
Reset Source(s): Any reset																		
Reset Value: xxxxxx00B																		
		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">SBRGS</td> <td style="text-align: center;">BRGEN</td> </tr> </table>	7	6	5	4	3	2	1	0	-	-	-	-	-	-	SBRGS	BRGEN
7	6	5	4	3	2	1	0											
-	-	-	-	-	-	SBRGS	BRGEN											
BIT	SYMBOL	FUNCTION																
BRGCON.7-2	-	Reserved for future use. Should not be set to 1 by user programs.																
BRGCON.1	SBRGS	Select Baud Rate Generator as the source for baud rates to UART in modes 1 & 3 (see Table for details)																
BRGCON.0	BRGEN	Baud Rate Generator Enable. Enables the baud rate generator. BRGR1 and BRGR0 can only be written when BRGEN = 0.																

Figure 8-1: BRGCON Register

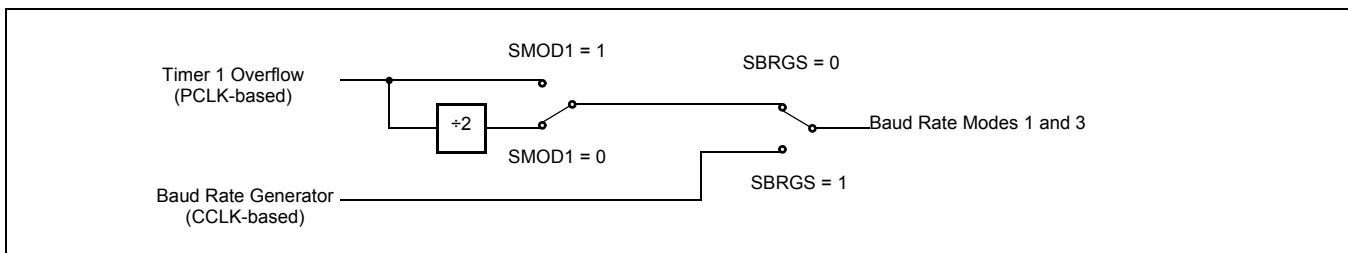


Figure 8-2: Baud Rate Generations for UART (Modes 1, 3)

## Framing Error

A Framing error occurs when the stop bit is sensed as a logic '0'. A Framing error is reported in the status register (SSTAT). In addition, if SMOD0 (PCON.6) is 1, framing errors can be made available in SCON.7. If SMOD0 is 0, SCON.7 is SM0. It is recommended that SM0 and SM1 (SCON.7-6) are programmed when SMOD0 is '0'.

## Break Detect

A break is detected when any 11 consecutive bits are sensed low. A break detect is reported in the status register (SSTAT). Since a break condition also satisfies the requirements for a framing error, a break condition will also result in reporting a framing error. Once a break condition has been detected, the UART will go into an idle state and remain in this idle state until a stop bit has been received. The break detect can be used to reset the device by setting the EBRR bit (AUXR1.6).

A break detect reset will force the high byte of the program counter to be equal to the Boot Vector contents and the low byte cleared to 00h. The first instruction will be fetched from this address.

# UART (P89LPC903)

P89LPC901/902/903

<b>SCON</b>								
Address: 98h								
Bit addressable								
Reset Source(s): Any reset								
Reset Value: 0000000B								
	7	6	5	4	3	2	1	0
	SM0/FE	SM1	SM2	REN	TB8	RB8	TI	RI
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>						
SCON.7	SM0/FE	The use of this bit is determined by SMOD0 in the PCON register. If SMOD0 = 0, this bit is read and written as SM0, which with SM1, defines the serial port mode. If SMOD0 = 1, this bit is read and written as FE (Framing Error). FE is set by the receiver when an invalid stop bit is detected. Once set, this bit cannot be cleared by valid frames but is cleared by software. (Note: UART mode bits SM0 and SM1 should be programmed when SMOD0 is '0' - default mode on any reset.)						
SCON.6	SM1	With SM0, defines the serial port mode (see table below).						
	<u>SM0, SM1</u>	<u>UART Mode</u>	<u>UART 0 Baud Rate</u>					
	0 0	0: shift register	CCLK/16 (default mode on any reset)					
	0 1	1: 8-bit UART	Variable (see Table )					
	1 0	2: 9-bit UART	CCLK/32 or CCLK/16					
	1 1	3: 9-bit UART	Variable (see Table )					
SCON.5	SM2	Enables the multiprocessor communication feature in Modes 2 and 3. In Mode 2 or 3, if SM2 is set to 1, then RI will not be activated if the received 9th data bit (RB8) is 0. In Mode 0, SM2 should be 0. In Mode 1, SM2 must be 0.						
SCON.4	REN	Enables serial reception. Set by software to enable reception. Clear by software to disable reception.						
SCON.3	TB8	The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.						
SCON.2	RB8	The 9th data bit that was received in Modes 2 and 3. In Mode 1 (SM2 must be 0), RB8 is the stop bit that was received. In Mode 0, RB8 is undefined.						
SCON.1	TI	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the the stop bit (see description of INTLO bit in SSTAT register) in the other modes. Must be cleared by software.						
SCON.0	RI	Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or approximately halfway through the stop bit time in Mode 1. For Mode 2 or Mode 3, if SMOD0, it is set near the middle of the 9th data bit (bit 8). If SMOD0 = 1, it is set near the middle of the stop bit (see SM2 - SCON.5 - for exceptions). Must be cleared by software.						

**Figure 8-3: Serial Port Control Register (SCON)**



# UART (P89LPC903)

P89LPC901/902/903

<b>SSTAT</b>																		
Address: BAh																		
Not bit addressable																		
Reset Source(s): Any reset																		
Reset Value: 0000000B																		
		<table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">6</td> <td style="text-align: center;">5</td> <td style="text-align: center;">4</td> <td style="text-align: center;">3</td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">DBMOD</td> <td style="text-align: center;">INTLO</td> <td style="text-align: center;">CIDIS</td> <td style="text-align: center;">DBISEL</td> <td style="text-align: center;">FE</td> <td style="text-align: center;">BR</td> <td style="text-align: center;">OE</td> <td style="text-align: center;">STINT</td> </tr> </table>	7	6	5	4	3	2	1	0	DBMOD	INTLO	CIDIS	DBISEL	FE	BR	OE	STINT
7	6	5	4	3	2	1	0											
DBMOD	INTLO	CIDIS	DBISEL	FE	BR	OE	STINT											
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>																
SSTAT.7	DBMOD	Double buffering mode. When set = 1 enables double buffering. Must be '0' for UART mode 0. In order to be compatible with existing 80C51 devices, this bit is reset to '0' to disable double buffering.																
SSTAT.6	INTLO	Transmit interrupt position. When cleared = 0, the Tx interrupt is issued at the beginning of the stop bit. When set =1, the Tx interrupt is issued at end of the stop bit. Must be '0' for mode 0. Note that in the case of single buffering, if the Tx interrupt occurs at the end of a STOP bit, a gap may exist before the next start bit.																
SSTAT.5	CIDIS	Combined Interrupt Disable. When set = 1, Rx and Tx interrupts are separate. When cleared = 0, the UART uses a combined Tx/Rx interrupt (like a conventional 80C51 UART). This bit is reset to '0' to select combined interrupts.																
SSTAT.4	DBISEL	Double buffering transmit interrupt select. Used only if double buffering is enabled. This bit controls the number of interrupts that can occur when double buffering is enabled. When set, one transmit interrupt is generated after each character written to SBUF, and there is also one more transmit interrupt generated at the beginning (INTLO = 0) or the end (INTLO = 1) of the STOP bit of the last character sent (i.e., no more data in buffer). This last interrupt can be used to indicate that all transmit operations are over. When cleared = 0, only one transmit interrupt is generated per character written to SBUF. Must be '0' when double buffering is disabled.  Note that except for the first character written (when buffer is empty), the location of the transmit interrupt is determined by INTLO. When the first character is written, the transmit interrupt is generated immediately after SBUF is written.																
SSTAT.3	FE	Framing error flag is set when the receiver fails to see a valid STOP bit at the end of the frame. Cleared by software.																
SSTAT.2	BR	Break Detect flag. A break is detected when any 11 consecutive bits are sensed low. Cleared by software.																
SSTAT.1	OE	Overrun Error flag is set if a new character is received in the receiver buffer while it is still full (before the software has read the previous character from the buffer), i.e., when bit 8 of a new byte is received while RI in SCON is still set. Cleared by software.																
SSTAT.0	STINT	Status Interrupt Enable. When set =1, FE, BR, or OE can cause an interrupt. The interrupt used (vector address 0023h) is shared with RI (CIDIS = 1) or the combined TI/RI (CIDIS = 0). When cleared = 0, FE, BR, OE cannot cause an interrupt. (Note: FE, BR, or OE is often accompanied by a RI, which will generate an interrupt regardless of the state of STINT). Note that BR can cause a break detect reset if EBRR (AUXR1.6) is set to '1'.																

**Figure 8-4: Serial Port Status Register (SSTAT)**

## More About UART Mode 0

In Mode 0, a write to SBUF will initiate a transmission. At the end of the transmission, TI (SCON.1) is set, which must be cleared in software. Double buffering must be disabled in this mode.

Reception is initiated by clearing RI (SCON.0). Synchronous serial transfer occurs and RI will be set again at the end of the transfer. When RI is cleared, the reception of the next character will begin. Refer to Figure 8-5 for timing.

# UART (P89LPC903)

P89LPC901/902/903

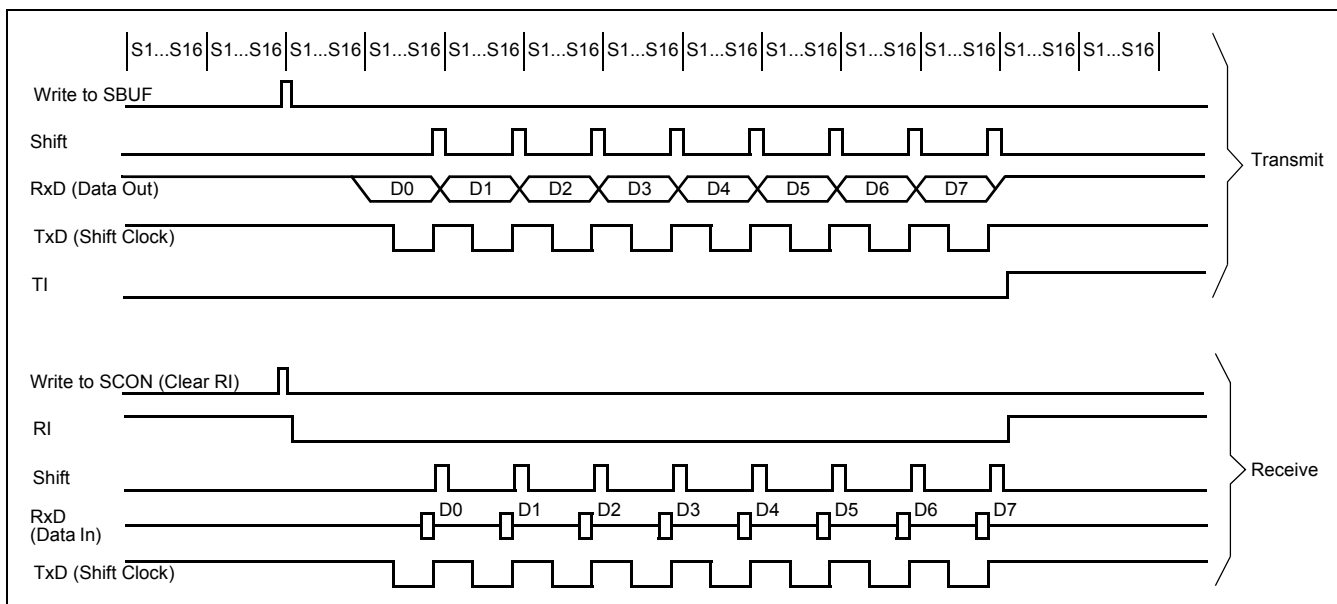


Figure 8-5: Serial Port Mode 0 (Double Buffering Must Be Disabled)

## More About UART Mode 1

Reception is initiated by detecting a 1-to-0 transition on RxD. RxD is sampled at a rate 16 times the programmed baud rate. When a transition is detected, the divide-by-16 counter is immediately reset. Each bit time is thus divided into 16 counter states. At the 7th, 8th, and 9th counter states, the bit detector samples the value of RxD. The value accepted is the value that was seen in at least 2 of the 3 samples. This is done for noise rejection. If the value accepted during the first bit time is not 0, the receive circuits are reset and the receiver goes back to looking for another 1-to-0 transition. This provides rejection of false start bits. If the start bit proves valid, it is shifted into the input shift register, and reception of the rest of the frame will proceed.

The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated: RI = 0 and either SM2=0 or the received stop bit =1. If either of these two conditions is not met, the received frame is lost. If both conditions are met, the stop bit goes into RB8, the 8 data bits go into SBUF, and RI is activated.

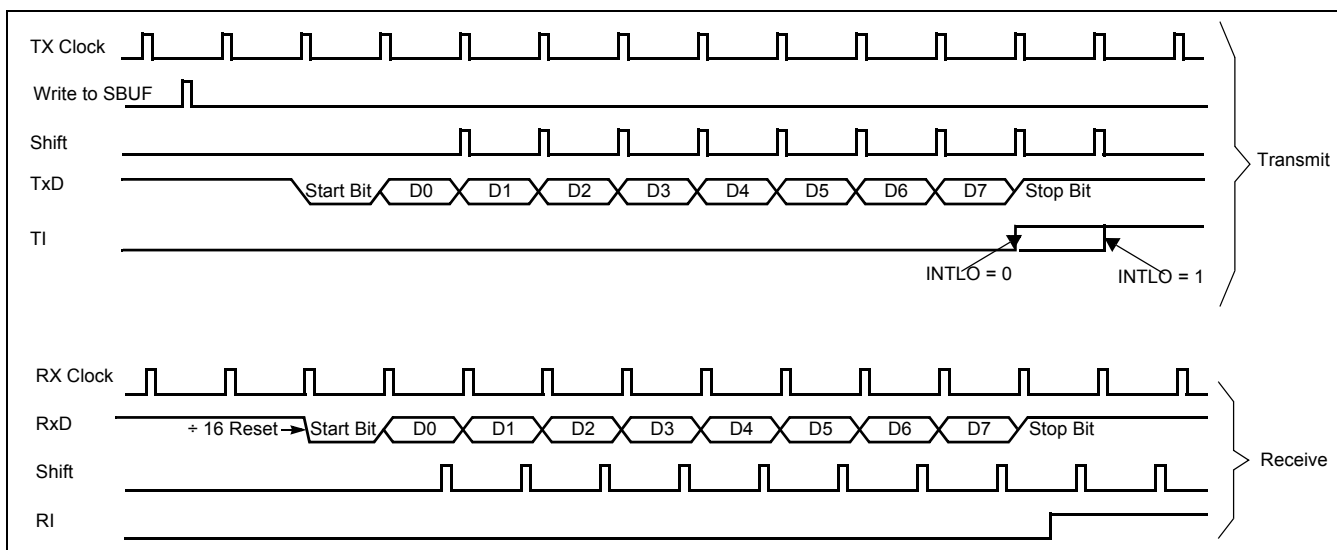


Figure 8-6: Serial Port Mode 1 (Only Single Transmit Buffering Case Is Shown)

# UART (P89LPC903)

P89LPC901/902/903

## More About UART Modes 2 and 3

Reception is the same as in Mode 1.

The signal to load SBUF and RB8, and to set RI, will be generated if, and only if, the following conditions are met at the time the final shift pulse is generated. (a) RI = 0, and (b) Either SM2 = 0, or the received 9th data bit = 1. If either of these conditions is not met, the received frame is lost, and RI is not set. If both conditions are met, the received 9th data bit goes into RB8, and the first 8 data bits go into SBUF.

