



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	6
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	8-VDFN Exposed Pad
Supplier Device Package	8-QFN (5x6)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0823qb005sc

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

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

Ordering Information	217
Part Number Suffix Designations	226
Index	227
Customer Support	237

## **Internal Precision Oscillator**

The internal precision oscillator (IPO) is a trimmable clock source that requires no external components.

## 10-Bit Analog-to-Digital Converter

The optional analog-to-digital converter (ADC) converts an analog input signal to a 10-bit binary number. The ADC accepts inputs from eight different analog input pins in both single-ended and differential modes.

## **Analog Comparator**

The analog comparator compares the signal at an input pin with either an internal programmable voltage reference or a second input pin. The comparator output can be used to drive either an output pin or to generate an interrupt.

## **Universal Asynchronous Receiver/Transmitter**

The UART is full-duplex and capable of handling asynchronous data transfers. The UART supports 8- and 9-bit data modes and selectable parity. The UART also supports multi-drop address processing in hardware. The UART baud rate generator can be configured and used as a basic 16-bit timer.

## Timers

Two enhanced 16-bit reloadable timers can be used for timing/counting events or for motor control operations. These timers provide a 16-bit programmable reload counter and operate in ONE-SHOT, CONTINUOUS, GATED, CAPTURE, CAPTURE RESTART, COMPARE, CAPTURE AND COMPARE, PWM SINGLE OUTPUT, and PWM DUAL OUTPUT modes.

## Interrupt Controller

Z8 Encore! XP<sup>®</sup> F0823 Series products support up to 20 interrupts. These interrupts consist of eight internal peripheral interrupts and 12 general-purpose I/O pin interrupt sources. The interrupts have three levels of programmable interrupt priority.

memory addresses returns FFH. Writing to these unimplemented Program Memory addresses produces no effect. Table 6 describes the Program Memory maps for the Z8 Encore! XP<sup>®</sup> F0823 Series products.

Program Memory Address (Hex)	Function
Z8F0823 and Z8F0813 Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–003D	Oscillator Fail Traps*
003E-0FFF	Program Memory
Z8F0423 and Z8F0413 Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–003D	Oscillator Fail Traps*
003E-0FFF	Program Memory
Z8F0223 and Z8F0213 Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–003D	Oscillator Fail Traps*
003E-07FF	Program Memory
Z8F0123 and Z8F0113 Products	
0000–0001	Flash Option Bits

 Table 6. Z8 Encore! XP F0823 Series Program Memory Maps

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

	Reset Characteristics and Latency								
Reset Type	Control Registers	eZ8 CPU	Reset Latency (Delay)						
System Reset	Reset (as applicable)	Reset	66 Internal Precision Oscillator Cycles						
Stop Mode Recovery	Unaffected, except WDT_CTL and OSC_CTL registers	Reset	66 Internal Precision Oscillator Cycles + IPO startup time						

#### Table 9. Reset and Stop Mode Recovery Characteristics and Latency

During a System Reset or Stop Mode Recovery, the IPO requires 4 µs to start up. Then the Z8 Encore! XP F0823 Series device is held in Reset for 66 cycles of the Internal Precision Oscillator. If the crystal oscillator is enabled in the Flash option bits, this reset period is increased to 5000 IPO cycles. When a reset occurs because of a low voltage condition or Power-On Reset, this delay is measured from the time that the supply voltage first exceeds the POR level. If the external pin reset remains asserted at the end of the reset period, the device remains in reset until the pin is deasserted.

At the beginning of Reset, all GPIO pins are configured as inputs with pull-up resistor disabled.

During Reset, the eZ8 CPU and on-chip peripherals are idle; however, the on-chip crystal oscillator and Watchdog Timer oscillator continue to run.

Upon Reset, control registers within the Register File that have a defined Reset value are loaded with their reset values. Other control registers (including the Stack Pointer, Register Pointer, and Flags) and general-purpose RAM are undefined following Reset. The eZ8 CPU fetches the Reset vector at Program Memory addresses 0002H and 0003H and loads that value into the Program Counter. Program execution begins at the Reset vector address.

When the control registers are re-initialized by a system reset, the system clock after reset is always the IPO. The software must reconfigure the oscillator control block, such that the correct system clock source is enabled and selected.

## **Reset Sources**

Table 10 lists the possible sources of a System Reset.

0 = The drains are enabled for any output mode (unless overridden by the alternate function).

1 = The drain of the associated pin is disabled (open-drain mode).

