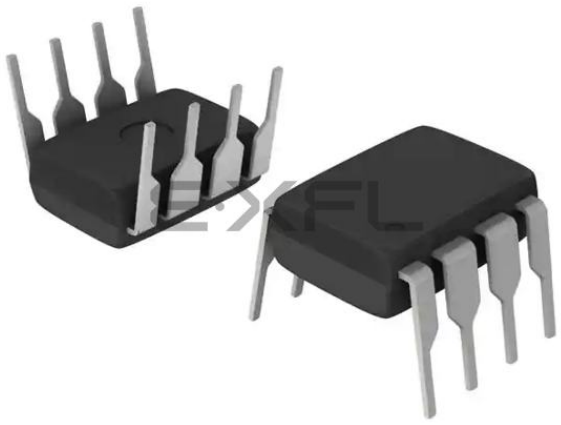

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	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	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0213pb005eg

On-Chip Debugger

F0823 Series products feature an integrated On-Chip Debugger. The OCD provides a rich-set of debugging capabilities, such as reading and writing registers, programming Flash memory, setting breakpoints and executing code. A single-pin interface provides communication to the OCD.

Reset Sources

Table 10 lists the possible sources of a System Reset.

Table 10. Reset Sources and Resulting Reset Type

Operating Mode	Reset Source	Special Conditions
NORMAL or HALT modes	Power-On Reset/Voltage Brown-Out.	Reset delay begins after supply voltage exceeds POR level.
	Watchdog Timer time-out when configured for Reset.	None.
	$\overline{\text{RESET}}$ 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 Brown-Out.	Reset delay begins after supply voltage exceeds POR level.
	$\overline{\text{RESET}}$ pin assertion.	All reset pulses less than the specified analog delay are ignored. See the Electrical Characteristics chapter on page 196.
	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 the Electrical Characteristics chapter on page 196.

Table 18. GPIO Port Registers and Subregisters (Continued)

Port Register Mnemonic	Port Register Name
P*x*HDE	High Drive Enable.
P*x*SMRE	Stop Mode Recovery Source Enable.
P*x*PUE	Pull-up Enable.
PxAFS1	Alternate Function Set 1.
PxAFS2	Alternate Function Set 2.

Port A–C Address Registers

The Port A–C Address registers select the GPIO port functionality accessible through the Port A–C Control registers. The Port A–C Address and Control registers combine to provide access to all GPIO port controls (Table 19).

Table 19. Port A–C GPIO Address Registers (P*x*ADDR)

Bit	7	6	5	4	3	2	1	0
Field	PADDR[7:0]							
RESET	00H							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	FD0H, FD4H, FD8H							

Bit	Description
[7:0] PADDR	**Port Address** The Port Address selects one of the subregisters accessible through the Port Control Register. See Table 20 for each subregister function.

Table 20. PADDR[7:0] Subregister Functions

PADDR[7:0]	Port Control Subregister Accessible Using the Port A–C Control Registers
00H	No function. Provides some protection against accidental Port reconfiguration.
01H	Data Direction.
02H	Alternate Function.
03H	Output Control (Open-Drain).
04H	High Drive Enable.

Table 43. IRQ1 Enable High Bit Register (IRQ1ENH)

Bit	7	6	5	4	3	2	1	0
Field	PA7VENH	PA6CENH	PA5ENH	PA4ENH	PA3ENH	PA2ENH	PA1ENH	PA0ENH
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	FC4H							

Bit	Description
[7] PA7VENH	**Port A Bit[7] Interrupt Request Enable High Bit**
[6] PA6CENH	**Port A Bit[7] or Comparator Interrupt Request Enable High Bit**
[5:0] PAxENH	**Port A Bit[*x*] Interrupt Request Enable High Bit** For selection of Port A as the interrupt source, see the Shared Interrupt Select Register section on page 67.

