# E·XFL



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

#### Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	5MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	16
Program Memory Size	1KB (1K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 7x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0123hh005sc

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

#### Z8 Encore! XP<sup>®</sup> F0823 Series Product Specification

Operation	152
OCD Interface	152
	153
OCD Data Format	154
	154
OCD Unlock Sequence (8-Pin Devices Only)	156
Breakpoints	156
Runtime Counter	156
On-Chip Debugger Commands	157
On-Chip Debugger Control Register Definitions	161
OCD Control Register	161
OCD Status Register	163
Oscillator Control	165
Operation	165
	165
	166
	167
	169
Operation	169
eZ8 CPU Instruction Set	171
Assembly Language Programming Introduction	171
Assembly Language Syntax	172
eZ8 CPU Instruction Notation	172
eZ8 CPU Instruction Classes	174
eZ8 CPU Instruction Summary	179
Opcode Maps	188
Electrical Characteristics	193
Absolute Maximum Ratings	193
DC Characteristics	194
AC Characteristics	197
On-Chip Peripheral AC and DC Electrical Characteristics	199
General Purpose I/O Port Input Data Sample Timing	202
	204
	205
Packaging	209

Table of Contents

al Conditions
dolov bogina ofter supply veltage execute

Table 10	Reset Sources	and Resulting	Reset Type
		and Resulting	

Operating Mode	Reset Source	Special Conditions
NORMAL or HALT modes	Power-On Reset/Voltage Brownout	Reset delay begins after supply voltage exceeds POR level.
	Watchdog Timer time-out when configured for Reset	None.
	RESET pin assertion	All reset pulses less than three system clocks in width are ignored.
	OCD initiated Reset (OCDCTL[0] set to 1)	System Reset, except the OCD is unaffected by the reset.
STOP mode	Power-On Reset/Voltage Brownout	Reset delay begins after supply voltage exceeds POR level.
	RESET pin assertion	All reset pulses less than the specified analog delay are ignored. See Electrical Characteristics on page 193.
	DBG pin driven Low	None.

#### **Power-On Reset**

Each device in the Z8 Encore! XP F0823 Series contains an internal POR circuit. The POR circuit monitors the supply voltage and holds the device in the Reset state until the supply voltage reaches a safe operating level. After the supply voltage exceeds the POR voltage threshold ( $V_{POR}$ ), the device is held in the Reset state until the POR Counter has timed out. If the crystal oscillator is enabled by the option bits, this time-out is longer.

After the Z8 Encore! XP F0823 Series device exits the POR state, the eZ8 CPU fetches the Reset vector. Following the POR, the POR status bit in Watchdog Timer Control (WDTCTL) register is set to 1.

Figure 5 displays POR operation. For the POR threshold voltage ( $V_{POR}$ ), see Electrical Characteristics on page 193.

#### STOP—Stop Mode Recovery Indicator

If this bit is set to 1, a Stop Mode Recovery is occurred. If the STOP and WDT bits are both set to 1, the Stop Mode Recovery occurred because of a WDT time-out. If the STOP bit is 1 and the WDT bit is 0, the Stop Mode Recovery was not caused by a WDT time-out. This bit is reset by a POR or a WDT time-out that occurred while not in STOP mode. Reading this register also resets this bit.

WDT-Watchdog Timer time-out Indicator

If this bit is set to 1, a WDT time-out occurred. A POR resets this pin. A Stop Mode Recovery from a change in an input pin also resets this bit. Reading this register resets this bit. This read must occur before clearing the WDT interrupt.

#### EXT-External Reset Indicator

If this bit is set to 1, a Reset initiated by the external  $\overline{\text{RESET}}$  pin occurred. A Power-On Reset or a Stop Mode Recovery from a change in an input pin resets this bit. Reading this register resets this bit.

Reserved-0 when read

tions as a GPIO pin. If it is not present, the debug feature is disabled until/unless another reset event occurs. For more details, see On-Chip Debugger on page 151.

# **Crystal Oscillator Override**

For systems using a crystal oscillator, PA0 and PA1 are used to connect the crystal. When the crystal oscillator is enabled (see Oscillator Control Register Definitions on page 167), the GPIO settings are overridden and PA0 and PA1 are disabled.

