



Welcome to [E-XFL.COM](#)

### What is "[Embedded - Microcontrollers](#)"?

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

### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	5MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	16
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 7x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	-
Purchase URL	<a href="https://www.e-xfl.com/product-detail/zilog/z8f0823hh005ec">https://www.e-xfl.com/product-detail/zilog/z8f0823hh005ec</a>

**Table 5. Pin Characteristics (8-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	5 V Tolerance
PA0/DBG	I/O	I (but can change during reset if key sequence detected)	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	Yes, unless pull-ups enabled
PA1	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	Yes, unless pull-ups enabled
$\overline{\text{RESET}}$ / PA2	I/O	I/O (defaults to RESET)	N/A	Yes	Programmable for PA2; always on for RESET	Yes	Programmable for PA2; always on for RESET	Yes, unless pull-ups enabled
PA[5:3]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	Yes, unless pull-ups enabled
VDD	N/A	N/A	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	N/A	N/A

**Table 8. Register File Address Map (Continued)**

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page No
F0C	Timer 1 PWM High Byte	T1PWMH	00	81
F0D	Timer 1 PWM Low Byte	T1PWML	00	82
F0E	Timer 1 Control 0	T1CTL0	00	82
F0F	Timer 1 Control 1	T1CTL1	00	80
F10–F3F	Reserved	—	XX	
<b>UART</b>				
F40	UART0 Transmit Data	U0TXD	XX	104
	UART0 Receive Data	U0RXD	XX	105
F41	UART0 Status 0	U0STAT0	0000011Xb	105
F42	UART0 Control 0	U0CTL0	00	107
F43	UART0 Control 1	U0CTL1	00	107
F44	UART0 Status 1	U0STAT1	00	106
F45	UART0 Address Compare	U0ADDR	00	109
F46	UART0 Baud Rate High Byte	U0BRH	FF	110
F47	UART0 Baud Rate Low Byte	U0BRL	FF	110
F48–F6F	Reserved	—	XX	
<b>Analog-to-Digital Converter (ADC)</b>				
F70	ADC Control 0	ADCCTL0	00	122
F71	ADC Control 1	ADCCTL1	80	122
F72	ADC Data High Byte	ADCD_H	XX	124
F73	ADC Data Low Bits	ADCD_L	XX	124
F74–F7F	Reserved	—	XX	
<b>Low Power Control</b>				
F80	Power Control 0	PWRCTL0	80	33
F81	Reserved	—	XX	
<b>LED Controller</b>				
F82	LED Drive Enable	LEDEN	00	51
F83	LED Drive Level High Byte	LEDLVLH	00	51
F84	LED Drive Level Low Byte	LEDLVLL	00	52
F85	Reserved	—	XX	
<b>Oscillator Control</b>				
F86	Oscillator Control	OSCCTL	A0	167
F87–F8F	Reserved	—	XX	
<b>Comparator 0</b>				
F90	Comparator 0 Control	CMP0	14	128

**Table 15. Port Alternate Function Mapping (Non 8-Pin Parts) (Continued)**

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port B	PB0	Reserved		AFS1[0]: 0
		ANA0	ADC Analog Input	AFS1[0]: 1
	PB1	Reserved		AFS1[1]: 0
		ANA1	ADC Analog Input	AFS1[1]: 1
	PB2	Reserved		AFS1[2]: 0
		ANA2	ADC Analog Input	AFS1[2]: 1
	PB3	CLKIN	External Clock Input	AFS1[3]: 0
		ANA3	ADC Analog Input	AFS1[3]: 1
	PB4	Reserved		AFS1[4]: 0
		ANA7	ADC Analog Input	AFS1[4]: 1
	PB5	Reserved		AFS1[5]: 0
		VREF*	ADC Voltage Reference	AFS1[5]: 1
	PB6	Reserved		AFS1[6]: 0
		Reserved		AFS1[6]: 1
	PB7	Reserved		AFS1[7]: 0
		Reserved		AFS1[7]: 1

**Note:** Because there are at most two choices of alternate function for any pin of Port B, the Alternate Function Set register AFS2 is implemented but not used to select the function. Also, alternate function selection as described in Port A–C Alternate Function Sub-Registers must also be enabled.