#### Port A–C High Drive Enable Sub-Registers

The Port A–C High Drive Enable sub-register (Table 23) is accessed through the Port A–C Control register by writing 04H to the Port A–C Address register. Setting the bits in the Port A–C High Drive Enable sub-registers to 1 configures the specified port pins for high current output drive operation. The Port A–C High Drive Enable sub-register affects the pins directly and, as a result, alternate functions are also affected.

Table 23. Port A–C High Drive Enable Sub-Registers (PxHDE)

BITS	7	6	5	4	3	2	1	0			
FIELD	PHDE7	PHDE6	PHDE5	PHDE4	PHDE3	PHDE2	PHDE1	PHDE0			
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	lf 04H i	If 04H in Port A–C Address Register, accessible through the Port A–C Control Register									

PHDE[7:0]—Port High Drive Enabled.

0 = The Port pin is configured for standard output current drive.

1 = The Port pin is configured for high output current drive.

#### Port A–C Stop Mode Recovery Source Enable Sub-Registers

The Port A–C Stop Mode Recovery Source Enable sub-register (Table 24) is accessed through the Port A–C Control register by writing 05H to the Port A–C Address register. Setting the bits in the Port A–C Stop Mode Recovery Source Enable sub-registers to 1 configures the specified Port pins as a Stop Mode Recovery source. During STOP mode, any logic transition on a Port pin enabled as a Stop Mode Recovery source initiates Stop Mode Recovery.

Table 24. Port A–C Stop	Mode Recovery	/ Source Enable Sub-I	Registers	(PxSMRE)
-------------------------	---------------	-----------------------	-----------	----------

BITS	7	6	5	4	3	2	1	0			
FIELD	PSMRE7	PSMRE6	PSMRE5	PSMRE4	PSMRE3	PSMRE2	PSMRE1	PSMRE0			
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	lf 05H i	If 05H in Port A–C Address Register, accessible through the Port A–C Control Register									

Writing a 1 to the IRQE bit in the Interrupt Control register

Interrupts are globally disabled by any of the following actions:

- Execution of a Disable Interrupt (DI) instruction
- eZ8 CPU acknowledgement of an interrupt service request from the interrupt controller
- Writing a 0 to the IRQE bit in the Interrupt Control register
- Reset
- Execution of a Trap instruction
- Illegal Instruction Trap
- Primary Oscillator Fail Trap
- Watchdog Timer Oscillator Fail Trap

#### **Interrupt Vectors and Priority**

The interrupt controller supports three levels of interrupt priority. Level 3 is the highest priority, Level 2 is the second highest priority, and Level 1 is the lowest priority. If all interrupts are enabled with identical interrupt priority (for example, all as Level 2 interrupts), the interrupt priority is assigned from highest to lowest as specified in Table 33 on page 54. Level 3 interrupts are always assigned higher priority than Level 2 interrupts which, in turn, always are assigned higher priority than Level 1 interrupts. Within each interrupt priority level (Level 1, Level 2, or Level 3), priority is assigned as specified in Table 33. Reset, Watchdog Timer interrupt (if enabled), Primary Oscillator Fail Trap, Watchdog Timer Oscillator Fail Trap, and Illegal Instruction Trap always have highest (Level 3) priority.

#### Interrupt Assertion

Interrupt sources assert their interrupt requests for only a single system clock period (single pulse). When the interrupt request is acknowledged by the eZ8 CPU, the corresponding bit in the Interrupt Request register is cleared until the next interrupt occurs. Writing a 0 to the corresponding bit in the Interrupt Request register likewise clears the interrupt request.

**Caution:** The following coding style that clears bits in the Interrupt Request registers is not recommended. All incoming interrupts received between execution of the first LDX command and the final LDX command are lost.

Poor coding style that can result in lost interrupt requests: LDX r0, IRQ0 AND r0, MASK LDX IRQ0, r0

#### **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

of the Timer Input signal. When the Capture event occurs, an interrupt is generated and the timer continues counting. The INPCAP bit in TxCTL1 register is set to indicate the timer interrupt is because of an input capture event.

The timer continues counting up to the 16-bit Reload value stored in the Timer Reload High and Low Byte registers. Upon reaching the Reload value, the timer generates an interrupt and continues counting. The INPCAP bit in TxCTL1 register clears indicating the timer interrupt is not because of an input capture event.

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 or a Reload event. 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 TxCTL1 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:

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

#### **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 TxCTL1 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 0001H and counting resumes. The INPCAP bit in TxCTL1 register is cleared to indicate the timer interrupt is not caused by an input capture event.