# **5 V Tolerance**

All six I/O pins on the 8-pin devices are 5 V-tolerant, unless the programmable pull-ups are enabled. If the pull-ups are enabled and inputs higher than  $V_{DD}$  are applied to these parts, excessive current flows through those pull-up devices and can damage the chip.

**Note:** In the 20- and 28-pin versions of this device, any pin which shares functionality with an ADC, crystal or comparator port is not 5 V-tolerant, including PA[1:0], PB[5:0], and PC[2:0]. All other signal pins are 5 V-tolerant, and can safely handle inputs higher than  $V_{DD}$  even with the pull-ups enabled.

# **External Clock Setup**

For systems using an external TTL drive, PB3 is the clock source for 20- and 28-pin devices. In this case, configure PB3 for alternate function CLKIN. Write the Oscillator Control Register (see Oscillator Control Register Definitions on page 167) such that the external oscillator is selected as the system clock. For 8-pin devices use PA1 instead of PB3.

50

0 PIN0 X R

BITS	7	6	5	4	3	2	1	
FIELD	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	
RESET	Х	Х	Х	Х	Х	Х	Х	
R/W	R	R	R	R	R	R	R	

# Table 28. Port A–C Input Data Registers (PxIN)

PIN[7:0]—Port Input Data

Sampled data from the corresponding port pin input.

0 = Input data is logical 0 (Low)

1 = Input data is logical 1 (High)

# Port A–C Output Data Register

The Port A–C Output Data register (Table 29) controls the output data to the pins.

FD2H, FD6H, FDAH

#### Table 29. Port A–C Output Data Register (PxOUT)

BITS	7	6	5	4	3	2	1	0
FIELD	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
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	FD3H, FD7H, FDBH							

#### POUT[7:0]—Port Output Data

These bits contain the data to be driven to the port pins. The values are only driven if the corresponding pin is configured as an output and the pin is not configured for alternate function operation.

0 =Drive a logical 0 (Low).

1 = Drive a logical 1 (High). High value is not driven if the drain has been disabled by setting the corresponding Port Output Control register bit to 1.

# LED Drive Enable Register

The LED Drive Enable register (Table 30) activates the controlled current drive. The Port C pin must first be enabled by setting the Alternate Function register to select the LED function.

ADDR

	Program Memory	
Priority	Vector Address	Interrupt or Trap Source
Highest	0002H	Reset (not an interrupt)
	0004H	Watchdog Timer (see Watchdog Timer on page 87)
	003AH	Primary Oscillator Fail Trap (not an interrupt)
	003CH	Watchdog Timer Oscillator Fail Trap (not an interrupt)
	0006H	Illegal Instruction Trap (not an interrupt)
	0008H	Reserved
	000AH	Timer 1
	000CH	Timer 0
	000EH	UART 0 receiver
	0010H	UART 0 transmitter
	0012H	Reserved
	0014H	Reserved
	0016H	ADC
	0018H	Port A Pin 7, selectable rising or falling input edge
	001AH	Port A Pin 6, selectable rising or falling input edge or Comparator Output
	001CH	Port A Pin 5, selectable rising or falling input edge
	001EH	Port A Pin 4, selectable rising or falling input edge
	0020H	Port A Pin 3 or Port D Pin 3, selectable rising or falling input edge
	0022H	Port A Pin 2 or Port D Pin 2, selectable rising or falling input edge
	0024H	Port A Pin 1, selectable rising or falling input edge
	0026H	Port A Pin 0, selectable rising or falling input edge
	0028H	Reserved
	002AH	Reserved
	002CH	Reserved
	002EH	Reserved
	0030H	Port C Pin 3, both input edges
	0032H	Port C Pin 2, both input edges
	0034H	Port C Pin 1, both input edges

#### Table 33. Trap and Interrupt Vectors in Order of Priority

# **Interrupt Control Register Definitions**

For all interrupts other than the Watchdog Timer interrupt, the Primary Oscillator Fail Trap, and the Watchdog Timer Oscillator Fail Trap, the interrupt control registers enable individual interrupts, set interrupt priorities, and indicate interrupt requests.

#### **Interrupt Request 0 Register**

The Interrupt Request 0 (IRQ0) register (Table 34) stores the interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ0 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 reads the Interrupt Request 0 register to determine if any interrupt requests are pending.