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

Table 44. IRQ1 Enable Low Bit Register (IRQ1ENL)

Bit	7	6	5	4	3	2	1	0
Field	PA7VENL	PA6CENL	PA5ENL	PA4ENL	PA3ENL	PA2ENL	PA1ENL	PA0ENL
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	FC5H							

Bit	Description
[7] PA7VENL	**Port A Bit[7] Interrupt Request Enable Low Bit**
[6] PA6CENL	**Port A Bit[7] or Comparator Interrupt Request Enable Low Bit**
[5:0] PAxENL	**Port A Bit[*x*] Interrupt Request Enable Low Bit**

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

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:

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

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 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. Observe the following steps 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.

7. Counting begins on the first appropriate transition of the Timer Input signal. No interrupt is generated by this first edge.

In CAPTURE/COMPARE Mode, the elapsed time from timer start to capture event can be calculated using the following equation:

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

Reading the Timer Count Values

The current count value in the timers can be read while counting (enabled). This capability has no effect on timer operation. When the timer is enabled and the Timer High Byte Register is read, the contents of the Timer Low Byte register are placed in a holding register. A subsequent read from the Timer Low Byte register returns the value in the holding register. This operation allows accurate reads of the full 16-bit timer count value while enabled. When the timers are not enabled, a read from the Timer Low Byte register returns the actual value in the counter.

Timer Pin Signal Operation

Timer Output is a GPIO port pin alternate function. The Timer Output is toggled every time the counter is reloaded.

The timer input can be used as a selectable counting source. It shares the same pin as the complementary timer output. When selected by the GPIO Alternate Function registers, this pin functions as a timer input in all modes except for the DUAL PWM OUTPUT mode. For this mode, there is no timer input available.

Timer Control Register Definitions

This section defines the features of the following Timer Control registers.

Timer 0–1 High and Low Byte Registers: see page 83

Timer Reload High and Low Byte Registers: see page 84

Timer 0–1 PWM High and Low Byte Registers: see page 86

Timer 0–1 Control Registers: see page 86

Timer 0–1 High and Low Byte Registers

The Timer 0–1 High and Low Byte (TxH and TxL) registers (Table 51 and Table 52) contain the current 16-bit timer count value. When the timer is enabled, a read from TxH

Watchdog Timer Refresh

When first enabled, the WDT is loaded with the value in the Watchdog Timer Reload registers. The Watchdog Timer counts down to 000000H unless a WDT instruction is executed by the eZ8 CPU. Execution of the WDT instruction causes the down counter to be reloaded with the WDT reload value stored in the Watchdog Timer Reload registers. Counting resumes following the reload operation.

When Z8 Encore! XP F0823 Series devices are operating in DEBUG Mode (using the OCD), the Watchdog Timer is continuously refreshed to prevent any Watchdog Timer time-outs.

Watchdog Timer Time-Out Response

The Watchdog Timer times out when the counter reaches 000000H. A time-out of the Watchdog Timer generates either an interrupt or a system reset. The WDT_RES Flash Option Bit determines the time-out response of the Watchdog Timer. For information about programming of the WDT_RES Flash Option Bit, see **the** Flash Option Bits chapter on page 146.

WDT Interrupt in Normal Operation

If configured to generate an interrupt when a time-out occurs, the Watchdog Timer issues an interrupt request to the interrupt controller and sets the WDT status bit in the Watchdog Timer Control Register. If interrupts are enabled, the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executing code from the vector address. After time-out and interrupt generation, the Watchdog Timer counter rolls over to its maximum value of FFFFFH and continues counting. The Watchdog Timer counter is not automatically returned to its Reload Value.

The Reset Status Register (see the Reset Status Register section on page 28) must be read before clearing the WDT interrupt. This read clears the WDT time-out Flag and prevents further WDT interrupts for immediately occurring.

