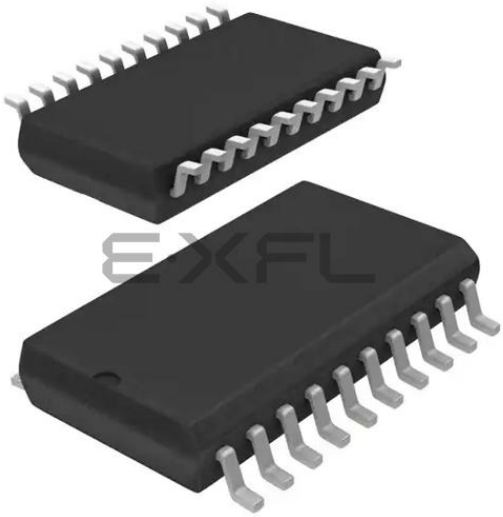


Welcome to E-XFL.COM

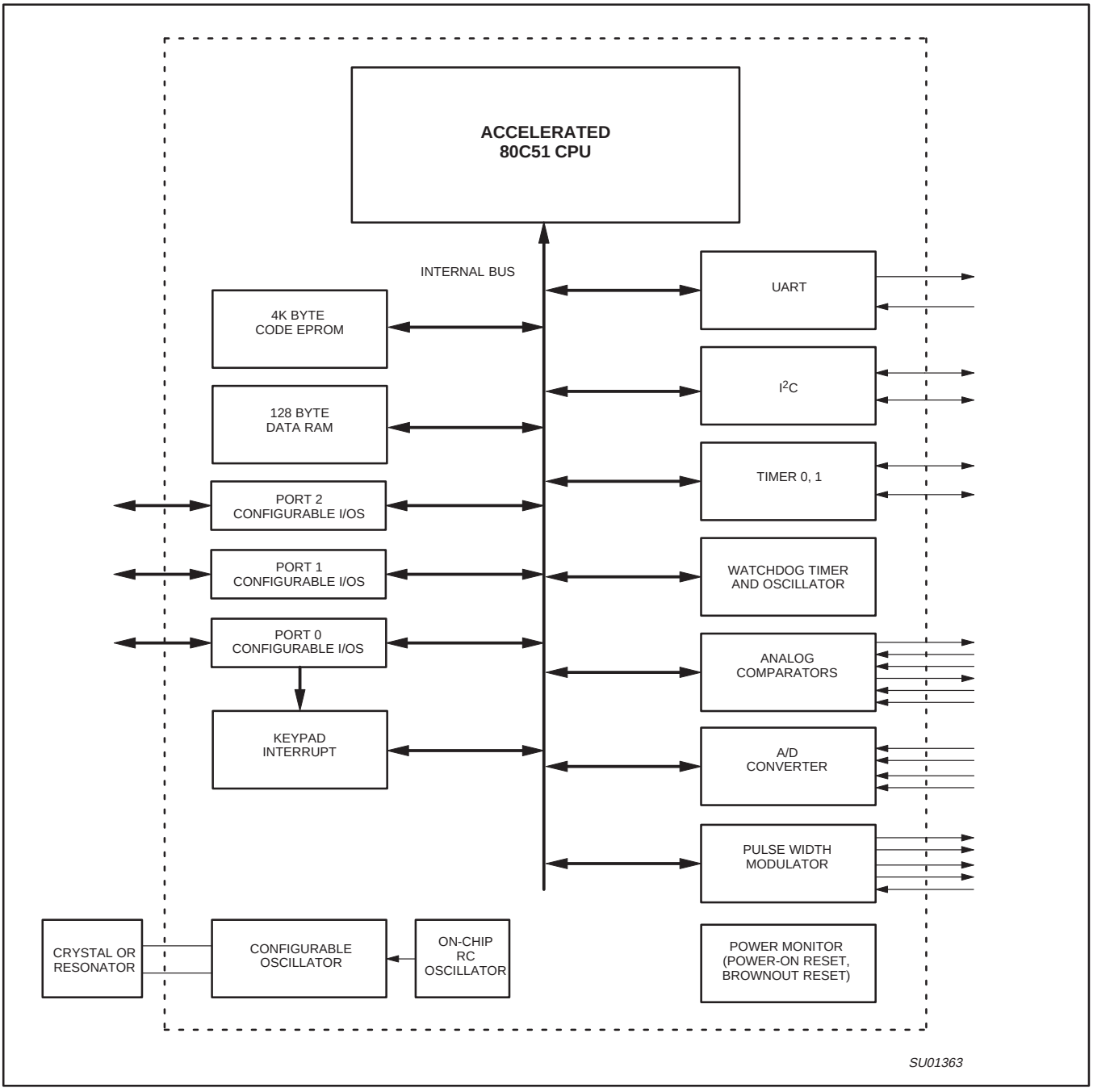
What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	18
Program Memory Size	4KB (4K x 8)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 6V
Data Converters	A/D 4x8b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SO
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p87lpc768bd-512

BLOCK DIAGRAM



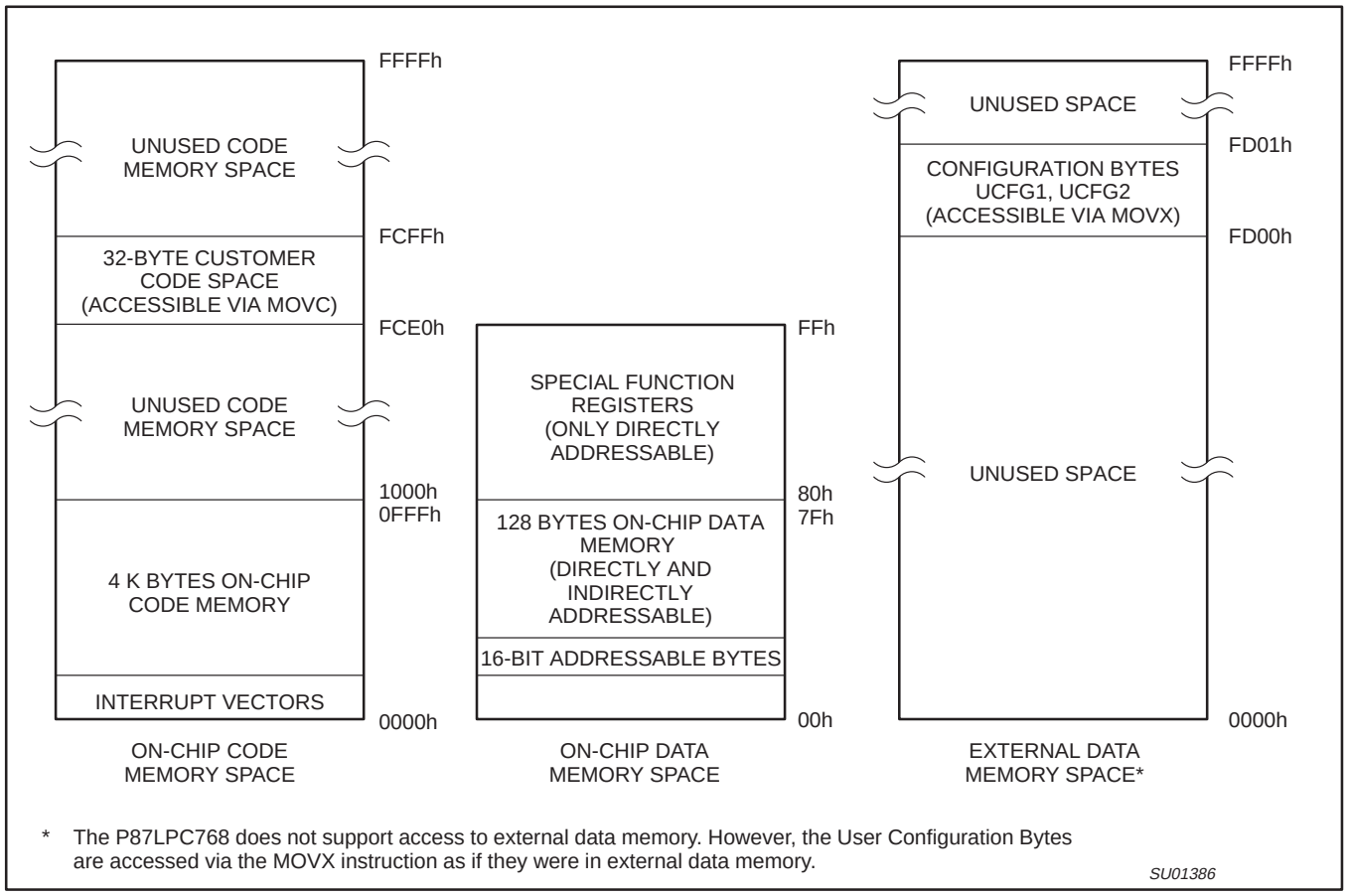


Figure 1. P87LPC768 Program and Data Memory Map

MNEMONIC	PIN NO.	TYPE	NAME AND FUNCTION
P2.0–P2.1	6, 7	I/O	**Port 2**: Port 2 is a 2-bit I/O port with a user-configurable output type. Port 2 latches are configured in the quasi-bidirectional mode and have either ones or zeros written to them during reset, as determined by the PRHI bit in the UCFG1 configuration byte. The operation of port 2 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. Port 2 also provides various special functions as described below.
	7	O	**P2.0** **X2** Output from the oscillator amplifier (when a crystal oscillator option is selected via the EPROM configuration). **CLKOUT** CPU clock divided by 6 clock output when enabled via SFR bit and in conjunction with internal RC oscillator or external clock input.
	6	I	**P2.1** **X1** Input to the oscillator circuit and internal clock generator circuits (when selected via the EPROM configuration).
V_{SS}	5	I	**Ground**: 0V reference.
V_{DD}	15	I	**Power Supply**: This is the power supply voltage for normal operation as well as Idle and Power Down modes.