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

- 1. Write to the Timer Control register to:
  - Disable the timer.
  - Configure the timer for CAPTURE RESTART mode. Setting the mode also involves writing to TMODEHI bit in TxCTL1 register.
  - 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. This allows the software to determine if interrupts were generated by either a Capture or a Reload event. 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 TxCTL1 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:

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

#### COMPARE Mode

In COMPARE mode, the timer counts up to the 16-bit maximum Compare value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the Compare value, the timer generates an interrupt and counting continues (the

#### **CAPTURE/COMPARE Mode**

0 = Counting is started on the first rising edge of the Timer Input signal. The current count is captured on subsequent rising edges of the Timer Input signal.

1 = Counting is started on the first falling edge of the Timer Input signal. The current count is captured on subsequent falling edges of the Timer Input signal.

#### **PWM DUAL OUTPUT Mode**

0 = Timer Output is forced Low (0) and Timer Output Complement is forced High (1) when the timer is disabled. When enabled, the Timer Output is forced High (1) upon PWM count match and forced Low (0) upon Reload. When enabled, the Timer Output Complement is forced Low (0) upon PWM count match and forced High (1) upon Reload. The PWMD field in TxCTL0 register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to High (1).

1 = Timer Output is forced High (1) and Timer Output Complement is forced Low (0) when the timer is disabled. When enabled, the Timer Output is forced Low (0) upon PWM count match and forced High (1) upon Reload. When enabled, the Timer Output Complement is forced High (1) upon PWM count match and forced Low (0) upon Reload. The PWMD field in TxCTL0 register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to Low (0).

#### **CAPTURE RESTART Mode**

0 = Count is captured on the rising edge of the Timer Input signal

1 = Count is captured on the falling edge of the Timer Input signal

#### **COMPARATOR COUNTER Mode**

When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer Reload.

**Caution:** When the Timer Output alternate function TxOUT on a GPIO port pin is enabled, Tx-OUT changes to whatever state the TPOL bit is in. The timer does not need to be enabled for that to happen. Also, the port data direction sub register is not needed to be set to output on TxOUT. Changing the TPOL bit with the timer enabled and running does not immediately change the TxOUT.

#### PRES—Prescale value.

The timer input clock is divided by  $2^{PRES}$ , where PRES can be set from 0 to 7. The prescaler is reset each time the Timer is disabled. This reset ensures proper clock division each time the Timer is restarted.

000 = Divide by 1001 = Divide by 2

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

#### 118



Figure 19. Analog-to-Digital Converter Block Diagram

## Operation

## **Data Format**

The output of the ADC is an 11-bit, signed, two's complement digital value. The output generally ranges from 0 to +1023, but offset errors can cause small negative values.

The ADC registers return 13 bits of data, but the two LSBs are intended for compensation use only. When the compensation routine is performed on the 13 bit raw ADC value, two

## Operation

The Flash Controller programs and erases Flash memory. The Flash Controller provides the proper Flash controls and timing for Byte Programming, Page Erase, and Mass Erase of Flash memory.

The Flash Controller contains several protection mechanisms to prevent accidental programming or erasure. These mechanism operate on the page, sector and full-memory levels.

The Flowchart in Figure 21 displays basic Flash Controller operation. The following subsections provide details about the various operations (Lock, Unlock, Byte Programming, Page Protect, Page Unprotect, Page Select Page Erase, and Mass Erase) displayed in Figure 21.

## **Flash Sector Protect Register**

The Flash Sector Protect (FPROT) register is shared with the Flash Page Select Register. When the Flash Control Register is written with 73H followed by 5EH, the next write to this address targets the Flash Sector Protect Register. In all other cases, it targets the Flash Page Select Register.

This register selects one of the 8 available Flash memory sectors to be protected. The reset state of each Sector Protect bit is an unprotected state. After a sector is protected by setting its corresponding register bit, it cannot be unprotected (the register bit cannot be cleared) without powering down the device.

BITS	7	6	5	4	3	2	1	0		
FIELD	SPROT7	SPROT6	SPROT5	SPROT4	SPROT3	SPROT2	SPROT1	SPROT0		
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		FF9H								

Table 82. Flash Sector Protect Register (FPROT)

SPROT7-SPROT0—Sector Protection

Each bit corresponds to a 512 bytes Flash sector. For the Z8F08x3 devices, the upper 3 bits must be zero. For the Z8F04x3 devices all bits are used. For the Z8F02x3 devices, the upper 4 bits are unused. For the Z8F01x3 devices, the upper 6 bits are unused.