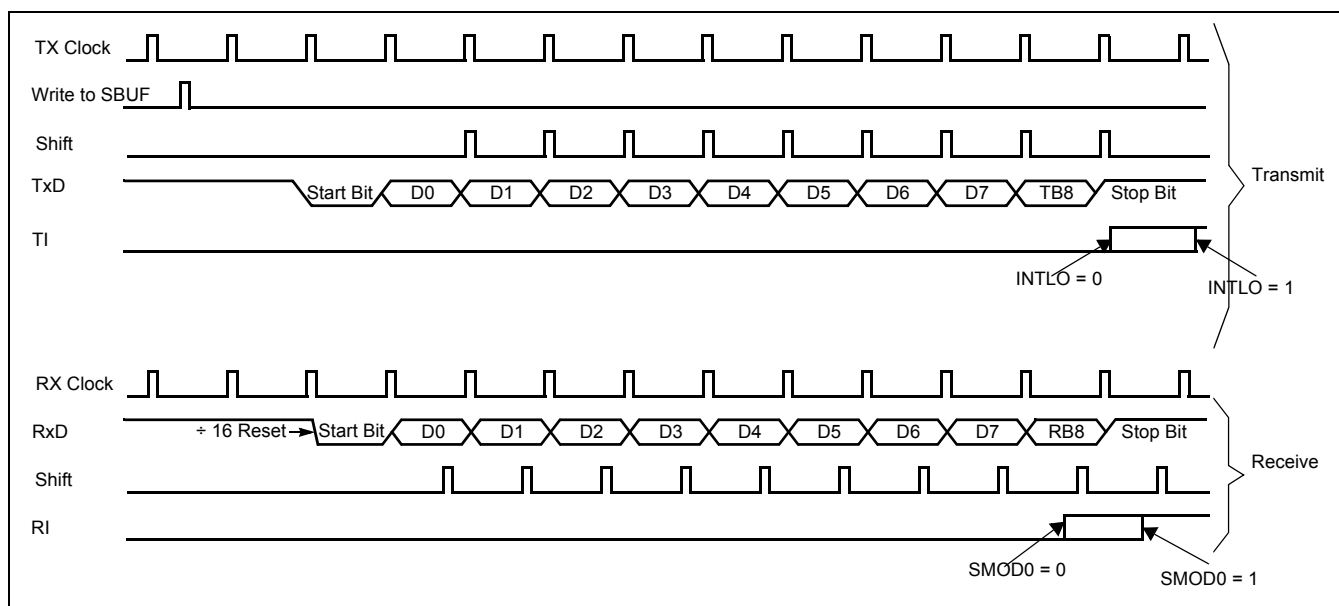


Figure 8-7: Serial Port Mode 2 or 3 (Only Single Transmit Buffering Case Is Shown)

## Framing Error and RI in Modes 2 and 3 with SM2 = 1

If SM2 = 1 in modes 2 and 3, RI and FE behave as in the following table.

Mode	PCON.6 (SMOD0)	RB8	RI	FE
2	0	0	No RI when RB8 = 0	Occurs during STOP bit
		1	Similar to Figure 8-7, with SMOD0 = 0, RI occurs during RB8, one bit before FE	Occurs during STOP bit
3	1	0	No RI when RB8 = 0	Will NOT occur
		1	Similar to Figure 8-7, with SMOD0 = 1, RI occurs during STOP bit	Occurs during STOP bit

Table 8-3: FE and RI when SM2 = 1 in Modes 2 and 3.

## Break Detect

A break is detected when 11 consecutive bits are sensed low and is reported in the status register (SSTAT). For Mode 1, this consists of the start bit, 8 data bits, and two stop bit times. For Modes 2 & 3, this consists of the start bit, 9 data bits, and one stop bit. The break detect bit is cleared in software or by a reset. The break detect can be used to reset the device. This occurs if the UART is enabled and the the EBRR bit (AUXR1.6) is set and a break occurs.

# UART (P89LPC903)

P89LPC901/902/903

## Double Buffering

The UART has a transmit double buffer that allows buffering of the next character to be written to SBUF while the first character is being transmitted. Double buffering allows transmission of a string of characters with only one stop bit between any two characters, provided the next character is written between the start bit and the stop bit of the previous character.

Double buffering can be disabled. If disabled (DBMOD, i.e. SSTAT.7 = 0), the UART is compatible with the conventional 80C51 UART. If enabled, the UART allows writing to SBUF while the previous data is being shifted out.

## Double Buffering in Different Modes

Double buffering is only allowed in Modes 1, 2 and 3. When operated in Mode 0, double buffering must be disabled (DBMOD = 0).

## Transmit Interrupts with Double Buffering Enabled (Modes 1, 2 and 3)

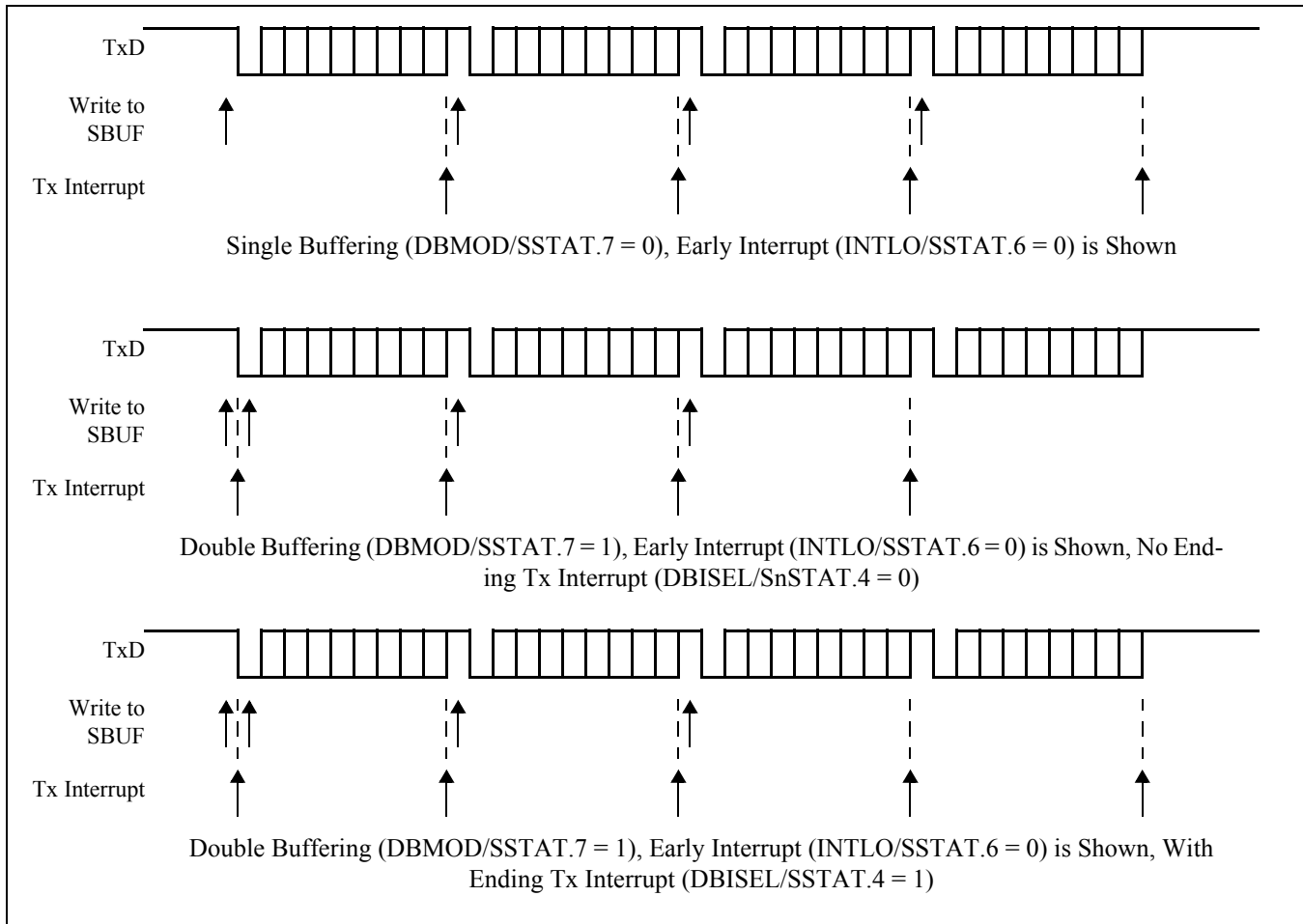
Unlike the conventional UART, when double buffering is enabled, the Tx interrupt is generated when the double buffer is ready to receive new data. The following occurs during a transmission (assuming eight data bits):

1. The double buffer is empty initially.
  2. The CPU writes to SBUF.
  3. The SBUF data is loaded to the shift register and a Tx interrupt is generated immediately.
  4. If there is more data, go to 6, else continue on 5.
  5. If there is no more data, then:
    - If DBISEL is '0', no more interrupts will occur.
    - If DBISEL is '1' and INTLO is '0', a Tx interrupt will occur at the beginning of the STOP bit of the data currently in the shifter (which is also the last data).
    - If DBISEL is '1' and INTLO is '1', a Tx interrupt will occur at the end of the STOP bit of the data currently in the shifter (which is also the last data).
  6. If there is more data, the CPU writes to SBUF again. Then:
    - If INTLO is '0', the new data will be loaded and a Tx interrupt will occur at the beginning of the STOP bit of the data currently in the shifter.
    - If INTLO is '1', the new data will be loaded and a Tx interrupt will occur at the end of the STOP bit of the data currently in the shifter.
- Go to 3.

Note that if DBISEL is '1' and the CPU is writing to SBUF when the STOP bit of the last data is shifted out, there can be an uncertainty of whether a Tx interrupt is generated already with the UART not knowing whether there is any more data following.

# UART (P89LPC903)

P89LPC901/902/903



**Figure 8-8: Transmission with and without Double Buffering**

## The 9th Bit (Bit 8) in Double Buffering (Modes 1, 2 and 3)

If double buffering is disabled (DBMOD, i.e. SSTAT.7 = 0), TB8 can be written before or after SBUF is written, provided TB8 is updated before that TB8 is shifted out. TB8 must not be changed again until after TB8 shifting has been completed, as indicated by the Tx interrupt.

If double buffering is enabled, TB8 MUST be updated before SBUF is written, as TB8 will be double-buffered together with SBUF data. The operation described in the section "Transmit Interrupts with Double Buffering Enabled (Modes 1, 2 and 3)" becomes as follows:

1. The double buffer is empty initially.
2. The CPU writes to TB8.
3. The CPU writes to SBUF.
4. The SBUF/TB8 data is loaded to the shift register and a Tx interrupt is generated immediately.
5. If there is more data, go to 7, else continue on 6.
6. If there is no more data, then:
  - If DBISEL is '0', no more interrupt will occur.
  - If DBISEL is '1' and INTLO is '0', a Tx interrupt will occur at the beginning of the STOP bit of the data currently in the shifter (which is also the last data).

## UART (P89LPC903)

P89LPC901/902/903

- If DBISEL is '1' and INTLO is '1', a Tx interrupt will occur at the end of the STOP bit of the data currently in the shifter (which is also the last data).
  - 7. If there is more data, the CPU writes to TB8 again.
  - 8. The CPU writes to SBUF again. Then:
    - If INTLO is '0', the new data will be loaded and a Tx interrupt will occur at the beginning of the STOP bit of the data currently in the shifter.
    - If INTLO is '1', the new data will be loaded and a Tx interrupt will occur at the end of the STOP bit of the data currently in the shifter.
- Go to 4.

Note that if DBISEL is '1' and the CPU is writing to SBUF when the STOP bit of the last data is shifted out, there can be an uncertainty of whether a Tx interrupt is generated already with the UART not knowing whether there is any more data following.

### Multiprocessor Communications

UART modes 2 and 3 have a special provision for multiprocessor communications. In these modes, 9 data bits are received or transmitted. When data is received, the 9th bit is stored in RB8. The UART can be programmed such that when the stop bit is received, the serial port interrupt will be activated only if RB8 = 1. This feature is enabled by setting bit SM2 in SCON. One way to use this feature in multiprocessor systems is as follows:

When the master processor wants to transmit a block of data to one of several slaves, it first sends out an address byte which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 0 in a data byte. With SM2 = 1, no slave will be interrupted by a data byte. An address byte, however, will interrupt all slaves, so that each slave can examine the received byte and see if it is being addressed. The addressed slave will clear its SM2 bit and prepare to receive the data bytes that follow. The slaves that weren't being addressed leave their SM2 bits set and go on about their business, ignoring the subsequent data bytes.

Note that SM2 has no effect in Mode 0, and must be '0' in Mode 1.

### Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes (mode 2 and mode 3), the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the "Given" address or the "Broadcast" address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave's address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to be used and which bits are "don't care". The SADEN mask can be logically ANDed with the SADDR to create the "Given" address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

```
Slave 0  SADDR = 1100 0000
         SADEN = 1111 1101
         Given  = 1100 00X0
```

```
Slave 1  SADDR = 1100 0000
         SADEN = 1111 1110
         Given  = 1100 000X
```

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010

## UART (P89LPC903)

P89LPC901/902/903

since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Slave 0 SADDR = 1100 0000  
SADEN = 1111 1001  
Given = 1100 0XX0

Slave 1 SADDR = 1110 0000  
SADEN = 1111 1010  
Given = 1110 0X0X

Slave 2 SADDR = 1110 0000  
SADEN = 1111 1100  
Given = 1110 00XX

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2. The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as don't-cares. In most cases, interpreting the don't-cares as ones, the broadcast address will be FF hexadecimal. Upon reset SADDR and SADEN are loaded with 0s. This produces a given address of all "don't cares" as well as a Broadcast address of all "don't cares". This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard UART drivers which do not make use of this feature.

---

# UART (P89LPC903)

---

P89LPC901/902/903



# RESET

**P89LPC901/902/903**

## 9. RESET

The P1.5/ $\overline{\text{RST}}$  pin can function as either an active low reset input or as a digital input, P1.5. The RPE (Reset Pin Enable) bit in UCFG1, when set to 1, enables the external reset input function on P1.5. When cleared, P1.5 may be used as an input pin.

NOTE: During a power-on sequence, The RPE selection is overridden and this pin will always function as a reset input. An external circuit connected to this pin should not hold this pin low during a Power-on sequence as this will keep the device in reset. After power-on this input will function either as an external reset input or as a digital input as defined by the RPE bit. Only a power-on reset will temporarily override the selection defined by RPE bit. Other sources of reset will not override the RPE bit.

NOTE: During a power cycle,  $V_{DD}$  must fall below  $V_{POR}$  (see "DC electrical characteristics" in the datasheet) before power is reapplied, in order to ensure a power-on reset.

Reset can be triggered from the following sources (see Figure 9-1):

- External reset pin (during power-on or if user configured via UCFG1);
- Power-on Detect;
- Brownout Detect;
- Watchdog Timer;
- Software reset;
- UART break-character detect reset. (**P89LPC903**)

For every reset source, there is a flag in the Reset Register, RSTSRC. The user can read this register to determine the most recent reset source. These flag bits can be cleared in software by writing a '0' to the corresponding bit. More than one flag bit may be set:

- During a power-on reset, both POF and BOF are set but the other flag bits are cleared.
- For any other reset, any previously set flag bits that have not been cleared will remain set.

### Power-On reset code execution

The **P89LPC901/902/903** contains two special Flash elements: the BOOT VECTOR and the Boot Status Bit. Following reset, the **P89LPC901/902/903** examines the contents of the Boot Status Bit. If the Boot Status Bit is set to zero, power-up execution starts at location 0000H, which is the normal start address of the user's application code. When the Boot Status Bit is set to a one, the contents of the Boot Vector is used as the high byte of the execution address and the low byte is set to 00H. The factory default setting is 00H. A UART break-detect reset (**P89LPC903**) will have the same effect as a non-zero Status Bit.

