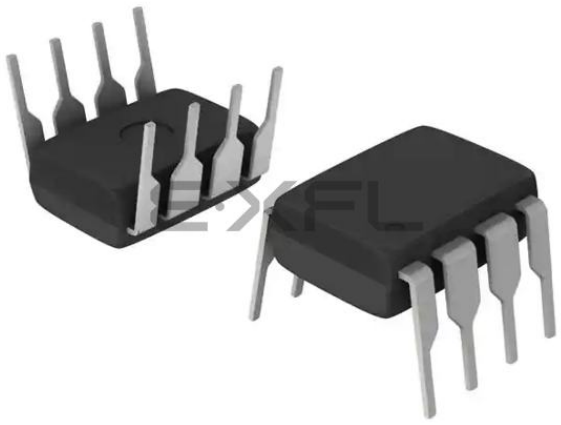


Welcome to E-XFL.COM

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	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	-
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/z8f0813pb005ec

CPU and Peripheral Overview

eZ8 CPU Features

The eZ8 CPU, Zilog's latest 8-bit central processing unit (CPU), meets the continuing demand for faster and code-efficient microcontrollers. The eZ8 CPU executes a superset of the original Z8® instruction set. The eZ8 CPU features include:

- Direct register-to-register architecture allows each register to function as an accumulator, improving execution time and decreasing the required program memory.
- Software stack allows much greater depth in subroutine calls and interrupts than hardware stacks.
- Compatible with existing Z8 code.
- Expanded internal Register File allows access of up to 4 KB.
- New instructions improve execution efficiency for code developed using higher-level programming languages, including C.
- Pipelined instruction fetch and execution.
- New instructions for improved performance including BIT, BSWAP, BTJ, CPC, LDC, LDCI, LEA, MULT, and SRL.
- New instructions support 12-bit linear addressing of the Register file.
- Up to 10 MIPS operation.
- C-Compiler friendly.
- 2 to 9 clock cycles per instruction.

For more information on eZ8 CPU, refer to *eZ8 CPU Core User Manual (UM0128)* available for download at www.zilog.com.

General-Purpose I/O

Z8 Encore! XP F0823 Series features 6 to 24 port pins (Ports A–C) for general-purpose I/O (GPIO). The number of GPIO pins available is a function of package. Each pin is individually programmable. 5 V tolerant input pins are available on all I/Os on 8-pin devices, most I/Os on other package types.

Flash Controller

The Flash Controller programs and erases Flash memory. The Flash Controller supports protection against accidental program and erasure, as well as factory serialization and read protection.

Low-Power Modes

Z8 Encore! XP® F0823 Series products contain power-saving features. The highest level of power reduction is provided by the STOP mode, in which nearly all device functions are powered down. The next lower level of power reduction is provided by the HALT mode, in which the CPU is powered down.

Further power savings can be implemented by disabling individual peripheral blocks while in ACTIVE mode (defined as being in neither STOP nor HALT mode).

STOP Mode

Executing the eZ8 CPU's Stop instruction places the device into STOP mode, powering down all peripherals except the Voltage Brownout detector, and the Watchdog Timer. These two blocks may also be disabled for additional power savings. In STOP mode, the operating characteristics are:

- Primary crystal oscillator and internal precision oscillator are stopped; XIN and XOUT (if previously enabled) are disabled, and PA0/PA1 revert to the states programmed by the GPIO registers.
- System clock is stopped.
- eZ8 CPU is stopped.
- Program counter (PC) stops incrementing.
- Watchdog Timer's internal RC oscillator continues to operate if enabled by the Oscillator Control Register.
- If enabled, the Watchdog Timer logic continues to operate.
- If enabled for operation in STOP mode by the associated Flash Option Bit, the Voltage Brownout protection circuit continues to operate.
- All other on-chip peripherals are idle.

To minimize current in STOP mode, all GPIO pins that are configured as digital inputs must be driven to one of the supply rails (V_{CC} or GND). Additionally, any GPIOs configured as outputs must also be driven to one of the supply rails. The device can be brought out of STOP mode using Stop Mode Recovery. For more information on Stop Mode Recovery, see Reset and Stop Mode Recovery on page 21.