WDT Interrupt in STOP Mode

If configured to generate an interrupt when a time-out occurs and F0823 Series are in STOP Mode, the Watchdog Timer automatically initiates a Stop Mode Recovery and generates an interrupt request. Both the WDT status bit and the STOP bit in the Watchdog Timer Control Register are set to 1 following a WDT time-out in STOP Mode. For more information about Stop Mode Recovery, see **the** Reset and Stop Mode Recovery chapter on page 21.

If interrupts are enabled, following completion of the Stop Mode Recovery the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executing code from the vector address.

– Set or clear the CTSE bit to enable or disable control from the remote receiver using the $\overline{\text{CTS}}$ pin

6. Check the TDRE bit in the UART Status 0 Register to determine if the Transmit Data Register is empty (indicated by a 1). If empty, continue to Step 7. If the Transmit Data Register is full (indicated by a 0), continue to monitor the TDRE bit until the Transmit Data Register becomes available to receive new data.
7. Write the UART Control 1 Register to select the outgoing address bit.
8. Set the Multiprocessor Bit Transmitter (MPBT) if sending an address byte, clear it if sending a data byte.
9. Write the data byte to the UART Transmit Data Register. The transmitter automatically transfers the data to the Transmit Shift register and transmits the data.
10. Make any changes to the Multiprocessor Bit Transmitter (MPBT) value, if appropriate and MULTIPROCESSOR Mode is enabled,.
11. To transmit additional bytes, return to Step 5.

Transmitting Data Using the Interrupt-Driven Method

The UART Transmitter interrupt indicates the availability of the Transmit Data Register to accept new data for transmission. Observe the following steps to configure the UART for interrupt-driven data transmission:

1. Write to the UART Baud Rate High and Low Byte registers to set the appropriate 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 Transmitter interrupt and set the acceptable priority.
5. Write to the UART Control 1 Register to enable MULTIPROCESSOR (9-bit) Mode functions, if MULTIPROCESSOR Mode is appropriate.
6. Set the MULTIPROCESSOR Mode Select (MPEN) to Enable MULTIPROCESSOR Mode.
7. Write to the UART Control 0 Register to:
   – Set the transmit enable bit (TEN) to enable the UART for data transmission.
   – Enable parity, if appropriate and if MULTIPROCESSOR Mode is not enabled, and select either even or odd parity.

Bit	Description (Continued)
[1] STOP	**Stop Bit Select** 0 = The transmitter sends one stop bit. 1 = The transmitter sends two stop bits.
[0] LBEN	**Loop Back Enable** 0 = Normal operation. 1 = All transmitted data is looped back to the receiver.

Table 69. UART Control 1 Register (U0CTL1)

Bit	7	6	5	4	3	2	1	0
Field	MPMD[1]	MPEN	MPMD[0]	MPBT	DEPOL	BRGCTL	RDAIRQ	IREN
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	F43H							

Bit	Description
[7,5] MPMD[1:0]	**MULTIPROCESSOR Mode** If MULTIPROCESSOR (9-bit) Mode is enabled. 00 = The UART generates an interrupt request on all received bytes (data and address). 01 = The UART generates an interrupt request only on received address bytes. 10 = The UART generates an interrupt request when a received address byte matches the value stored in the Address Compare Register and on all successive data bytes until an address mismatch occurs. 11 = The UART generates an interrupt request on all received data bytes for which the most recent address byte matched the value in the Address Compare Register.
[6] MPEN	**MULTIPROCESSOR (9-bit) Enable** This bit is used to enable MULTIPROCESSOR (9-bit) Mode. 0 = Disable MULTIPROCESSOR (9-bit) Mode. 1 = Enable MULTIPROCESSOR (9-bit) Mode.
[4] MPBT	**Multiprocessor Bit Transmit** This bit is applicable only when MULTIPROCESSOR (9-bit) Mode is enabled. The 9th bit is used by the receiving device to determine if the data byte contains address or data information. 0 = Send a 0 in the multiprocessor bit location of the data stream (data byte). 1 = Send a 1 in the multiprocessor bit location of the data stream (address byte).
[3] DEPOL	**Driver Enable Polarity** 0 = DE signal is Active High. 1 = DE signal is Active Low.