## Flash Frequency High and Low Byte Registers

The Flash Frequency High (FFREQH) and Low Byte (FFREQL) registers combine to form a 16-bit value, FFREQ, to control timing for Flash program and erase operations. The 16-bit binary Flash Frequency value must contain the system clock frequency (in kHz) and is calculated using the following equation:

```
FFREQ[15:0] = {FFREQH[7:0],FFREQL[7:0]} = System Clock Frequency
1000
```

**Caution:** The Flash Frequency High and Low Byte registers must be loaded with the correct value to ensure proper operation of the device. Also, Flash programming and erasure is not supported for system clock frequencies below 20 kHz or above 20 MHz.

# **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).

#### Watchdog Timer Failure

In the event of a Watchdog Timer oscillator failure, a similar non-maskable interrupt-like event is issued. This event does not trigger an attendant clock switch-over, but alerts the CPU of the failure. After a Watchdog Timer failure, it is no longer possible to detect a primary oscillator failure. The failure detection circuitry does not function if the Watchdog Timer is used as the primary oscillator or if the Watchdog Timer oscillator has been disabled. For either of these cases, it is necessary to disable the detection circuitry by deasserting the WDFEN bit of the OSCCTL register.

The Watchdog Timer oscillator failure detection circuit counts system clocks while searching for a Watchdog Timer clock. The logic counts 8004 system clock cycles before determining that a failure has occurred. The system clock rate determines the speed at which the Watchdog Timer failure can be detected. A very slow system clock results in very slow detection times.

**Caution:** It is possible to disable the clock failure detection circuitry as well as all functioning clock sources. In this case, the Z8 Encore! XP F0823 Series device ceases functioning and can only be recovered by Power-On Reset.

## **Oscillator Control Register Definitions**

The following section provides the bit definitions for the Oscillator Control register.

#### **Oscillator Control Register**

The Oscillator Control register (OSCCTL) enables/disables the various oscillator circuits, enables/disables the failure detection/recovery circuitry and selects the primary oscillator, which becomes the system clock.

The Oscillator Control register must be unlocked before writing. Writing the two step sequence E7H followed by 18H to the Oscillator Control Register unlocks it. The register is locked at successful completion of a register write to the OSCCTL.

BITS	7	6	5	4	3	2	1	0		
FIELD	INTEN	Reserved	WDTEN	POFEN	WDFEN		SCKSEL			
RESET	1	0	1	0	0	0	0	0		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		
ADDR		F86H								

Table 102. Oscillator Control Register (OSCCTL)

200

V <sub>DD</sub> = 2.7 V to 3.6 V T <sub>A</sub> = -40 °C to +105 °C (unless otherwise stated)						
Parameter	Minimum Typical Maximum		Maximum	Units	Notes	
Flash Byte Read Time	100	-	-	ns		
Flash Byte Program Time	20	-	40	μs		
Flash Page Erase Time	10	-	-	ms		
Flash Mass Erase Time	200	-	-	ms		
Writes to Single Address Before Next Erase	-	-	2			
Flash Row Program Time	_	-	8	ms	Cumulative program time for single row cannot exceed limit before next erase. This parameter is only an issue when bypassing the Flash Controller.	
Data Retention	100	-	-	years	25 °C	
Endurance	10,000	-	-	cycles	Program/erase cycles	

#### Table 123. Flash Memory Electrical Characteristics and Timing

#### Table 124. Watchdog Timer Electrical Characteristics and Timing

		V <sub>DD</sub> T <sub>A</sub> = (unless	$V_{DD}$ = 2.7 V to 3.6 V T <sub>A</sub> = -40 °C to +105 °C (unless otherwise stated)			
Symbol	Parameter	Minimum	Typical	Maximum	Units	Conditions
F <sub>WDT</sub>	WDT Oscillator Frequency		10		kHz	
F <sub>WDT</sub>	WDT Oscillator Error			<u>+</u> 50	%	
T <sub>WDTCAL</sub>	WDT Calibrated Timeout	0.98	1	1.02	S	V <sub>DD</sub> = 3.3 V; T <sub>A</sub> = 30 °C
		0.70	1	1.30	S	V <sub>DD</sub> = 2.7 V to 3.6 V T <sub>A</sub> = 0 °C to 70 °C
		0.50	1	1.50	S	V <sub>DD</sub> = 2.7 V to 3.6 V T <sub>A</sub> = -40 °C to +105 °C

#### 209

## Packaging

Figure 34 displays the 8-pin Plastic Dual Inline Package (PDIP) available for the Z8 Encore!  $XP^{\textcircled{R}}$  F0823 Series devices.



