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	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, WDT
Number of I/O	25
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f041asj020ec

addresses outside the available Flash memory addresses returns FFH. Writing to these unimplemented Program Memory addresses produces no effect. Table 5 describes the Program Memory Maps for the Z8 Encore! XP F082A Series products.

Table 5. Z8 Encore! XP F082A Series Program Memory Maps

Program Memory Address (Hex)	Function
Z8F082A and Z8F081A Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–0039	Reserved
003A–003D	Oscillator Fail Trap Vectors
003E–1FFF	Program Memory
Z8F042A and Z8F041A Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–0039	Reserved
003A–003D	Oscillator Fail Trap Vectors
003E–0FFF	Program Memory

Reset, Stop Mode Recovery, and Low Voltage Detection

The Reset Controller within the Z8 Encore! XP® F082A Series controls Reset and Stop Mode Recovery operation and provides indication of low supply voltage conditions. In typical operation, the following events cause a Reset:

- Power-On Reset (POR)
- Voltage Brownout (VBO)
- Watchdog Timer time-out (when configured by the WDT_RES Flash Option Bit to initiate a reset)
- External $\overline{\text{RESET}}$ pin assertion (when the alternate RESET function is enabled by the GPIO register)
- On-chip debugger initiated Reset (OCDCTL[0] set to 1)

When the device is in STOP mode, a Stop Mode Recovery is initiated by either of the following:

- Watchdog Timer time-out
- GPIO Port input pin transition on an enabled Stop Mode Recovery source

The low voltage detection circuitry on the device (available on the 8-pin product versions only) performs the following functions:

- Generates the VBO reset when the supply voltage drops below a minimum safe level.
- Generates an interrupt when the supply voltage drops below a user-defined level (8-pin devices only).

Reset Types

The Z8 Encore! XP F082A Series provides several different types of Reset operation. Stop Mode Recovery is considered as a form of Reset. Table 8 lists the types of Reset and their operating characteristics. The System Reset is longer if the external crystal oscillator is enabled by the Flash option bits, allowing additional time for oscillator start-up.

Low-Power Modes

The Z8 Encore! XP F082A Series products contain power-saving features. The highest level of power reduction is provided by the STOP mode, in which nearly all device functions are powered down. The next lower level of power reduction is provided by the HALT mode, in which the CPU is powered down.

Further power savings can be implemented by disabling individual peripheral blocks while in Active mode (defined as being in neither STOP nor HALT mode).

STOP Mode

Executing the eZ8 CPU's STOP instruction places the device into STOP mode, powering down all peripherals except the Voltage Brownout detector, the Low-power Operational Amplifier and the Watchdog Timer. These three blocks may also be disabled for additional power savings. Specifically, the operating characteristics are:

- Primary crystal oscillator and internal precision oscillator are stopped; XIN and XOUT (if previously enabled) are disabled, and PA0/PA1 revert to the states programmed by the GPIO registers.
- System clock is stopped.
- eZ8 CPU is stopped.
- Program counter (PC) stops incrementing.
- Watchdog Timer's internal RC oscillator continues to operate if enabled by the Oscillator Control register.
- If enabled, the Watchdog Timer logic continues to operate.
- If enabled for operation in STOP mode by the associated Flash Option Bit, the Voltage Brownout protection circuit continues to operate.
- Low-power operational amplifier continues to operate if enabled by the Power Control register to do so.
- All other on-chip peripherals are idle.

To minimize current in STOP mode, all GPIO pins that are configured as digital inputs must be driven to one of the supply rails (V_{CC} or GND). Additionally, any GPIOs configured as outputs must also be driven to one of the supply rails. The device can be brought out of STOP mode using Stop Mode Recovery. For more information on Stop Mode Recovery, see Reset, Stop Mode Recovery, and Low Voltage Detection on page 23.

HALT Mode

Executing the eZ8 CPU's HALT instruction places the device into HALT mode, which powers down the CPU but leaves all other peripherals active. In HALT mode, the operating characteristics are:

- Primary oscillator is enabled and continues to operate.
- System clock is enabled and continues to operate.
- eZ8 CPU is stopped.
- Program counter (PC) stops incrementing.
- Watchdog Timer's internal RC oscillator continues to operate.
- If enabled, the Watchdog Timer continues to operate.
- All other on-chip peripherals continue to operate, if enabled.