Flash Memory

The products in Z8 Encore! XP F0823 Series features either 8KB (8192), 4KB (4096), 2KB (2048) or 1KB (1024) of nonvolatile Flash memory with read/write/erase capability. Flash Memory can be programmed and erased in-circuit by either user code or through the On-Chip Debugger.

The Flash Memory array is arranged in pages with 512 bytes per page. The 512-byte page is the minimum Flash block size that can be erased. Each page is divided into 8 rows of 64 bytes.

For program/data protection, the Flash memory is also divided into sectors. In the Z8 Encore! XP F0823 Series, these sectors are either 1024 bytes (in the 8KB devices) or 512 bytes in size (all other memory sizes); each sector maps to a page. Page and sector sizes are not generally equal.

The first two bytes of the Flash program memory are used as Flash Option bits. For more information about their operation, see the Flash Option Bits chapter on page 146.

Table 79 describes the Flash memory configuration for each device in the Z8 Encore! XP F0823 Series. Figure 20 displays the Flash memory arrangement.

Table 79. Z8 Encore! XP F0823 Series Flash Memory Configurations

Part Number	Flash Size KB (Bytes)	Flash Pages	Program Memory Addresses	Flash Sector Size (bytes)
Z8F08x3	8 (8192)	16	0000H–1FFFH	1024
Z8F04x3	4 (4096)	8	0000H–0FFFH	512
Z8F02x3	2 (2048)	4	0000H–07FFH	512
Z8F01x3	1 (1024)	2	0000H–03FFH	512

Table 80. Flash Code Protection Using the Flash Option Bits

FWP	Flash Code Protection Description
0	Programming and erasing disabled for all of Flash Program Memory. In user code programming, Page Erase, and Mass Erase are all disabled. Mass Erase is available through the On-Chip Debugger.
1	Programming, Page Erase, and Mass Erase are enabled for all of Flash Program Memory.

Flash Code Protection Using the Flash Controller

At Reset, the Flash Controller locks to prevent accidental program or erasure of the Flash memory. To program or erase the Flash memory, first write the Page Select Register with the target page. Unlock the Flash Controller by making two consecutive writes to the Flash Control Register with the values 73H and 8CH, sequentially. The Page Select Register must be rewritten with the same page previously stored there. If the two Page Select writes do not match, the controller reverts to a locked state. If the two writes match, the selected page becomes active. For more details, see Figure 21.

After unlocking a specific page, you can enable either Page Program or Erase. Writing the value 95H causes a Page Erase only if the active page resides in a sector that is not protected. Any other value written to the Flash Control Register locks the Flash Controller. Mass Erase is not allowed in the user code but only in through the Debug Port.

After unlocking a specific page, you can also write to any byte on that page. After a byte is written, the page remains unlocked, allowing for subsequent writes to other bytes on the same page. Further writes to the Flash Control Register cause the active page to revert to a locked state.

Sector-Based Flash Protection

The final protection mechanism is implemented on a per-sector basis. The Flash memories of Z8 Encore! XP devices are divided into maximum number of 8 sectors. A sector is 1/8 of the total Flash memory size unless this value is smaller than the page size – in which case, the sector and page sizes are equal. On Z8 Encore! F0823 Series devices, the sector size is varied according to the Flash memory configuration shown in Table 79 on page 134.

The Flash Sector Protect Register can be configured to prevent sectors from being programmed or erased. After a sector is protected, it cannot be unprotected by user code. The Flash Sector Protect Register is cleared after reset, and any previously-written protection values are lost. User code must write this register in their initialization routine if they prefer to enable sector protection.

