

Welcome to [E-XFL.COM](https://www.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	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	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 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	8-SOIC (0.154", 3.90mm Width)
Supplier Device Package	-
Purchase URL	<a href="https://www.e-xfl.com/product-detail/zilog/z8f0213sb005eg">https://www.e-xfl.com/product-detail/zilog/z8f0213sb005eg</a>

MULTIPROCESSOR (9-Bit) Mode .....	103
External Driver Enable .....	105
UART Interrupts .....	105
UART Baud Rate Generator .....	108
UART Control Register Definitions .....	108
UART Transmit Data Register .....	109
UART Receive Data Register .....	109
UART Status 0 Register .....	110
UART Status 1 Register .....	111
UART Control 0 and Control 1 Registers .....	112
UART Address Compare Register .....	115
UART Baud Rate High and Low Byte Registers .....	115
Infrared Encoder/Decoder .....	117
Architecture .....	117
Operation .....	117
Transmitting IrDA Data .....	118
Receiving IrDA Data .....	119
Infrared Encoder/Decoder Control Register Definitions .....	120
Analog-to-Digital Converter .....	121
Architecture .....	121
Operation .....	122
Automatic Powerdown .....	123
Single-Shot Conversion .....	123
Continuous Conversion .....	124
Interrupts .....	125
Calibration and Compensation .....	125
ADC Control Register Definitions .....	126
ADC Control Register 0 .....	126
ADC Control/Status Register 1 .....	129
ADC Data High Byte Register .....	130
ADC Data Low Bits Register .....	131
Comparator .....	132
Operation .....	132
Comparator Control Register Definition .....	133
Flash Memory .....	134
Flash Information Area .....	135
Operation .....	135
Flash Operation Timing Using the Flash Frequency Registers .....	137
Flash Code Protection Against External Access .....	137

## **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® 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.

## **Reset Controller**

Z8 Encore! XP® F0823 Series products can be reset using the  $\overline{\text{RESET}}$  pin, POR, WDT time-out, STOP Mode exit, or Voltage Brown-Out warning signal. The  $\overline{\text{RESET}}$  pin is bidirectional, that is, it functions as reset source as well as a reset indicator.

## Pin Characteristics

Table 4 provides detailed information about the characteristics for each pin available on Z8 Encore! XP F0823 Series 20- and 28-pin devices. Data in Table 4 is sorted alphabetically by the pin symbol mnemonic.

► **Note:** All six I/O pins on the 8-pin packages are 5 V-tolerant (unless the pull-up devices are enabled). The right-most column in Table 4 describes 5 V tolerance for the 20- and 28-pin packages only.

**Table 4. Pin Characteristics (20- and 28-pin Devices)\***

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tristate Output	Internal Pull-up or Pull- down	Schmitt- Trigger Input	Open Drain Output	5V Tolerance
AVDD	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
AVSS	N/A	N/A	N/A	N/A	N/A	N/A	N/A	NA
DBG	I/O	I	N/A	Yes	No	Yes	Yes	Yes
PA[7:0]	I/O	I	N/A	Yes	Program- mable Pull-up	Yes	Yes, Programmable	PA[7:2] only
PB[7:0]	I/O	I	N/A	Yes	Program- mable Pull-up	Yes	Yes, Programmable	PB[7:6] only
PC[7:0]	I/O	I	N/A	Yes	Program- mable Pull-up	Yes	Yes, Programmable	PC[7:3] only
RESET	I/O	I/O (defaults to RESET)	Low (in Reset mode)	Yes (PD0 only)	Always on for RESET	Yes	Always on for RESET	Yes
VDD	N/A	N/A	N/A	N/A			N/A	N/A
VSS	N/A	N/A	N/A	N/A			N/A	N/A

Note: PB6 and PB7 are available only in the devices without ADC.

Bit	Description (Continued)
[4] EXT	<b>External Reset Indicator</b> If this bit is set to 1, a Reset initiated by the external <u>RESET</u> 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. For POR/Stop Mode Recover event values, please see Table 13.
[3:0]	<b>Reserved</b> These bits are reserved and must be programmed to 0000 when read.