\* VREF is available on PB5 in 28-pin products only.

# Timers

Z8 Encore! XP<sup>®</sup> F0823 Series products contain up to two 16-bit reloadable timers that are used for timing, event counting, or generation of PWM signals. The timers' features include:

- 16-bit reload counter.
- Programmable prescaler with prescale values from 1 to 128.
- PWM output generation.
- Capture and compare capability.
- External input pin for timer input, clock gating, or capture signal. External input pin signal frequency is limited to a maximum of one-fourth the system clock frequency.
- Timer output pin.
- Timer interrupt.

In addition to the timers described in this chapter, the baud rate generator of the UART (if unused) also provides basic timing functionality. For information on using the baud rate generator as an additional timer, see Universal Asynchronous Receiver/Transmitter on page 93.

## Architecture

Figure 9 displays the architecture of the timers.

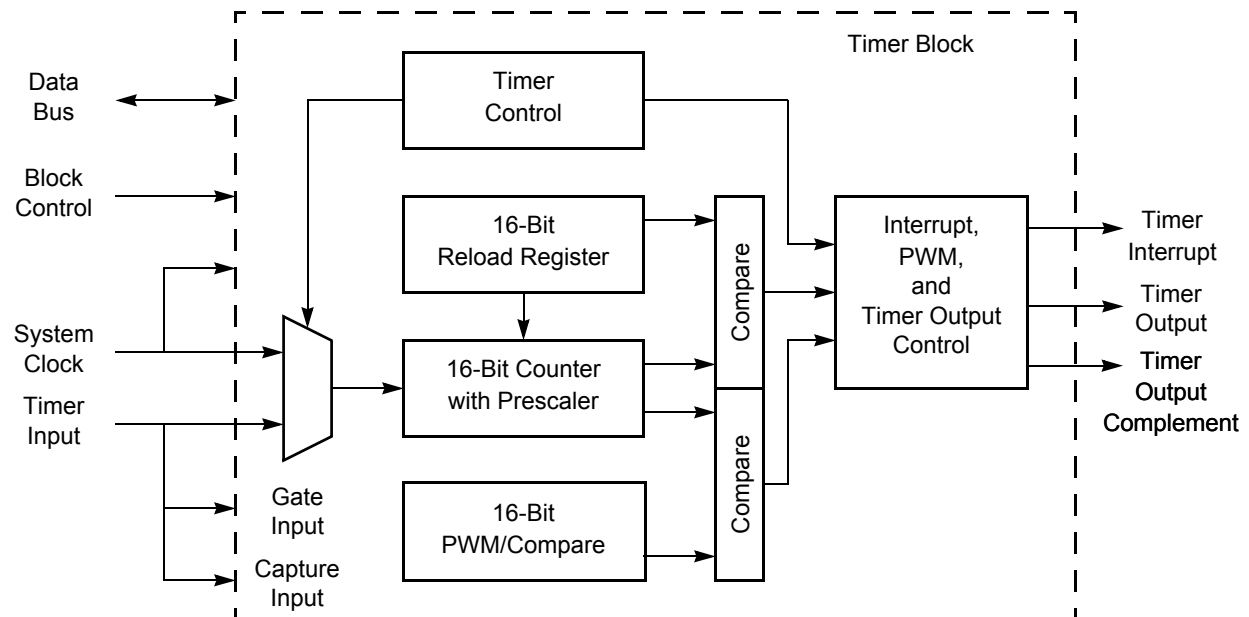


Figure 9. Timer Block Diagram

## COUNTER Mode

In COUNTER mode, the timer counts input transitions from a GPIO port pin. The timer input is taken from the GPIO port pin Timer Input alternate function. The **TPOL** bit in the Timer Control register selects whether the count occurs on the rising edge or the falling edge of the timer input signal. In COUNTER mode, the prescaler is disabled.