Table 19. Port A–C Control Registers (P*x*CTL)

BITS	7	6	5	4	3	2	1	0
FIELD	PCTL							
RESET	00H							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FD1H, FD5H, FD9H							

PCTL[7:0]—Port Control
The Port Control register provides access to all sub-registers that configure the GPIO Port operation.

Port A–C Data Direction Sub-Registers

The Port A–C Data Direction sub-register is accessed through the Port A–C Control register by writing 01H to the Port A–C Address register (Table 20).

Table 20. Port A–C Data Direction Sub-Registers (PxDD)

BITS	7	6	5	4	3	2	1	0
FIELD	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
RESET	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	If 01H in Port A–C Address Register, accessible through the Port A–C Control Register							

DD[7:0]—Data Direction
These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction register setting.
0 = Output. Data in the Port A–C Output Data register is driven onto the port pin.
1 = Input. The port pin is sampled and the value written into the Port A–C Input Data Register. The output driver is tristated.

Port A–C Alternate Function Sub-Registers

The Port A–C Alternate Function sub-register (Table 21) is accessed through the Port A–C Control register by writing `02H` to the Port A–C Address register. The Port A–C Alternate Function sub-registers enable the alternate function selection on pins. If disabled, pins functions as GPIO. If enabled, select one of four alternate functions using alternate function set subregisters 1 and 2 as described in the Port A–C Alternate Function Set 1 Sub-Registers on page 48 and Port A–C Alternate Function Set 2 Sub-Registers on

Timers

Z8 Encore! XP® 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.

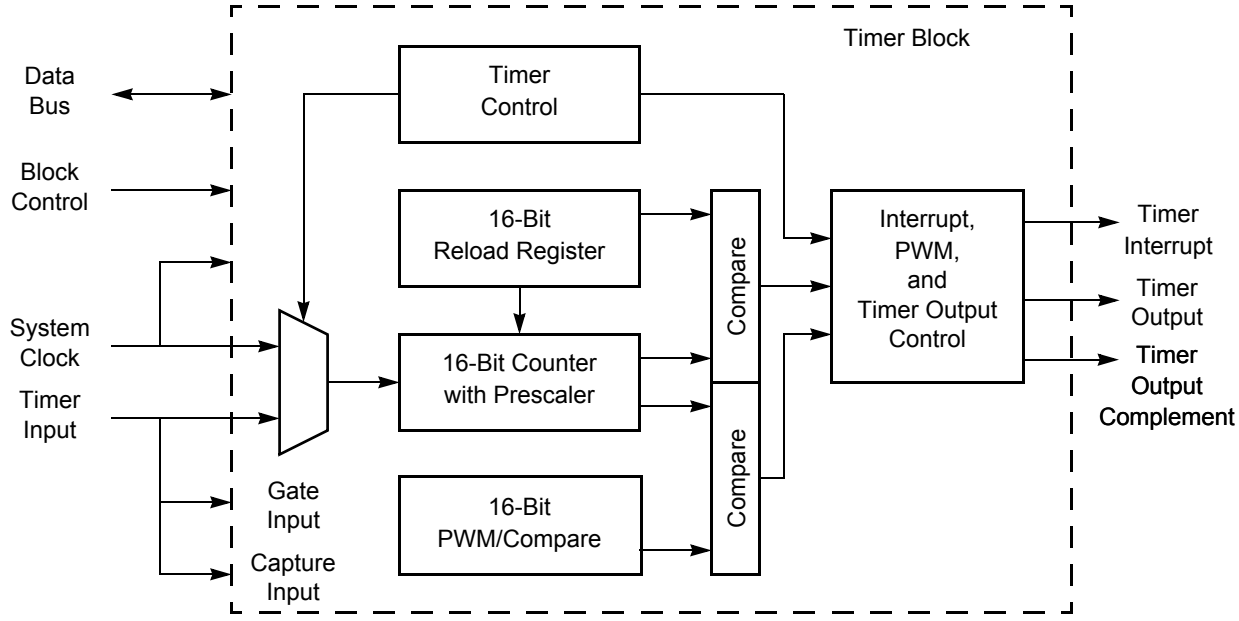


Figure 9. Timer Block Diagram

Table 54. Timer 0–1 PWM Low Byte Register (TxPWML)