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved	T1I	ТОІ	U0RXI	U0TXI	Reserved	Reserved	ADCI
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	FC0H							

Table 34. Interrupt Request 0 Register (IRQ0)

Reserved—Must be 0

T1I—Timer 1 Interrupt Request

- 0 = No interrupt request is pending for Timer 1
- 1 = An interrupt request from Timer 1 is awaiting service

T0I—Timer 0 Interrupt Request

- 0 = No interrupt request is pending for Timer 0
- 1 = An interrupt request from Timer 0 is awaiting service

U0RXI-UART 0 Receiver Interrupt Request

- 0 = No interrupt request is pending for the UART 0 receiver
- 1 = An interrupt request from the UART 0 receiver is awaiting service

U0TXI-UART 0 Transmitter Interrupt Request

- 0 = No interrupt request is pending for the UART 0 transmitter
- 1 = An interrupt request from the UART 0 transmitter is awaiting service

ADCI—ADC Interrupt Request

- 0 = No interrupt request is pending for the ADC
- 1 = An interrupt request from the ADC is awaiting service

63

#### Table 43. IRQ2 Enable and Priority Encoding (Continued)

IRQ2ENH[x]	IRQ2ENL[	c] Priority	Description
1	1	Level 3	High

where x indicates the register bits from 0–7.

#### Table 44. IRQ2 Enable High Bit Register (IRQ2ENH)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved				C3ENH	C2ENH	C1ENH	C0ENH
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	FC7H							

Reserved—Must be 0

C3ENH—Port C3 Interrupt Request Enable High Bit C2ENH—Port C2 Interrupt Request Enable High Bit C1ENH—Port C1 Interrupt Request Enable High Bit C0ENH—Port C0 Interrupt Request Enable High Bit

#### Table 45. IRQ2 Enable Low Bit Register (IRQ2ENL)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved				C3ENL	C2ENL	C1ENL	C0ENL
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	FC8H							

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 46) determines whether an interrupt is generated for the rising edge or falling edge on the selected GPIO Port A or Port D input pin.

# Z8 Encore! XP<sup>®</sup> F0823 Series Product Specification

6. Write to the Timer Control register to enable the timer and initiate counting.

In ONE-SHOT mode, the system clock always provides the timer input. The timer period is given by the following equation:

 $ONE-SHOT Mode Time-Out Period (s) = \frac{(Reload Value - Start Value) \times Prescale}{System Clock Frequency (Hz)}$ 

#### **CONTINUOUS Mode**

In CONTINUOUS mode, the timer counts up to the 16-bit Reload value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the Reload value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) at timer Reload.

Follow the steps below to configure a timer for CONTINUOUS mode and to initiate the count:

- 1. Write to the Timer Control register to:
  - Disable the timer
  - Configure the timer for CONTINUOUS mode
  - Set the prescale value
  - If using the Timer Output alternate function, set the initial output level (High or Low)
- 2. Write to the Timer High and Low Byte registers to set the starting count value (usually 0001H). This action only affects the first pass in CONTINUOUS mode. After the first timer Reload in CONTINUOUS mode, counting always begins at the reset value of 0001H.
- 3. Write to the Timer Reload High and Low Byte registers to set the Reload value.
- 4. Enable the timer interrupt (if appropriate) and set the timer interrupt priority by writing to the relevant interrupt registers.
- 5. Configure the associated GPIO port pin (if using the Timer Output function) for the Timer Output alternate function.
- 6. Write to the Timer Control register to enable the timer and initiate counting.

In CONTINUOUS mode, the system clock always provides the timer input. The timer period is given by the following equation:

CONTINUOUS Mode Time-Out Period (s) =  $\frac{\text{Reload Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$ 

If an initial starting value other than 0001H is loaded into the Timer High and Low Byte registers, use the ONE-SHOT mode equation to determine the first time-out period.

#### **COMPARATOR COUNTER Mode**

In COMPARATOR COUNTER mode, the timer counts input transitions from the analog comparator output. The TPOL bit in the Timer Control Register selects whether the count occurs on the rising edge or the falling edge of the comparator output signal. In COMPARATOR COUNTER mode, the prescaler is disabled.

**Caution:** *The frequency of the comparator output signal must not exceed one-fourth the system clock frequency.* 

After reaching the Reload value stored in the Timer Reload High and Low Byte registers, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) at timer Reload.