**! Caution:** *The input frequency of the timer input signal must not exceed one-fourth the system clock frequency.*

Upon 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 COUNTER mode and initiating the count:

1. Write to the Timer Control register to:
  - Disable the timer.
  - Configure the timer for COUNTER mode.
  - Select either the rising edge or falling edge of the Timer Input signal for the count. This selection 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 only affects the first pass in COUNTER mode. After the first timer Reload in COUNTER mode, counting always begins at the reset value of 0001H. In 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. Configure the associated GPIO port pin for the Timer Input alternate function.
6. If using the Timer Output function, configure the associated GPIO port pin for the Timer Output alternate function.
7. Write to the Timer Control register to enable the timer.

In COUNTER mode, the number of Timer Input transitions since the timer start is given by the following equation:

$$\text{COUNTER Mode Timer Input Transitions} = \text{Current Count Value} - \text{Start Value}$$

### 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:

$$\text{Comparator Output Transitions} = \text{Current Count Value} - \text{Start Value}$$

## Timer Control Register Definitions

### Timer 0–1 High and Low Byte Registers

The Timer 0–1 High and Low Byte (TxH and TxL) registers (Table 49 and Table 50) contain the current 16-bit timer count value. When the timer is enabled, a read from TxH causes the value in TxL to be stored in a temporary holding register. A read from TxL always returns this temporary register when the timers are enabled. When the timer is disabled, reads from the TxL reads the register directly.

Writing to the Timer High and Low Byte registers while the timer is enabled is not recommended. There are no temporary holding registers available for write operations, so simultaneous 16-bit writes are not possible. If either the Timer High or Low Byte registers are written during counting, the 8-bit written value is placed in the counter (High or Low Byte) at the next clock edge. The counter continues counting from the new value.

**Table 49. Timer 0–1 High Byte Register (TxH)**

BITS	7	6	5	4	3	2	1	0
FIELD	TH							
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	F00H, F08H							

**Table 50. Timer 0–1 Low Byte Register (TxL)**

BITS	7	6	5	4	3	2	1	0
FIELD	TL							
RESET	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F01H, F09H							

TH and TL—Timer High and Low Bytes

These 2 bytes, {TH[7:0], TL[7:0]}, contain the current 16-bit timer count value

### Timer Reload High and Low Byte Registers

The Timer 0–1 Reload High and Low Byte (TxRH and TxRL) registers (Table 51 and Table 52) store a 16-bit Reload value, {TRH[7:0], TRL[7:0]}. Values written to the Timer Reload High Byte register are stored in a temporary holding register. When a write to the Timer Reload Low Byte register occurs, the temporary holding register value is written to the Timer High Byte register. This operation allows simultaneous updates of the 16-bit Timer Reload value.



### WDT Reset in NORMAL Operation

If configured to generate a Reset when a time-out occurs, the Watchdog Timer forces the device into the System Reset state. The WDT status bit in the Watchdog Timer Control register is set to 1. For more information on System Reset, see Reset and Stop Mode Recovery on page 21.

### WDT Reset in STOP Mode

If configured to generate a Reset when a time-out occurs and the device is in STOP mode, the Watchdog Timer initiates a Stop Mode Recovery. Both the WDT status bit and the STOP bit in the Watchdog Timer Control register are set to 1 following WDT time-out in STOP mode. For more information, see Reset and Stop Mode Recovery on page 21.

## Watchdog Timer Reload Unlock Sequence

Writing the unlock sequence to the Watchdog Timer Control Register (WDTCTL) address unlocks the three Watchdog Timer Reload Byte Registers (WDTU, WDTL, and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL register address produce no effect on the bits in the WDTCTL register. The locking mechanism prevents spurious writes to the Reload registers. The following sequence is required to unlock the Watchdog Timer Reload Byte Registers (WDTU, WDTL, and WDTL) for write access.