SPECIAL FUNCTION REGISTERS

Name	Description	SFR Address	Bit Functions and Addresses MSB							LSB	Reset Value
			E7	E6	E5	E4	E3	E2	E1	E0	
ACC*	Accumulator	E0h									00h
			C7	C6	C5	C4	C3	C2	C1	C0	
ADCON#*	A/D Control	C0h	ENADC	–	–	ADCI	ADCS	RCCLK	AADR1	AADR0	00h
AUXR1#	Auxiliary Function Register	A2h	KBF	BOD	BOI	LPEP	SRST	0	–	DPS	02h[1]
			F7	F6	F5	F4	F3	F2	F1	F0	
B*	B register	F0h									00h
CMP1#	Comparator 1 control register	ACh	–	–	CE1	CP1	CN1	OE1	CO1	CMF1	00h[1]
CMP2#	Comparator 2 control register	ADh	–	–	CE2	CP2	CN2	OE2	CO2	CMF2	00h[1]
CNSW0	PWM Counter Shadow Register 0	D1h	CNSW7	CNSW6	CNSW5	CNSW4	CNSW3	CNSW2	CNSW1	CNSW0	FFh
CNSW1	PWM Counter Shadow Register 1	D2h	–	–	–	–	–	–	CNSW9	CNSW8	FFh
CPSW0	PWM Compare Shadow Register 0	D3h	CPSW07	CPSW06	CPSW05	CPSW04	CPSW03	CPSW02	CPSW01	CPSW00	00h
CPSW1	PWM Compare Shadow Register 1	D4h	CPSW17	CPSW16	CPSW15	CPSW14	CPSW13	CPSW12	CPSW11	CPSW10	00h
CPSW2	PWM Compare Shadow Register 2	D5h	CPSW27	CPSW26	CPSW25	CPSW24	CPSW23	CPSW22	CPSW21	CPSW20	00h
CPSW3	PWM Compare Shadow Register 3	D6h	CPSW37	CPSW36	CPSW35	CPSW34	CPSW33	CPSW32	CPSW31	CPSW30	00h
CPSW4	PWM Compare Shadow Register 4	D7h	CPSW39	CPSW38	CPSW29	CPSW28	CPSW19	CPSW18	CPSW09	CPSW08	00h
DAC0#	A/D Result	C5h									00h
DIVM#	CPU clock divide-by-M control	95h									00h
DPTR:	Data pointer (2 bytes)										
DPH	Data pointer high byte	83h									00h
DPL	Data pointer low byte	82h									00h
			CF	CE	CD	CC	CB	CA	C9	C8	
I2CFG#*	I^2C configuration register	C8h/RD	SLAVEN	MASTRQ	0	TIRUN	–	–	CT1	CT0	00h[1]
		C8h/WR	SLAVEN	MASTRQ	CLRTI	TIRUN	–	–	CT1	CT0	
			DF	DE	DD	DC	DB	DA	D9	D8	
I2CON#*	I^2C control register	D8h/RD	RDAT	ATN	DRDY	ARL	STR	STP	MASTER	–	80h[1]
		D8h/WR	CXA	IDLE	CDR	CARL	CSTR	CSTP	XSTR	XSTP	
I2DAT#	I^2C data register	D9h/RD	RDAT	0	0	0	0	0	0	0	80h
		D9h/WR	XDAT	x	x	x	x	x	x	x	
			AF	AE	AD	AC	AB	AA	A9	A8	
IEN0*	Interrupt enable 0	A8h	EA	EWD	EBO	ES	ET1	EX1	ET0	EX0	00h
			EF	EE	ED	EC	EB	EA	E9	E8	
IEN1#*	Interrupt enable 1	E8h	ETI	–	EC1	EAD	–	EC2	EKB	EI2	00h[1]
			BF	BE	BD	BC	BB	BA	B9	B8	
IP0*	Interrupt priority 0	B8h	–	PWD	PBO	PS	PT1	PX1	PT0	PX0	00h[1]
IP0H#	Interrupt priority 0 high byte	B7h	–	PWDH	PBOH	PSH	PT1H	PX1H	PT0H	PX0H	00h[1]
			FF	FE	FD	FC	FB	FA	F9	F8	
IP1*	Interrupt priority 1	F8h	PTI	–	PC1	PAD	–	PC2	PKB	PI2	00h[1]
IP1H#	Interrupt priority 1 high byte	F7h	PTIH	–	PC1H	PADH	–	PC2H	PKBH	PI2H	00h[1]

Analog Comparators

Two analog comparators are provided on the P87LPC768. Input and output options allow use of the comparators in a number of different configurations. Comparator operation is such that the output is a logical one (which may be read in a register and/or routed to a pin) when the positive input (one of two selectable pins) is greater than the negative input (selectable from a pin or an internal reference voltage). Otherwise the output is a zero. Each comparator may be configured to cause an interrupt when the output value changes.

Comparator Configuration

Each comparator has a control register, CMP1 for comparator 1 and CMP2 for comparator 2. The control registers are identical and are shown in Figure 4.

The overall connections to both comparators are shown in Figure 5. There are eight possible configurations for each comparator, as determined by the control bits in the corresponding CMPn register: CPn, CNn, and OEn. These configurations are shown in Figure 6. The comparators function down to a V_{DD} of 3.0V.

When each comparator is first enabled, the comparator output and interrupt flag are not guaranteed to be stable for 10 microseconds. The corresponding 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.

CMPn Address: ACh for CMP1, ADh for CMP2 Reset Value: 00h

Not Bit Addressable

7	6	5	4	3	2	1	0
—	—	CEn	CPn	CNn	OEn	COn	CMFn

BIT	SYMBOL	FUNCTION
CMPn.7, 6	—	Reserved for future use. Should not be set to 1 by user programs.
CMPn.5	CEn	Comparator enable. When set by software, the corresponding comparator function is enabled. Comparator output is stable 10 microseconds after CEn is first set.
CMPn.4	CPn	Comparator positive input select. When 0, CINnA is selected as the positive comparator input. When 1, CINnB is selected as the positive comparator input.
CMPn.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 V_{ref} is selected as the negative comparator input.
CMPn.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.
CMPn.1	COn	Comparator output, synchronized to the CPU clock to allow reading by software. Cleared when the comparator is disabled (CEn = 0).
CMPn.0	CMFn	Comparator interrupt flag. This bit is set by hardware whenever the comparator output COn changes state. This bit will cause a hardware interrupt if enabled and of sufficient priority. Cleared by software and when the comparator is disabled (CEn = 0).

SU01152

Figure 4. Comparator Control Registers (CMP1 and CMP2)

Internal Reference Voltage
An internal reference voltage generator may supply a default reference when a single comparator input pin is used. The value of the internal reference voltage, referred to as V_{ref}, is 1.28 V ±10%.

Comparator Interrupt
Each 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 corresponding enable bit ECn in the IEN1 register is set and the interrupt system is enabled via the EA bit in the IEN0 register.

Comparators and Power Reduction Modes
Either or both comparators may remain enabled when Power Down or Idle mode is activated. The comparators will continue to function in the power reduction mode. If a comparator interrupt is enabled, 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 pull-up that normally occurs during switching on a quasi-bidirectional port pin does not take place.

Comparators consume 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.

Comparator Configuration Example
The code shown in Figure 7 is an example of initializing one comparator. Comparator 1 is configured to use the CIN1A and CMPREF inputs, outputs the comparator result to the CMP1 pin, and generates an interrupt when the comparator output changes.