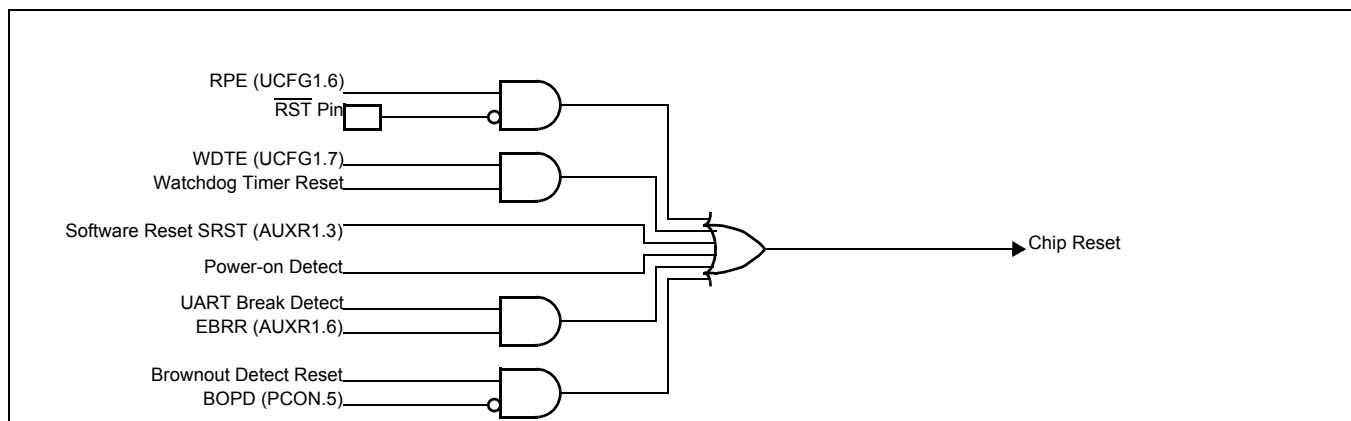


Figure 9-1: Block Diagram of Reset

# RESET

## P89LPC901/902/903

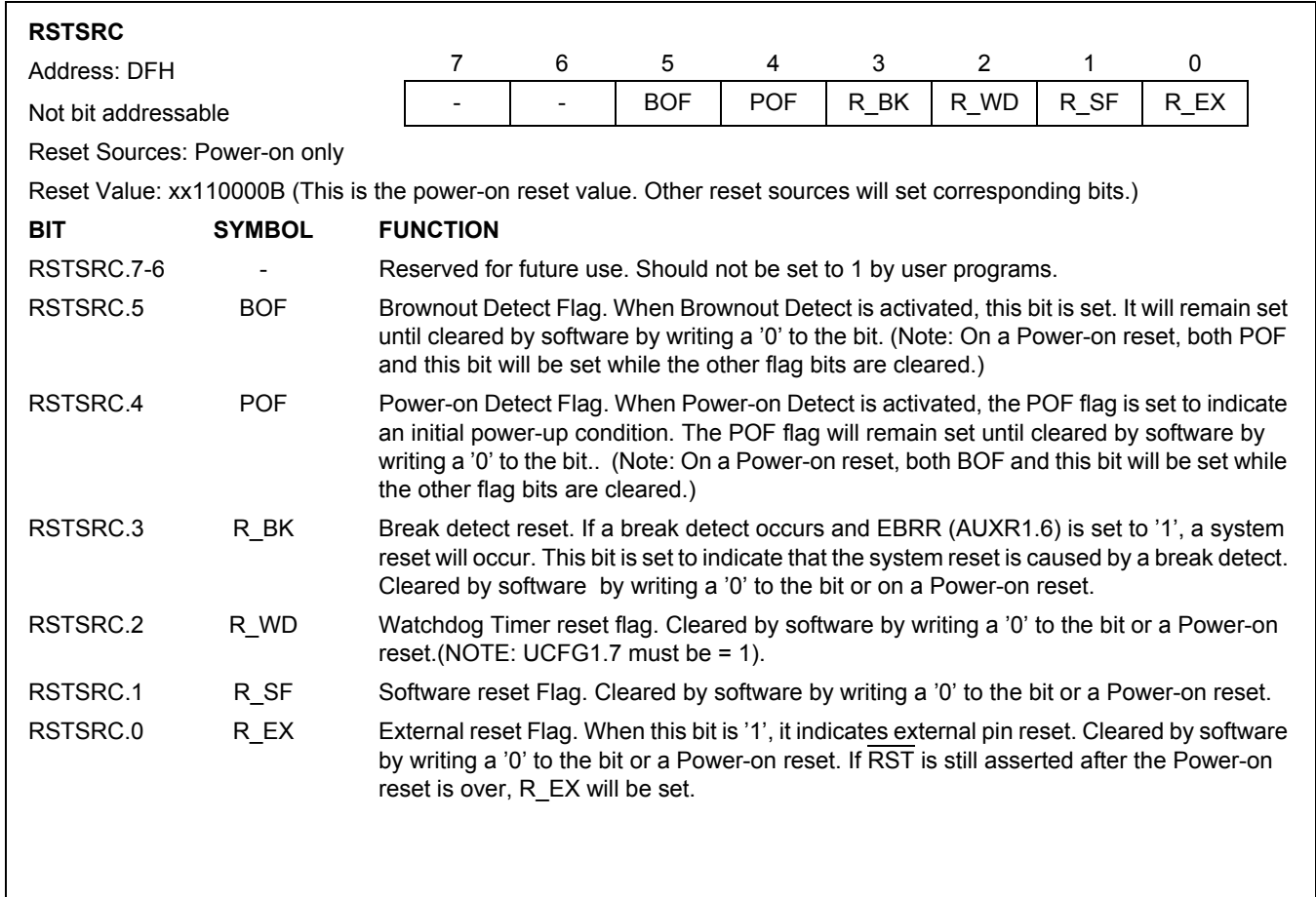


Figure 9-2: Reset Sources Register

# ANALOG COMPARATORS

## P89LPC901/902/903

### 10. ANALOG COMPARATORS

One analog comparator is provided on the **P89LPC901** and two analog comparators are provided on both the **P89LPC902** and **P89LPC903**. Comparator operation is such that the output is a logical one when the positive input is greater than the negative input (selectable from a pin or an internal reference voltage). Otherwise the output is a zero. The output may be read in a register. On the **P89LPC902** the output may also be routed to a pin. The comparator(s) may be configured to cause an interrupt when the output value changes.

The connections to the comparator(s) are shown in Figure 10-2 - Figure 10-4. The comparator functions to  $V_{DD} = 2.4V$ .

When the comparator is first enabled, the comparator output and interrupt flag are not guaranteed to be stable for 10 microseconds. The comparator interrupt should not be enabled during that time, and the comparator interrupt flag must be cleared before the interrupt is enabled in order to prevent an immediate interrupt service.

#### Comparator Configuration

The comparator(s) have a control register(s), **CMPn**, and is shown in Figure 10-1. The possible configurations for the comparator are shown in Figure 10-5.

<b>CMPn</b>																		
Address: ACh																		
Not bit addressable																		
Reset Source(s): Any reset																		
Reset Value: xx00000B																		
		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 12.5%; text-align: center;">7</td> <td style="width: 12.5%; text-align: center;">6</td> <td style="width: 12.5%; text-align: center;">5</td> <td style="width: 12.5%; text-align: center;">4</td> <td style="width: 12.5%; text-align: center;">3</td> <td style="width: 12.5%; text-align: center;">2</td> <td style="width: 12.5%; text-align: center;">1</td> <td style="width: 12.5%; text-align: center;">0</td> </tr> <tr> <td style="text-align: center;">-</td> <td style="text-align: center;">-</td> <td style="text-align: center;">CEn</td> <td style="text-align: center;">-</td> <td style="text-align: center;">CNn</td> <td style="text-align: center;">OEn</td> <td style="text-align: center;">CO<sub>n</sub></td> <td style="text-align: center;">CMF<sub>n</sub></td> </tr> </table>	7	6	5	4	3	2	1	0	-	-	CEn	-	CNn	OEn	CO <sub>n</sub>	CMF <sub>n</sub>
7	6	5	4	3	2	1	0											
-	-	CEn	-	CNn	OEn	CO <sub>n</sub>	CMF <sub>n</sub>											
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>																
CMP.7, 6	-	Reserved for future use.																
CMP.5	CEn	Comparator enable. When set, the comparator function is enabled. Comparator output is stable 10 microseconds after CEn is set.																
CMP.4	-	Reserved for future use.																
CMP.3	CNn	Comparator negative input select. When 0, the comparator reference pin CMPREF is selected as the negative comparator input. When 1, the internal comparator reference, Vref, is selected as the negative comparator input.																
CMP.2	OEn	Output enable. When 1, the comparator output is connected to the CMPn pin if the comparator is enabled (CEn = 1). This output is asynchronous to the CPU clock.																
CMP.1	CO <sub>n</sub>	Comparator output, synchronized to the CPU clock to allow reading by software.																
CMP.0	CMF <sub>n</sub>	Comparator interrupt flag. This bit is set by hardware whenever the comparator output CO <sub>n</sub> changes state. This bit will cause a hardware interrupt if enabled. Cleared by software.																

Figure 10-1: Comparator Control Registers (CMP1 and CMP2)

# ANALOG COMPARATORS

P89LPC901/902/903

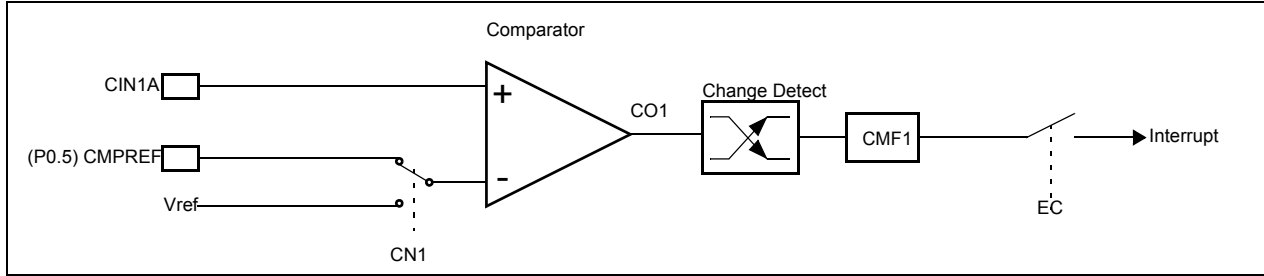


Figure 10-2: Comparator Input and Output Connections - P89LPC901

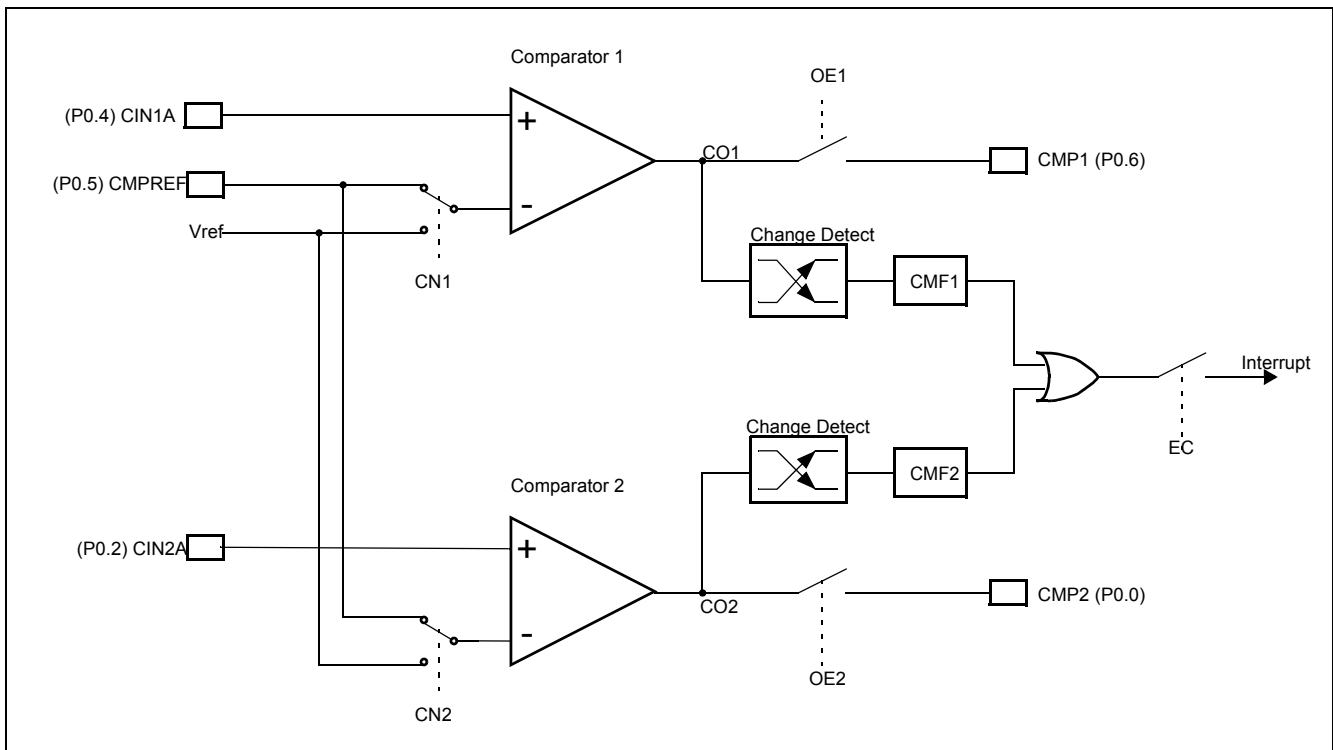


Figure 10-3: Comparator Input and Output Connections - P89LPC902

# ANALOG COMPARATORS

P89LPC901/902/903

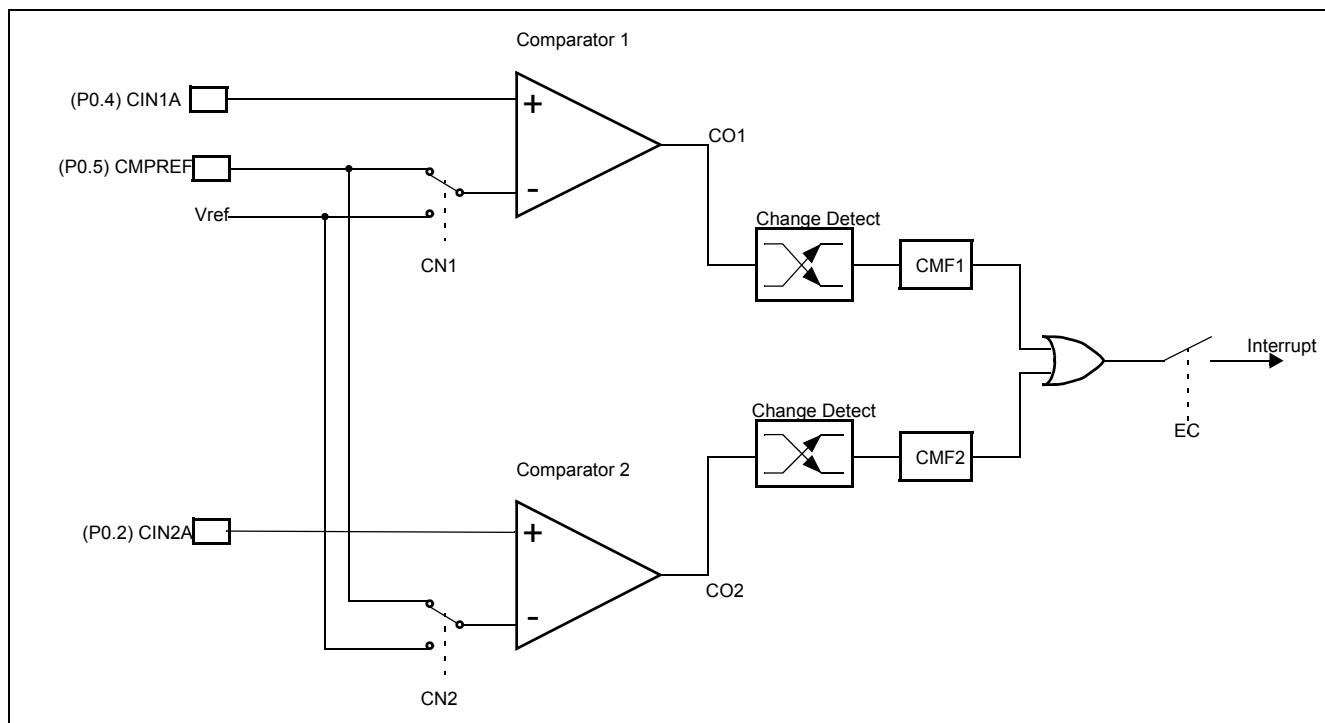


Figure 10-4: Comparator Input and Output Connections - P89LPC903

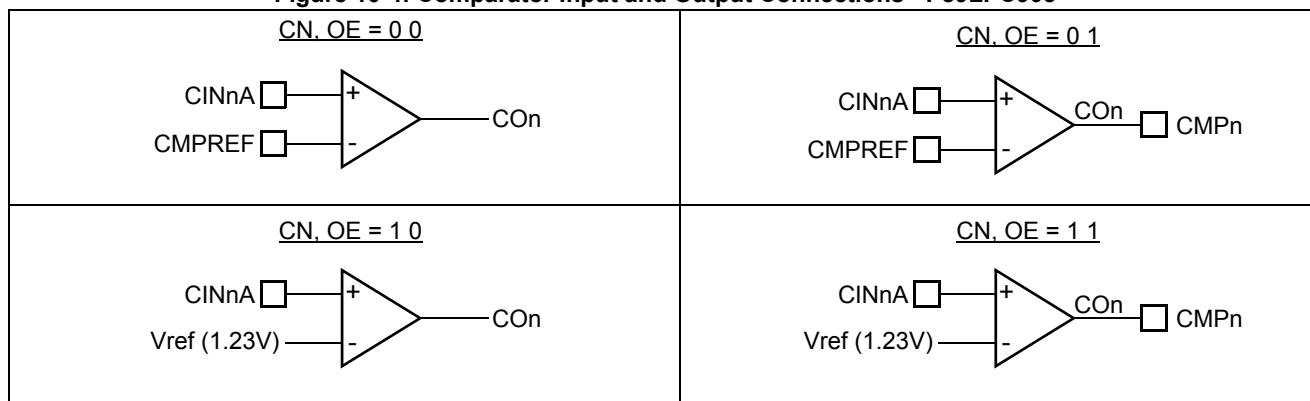


Figure 10-5: Comparator Configurations

## Internal Reference Voltage

An internal reference voltage, Vref, may supply a default reference when a single comparator input pin is used. Please refer to the Datasheet for specifications.