**Table 13. POR Indicator Values**

Reset or Stop Mode Recovery Event	POR	STOP	WDT	EXT
Power-On Reset	1	0	0	0
Reset using <u>RESET</u> pin assertion	0	0	0	1
Reset using WDT time-out	0	0	1	0
Reset using the OCD (OCTCTL[1] set to 1)	1	0	0	0
Reset from STOP Mode using DBG Pin driven Low	1	0	0	0
Stop Mode Recovery using GPIO pin transition	0	1	0	0
Stop Mode Recovery using WDT time-out	0	1	1	0

► **Note:** This register is only reset during a Power-On Reset sequence. Other System Reset events do not affect it.

**Table 14. Power Control Register 0 (PWRCTL0)**

Bit	7	6	5	4	3	2	1	0
Field	Reserved	Reserved		VBO	Reserved	ADC	COMP	Reserved
RESET	1	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
Address	F80H							

Bit	Description
[7]	<b>Reserved</b> This bit is reserved and must be programmed to 1.
[6:5]	<b>Reserved</b> These bits are reserved and must be programmed to 00.
[4] VBO	<b>Voltage Brown-Out Detector Disable</b> This bit and the VBO_AO Flash option bit must both enable the VBO for the VBO to be active. 0 = VBO enabled. 1 = VBO disabled.
[3]	<b>Reserved</b> This bit is reserved and must be programmed to 0.
[2] ADC	<b>Analog-to-Digital Converter Disable</b> 0 = Analog-to-Digital Converter enabled. 1 = Analog-to-Digital Converter disabled.
[1] COMP	<b>Comparator Disable</b> 0 = Comparator is enabled. 1 = Comparator is disabled.
[0]	<b>Reserved</b> This bit is reserved and must be programmed to 0.

# General-Purpose Input/Output

Z8 Encore! XP F0823 Series products support a maximum of 24 port pins (Ports A–C) for general-purpose input/output (GPIO) operations. Each port contains control and data registers. The GPIO control registers determine data direction, open-drain, output drive current, programmable pull-ups, Stop Mode Recovery functionality, and alternate pin functions. Each port pin is individually programmable. In addition, the Port C pins are capable of direct LED drive at programmable drive strengths.

## GPIO Port Availability By Device

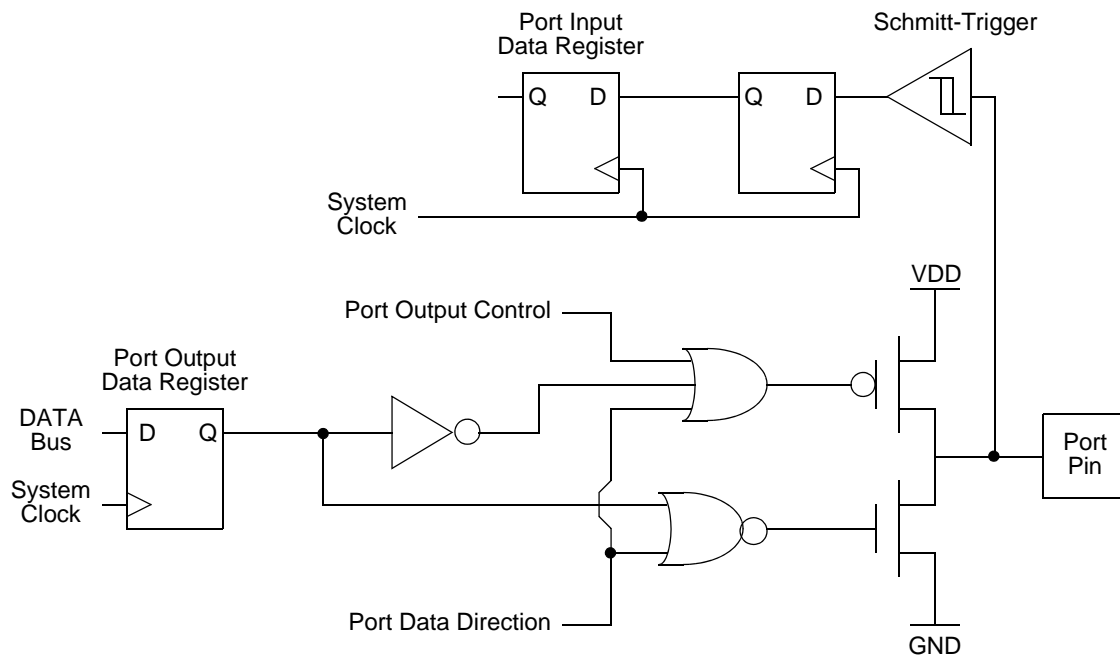
Table 15 lists the port pins available with each device and package type.