Figure 34. 8-Pin Plastic Dual Inline Package (PDIP)

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Description			
Z8 Encore! XP with 4	KB Flash	, 10-Bit /	Analog	g-to-D	igital C	onve	erter				
Standard Temperature: 0 °C to 70 °C											
Z8F0423PB005SC	4 KB	1 KB	6	12	2	4	1	PDIP 8-pin package			
Z8F0423QB005SC	4 KB	1 KB	6	12	2	4	1	QFN 8-pin package			
Z8F0423SB005SC	4 KB	1 KB	6	12	2	4	1	SOIC 8-pin package			
Z8F0423SH005SC	4 KB	1 KB	16	18	2	7	1	SOIC 20-pin package			
Z8F0423HH005SC	4 KB	1 KB	16	18	2	7	1	SSOP 20-pin package			
Z8F0423PH005SC	4 KB	1 KB	16	18	2	7	1	PDIP 20-pin package			
Z8F0423SJ005SC	4 KB	1 KB	22	18	2	8	1	SOIC 28-pin package			
Z8F0423HJ005SC	4 KB	1 KB	22	18	2	8	1	SSOP 28-pin package			
Z8F0423PJ005SC	4 KB	1 KB	22	18	2	8	1	PDIP 28-pin package			
Extended Temperatu	re: -40 °C	to 105 °C	C								
Z8F0423PB005EC	4 KB	1 KB	6	12	2	4	1	PDIP 8-pin package			
Z8F0423QB005EC	4 KB	1 KB	6	12	2	4	1	QFN 8-pin package			
Z8F0423SB005EC	4 KB	1 KB	6	12	2	4	1	SOIC 8-pin package			
Z8F0423SH005EC	4 KB	1 KB	16	18	2	7	1	SOIC 20-pin package			
Z8F0423HH005EC	4 KB	1 KB	16	18	2	7	1	SSOP 20-pin package			
Z8F0423PH005EC	4 KB	1 KB	16	18	2	7	1	PDIP 20-pin package			
Z8F0423SJ005EC	4 KB	1 KB	22	18	2	8	1	SOIC 28-pin package			
Z8F0423HJ005EC	4 KB	1 KB	22	18	2	8	1	SSOP 28-pin package			
Z8F0423PJ005EC	4 KB	1 KB	22	18	2	8	1	PDIP 28-pin package			
Replace C with G for Lea	d-Free Pac	kaging									

219

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

timing 205 OCD commands execute instruction (12H) 161 read data memory (0DH) 160 read OCD control register (05H) 158 read OCD revision (00H) 158 read OCD status register (02H) 158 read program counter (07H) 159 read program memory (0BH) 160 read program memory CRC (0EH) 161 read register (09H) 159 read runtime counter (03H) 158 step instruction (10H) 161 stuff instruction (11H) 161 write data memory (0CH) 160 write OCD control register (04H) 158 write program counter (06H) 159 write program memory (0AH) 159 write register (08H) 159 on-chip debugger (OCD) 151 on-chip debugger signals 10 ONE-SHOT mode 84 opcode map abbreviations 189 cell description 188 first 190 second after 1FH 191 Operational Description 21, 31, 35, 53, 67, 87, 93, 113, 117, 127, 129, 141, 151, 165, 169 OR 177 ordering information 217 **ORX 178** 

status register 163

## Ρ

p 173 packaging 20-pin PDIP 211, 212 20-pin SSOP 212, 215 28-pin PDIP 213 28-pin SOIC 214 8-pin PDIP 209 8-pin SOIC 210

PDIP 214, 215 part selection guide 2 PC 174 PDIP 214, 215 peripheral AC and DC electrical characteristics 199 pin characteristics 10 Pin Descriptions 7 polarity 173 POP 177 pop using extended addressing 177 **POPX 177** port availability, device 35 port input timing (GPIO) 203 port output timing, GPIO 204 power supply signals 10 power-down, automatic (ADC) 118 Power-on and Voltage Brownout electrical characteristics and timing 199 Power-On Reset (POR) 23 program control instructions 178 program counter 174 program memory 13 **PUSH 177** push using extended addressing 177 PUSHX 177 PWM mode 84, 85 PxADDR register 44

## R

PxCTL register 45

R 173 r 173 RA register address 173 RCF 176 receive IrDA data 115 receiving UART data-interrupt-driven method 98 receiving UART data-polled method 97 register 173 ADC control (ADCCTL) 122, 124 232