BITS	7	6	5	4	3	2	1	0
FIELD	PWML							
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	F05H, F0DH							

PWMH and PWML—Pulse-Width Modulator High and Low Bytes
These two bytes, {PWMH[7:0], PWML[7:0]}, form a 16-bit value that is compared to the current 16-bit timer count. When a match occurs, the PWM output changes state. The PWM output value is set by the TPOL bit in the Timer Control Register (TxCTL1) register.

The TxPWMH and TxPWML registers also store the 16-bit captured timer value when operating in CAPTURE or CAPTURE/COMPARE modes.

Timer 0–1 Control Registers

Time 0–1 Control Register 0

The Timer Control Register 0 (TxCTL0) and Timer Control Register 1 (TxCTL1) determine the timer operating mode. It also includes a programmable PWM deadband delay, two bits to configure timer interrupt definition, and a status bit to identify if the most recent timer interrupt is caused by an input Capture event.

Table 55. Timer 0–1 Control Register 0 (TxCTL0)

BITS	7	6	5	4	3	2	1	0
FIELD	TMODEHI	TICONFIG		Reserved	PWMD			INPCAP
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	F06H, F0EH							

TMODEHI—Timer Mode High Bit
This bit along with the TMODE field in TxCTL1 register determines the operating mode of the timer. This is the most-significant bit of the Timer mode selection value. See the TxCTL1 register description for more details.

TICONFIG—Timer Interrupt Configuration
This field configures timer interrupt definition.