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

```
CmpInit:
    mov     PT0AD,#30h      ; Disable digital inputs on pins that are used
                            ;   for analog functions: CIN1A, CMPREF.
    anl     P0M2,#0cfh      ; Disable digital outputs on pins that are used
    orl     P0M1,#30h       ;   for analog functions: CIN1A, CMPREF.
    mov     CMP1,#24h       ; Turn on comparator 1 and set up for:
                            ;   – Positive input on CIN1A.
                            ;   – Negative input from CMPREF pin.
                            ;   – Output to CMP1 pin enabled.
    call    delay10us       ; The comparator has to start up for at
                            ;   least 10 microseconds before use.
    anl     CMP1,#0feh      ; Clear comparator 1 interrupt flag.
    setb    EC1             ; Enable the comparator 1 interrupt. The
                            ;   priority is left at the current value.
    setb    EA              ; Enable the interrupt system (if needed).
    ret                     ; Return to caller.
```

SU01189

Figure 7.

Pulse Width Modulator
The P87LPC768 contains four Pulse Width Modulated (PWM) channels which generate pulses of programmable length and interval. The output for PWM0 is on P0.1, PWM1 on P1.6, PWM2 on P1.7 and PWM3 on P0.0. After chip reset the internal output of the each PWM channel is a “1.” Note that the state of the pin will not reflect this if UCFG1.5, PRHI, is set to a zero. In this case before the pin will reflect the state of the internal PWM output a “1” must be written to each port bit that serves as a PWM output. A block diagram is shown in Figure 8.

The interval between successive outputs is controlled by a 10–bit down counter which uses the internal microcontroller clock as its input. When bit 3 in the UCFG1 register is a “1” the microcontroller clock, and therefore the PWM counter clock, has the same frequency as the clock source $F_{CPWM} = F_{OSC}$. When bit 3 in the UCFG1 register is a “0” the microcontroller and PWM counter clocks operate at half the frequency of clock source, $F_{CPWM} = F_{OSC}/2$. When the counter reaches underflow it is reloaded with a user selectable value. This mechanism allows the user to set the PWM frequency at any integer sub–multiple of the microcontroller clock frequency. The repetition frequency of the PWM is given by:

$$f_{PWM} = F_{CPWM} / (CNSW+1)$$

where CNSW is contained in CNSW0 and CNSW1 as described in the following tables.

CNSW0: Counter Shadow register 0

Addr: 0D1H

Reset Value: FFH

7	6	5	4	3	2	1	0
CNSW7	CNSW6	CNSW5	CNSW4	CNSW3	CNSW2	CNSW1	CNSW0

CNSW1: Counter Shadow register 1

Addr: 0D2H

Reset Value: FFH

7	6	5	4	3	2	1	0
Unused	Unused	Unused	Unused	Unused	Unused	CNSW9	CNSW8

The word “Shadow” in the above refers to the fact that writes are not into the register that controls the counter; rather they are into a holding register. As described below the transfer of data from this holding register, into the register which contains the actual reload value, is controlled by the user’s program.

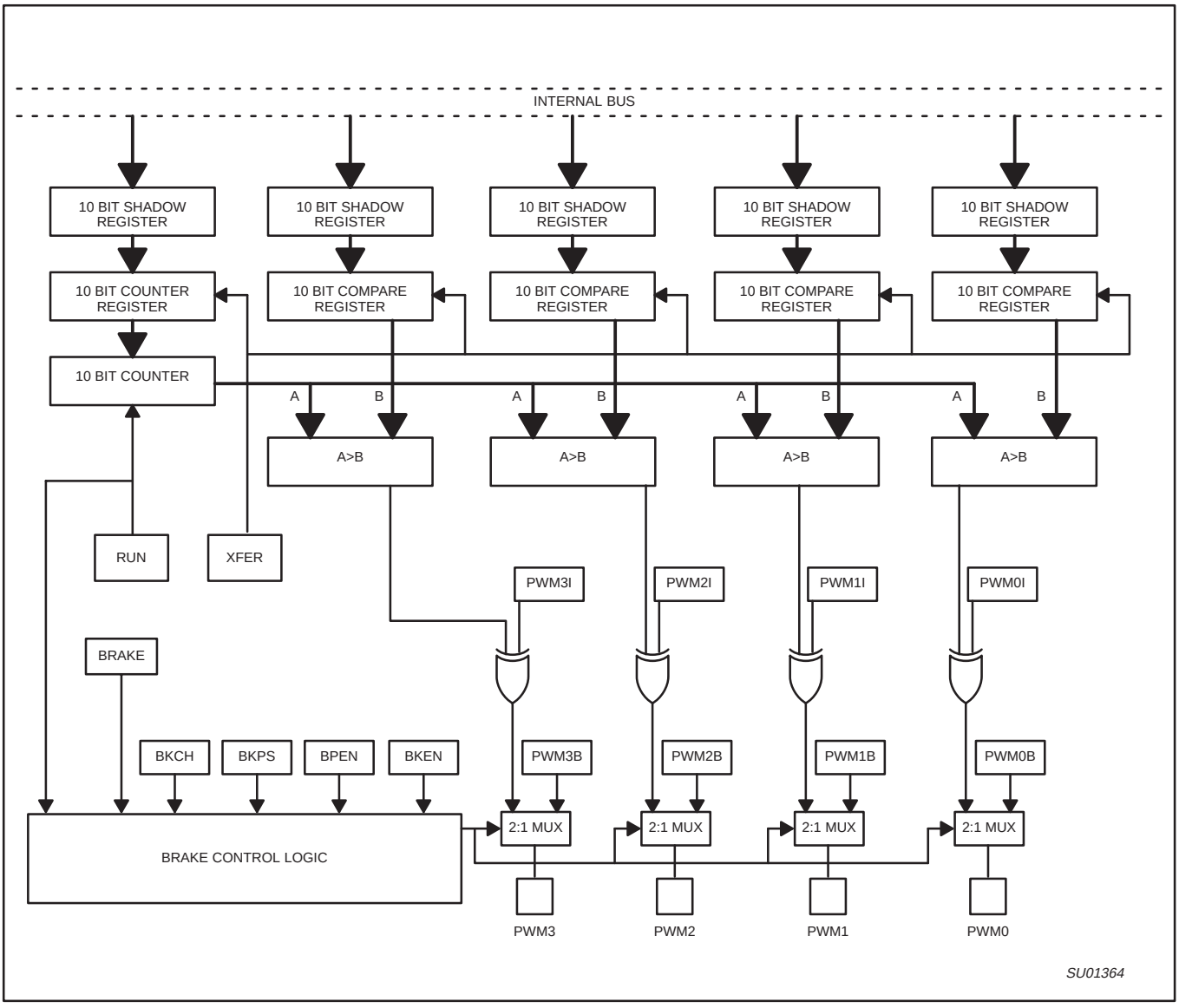


Figure 8. PWM Block Diagram

I²C Serial Interface

The I²C bus uses two wires (SDA and SCL) to transfer information between devices connected to the bus. The main features of the bus are:

- Bidirectional data transfer between masters and slaves.
- Serial addressing of slaves (no added wiring).
- Acknowledgment after each transferred byte.
- Multimaster bus.
- Arbitration between simultaneously transmitting masters without corruption of serial data on bus.

The I²C subsystem includes hardware to simplify the software required to drive the I²C bus. The hardware is a single bit interface which in addition to including the necessary arbitration and framing error checks, includes clock stretching and a bus timeout timer. The interface is synchronized to software either through polled loops or interrupts.

Refer to the application note AN422, entitled “Using the 8XC751 Microcontroller as an I²C Bus Master” for additional discussion of the 8xC76x I²C interface and sample driver routines.

The P87LPC768 I2C implementation duplicates that of the 87C751 and 87C752 except for the following details:

- The interrupt vector addresses for both the I²C interrupt and the Timer I interrupt.
- The I²C SFR addresses (I2CON, !2CFG, I2DAT).
- The location of the I²C interrupt enable bit and the name of the SFR it is located within (EI2 is Bit 0 in IEN1).
- The location of the Timer I interrupt enable bit and the name of the SFR it is located within (ETI is Bit 7 in IEN1).
- The I²C and Timer I interrupts have a settable priority.

Timer I is used to both control the timing of the I²C bus and also to detect a “bus locked” condition, by causing an interrupt when nothing happens on the I²C bus for an inordinately long period of time while a transmission is in progress. If this interrupt occurs, the program has the opportunity to attempt to correct the fault and resume I²C operation.

Six time spans are important in I²C operation and are insured by timer I:

- The MINIMUM HIGH time for SCL when this device is the master.
- The MINIMUM LOW time for SCL when this device is a master. This is not very important for a single-bit hardware interface like this one, because the SCL low time is stretched until the software responds to the I²C flags. The software response time normally meets or exceeds the MIN LO time. In cases where the software responds within MIN HI + MIN LO) time, timer I will ensure that the minimum time is met.
- The MINIMUM SCL HIGH TO SDA HIGH time in a stop condition.
- The MINIMUM SDA HIGH TO SDA LOW time between I²C stop and start conditions (4.7ms, see I²C specification).
- The MINIMUM SDA LOW TO SCL LOW time in a start condition.
- The MAXIMUM SCL CHANGE time while an I²C frame is in progress. A frame is in progress between a start condition and the following stop condition. This time span serves to detect a lack of software response on this device as well as external I²C problems. SCL “stuck low” indicates a faulty master or slave. SCL “stuck high” may mean a faulty device, or that noise induced onto the I²C bus caused all masters to withdraw from I²C arbitration.