The Flash Sector Protect Register shares its Register File address with the Page Select Register. The Flash Sector Protect Register is accessed by writing the Flash Control Register with 5EH. After the Flash Sector Protect Register is selected, it can be accessed at the Page Select Register address. When user code writes the Flash Sector Protect Register,

On-Chip Debugger

Z8 Encore! XP F0823 Series devices contain an integrated On-Chip Debugger (OCD) which provides advanced debugging features that include:

- Single pin interface
- Reading and writing of the register file
- Reading and writing of program and data memory
- Setting of breakpoints and watchpoints
- Executing eZ8 CPU instructions
- Debug pin sharing with general-purpose input-output function to maximize the pins available

Architecture

The on-chip debugger consists of four primary functional blocks: transmitter, receiver, auto-baud detector/generator, and debug controller. Figure 22 displays the architecture of the OCD.

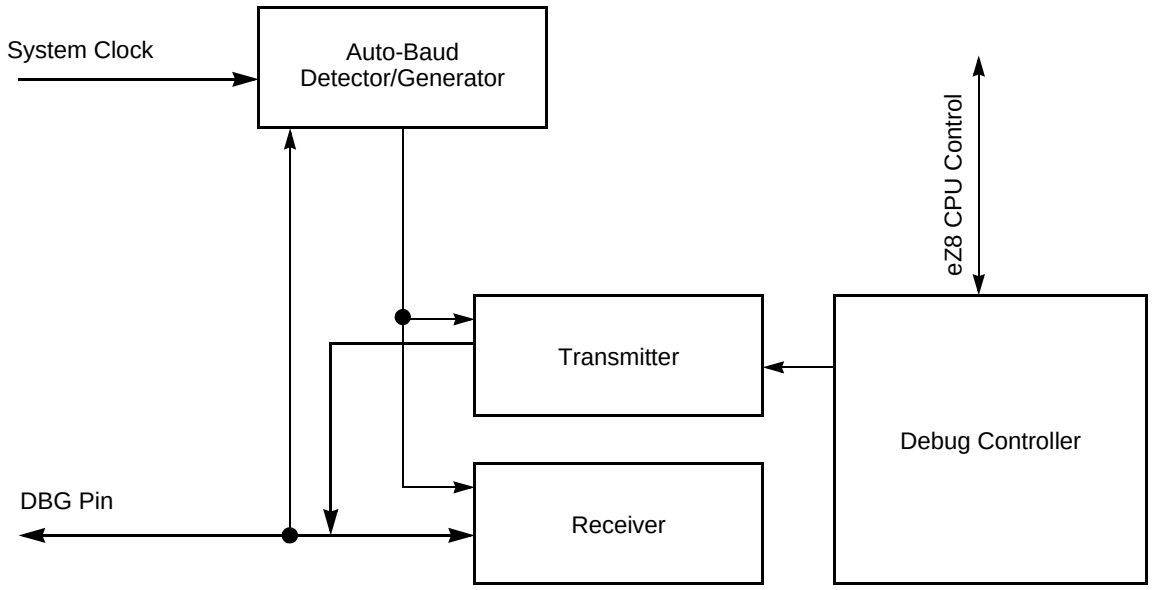

Figure 22. On-Chip Debugger Block Diagram

OCD Unlock Sequence (8-Pin Devices Only)

Because of pin-sharing on the 8-pin device, an unlock sequence must be performed to access the DBG pin. If this sequence is not completed during a system reset, then the PA0/DBG pin functions only as a GPIO pin.

The following sequence unlocks the DBG pin:

1. Hold PA2/$\overline{\text{RESET}}$ Low.
2. Wait 5 ms for the internal reset sequence to complete.
3. Send the following bytes serially to the debug pin:

```
DBG ← 80H (autobaud)
DBG ← EBH
DBG ← 5AH
DBG ← 70H
DBG ← CDH (32-bit unlock key)
```

4. Release PA2/$\overline{\text{RESET}}$. The PA0/DBG pin is now identical in function to that of the DBG pin on the 20- or 28-pin device. To enter DEBUG Mode, reautobaud and write 80H to the OCD Control Register (see the On-Chip Debugger Commands section on page 162).