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

- 1. Write to the Timer Control register to:
  - Disable the timer.
  - Configure the timer for COMPARATOR COUNTER mode.
  - Select either the rising edge or falling edge of the comparator output signal for the count. This also sets the initial logic level (High or Low) for the Timer Output alternate function. However, the Timer Output function is not required to be enabled.
- 2. Write to the Timer High and Low Byte registers to set the starting count value. This action only affects the first pass in COMPARATOR COUNTER mode. After the first timer Reload in COMPARATOR COUNTER mode, counting always begins at the reset value of 0001H. Generally, in COMPARATOR COUNTER mode the Timer High and Low Byte registers must be written with the value 0001H.
- 3. Write to the Timer Reload High and Low Byte registers to set the Reload value.
- 4. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
- 5. If using the Timer Output function, configure the associated GPIO port pin for the Timer Output alternate function.
- 6. Write to the Timer Control register to enable the timer.

In COMPARATOR COUNTER mode, the number of comparator output transitions since the timer start is given by the following equation:

Comparator Output Transitions = Current Count Value – Start Value

timer value is not reset to 0001H). Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) upon Compare.

If the Timer reaches FFFFH, the timer rolls over to 0000H and continue counting. Follow the steps below to configure a timer for COMPARE mode and to initiate the count:

- 1. Write to the Timer Control register to:
  - Disable the timer.
  - Configure the timer for Compare mode.
  - Set the prescale value.
  - Set the initial logic level (High or Low) for the Timer Output alternate function, if appropriate.
- 2. Write to the Timer High and Low Byte registers to set the starting count value.
- 3. Write to the Timer Reload High and Low Byte registers to set the Compare value.
- 4. Enable the timer interrupt, if appropriate, and set the timer interrupt priority by writing to the relevant interrupt registers.
- 5. If using the Timer Output function, configure the associated GPIO port pin for the Timer Output alternate function.
- 6. Write to the Timer Control register to enable the timer and initiate counting.

In COMPARE mode, the system clock always provides the timer input. The Compare time can be calculated by the following equation:

COMPARE Mode Time (s) =  $\frac{(Compare Value - Start Value) \times Prescale}{System Clock Frequency (Hz)}$ 

#### GATED Mode

In GATED mode, the timer counts only when the Timer Input signal is in its active state (asserted), as determined by the TPOL bit in the Timer Control register. When the Timer Input signal is asserted, counting begins. A timer interrupt is generated when the Timer Input signal is deasserted or a timer Reload occurs. To determine if a Timer Input signal deassertion generated the interrupt, read the associated GPIO input value and compare to the value stored in the TPOL bit.

The timer counts up to the 16-bit Reload value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. When reaching the Reload value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes (assuming the Timer Input signal remains asserted). Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) at timer reset.

### **Receiving Data using the Interrupt-Driven Method**

The UART Receiver interrupt indicates the availability of new data (as well as error conditions). Follow the steps below to configure the UART receiver for interrupt-driven operation:

- 1. Write to the UART Baud Rate High and Low Byte registers to set the acceptable baud rate.
- 2. Enable the UART pin functions by configuring the associated GPIO port pins for alternate function operation.
- 3. Execute a DI instruction to disable interrupts.
- 4. Write to the Interrupt control registers to enable the UART Receiver interrupt and set the acceptable priority.
- 5. Clear the UART Receiver interrupt in the applicable Interrupt Request register.
- 6. Write to the UART Control 1 Register to enable Multiprocessor (9-bit) mode functions, if appropriate.
  - Set the Multiprocessor Mode Select (MPEN) to Enable MULTIPROCESSOR mode
  - Set the Multiprocessor Mode Bits, MPMD[1:0], to select the acceptable address matching scheme
  - Configure the UART to interrupt on received data and errors or errors only (interrupt on errors only is unlikely to be useful for Z8 Encore! XP devices without a DMA block)
- 7. Write the device address to the Address Compare Register (automatic MULTIPROCESSOR modes only).
- 8. Write to the UART Control 0 register to:
  - Set the receive enable bit (REN) to enable the UART for data reception
  - Enable parity, if appropriate and if multiprocessor mode is not enabled, and select either even or odd parity
- 9. Execute an EI instruction to enable interrupts.

