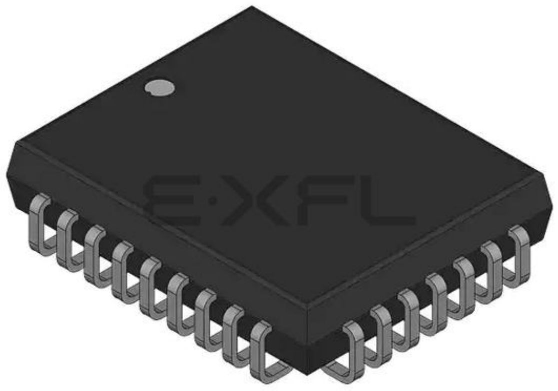
Atmel - AT89LP828-20JU Datasheet



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

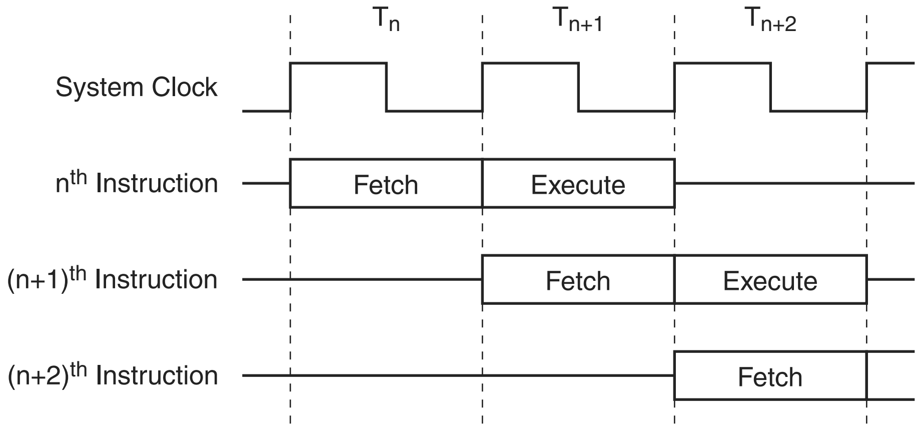
Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	8051
Core Size	8-Bit
Speed	20MHz
Connectivity	SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	30
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	1K x 8
RAM Size	768 x 8
Voltage - Supply (Vcc/Vdd)	2.4V ~ 5.5V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-LCC (J-Lead)
Supplier Device Package	32-PLCC (13.97x11.43)
Purchase URL	https://www.e-xfl.com/product-detail/atmel/at89lp828-20ju

5. Enhanced CPU

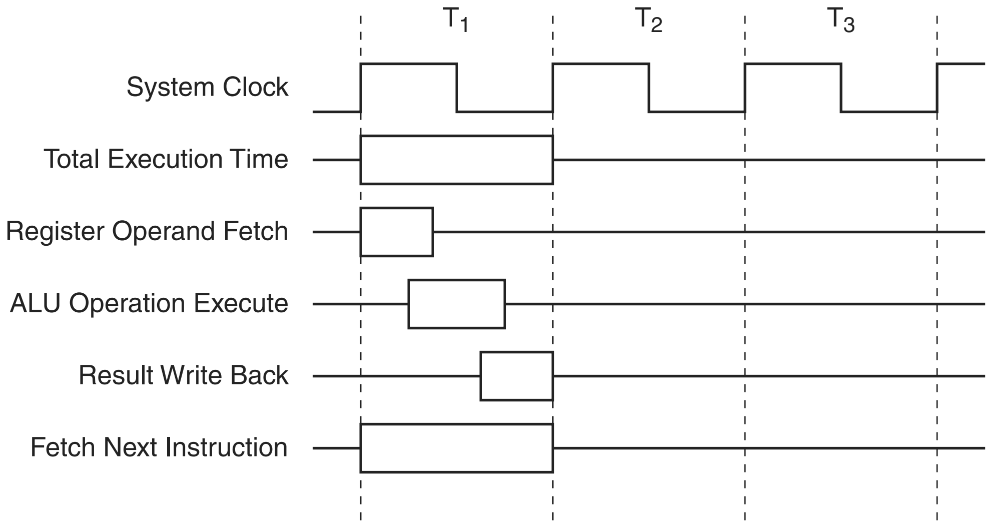
The AT89LP428/828 uses an enhanced 8051 CPU that runs at 6 to 12 times the speed of standard 8051 devices (or 3 to 6 times the speed of X2 8051 devices). The increase in performance is due to two factors. First, the CPU fetches one instruction byte from the code memory every clock cycle. Second, the CPU uses a simple two-stage pipeline to fetch and execute instructions in parallel. This basic pipelining concept allows the CPU to obtain up to 1 MIPS per MHz. A simple example is shown in Figure 5-1.

Figure 5-1. Parallel Instruction Fetches and Executions



The MCS-51 instruction set allows for instructions of variable length from 1 to 3 bytes. In a single-clock-per-byte-fetch system this means each instruction takes at least as many clocks as it has bytes to execute. The majority of instructions in the AT89LP428/828 follow this rule: the instruction execution time in clock cycles equals the number of bytes per instruction with a few exceptions. Branches and Calls require an additional cycle to compute the target address and some other complex instructions require multiple cycles. See “Instruction Set Summary” on page 107 for more detailed information on individual instructions. Figures 5-2 and 5-3 show examples of 1- and 2-byte instructions.

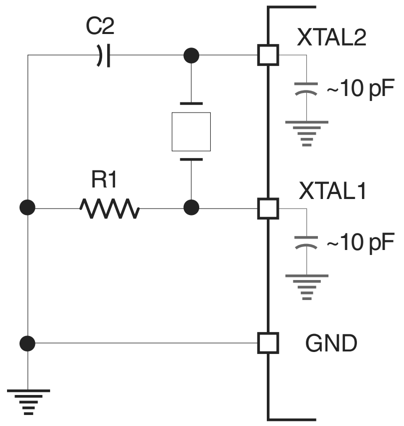
Figure 5-2. Single-cycle ALU Operation (Example: INC R0)



6.1 Crystal Oscillator

When enabled, the internal inverting oscillator amplifier is connected between XTAL1 and XTAL2 for connection to an external quartz crystal or ceramic resonator. The oscillator may operate in either high-speed or low-speed mode. Low-speed mode is intended for 32.768 kHz watch crystals and consumes less power than high-speed mode. The configuration as shown in Figure 6-1 applies for both high and low speed oscillators. Note that the internal structure of the device adds about 10 pF of capacitance to both XTAL1 and XTAL2, so that in some cases an external capacitor may **NOT** be required. It is recommended that a resistor R1 be connected to XTAL1, instead of load capacitor C1, for improved startup performance. The total capacitance on XTAL1 or XTAL2, including the external load capacitor plus internal device load, board trace and crystal loadings, should not exceed 20 pF.When using the crystal oscillator, P4.0 and P4.1 will have their inputs and outputs disabled. Also, XTAL2 in crystal oscillator mode should not be used to directly drive a board-level clock without a buffer.

Figure 6-1. Crystal Oscillator Connections

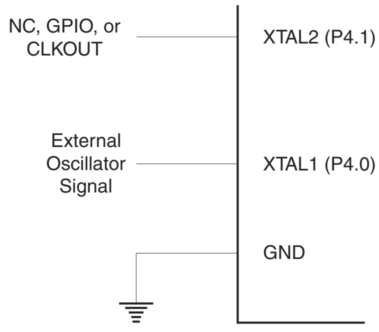


Note:	1. C2	=	0–10 pF for Crystals
		=	0–10 pF for Ceramic Resonators
	R1	=	4–5 MΩ

6.2 External Clock Source

The external clock option disables the oscillator amplifier and allows XTAL1 to be driven directly by an external clock source as shown in Figure 6-2. XTAL2 may be left unconnected, used as general-purpose I/O P4.1, or configured to output a divided version of the system clock.