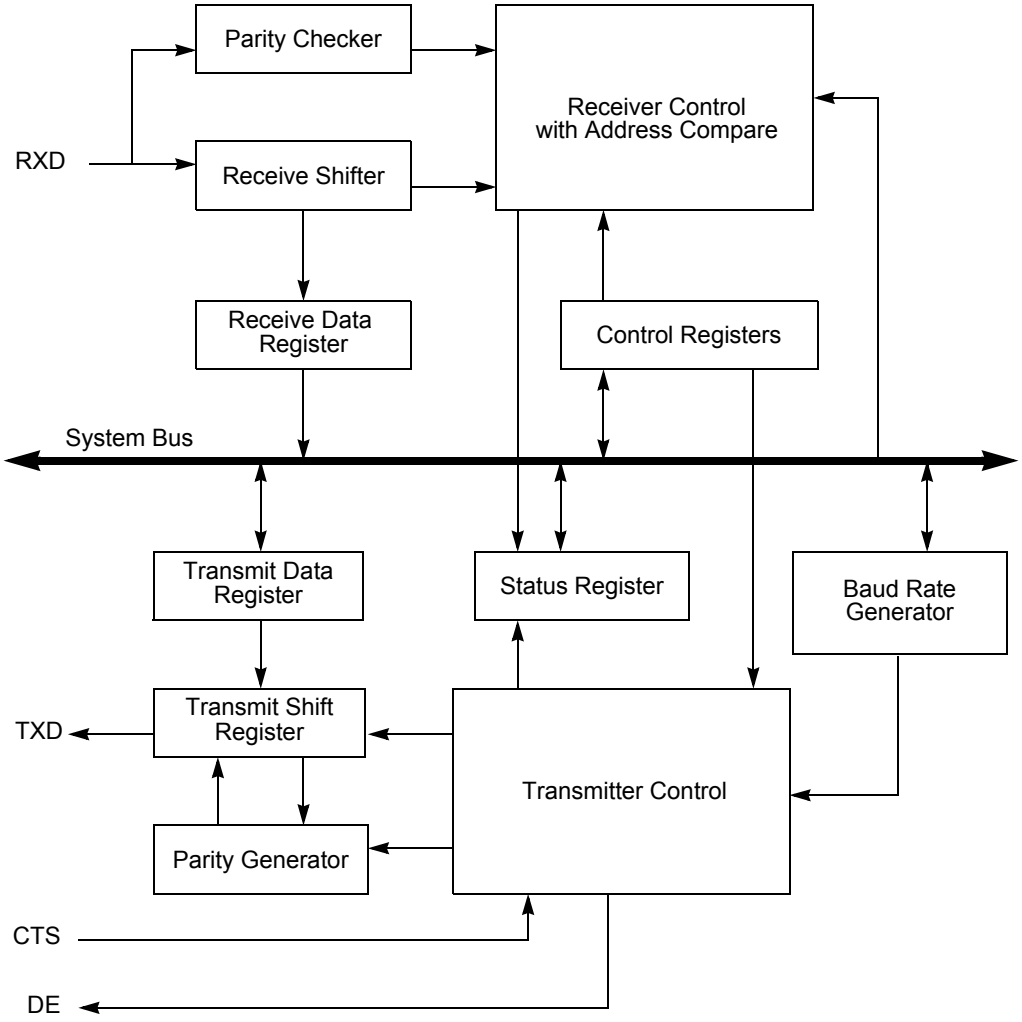


Figure 10. UART Block Diagram

Operation

Data Format

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.

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. Follow the steps below 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.
   - Set or clear CTSE to enable or disable control from the remote receiver using the $\overline{\text{CTS}}$ pin.
8. Execute an EI instruction to enable interrupts.

Receiving IrDA Data

Data received from the infrared transceiver using the IR_RXD signal through the RXD pin is decoded by the Infrared Endec and passed to the UART. The UART's baud rate clock is used by the Infrared Endec to generate the demodulated signal (RXD) that drives the UART. Each UART/Infrared data bit is 16-clocks wide. Figure 18 displays data reception. When the Infrared Endec is enabled, the UART's RXD signal is internal to the Z8 Encore! XP® F0823 Series products while the IR_RXD signal is received through the RXD pin.

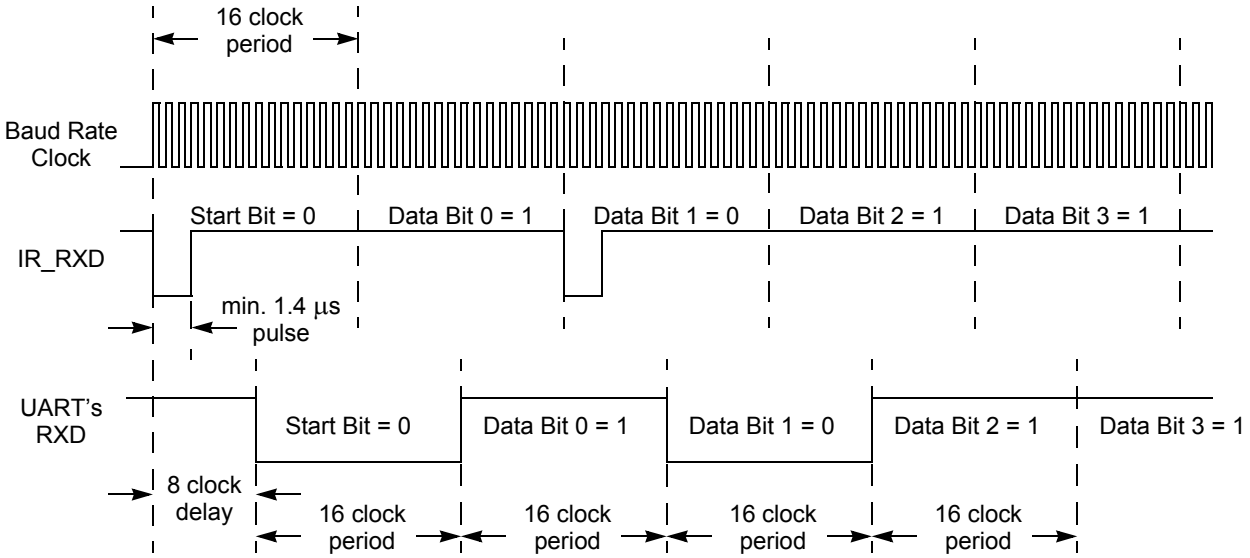


Figure 18. IrDA Data Reception

Infrared Data Reception

Caution: *The system clock frequency must be at least 1.0 MHz to ensure proper reception of the 1.4 µs minimum width pulses allowed by the IrDA standard.*