**Table 15. Port Availability by Device and Package Type**

Devices	Package	10-Bit ADC	Port A	Port B	Port C	Total I/O
Z8F0823SB, Z8F0823PB Z8F0423SB, Z8F0423PB Z8F0223SB, Z8F0223PB Z8F0123SB, Z8F0123PB	8-pin	Yes	[5:0]	No	No	6
Z8F0813SB, Z8F0813PB Z8F0413SB, Z8F0413PB Z8F0213SB, Z8F0213PB Z8F0113SB, Z8F011vPB	8-pin	No	[5:0]	No	No	6
Z8F0823PH, Z8F0823HH Z8F0423PH, Z8F0423HH Z8F0223PH, Z8F0223HH Z8F0123PH, Z8F0123HH	20-pin	Yes	[7:0]	[3:0]	[3:0]	16
Z8F0813PH, Z8F0813HH Z8F0413PH, Z8F0413HH Z8F0213PH, Z8F0213HH Z8F0113PH, Z8F0113HH	20-pin	No	[7:0]	[3:0]	[3:0]	16
Z8F0823PJ, Z8F0823SJ Z8F0423PJ, Z8F0423SJ Z8F0223PJ, Z8F0223SJ Z8F0123PJ, Z8F0123SJ	28-pin	Yes	[7:0]	[5:0]	[7:0]	22
Z8F0813PJ, Z8F0813SJ Z8F0413PJ, Z8F0413SJ Z8F0213PJ, Z8F0213SJ Z8F0113PJ, Z8F0113SJ	28-pin	No	[7:0]	[7:0]	[7:0]	24

## Architecture

Figure 7 displays a simplified block diagram of a GPIO port pin. In this figure, the ability to accommodate alternate functions and variable port current drive strength is not displayed.



**Figure 7. GPIO Port Pin Block Diagram**

## GPIO Alternate Functions

Many of the GPIO port pins are used for general-purpose I/O and access to on-chip peripheral functions such as the timers and serial communication devices. The port A–D Alternate Function subregisters configure these pins for either GPIO or alternate function operation. When a pin is configured for alternate function, control of the port pin direction (input/output) is passed from the Port A–D Data Direction registers to the alternate function assigned to this pin. Tables 16 and 17 list the alternate functions possible with each port pin for 8-pin and non-8-pin parts, respectively. The alternate function associated at a pin is defined through Alternate Function Sets subregisters AFS1 and AFS2.

The crystal oscillator functionality is not controlled by the GPIO block. When the crystal oscillator is enabled in the oscillator control block, the GPIO functionality of PA0 and PA1 is overridden. In that case, those pins function as input and output for the crystal oscillator.

**Example 1.** A poor coding style that can result in lost interrupt requests:

```
LDX r0, IRQ0
AND r0, MASK
LDX IRQ0, r0
```

To avoid missing interrupts, use the coding style in Example 2 to clear bits in the Interrupt Request 0 Register:

**Example 2.** A good coding style that avoids lost interrupt requests:

```
ANDX IRQ0, MASK
```

## Software Interrupt Assertion

Program code generates interrupts directly. Writing a 1 to the correct bit in the Interrupt Request register triggers an interrupt (assuming that interrupt is enabled). When the interrupt request is acknowledged by the eZ8 CPU, the bit in the Interrupt Request register is automatically cleared to 0.

---

**! Caution:** Zilog recommends not using a coding style to generate software interrupts by setting bits in the Interrupt Request registers. All incoming interrupts received between execution of the first LDX command and the final LDX command are lost. See Example 3, which follows.