Figure 6-2. External Clock Drive Configuration



10.1.2 Input-only Mode

The input only port configuration is shown in Figure 10-2. The output drivers are tristated. The input includes a Schmitt-triggered input for improved input noise rejection. The input circuitry of P3.2, P3.3, P3.6, P4.0 and P4.1 is not disabled during Power-down (see Figure 10-3) and therefore these pins should not be left floating during Power-down when configured in this mode.

Figure 10-2. Input Only

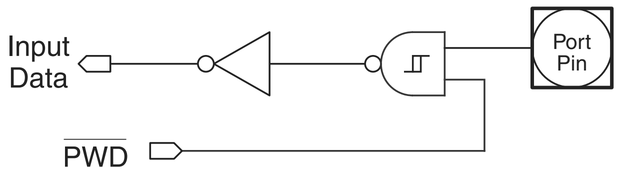
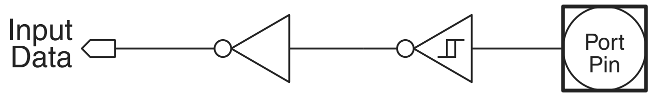


Figure 10-3. Input Circuit for P3.2, P3.3 and P3.6



10.1.3 Open-drain Output

The open-drain output configuration turns off all pull-ups 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 pull-up, typically a resistor tied to V_{CC}. The pull-down for this mode is the same as for the quasi-bidirectional mode. The open-drain port configuration is shown in Figure 10-4. The input circuitry of P3.2, P3.3 and P3.6 is not disabled during Power-down (see Figure 10-3) and therefore these pins should not be left floating during Power-down when configured in this mode.

Figure 10-4. Open-drain Output

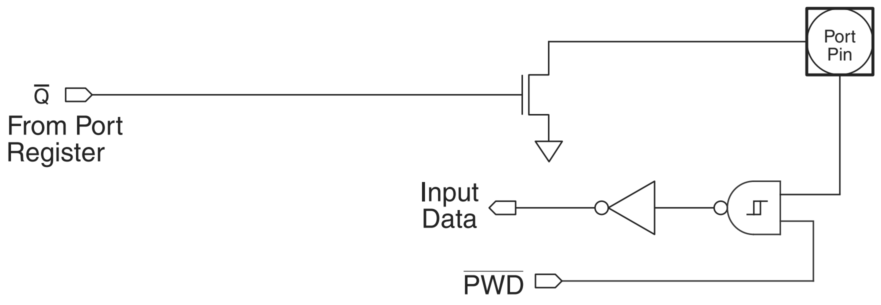


Table 10-6. Port Pin Alternate Functions

Port Pin	Configuration Bits		Alternate Function	Notes
	PxM0.y	PxM1.y		
P1.0	P1M0.0	P1M1.0	T2	
			GPI0	
P1.1	P1M0.1	P1M1.1	T2EX	
			GPI1	
P1.2	P1M0.2	P1M1.2	GPI2	
P1.3	P1M0.3	P1M1.3	GPI3	
P1.4	P1M0.4	P1M1.4	SS	
			GPI4	
P1.5	P1M0.5	P1M1.5	MOSI	
			GPI5	
P1.6	P1M0.6	P1M1.6	MISO	
			GPI6	
P1.7	P1M0.7	P1M1.7	SCK	
			GPI7	
P2.0	P2M0.0	P2M1.0	CCA	
P2.1	P2M0.1	P2M1.1	CCB	
P2.2	P2M0.2	P2M1.2	CCC	
P2.3	P2M0.3	P2M1.3	CCD	
P2.4	P2M0.4	P2M1.4	AIN0	Input-only
P2.5	P2M0.5	P2M1.5	AIN1	Input-only
P2.6	P2M0.6	P2M1.6	AIN2	Input-only
P2.7	P2M0.7	P2M1.7	AIN3	Input-only
P3.0	P3M0.0	P3M1.0	RXD	
P3.1	P3M0.1	P3M1.1	TXD	
P3.2	P3M0.2	P3M1.2	INT0	
P3.3	P3M0.3	P3M1.3	INT1	
P3.4	P3M0.4	P3M1.4	T0	
P3.5	P3M0.5	P3M1.5	T1	
P3.6	P3M0.5	P3M1.5	RST	$\overline{\text{RST}}$ must be disabled to use P3.6
P4.6	Not configurable		CMPA	Pin is tied to comparator output
P4.7	Not configurable		CMPB	Pin is tied to comparator output

11. Enhanced Timer 0 and Timer 1 with PWM

The AT89LP428/828 has two 16-bit Timer/Counters, Timer 0 and Timer 1, with the following features:

- Two 16-bit timer/counters with 16-bit reload registers
- Two independent 8-bit precision PWM outputs with 8-bit prescalers
- UART or SPI baud rate generation using Timer 1
- Output pin toggle on timer overflow
- Split timer mode allows for three separate timers (two 8-bit, one 16-bit)
- Gated modes allow timers to run/halt based on an external input

Timer 0 and Timer 1 have similar modes of operation. As timers, they increase every clock cycle by default. Thus, the registers count clock cycles. Since a clock cycle consists of one oscillator period, the count rate is equal to the oscillator frequency. The timer rate can be prescaled by a value between 1 and 16 using the Timer Prescaler (see Table 6-2 on page 23). Both Timers share the same prescaler.

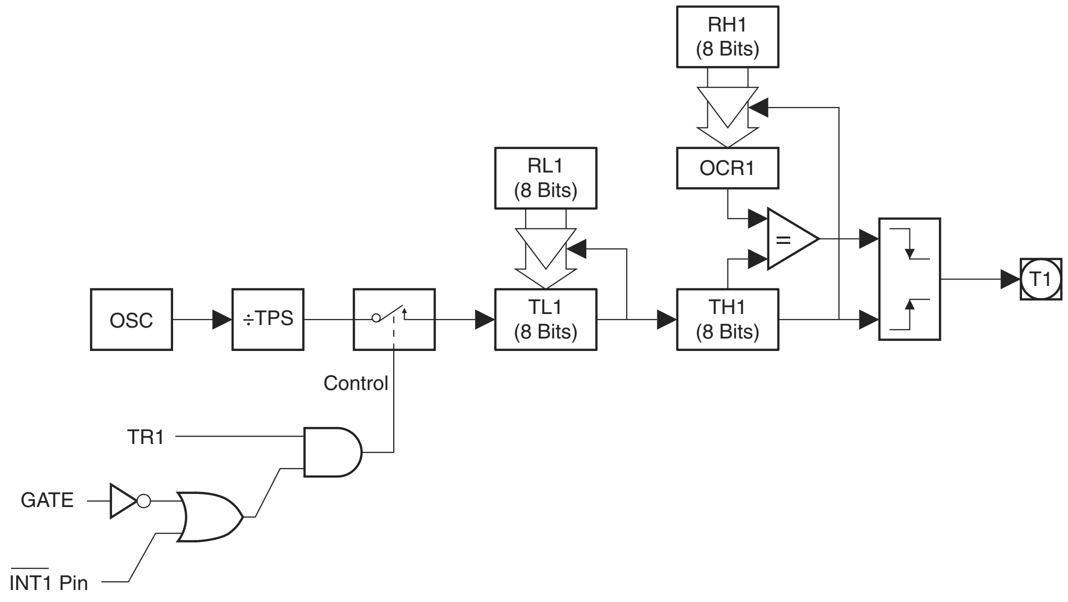
As counters, the timer registers are incremented in response to a 1-to-0 transition at the corresponding input pins, T0 or T1. The external input is sampled every clock cycle. When the samples show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during the cycle following the one in which the transition was detected. Since 2 clock cycles are required to recognize a 1-to-0 transition, the maximum count rate is 1/2 of the oscillator frequency. There are no restrictions on the duty cycle of the input signal, but it should be held for at least one full clock cycle to ensure that a given level is sampled at least once before it changes.