Endec Receiver Synchronization

The IrDA receiver uses a local baud rate clock counter (0 to 15 clock periods) to generate an input stream for the UART and to create a sampling window for detection of incoming pulses. The generated UART input (UART RXD) is delayed by 8 baud rate clock periods with respect to the incoming IrDA data stream. When a falling edge in the input data stream is detected, the Endec counter is reset. When the count reaches a value of 8, the UART RXD value is updated to reflect the value of the decoded data. When the count reaches 12 baud clock periods, the sampling window for the next incoming pulse opens. The window remains open until the count again reaches 8 (that is, 24 baud clock periods since the previous pulse was detected), giving the Endec a sampling window of minus four

bits of resolution are lost because of a rounding error. As a result, the final value is an 11- bit number.

Automatic Powerdown

If the ADC is idle (no conversions in progress) for 160 consecutive system clock cycles, portions of the ADC are automatically powered down. From this powerdown state, the ADC requires 40 system clock cycles to powerup. The ADC powers up when a conversion is requested by the ADC Control register.

Single-Shot Conversion

When configured for single-shot conversion, the ADC performs a single analog-to-digital conversion on the selected analog input channel. After completion of the conversion, the ADC shuts down. Follow the steps below for setting up the ADC and initiating a single-shot conversion:

1. Enable the acceptable analog inputs by configuring the general-purpose I/O pins for alternate function. This configuration disables the digital input and output drivers.
2. Write the ADC Control/Status Register 1 to configure the ADC
   - Write the REFSELH bit of the pair {REFSELH, REFSELL} to select the internal voltage reference level or to disable the internal reference. The REFSELH bit is contained in the ADC Control/Status Register 1.
3. Write to the ADC Control Register 0 to configure the ADC and begin the conversion. The bit fields in the ADC Control register can be written simultaneously:
   - Write to the ANAIN[3:0] field to select from the available analog input sources (different input pins available depending on the device).
   - Clear CONT to 0 to select a single-shot conversion.
   - If the internal voltage reference must be output to a pin, set the REFEXT bit to 1. The internal voltage reference must be enabled in this case.
   - Write the REFSELL bit of the pair {REFSELH, REFSELL} to select the internal voltage reference level or to disable the internal reference. The REFSELL bit is contained in the ADC Control Register 0.
   - Set CEN to 1 to start the conversion.
4. CEN remains 1 while the conversion is in progress. A single-shot conversion requires 5129 system clock cycles to complete. If a single-shot conversion is requested from an ADC powered-down state, the ADC uses 40 additional clock cycles to power-up before beginning the 5129 cycle conversion.

value 63H to the Flash Control register initiates the Mass Erase operation. While the Flash Controller executes the Mass Erase operation, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate. Using the On-Chip Debugger, poll the Flash Status register to determine when the Mass Erase operation is complete. When the Mass Erase is complete, the Flash Controller returns to its locked state.

Flash Controller Bypass

The Flash Controller can be bypassed and the control signals for the Flash memory brought out to the GPIO pins. Bypassing the Flash Controller allows faster Row Programming algorithms by controlling the Flash programming signals directly.

Row programing is recommended for gang programming applications and large volume customers who do not require in-circuit initial programming of the Flash memory. Page Erase operations are also supported when the Flash Controller is bypassed.

For more information on bypassing the Flash Controller, refer to *Third-Party Flash Programming Support for Z8 Encore! (AN0117)* available for download at www.zilog.com.