---

**Example 3.** A poor coding style that can result in lost interrupt requests:

```
LDX r0, IRQ0
OR r0, MASK
LDX IRQ0, r0
```

To avoid missing interrupts, use the coding style in Example 4 to set bits in the Interrupt Request registers:

**Example 4.** A good coding style that avoids lost interrupt requests:

```
ORX IRQ0, MASK
```

## Watchdog Timer Interrupt Assertion

The Watchdog Timer interrupt behavior is different from interrupts generated by other sources. The Watchdog Timer continues to assert an interrupt as long as the timeout condition continues. As it operates on a different (and usually slower) clock domain than the rest of the device, the Watchdog Timer continues to assert this interrupt for many system clocks until the counter rolls over.

Bit	Description (Continued)
[4] U0RXI	<b>UART 0 Receiver Interrupt Request</b> 0 = No interrupt request is pending for the UART 0 receiver. 1 = An interrupt request from the UART 0 receiver is awaiting service.
[3] U0TXI	<b>UART 0 Transmitter Interrupt Request</b> 0 = No interrupt request is pending for the UART 0 transmitter. 1 = An interrupt request from the UART 0 transmitter is awaiting service.
[2:1]	<b>Reserved</b> These bits are reserved and must be programmed to 00.
[0] ADC1	<b>ADC Interrupt Request</b> 0 = No interrupt request is pending for the ADC. 1 = An interrupt request from the ADC is awaiting service.

## Interrupt Request 1 Register

The Interrupt Request 1 (IRQ1) register (Table 37) stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ1 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 1 Register to determine if any interrupt requests are pending.

**Table 37. Interrupt Request 1 Register (IRQ1)**

Bit	7	6	5	4	3	2	1	0
Field	PA7VI	PA6CI	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
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
Address	FC3H							

Bit	Description
[7] PA7VI	<b>Port A7 Interrupt Request</b> 0 = No interrupt request is pending for GPIO Port A. 1 = An interrupt request from GPIO Port A.
[6] PA6CI	<b>Port A6 or Comparator Interrupt Request</b> 0 = No interrupt request is pending for GPIO Port A or Comparator. 1 = An interrupt request from GPIO Port A or Comparator.
[5:0] PAxI	<b>Port A Pin x Interrupt Request</b> 0 = No interrupt request is pending for GPIO Port A pin x. 1 = An interrupt request from GPIO Port A pin x is awaiting service.

Note: x indicates the specific GPIO Port pin number (0–5).

## Interrupt Control Register

The Interrupt Control (IRQCTL) Register (Table 50) contains the master enable bit for all interrupts.

**Table 50. Interrupt Control Register (IRQCTL)**

Bit	7	6	5	4	3	2	1	0
Field	IRQE	Reserved						
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R
Address	FCFH							

Bit	Description
[7] IRQE	<p><b>Interrupt Request Enable</b></p> <p>This bit is set to 1 by executing an Enable Interrupts (EI) or Interrupt Return (IRET) instruction, or by a direct register write of a 1 to this bit. It is reset to 0 by executing a DI instruction, eZ8 CPU acknowledgement of an interrupt request, reset or by a direct register write of a 0 to this bit.</p> <p>0 = Interrupts are disabled. 1 = Interrupts are enabled.</p>
[6:0]	<p><b>Reserved</b></p> <p>These bits are reserved and must be programmed to 0000000 when read.</p>

Observe the following steps to configure a timer for PWM DUAL OUTPUT Mode and initiating the PWM operation:

1. Write to the Timer Control Register to:
  - Disable the timer
  - Configure the timer for PWM DUAL OUTPUT Mode. Setting the mode also involves writing to the TMODEHI bit in the TxCTL1 Register
  - Set the prescale value
  - Set the initial logic level (High or Low) and PWM High/Low transition for the Timer Output alternate function
2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H); this write only affects the first pass in PWM Mode. After the first timer reset in PWM Mode, counting always begins at the reset value of 0001H.
3. Write to the PWM High and Low Byte registers to set the PWM value.
4. Write to the PWM Control Register to set the PWM dead band delay value. The dead-band delay must be less than the duration of the positive phase of the PWM signal (as defined by the PWM high and low byte registers). It must also be less than the duration of the negative phase of the PWM signal (as defined by the difference between the PWM registers and the Timer Reload registers).
5. Write to the Timer Reload High and Low Byte registers to set the reload value (PWM period). The reload value must be greater than the PWM value.
6. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
7. Configure the associated GPIO port pin for the Timer Output and Timer Output Complement alternate functions. The Timer Output Complement function is shared with the Timer Input function for both timers. Setting the timer mode to Dual PWM automatically switches the function from Timer In to Timer Out Complement.
8. Write to the Timer Control Register to enable the timer and initiate counting.