Furthermore, the Timer or Counter functions for Timer 0 and Timer 1 have four operating modes: variable width timer, 16-bit auto-reload timer, 8-bit auto-reload timer, and split timer. The control bits C/T in the Special Function Register TMOD select the Timer or Counter function. The bit pairs (M1, M0) in TMOD select the operating modes.

Table 11-1. Timer 0/1 Register Summary

Name	Address	Purpose	Bit-Addressable
TCON	88H	Control	Y
TMOD	89H	Mode	N
TL0	8AH	Timer 0 low-byte	N
TL1	8BH	Timer 1 low-byte	N
TH0	8CH	Timer 0 high-byte	N
TH1	8DH	Timer 1 high-byte	N
TCONB	91H	Mode	N
RL0	92H	Timer 0 reload low-byte	N
RL1	93H	Timer 1 reload low-byte	N
RH0	94H	Timer 0 reload high-byte	N
RH1	95H	Timer 1 reload high-byte	N

Figure 11-7. Timer/Counter 1 PWM Mode 1

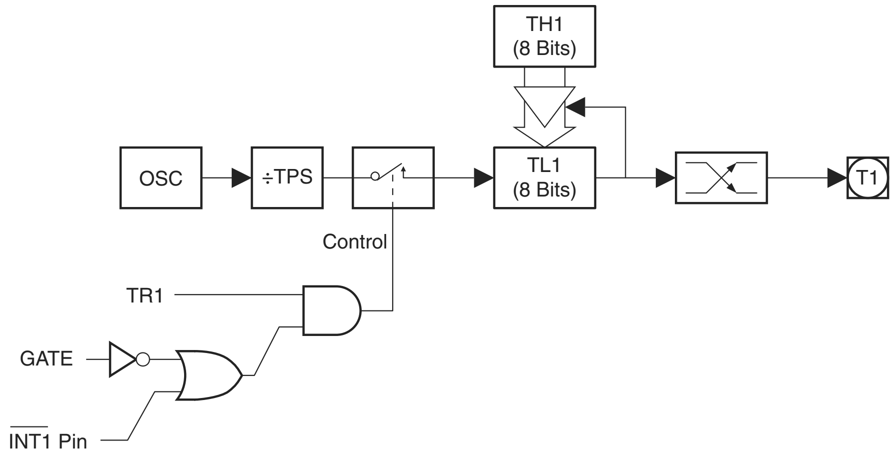


11.5.3 Mode 2 – 8-bit Frequency Generator

Timer 0 in PWM mode 2 functions as an 8-bit Auto-reload timer, the same as normal Mode 2, with the exception that the output pin T0 is toggled at every TL0 overflow (see Figure 11-8 and Figure 11-9 on page 50). Timer 1 in PWM mode 2 is identical to Timer 0. PWM mode 2 can be used to output a square wave of varying frequency. THx acts as an 8-bit counter. The following formula gives the output frequency for Timer 0 in PWM mode 2.

$$\text{Mode 2:} \qquad f_{OUT} = \frac{\text{Oscillator Frequency}}{2 \times (256 - \text{TH0})} \times \frac{1}{\text{TPS} + 1}$$

Figure 11-8. Timer/Counter 1 PWM Mode 2



Note: {RH0 & RL0}/{RH1 & RL1} are not required by Timer 0/Timer 1 during PWM mode 2 and may be used as temporary storage registers.

Table 13-4. T2CCC – Timer/Counter 2 Compare/Capture Control

T2CCC Address = 0D4H Reset Value = 00X0 0000B

Not Bit Addressable

	CIEN*x*	CDIR*x*	–	CTC*x*	CCM*x*	C*x*M2	C*x*M1	C*x*M0
Bit	7	6	5	4	3	2	1	0

Symbol	Function
CIEN*x*	Channel *X* Interrupt Enable. When set, channel *X*'s interrupt flag, CCF*x* in T2CCF, will generate an interrupt when ECC = 1. Clear to disable interrupts from channel *X*.
CDIR*x*	Channel *X* Capture Direction. In dual-slope modes, a compare/capture event on channel *X* will store the current count direction into CDIR*x*. Up-counting = 0 and down-counting = 1. Modifying this bit has no effect.
CTC*x*	Clear Timer on Compare/Capture of Channel *X*. When set, the Timer 2 registers TL2 and TH2 will be cleared by a compare/capture event on channel *X*. When cleared, Timer 2 is unaffected by channel *X* events.
CCM*x*	Channel *X* Compare/Capture Mode. When CCM*x* = 1, channel *X* operates in compare mode. When CCM*x* = 0, channel *X* operates in capture mode.
CxM [2 - 0]	Channel *X* Mode. Selects the output/input events for compare/capture channel *X*.

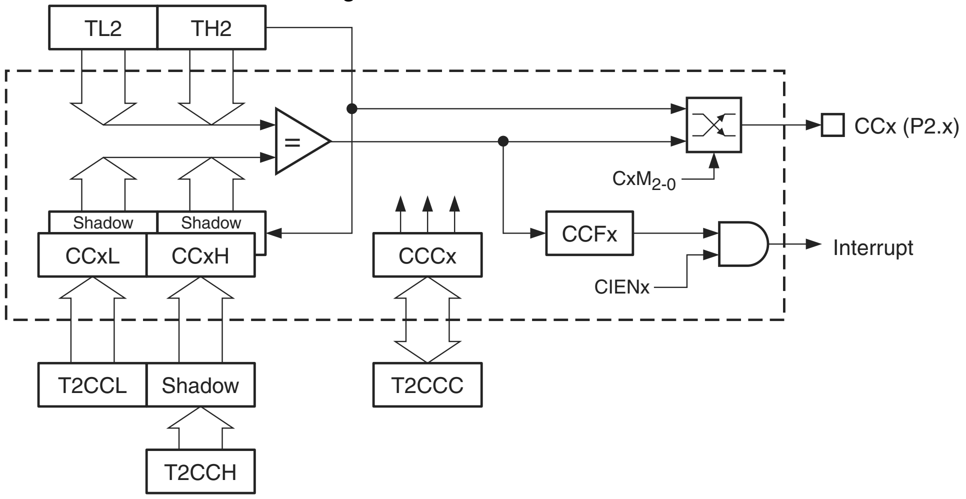
C*x*M2	C*x*M1	C*x*M0	Capture Event (CCM*x* = 0)
0	0	0	Disabled
0	0	1	Trigger on negative edge of CC*x* pin
0	1	0	Trigger on positive edge of CC*x* pin
0	1	1	Trigger on either edge of CC*x* pin
1	0	0	Trigger on Timer 0 overflow
1	0	1	Trigger on Timer 1 overflow
1	1	0	Trigger on Analog Comparator A Event(2)
1	1	1	Trigger on Analog Comparator B Event(3)

C*x*M2	C*x*M1	C*x*M0	Compare Action (CCM*x* = 1)
0	0	0	Output disabled (interrupt only)
0	0	1	Set CC*x* pin on compare match
0	1	0	Clear CC*x* pin on compare match
0	1	1	Toggle CC*x* pin on compare match
1	0	0	Inverting Pulse Width Modulation(4)
1	0	1	Non-inverting Pulse Width Modulation(4)
1	1	0	Reserved
1	1	1	Reserved

Notes:
1. All writes/reads to/from T2CCC will access channel *X* as currently selected by T2CCA.The control registers for the remaining unselected channels are not accessible.
2. Analog Comparator A events are determined by the CMA_{2-0} bits in ACSRA. See Table 18-1 on page 102.
3. Analog Comparator B events are determined by the CMB_{2-0} bits in ACSRB. See Table 18-2 on page 103.
4. Asymmetrical versus Symmetrical PWM is determined by the Timer 2 Count Mode. See “Pulse Width Modulation Mode” on page 68.

and precision is made by changing the TOP value of the timer. The CCA PWM always uses the greatest precision allowable for the selected output frequency, as compared to Timer 0 and 1 whose PWMs are fixed at 8-bit precision regardless of frequency.