## Comparator Interrupt

The comparator has an interrupt flag CMFn contained in its configuration register. This flag is set whenever the comparator output changes state. The flag may be polled by software or may be used to generate an interrupt. The interrupt will be generated when the interrupt enable bit EC in the IEN1 register is set and the interrupt system is enabled via the EA bit in the IEN0 register.

When a comparator is disabled the comparator's output, COx, goes high. If the comparator output was low and then is disabled, the resulting transition of the comparator output from a low to high state will set the the comparator flag, CMFx. This will cause

# ANALOG COMPARATORS

P89LPC901/902/903

an interrupt if the comparator interrupt is enabled. The user should therefore disable the comparator interrupt prior to disabling the comparator. Additionally, the user should clear the comparator flag, CMFx, after disabling the comparator.

## Comparator and Power Reduction Modes

The comparator(s) may remain enabled when Power down or Idle mode is activated, but the comparator(s) are disabled automatically in Total Power down mode.

If the comparator interrupt is enabled (except in Total Power down mode), a change of the comparator output state will generate an interrupt and wake up the processor.

If the comparator output to a pin is enabled, the pin should be configured in the push-pull mode in order to obtain fast switching times while in power down mode. The reason is that with the oscillator stopped, the temporary strong pullup that normally occurs during switching on a quasi-bidirectional port pin does not take place.

The comparator consumes power in Power down and Idle modes, as well as in the normal operating mode. This fact should be taken into account when system power consumption is an issue. To minimize power consumption, the user can disable the comparator via PCONA.5 or put the device in Total Power down mode.

## Comparator Configuration Example

The code shown below is an example of initializing the comparator. This comparator is configured to use the CMPREF inputs. The comparator output drives the CMP pin and generates an interrupt when the comparator output changes.

CMPINIT:

```

MOV    PT0AD,#030h    ; Disable digital INPUTS on pins that are used
                        ; for analog functions: CIN, CMPREF.

ANL    P0M2,#0CFh    ; Disable digital OUTPUTS on pins that are used
ORL    P0M1,#030h    ; for analog functions: CIN, CMPREF.

MOV    CMP1,#024h    ; Turn on comparator and set up for:
                        ; - Negative input from CMPREF pin.
                        ; - Output to CMP pin enabled.

CALL   delay10us     ; The comparator has to start up for at
                        ; least 10 microseconds before use.

ANL    CMP1,#0FEh    ; Clear comparator interrupt flag.
SETB   EC            ; Enable the comparator interrupt. The
                        ; priority is left at the current value.

SETB   EA            ; Enable the interrupt system (if needed).
RET                                ; Return to caller.

```

The interrupt routine used for the comparator must clear the interrupt flag (CMF1 in this case) before returning.

# KEYPAD INTERRUPT (KBI)

P89LPC901/902/903

## 11. KEYPAD INTERRUPT (KBI)

The Keypad Interrupt function is intended primarily to allow a single interrupt to be generated when the Port 0 bits are equal to or not equal to a certain pattern. This function can be used for keypad recognition. The user can configure the port via SFRs for different tasks.

There are three SFRs used for this function. The Keypad Interrupt Mask Register (KBMASK) is used to define which input pins connected to Port 0 are enabled to trigger the interrupt. The Keypad Pattern Register (KBPATN) is used to define a pattern that is compared to the value of Port 0. The Keypad Interrupt Flag (KBIF) in the Keypad Interrupt Control Register (KBCON) is set when the condition is matched while the Keypad Interrupt function is active. An interrupt will be generated if it has been enabled by setting the EKBI bit in IEN1 register and EA = 1. The PATN\_SEL bit in the Keypad Interrupt Control Register (KBCON) is used to define equal or not-equal for the comparison.

In order to use the Keypad Interrupt as an original KBI function like in the 87LPC76x series, the user needs to set KBPATN = 0FFH and PATN\_SEL = 0 (not equal), then any key connected to Port0 which is enabled by KBMASK register will cause the hardware to set KBIF = 1 and generate an interrupt if it has been enabled. The interrupt may be used to wake up the CPU from Idle or Power down modes. This feature is particularly useful in handheld, battery powered systems that need to carefully manage power consumption yet also need to be convenient to use.

In order to set the flag and cause an interrupt, the pattern on Port 0 must be held longer than 6 CCLKs.

<b>KBPATN</b>			7	6	5	4	3	2	1	0
Address: 93h			-	-	KBPATN.5	KBPATN.4	-	-	-	-
Not bit addressable										
Reset Source(s): Any reset										
Reset Value: 11111111B										
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>								
KBPATN.5,4	-	Pattern bit 5, bit 4								

Figure 11-1: Keypad Pattern Register- P89LPC901

<b>KBPATN</b>			7	6	5	4	3	2	1	0
Address: 93h			-	KBPATN.6	KBPATN.5	KBPATN.4	-	KBPATN.2	-	KBPATN.0
Not bit addressable										
Reset Source(s): Any reset										
Reset Value: 11111111B										
<b>BIT</b>	<b>SYMBOL</b>	<b>FUNCTION</b>								
KBPATN.6-4,2,0	-	Pattern bit 6 - bit 4, bit 2, bit 0								