The PWM period is represented by the following equation:

$$\text{PWM Period (s)} = \frac{\text{Reload Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

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

If TPOL is set to 0, the ratio of the PWM output High time to the total period is represented by:

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:

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

## **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 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.

Observe the following steps to configure 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.

5. Configure the associated GPIO port pin for the Timer Input alternate function.
6. Write to the Timer Control Register to enable the timer.
7. Assert the Timer Input signal to initiate the counting.

### **CAPTURE/COMPARE Mode**

In CAPTURE/COMPARE Mode, the timer begins counting on the first external Timer Input transition. The acceptable transition (rising edge or falling edge) is set by the TPOL bit in the Timer Control Register. The timer input is the system clock.

Every subsequent acceptable transition (after the first) of the Timer Input signal captures the current count value. The capture value is written to the Timer PWM High and Low Byte registers. When the capture event occurs, an interrupt is generated, 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 caused by 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 compare 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 because of an input capture event.

Observe the following steps to configure a timer for CAPTURE/COMPARE Mode and initiating the count:

1. Write to the Timer Control Register to:
  - Disable the timer
  - Configure the timer for CAPTURE/COMPARE 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 compare value.
4. Enable the timer interrupt, if appropriate, and set the timer interrupt priority by writing to the relevant interrupt registers. By default, the timer interrupt are 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.
5. Configure the associated GPIO port pin for the Timer Input alternate function.
6. Write to the Timer Control Register to enable the timer.

# Watchdog Timer

The Watchdog Timer (WDT) protects against corrupt or unreliable software, power faults, and other system-level problems which can place Z8 Encore! XP F0823 Series devices into unsuitable operating states. The features of Watchdog Timer include:

- On-chip RC oscillator
- A selectable time-out response: reset or interrupt
- 24-bit programmable time-out value

## Operation

The WDT is a retriggerable one-shot timer that resets or interrupts F0823 Series devices when the WDT reaches its terminal count. The Watchdog Timer uses a dedicated on-chip RC oscillator as its clock source. The Watchdog Timer operates in only two modes: ON and OFF. Once enabled, it always counts and must be refreshed to prevent a time-out. Perform an enable by executing the WDT instruction or by setting the WDT\_AO Flash Option Bit. The WDT\_AO bit forces the Watchdog Timer to operate immediately upon reset, even if a WDT instruction has not been executed.

The Watchdog Timer is a 24-bit reloadable down counter that uses three 8-bit registers in the eZ8 CPU register space to set the reload value. The nominal WDT time-out period is described by the following equation:

$$\text{WDT Time-out Period (ms)} = \frac{\text{WDT Reload Value}}{10}$$

where the WDT reload value is the decimal value of the 24-bit value given by {WDTU[7:0], WDTM[7:0], WDTL[7:0]} and the typical Watchdog Timer RC oscillator frequency is 10kHz. The Watchdog Timer cannot be refreshed after it reaches 000002H. The WDT Reload Value must not be set to values below 000004H. Table 59 provides information about approximate time-out delays for the minimum and maximum WDT reload values.