Figure 13-7. CCA PWM Mode Diagram



13.4.1 Asymmetrical PWM

For Asymmetrical PWM, Timer 2 should be configured for Auto-reload mode and Count Mode 1 (CP/$\overline{\text{RL2}}$ = 0, DCEN = 0, T2CM$\overline{\text{1-0}}$ = 01B). Asymmetrical PWM uses single slope operation as shown in Figure 13-8. The timer counts up from BOTTOM to TOP and then restarts from BOTTOM. In non-inverting mode, the output CC*x* is set on the compare match between Timer 2 (TL2, TH2) and the channel data register (CC*x*L, CC*x*H), and cleared at BOTTOM. In inverting mode, the output CC*x* is cleared on the compare match between Timer 2 and the data register, and set at BOTTOM. The resulting asymmetrical output waveform is left-edge aligned.

The TOP value in RCAP2L and RCAP2H is double buffered such that the output frequency is only updated at the TOP to BOTTOM overflow. The channel data register (CC*x*L, CC*x*H) is also double-buffered such that the duty cycle is only updated at the TOP to BOTTOM overflow to prevent glitches. The output frequency and duty cycle for asymmetrical PWM are given by the following equations:

$$f_{OUT} = \frac{\text{Oscillator Frequency}}{\{\text{RCAP2H, RCAP2L}\} + 1} \times \frac{1}{\text{TPS} + 1}$$

$$\text{Inverting:} \qquad \text{Duty Cycle} = 100\% \times \frac{\{\text{CCxH, CCxL}\}}{\{\text{RCAP2H, RCAP2L}\} + 1}$$

$$\text{Non-Inverting:} \qquad \text{Duty Cycle} = 100\% \times \frac{\{\text{RCAP2H, RCAP2L}\} - \{\text{CCxH, CCxL}\} + 1}{\{\text{RCAP2H, RCAP2L}\} + 1}$$

The extreme compare values represent special cases when generating a PWM waveform. If the compare value is set equal to (or greater than) TOP, the output will remain low or high for non-inverting and inverting modes, respectively. If the compare value is set to BOTTOM (0000H), the output will remain high or low for non-inverting and inverting modes, respectively.

16.4 More About Mode 1

Ten bits are transmitted (through TXD), or received (through RXD): a start bit (0), 8 data bits (LSB first), and a stop bit (1). On receive, the stop bit goes into RB8 in SCON. In the AT89LP428/828, the baud rate is determined either by the Timer 1 overflow rate, the TImer 2 overflow rate, or both. In this case one timer is for transmit and the other is for receive. Figure 16-4 shows a simplified functional diagram of the serial port in Mode 1 and associated timings for transmit and receive.

Transmission is initiated by any instruction that uses SBUF as a destination register. The "write to SBUF" signal also loads a "1" into the 9th bit position of the transmit shift register and flags the TX Control unit that a transmission is requested. Transmission actually commences at S1P1 of the machine cycle following the next rollover in the divide-by-16 counter. Thus, the bit times are synchronized to the divide-by-16 counter, not to the "write to SBUF" signal.

The transmission begins when $\overline{\text{SEND}}$ is activated, which puts the start bit at TXD. One bit time later, DATA is activated, which enables the output bit of the transmit shift register to TXD. The first shift pulse occurs one bit time after that.

As data bits shift out to the right, "0"s are clocked in from the left. When the MSB of the data byte is at the output position of the shift register, the "1" that was initially loaded into the 9th position is just to the left of the MSB, and all positions to the left of that contain "0"s. This condition flags the TX Control unit to do one last shift, then deactivate $\overline{\text{SEND}}$ and set TI. This occurs at the tenth divide-by-16 rollover after "write to SBUF."

Reception is initiated by a 1-to-0 transition detected at RXD. For this purpose, RXD is sampled at a rate of 16 times the established baud rate. When a transition is detected, the divide-by-16 counter is immediately reset, and 1FFH is written into the input shift register. Resetting the divide-by-16 counter aligns its roll-overs with the boundaries of the incoming bit times.

The 16 states of the counter divide each bit time into 16ths. At the 7th, 8th, and 9th counter states of each bit time, 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 to reject noise. In order to reject false bits, if the value accepted during the first bit time is not 0, the receive circuits are reset and the unit continues looking for another 1-to-0 transition. If the start bit is valid, it is shifted into the input shift register, and reception of the rest of the frame proceeds.

As data bits come in from the right, "1"s shift out to the left. When the start bit arrives at the left-most position in the shift register, (which is a 9-bit register in Mode 1), it flags the RX Control block to do one last shift, load SBUF and RB8, and set RI. The signal to load SBUF and RB8 and to set RI is 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 irretrievably lost. If both conditions are met, the stop bit goes into RB8, the 8 data bits go into SBUF, and RI is activated. At this time, whether or not the above conditions are met, the unit continues looking for a 1-to-0 transition in RXD.

While the TXE flag is set, the transmit buffer is empty. TXE can be cleared by software or by writing to SPDR. Writing to SPDR will clear TXE and load the transmit buffer. The user may load the buffer while the shift register is busy, i.e. before the current transfer completes. When the current transfer completes, the queued byte in the transmit buffer is moved to the shift register and the next transfer commences. TXE will generate an interrupt if the SPI interrupt is enabled and if the ENH bit in SPSR is set. For multi-byte transfers, TXE may be used to remove any dead time between byte transmissions.

The SPI master can operate in two modes: multi-master mode and single-master mode. By default, multi-master mode is active when SSIG = 0. In this mode, the $\overline{SS}$ input is used to disable a master device when another master is accessing the bus. When $\overline{SS}$ is driven low, the master device becomes a slave by clearing its MSTR bit and a Mode Fault is generated by setting the MODF bit in SPSR. MODF will generate an interrupt if enabled. The MSTR bit must be set in software before the device may become a master again. Single-master mode is enabled by setting SSIG = 1. In this mode $\overline{SS}$ is ignored and the master is always active. $\overline{SS}$ may be used as a general-purpose I/O in this mode.

17.2 Slave Operation

When the AT89LP428/828 is not configured for master operation, MSTR = 0, it will operate as an SPI slave. In slave mode, bytes are shifted in through MOSI and out through MISO by a master device controlling the serial clock on SCK. When a byte has been transferred, the SPIF flag is set to “1” and an interrupt request is generated, if enabled. The data received from the addressed master device is also transferred from the shift register to the receive buffer. The received data is accessed by reading SPDR. A slave device cannot initiate transfers. Data to be transferred to the master device must be preloaded by writing to SPDR. Writes to SPDR are double-buffered. The transmit buffer is loaded first and if the shift register is empty, the contents of the buffer will be transferred to the shift register.