1. Write 55H to the Watchdog Timer Control register (WDTCTL).
2. Write AAH to the Watchdog Timer Control register (WDTCTL).
3. Write the Watchdog Timer Reload Upper Byte register (WDTU).
4. Write the Watchdog Timer Reload High Byte register (WDTL).
5. Write the Watchdog Timer Reload Low Byte register (WDTL).

All three Watchdog Timer Reload registers must be written in the order just listed. There must be no other register writes between each of these operations. If a register write occurs, the lock state machine resets and no further writes can occur unless the sequence is restarted. The value in the Watchdog Timer Reload registers is loaded into the counter when the Watchdog Timer is first enabled and every time a WDT instruction is executed.

## Watchdog Timer Control Register Definitions

### Watchdog Timer Control Register

The Watchdog Timer Control (WDTCTL) register is a write-only control register. Writing the 55H, AAH unlock sequence to the WDTCTL register address unlocks the three

in hardware, software or some combination of the two, depending on the multiprocessor configuration bits. In general, the address compare feature reduces the load on the CPU, because it does not require access to the UART when it receives data directed to other devices on the multi-node network. The following three MULTIPROCESSOR modes are available in hardware:

- Interrupt on all address bytes
- Interrupt on matched address bytes and correctly framed data bytes
- Interrupt only on correctly framed data bytes

These modes are selected with `MPMD[1:0]` in the UART Control 1 Register. For all multiprocessor modes, bit `MPEN` of the UART Control 1 Register must be set to 1.

The first scheme is enabled by writing `01b` to `MPMD[1:0]`. In this mode, all incoming address bytes cause an interrupt, while data bytes never cause an interrupt. The interrupt service routine must manually check the address byte that caused triggered the interrupt. If it matches the UART address, the software clears `MPMD[0]`. Each new incoming byte interrupts the CPU. The software is responsible for determining the end of the frame. It checks for the end-of-frame by reading the `MPRX` bit of the UART Status 1 Register for each incoming byte. If `MPRX=1`, a new frame has begun. If the address of this new frame is different from the UART's address, `MPMD[0]` must be set to 1 causing the UART interrupts to go inactive until the next address byte. If the new frame's address matches the UART's, the data in the new frame is processed as well.

The second scheme requires the following: set `MPMD[1:0]` to `10B` and write the UART's address into the UART Address Compare register. This mode introduces additional hardware control, interrupting only on frames that match the UART's address. When an incoming address byte does not match the UART's address, it is ignored. All successive data bytes in this frame are also ignored. When a matching address byte occurs, an interrupt is issued and further interrupts now occur on each successive data byte. When the first data byte in the frame is read, the `NEWFRM` bit of the UART Status 1 Register is asserted. All successive data bytes have `NEWFRM=0`. When the next address byte occurs, the hardware compares it to the UART's address. If there is a match, the interrupts continues and the `NEWFRM` bit is set for the first byte of the new frame. If there is no match, the UART ignores all incoming bytes until the next address match.

The third scheme is enabled by setting `MPMD[1:0]` to `11b` and by writing the UART's address into the UART Address Compare Register. This mode is identical to the second scheme, except that there are no interrupts on address bytes. The first data byte of each frame remains accompanied by a `NEWFRM` assertion.

Reserved—R/W bits must be 0 during writes; 0 when read.

NEWFRM—Status bit denoting the start of a new frame. Reading the UART Receive Data register resets this bit to 0.

0 = The current byte is not the first data byte of a new frame

1 = The current byte is the first data byte of a new frame

MPRX—Multiprocessor Receive

Returns the value of the most recent multiprocessor bit received. Reading from the UART Receive Data register resets this bit to 0.

## UART Control 0 and Control 1 Registers

The UART Control 0 and Control 1 registers (Table 66 and Table 67) configure the properties of the UART's transmit and receive operations. The UART Control registers must not be written while the UART is enabled.

**Table 66. UART Control 0 Register (U0CTL0)**

BITS	7	6	5	4	3	2	1	0
FIELD	TEN	REN	CTSE	PEN	PSEL	SBRK	STOP	LBEN
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	F42H							

TEN—Transmit Enable

This bit enables or disables the transmitter. The enable is also controlled by the  $\overline{\text{CTS}}$  signal and the CTSE bit. If the  $\overline{\text{CTS}}$  signal is low and the CTSE bit is 1, the transmitter is enabled.