The eZ8 CPU can be brought out of HALT mode by any of the following operations:

- Interrupt
- Watchdog Timer time-out (interrupt or reset)
- Power-On Reset
- Voltage Brownout reset
- External $\overline{\text{RESET}}$ pin assertion

To minimize current in HALT mode, all GPIO pins that are configured as inputs must be driven to one of the supply rails (V_{CC} or GND).

Peripheral-Level Power Control

In addition to the STOP and HALT modes, it is possible to disable each peripheral on each of the Z8 Encore! XP F082A Series devices. Disabling a given peripheral minimizes its power consumption.

Power Control Register Definitions

The following sections define the Power Control registers.

Power Control Register 0

Each bit of the following registers disables a peripheral block, either by gating its system clock input or by removing power from the block. The default state of the low-power

Interrupt Controller

The interrupt controller on the Z8 Encore! XP F082A Series products prioritizes the interrupt requests from the on-chip peripherals and the GPIO port pins. The features of interrupt controller include:

- 20 possible interrupt sources with 18 unique interrupt vectors:
  - Twelve GPIO port pin interrupt sources (two interrupt vectors are shared).
  - Eight on-chip peripheral interrupt sources (two interrupt vectors are shared).
- Flexible GPIO interrupts:
  - Eight selectable rising and falling edge GPIO interrupts.
  - Four dual-edge interrupts.
- Three levels of individually programmable interrupt priority.
- Watchdog Timer and LVD can be configured to generate an interrupt.
- Supports vectored as well as polled interrupts

Interrupt requests (IRQs) allow peripheral devices to suspend CPU operation in an orderly manner and force the CPU to start an interrupt service routine (ISR). Usually this interrupt service routine is involved with the exchange of data, status information, or control information between the CPU and the interrupting peripheral. When the service routine is completed, the CPU returns to the operation from which it was interrupted.

The eZ8 CPU supports both vectored and polled interrupt handling. For polled interrupts, the interrupt controller has no effect on operation. For more information on interrupt servicing by the eZ8 CPU, refer to *eZ8 CPU Core User Manual (UM0128)* available for download at www.zilog.com.

Interrupt Vector Listing

Table 32 on page 56 lists all of the interrupts available in order of priority. The interrupt vector is stored with the most-significant byte (MSB) at the even Program Memory address and the least-significant byte (LSB) at the following odd Program Memory address.

Note: *Some port interrupts are not available on the 8- and 20-pin packages. The ADC interrupt is unavailable on devices not containing an ADC.*

Interrupt Request 2 Register

The Interrupt Request 2 (IRQ2) register (Table 35) stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ2 register becomes 1. If interrupts are globally enabled (vectored interrupts), the interrupt controller passes an interrupt request to the eZ8 CPU. If interrupts are globally disabled (polled interrupts), the eZ8 CPU can read the Interrupt Request 2 register to determine if any interrupt requests are pending.

Table 35. Interrupt Request 2 Register (IRQ2)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved				PC3I	PC2I	PC1I	PC0I
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FC6H							

Reserved—Must be 0.

PC*x*I—Port C Pin *x* Interrupt Request
0 = No interrupt request is pending for GPIO Port C pin *x*.
1 = An interrupt request from GPIO Port C pin *x* is awaiting service.

where *x* indicates the specific GPIO Port C pin number (0–3).

IRQ0 Enable High and Low Bit Registers

Table 36 describes the priority control for IRQ0. The IRQ0 Enable High and Low Bit registers (Table 37 and Table 38) form a priority encoded enabling for interrupts in the Interrupt Request 0 register.

Table 36. IRQ0 Enable and Priority Encoding

IRQ0ENH[*x*]	IRQ0ENL[*x*]	Priority	Description
0	0	Disabled	Disabled
0	1	Level 1	Low
1	0	Level 2	Medium
1	1	Level 3	High

where *x* indicates the register bits from 0–7.

Reserved—Must be 0.

C3ENL—Port C3 Interrupt Request Enable Low Bit
C2ENL—Port C2 Interrupt Request Enable Low Bit
C1ENL—Port C1 Interrupt Request Enable Low Bit
C0ENL—Port C0 Interrupt Request Enable Low Bit

Interrupt Edge Select Register

The Interrupt Edge Select (IRQES) register (Table 45) determines whether an interrupt is generated for the rising edge or falling edge on the selected GPIO Port A input pin.