Flash Controller Behavior in DEBUG Mode

The following changes in behavior of the Flash Controller occur when the Flash Controller is accessed using the On-Chip Debugger:

- The Flash Write Protect option bit is ignored
- The Flash Sector Protect register is ignored for programming and erase operations
- Programming operations are not limited to the page selected in the Page Select register
- Bits in the Flash Sector Protect register can be written to one or zero
- The second write of the Page Select register to unlock the Flash Controller is not necessary
- The Page Select register can be written when the Flash Controller is unlocked
- The Mass Erase command is enabled through the Flash Control register

Caution: *For security reasons, Flash controller allows only a single page to be opened for write/erase. When writing multiple Flash pages, the Flash controller must go through the unlock sequence again to select another page.*

Table 83. Flash Frequency High Byte Register (FFREQH)

BITS	7	6	5	4	3	2	1	0
FIELD	FFREQH							
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	FFAH							

FFREQH—Flash Frequency High Byte
High byte of the 16-bit Flash Frequency value

Table 84. Flash Frequency Low Byte Register (FFREQL)

BITS	7	6	5	4	3	2	1	0
FIELD	FFREQL							
RESET	0							
R/W	R/W							
ADDR	FFBH							

FFREQL—Flash Frequency Low Byte
Low byte of the 16-bit Flash Frequency value

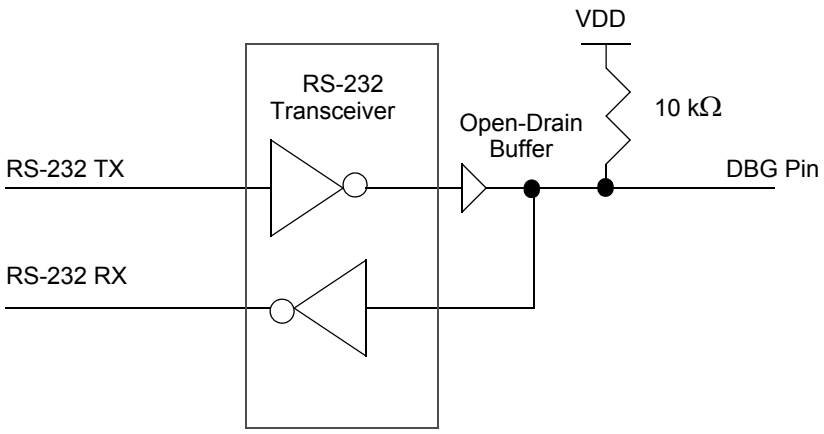


Figure 24. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface (2)

DEBUG Mode

The operating characteristics of the devices in DEBUG mode are:

- The eZ8 CPU fetch unit stops, idling the eZ8 CPU, unless directed by the OCD to execute specific instructions
- The system clock operates unless in STOP mode
- All enabled on-chip peripherals operate unless in STOP mode
- Automatically exits HALT mode
- Constantly refreshes the Watchdog Timer, if enabled.

Entering DEBUG Mode

The device enters DEBUG mode following the operations below:

- The device enters DEBUG mode after the eZ8 CPU executes a BRK (breakpoint) instruction
- If the DBG pin is held Low during the most recent clock cycle of System Reset, the part enters DEBUG mode upon exiting System Reset

Note: *Holding the DBG pin Low for an additional 5000 (minimum) clock cycles after reset (making sure to account for any specified frequency error if using an internal oscillator) prevents a false interpretation of an Autobaud sequence (see* OCD Auto-Baud Detector/ Generator *on page 154).*

- If the PA2/$\overline{\text{RESET}}$ pin is held Low while a 32-bit key sequence is issued to the PA0/DBG pin, the DBG feature is unlocked. After releasing PA2/$\overline{\text{RESET}}$, it is pulled high. At this

A reset and stop function can be achieved by writing 81H to this register. A reset and go function can be achieved by writing 41H to this register. If the device is in DEBUG mode, a run function can be implemented by writing 40H to this register.