The first five of these times are 4.7 ms (see I²C specification) and are covered by the low order three bits of timer I. Timer I is clocked by the P87LPC768 CPU clock. Timer I can be pre-loaded with one of four values to optimize timing for different oscillator frequencies. At lower frequencies, software response time is increased and will degrade maximum performance of the I²C bus. See special function register I2CFG description for prescale values (CT0, CT1).

The MAXIMUM SCL CHANGE time is important, but its exact span is not critical. The complete 10 bits of timer I are used to count out the maximum time. When I²C operation is enabled, this counter is cleared by transitions on the SCL pin. The timer does not run between I²C frames (i.e., whenever reset or stop occurred more recently than the last start). When this counter is running, it will carry out after 1020 to 1023 machine cycles have elapsed since a change on SCL. A carry out causes a hardware reset of the I²C interface and generates an interrupt if the Timer I interrupt is enabled. In cases where the bus hang-up is due to a lack of software response by this device, the reset releases SCL and allows I²C operation among other devices to continue.

Timer I is enabled to run, and will reset the I²C interface upon overflow, if the TIRUN bit in the I2CFG register is set. The Timer I interrupt may be enabled via the ETI bit in IEN1, and its priority set by the PTIH and PTI bits in the Ip1H and IP1 registers respectively.

I²C Interrupts

If I²C interrupts are enabled (EA and EI2 are both set to 1), an I²C interrupt will occur whenever the ATN flag is set by a start, stop, arbitration loss, or data ready condition (refer to the description of ATN following). In practice, it is not efficient to operate the I²C interface in this fashion because the I²C interrupt service routine would somehow have to distinguish between hundreds of possible conditions. Also, since I²C can operate at a fairly high rate, the software may execute faster if the code simply waits for the I²C interface.

Typically, the I²C interrupt should only be used to indicate a start condition at an idle slave device, or a stop condition at an idle master device (if it is waiting to use the I²C bus). This is accomplished by enabling the I²C interrupt only during the aforementioned conditions.

Reading I2CON

RDAT — The data from SDA is captured into “Receive DATa” whenever a rising edge occurs on SCL. RDAT is also available (with seven low-order zeros) in the I2DAT register. The difference between reading it here and there is that reading I2DAT clears DRDY, allowing the I²C to proceed on to another bit. Typically, the first seven bits of a received byte are read from I2DAT, while the 8th is read here. Then I2DAT can be written to send the Acknowledge bit and clear DRDY.

ATN — “ATteNtion” is 1 when one or more of DRDY, ARL, STR, or STP is 1. Thus, ATN comprises a single bit that can be tested to release the I²C service routine from a “wait loop.”

DRDY — “Data ReaDY” (and thus ATN) is set when a rising edge occurs on SCL, except at idle slave. DRDY is cleared by writing CDR = 1, or by writing or reading the I2DAT register. The following low period on SCL is stretched until the program responds by clearing DRDY.

I2CON Address: D8h Reset Value: 81h

Bit Addressable[1]

	7	6	5	4	3	2	1	0
READ	RDAT	ATN	DRDY	ARL	STR	STP	MASTER	—
WRITE	CXA	IDLE	CDR	CARL	CSTR	CSTP	XSTR	XSTP

BIT	SYMBOL	FUNCTION
I2CON.7	RDAT	Read: the most recently received data bit.
"	CXA	Write: clears the transmit active flag.
I2CON.6	ATN	Read: ATN = 1 if any of the flags DRDY, ARL, STR, or STP = 1.
"	IDLE	Write: in the I^2C slave mode, writing a 1 to this bit causes the I^2C hardware to ignore the bus until it is needed again.
I2CON.5	DRDY	Read: Data Ready flag, set when there is a rising edge on SCL.
"	CDR	Write: writing a 1 to this bit clears the DRDY flag.
I2CON.4	ARL	Read: Arbitration Loss flag, set when arbitration is lost while in the transmit mode.
"	CARL	Write: writing a 1 to this bit clears the CARL flag.
I2CON.3	STR	Read: Start flag, set when a start condition is detected at a master or non-idle slave.
"	CSTR	Write: writing a 1 to this bit clears the STR flag.
I2CON.2	STP	Read: Stop flag, set when a stop condition is detected at a master or non-idle slave.
"	CSTP	Write: writing a 1 to this bit clears the STP flag.
I2CON.1	MASTER	Read: indicates whether this device is currently as bus master.
"	XSTR	Write: writing a 1 to this bit causes a repeated start condition to be generated.
I2CON.0	—	Read: undefined.
"	XSTP	Write: writing a 1 to this bit causes a stop condition to be generated.

SU01155

Figure 9. I^2C Control Register (I2CON)

I2DAT Address: D9h Reset Value: xxh

Not Bit Addressable

	7	6	5	4	3	2	1	0
READ	RDAT	—	—	—	—	—	—	—
WRITE	XDAT	—	—	—	—	—	—	—

BIT	SYMBOL	FUNCTION
I2DAT.7	RDAT	Read: the most recently received data bit, captured from SDA at every rising edge of SCL. Reading I2DAT also clears DRDY and the Transmit Active state.
"	XDAT	Write: sets the data for the next transmitted bit. Writing I2DAT also clears DRDY and sets the Transmit Active state.
I2DAT.6–0	–	Unused.

SU01156

Figure 10. I^2C Data Register (I2DAT)

Checking ATN and DRDY

When a program detects ATN = 1, it should next check DRDY. If DRDY = 1, then if it receives the last bit, it should capture the data from RDAT (in I2DAT or I2CON). Next, if the next bit is to be sent, it should be written to I2DAT. One way or another, it should clear DRDY and then return to monitoring ATN. Note that if any of ARL, STR, or STP is set, clearing DRDY will not release SCL to high, so that the I^2C will not go on to the next bit. If a program detects ATN = 1, and DRDY = 0, it should go on to examine ARL, STR, and STP.

ARL "Arbitration Loss" is 1 when transmit Active was set, but this device lost arbitration to another transmitter. Transmit Active is cleared when ARL is 1. There are four separate cases in which ARL is set.

1. If the program sent a 1 or repeated start, but another device sent a 0, or a stop, so that SDA is 0 at the rising edge of SCL. (If the other device sent a stop, the setting of ARL will be followed shortly by STP being set.)

2. If the program sent a 1, but another device sent a repeated start, and it drove SDA low before SCL could be driven low. (This type of ARL is always accompanied by STR = 1.)

3. In master mode, if the program sent a repeated start, but another device sent a 1, and it drove SCL low before this device could drive SDA low.

4. In master mode, if the program sent stop, but it could not be sent because another device sent a 0.

STR "STaRt" is set to a 1 when an I^2C start condition is detected at a non-idle slave or at a master. (STR is not set when an idle slave becomes active due to a start bit; the slave has nothing useful to do until the rising edge of SCL sets DRDY.)

STP "SToP" is set to 1 when an I^2C stop condition is detected at a non-idle slave or at a master. (STP is not set for a stop condition at an idle slave.)

MASTER "MASTER" is 1 if this device is currently a master on the I^2C. MASTER is set when MASTRQ is 1 and the bus is not busy (i.e., if a start bit hasn't been received since reset or a "Timer I" time-out, or if a stop has been received since the last start). MASTER is cleared when ARL is set, or after the software writes MASTRQ = 0 and then XSTP = 1.

Writing I2CON

Typically, for each bit in an I^2C message, a service routine waits for ATN = 1. Based on DRDY, ARL, STR, and STP, and on the current bit position in the message, it may then write I2CON with one or more of the following bits, or it may read or write the I2DAT register.

CXA Writing a 1 to "Clear Xmit Active" clears the Transmit Active state. (Reading the I2DAT register also does this.)

Regarding Transmit Active

Transmit Active is set by writing the I2DAT register, or by writing I2CON with XSTR = 1 or XSTP = 1. The I^2C interface will only drive the SDA line low when Transmit Active is set, and the ARL bit will only be set to 1 when Transmit Active is set. Transmit Active is cleared by reading the I2DAT register, or by writing I2CON with CXA = 1. Transmit Active is automatically cleared when ARL is 1.

IDLE Writing 1 to "IDLE" causes a slave's I^2C hardware to ignore the I^2C until the next start condition (but if MASTRQ is 1, then a stop condition will cause this device to become a master).

CDR Writing a 1 to "Clear Data Ready" clears DRDY. (Reading or writing the I2DAT register also does this.)