Table 45. Interrupt Edge Select Register (IRQES)

BITS	7	6	5	4	3	2	1	0
FIELD	IES7	IES6	IES5	IES4	IES3	IES2	IES1	IES0
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FCDH							

IES*x*—Interrupt Edge Select *x*
0 = An interrupt request is generated on the falling edge of the PA*x* input.
1 = An interrupt request is generated on the rising edge of the PA*x* input.
where *x* indicates the specific GPIO Port pin number (0 through 7).

Shared Interrupt Select Register

The Shared Interrupt Select (IRQSS) register (Table 46) determines the source of the PADxS interrupts. The Shared Interrupt Select register selects between Port A and alternate sources for the individual interrupts.

Because these shared interrupts are edge-triggered, it is possible to generate an interrupt just by switching from one shared source to another. For this reason, an interrupt must be disabled before switching between sources.

Follow the steps below for configuring a timer for CAPTURE mode and initiating the count:

1. Write to the Timer Control register to:
   - Disable the timer.
   - Configure the timer for CAPTURE mode.
   - Set the prescale value.
   - Set the Capture edge (rising or falling) for the Timer Input.
2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H).
3. Write to the Timer Reload High and Low Byte registers to set the Reload value.
4. Clear the Timer PWM High and Low Byte registers to 0000H. Clearing these registers allows the software to determine if interrupts were generated by either a capture event or a reload. If the PWM High and Low Byte registers still contain 0000H after the interrupt, the interrupt was generated by a Reload.
5. Enable the timer interrupt, if appropriate, and set the timer interrupt priority by writing to the relevant interrupt registers. By default, the timer interrupt is generated for both input capture and reload events. If appropriate, configure the timer interrupt to be generated only at the input capture event or the reload event by setting TICONFIG field of the TxCTL0 register.
6. Configure the associated GPIO port pin for the Timer Input alternate function.
7. Write to the Timer Control register to enable the timer and initiate counting.

In CAPTURE mode, the elapsed time from timer start to Capture event can be calculated using the following equation:

$$\text{Capture Elapsed Time (s)} = \frac{(\text{Capture Value} - \text{Start Value}) \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

CAPTURE RESTART Mode

In CAPTURE RESTART mode, the current timer count value is recorded when the acceptable external Timer Input transition occurs. The Capture count value is written to the Timer PWM High and Low Byte Registers. The timer input is the system clock. The TPOL bit in the Timer Control register determines if the Capture occurs on a rising edge or a falling edge of the Timer Input signal. When the Capture event occurs, an interrupt is generated and the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. The INPCAP bit in TxCTL0 register is set to indicate the timer interrupt is because of an input capture event.

If no Capture event occurs, the timer counts up to the 16-bit Compare value stored in the Timer Reload High and Low Byte registers. Upon reaching the Reload value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to

Table 70. UART Baud Rates (Continued)

3.579545 MHz System Clock				1.8432 MHz System Clock			
Acceptable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Acceptable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	N/A	N/A	N/A	1250.0	N/A	N/A	N/A
625.0	N/A	N/A	N/A	625.0	N/A	N/A	N/A
250.0	1	223.72	-10.51	250.0	N/A	N/A	N/A
115.2	2	111.9	-2.90	115.2	1	115.2	0.00
57.6	4	55.9	-2.90	57.6	2	57.6	0.00
38.4	6	37.3	-2.90	38.4	3	38.4	0.00
19.2	12	18.6	-2.90	19.2	6	19.2	0.00
9.60	23	9.73	1.32	9.60	12	9.60	0.00
4.80	47	4.76	-0.83	4.80	24	4.80	0.00
2.40	93	2.41	0.23	2.40	48	2.40	0.00
1.20	186	1.20	0.23	1.20	96	1.20	0.00
0.60	373	0.60	-0.04	0.60	192	0.60	0.00
0.30	746	0.30	-0.04	0.30	384	0.30	0.00

Caution: *Although the ADC can be used without the gain and offset compensation, it does exhibit non-unity gain. Designing the ADC with sub-unity gain reduces noise across the ADC range but requires the ADC results to be scaled by a factor of 8/7.*

ADC Compensation Details

High efficiency assembly code that performs this compensation is available for download on www.zilog.com. The following is a bit-specific description of the ADC compensation process used by this code.