**Table 59. Watchdog Timer Approximate Time-Out Delays**

WDT Reload Value (Hex)	WDT Reload Value (Decimal)	Approximate Time-Out Delay (with 10kHz typical WDT oscillator frequency)	
		Typical	Description
000004	4	400 μs	Minimum time-out delay
FFFFFF	16,777,215	28 minutes	Maximum time-out delay

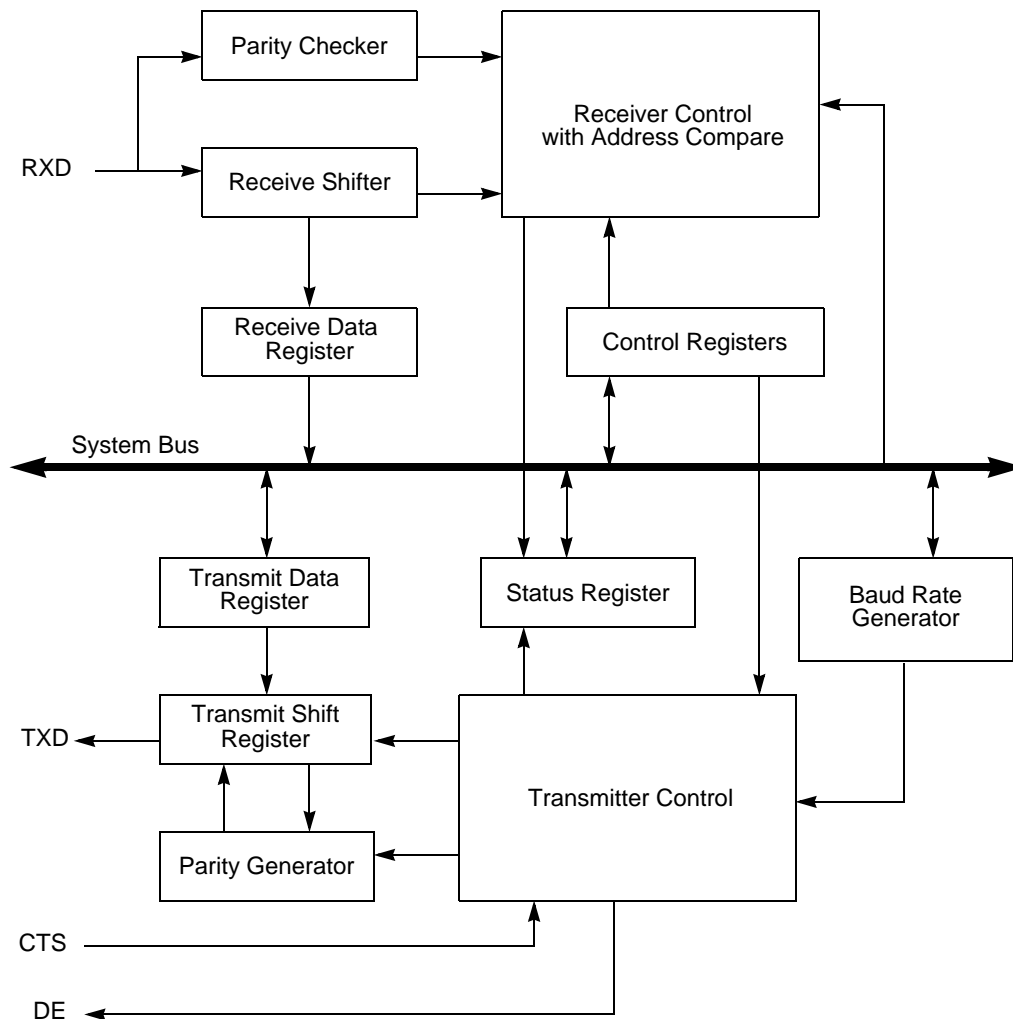


Figure 10. UART Block Diagram

## Operation

The UART always transmits and receives data in an 8-bit data format, least-significant bit (lsb) first. An even or odd parity bit can be added to the data stream. Each character begins with an active Low Start bit and ends with either 1 or 2 active High Stop bits. Figure 11 and Figure 12 display the asynchronous data format employed by the UART without parity and with parity, respectively.

7. Return to Step 4 to receive additional data.

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

The UART Receiver interrupt indicates the availability of new data (as well as error conditions). Observe the following steps 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:

# Oscillator Control

Z8 Encore! XP F0823 Series devices uses three possible clocking schemes, each user-selectable. These three schemes are:

- On-chip precision trimmed RC oscillator
- External clock drive
- On-chip low power Watchdog Timer oscillator

In addition, F0823 Series devices contain clock failure detection and recovery circuitry, which allow continued operation despite a failure of the primary oscillator.