CARL Writing a 1 to "Clear Arbitration Loss" clears the ARL bit.

CSTR Writing a 1 to "Clear STaRt" clears the STR bit.

CSTP Writing a 1 to "Clear SToP" clears the STP bit. Note that if one or more of DRDY, ARL, STR, or STP is 1, the low time of SCL is stretched until the service routine responds by clearing them.

XSTR Writing 1s to "Xmit repeated STaRt" and CDR tells the I^2C hardware to send a repeated start condition. This should only be at a master. Note that XSTR need not and should not be used to send an "initial" (non-repeated) start; it is sent automatically by the I^2C hardware. Writing XSTR = 1 includes the effect of writing I2DAT with XDAT = 1; it sets Transmit Active and releases SDA to high during the SCL low time. After SCL goes high, the I^2C hardware waits for the suitable minimum time and then drives SDA low to make the start condition.

XSTP Writing 1s to "Xmit SToP" and CDR tells the I^2C hardware to send a stop condition. This should only be done at a master. If there are no more messages to initiate, the service routine should clear the MASTRQ bit in I2CFG to 0 before writing XSTP with 1. Writing XSTP = 1 includes the effect of writing I2DAT with XDAT = 0; it sets Transmit Active and drives SDA low during the SCL low time. After SCL goes high, the I^2C hardware waits for the suitable minimum time and then releases SDA to high to make the stop condition.

Timer/Counters

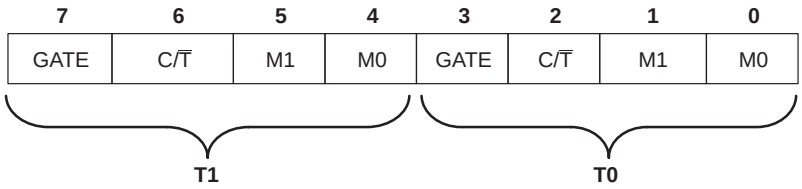
The P87LPC768 has two general purpose counter/timers which are upward compatible with the standard 80C51 Timer 0 and Timer 1. Both can be configured to operate either as timers or event counters (see Figure 25). An option to automatically toggle the T0 and/or T1 pins upon timer overflow has been added.

In the “Timer” function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 6 CPU clock periods, the count rate is 1/6 of the CPU clock frequency. Refer to the section Enhanced CPU for a description of the CPU clock.

In the “Counter” function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled once during every machine cycle. When the samples of the pin state show a high in one cycle and a 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 (12 CPU clocks) to recognize a 1-to-0 transition, the maximum count rate is 1/6 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 bits C/T in the Special Function Register TMOD. In addition to the “Timer” or “Counter” selection, Timer 0 and Timer 1 have four operating modes, which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timers/Counters. Mode 3 is different. The four operating modes are described in the following text.

TMOD Address: 89h Reset Value: 00h
Not Bit Addressable



BIT	SYMBOL	FUNCTION
TMOD.7	GATE	Gating control for Timer 1. When set, Timer/Counter is enabled only while the $\overline{INT1}$ pin is high and the TR1 control pin is set. When cleared, Timer 1 is enabled when the TR1 control bit is set.
TMOD.6	C/$\overline{T}$	Timer or Counter Selector for Timer 1. Cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from T1 input pin).
TMOD.5, 4	M1, M0	Mode Select for Timer 1 (see table below).
TMOD.3	GATE	Gating control for Timer 0. When set, Timer/Counter is enabled only while the $\overline{INT0}$ pin is high and the TR0 control pin is set. When cleared, Timer 0 is enabled when the TR0 control bit is set.
TMOD.2	C/$\overline{T}$	Timer or Counter Selector for Timer 0. Cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from T0 input pin).
TMOD.1, 0	M1, M0	Mode Select for Timer 0 (see table below).
	M1, M0	Timer Mode
	0 0	8048 Timer “TLn” serves as 5-bit prescaler.
	0 1	16-bit Timer/Counter “THn” and “TLn” are cascaded; there is no prescaler.
	1 0	8-bit auto-reload Timer/Counter. THn holds a value which is loaded into TLn when it overflows.
	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.

SU01171

Figure 25. Timer/Counter Mode Control Register (TMOD)

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 27 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 TFn. The count input is enabled to the Timer when TRn = 1 and either GATE = 0 or $\overline{INTn}$ = 1. (Setting GATE = 1 allows the Timer to be controlled by external input $\overline{INTn}$, to facilitate pulse width measurements). TRn is a control bit in the Special Function Register TCON (Figure 26). The GATE bit is in the TMOD register.

The 13-bit register consists of all 8 bits of THn and the lower 5 bits of TLn. The upper 3 bits of TLn are indeterminate and should be ignored. Setting the run flag (TRn) does not clear the registers.

Mode 0 operation is the same for Timer 0 and Timer 1. See Figure 27. There are two different GATE bits, one for Timer 1 (TMOD.7) and one for Timer 0 (TMOD.3).

TCON Address: 88h Reset Value: 00h

Bit Addressable

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

BIT	SYMBOL	FUNCTION
TCON.7	TF1	Timer 1 overflow flag. Set by hardware on Timer/Counter 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/Counter 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.
TCON.4	TR0	Timer 0 Run control bit. Set/cleared by software to turn Timer/Counter 0 on/off.
TCON.3	IE1	Interrupt 1 Edge flag. Set by hardware when external interrupt 1 edge is detected. Cleared by hardware when the interrupt is processed, or by software.
TCON.2	IT1	Interrupt 1 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
TCON.1	IE0	Interrupt 0 Edge flag. Set by hardware when external interrupt 0 edge is detected. Cleared by hardware when the interrupt is processed, or by software.
TCON.0	IT0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

SU01172

Figure 26. Timer/Counter Control Register (TCON)

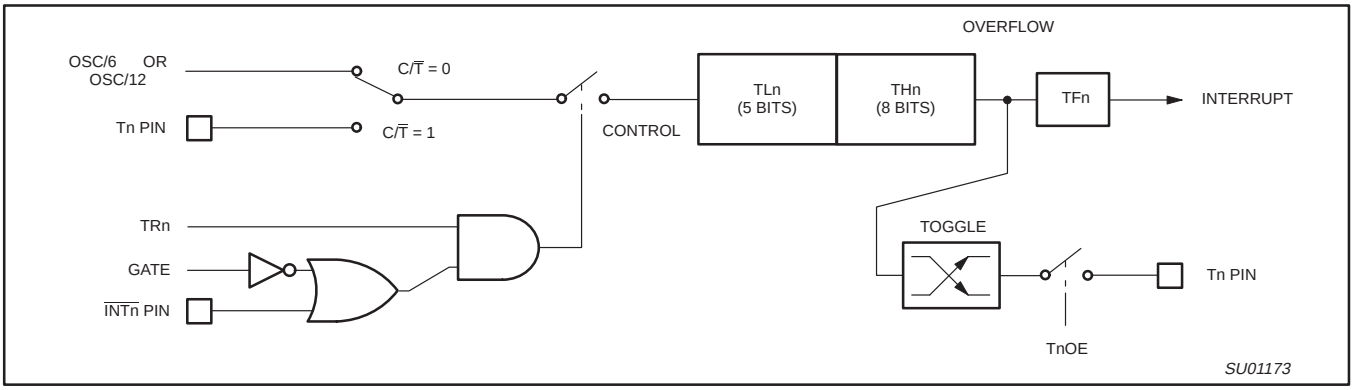


Figure 27. Timer/Counter 0 or 1 in Mode 0 (13-Bit Counter)

Mode 1
Mode 1 is the same as Mode 0, except that all 16 bits of the timer register (THn and TLn) are used. See Figure 28

Mode 2
Mode 2 configures the Timer register as an 8-bit Counter (TL1) with automatic reload, as shown in Figure 29. Overflow from TLn not only sets TFn, but also reloads TLn with the contents of THn, which must be preset by software. The reload leaves THn unchanged. Mode 2 operation is the same for Timer 0 and Timer 1.

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 30. TL0 uses the Timer 0 control bits: C/T, GATE, TR0, $\overline{INT0}$, 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. With Timer 0 in Mode 3, an P87LPC768 can look like it has three Timer/Counters. 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, or in any application not requiring an interrupt.