The following data bit definitions are used:

0-9, a-f = bit indices in hexadecimal
s = sign bit
v = overflow bit
- = unused

Input Data

MSB	LSB	
s b a 9 8 7 6 5	4 3 2 1 0 - - v (ADC)	ADC Output Word; if v = 1, the data is invalid
	s 6 5 4 3 2 1 0	Offset Correction Byte
s s s s s 7 6 5	4 3 2 1 0 0 0 0 (Offset)	Offset Byte shifted to align with ADC data
s e d c b a 9 8	7 6 5 4 3 2 1 0 (Gain)	Gain Correction Word

#5 MSB	#5 LSB

6. Add the gain correction factor to the original offset corrected value.

#5 MSB	#5 LSB

+

#1 MSB	#1 LSB

=

#6 MSB	#6 LSB

7. Shift the result to the right, using the sign bit determined in Step 1. This allows for the detection of computational overflow.

S->	#6 MSB	#6 LSB

Output Data

The following is the output format of the corrected ADC value.

MSB	LSB
s v b a 9 8 7 6	5 4 3 2 1 0 - -

The overflow bit in the corrected output indicates that the computed value was greater than the maximum logical value (+1023) or less than the minimum logical value (-1024). Unlike the hardware overflow bit, this is not a simple binary Flag. For a normal sample (non-overflow), the sign and the overflow bit matches. If the sign bit and overflow bit do not match, a computational overflow has occurred.

Input Buffer Stage

Many applications require the measurement of an input voltage source with a high output impedance. This ADC provides a buffered input for such situations. The drawback of the buffered input is a limitation of the input range. When using unity gain buffered mode, the input signal must be prevented from coming too close to either V_{SS} or V_{DD}. See Table 135 on page 231 for details.

This condition applies only to the input voltage level (with respect to ground) of each differential input signal. The actual differential input voltage magnitude may be less than 300 mV.

The input range of the unbuffered ADC swings from V_{SS} to V_{DD}. Input signals smaller than 300 mV must use the unbuffered input mode. If these signals do not contain low output impedances, they might require off-chip buffering.

Signals outside the allowable input range can be used without instability or device damage. Any ADC readings made outside the input range are subject to greater inaccuracy than specified.

```
nop      ; wait for output to settle
clr irq0 ; clear any spurious interrupts pending
ei
```

Comparator Control Register Definitions

Comparator Control Register

The Comparator Control Register (CMP0) configures the comparator inputs and sets the value of the internal voltage reference.

Table 75. Comparator Control Register (CMP0)

BITS	7	6	5	4	3	2	1	0
FIELD	INPSEL	INNSEL	REFLVL				Reserved (20-/28-pin) REFLVL (8-pin)	
RESET	0	0	0	1	0	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F90H							

INPSEL—Signal Select for Positive Input
0 = GPIO pin used as positive comparator input
1 = temperature sensor used as positive comparator input

INNSEL—Signal Select for Negative Input
0 = internal reference disabled, GPIO pin used as negative comparator input
1 = internal reference enabled as negative comparator input

REFLVL—Internal Reference Voltage Level (this reference is independent of the ADC voltage reference). Note that the 8-pin devices contain two additional LSBs for increased resolution.

For 20-/28-pin devices:

0000 = 0.0 V
0001 = 0.2 V
0010 = 0.4 V
0011 = 0.6 V
0100 = 0.8 V
0101 = 1.0 V (Default)
0110 = 1.2 V
0111 = 1.4 V
1000 = 1.6 V

Randomized Lot Identification Bits

As an optional feature, Zilog is able to provide a factory-programmed random lot identifier. With this feature, all devices in a given production lot are programmed with the same random number. This random number is uniquely regenerated for each successive production lot and is not likely to be repeated.

The randomized lot identifier is a 32 byte binary value, stored in the Flash information page (see Reading the Flash Information Page on page 155 and Randomized Lot Identifier on page 166 for more details) and is unaffected by mass erasure of the device's Flash memory.

Reading the Flash Information Page

The following code example shows how to read data from the Flash information area.

```
; get value at info address 60 (FE60h)
ldx FPS, #%80 ; enable access to flash info page
ld R0, #%FE
ld R1, #%60
ldc R2, @RR0 ; R2 now contains the calibration value
```

Flash Option Bit Control Register Definitions

Trim Bit Address Register

The Trim Bit Address (TRMADR) register contains the target address for an access to the trim option bits (Table 84).