## Operation

This chapter discusses the logic used to select the system clock and handle primary oscillator failures. A description of the specific operation of each oscillator is outlined elsewhere in this document.

## System Clock Selection

The oscillator control block selects from the available clocks. Table 104 details each clock source and its usage.

**Table 104. Oscillator Configuration and Selection**

Clock Source	Characteristics	Required Setup
Internal Precision RC Oscillator	<ul style="list-style-type: none"><li>• 32.8kHz or 5.53MHz</li><li>• <math>\pm 4\%</math> accuracy when trimmed</li><li>• No external components required</li></ul>	<ul style="list-style-type: none"><li>• Unlock and write Oscillator Control Register (OSCCTL) to enable and select oscillator at either 5.53MHz or 32.8kHz</li></ul>
External Clock Drive	<ul style="list-style-type: none"><li>• 0 to 20MHz</li><li>• Accuracy dependent on external clock source</li></ul>	<ul style="list-style-type: none"><li>• Write GPIO registers to configure PB3 pin for external clock function</li><li>• Unlock and write OSCCTL to select external system clock</li><li>• Apply external clock signal to GPIO</li></ul>
Internal Watchdog Timer Oscillator	<ul style="list-style-type: none"><li>• 10kHz nominal</li><li>• <math>\pm 40\%</math> accuracy; no external components required</li><li>• Very Low power consumption</li></ul>	<ul style="list-style-type: none"><li>• Enable WDT if not enabled and wait until WDT Oscillator is operating.</li><li>• Unlock and write Oscillator Control Register (OSCCTL) to enable and select oscillator</li></ul>

## DC Characteristics

Table 121 lists the DC characteristics of the Z8 Encore! XP F0823 Series products. All voltages are referenced to  $V_{SS}$ , the primary system ground.

**Table 121. DC Characteristics**

$T_A = -40^{\circ}\text{C to } +105^{\circ}\text{C}$ (unless otherwise specified)						
Symbol	Parameter	Minimum	Typical	Maximum	Units	Conditions
$V_{DD}$	Supply Voltage	2.7	–	3.6	V	
$V_{IL1}$	Low Level Input Voltage	–0.3	–	$0.3 \cdot V_{DD}$	V	
$V_{IH1}$	High Level Input Voltage	$0.7 \cdot V_{DD}$	–	5.5	V	For all input pins without analog or oscillator function. For all signal pins on the 8-pin devices. Programmable pull-ups must also be disabled.
$V_{IH2}$	High Level Input Voltage	$0.7 \cdot V_{DD}$	–	$V_{DD} + 0.3$	V	For those pins with analog or oscillator function (20-/28-pin devices only), or when programmable pull-ups are enabled.
$V_{OL1}$	Low Level Output Voltage	–	–	0.4	V	$I_{OL} = 2\text{mA}$ ; $V_{DD} = 3.0\text{V}$ High Output Drive disabled.
$V_{OH1}$	High Level Output Voltage	2.4	–	–	V	$I_{OH} = -2\text{mA}$ ; $V_{DD} = 3.0\text{V}$ High Output Drive disabled.
$V_{OL2}$	Low Level Output Voltage	–	–	0.6	V	$I_{OL} = 20\text{mA}$ ; $V_{DD} = 3.3\text{V}$ High Output Drive enabled.
$V_{OH2}$	High Level Output Voltage	2.4	–	–	V	$I_{OH} = -20\text{mA}$ ; $V_{DD} = 3.3\text{V}$ High Output Drive enabled.
$I_{IH}$	Input Leakage Current	–	$\pm 0.002$	$\pm 5$	$\mu\text{A}$	$V_{IN} = V_{DD}$ $V_{DD} = 3.3\text{V}$
$I_{IL}$	Input Leakage Current	–	$\pm 0.007$	$\pm 5$	$\mu\text{A}$	$V_{IN} = V_{SS}$ $V_{DD} = 3.3\text{V}$
$I_{TL}$	Tristate Leakage Current	–	–	$\pm 5$	$\mu\text{A}$	

**Notes:**

1. This condition excludes all pins that have on-chip pull-ups, when driven Low.
2. These values are provided for design guidance only and are not tested in production.