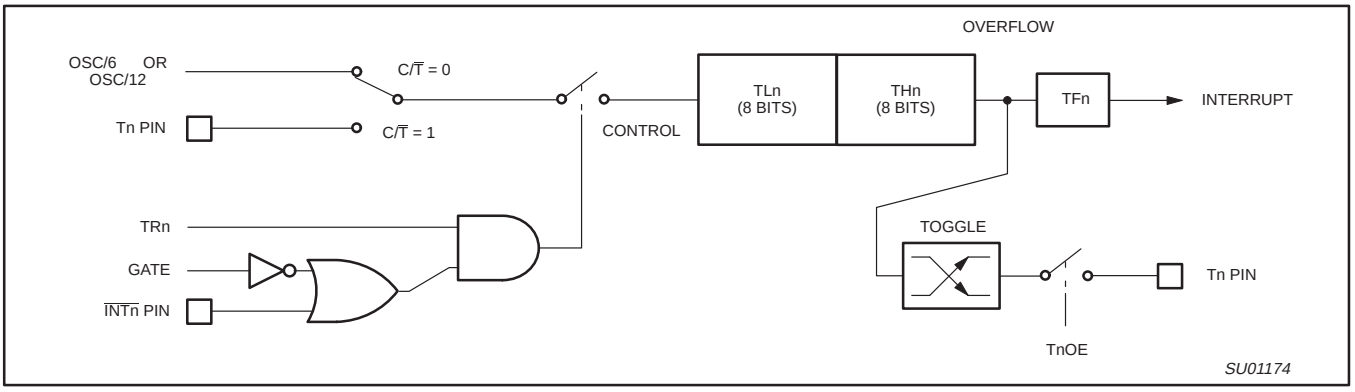


Figure 28. Timer/Counter 0 or 1 in Mode 1 (16-Bit Counter)

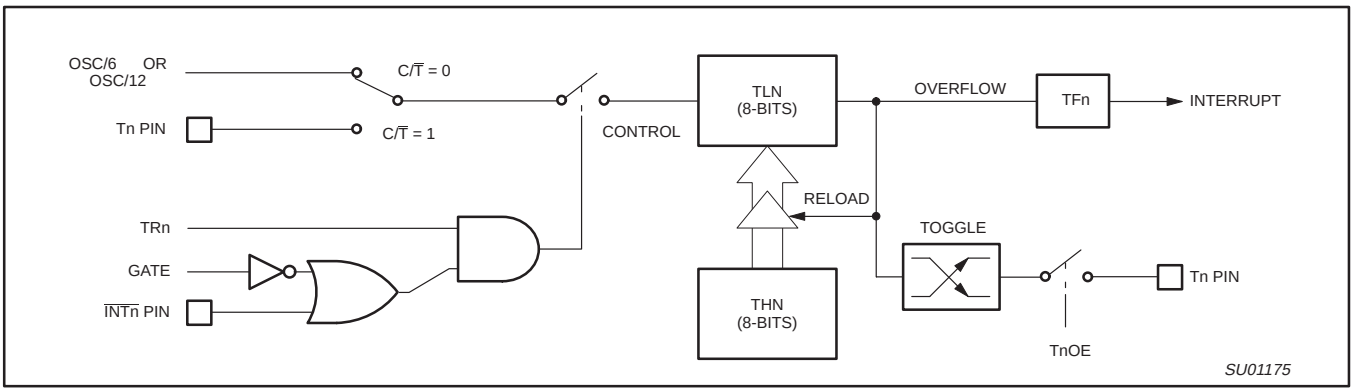


Figure 29. Timer/Counter 0 or 1 in Mode 2 (8-Bit Auto-Reload)

Serial Port Control Register (SCON)

The serial port control and status register is the Special Function Register SCON, shown in Figure 31. This register contains not only the mode selection bits, but also the 9th data bit for transmit and receive (TB8 and RB8), and the serial port interrupt bits (TI and RI).

The Framing Error bit (FE) allows detection of missing stop bits in the received data stream. The FE bit shares the bit position SCON.7 with the SM0 bit. Which bit appears in SCON at any particular time is determined by the SMOD0 bit in the PCON register. If SMOD0 = 0, SCON.7 is the SM0 bit. If SMOD0 = 1, SCON.7 is the FE bit. Once set, the FE bit remains set until it is cleared by software. This allows detection of framing errors for a group of characters without the need for monitoring it for every character individually.

SCON Address: 98h Reset Value: 00h

Bit Addressable

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

BIT	SYMBOL	FUNCTION
SCON.7	FE	Framing Error. This bit is set by the UART receiver when an invalid stop bit is detected. Must be cleared by software. The SMOD0 bit in the PCON register must be 1 for this bit to be accessible. See SM0 bit below.
SCON.7	SM0	With SM1, defines the serial port mode. The SMOD0 bit in the PCON register must be 0 for this bit to be accessible. See FE bit above.
SCON. 6	SM1	With SM0, defines the serial port mode (see table below).

SM0, SM1	UART Mode	Baud Rate
0 0	0: shift register	CPU clock/6
0 1	1: 8-bit UART	Variable (see text)
1 0	2: 9-bit UART	CPU clock/32 or CPU clock/16
1 1	3: 9-bit UART	Variable (see text)

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 1, if SM2=1 then RI will not be activated if a valid stop bit was not received. In Mode 0, SM2 should 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	In Modes 2 and 3, is the 9th data bit that was received. In Mode 1, it SM2=0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.
SCON.1	TI	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. 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 halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

SU01157

Figure 31. Serial Port Control Register (SCON)