Table 84. Trim Bit Address Register (TRMADR)

BITS	7	6	5	4	3	2	1	0
FIELD	TRMADR - Trim Bit Address (00H to 1FH)							
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FF6H							

Crystal Oscillator

The products in the Z8 Encore! XP® F082A Series contain an on-chip crystal oscillator for use with external crystals with 32 kHz to 20 MHz frequencies. In addition, the oscillator supports external RC networks with oscillation frequencies up to 4 MHz or ceramic resonators with frequencies up to 8 MHz. The on-chip crystal oscillator can be used to generate the primary system clock for the internal eZ8 CPU and the majority of the on-chip peripherals. Alternatively, the X_{IN} input pin can also accept a CMOS-level clock input signal (32 kHz–20 MHz). If an external clock generator is used, the X_{OUT} pin must be left unconnected. The Z8 Encore! XP F082A Series products do not contain an internal clock divider. The frequency of the signal on the X_{IN} input pin determines the frequency of the system clock.

Note: *Although the XIN pin can be used as an input for an external clock generator, the CLKIN pin is better suited for such use (see* System Clock Selection *on page 187**).*

Operating Modes

The Z8 Encore! XP F082A Series products support four oscillator modes:

- Minimum power for use with very low frequency crystals (32 kHz–1 MHz).
- Medium power for use with medium frequency crystals or ceramic resonators (0.5 MHz to 8 MHz).
- Maximum power for use with high frequency crystals (8 MHz to 20 MHz).
- On-chip oscillator configured for use with external RC networks (<4 MHz).

The oscillator mode is selected using user-programmable Flash Option Bits. See Flash Option Bits on page 153 for information.

Crystal Oscillator Operation

The Flash Option bit XTLDIS controls whether the crystal oscillator is enabled during reset. The crystal may later be disabled after reset if a new oscillator has been selected as the system clock. If the crystal is manually enabled after reset through the OSCCTL register, the user code must wait at least 1000 crystal oscillator cycles for the crystal to stabilize. After this, the crystal oscillator may be selected as the system clock.

Note: *The stabilization time varies depending on the crystal or resonator used, as well as on the feedback network. See* Table 111 *for transconductance values to compute oscillator stabilization times.*

Internal Precision Oscillator

The internal precision oscillator (IPO) is designed for use without external components. You can either manually trim the oscillator for a non-standard frequency or use the automatic factory-trimmed version to achieve a 5.53 MHz frequency. IPO features include:

- On-chip RC oscillator that does not require external components
- Output frequency of either 5.53 MHz or 32.8 kHz (contains both a fast and a slow mode)
- Trimmed through Flash option bits with user override
- Elimination of crystals or ceramic resonators in applications where very high timing accuracy is not required

Operation

An 8-bit trimming register, incorporated into the design, compensates for absolute variation of oscillator frequency. Once trimmed the oscillator frequency is stable and does not require subsequent calibration. Trimming is performed during manufacturing and is not necessary for you to repeat unless a frequency other than 5.53 MHz (fast mode) or 32.8 kHz (slow mode) is required. This trimming is done at +30 ºC and a supply voltage of 3.3 V, so accuracy of this operating point is optimal.

If not used, the IPO can be disabled by the Oscillator Control register (see Oscillator Control Register Definitions on page 190).

By default, the oscillator frequency is set by the factory trim value stored in the write-protected Flash information page. However, the user code can override these trim values as described in Trim Bit Address Space on page 158.

Select one of two frequencies for the oscillator: 5.53 MHz and 32.8 kHz, using the OSCSEL bits in the Oscillator Control on page 187.

Table 123. Rotate and Shift Instructions (Continued)

Mnemonic	Operands	Instruction
SRA	dst	Shift Right Arithmetic
SRL	dst	Shift Right Logical
SWAP	dst	Swap Nibbles