The UART is now configured for interrupt-driven data reception. When the UART Receiver interrupt is detected, the associated interrupt service routine (ISR) performs the following:

- 1. Checks the UART Status 0 register to determine the source of the interrupt error, break, or received data.
- 2. Reads the data from the UART Receive Data register if the interrupt was because of data available. If operating in MULTIPROCESSOR (9-bit) mode, further actions may be required depending on the MULTIPROCESSOR mode bits MPMD[1:0].

#### Z8 Encore! XP<sup>®</sup> F0823 Series Product Specification



No

If the Baud Rate Generator (BRG) interrupt enable is set, the UART Receiver interrupt asserts when the UART Baud Rate Generator reloads. This condition allows the Baud Rate Generator to function as an additional counter if the UART functionality is not employed.

Receiver Ready

Receiver Interrupt

Read Status

Errors?

#### **UART Baud Rate Generator**

The UART Baud Rate Generator creates a lower frequency baud rate clock for data transmission. The input to the Baud Rate Generator is the system clock. The UART Baud Rate High and Low Byte registers combine to create a 16-bit baud rate divisor value



# **Caution:** Unintentional accesses to the oscillator control register can actually stop the chip by switching to a non-functioning oscillator. To prevent this condition, the oscillator control block employs a register unlocking/locking scheme.

#### **OSC Control Register Unlocking/Locking**

To write the oscillator control register, unlock it by making two writes to the OSCCTL register with the values E7H followed by 18H. A third write to the OSCCTL register changes the value of the actual register and returns the register to a locked state. Any other sequence of oscillator control register writes has no effect. The values written to unlock the register must be ordered correctly, but are not necessarily consecutive. It is possible to write to or read from other registers within the unlocking/locking operation.

When selecting a new clock source, the primary oscillator failure detection circuitry and the Watchdog Timer oscillator failure circuitry must be disabled. If POFEN and WOFEN are not disabled prior to a clock switch-over, it is possible to generate an interrupt for a failure of either oscillator. The Failure detection circuitry can be enabled anytime after a successful write of OSCSEL in the oscillator control register.

The internal precision oscillator is enabled by default. If the user code changes to a different oscillator, it is appropriate to disable the IPO for power savings. Disabling the IPO does not occur automatically.

#### **Clock Failure Detection and Recovery**

#### **Primary Oscillator Failure**

Z8 Encore! XP<sup>®</sup> F0823 Series devices can generate non-maskable interrupt-like events when the primary oscillator fails. To maintain system function in this situation, the clock failure recovery circuitry automatically forces the Watchdog Timer oscillator to drive the system clock. The Watchdog Timer oscillator must be enabled to allow the recovery. Although this oscillator runs at a much slower speed than the original system clock, the CPU continues to operate, allowing execution of a clock failure vector and software routines that either remedy the oscillator failure or issue a failure alert. This automatic switchover is not available if the Watchdog Timer is the primary oscillator. It is also unavailable if the Watchdog Timer oscillator is disabled, though it is not necessary to enable the Watchdog Timer reset function outlined in the Watchdog Timer on page 87.

The primary oscillator failure detection circuitry asserts if the system clock frequency drops below 1 kHz  $\pm$ 50%. If an external signal is selected as the system oscillator, it is possible that a very slow but non-failing clock can generate a failure condition. Under these conditions, do not enable the clock failure circuitry (POFEN must be deasserted in the OSCCTL register).

# **Opcode Maps**

A description of the opcode map data and the abbreviations are provided in Figure 26. Figure 27 and Figure 28 provide information about each of the eZ8 CPU instructions. Table 116 lists Opcode Map abbreviations.



Figure 26. Opcode Map Cell Description

# General Purpose I/O Port Output Timing

Figure 30 and Table 128 provide timing information for GPIO Port pins.



### Figure 30. GPIO Port Output Timing

		Delay (ns)		
Parameter	Abbreviation	Minimum	Maximum	
GPIO Port p	ins			
T <sub>1</sub>	XIN Rise to Port Output Valid Delay	_	15	
T <sub>2</sub>	XIN Rise to Port Output Hold Time	2	_	

#### Table 128. GPIO Port Output Timing



Figure 38 displays the 20-pin Small Outline Integrated Circuit Package (SOIC) available for Z8 Encore! XP F0823 Series devices.