Breakpoints

Execution breakpoints are generated using the BRK instruction (opcode 00H). When the eZ8 CPU decodes a BRK instruction, it signals the OCD. If breakpoints are enabled, the OCD enters DEBUG Mode and idles the eZ8 CPU. If breakpoints are not enabled, the OCD ignores the BRK signal and the BRK instruction operates as an NOP instruction.

Breakpoints in Flash Memory

The BRK instruction is opcode 00H, which corresponds to the fully programmed state of a byte in Flash memory. To implement a breakpoint, write 00H to the required break address, overwriting the current instruction. To remove a breakpoint, the corresponding page of Flash memory must be erased and reprogrammed with the original data.

Runtime Counter

The OCD contains a 16-bit Runtime Counter. It counts system clock cycles between breakpoints. The counter starts counting when the OCD leaves DEBUG Mode and stops counting when it enters DEBUG Mode again or when it reaches the maximum count of FFFFH.

Table 117. Rotate and Shift Instructions (Continued)

Mnemonic	Operands	Instruction
RR	dst	Rotate Right
RRC	dst	Rotate Right through Carry
SRA	dst	Shift Right Arithmetic
SRL	dst	Shift Right Logical
SWAP	dst	Swap Nibbles

eZ8 CPU Instruction Summary

Table 118 summarizes the eZ8 CPU instruction set. The table identifies the addressing modes employed by the instruction, the effect upon the Flags Register, the number of CPU clock cycles required for the instruction fetch, and the number of CPU clock cycles required for the instruction execution.

Table 118. eZ8 CPU Instruction Summary

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
ADC dst, src	dst ← dst + src + C	r	r	12	*	*	*	*	0	*	2	3
		r	Ir	13							2	4
		R	R	14							3	3
		R	IR	15							3	4
		R	IM	16							3	3
		IR	IM	17							3	4
ADCX dst, src	dst ← dst + src + C	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							4	3

Note: Flags Notation:
* = Value is a function of the result of the operation.
– = Unaffected.
X = Undefined.
0 = Reset to 0.
1 = Set to 1.

Table 118. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
TCM dst, src	(NOT dst) AND src	r	r	62	–	*	*	0	–	–	2	3
		r	Ir	63							2	4
		R	R	64							3	3
		R	IR	65							3	4
		R	IM	66							3	3
		IR	IM	67							3	4
TCMX dst, src	(NOT dst) AND src	ER	ER	68	–	*	*	0	–	–	4	3
		ER	IM	69							4	3
TM dst, src	dst AND src	r	r	72	–	*	*	0	–	–	2	3
		r	Ir	73							2	4
		R	R	74							3	3
		R	IR	75							3	4
		R	IM	76							3	3
		IR	IM	77							3	4
TMX dst, src	dst AND src	ER	ER	78	–	*	*	0	–	–	4	3
		ER	IM	79							4	3
TRAP Vector	SP ← SP – 2 @SP ← PC SP ← SP – 1 @SP ← FLAGS PC ← @Vector		Vector	F2	–	–	–	–	–	–	2	6
WDT				5F	–	–	–	–	–	–	1	2

Note: Flags Notation:
* = Value is a function of the result of the operation.
– = Unaffected.
X = Undefined.
0 = Reset to 0.
1 = Set to 1.