0 = Transmitter disabled

1 = Transmitter enabled

REN—Receive Enable

This bit enables or disables the receiver.

0 = Receiver disabled

1 = Receiver enabled

CTSE—CTS Enable

0 = The  $\overline{\text{CTS}}$  signal has no effect on the transmitter

1 = The UART recognizes the  $\overline{\text{CTS}}$  signal as an enable control from the transmitter

PEN—Parity Enable

This bit enables or disables parity. Even or odd is determined by the PSEL bit.

0 = Parity is disabled

1 = The transmitter sends data with an additional parity bit and the receiver receives an additional parity bit

**Table 76. Comparator Control Register (CMP0)**

BITS	7	6	5	4	3	2	1	0
FIELD	INPSEL	INNSEL	REFLVL				Reserved	
RESET	0	0	0	1	0	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F90H							

INPSEL—Signal Select for Positive Input

0 = GPIO pin used as positive comparator input

1 = temperature sensor used as positive comparator input

INNSEL—Signal Select for Negative Input

0 = internal reference disabled, GPIO pin used as negative comparator input

1 = internal reference enabled as negative comparator input

REFLVL—Internal Reference Voltage Level

► **Note:** *This reference is independent of the ADC voltage reference.*

0000 = 0.0 V

0001 = 0.2 V

0010 = 0.4 V

0011 = 0.6 V

0100 = 0.8 V

0101 = 1.0 V (Default)

0110 = 1.2 V

0111 = 1.4 V

1000 = 1.6 V

1001 = 1.8 V

1010–1111 = Reserved

Reserved—R/W bits must be 0 during writes; 0 when read

**Table 97. Randomized Lot ID Locations (Continued)**

<b>Info Page Address</b>	<b>Memory Address</b>	<b>Usage</b>
5C	FE5C	Randomized Lot ID Byte 23
5D	FE5D	Randomized Lot ID Byte 22
5E	FE5E	Randomized Lot ID Byte 21
5F	FE5F	Randomized Lot ID Byte 20
61	FE61	Randomized Lot ID Byte 19
62	FE62	Randomized Lot ID Byte 18
64	FE64	Randomized Lot ID Byte 17
65	FE65	Randomized Lot ID Byte 16
67	FE67	Randomized Lot ID Byte 15
68	FE68	Randomized Lot ID Byte 14
6A	FE6A	Randomized Lot ID Byte 13
6B	FE6B	Randomized Lot ID Byte 12
6D	FE6D	Randomized Lot ID Byte 11
6E	FE6E	Randomized Lot ID Byte 10
70	FE70	Randomized Lot ID Byte 9
71	FE71	Randomized Lot ID Byte 8
73	FE73	Randomized Lot ID Byte 7
74	FE74	Randomized Lot ID Byte 6
76	FE76	Randomized Lot ID Byte 5
77	FE77	Randomized Lot ID Byte 4
79	FE79	Randomized Lot ID Byte 3
7A	FE7A	Randomized Lot ID Byte 2
7C	FE7C	Randomized Lot ID Byte 1
7D	FE7D	Randomized Lot ID Byte 0 (least significant)

## 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  $\leftarrow$  80H (autobaud)  
DBG  $\leftarrow$  EBH  
DBG  $\leftarrow$  5AH  
DBG  $\leftarrow$  70H  
DBG  $\leftarrow$  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, re-autobaud and write 80H to the OCD control register (see On-Chip Debugger Commands on page 157).

## 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.

# Oscillator Control

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

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

In addition, Z8 Encore! XP F0823 Series devices contain clock failure detection and recovery circuitry, allowing 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 101 details each clock source and its usage.