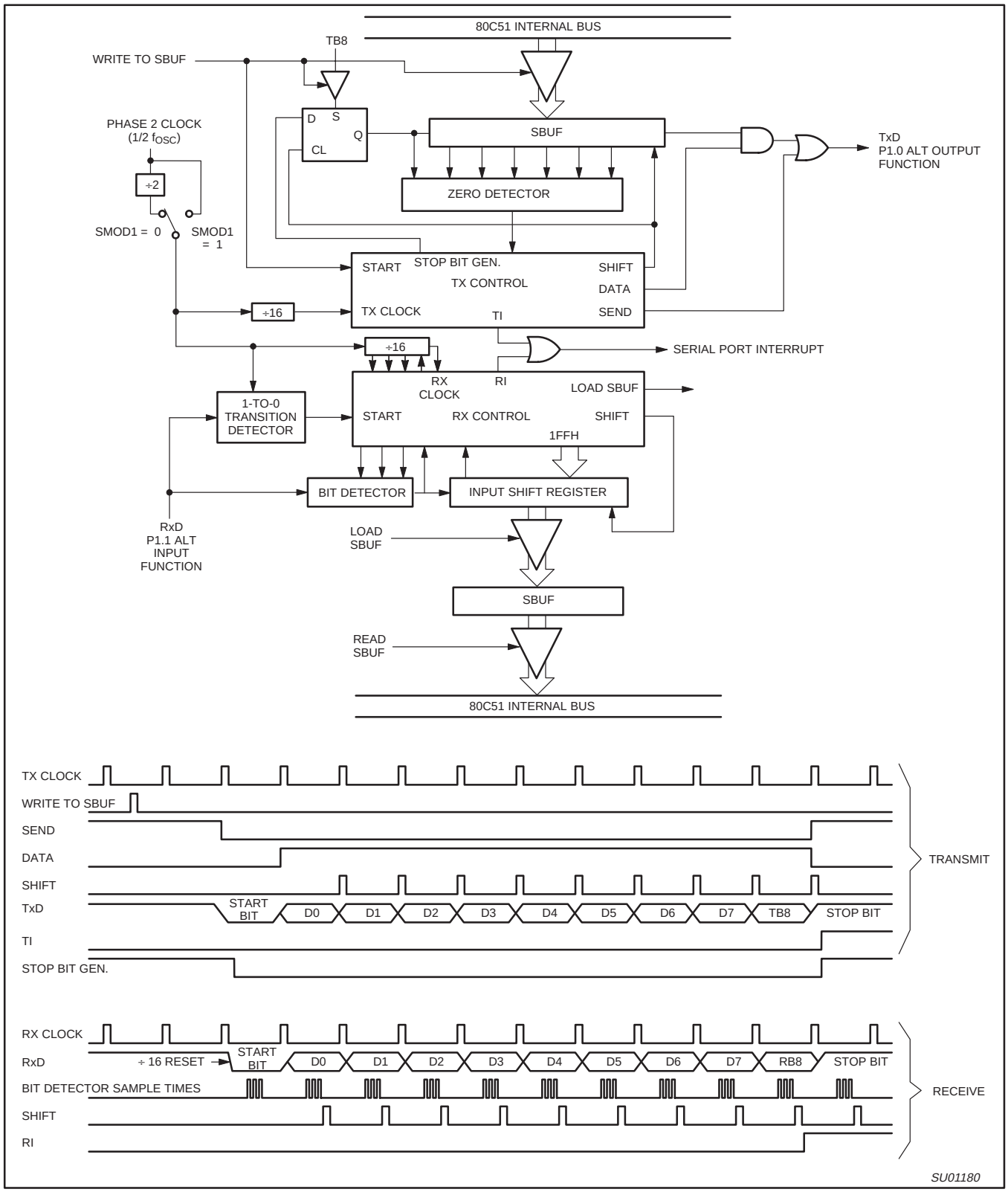


Figure 34. Serial Port Mode 2

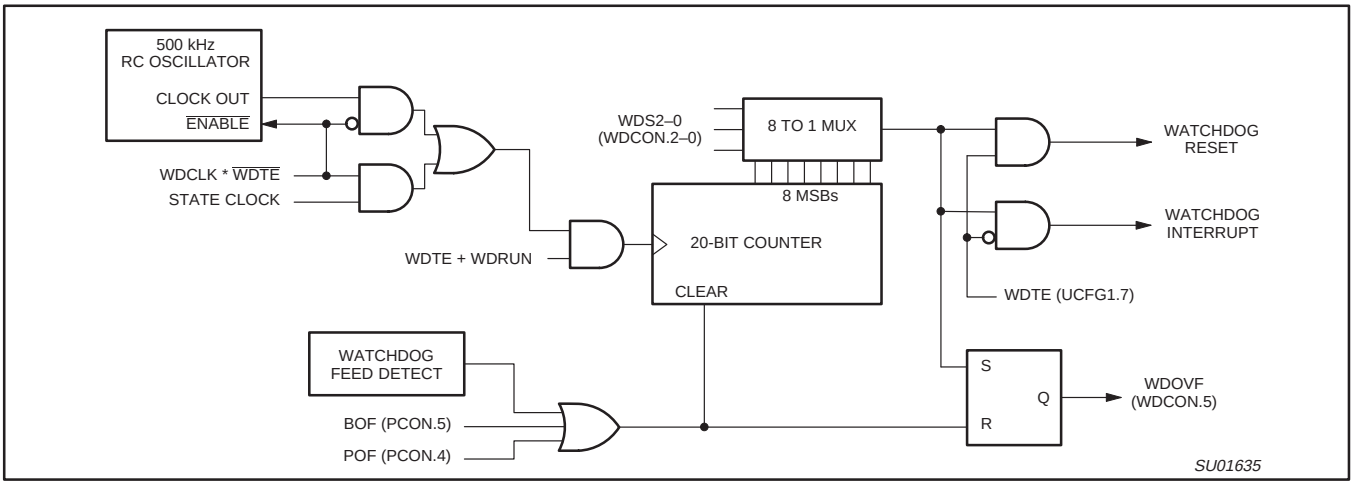


Figure 36. Block Diagram of the Watchdog Timer

WDCON Address: A7h

Not Bit Addressable

Reset Value:
- 30h for a watchdog reset.
- 10h for other rest sources if the watchdog is enabled via the WDTE configuration bit.
- 00h for other reset sources if the watchdog is disabled via the WDTE configuration bit.

7	6	5	4	3	2	1	0
—	—	WDOVF	WDRUN	WDCLK	WDS2	WDS1	WDS0

BIT	SYMBOL	FUNCTION
WDCON.7, 6	—	Reserved for future use. Should not be set to 1 by user programs.
WDCON.5	WDOVF	Watchdog timer overflow flag. Set when a watchdog reset or timer overflow occurs. Cleared when the watchdog is fed.
WDCON.4	WDRUN	Watchdog run control. The watchdog timer is started when WDRUN = 1 and stopped when WDRUN = 0. This bit is forced to 1 (watchdog running) if the WDTE configuration bit = 1.
WDCON.3	WDCLK	Watchdog clock select. The watchdog timer is clocked by CPU clock/6 when WDCLK = 1 and by the watchdog RC oscillator when WDCLK = 0. This bit is forced to 0 (using the watchdog RC oscillator) if the WDTE configuration bit = 1.
WDCON.2–0	WDS2–0	Watchdog rate select.

WDS2–0	Timeout Clocks	Minimum Time	Nominal Time	Maximum Time
0 0 0	8,192	10 ms	16 ms	23 ms
0 0 1	16,384	20 ms	32 ms	45 ms
0 1 0	32,768	41 ms	65 ms	90 ms
0 1 1	65,536	82 ms	131 ms	180 ms
1 0 0	131,072	165 ms	262 ms	360 ms
1 0 1	262,144	330 ms	524 ms	719 ms
1 1 0	524,288	660 ms	1.05 sec	1.44 sec
1 1 1	1,048,576	1.3 sec	2.1 sec	2.9 sec

SU01183

Figure 37. Watchdog Timer Control Register (WDCON)

DC ELECTRICAL CHARACTERISTICS

V_{DD} = 2.7 V to 6.0 V unless otherwise specified; T_{amb} = 0°C to +70°C or –40°C to +85°C, unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS			UNIT
			MIN	TYP[1,2]	MAX	
I_{DD}	Power supply current, operating	5.0 V, 20 MHz[11]		15	25	mA
		3.0 V, 10 MHz[11]		4	7	mA
I_{ID}	Power supply current, Idle mode	5.0 V, 20 MHz[11]		6	10	mA
		3.0 V, 10 MHz[11]		2	4	mA
I_{PD}	Power supply current, Power Down mode	5.0 V[11]		1	10	µA
		3.0 V[11]		1	5	µA
V_{RAM}	RAM keep-alive voltage		1.5			V
V_{IL}	Input low voltage (TTL input)	4.0 V < V_{DD} < 6.0 V	–0.5		0.2 V_{DD}–0.1	V
		2.7 V < V_{DD} < 4.0 V	–0.5		0.7	V
V_{IL1}	Negative going threshold (Schmitt input)		–0.5 V_{DD}	0.4 V_{DD}	0.3 V_{DD}	V
V_{IH}	Input high voltage (TTL input)		0.2 V_{DD}+0.9		V_{DD}+0.5	V
V_{IH1}	Positive going threshold (Schmitt input)		0.7 V_{DD}	0.6 V_{DD}	V_{DD}+0.5	V
HYS	Hysteresis voltage			0.2 V_{DD}		V
V_{OL}	Output low voltage all ports[5, 9]	I_{OL} = 3.2 mA, V_{DD} = 2.7 V			0.4	V
V_{OL1}	Output low voltage all ports[5, 9]	I_{OL} = 20 mA, V_{DD} = 2.7 V			1.0	V
V_{OH}	Output high voltage, all ports[3]	I_{OH} = –20 µA, V_{DD} = 2.7 V	V_{DD}–0.7			V
		I_{OH} = –30 µA, V_{DD} = 4.5 V	V_{DD}–0.7			V
V_{OH1}	Output high voltage, all ports[4]	I_{OH} = –1.0 mA, V_{DD} = 2.7 V	V_{DD}–0.7			V
C_{IO}	Input/Output pin capacitance[10]				15	pF
I_{IL}	Logical 0 input current, all ports[8]	V_{IN} = 0.4 V			–50	µA
I_{LI}	Input leakage current, all ports[7]	V_{IN} = V_{IL} or V_{IH}			±2	µA
I_{TL}	Logical 1 to 0 transition current, all ports[3, 6]	V_{IN} = 1.5 V at V_{DD} = 3.0 V	–30		–250	µA
		V_{IN} = 2.0 V at V_{DD} = 5.5 V	–150		–650	µA
R_{RST}	Internal reset pull-up resistor		40		225	kΩ
$V_{BO2.5}$	Brownout trip voltage with BOV = 1[12]	T_{amb} = 0°C to +70°C	2.45	2.5	2.65	V
$V_{BO3.8}$	Brownout trip voltage with BOV = 0		3.45	3.8	3.90	V
V_{REF}	Bandgap reference voltage		1.11	1.26	1.41	V

NOTES:

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5 V.
2. See other Figures for details.
3. Ports in quasi-bidirectional mode with weak pull-up (applies to all port pins with pull-ups). Does not apply to open drain pins.
4. Ports in PUSH-PULL mode. Does not apply to open drain pins.
5. In all output modes except high impedance mode.
6. Port pins source a transition current when used in quasi-bidirectional mode and externally driven from 1 to 0. This current is highest when V_{IN} is approximately 2 V.
7. Measured with port in high impedance mode. Parameter is guaranteed but not tested at cold temperature.
8. Measured with port in quasi-bidirectional mode.
9. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 20 mA
   Maximum total I_{OL} for all outputs: 80 mA
   Maximum total I_{OH} for all outputs: 5 mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
10. Pin capacitance is characterized but not tested.
11. The I_{DD}, I_{ID}, and I_{PD} specifications are measured using an external clock with the following functions disabled: comparators, brownout detect, and watchdog timer. For V_{DD} = 3 V, LPEP = 1. Refer to the appropriate figures on the following pages for additional current drawn by each of these functions and detailed graphs for other frequency and voltage combinations.
12. Devices initially operating at V_{DD} = 2.7V or above and at f_{OSC} = 10 MHz or less are guaranteed to continue to execute instructions correctly at the brownout trip point. Initial power-on operation below V_{DD} = 2.7 V is not guaranteed.