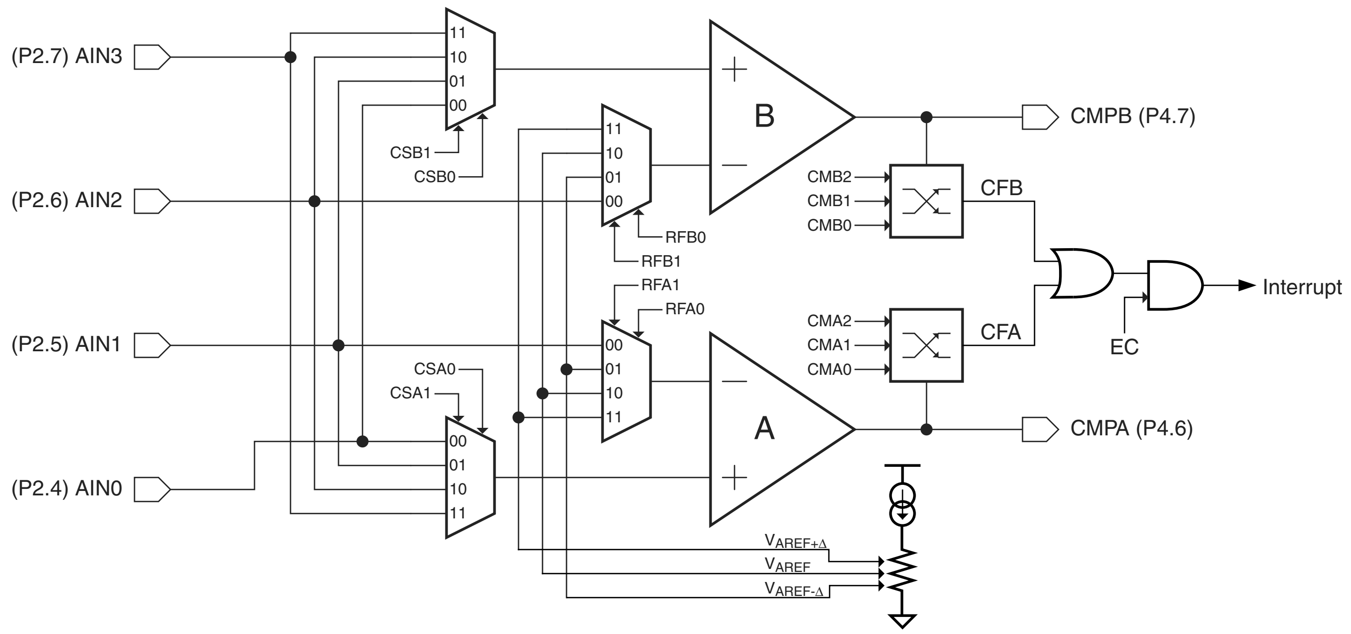
While the TXE flag is set, the transmit buffer is empty. TXE can be cleared by software or by writing to SPDR. Writing to SPDR will clear TXE and load the transmit buffer. The user may load the buffer while the shift register is busy, i.e. before the current transfer completes. When the current transfer completes, the queued byte in the transmit buffer is moved to the shift register and waits for the master to initiate another transfer. TXE will generate an interrupt if the SPI interrupt is enabled and if the ENH bit in SPSR is set.

The SPI slave can operate in two modes: 4-wire mode and 3-wire mode. By default, 4-wire mode is active when SSIG = 0. In this mode, the $\overline{SS}$ input is used to enable/disable the slave device when addressed by a master. When $\overline{SS}$ is driven low, the slave device is enabled and will shift out data on MISO in response to the serial clock on SCK. While $\overline{SS}$ is high, the SPI slave will remain sleeping with MISO inactive. 3-wire mode is enabled by setting SSIG = 1. In this mode $\overline{SS}$ is ignored and the slave is always active. $\overline{SS}$ may be used as a general-purpose I/O in this mode.

The Disable Slave Output bit, DISSO in SPSR, may be used to disable the MISO line of a slave device. DISSO can allow several slave devices to share MISO while operating in 3-wire mode. In this case some protocol other than $\overline{SS}$ may be used to determine which slave is enabled.

interrupt flag are not guaranteed to be stable for 10 µs. 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. Before enabling the comparators, the analog inputs should be tristated by putting P2.4, P2.5, P2.6 and P2.7 into input-only mode. See "Port 2 Analog Functions" on page 38.

Figure 18-1. Dual Comparator Block Diagram



Each comparator may be configured to cause an interrupt under a variety of output value conditions by setting the CMx_{2-0} bits in ACSR*x*. The comparator interrupt flags CF*x* in ACSR*x* are set whenever the comparator outputs match the conditions specified by CM$x_{2-0.}$ The flags may be polled by software or may be used to generate an interrupt and must be cleared by software. Both comparators share a common interrupt vector. If both comparators are enabled, the user needs to read the flags after entering the interrupt service routine to determine which comparator caused the interrupt.

The CAC$_{1-0}$ and CBC$_{1-0}$ bits in AREF control when the comparator interrupts sample the comparator outputs. Normally, the outputs are sampled every clock system; however, the outputs may also be sampled whenever Timer 0, Timer 1 or Timer 2 overflows. These settings allow the comparators to be sampled at a specific time or to reduce the number of comparator events seen by the system when using level sensitive modes. The comparators will continue to function during Idle mode. If this is not the desired behavior, the comparators should be disabled before entering Idle. The comparators are always disabled during Power-down mode.

18.1 Analog Input Muxes

The positive input terminal of each comparator may be connected to any of the four analog input pins by changing the CSA_{1-0} or CSB_{1-0} bits in ACSRA and ACSRB. When changing the analog input pins, the comparator must be disconnected from its inputs by clearing the CONA or CONB bits. The connection is restored by setting the bits again after the muxes have been modified.

```
    CLR  EC           ; Disable comparator interrupts
    ANL  ACSRA, #0DFh ; Clear CONA to disconnect COMP A
    ...               ; Modify CSA or RFA bits
    ORL  ACSRA, #020h ; Set CONA to connect COMP A
    ANL  ACSRA, #0EFh ; Clear any spurious interrupt
   SETB  EC           ; Re-enable comparator interrupts
```

The corresponding comparator interrupt should not be enabled while the inputs are being changed, and the comparator interrupt flag must be cleared before the interrupt is re-enabled in order to prevent an unintentional interrupt request.

18.2 Internal Reference Voltage

The negative input terminal of each comparator may be connected to an internal voltage reference by changing the RFB_{1-0} or RFA_{1-0} bits in AREF. The internal reference voltage, V_{AREF}, is set to 1.25V ±5%. The voltage reference also provides two additional voltage levels approximately 125 mV above and below V_{AREF}. These levels may be used to configure the comparators as an internally referenced window comparator with up to four input channels. Changing the reference input must follow the same routine used for changing the positive input as described in the “Analog Input Muxes” section.

18.3 Comparator Interrupt Debouncing

The comparator output is normally sampled every clock cycle. The conditions on the analog inputs may be such that the comparator output will toggle excessively. This is especially true if applying slow moving analog inputs. Three debouncing modes are provided to filter out this noise for edge-triggered interrupts. In debouncing mode, the comparator uses Timer 1 to modulate its sampling time when CxC_{1-0} = 00B. When a relevant transition occurs, the comparator waits until two Timer 1 overflows have occurred before resampling the output. If the new sample agrees with the expected value, CF*x* is set. Otherwise, the event is ignored. The filter may be tuned by adjusting the time-out period of Timer 1. Because Timer 1 is running free, the debouncer must wait for two overflows to guarantee that the sampling delay is at least 1 time-out period. Therefore, after the initial edge event, the interrupt may occur between 1 and 2 time-out periods later. See Figure 18-2. When the comparator clock is provided by one of the timer overflows, i.e. CxC_{1-0} = 00B, any change in the comparator output must be valid after 4 samples to be accepted as an edge event.