**Table 101. Oscillator Configuration and Selection**

Clock Source	Characteristics	Required Setup
Internal Precision RC Oscillator	<ul style="list-style-type: none"><li>• 32.8 kHz or 5.53 MHz</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 5.53 MHz or 32.8 kHz</li></ul>
External Clock Drive	<ul style="list-style-type: none"><li>• 0 to 20 MHz</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>• 10 kHz 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>

**Table 110. CPU Control Instructions (Continued)**

Mnemonic	Operands	Instruction
SCF	—	Set Carry Flag
SRP	src	Set Register Pointer
STOP	—	STOP Mode
WDT	—	Watchdog Timer Refresh

**Table 111. Load Instructions**

Mnemonic	Operands	Instruction
CLR	dst	Clear
LD	dst, src	Load
LDC	dst, src	Load Constant to/from Program Memory
LDCI	dst, src	Load Constant to/from Program Memory and Auto-Increment Addresses
LDE	dst, src	Load External Data to/from Data Memory
LDEI	dst, src	Load External Data to/from Data Memory and Auto-Increment Addresses
LDWX	dst, src	Load Word using Extended Addressing
LDX	dst, src	Load using Extended Addressing
LEA	dst, X(src)	Load Effective Address
POP	dst	Pop
POPX	dst	Pop using Extended Addressing
PUSH	src	Push
PUSHX	src	Push using Extended Addressing

**Table 112. Logical Instructions**

Mnemonic	Operands	Instruction
AND	dst, src	Logical AND
ANDX	dst, src	Logical AND using Extended Addressing
COM	dst	Complement
OR	dst, src	Logical OR



Table 115. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
COM dst	dst ← ~dst	R		60	–	*	*	0	–	–	2	2
		IR		61							2	3
CP dst, src	dst - src	r	r	A2	*	*	*	*	–	–	2	3
		r	lr	A3							2	4
		R	R	A4							3	3
		R	IR	A5							3	4
		R	IM	A6							3	3
		IR	IM	A7							3	4
CPC dst, src	dst - src - C	r	r	1F A2	*	*	*	*	–	–	3	3
		r	lr	1F A3							3	4
		R	R	1F A4							4	3
		R	IR	1F A5							4	4
		R	IM	1F A6							4	3
		IR	IM	1F A7							4	4
CPCX dst, src	dst - src - C	ER	ER	1F A8	*	*	*	*	–	–	5	3
		ER	IM	1F A9							5	3
CPX dst, src	dst - src	ER	ER	A8	*	*	*	*	–	–	4	3
		ER	IM	A9							4	3
DA dst	dst ← DA(dst)	R		40	*	*	*	X	–	–	2	2
		IR		41							2	3
DEC dst	dst ← dst - 1	R		30	–	*	*	*	–	–	2	2
		IR		31							2	3
DECW dst	dst ← dst - 1	RR		80	–	*	*	*	–	–	2	5
		IRR		81							2	6
DI	IRQCTL[7] ← 0			8F	–	–	–	–	–	–	1	2
DJNZ dst, RA	dst ← dst - 1 if dst ≠ 0 PC ← PC + X	r		0A-FA	–	–	–	–	–	–	2	3
EI	IRQCTL[7] ← 1			9F	–	–	–	–	–	–	1	2
Flags Notation:		* = Value is a function of the result of the operation. – = Unaffected X = Undefined			0 = Reset to 0 1 = Set to 1							

# Electrical Characteristics

The data in this chapter is pre-qualification and pre-characterization and is subject to change. Additional electrical characteristics may be found in the individual chapters.

## Absolute Maximum Ratings

Stresses greater than those listed in Table 117 may cause permanent damage to the device. These ratings are stress ratings only. Operation of the device at any condition outside those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. For improved reliability, tie unused inputs to one of the supply voltages ( $V_{DD}$  or  $V_{SS}$ ).