COMPARATOR ELECTRICAL CHARACTERISTICS

V_{DD} = 3.0 V to 6.0 V unless otherwise specified; T_{amb} = 0°C to +70°C or –40°C to +85°C, unless otherwise specified

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS			UNIT
			MIN	TYP	MAX	
V_{IO}	Offset voltage comparator inputs[1]				±10	mV
V_{CR}	Common mode range comparator inputs		0		V_{DD}–0.3	V
CMRR	Common mode rejection ratio[1]				–50	dB
	Response time			250	500	ns
	Comparator enable to output valid				10	µs
I_{IL}	Input leakage current, comparator	$0 < V_{IN} < V_{DD}$			±10	µA

NOTE:
1. This parameter is guaranteed by characterization, but not tested in production.

A/D CONVERTER DC ELECTRICAL CHARACTERISTICS

Vdd = 3.0V to 6.0V unless otherwise specified;
Tamb = 0 to +70°C for commercial, -40°C to +85°C for industrial, unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS		UNIT
			MIN	MAX	
AV_{IN}	Analog input voltage		V_{SS} - 0.2	V_{DD} + 0.2	V
R_{REF}	Resistance between V_{DD} and V_{SS}	A/D enabled	tbd	tbd	kΩ
C_{IA}	Analog input capacitance			15	pF
DL_e	Differential non-linearity[1,2,3]			±1	LSB
IL_e	Integral non-linearity[1,4]			±1	LSB
OS_e	Offset error[1,5]			±2	LSB
G_e	Gain error[1,6]			±1	%
A_e	Absolute voltage error[1,7]			±1	LSB
M_{CTC}	Channel-to-channel matching			±1	LSB
C_t	Crosstalk between inputs of port[8]	0 - 100kHz		-60	dB
-	Input slew rate			100	V/ms
-	Input source impedance			10	kΩ

NOTES:
1. Conditions: V_{SS} = 0V; V_{DD} = 5.12V.
2. The A/D is monotonic, there are no missing codes
3. The differential non-linearity (DL_e) is the difference between the actual step width and the ideal step width. See Figure 41.
4. The integral non-linearity (IL_e) is the peak difference between the center of the steps of the actual and the ideal transfer curve after appropriate adjustment of gain and offset errors. See Figure 41.
5. The offset error (OS_e) is the absolute difference between the straight line which fits the actual transfer curve (after removing gain error), and the straight line which fits the ideal transfer curve. See Figure 41.
6. The gain error (G_e) is the relative difference in percent between the straight line fitting the actual transfer curve (after removing offset error), and the straight line which fits the ideal transfer curve. Gain error is constant at every point on the transfer curve. See Figure 41.
7. The absolute voltage error (A_e) is the maximum difference between the center of the steps of the actual transfer curve of the non-calibrated ADC and the ideal transfer curve.
8. This should be considered when both analog and digital signals are input simultaneously to A/D pins.
9. Changing the input voltage faster than this may cause erroneous readings.
10. A source impedance higher than this driving an A/D input may result in loss of precision and erroneous readings.

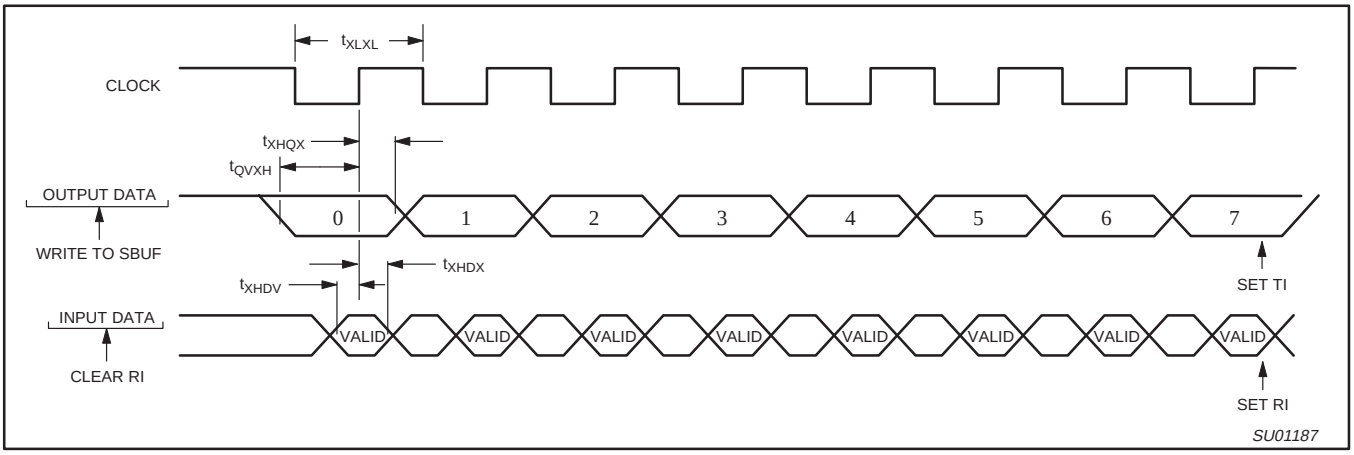


Figure 42. Shift Register Mode Timing

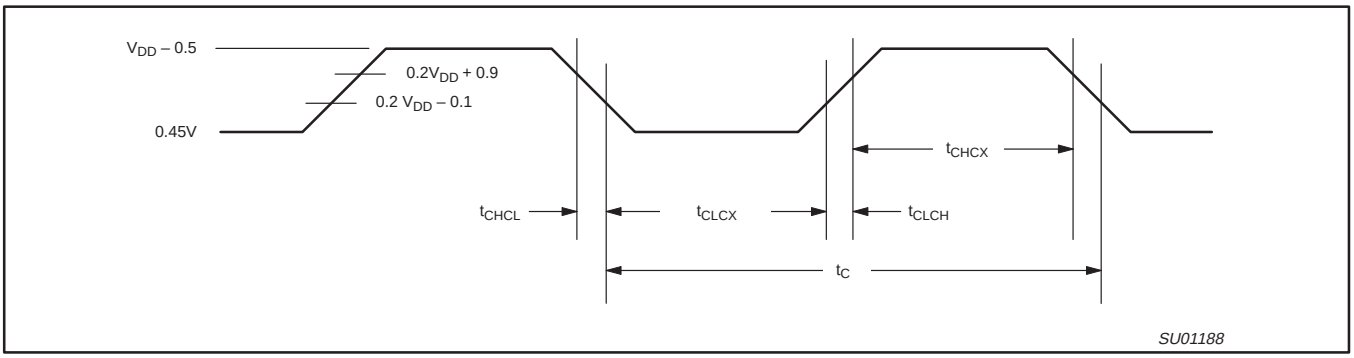


Figure 43. External Clock Timing

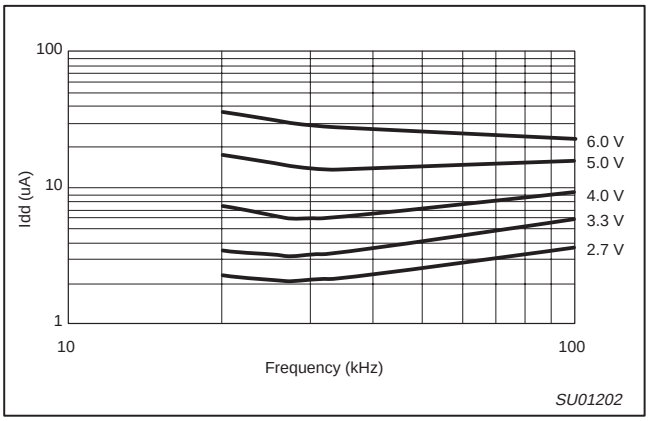


Figure 44. Typical low frequency oscillator Idd at 25°C (See Note 1)

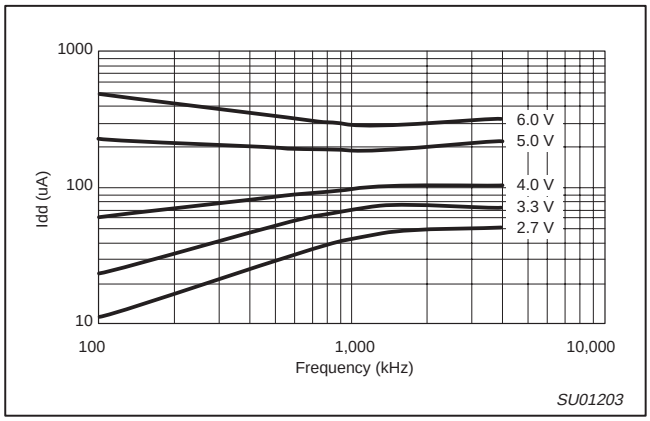


Figure 45. Typical medium frequency oscillator Idd at 25°C (See Note 1)

REVISION HISTORY

Date	CPCN	Description
2002 Mar 12	9397 750 09558	– Added revision history – Updated Reset section
2001 Aug 06	9397 750 08661	Previous release