Figure 18-2. Negative Edge with Debouncing Example

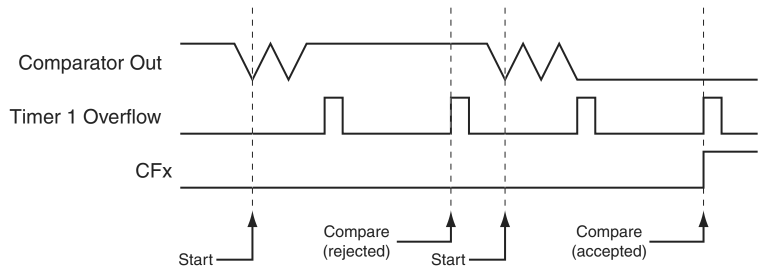


Table 20-1. Instruction Execution Times and Exceptions (Continued)

MOV direct, A	2	12	2	F5
MOV direct, Rn	2	24	2	88 - 8F
MOV direct, direct	3	24	3	85
MOV direct, @Ri	2	24	2	86 - 87
MOV direct, #data	3	24	3	75
MOV @Ri, A	1	12	1	F6 - F7
MOV @Ri, direct	2	24	2	A6 - A7
MOV @Ri, #data	2	12	2	76 - 77
MOV DPTR, #data16	3	24	3	90
MOV /DPTR, #data16(1)	4	–	4	A5 90
MOVC A, @A+DPTR	1	24	3	93
MOVC A, @A+/DPTR(1)	2	–	4	A5 93
MOVC A, @A+PC	1	24	3	83
MOVX A, @Ri	1	24	2	E2 - E3
MOVX A, @DPTR	1	24	2/4(2)	E0
MOVX A, @/DPTR(1)	2	–	3/5(2)	A5 E0
MOVX @Ri, A	1	24	2	F2 - F3
MOVX @DPTR, A	1	24	2/4(2)	F0
MOVX @/DPTR, A(1)	2	–	3/5(2)	A5 F0
PUSH direct	2	24	2	C0
POP direct	2	24	2	D0
XCH A, Rn	1	12	1	C8 - CF
XCH A, direct	2	12	2	C5
XCH A, @Ri	1	12	2	C6 - C7
XCHD A, @Ri	1	12	2	D6 - D7
		Clock Cycles		
Bit Operations	**Bytes**	**8051**	**AT89LP**	**Hex Code**
CLR C	1	12	1	C3
CLR bit	2	12	2	C2
SETB C	1	12	1	D3
SETB bit	2	12	2	D2
CPL C	1	12	1	B3
CPL bit	2	12	2	B2
ANL C, bit	2	24	2	82
ANL C, bit	2	24	2	B0
ORL C, bit	2	24	2	72
ORL C, /bit	2	24	2	A0
MOV C, bit	2	12	2	A2
MOV bit, C	2	24	2	92

23. Programming the Flash Memory

The Atmel AT89LP428/828 microcontroller features 4K/8K bytes of on-chip In-System Programmable Flash program memory and 512/1024 bytes of nonvolatile Flash data memory. In-System Programming allows programming and reprogramming of the microcontroller positioned inside the end system. Using a simple 4-wire SPI interface, the programmer communicates serially with the AT89LP428/828 microcontroller, reprogramming all nonvolatile memories on the chip. In-System Programming eliminates the need for physical removal of the chips from the system. This will save time and money, both during development in the lab, and when updating the software or parameters in the field. The programming interface of the AT89LP428/828 includes the following features:

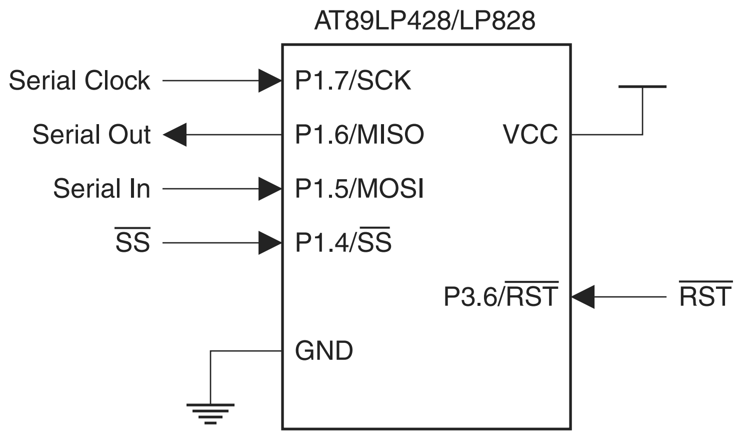
- 4-wire SPI Programming Interface
- Active-low Reset Entry into Programming
- Slave Select Allows Multiple Devices on Same Interface
- User Signature Array
- Flexible Page Programming
- Row Erase Capability
- Page Write with Auto-Erase Commands
- Programming Status Register

For more detailed information on In-System Programming, refer to the Application Note entitled "AT89LP In-System Programming Specification".

23.1 Physical Interface

Flash Programming utilizes the Serial Peripheral Interface (SPI) pins of an AT89LP428/828 microcontroller. The SPI is a full-duplex synchronous serial interface consisting of four wires: Serial Clock (SCK), Master-in/Slave-out (MISO), Master-out/Slave-in (MOSI), and an active-low Slave Select ($\overline{\text{SS}}$). When programming an AT89LP428/828 device, the programmer always operates as the SPI master, and the target system always operates as the SPI slave. To enter or remain in Programming mode the device's reset line ($\overline{\text{RST}}$) must be held active (low). With the addition of VCC and GND, an AT89LP428/828 microcontroller can be programmed with a minimum of seven connections as shown in Figure 23-1.

Figure 23-1. In-System Programming Device Connections



The Programming Interface is the only means of externally programming the AT89LP428/828 microcontroller. The Interface can be used to program the device both in-system and in a stand-alone serial programmer. The Interface does not require any clock other than SCK and is not limited by the system clock frequency. During Programming, the system clock source of the target device can operate normally.

When designing a system where In-System Programming will be used, the following observations must be considered for correct operation:

- The ISP interface uses the SPI clock mode 0 (CPOL = 0, CPHA = 0) exclusively with a maximum frequency of 5 MHz.
- The AT89LP428/828 will enter programming mode only when its reset line ($\overline{\text{RST}}$) is active (low). To simplify this operation, it is recommended that the target reset can be controlled by the In-System programmer. To avoid problems, the In-System programmer should be able to keep the entire target system reset for the duration of the programming cycle. The target system should never attempt to drive the four SPI lines while reset is active.
- The $\overline{\text{RST}}$ input may be disabled to gain an extra I/O pin. In these cases, the $\overline{\text{RST}}$ pin will always function as a reset during power up. To enter programming the $\overline{\text{RST}}$ pin must be driven low prior to the end of Power-on Reset (POR). After POR has completed, the device will remain in ISP mode until $\overline{\text{RST}}$ is brought high. Once the initial ISP session has ended, the power to the target device must be cycled OFF and ON to enter another session.
- The $\overline{\text{SS}}$ pin should not be left floating during reset if ISP is enabled.
- The ISP Enable Fuse must be set to allow programming during any reset period. If the ISP Fuse is disabled, ISP may only be entered at POR.
- For standalone programmers, $\overline{\text{RST}}$ may be tied directly to GND to ensure correct entry into Programming mode regardless of the device settings.

23.2 Memory Organization

The AT89LP428/828 offers 4K/8K bytes of In-System Programmable (ISP) nonvolatile Flash code memory and 512/1024 bytes of nonvolatile Flash data memory. In addition, the device contains a 128-byte User Signature Array and a 64-byte read-only Atmel Signature Array. The memory organization is shown in Tables 23-1 and 23-2 and Figure 23-2. The memory is divided into pages of 64 bytes each. A single read or write command may only access a single page in the memory. Each memory type resides in its own address space and is accessed by commands specific to that memory. However, all memory types share the same page size.

User configuration fuses are mapped as a row in the memory, with each byte representing one fuse. From a programming standpoint, fuses are treated the same as normal code bytes except they are not affected by Chip Erase. Fuses can be enabled at any time by writing 00h to the appropriate locations in the fuse row. However, to disable a fuse, i.e. set it to FFh, the **entire** fuse row must be erased and then reprogrammed. The programmer should read the state of all the fuses into a temporary location, modify those fuses which need to be disabled, then issue a Fuse Write with Auto-Erase command using the temporary data. Lock bits are treated in a similar manner to fuses except they may only be erased (unlocked) by Chip Erase.