**Table 117. Absolute Maximum Ratings**

Parameter	Minimum	Maximum	Units	Notes
Ambient temperature under bias	-40	+105	°C	
Storage temperature	-65	+150	°C	
Voltage on any pin with respect to $V_{SS}$	-0.3	+5.5	V	1
	-0.3	+3.9	V	2
Voltage on $V_{DD}$ pin with respect to $V_{SS}$	-0.3	+3.6	V	
Maximum current on input and/or inactive output pin	-5	+5	μA	
Maximum output current from active output pin	-25	+25	mA	
<b>8-pin Packages Maximum Ratings at 0 °C to 70 °C</b>				
Total power dissipation		220	mW	
Maximum current into $V_{DD}$ or out of $V_{SS}$		60	mA	
<b>20-pin Packages Maximum Ratings at 0 °C to 70 °C</b>				
Total power dissipation		430	mW	
Maximum current into $V_{DD}$ or out of $V_{SS}$		120	mA	
<b>28-pin Packages Maximum Ratings at 0 °C to 70 °C</b>				
Total power dissipation		450	mW	

Figure 33 and Table 131 provide timing information for UART pins for the case where CTS is not used for flow control. DE asserts after the transmit data register has been written. DE remains asserted for multiple characters as long as the transmit data register is written with the next character before the current character has completed.

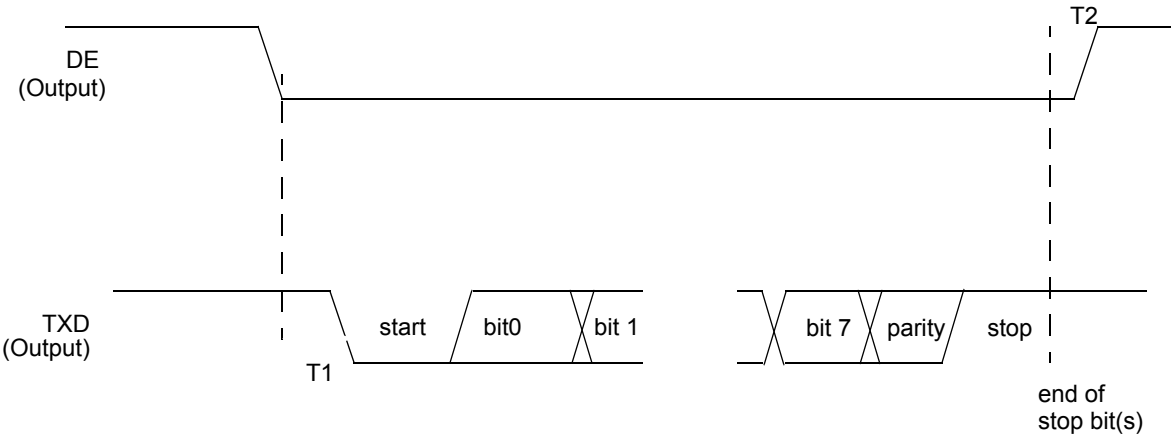
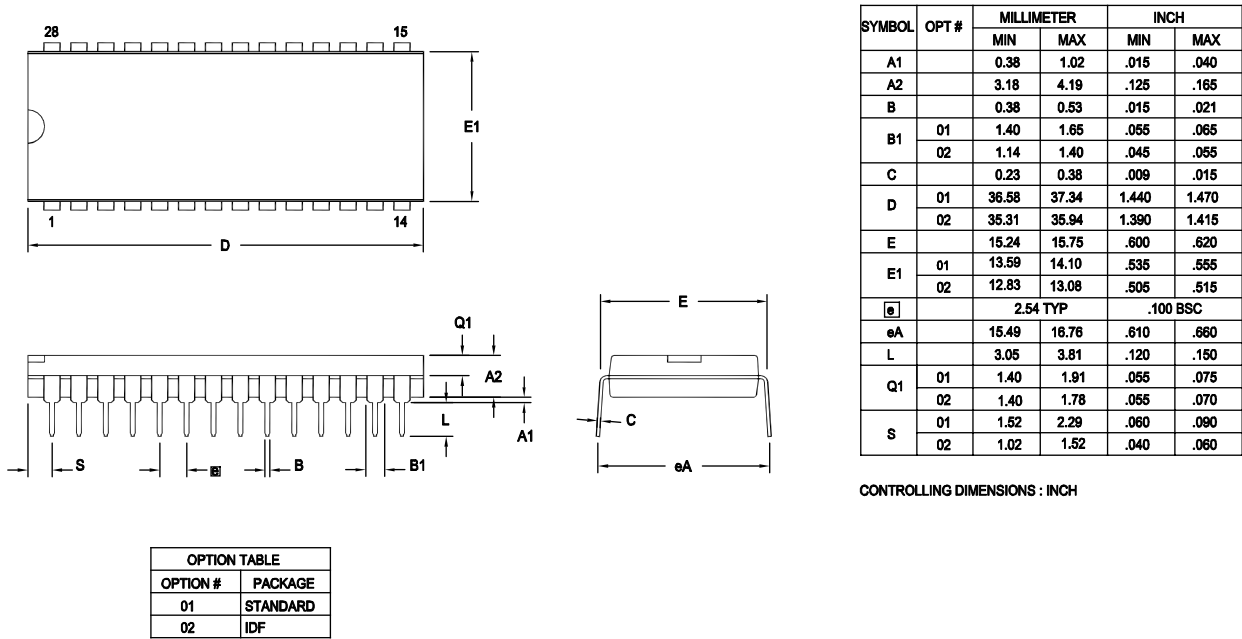