Figure 11-2: Keypad Pattern Register - P89LPC902

# KEYPAD INTERRUPT (KBI)

P89LPC901/902/903

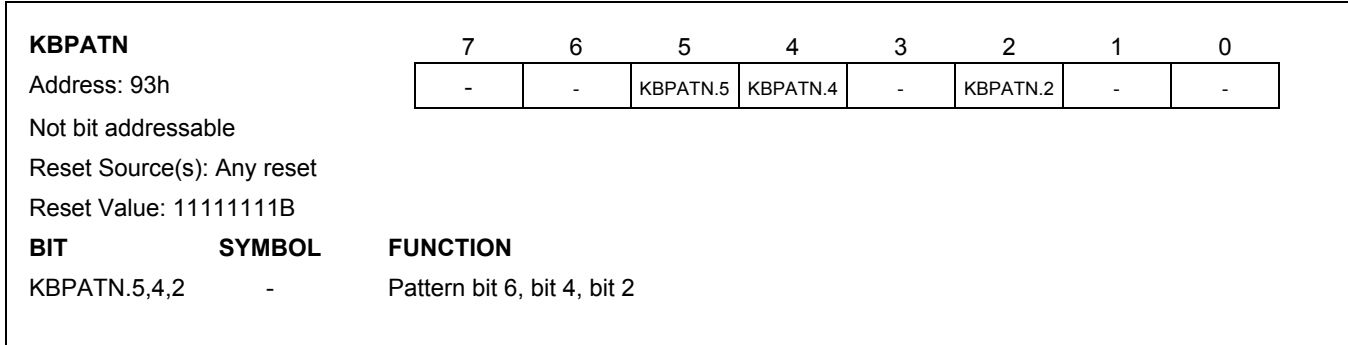


Figure 11-3: Keypad Pattern Register - P89LPC903

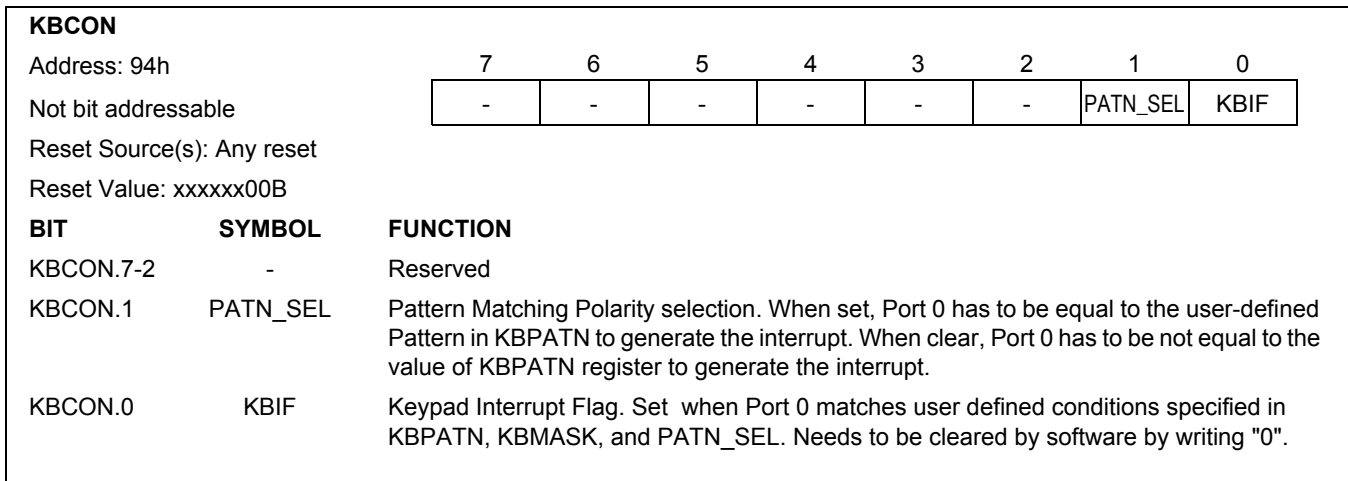


Figure 11-4: Keypad Control Register

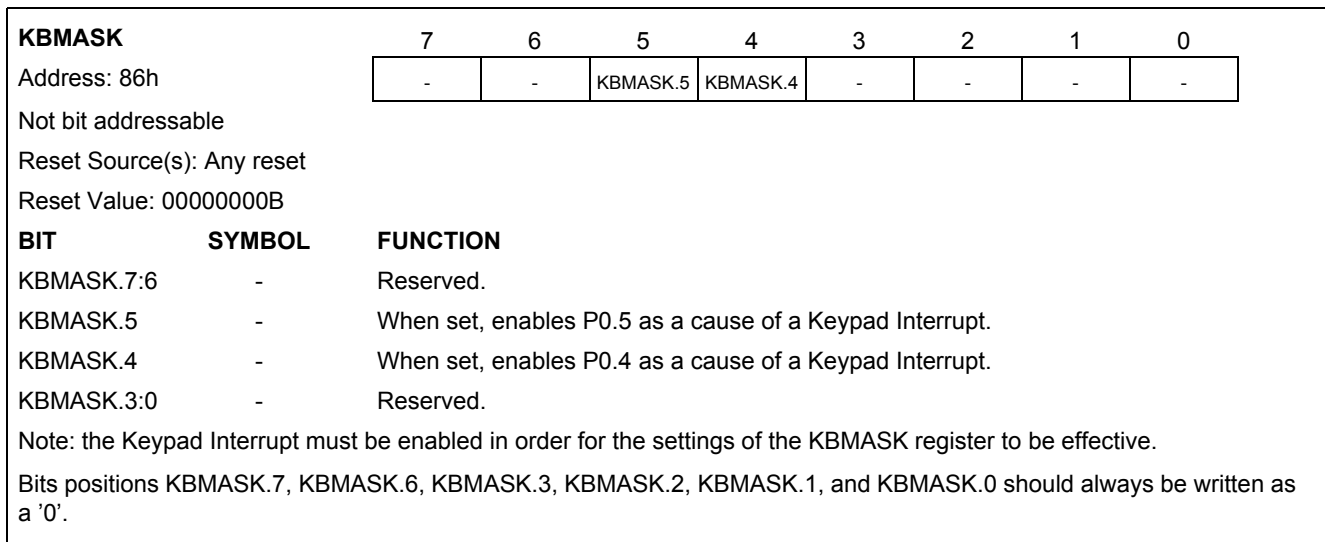


Figure 11-5: Keypad Interrupt Mask Register (KBM) - P89LPC901



# KEYPAD INTERRUPT (KBI)

P89LPC901/902/903

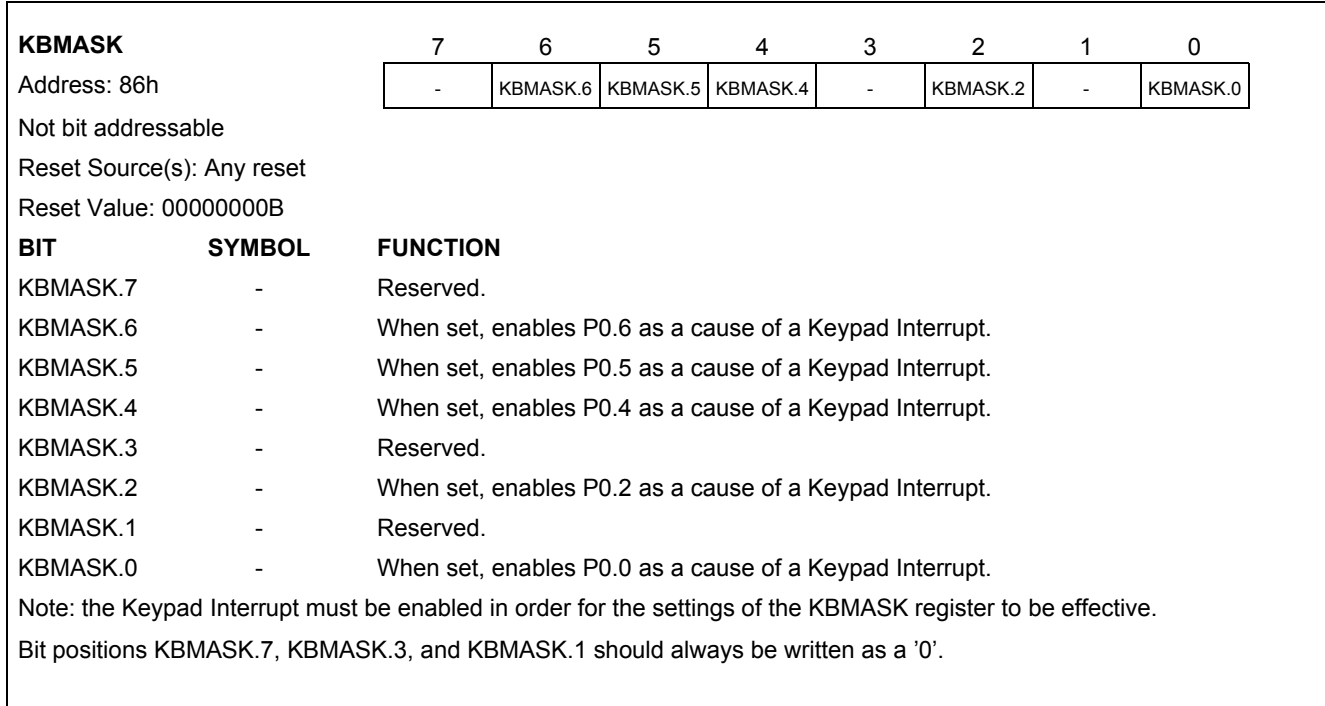


Figure 11-6: Keypad Interrupt Mask Register (KBM) - P89LPC902

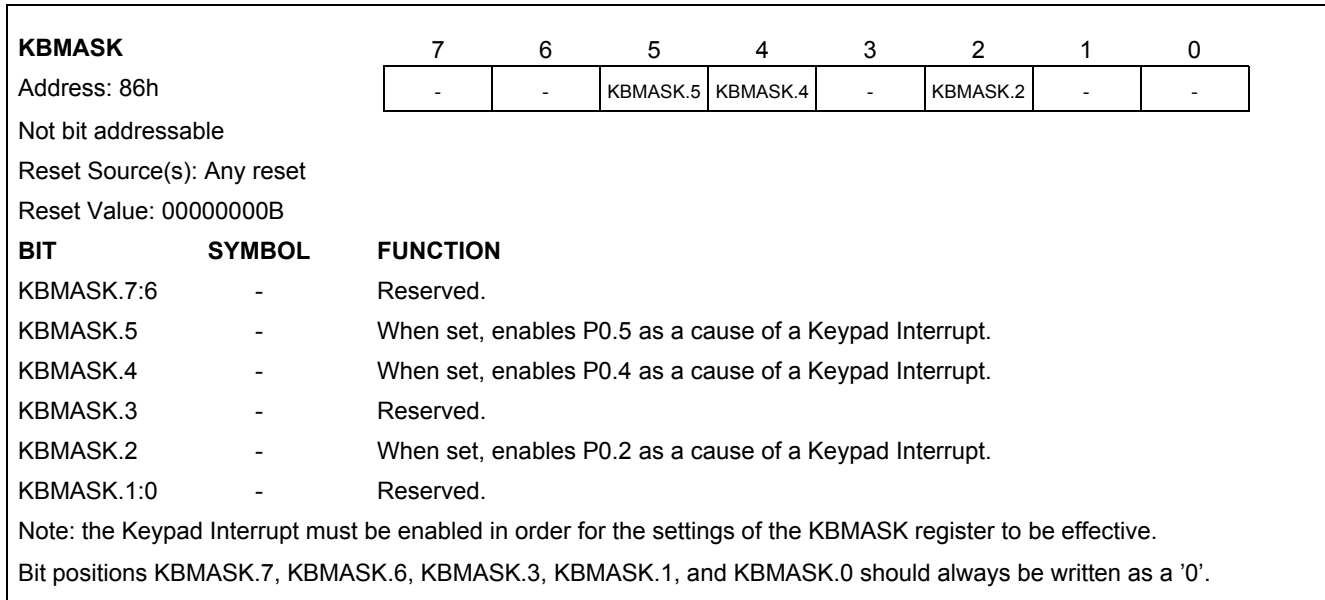


Figure 11-7: Keypad Interrupt Mask Register (KBM) - P89LPC903

---

# KEYPAD INTERRUPT (KBI)

P89LPC901/902/903

---

# WATCHDOG TIMER

P89LPC901/902/903

## 12. WATCHDOG TIMER

The watchdog timer subsystem protects the system from incorrect code execution by causing a system reset when it underflows as a result of a failure of software to feed the timer prior to the timer reaching its terminal count. The watchdog timer can only be reset by a power-on reset.

### Watchdog Function

The user has the ability using the WDCON and UCFG1 registers to control the run /stop condition of the WDT, the clock source for the WDT, the prescaler value, and whether the WDT is enabled to reset the device on underflow. In addition, there is a safety mechanism which forces the WDT to be enabled by values programmed into UCFG1 either through IAP or a commercial programmer.

The WDTE bit (UCFG1.7), if set, enables the WDT to reset the device on underflow. Following reset, the WDT will be running regardless of the state of the WDTE bit.

The WDRUN bit (WDCON.2) can be set to start the WDT and cleared to stop the WDT. Following reset this bit will be set and the WDT will be running. All writes to WDCON need to be followed by a feed sequence (see section "Feed Sequence" on page 84). Additional bits in WDCON allow the user to select the clocksource for the WDT and the prescaler.

When the timer is not enabled to reset the device on underflow, the WDT can be used in "timer mode" and be enabled to produce an interrupt (IEN0.6) if desired.

The Watchdog Safety Enable bit, WDSE (UCFG1.4) along with WDTE, is designed to force certain operating conditions at power-up. Refer to the Table for details.

**Table 12-1: Watchdog timer configuration**

WDTE (UCFG1.7)	WDSE (UCFG1.4)	FUNCTION
0	x	The watchdog reset is disabled. The timer can be used as an internal timer and can be used to generate an interrupt. WDSE has no effect.
1	0	The watchdog reset is enabled. The user can set WDCLK to choose the clock source.
1	1	The watchdog reset is enabled, along with additional safety features: 1. WDCLK is forced to 1 (using watchdog oscillator) 2. WDCON and WDL register can only be written once 3. WDRUN is forced to 1 and cannot be cleared by software.

Figure 12-3 shows the watchdog timer in watchdog mode. It consists of a programmable 13-bit prescaler, and an 8-bit down counter. The down counter is clocked (decremented) by a tap taken from the prescaler. The clock source for the prescaler is either PCLK or the watchdog oscillator selected by the WDCLK bit in the WDCON register. (Note that switching of the clock sources will not take effect immediately - see section "Watchdog Clock Source" on page 87).

The watchdog asserts the watchdog reset when the watchdog count underflows and the watchdog reset is enabled. When the watchdog reset is enabled, writing to WDL or WDCON must be followed by a feed sequence for the new values to take effect.

If a watchdog reset occurs, the internal reset is active for at least one watchdog clock cycle (PCLK or the watchdog oscillator clock). If CCLK is still running, code execution will begin immediately after the reset cycle. If the processor was in Power down mode, the watchdog reset will start the oscillator and code execution will resume after the oscillator is stable.

# WATCHDOG TIMER

## P89LPC901/902/903

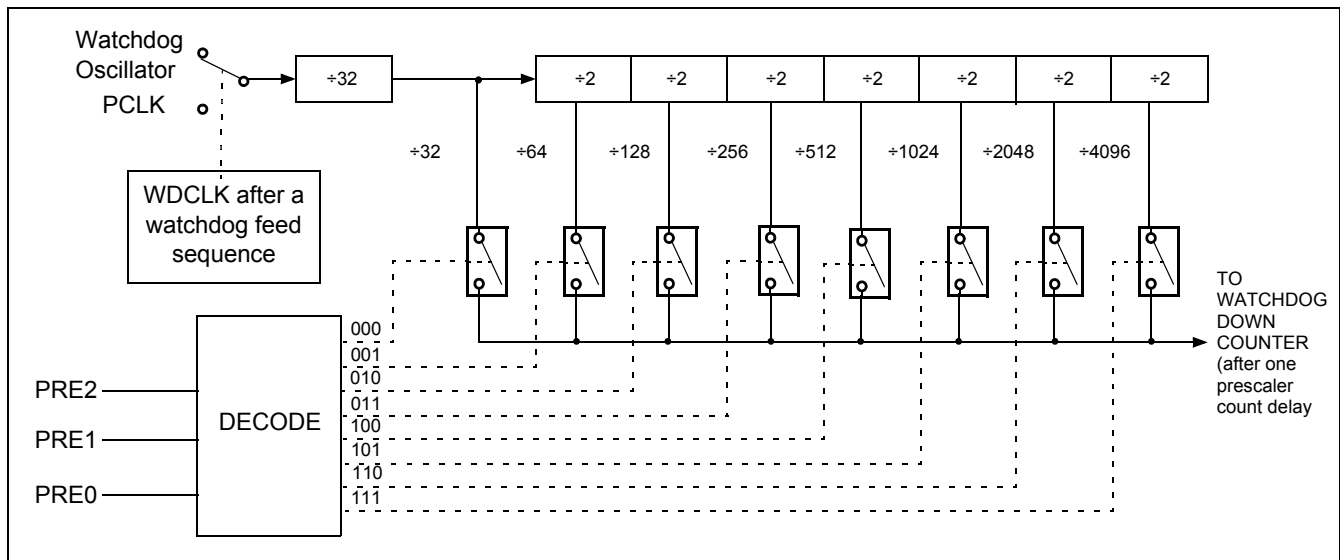


Figure 12-1: Watchdog Prescaler

## Feed Sequence

The watchdog timer control register and the 8-bit down counter (Figure 12-3) are not directly loaded by the user. The user writes to the WDCON and the WDL SFRs. At the end of a feed sequence, the values in the WDCON and WDL SFRs are loaded to the control register and the 8-bit down counter. Before the feed sequence, any new values written to these two SFRs will not take effect. To avoid a watchdog reset, the watchdog timer needs to be fed (via a special sequence of software action called the feed sequence) prior to reaching an underflow.

To feed the watchdog, two write instructions must be sequentially executed successfully. Between the two write instructions, SFR reads are allowed, but writes are not allowed. The instructions should move A5H to the WFEED1 register and then 5AH to the WFEED2 register. An incorrect feed sequence will cause an immediate watchdog reset. The program sequence to feed the watchdog timer is as follows:

```
CLR    EA                ; disable interrupt
MOV    WFEED1,#0A5h     ; do watchdog feed part 1
MOV    WFEED2,#05Ah     ; do watchdog feed part 2
SETB   EA                ; enable interrupt
```

This sequence assumes that the P89LPC901/902/903 interrupt system is enabled and there is a possibility of an interrupt request occurring during the feed sequence. If an interrupt was allowed to be serviced and the service routine contained any SFR writes, it would trigger a watchdog reset. If it is known that no interrupt could occur during the feed sequence, the instructions to disable and re-enable interrupts may be removed.

In watchdog mode (WDTE = 1), writing the WDCON register must be IMMEDIATELY followed by a feed sequence to load the WDL to the 8-bit down counter, and the WDCON to the shadow register. If writing to the WDCON register is not immediately followed by the feed sequence, a watchdog reset will occur.

For example: setting WDRUN = 1:

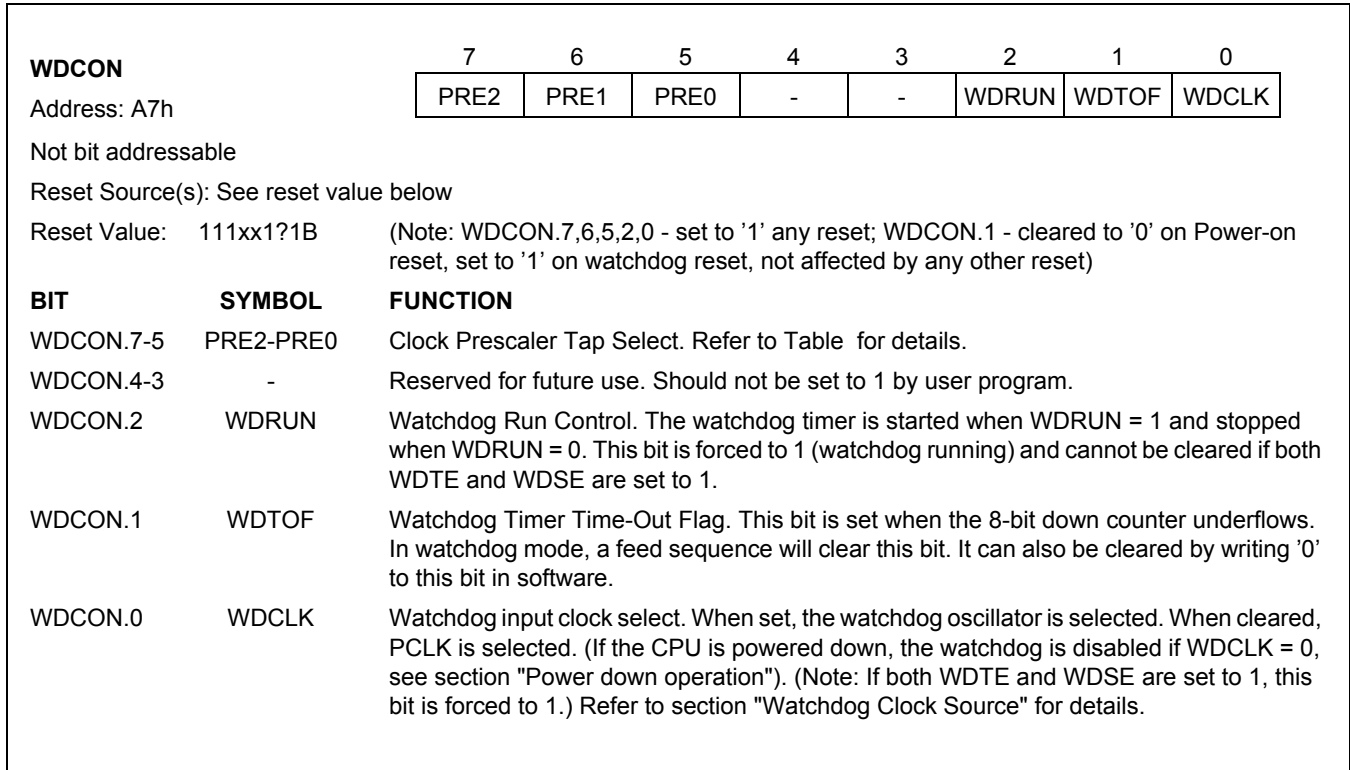
```
MOV    ACC,WDCON        ; get WDCON
SETB   ACC.2            ; set WD_RUN=1
MOV    WDL,#0FFh       ; New count to be loaded to 8-bit down counter
CLR    EA                ; disable interrupt
MOV    WDCON,ACC        ; write back to WDCON (after the watchdog is enabled, a feed must occur
                        ; immediately)
```

# WATCHDOG TIMER

P89LPC901/902/903

```
MOV  WFEED1,#0A5h      ; do watchdog feed part 1
MOV  WFEED2,#05Ah      ; do watchdog feed part 2
SETB EA                 ; enable interrupt
```

In timer mode (WDTE = 0), WDCON is loaded to the control register every CCLK cycle (no feed sequence is required to load the control register), but a feed sequence **is required** to load from the WDL SFR to the 8-bit down counter before a time-out occurs.



**Figure 12-2: Watchdog Timer Control Register**

The number of watchdog clocks before timing out is calculated by the following equations:

$$tclks = (2^{(5+PRE)})(WDL+1)+1$$

where:

- PRE is the value of prescaler (PRE2-PRE0) which can be the range 0-7, and;
- WDL is the value of watchdog load register which can be the range of 0-255.