Table 23-1. Code Memory Size

Device #	Code Size	Page Size	# Pages	Address Range
AT89LP428	4K bytes	64 bytes	64	0000H - 0FFFH
AT89LP828	8K bytes	64 bytes	128	0000H - 1FFFH

24. Electrical Characteristics

24.1 Absolute Maximum Ratings*

Operating Temperature	-40°C to +85°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-0.7V to +5.5V
Maximum Operating Voltage	5.5V
DC Output Current	15.0 mA

*NOTICE: Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

24.2 DC Characteristics

T_A = -40°C to 85°C, V_{CC} = 2.4V to 5.5V (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low-voltage		-0.5	0.3 V_{CC}	V
V_{IH}	Input High-voltage		0.7 V_{CC}	V_{CC} + 0.5	V
V_{OL}	Output Low-voltage[(1)]	I_{OL} = 8 mA, V_{CC} = 5V ± 10%		0.4	V
		I_{OL} = 4 mA		0.4	
V_{OH}	Output High-voltage With Weak Pull-ups Enabled	I_{OH} = -100 μA, V_{CC} = 5V ± 10%	2.4		V
		I_{OH} = -25 μA	0.75 V_{CC}		V
		I_{OH} = -10 μA	0.9 V_{CC}		V
V_{OH1}	Output High-voltage With Strong Pull-ups Enabled	I_{OH} = -20 mA, V_{CC} = 5V ± 10%	0.75 V_{CC}		
		I_{OH} = -8 mA, V_{CC} = 5V ± 10%	0.9 V_{CC}		
		I_{OH} = -6 mA	0.75 V_{CC}		
		I_{OH} = -2 mA	0.9 V_{CC}		
I_{IL}	Logic 0 Input Current	V_{IN} = 0.45V		-100	μA
I_{TL}	Logic 1 to 0 Transition Current	V_{IN} = 2.7V, V_{CC} = 5V ± 10%		-300	μA
I_{LI}	Input Leakage Current	0 < V_{IN} < V_{CC}		±10	μA
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, T_A = 25°C		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz, V_{CC} = 5V/3V		10/6	mA
		Idle Mode, 12 MHz, V_{CC} = 5V/3V		5/3	mA
	Power-down Mode[(2)]	V_{CC} = 5V		5	μA
		V_{CC} = 3V		2	μA

Notes: 1. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
Maximum I_{OL} per port pin: 10 mA
Maximum total I_{OL} for all output pins: 15 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.
2. Minimum V_{CC} for Power-down is 2V.

24.3.2 Supply Current (External Clock)

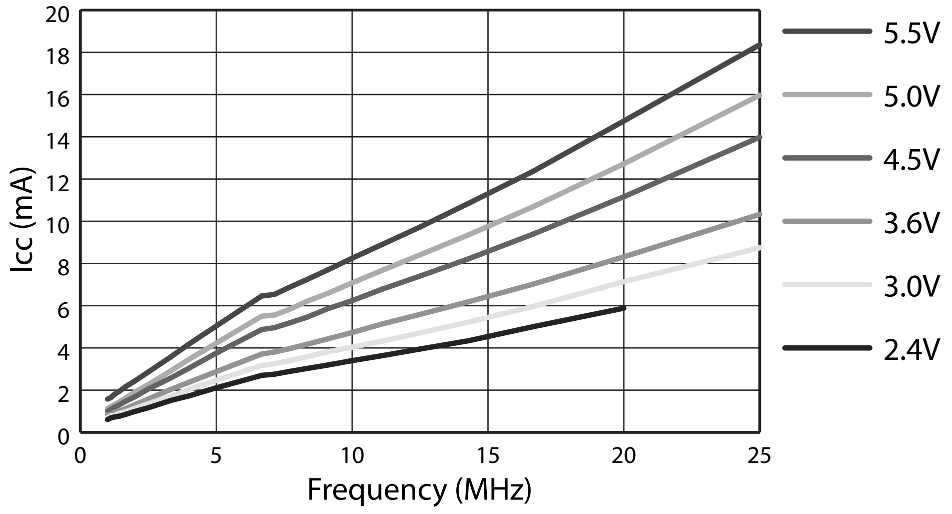


Figure 24-3. Active Supply Current vs. Frequency

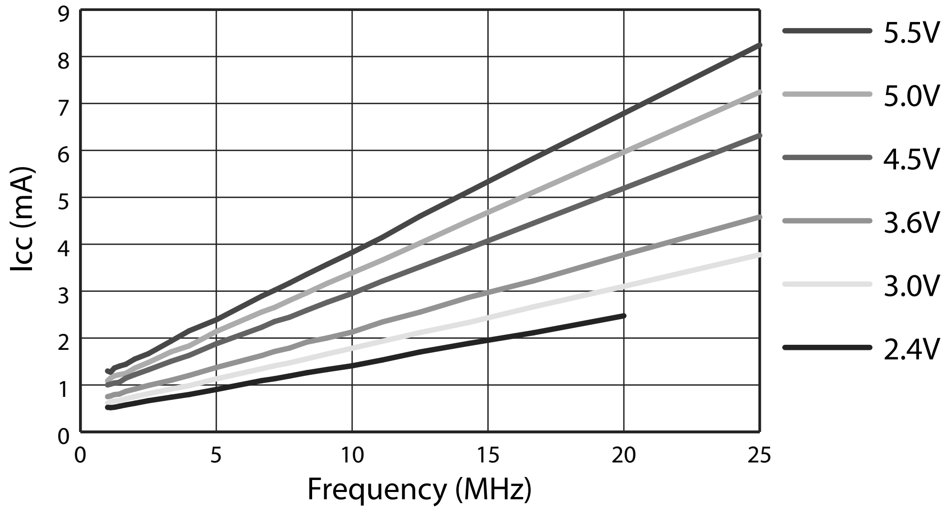


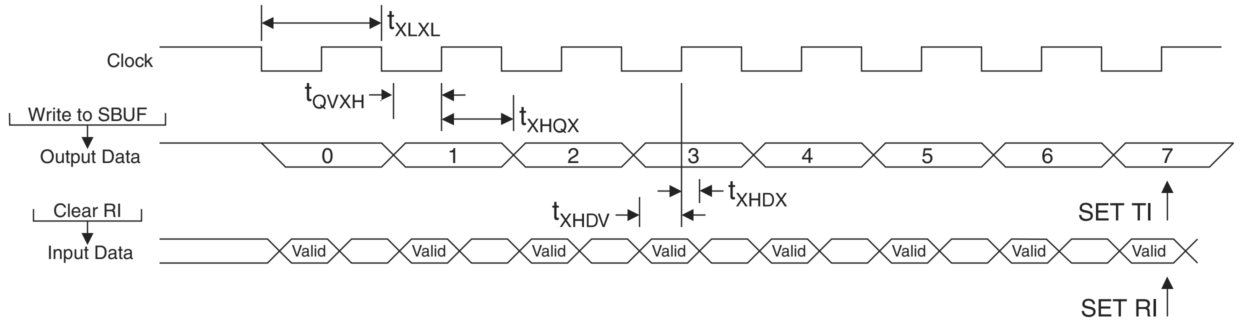
Figure 24-4. Idle Supply Current vs. Frequency

24.8 Serial Port Timing: Shift Register Mode

The values in this table are valid for V_{CC} = 2.4V to 5.5V and Load Capacitance = 80 pF.