## General Purpose I/O Port Output Timing

Figure 30 and Table 131 provide timing information for GPIO Port pins.

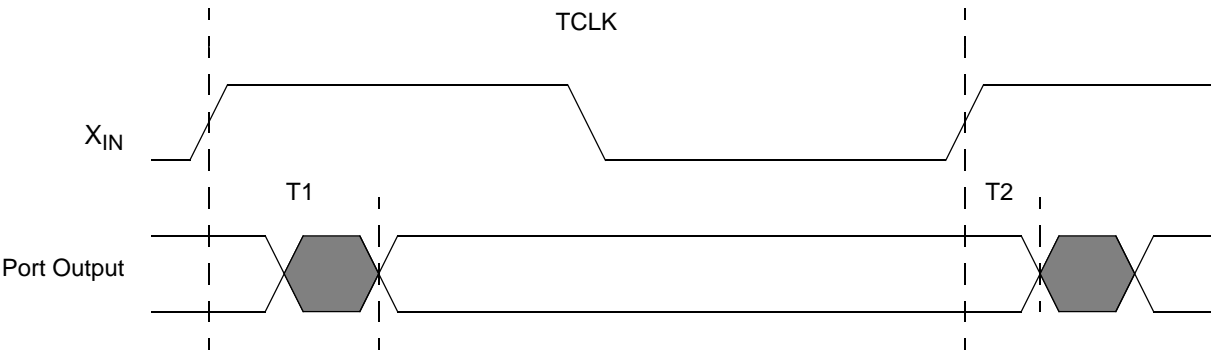


Figure 30. GPIO Port Output Timing

Table 131. GPIO Port Output Timing

Parameter	Abbreviation	Delay (ns)	
		Minimum	Maximum
GPIO Port pins			
T <sub>1</sub>	X <sub>IN</sub> Rise to Port Output Valid Delay	–	15
T <sub>2</sub>	X <sub>IN</sub> Rise to Port Output Hold Time	2	–

**Table 135. Z8 Encore! XP F0823 Series Ordering Matrix (Continued)**

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Description
<b>Z8 Encore! XP F0823 Series with 1 KB Flash</b>								
<b>Standard Temperature: 0°C to 70°C</b>								
Z8F0113PB005SG	1 KB	256 B	6	12	2	0	1	PDIP 8-pin package
Z8F0113QB005SG	1 KB	256 B	6	12	2	0	1	QFN 8-pin package
Z8F0113SB005SG	1 KB	256 B	6	12	2	0	1	SOIC 8-pin package
Z8F0113SH005SG	1 KB	256 B	16	18	2	0	1	SOIC 20-pin package
Z8F0113HH005SG	1 KB	256 B	16	18	2	0	1	SSOP 20-pin package
Z8F0113PH005SG	1 KB	256 B	16	18	2	0	1	PDIP 20-pin package
Z8F0113SJ005SG	1 KB	256 B	24	18	2	0	1	SOIC 28-pin package
Z8F0113HJ005SG	1 KB	256 B	24	18	2	0	1	SSOP 28-pin package
Z8F0113PJ005SG	1 KB	256 B	24	18	2	0	1	PDIP 28-pin package
<b>Extended Temperature: -40°C to 105°C</b>								
Z8F0113PB005EG	1 KB	256 B	6	12	2	0	1	PDIP 8-pin package
Z8F0113QB005EG	1 KB	256 B	6	12	2	0	1	QFN 8-pin package
Z8F0113SB005EG	1 KB	256 B	6	12	2	0	1	SOIC 8-pin package
Z8F0113SH005EG	1 KB	256 B	16	18	2	0	1	SOIC 20-pin package
Z8F0113HH005EG	1 KB	256 B	16	18	2	0	1	SSOP 20-pin package
Z8F0113PH005EG	1 KB	256 B	16	18	2	0	1	PDIP 20-pin package
Z8F0113SJ005EG	1 KB	256 B	24	18	2	0	1	SOIC 28-pin package
Z8F0113HJ005EG	1 KB	256 B	24	18	2	0	1	SSOP 28-pin package
Z8F0113PJ005EG	1 KB	256 B	24	18	2	0	1	PDIP 28-pin package