The minimum number of tclks is:

$$tclks = (2^{(5+0)})(0+1)+1 = 33$$

The maximum number of tclks is:

$$tclks = (2^{(5+7)})(255+1)+1 = 1,048,577$$

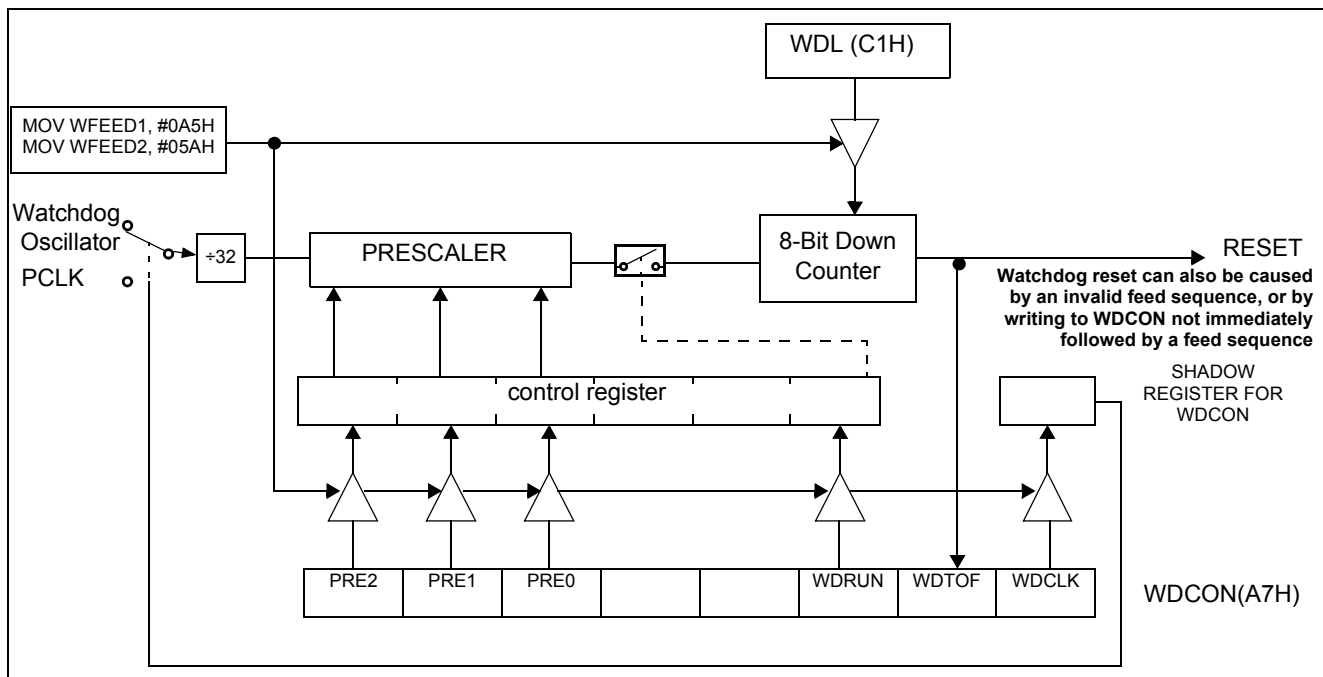
The following table shows sample P89LPC901/902/903 timeout values.

# WATCHDOG TIMER

## P89LPC901/902/903

**Table 12-2: P89LPC901/902/903 Watchdog Timeout Values**

PRE2-PRE0	WDL in decimal)	Timeout Period (in watchdog clock cycles)	Watchdog Clock Source	
			400KHz Watchdog Oscillator Clock (Nominal)	12MHz CCLK (6MHz CCLK/2 Watchdog Clock)
000	0	33	82.5µs	5.50µs
	255	8,193	20.5ms	1.37ms
001	0	65	162.5µs	10.8µs
	255	16,385	41.0ms	2.73ms
010	0	129	322.5µs	21.5µs
	255	32,769	81.9ms	5.46ms
011	0	257	642.5µs	42.8µs
	255	65,537	163.8ms	10.9ms
100	0	513	1.28ms	85.5µs
	255	131,073	327.7ms	21.8ms
101	0	1,025	2.56ms	170.8µs
	255	262,145	655.4ms	43.7ms
110	0	2,049	5.12ms	341.5µs
	255	524,289	1.31s	87.4ms
111	0	4097	10.2ms	682.8µs
	255	1,048,577	2.62s	174.8ms



**Figure 12-3: Watchdog Timer in Watchdog Mode (WDTE = 1)**

# WATCHDOG TIMER

P89LPC901/902/903

## Watchdog Timer in Timer Mode

Figure 12-4 shows the Watchdog Timer in Timer Mode. In this mode, any changes to WDCON are written to the shadow register after one watchdog clock cycle. A watchdog underflow will set the WDTOF bit. If IEN0.6 is set, the watchdog underflow is enabled to cause an interrupt. WDTOF is cleared by writing a '0' to this bit in software. When an underflow occurs, the contents of WDL is reloaded into the down counter and the watchdog timer immediately begins to count down again.

A feed is necessary to cause WDL to be loaded into the down counter before an underflow occurs. Incorrect feeds are ignored in this mode.

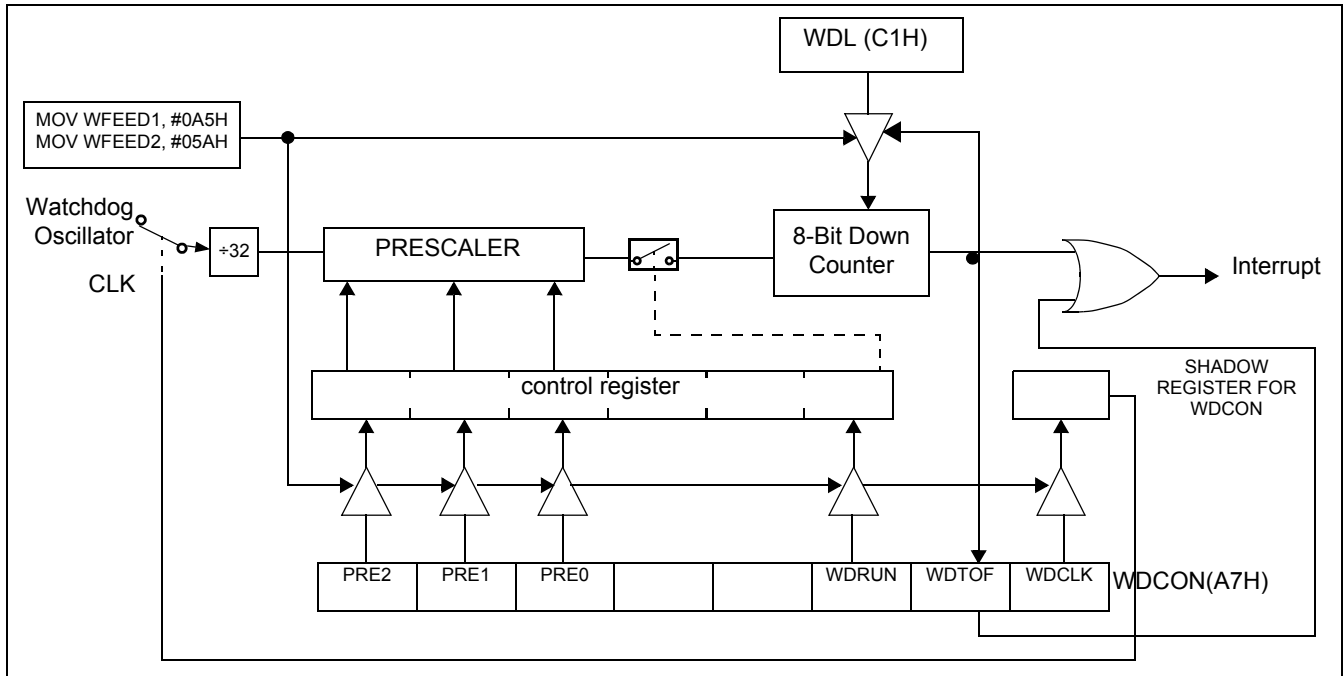


Figure 12-4: Watchdog Timer in Timer Mode (WDTE = 0)

### Power down operation

The WDT oscillator will continue to run in power down, consuming approximately 50uA, as long as the WDT oscillator is selected as the clock source for the WDT. Selecting PCLK as the WDT source will result in the WDT oscillator going into power down with the rest of the device (see section "Watchdog Clock Source", below). Power down mode will also prevent PCLK from running and therefore the watchdog is effectively disabled.

### Watchdog Clock Source

The watchdog timer system has an on-chip 400KHz oscillator. The watchdog timer can be clocked from either the watchdog oscillator or from PCLK (refer to Figure 12-1) by configuring the WDCLK bit in the Watchdog Control Register WDCON. When the watchdog feature is enabled, the timer must be fed regularly by software in order to prevent it from resetting the CPU.

After changing WDCLK (WDCON.0), switching of the clock source will not immediately take effect. As shown in Figure 12-3, the selection is loaded after a watchdog feed sequence. In addition, due to clock synchronization logic, it can take two old clock cycles before the old clock source is deselected, and then an additional two new clock cycles before the new clock source is selected.

Since the prescaler starts counting immediately after a feed, switching clocks can cause some inaccuracy in the prescaler count. The inaccuracy could be as much as 2 old clock source counts plus 2 new clock cycles.

---

# WATCHDOG TIMER

P89LPC901/902/903

---

**Note:** When switching clocks, it is important that the old clock source is left enabled for 2 clock cycles after the feed completes. Otherwise, the watchdog may become disabled when the old clock source is disabled. For example, suppose PCLK (WCLK=0) is the current clock source. After WCLK is set to '1', the program should wait at least two PCLK cycles (4 CCLKs) after the feed completes before going into Power down mode. Otherwise, the watchdog could become disabled when CCLK turns off. The watchdog oscillator will never become selected as the clock source unless CCLK is turned on again first.



---

# WATCHDOG TIMER

---

P89LPC901/902/903

## Periodic wakeup from Power down without an external oscillator

Without using an external oscillator source, the power consumption required in order to have a periodic wakeup is determined by the power consumption of the internal oscillator source used to produce the wakeup. The Real-time clock running from the internal RC oscillator can be used. The power consumption of this oscillator is approximately 300uA. Instead, if the WDT is used to generate interrupts the current is reduced to approximately 50uA. Whenever the WDT underflows, the device will wake up.

---

# WATCHDOG TIMER

---

P89LPC901/902/903

# ADDITIONAL FEATURES

**P89LPC901/902/903**

## 13. ADDITIONAL FEATURES

The AUXR1 register contains several special purpose control bits that relate to several chip features. AUXR1 is described in Figure 13-1.

AUXR1			7	6	5	4	3	2	1	0
Address: A2h			CLKLP	EBRR	-	-	SRST	0	-	DPS
Not bit addressable										
Reset Source(s): Any reset										
Reset Value: 000000x0B										
BIT	SYMBOL	FUNCTION								
AUXR1.7	CLKLP	Clock Low Power Select. When set, reduces power consumption in the clock circuits. Can be used when the clock frequency is 8MHz or less. After reset this bit is cleared to support up to 12MHz operation ( <b>P89LPC901</b> ).								
AUXR1.6	EBRR	UART Break Detect Reset Enable. If '1', UART Break Detect will cause a chip reset ( <b>P89LPC903</b> ). When writing to this register on the <b>P89LPC901</b> or <b>P89LPC902</b> devices, this bit position should be written with a zero.								
AUXR1.5	-	Reserved								
AUXR1.4	-	Reserved								
AUXR1.3	SRST	Software Reset. When set by software, resets the P89LPC901/902/903 as if a hardware reset occurred.								
AUXR1.2	0	This bit contains a hard-wired 0. Allows toggling of the DPS bit by incrementing AUXR1, without interfering with other bits in the register.								
AUXR1.1	-	Not used. Allowable to set to a "1" .								
AUXR1.0	DPS	Data Pointer Select. Chooses one of two Data Pointers.								

**Figure 13-1: AUXR1 Register**

### Software Reset

The SRST bit in AUXR1 gives software the opportunity to reset the processor completely, as if an external reset or watchdog reset had occurred. If a value is written to AUXR1 that contains a 1 at bit position 3, all SFRs will be initialized and execution will resume at program address 0000. Care should be taken when writing to AUXR1 to avoid accidental software resets.

### Dual Data Pointers

The dual Data Pointers (DPTR) adds to the ways in which the processor can specify the address used with certain instructions. The DPS bit in the AUXR1 register selects one of the two Data Pointers. The DPTR that is not currently selected is not accessible to software unless the DPS bit is toggled.

Specific instructions affected by the Data Pointer selection are:

- INC DPTR Increments the Data Pointer by 1.
- JMP @A+DPTR Jump indirect relative to DPTR value.

---

## ADDITIONAL FEATURES

**P89LPC901/902/903**

---

- **MOV DPTR, #data16** Load the Data Pointer with a 16-bit constant.
- **MOVCA, @A+DPTR** Move code byte relative to DPTR to the accumulator.
- **MOVXA, @DPTR** Move data byte the accumulator to data memory relative to DPTR.
- **MOVX@DPTR, A** Move data byte from data memory relative to DPTR to the accumulator.

Also, any instruction that reads or manipulates the DPH and DPL registers (the upper and lower bytes of the current DPTR) will be affected by the setting of DPS. The MOVX instructions have limited application for the P89LPC901/902/903 since the part does not have an external data bus. However, they may be used to access Flash configuration information (see Flash Configuration section).

Bit 2 of AUXR1 is permanently wired as a logic 0. This is so that the DPS bit may be toggled (thereby switching Data Pointers) simply by incrementing the AUXR1 register, without the possibility of inadvertently altering other bits in the register.

# FLASH PROGRAM MEMORY

P89LPC901/902/903

## 14. FLASH PROGRAM MEMORY

### General description

The **P89LPC901/902/903** Flash memory provides in-circuit electrical erasure and programming. The Flash can be read and written as bytes. On-chip erase and write timing generation contribute to a user-friendly programming interface. The cell is designed to optimize the erase and programming mechanisms. The **P89LPC901/902/903** uses  $V_{DD}$  as the supply voltage to perform the Program/Erase algorithms. Additionally, serial programming using commercially available programmers provides a simple interface to achieve in-circuit programming. The **P89LPC901/902/903** Flash reliably stores memory contents after 100,000 erase and program cycles (typical).

### Features

- IAP-Lite allows individual and multiple bytes of code memory to be used for data storage.
- Programming and erase over the full operating voltage range
- Read/Programming/Erase using IAP-Lite
- Any flash program operation in 2 ms (4ms for erase/program)
- Serial programming with industry-standard commercial programmers allows in-circuit programming.
- Programmable security for the code in the Flash for each sector.
- >100,000 typical erase/program cycles for each byte.
- 256 byte sector size, 16 byte page size
- 10-year minimum data retention.

### Introduction to IAP-Lite

The Flash code memory array of this device supports IAP-Lite programming and erase functions. Any byte in a non-secured sector of the code memory array may be read using the MOVC instruction and thus is suitable for use as non-volatile data storage. In addition, the user's code may access additional flash elements. These include UCFG1, the Boot Vector, Status Bit, security bytes, and signature bytes. Access of these elements uses a slightly different method than that used to access the user code memory.

### Using Flash as data storage

IAP-Lite provides an erase-program function that makes it easy for one or more bytes within a page to be erased and programmed in a single operation without the need to erase or program any other bytes in the page. IAP-Lite is performed in the application under the control of the microcontroller's firmware using four SFRs and an internal 16-byte "page register" to facilitate erasing and programming within unsecured sectors. These SFRs are:

- FMCON (Flash Control Register). When read, this is the status register. When written, this is a command register. Note that the status bits are cleared to '0's when the command is written.
- FMDATA (Flash Data Register). Accepts data to be loaded into the page register.
- FMADRL, FMADRH (Flash memory address low, Flash memory address high). Used to specify the byte address within the page register or specify the page within user code memory.

The page register consists of 16 bytes and an update flag for each byte. When a LOAD command is issued to FMCON the page register contents and all of the update flags will be cleared. When FMDATA is written, the value written to FMDATA will be stored in the page register at the location specified by the lower 6 bits of FMADRL. In addition, the update flag for that location will be set. FMADRL will auto-increment to the next location. Auto-increment after writing to the last byte in the page register will "wrap -around" to the first byte in the page register, but will not affect FMADRL[7:4]. Bytes loaded into the page register do not have to be continuous. Any byte location can be loaded into the page register by changing the contents of FMADRL prior to writ-

# FLASH PROGRAM MEMORY

**P89LPC901/902/903**

ing to FMDATA. However, each location in the page register can only be written once following each LOAD command. Attempts to write to a page register location more than once should be avoided.

FMADRH and FMADRL[7:4] are used to select a page of code memory for the erase-program function. When the erase-program command is written to FMCON, the locations within the code memory page that correspond to updated locations in the page register will have their contents erased and programmed with the contents of their corresponding locations in the page register. Only the bytes that were loaded into the page register will be erased and programmed in the user code array. Other bytes within the user code memory will not be affected.

Writing the erase-program command (68H) to FMCON will start the erase-program process and place the CPU in a program-idle state. The CPU will remain in this idle state until the erase-program cycle is either completed or terminated by an interrupt. When the program-idle state is exited, FMCON will contain status information for the cycle.

If an interrupt occurs during an erase/programming cycle, the erase/programming cycle will be aborted and the OI flag (Operation Interrupted) in FMCON will be set. If the application permits interrupts during erasing-programming, the user code should check the OI flag (FMCON.0) after each erase-programming operation to see if the operation was aborted. If the operation was aborted, the user's code will need to repeat the process starting with loading the page register.