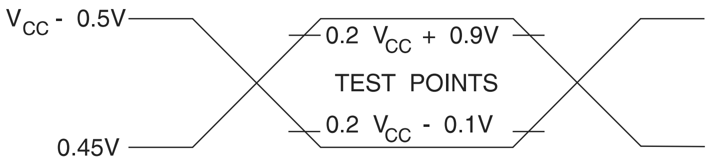
Symbol	Parameter	Variable Oscillator		Units
		Min	Max	
t_{XLXL}	Serial Port Clock Cycle Time	$2t_{CLCL}$ -15		μs
t_{QVXH}	Output Data Setup to Clock Rising Edge	t_{CLCL} -15		ns
t_{XHQX}	Output Data Hold after Clock Rising Edge	t_{CLCL} -15		ns
t_{XHDX}	Input Data Hold after Clock Rising Edge	0		ns
t_{XHDV}	Input Data Valid to Clock Rising Edge	15		ns

Figure 24-15. Shift Register Mode Timing Waveform



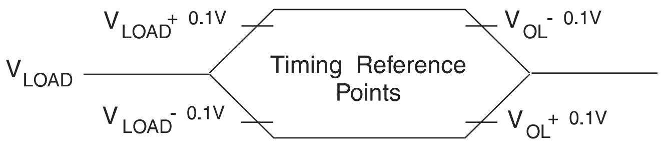
24.9 Test Conditions

24.9.1 AC Testing Input/Output Waveform(1)



Note: 1. AC Inputs during testing are driven at V_{CC} - 0.5V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at V_{IH} min. for a logic "1" and V_{IL} max. for a logic "0".

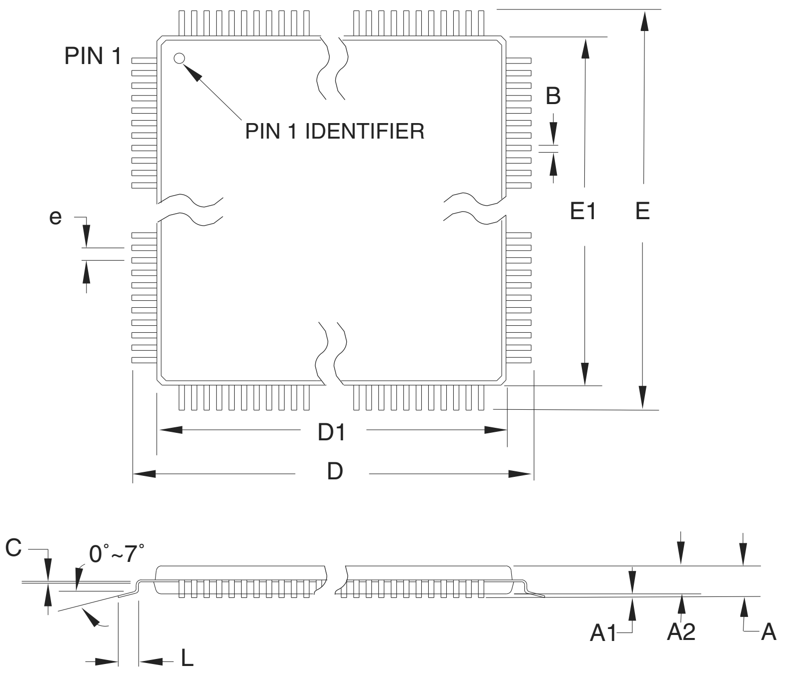
24.9.2 Float Waveform(1)



Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

26. Packaging Information

26.1 32A – TQFP



COMMON DIMENSIONS
(Unit of Measure = mm)

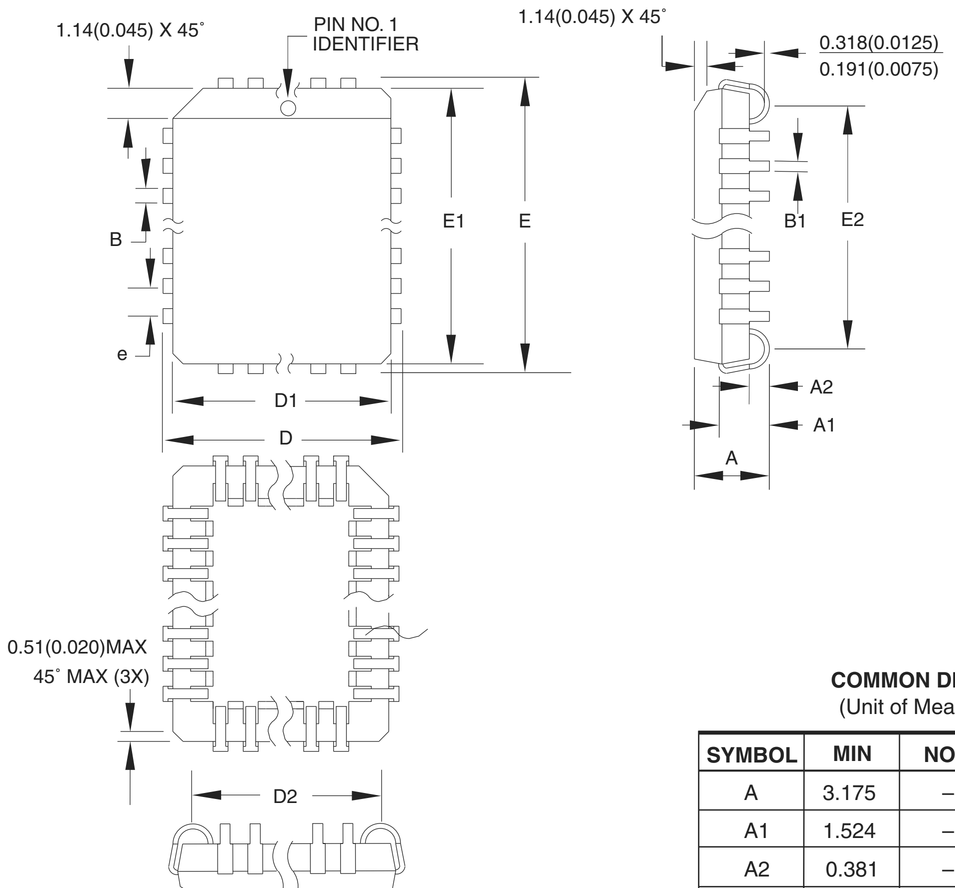
SYMBOL	MIN	NOM	MAX	NOTE
A	–	–	1.20	
A1	0.05	–	0.15	
A2	0.95	1.00	1.05	
D	8.75	9.00	9.25	
D1	6.90	7.00	7.10	Note 2
E	8.75	9.00	9.25	
E1	6.90	7.00	7.10	Note 2
B	0.30	–	0.45	
C	0.09	–	0.20	
L	0.45	–	0.75	
e	0.80 TYP			

Notes:
1. This package conforms to JEDEC reference MS-026, Variation ABA.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is 0.25 mm per side. Dimensions D1 and E1 are maximum plastic body size dimensions including mold mismatch.
3. Lead coplanarity is 0.10 mm maximum.

10/5/2001

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**32A,** 32-lead, 7 x 7 mm Body Size, 1.0 mm Body Thickness, 0.8 mm Lead Pitch, Thin Profile Plastic Quad Flat Package (TQFP)	32A	B

26.3 32J – PLCC



Notes: 1. This package conforms to JEDEC reference MS-016, Variation AE.
2. Dimensions D1 and E1 do not include mold protrusion. Allowable protrusion is .010"(0.254 mm) per side. Dimension D1 and E1 include mold mismatch and are measured at the extreme material condition at the upper or lower parting line.
3. Lead coplanarity is 0.004" (0.102 mm) maximum.

COMMON DIMENSIONS
(Unit of Measure = mm)

SYMBOL	MIN	NOM	MAX	NOTE
A	3.175	–	3.556	
A1	1.524	–	2.413	
A2	0.381	–	–	
D	12.319	–	12.573	
D1	11.354	–	11.506	Note 2
D2	9.906	–	10.922	
E	14.859	–	15.113	
E1	13.894	–	14.046	Note 2
E2	12.471	–	13.487	
B	0.660	–	0.813	
B1	0.330	–	0.533	
e	1.270 TYP			

10/04/01

	TITLE	DRAWING NO.	REV.
2325 Orchard Parkway San Jose, CA 95131	**32J**, 32-lead, Plastic J-leaded Chip Carrier (PLCC)	32J	B