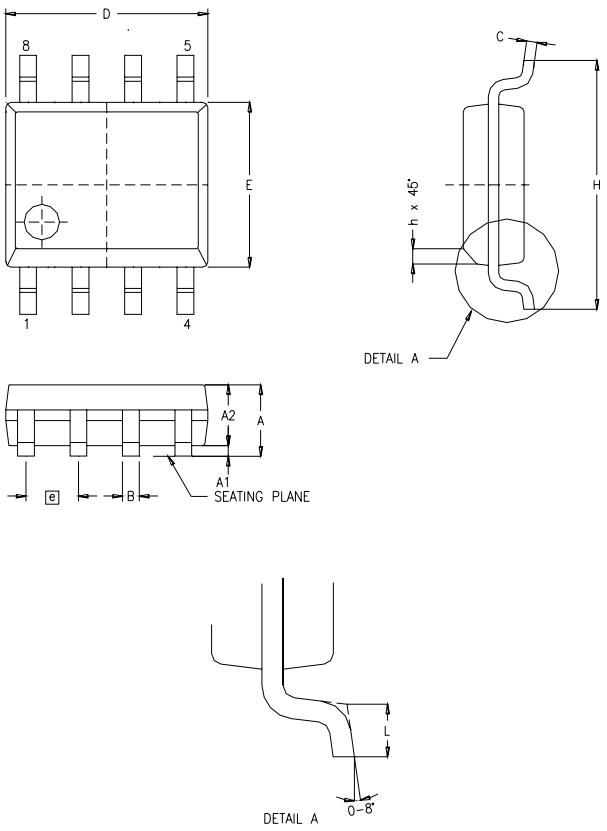
eZ8 CPU Instruction Summary

Table 124 summarizes the eZ8 CPU instructions. The table identifies the addressing modes employed by the instruction, the effect upon the Flags register, the number of CPU clock cycles required for the instruction fetch, and the number of CPU clock cycles required for the instruction execution.

Table 124. eZ8 CPU Instruction Summary

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
ADC dst, src	dst ← dst + src + C	r	r	12	*	*	*	*	0	*	2	3
		r	Ir	13							2	4
		R	R	14							3	3
		R	IR	15							3	4
		R	IM	16							3	3
		IR	IM	17							3	4
ADCX dst, src	dst ← dst + src + C	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							4	3
ADD dst, src	dst ← dst + src	r	r	02	*	*	*	*	0	*	2	3
		r	Ir	03							2	4
		R	R	04							3	3
		R	IR	05							3	4
		R	IM	06							3	3
		IR	IM	07							3	4
ADDX dst, src	dst ← dst + src	ER	ER	08	*	*	*	*	0	*	4	3
		ER	IM	09							4	3
Flags Notation:	* = Value is a function of the result of the operation. – = Unaffected X = Undefined				0 = Reset to 0 1 = Set to 1							

Figure 40 displays the 8-pin Small Outline Integrated Circuit package (SOIC) available for the Z8 Encore! XP® F082A Series devices.

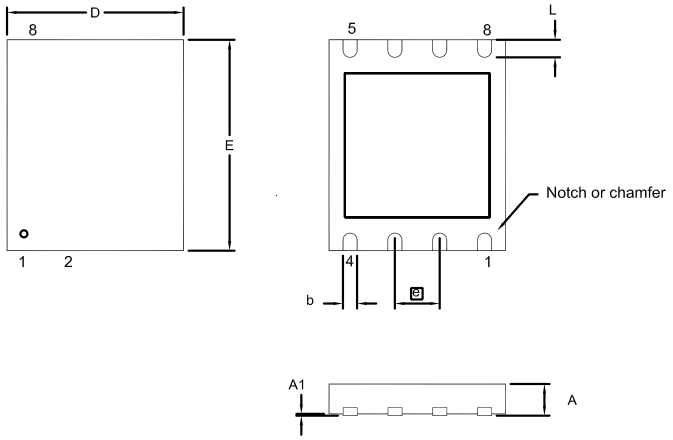


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	1.55	1.73	0.061	0.068
A1	0.10	0.25	0.004	0.010
A2	1.40	1.55	0.055	0.061
B	0.36	0.48	0.014	0.019
C	0.18	0.25	0.007	0.010
D	4.80	4.98	0.189	0.196
E	3.81	3.99	0.150	0.157
e	1.27 BSC		.050 BSC	
H	5.84	6.15	0.230	0.242
h	0.25	0.40	0.010	0.016
L	0.46	0.81	0.018	0.032

CONTROLLING DIMENSIONS : MM
LEADS ARE COPLANAR WITHIN .004 INCH.

Figure 40. 8-Pin Small Outline Integrated Circuit Package (SOIC)

Figure 41 displays the 8-pin Quad Flat No-Lead package (QFN)/MLF-S available for the Z8 Encore! XP F082A Series devices. This package has a footprint identical to that of the 8-pin SOIC, but with a lower profile.



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	0.80	1.00	0.032	0.039
A1	0	0.05	0	0.002
b	0.35	0.45	0.014	0.018
D	4.95	5.05	0.195	0.199
E	5.95	6.05	0.234	0.238
e	1.27 BSC		0.050 BSC	
L	0.50	0.70	0.020	0.028

1. CONTROLLING DIMENSIONS : mm
2. MAX. COPLANARITY : 0.08 mm / 0.003"

Figure 41. 8-Pin Quad Flat No-Lead Package (QFN)/MLF-S

G

H

I