The erase-program cycle takes 4ms to complete, regardless of the number of bytes that were loaded into the page register.

Erasing-programming of a single byte (or multiple bytes) in code memory is accomplished using the following steps:

- Write the LOAD command (00H) to FMCON. The LOAD command will clear all locations in the page register and their corresponding update flags.
- Write the address within the page register to FMADRL. Since the loading the page register uses FMADRL[5:0], and since the erase-program command uses FMADRH and FMADRL[7:4], the user can write the byte location within the page register (FMADRL[3:0]) and the code memory page address (FMADRH and FMADRL[7:4]) at this time.
- Write the data to be programmed to FMDATA. This will increment FMADRL pointing to the next byte in the page register.
- Write the address of the next byte to be programmed to FMADRL, if desired. (Not needed for contiguous bytes since FMADRL is auto-incremented). All bytes to be programmed must be within the same page.
- Write the data for the next byte to be programmed to FMDATA.
- Repeat writing of FMADRL and/or FMDATA until all desired bytes have been loaded into the page register.
- Write the page address in user code memory to FMADRH and FMADRL[7:4], if not previously included when writing the page register address to FMADRL[3:0].
- Write the erase-program command (68H) to FMCON, starting the erase-program cycle.
- Read FMCON to check status. If aborted, repeat starting with the LOAD command.

An assembly language routine to load the page register and perform an erase/program operation is shown in Figure 14-2. A similar C-language routine is shown in Figure 14-3.

# FLASH PROGRAM MEMORY

P89LPC901/902/903

FMCON		7	6	5	4	3	2	1	0
Address: E4h		-	-	-	-	HVA	HVE	SV	OI
Not bit addressable									
Reset Source(s): Any reset									
Reset Value:									
BIT	SYMBOL	FUNCTION							
FMCON.7-4	-	Reserved.							
FMCON.3	HVA	Set if either an interrupt or a brown-out is detected during a program or erase cycle. Also set if the brown-out detector is disabled at the start of a program or erase cycle.							
FMCON.2	HVE	High voltage error. Set when an error occurs in the high voltage generator.							
FMCON.1	SV	Security violation. Set when an attempt is made to program, erase, or CRC a secured sector or page.							
FMCON.0	OI	Operation interrupted. Set when cycle aborted due to an interrupt or reset.							

Figure 14-1: Flash Memory Control Register

```

;* Inputs:
;* R3 = number of bytes to program (byte)
;* R4 = page address MSB (byte)
;* R5 = page address LSB (byte)
;* R7 = pointer to data buffer in RAM (byte)
;* Outputs:
;* R7 = status (byte)
;* C = clear on no error, set on error

LOAD EQU 00H
EP EQU 68H

PGM_USER:
    MOV FMCON,#LOAD ;load command, clears page register
    MOV FMADRH,R4 ;get high address
    MOV FMADRL,R5 ;get low address
    MOV A,R7 ;
    MOV R0,A ;get pointer into R0
LOAD_PAGE:
    MOV FMDAT,@R0 ;write data to page register
    INC R0 ;point to next byte
    DJNZ R3,LOAD_PAGE ;do until count is zero
    MOV FMCON,#EP ;else erase & program the page

    MOV R7,FMCON ;copy status for return
    MOV A,R7 ;read status
    ANL A,#0FH ;save only four lower bits
    JNZ BAD ;
    CLR C ;clear error flag if good
    RET ;and return
BAD:
    SETB C ;set error flag
    RET ;and return

```

Figure 14-2: Assembly language routine to erase/program all or part of a page

# FLASH PROGRAM MEMORY

P89LPC901/902/903

```

unsigned char idata dbytes[16];           // data buffer
unsigned char Fm_stat;                   // status result

bit PGM_USER (unsigned char, unsigned char);
bit prog_fail;

void main ()
{
    prog_fail=PGM_USER(0x1F,0xC0);
}

bit PGM_USER (unsigned char page_hi, unsigned char page_lo)
{
    #define LOAD      0x00 // clear page register, enable loading
    #define EP        0x68 // erase & program page
    unsigned char    i;    // loop count

    FMCON = LOAD;           //load command, clears page reg
    FMADRH = page_hi;      //
    FMADRL = page_lo;      //write my page address to addr regs

    for (i=0;i<16;i=i+1)
    {
        FMDATA = dbytes[i];
    }
    FMCON = EP;           //erase & prog page command
    Fm_stat = FMCON;      //read the result status
    if ((Fm_stat & 0x0F)!=0) prog_fail=1; else prog_fail=0;
    return(prog_fail);
}

```

Figure 14-3: C-language routine to erase/program all or part of a page

## Accessing additional flash elements

In addition to the user code array, the user's firmware may access additional flash elements. These include UCFG1, the Boot Vector, Status Bit, and signature bytes. Access of these elements uses a slightly different method than that used to access the user code memory. Signature bytes are read-only. Security bytes may be erased only under certain conditions. IAP-Lite is performed in the application under the control of the microcontroller's firmware using four SFRs to facilitate erasing, programming, or reading. These SFRs are:

- FMCON (Flash Control Register). When read, this is the status register. When written, this is a command register. Note that the status bits are cleared to '0's when the command is written.
- FMDATA (Flash Data Register). Accepts data to be loaded into or from the flash element.
- FMADRL (Flash memory address low). Used to specify the flash element.

The flash elements that may be accessed and their addresses are shown in Table 14-1.



# FLASH PROGRAM MEMORY

P89LPC901/902/903

**Table 14-1: Flash elements accesable through IAP-Lite**

Element	Address	Description
UCFG1	00h	User Configuration byte 1.
Boot Vector	02h	Boot vector
Status Bit	03h	Status bit byte
Security byte 0	08h	Security byte, sector 0
Security byte 1	09h	Security byte, sector 1
Security byte 2	0Ah	Security byte, sector 2
Security byte3	0Bh	Security byte, sector 3
Mfgr Id	10h	Signature byte, manufacturer id
Id_1	11h	Signature byte,id 1
Id_2	12h	Signature byte,id 2

## Erase-programming additional flash elements

The erase-program cycle takes 4ms to complete and is accomplished using the following steps:

- Write the address of the flash element to FMADRL.
- Write the CONF command (6CH) to FMCON.
- Write the data to be programmed to FMDATA.
- Read FMCON to check status. If aborted, repeat this sequence.

Writing the data to be programmed to FMDATA will start the erase-program process and place the CPU in a program-idle state. The CPU will remain in this idle state until the erase-program cycle is either completed or terminated by an interrupt. When the program-idle state is exited, FMCON will contain status information for the cycle.

If an interrupt occurs during an erase/programming cycle, the erase/programming cycle will be aborted and the OI flag (Operation Interrupted) in FMCON will be set. If the application permits interrupts during erasing-programming the user code should check the OI flag (FMCON.0) after each erase-programming operation to see if the operation was aborted. If the operation was aborted, the user's code will need to repeat the process.

## Reading additional flash elements

The read cycle is accomplished using the following steps:

- Write the address of the flash element to FMADRL.
- Write the CONF command (6CH) to FMCON.
- Read the data from FMDATA

The read cycle completes in a single machine cycle and thus will not enter an idle state. It can be interrupted. However, there is no need to check status.

An assembly language routine to perform an erase/program operation of a flash element is shown in Figure 14-4. A similar C-language routine is shown in Figure 14-5. A C-language routine to read a flash element is shown in Figure 14-6.

# FLASH PROGRAM MEMORY

**P89LPC901/902/903**

```

r
; * Inputs: *
; * R5 = data to write(byte) *
; * R7 = element address(byte) *
; * Outputs: *
; * None *

CONF EQU 6CH

WR_ELEM:
MOV FMADRL,R7 ;write the address
MOV FMCON,#CONF ;load CONF command
MOV FMDAT,R5 ;write the data
MOV R7,FMCON ;copy status for return
MOV A,R7 ;read status
ANL A,#0FH ;save only four lower bits
JNZ BAD ;see if good or bad
CLR C ;clear error flag if good
RET ;and return

BAD:
SETB C ;set error flag if bad
RET ;and return

```

**Figure 14-4: Assembly language routine to erase/program a flash element**

```

unsigned char Fm_stat; // status result

bit PGM_EL (unsigned char, unsigned char);
bit prog_fail;

void main ()

{
    prog_fail=PGM_EL(0x02,0x1C);
}

bit PGM_EL (unsigned char el_addr, unsigned char el_data)
{
    #define CONF 0x6C // access flash elements

    FMADRL = el_addr; //write element address to addr reg
    FMCON = CONF; //load command, clears page reg
    FMDATA = el_data; //write data and start the cycle
    Fm_stat = FMCON; //read the result status
    if ((Fm_stat & 0x0F)!=0) prog_fail=1; else prog_fail=0;
    return(prog_fail);
}

```

**Figure 14-5: C-language routine to erase/program a flash element**

# FLASH PROGRAM MEMORY

P89LPC901/902/903

```
#include <REG921.H>

unsigned char READ_EL (unsigned char);
unsigned char GET_EL;

void main ()

{
    GET_EL = READ_EL(0x02);
}

unsigned char READ_EL (unsigned char el_addr)
{
    #define CONF      0x6C          // access flash elements
    unsigned char el_data;        // local for element data
    FMADRL    = el_addr;          //write element address to addr reg
    FMCON     = CONF;             //access flash elements command
    el_data   = FMDATA;           /read the element data
    return(el_data);
}
```

Figure 14-6: C-language routine to read a flash element

## User Configuration Bytes

A number of user-configurable features of the **P89LPC901/902/903** must be defined at power-up and therefore cannot be set by the program after start of execution. These features are configured through the use of Flash byte UCFG1 shown in Figure 14-7.

# FLASH PROGRAM MEMORY

P89LPC901/902/903

UCFG1		7	6	5	4	3	2	1	0
Address: xxxh		WDTE	RPE	BOE	WDSE	-	FOSC2	FOSC1	FOSC0
Default: 63h									
BIT	SYMBOL	FUNCTION							
UCFG1.7	WDTE	Watchdog timer reset enable. When set =1, enables the watchdog timer reset. When cleared = 0 , disables the watchdog timer reset. The timer may still be used to generate an interrupt. Refer to Table 12-1 for details.							
UCFG1.6	RPE	Reset pin enable. When set =1, enables the reset function of pin P1.5. When cleared, P1.5 may be used as an input pin. NOTE: During a power-up sequence, the RPE selection is overridden and this pin will always functions as a reset input. After power-up the pin will function as defined by the RPE bit. Only a power-up reset will temporarily override the selection defined by RPE bit. Other sources of reset will not override the RPE bit.							
UCFG1.5	BOE	Brownout Detect Enable (see section "Brownout Detection").							
UCFG1.4	WDSE	Watchdog Safety Enable bit. Refer to Table for details.							
UCFG1.3	-	Reserved (should remain unprogrammed at zero).							
UCFG1.2-0	FOSC2-FSOC0	CPU oscillator type select. See section "Low Power Select (P89LPC901)" for additional information. Combinations other than those shown below should not be used. They are reserved for future use. When FOSC2:0 select either the internal RC or Watchdog oscillators, the crystal oscillator configuration is controlled by RTCCON. See Table and Table . <b>Note: External clock input and crystal options are available on the P89LPC901.</b>							
	<u>FOSC2-FOSC0</u>	<u>Oscillator Configuration</u>							
		1 1 1	External clock input on XTAL1.						
		1 0 0	Watchdog Oscillator, 400KHz (+20/ -30% tolerance).						
		0 1 1	Internal RC oscillator, 7.373MHz ±2.5%.						
		0 1 0	Low frequency crystal, 20 kHz to 100 kHz.						
		0 0 1	Medium frequency crystal or resonator, 100 kHz to 4 MHz.						
		0 0 0	High frequency crystal or resonator, 4 MHz to 12 MHz.						
Factory default value for UCFG1 is set for watchdog reset disabled, reset pin enabled, brownout detect enabled, and using the internal RC oscillator									

Figure 14-7: Flash User Configuration Byte 1 (UCFG1)

# FLASH PROGRAM MEMORY

P89LPC901/902/903

## USER SECURITY BYTES

There are four User Sector Security Bytes (SEC0, ..., SEC3), each corresponding to one sector and having the following bit assignments:

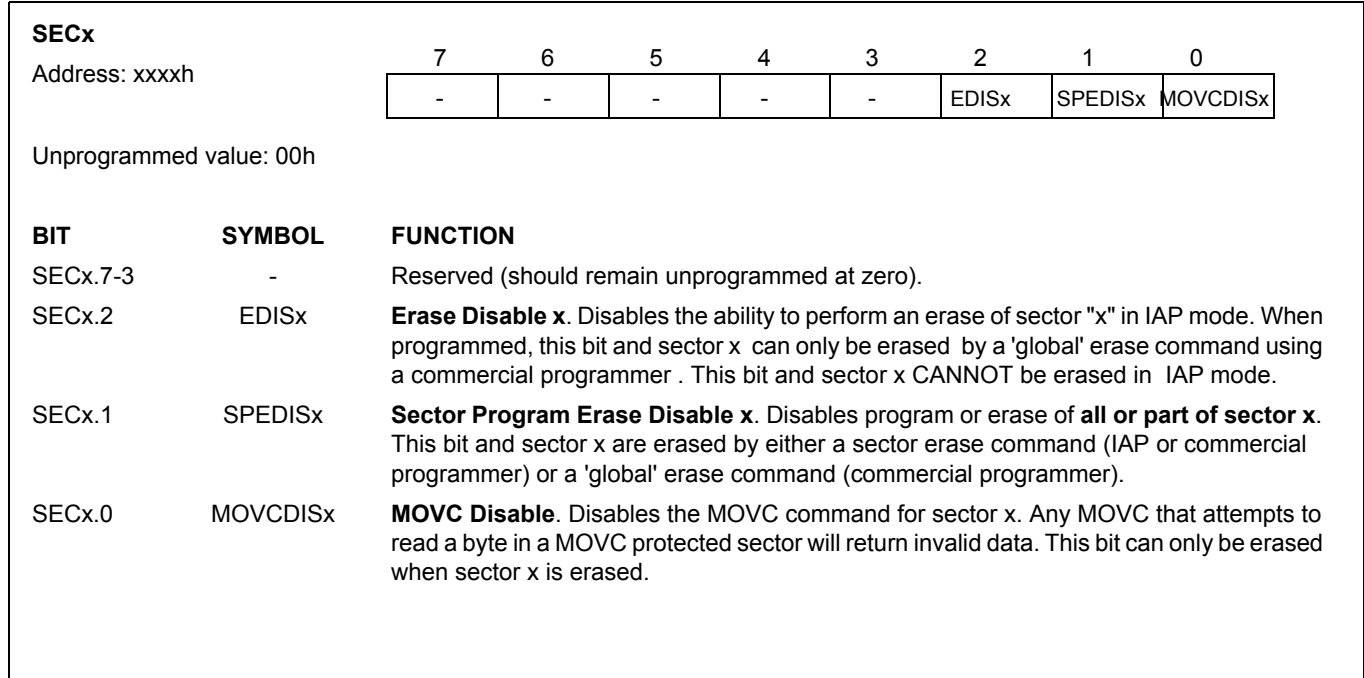


Figure 14-8: User Sector Security Bytes (SEC0 ... SEC3)

Table 14-2: Effects of Security Bits

EDISx	SPEDISx	MOVCDISx	Effects on Programming
0	0	0	None.
0	0	1	Security violation flag set for sector CRC calculation for the specific sector. Security violation flag set for global CRC calculation if any MOVCDISx bit is set. Cycle aborted. Memory contents unchanged. CRC invalid. Program/erase commands will not result in a security violation.
0	1	x	Security violation flag set for program commands or an erase page command. Cycle aborted. Memory contents unchanged. Sector erase and global erase are allowed.
1	x	x	Security violation flag set for program or erase commands. Cycle aborted. Memory contents unchanged. Global erase is allowed.

# FLASH PROGRAM MEMORY

P89LPC901/902/903

## Boot Vector

<b>BOOTVEC</b>	7	6	5	4	3	2	1	0
Address: xxxxh	-	-	-	BOOTV4	BOOTV3	BOOTV2	BOOTV1	BOOTV0

Factory default value: 00h

BIT	SYMBOL	FUNCTION
BOOTVEC.7-5	-	Reserved (should remain unprogrammed at zero).
BOOTVEC.4-0	-	Boot Vector. If the Boot Vector is selected as the reset address, the <b>P89LPC901/902/903</b> will start execution at an address comprised of 00H in the lower eight bits and this BOOTVEC as the upper bits after a reset. (See section "Power-On reset code execution" on page 73).