Figure 38. 20-Pin Small Outline Integrated Circuit Package (SOIC)

Figure 39 displays the 20-pin Small Shrink Outline Package (SSOP) available for Z8 Encore! XP F0823 Series devices.



Figure 39. 20-Pin Small Shrink Outline Package (SSOP)

Number	٩	5	ines	rrupts	lit Timers WM	sit A/D Channels	tT with IrDA	cription			
Part	Flas	RAN	101	Intel	16-E w/P\	10-E	UAF	Des			
Z8 Encore! XP with 8 KB Flash											
Standard Temperature: 0 °C to 70 °C											
Z8F0813PB005SC	8 KB	1 KB	6	12	2	0	1	PDIP 8-pin package			
Z8F0813QB005SC	8 KB	1 KB	6	12	2	0	1	QFN 8-pin package			
Z8F0813SB005SC	8 KB	1 KB	6	12	2	0	1	SOIC 8-pin package			
Z8F0813SH005SC	8 KB	1 KB	16	18	2	0	1	SOIC 20-pin package			
Z8F0813HH005SC	8 KB	1 KB	16	18	2	0	1	SSOP 20-pin package			
Z8F0813PH005SC	8 KB	1 KB	16	18	2	0	1	PDIP 20-pin package			
Z8F0813SJ005SC	8 KB	1 KB	24	18	2	0	1	SOIC 28-pin package			
Z8F0813HJ005SC	8 KB	1 KB	24	18	2	0	1	SSOP 28-pin package			
Z8F0813PJ005SC	8 KB	1 KB	24	18	2	0	1	PDIP 28-pin package			
Extended Temperature: -40 °C to 105 °C											
Z8F0813PB005EC	8 KB	1 KB	6	12	2	0	1	PDIP 8-pin package			
Z8F0813QB005EC	8 KB	1 KB	6	12	2	0	1	QFN 8-pin package			
Z8F0813SB005EC	8 KB	1 KB	6	12	2	0	1	SOIC 8-pin package			
Z8F0813SH005EC	8 KB	1 KB	16	18	2	0	1	SOIC 20-pin package			
Z8F0813HH005EC	8 KB	1 KB	16	18	2	0	1	SSOP 20-pin package			
Z8F0813PH005EC	8 KB	1 KB	16	18	2	0	1	PDIP 20-pin package			
Z8F0813SJ005EC	8 KB	1 KB	24	18	2	0	1	SOIC 28-pin package			
Z8F0813HJ005EC	8 KB	1 KB	24	18	2	0	1	SSOP 28-pin package			
Z8F0813PJ005EC	8 KB	1 KB	24	18	2	0	1	PDIP 28-pin package			
Replace C with G for Lead-Free Packaging											

218

#### Z8 Encore! XP<sup>®</sup> F0823 Series Product Specification

# Index

# Symbols

# 174 % 174 @ 174

# **Numerics**

10-bit ADC 4 40-lead plastic dual-inline package 214, 215

# Α

absolute maximum ratings 193 AC characteristics 197 ADC 175 architecture 117 automatic power-down 118 block diagram 118 continuous conversion 120 control register 122, 124 control register definitions 122 data high byte register 124 data low bits register 125 electrical characteristics and timing 201 operation 118 single-shot conversion 119 ADCCTL register 122, 124 ADCDH register 124 ADCDL register 125 ADCX 175 ADD 175 add - extended addressing 175 add with carry 175 add with carry - extended addressing 175 additional symbols 174 address space 13 ADDX 175 analog signals 10 analog-to-digital converter (ADC) 117 AND 177

ANDX 177 arithmetic instructions 175 assembly language programming 171 assembly language syntax 172

# В

B 174 b 173 baud rate generator, UART 103 **BCLR 176** binary number suffix 174 **BIT 176** bit 173 clear 176 manipulation instructions 176 set 176 set or clear 176 swap 176 test and jump 178 test and jump if non-zero 178 test and jump if zero 178 bit jump and test if non-zero 178 bit swap 178 block diagram 3 block transfer instructions 176 **BRK 178 BSET 176** BSWAP 176, 178 **BTJ 178 BTJNZ 178 BTJZ 178** 

# С

CALL procedure 178 CAPTURE mode 84, 85 CAPTURE/COMPARE mode 85 cc 173 CCF 176 characteristics, electrical 193 clear 177 CLR 177 COM 177 227