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
Z8 Encore! XP F0823 Series with 4 KB Flash								
Standard Temperature: 0°C to 70°C								
Z8F0413PB005SG	4 KB	1 KB	6	12	2	0	1	PDIP 8-pin package
Z8F0413QB005SG	4 KB	1 KB	6	12	2	0	1	QFN 8-pin package
Z8F0413SB005SG	4 KB	1 KB	6	12	2	0	1	SOIC 8-pin package
Z8F0413SH005SG	4 KB	1 KB	16	18	2	0	1	SOIC 20-pin package
Z8F0413HH005SG	4 KB	1 KB	16	18	2	0	1	SSOP 20-pin package
Z8F0413PH005SG	4 KB	1 KB	16	18	2	0	1	PDIP 20-pin package
Z8F0413SJ005SG	4 KB	1 KB	24	18	2	0	1	SOIC 28-pin package
Z8F0413HJ005SG	4 KB	1 KB	24	18	2	0	1	SSOP 28-pin package
Z8F0413PJ005SG	4 KB	1 KB	24	18	2	0	1	PDIP 28-pin package
Extended Temperature: –40°C to 105°C								
Z8F0413PB005EG	4 KB	1 KB	6	12	2	0	1	PDIP 8-pin package
Z8F0413QB005EG	4 KB	1 KB	6	12	2	0	1	QFN 8-pin package
Z8F0413SB005EG	4 KB	1 KB	6	12	2	0	1	SOIC 8-pin package
Z8F0413SH005EG	4 KB	1 KB	16	18	2	0	1	SOIC 20-pin package
Z8F0413HH005EG	4 KB	1 KB	16	18	2	0	1	SSOP 20-pin package
Z8F0413PH005EG	4 KB	1 KB	16	18	2	0	1	PDIP 20-pin package
Z8F0413SJ005EG	4 KB	1 KB	24	18	2	0	1	SOIC 28-pin package
Z8F0413HJ005EG	4 KB	1 KB	24	18	2	0	1	SSOP 28-pin package
Z8F0413PJ005EG	4 KB	1 KB	24	18	2	0	1	PDIP 28-pin package

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
Z8 Encore! XP F0823 Series with 2 KB Flash								
Standard Temperature: 0°C to 70°C								
Z8F0213PB005SG	2 KB	512 B	6	12	2	0	1	PDIP 8-pin package
Z8F0213QB005SG	2 KB	512 B	6	12	2	0	1	QFN 8-pin package
Z8F0213SB005SG	2 KB	512 B	6	12	2	0	1	SOIC 8-pin package
Z8F0213SH005SG	2 KB	512 B	16	18	2	0	1	SOIC 20-pin package
Z8F0213HH005SG	2 KB	512 B	16	18	2	0	1	SSOP 20-pin package
Z8F0213PH005SG	2 KB	512 B	16	18	2	0	1	PDIP 20-pin package
Z8F0213SJ005SG	2 KB	512 B	24	18	2	0	1	SOIC 28-pin package
Z8F0213HJ005SG	2 KB	512 B	24	18	2	0	1	SSOP 28-pin package
Z8F0213PJ005SG	2 KB	512 B	24	18	2	0	1	PDIP 28-pin package
Extended Temperature: –40°C to 105°C								
Z8F0213PB005EG	2 KB	512 B	6	12	2	0	1	PDIP 8-pin package
Z8F0213QB005EG	2 KB	512 B	6	12	2	0	1	QFN 8-pin package
Z8F0213SB005EG	2 KB	512 B	6	12	2	0	1	SOIC 8-pin package
Z8F0213SH005EG	2 KB	512 B	16	18	2	0	1	SOIC 20-pin package
Z8F0213HH005EG	2 KB	512 B	16	18	2	0	1	SSOP 20-pin package
Z8F0213PH005EG	2 KB	512 B	16	18	2	0	1	PDIP 20-pin package
Z8F0213SJ005EG	2 KB	512 B	24	18	2	0	1	SOIC 28-pin package
Z8F0213HJ005EG	2 KB	512 B	24	18	2	0	1	SSOP 28-pin package
Z8F0213PJ005EG	2 KB	512 B	24	18	2	0	1	PDIP 28-pin package

J

L

M

N

O