Figure 33. UART Timing Without CTS

Table 131. UART Timing Without CTS

Parameter	Abbreviation	Delay (ns)	
		Minimum	Maximum
UART			
T <sub>1</sub>	DE assertion to TXD falling edge (start bit) delay	1 * XIN period	1 bit time
T <sub>2</sub>	End of Stop Bit(s) to DE deassertion delay (Tx data register is empty)	± 5	

Figure 40 displays the 28-pin Plastic Dual Inline Package (PDIP) available for Z8 Encore! XP F0823 Series devices.



CONTROLLING DIMENSIONS : INCH

Note: ZILOG supplies both options for production. Component layout  
PCB design should cover bigger option 01.

Figure 40. 28-Pin Plastic Dual Inline Package (PDIP)

page erase 135  
page select register 138, 139  
FPS register 138, 139  
FSTAT register 137

## **G**

GATED mode 84  
general-purpose I/O 35  
GPIO 4, 35  
    alternate functions 36  
    architecture 36  
    control register definitions 43  
    input data sample timing 202  
    interrupts 43  
    port A-C pull-up enable sub-registers 48, 49  
    port A-H address registers 44  
    port A-H alternate function sub-registers 45  
    port A-H control registers 44  
    port A-H data direction sub-registers 45  
    port A-H high drive enable sub-registers 47  
    port A-H input data registers 49  
    port A-H output control sub-registers 46  
    port A-H output data registers 50  
    port A-H stop mode recovery sub-registers 47  
    port availability by device 35  
    port input timing 203  
    port output timing 204

## **H**

H 174  
HALT 176  
halt mode 32, 176  
hexadecimal number prefix/suffix 174

## **I**

I2C 4  
IM 173  
immediate data 173  
immediate operand prefix 174

INC 175  
increment 175  
increment word 175  
INCW 175  
indexed 173  
indirect address prefix 174  
indirect register 173  
indirect register pair 173  
indirect working register 173  
indirect working register pair 173  
infrared encoder/decoder (IrDA) 113  
Instruction Set 171  
instruction set, eZ8 CPU 171  
instructions  
    ADC 175  
    ADCX 175  
    ADD 175  
    ADDX 175  
    AND 177  
    ANDX 177  
    arithmetic 175  
    BCLR 176  
    BIT 176  
    bit manipulation 176  
    block transfer 176  
    BRK 178  
    BSET 176  
    BSWAP 176, 178  
    BTJ 178  
    BTJNZ 178  
    BTJZ 178  
    CALL 178  
    CCF 176  
    CLR 177  
    COM 177  
    CP 175  
    CPC 175  
    CPCX 175  
    CPU control 176  
    CPX 175  
    DA 175  
    DEC 175  
    DECW 175  
    DI 176