Figure 14-9: Boot Vector (BOOTVEC)

## Boot Status

<b>BOOTSTAT</b>	7	6	5	4	3	2	1	0
Address: xxxxh	-	-	-	-	-	-	-	BSB

Factory default value: 00h

BIT	SYMBOL	FUNCTION
BOOTSTAT.7-1	-	Reserved (should remain unprogrammed at zero).
BOOTSTAT.0	BSB	Boot Status Bit. If programmed to '1', the <b>P89LPC901/902/903</b> will always start execution at an address comprised of 00H in the lower eight bits and BOOTVEC as the upper bits after a reset. (See section "Power-On reset code execution" on page 73).

Figure 14-10: Boot Status (BOOTSTAT)

**INSTRUCTION SET****P89LPC901/902/903****15. INSTRUCTION SET****Table 15-1: Instruction set summary**

<b>Mnemonic</b>	<b>Description</b>	<b>Bytes</b>	<b>Cycles</b>	<b>Hex code</b>
<b>ARITHMETIC</b>				
ADD A,Rn	Add register to A	1	1	28-2F
ADD A,dir	Add direct byte to A	2	1	25
ADD A,@Ri	Add indirect memory to A	1	1	26-27
ADD A,#data	Add immediate to A	2	1	24
ADDC A,Rn	Add register to A with carry	1	1	38-3F
ADDC A,dir	Add direct byte to A with carry	2	1	35
ADDC A,@Ri	Add indirect memory to A with carry	1	1	36-37
ADDC A,#data	Add immediate to A with carry	2	1	34
SUBB A,Rn	Subtract register from A with borrow	1	1	98-9F
SUBB A,dir	Subtract direct byte from A with borrow	2	1	95
SUBB A,@Ri	Subtract indirect memory from A with borrow	1	1	96-97
SUBB A,#data	Subtract immediate from A with borrow	2	1	94
INC A	Increment A	1	1	04
INC Rn	Increment register	1	1	08-0F
INC dir	Increment direct byte	2	1	05
INC @Ri	Increment indirect memory	1	1	06-07
DEC A	Decrement A	1	1	14
DEC Rn	Decrement register	1	1	18-1F
DEC dir	Decrement direct byte	2	1	15
DEC @Ri	Decrement indirect memory	1	1	16-17
INC DPTR	Increment data pointer	1	2	A3
MUL AB	Multiply A by B	1	4	A4
DIV AB	Divide A by B	1	4	84
DA A	Decimal Adjust A	1	1	D4
<b>LOGICAL</b>				
ANL A,Rn	AND register to A	1	1	58-5F
ANL A,dir	AND direct byte to A	2	1	55
ANL A,@Ri	AND indirect memory to A	1	1	56-57
ANL A,#data	AND immediate to A	2	1	54

**INSTRUCTION SET****P89LPC901/902/903**

<b>Mnemonic</b>	<b>Description</b>	<b>Bytes</b>	<b>Cycles</b>	<b>Hex code</b>
ANL dir,A	AND A to direct byte	2	1	52
ANL dir,#data	AND immediate to direct byte	3	2	53
ORL A,Rn	OR register to A	1	1	48-4F
ORL A,dir	OR direct byte to A	2	1	45
ORL A,@Ri	OR indirect memory to A	1	1	46-47
ORL A,#data	OR immediate to A	2	1	44
ORL dir,A	OR A to direct byte	2	1	42
ORL dir,#data	OR immediate to direct byte	3	2	43
XRL A,Rn	Exclusive-OR register to A	1	1	68-6F
XRL A,dir	Exclusive-OR direct byte to A	2	1	65
XRL A, @Ri	Exclusive-OR indirect memory to A	1	1	66-67
XRL A,#data	Exclusive-OR immediate to A	2	1	64
XRL dir,A	Exclusive-OR A to direct byte	2	1	62
XRL dir,#data	Exclusive-OR immediate to direct byte	3	2	63
CLR A	Clear A	1	1	E4
CPL A	Complement A	1	1	F4
SWAP A	Swap Nibbles of A	1	1	C4
RL A	Rotate A left	1	1	23
RLC A	Rotate A left through carry	1	1	33
RR A	Rotate A right	1	1	03
RRC A	Rotate A right through carry	1	1	13
	<b>DATA TRANSFER</b>			
MOV A,Rn	Move register to A	1	1	E8-EF
MOV A,dir	Move direct byte to A	2	1	E5
MOV A,@Ri	Move indirect memory to A	1	1	E6-E7
MOV A,#data	Move immediate to A	2	1	74
MOV Rn,A	Move A to register	1	1	F8-FF
MOV Rn,dir	Move direct byte to register	2	2	A8-AF
MOV Rn,#data	Move immediate to register	2	1	78-7F
MOV dir,A	Move A to direct byte	2	1	F5
MOV dir,Rn	Move register to direct byte	2	2	88-8F
MOV dir,dir	Move direct byte to direct byte	3	2	85
MOV dir,@Ri	Move indirect memory to direct byte	2	2	86-87



**INSTRUCTION SET****P89LPC901/902/903**

<b>Mnemonic</b>	<b>Description</b>	<b>Bytes</b>	<b>Cycles</b>	<b>Hex code</b>
MOV dir,#data	Move immediate to direct byte	3	2	75
MOV @Ri,A	Move A to indirect memory	1	1	F6-F7
MOV @Ri,dir	Move direct byte to indirect memory	2	2	A6-A7
MOV @Ri,#data	Move immediate to indirect memory	2	1	76-77
MOV DPTR,#data	Move immediate to data pointer	3	2	90
MOVC A,@A+DPTR	Move code byte relative DPTR to A	1	2	93
MOVC A,@A+PC	Move code byte relative PC to A	1	2	94
MOVX A,@Ri	Move external data(A8) to A	1	2	E2-E3
MOVX A,@DPTR	Move external data(A16) to A	1	2	E0
MOVX @Ri,A	Move A to external data(A8)	1	2	F2-F3
MOVX @DPTR,A	Move A to external data(A16)	1	2	F0
PUSH dir	Push direct byte onto stack	2	2	C0
POP dir	Pop direct byte from stack	2	2	D0
XCH A,Rn	Exchange A and register	1	1	C8-CF
XCH A,dir	Exchange A and direct byte	2	1	C5
XCH A,@Ri	Exchange A and indirect memory	1	1	C6-C7
XCHD A,@Ri	Exchange A and indirect memory nibble	1	1	D6-D7
	<b>BOOLEAN</b>			
<b>Mnemonic</b>	<b>Description</b>	<b>Bytes</b>	<b>Cycles</b>	<b>Hex code</b>
CLR C	Clear carry	1	1	C3
CLR bit	Clear direct bit	2	1	C2
SETB C	Set carry	1	1	D3
SETB bit	Set direct bit	2	1	D2
CPL C	Complement carry	1	1	B3
CPL bit	Complement direct bit	2	1	B2
ANL C,bit	AND direct bit to carry	2	2	82
ANL C,/bit	AND direct bit inverse to carry	2	2	B0
ORL C,bit	OR direct bit to carry	2	2	72
ORL C,/bit	OR direct bit inverse to carry	2	2	A0
MOV C,bit	Move direct bit to carry	2	1	A2
MOV bit,C	Move carry to direct bit	2	2	92
	<b>BRANCHING</b>			

**INSTRUCTION SET****P89LPC901/902/903**

<b>Mnemonic</b>	<b>Description</b>	<b>Bytes</b>	<b>Cycles</b>	<b>Hex code</b>
ACALL addr 11	Absolute jump to subroutine	2	2	116F1
LCALL addr 16	Long jump to subroutine	3	2	12
RET	Return from subroutine	1	2	22
RETI	Return from interrupt	1	2	32
AJMP addr 11	Absolute jump unconditional	2	2	016E1
LJMP addr 16	Long jump unconditional	3	2	02
SJMP rel	Short jump (relative address)	2	2	80
JC rel	Jump on carry = 1	2	2	40
JNC rel	Jump on carry = 0	2	2	50
JB bit,rel	Jump on direct bit = 1	3	2	20
JNB bit,rel	Jump on direct bit = 0	3	2	30
JBC bit,rel	Jump on direct bit = 1 and clear	3	2	10
JMP @A+DPTR	Jump indirect relative DPTR	1	2	73
JZ rel	Jump on accumulator = 0	2	2	60
JNZ rel	Jump on accumulator $\neq$ 0	2	2	70
CJNE A,dir,rel	Compare A,direct jne relative	3	2	B5
CJNE A,#d,rel	Compare A,immediate jne relative	3	2	B4
CJNE Rn,#d,rel	Compare register, immediate jne relative	3	2	B8-BF
CJNE @Ri,#d,rel	Compare indirect, immediate jne relative	3	2	B6-B7
DJNZ Rn,rel	Decrement register, jnz relative	2	2	D8-DF
DJNZ dir,rel	Decrement direct byte, jnz relative	3	2	D5
	<b>MISCELLANEOUS</b>			
NOP	No operation	1	1	00

---

# REVISION HISTORY

---

**P89LPC901/902/903**

## 16. REVISION HISTORY

2003 Dec 8

Removed ENCLK bit from P89LPC902 and P89LPC903 TRIM SFRs. Modified Fig 2-1 to reflect ENCLK only on the P89LPC901. Removed RCCLK from TRIM SFR description (Fig2-1).

Changed comparator SFR and bit names to match 89LPC9xx product line terminology.

Added Revision History chapter.

Added instruction set chapter.

Added information that disabling comparator could cause an interrupt if comparator output was low when disabled and comparator interrupt is enabled.

Added information regarding  $V_{POR}$  specifications.

Changed KBI on interrupt figure to KBIF.

New WDT description replaces previous to correct technical information.

Added comment regarding direction of RC oscillator change when changing TRIM value.

Added information regarding Bootvector and Status Bit and their factory default values.

Corrected a mistake in SECx table description regarding function of the security bits.

Added information that an active interrupt will abort the IAP Lite programming/erase process.

---

# REVISION HISTORY

---

P89LPC901/902/903

---

**INDEX**

---

**P89LPC901/902/903**

---

**17. INDEX****A**

- Analog comparators 41, 75
  - configuration 75
  - configuration example 78
  - enabling 75
  - internal reference voltage 83
  - interrupt 77
  - power reduction modes 78
- Analog comparators and power reduction 41

**B**

- Block diagram 9
- BRGCON
  - writing to 24
- Brownout detection 55
  - enabling and disabling 55
  - operating range 55
  - options 56
  - rise and fall times of Vdd 55

**C**

- CLKLP 33
- Clock
  - CPU clock 27
  - CPU divider (DIVM) 32
  - definitions 27
  - external input option 29
  - PCLK 27
  - RCCLK 27
  - wakeup delay 32
- Clock output 28

**D**

- Data EEPROM

---

**INDEX****P89LPC901/902/903**

---

block fill 7, 27, 35, 39, 45, 51, 55, 61, 73, 75, 79, 83, 91, 93, 103, 107

hardware reset 7, 27, 35, 39, 45, 51, 55, 61, 73, 75, 79, 83, 91, 93, 103, 107

Dual Data Pointers 91

**F**

FLASH 7, 27, 35, 39, 45, 51, 55, 61, 73, 75, 79, 83, 91, 93, 103, 107

  Boot Status 102

  Boot Vector 102

  features 93

  hardware activation of the boot loader 73

  power-on reset code execution 73

**I**

IAP programming 93

Interrupts 39

  arbitration ranking 35

  external input pin glitch suppression 37

  external inputs 36

  keypad 37

  priority structure 35

  wake-up from power-down 37

Interrutps

  edge-triggered 37

ISP programming 93

**K**

Keypad interrupt (KBI) 83

**L**

Low power (CLKLP) 33

**M**

Memory

  Code 25

  Data 25

# INDEX

**P89LPC901/902/903**

FLASH code 93  
organization 25

## O

### Oscillator

high speed crystal option 27, 28  
low speed crystal option 27  
medium speed crystal option 27  
R-C option 28  
watchdog (WDT) option 29

## P

Pin configuration 7, 8

Port 0 12, 14, 15

Port 3 12

### Ports

additional features 42  
I/O 39  
input only configuration 41  
open drain output configuration 40  
Port 0 analog functions 41  
Port 2 in 20-pin package 41  
push-pull output configuration 41  
quasi-bidirectional output configuration 39

Power monitoring functions 73

Power reduction modes 57

normal mode 57  
power down mode (partial) 57

Power-down mode (total) 57

Power-on detection 56

## R

Real time clock 51

clock sources 51  
interrupt/wake up 53

Reset 73

enabling the external reset input pin 73, 100  
software reset 91  
sources 73

---

**INDEX**

---

**P89LPC901/902/903**

---

**S****SFR**

- AUXR1 91
- BRGCON 63
- CMPn 75
- KBCON 80
- KBMASK 80, 81
- KBPATN 79, 80
- PCON 58
- PCONA 59
- RSTSRC 74
- RTCCON 54
- SCON 64
- SSTAT 65
- TAMOD 46
- TCON 47
- TMOD 45
- TRIM 28, 29, 95
- UCFG1 100
- WDCON 85

**SFRs**

- undefined locations, use of 16

Special Function Registers (SFR) table 16, 19, 22

**T****Timer/counters 45**

- mode 0 46
- mode 1 46
- mode 2 (8-bit auto reload) 46
- mode 3 (seperates TL0 & TH0) 47
- mode 6 (8-bit PWM) 47
- toggle output 49

**TRIM (SFR)**

- power-on reset value 24

**U****UART 61**

- automatic address recognition 70
- baud rate generator 62
- BRGR1 and BRGR0, updating 62



---

**INDEX****P89LPC901/902/903**

---

- double buffering in 9-bit mode 69
- double buffering in different modes 68
- framing error 63, 67
- mode 0 65
- mode 0 (shift register) 61
- mode 1 66
- mode 1 (8-bit variable baud rate) 61
- mode 2 67
- mode 2 (9-bit fixed baud rate) 61
- mode 3 67
- mode 3 (9-bit variable baud rate) 61
- multiprocessor communications 70
- SFR locations. 62
- status register 65
- transmit interrupts with double buffering enabled (modes 1, 2 and 3) 68

**W**

- Watchdog timer 83
  - feed sequence 84
  - timer mode 87
  - watchdog function 83
  - watchdog timeout values 86
  - WDCLK = 0 and CPU power down 87

## Definitions

**Short-form specification** — The data in a short-form specification is extracted from a full data sheet with the same type number and title. For detailed information see the relevant data sheet or data handbook.

**Limiting values definition** — Limiting values given are in accordance with the Absolute Maximum Rating System (IEC 60134). Stress above one or more of the limiting values may cause permanent damage to the device. These are stress ratings only and operation of the device at these or at any other conditions above those given in the Characteristics sections of the specification is not implied. Exposure to limiting values for extended periods may affect device reliability.

**Application information** — Applications that are described herein for any of these products are for illustrative purposes only. Philips Semiconductors make no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

## Disclaimers

**Life support** — These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Philips Semiconductors customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Philips Semiconductors for any damages resulting from such application.

**Right to make changes** — Philips Semiconductors reserves the right to make changes in the products—including circuits, standard cells, and/or software—described or contained herein in order to improve design and/or performance. When the product is in full production (status 'Production'), relevant changes will be communicated via a Customer Product/Process Change Notification (CPCN). Philips Semiconductors assumes no responsibility or liability for the use of any of these products, conveys no license or title under any patent, copyright, or mask work right to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

---

## Contact information

For additional information please visit  
<http://www.semiconductors.philips.com>. Fax: +31 40 27 24825

© Koninklijke Philips Electronics N.V. 2003  
All rights reserved. Printed in U.S.A.

For sales offices addresses send e-mail to:  
[sales.addresses@www.semiconductors.philips.com](mailto:sales.addresses@www.semiconductors.philips.com)

Date of release: 12-03  
Document order number: 9397 750 12489

*Let's make things better.*