Table 99. OCD Control Register (OCDCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	DBGMODE	BRKEN	DBGACK	Reserved				RST
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R/W

DBGMODE—DEBUG Mode
The device enters DEBUG mode when this bit is 1. When in DEBUG mode, the eZ8 CPU stops fetching new instructions. Clearing this bit causes the eZ8 CPU to restart. This bit is automatically set when a BRK instruction is decoded and breakpoints are enabled. If the Flash Read Protect Option Bit is enabled, this bit can only be cleared by resetting the device. It cannot be written to 0.
0 = Z8 Encore! XP F0823 Series device is operating in NORMAL mode
1 = Z8 Encore! XP F0823 Series device is in DEBUG mode

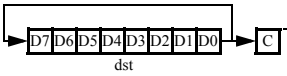
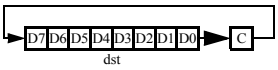
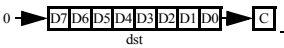
BRKEN—Breakpoint Enable
This bit controls the behavior of the BRK instruction (opcode 00H). By default, breakpoints are disabled and the BRK instruction behaves similar to an NOP instruction. If this bit is 1, when a BRK instruction is decoded, the DBGMODE bit of the OCDCTL register is automatically set to 1.
0 = Breakpoints are disabled
1 = Breakpoints are enabled

DBGACK—Debug Acknowledge
This bit enables the debug acknowledge feature. If this bit is set to 1, the OCD sends a Debug Acknowledge character (FFH) to the host when a Breakpoint occurs.
0 = Debug Acknowledge is disabled
1 = Debug Acknowledge is enabled

Reserved—0 when read

RST—Reset
Setting this bit to 1 resets the Z8F04xA family device. The device goes through a normal Power-On Reset sequence with the exception that the OCD is not reset. This bit is automatically cleared to 0 at the end of reset.
0 = No effect
1 = Reset the Flash Read Protect Option Bit device

Table 115. 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
RR dst		R		E0	*	*	*	*	–	–	2	2
		IR		E1							2	3
RRC dst		R		C0	*	*	*	*	–	–	2	2
		IR		C1							2	3
SBC dst, src	dst ← dst – src - C	r	r	32	*	*	*	*	1	*	2	3
		r	Ir	33							2	4
		R	R	34							3	3
		R	IR	35							3	4
		R	IM	36							3	3
		IR	IM	37							3	4
SBCX dst, src	dst ← dst – src - C	ER	ER	38	*	*	*	*	1	*	4	3
		ER	IM	39							4	3
SCF	C ← 1			DF	1	–	–	–	–	–	1	2
SRA dst		R		D0	*	*	*	0	–	–	2	2
		IR		D1							2	3
SRL dst		R		1F C0	*	*	0	*	–	–	3	2
		IR		1F C1							3	3
SRP src	RP ← src		IM	01	–	–	–	–	–	–	2	2
STOP	STOP Mode			6F	–	–	–	–	–	–	1	2
SUB dst, src	dst ← dst – src	r	r	22	*	*	*	*	1	*	2	3
		r	Ir	23							2	4
		R	R	24							3	3
		R	IR	25							3	4
		R	IM	26							3	3
		IR	IM	27							3	4

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

Upper Nibble (Hex) \ Lower Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	1.1 **BRK**	2.2 **SRP** IM	2.3 **ADD** r1,r2	2.4 **ADD** r1,Ir2	3.3 **ADD** R2,R1	3.4 **ADD** IR2,R1	3.3 **ADD** R1,IM	3.4 **ADD** IR1,IM	4.3 **ADDX** ER2,ER1	4.3 **ADDX** IM,ER1	2.3 **DJNZ** r1,X	2.2 **JR** cc,X	2.2 **LD** r1,IM	3.2 **JP** cc,DA	1.2 **INC** r1	1.2 **NOP**
1	2.2 **RLC** R1	2.3 **RLC** IR1	2.3 **ADC** r1,r2	2.4 **ADC** r1,Ir2	3.3 **ADC** R2,R1	3.4 **ADC** IR2,R1	3.3 **ADC** R1,IM	3.4 **ADC** IR1,IM	4.3 **ADCX** ER2,ER1	4.3 **ADCX** IM,ER1						See 2nd Opcode Map
2	2.2 **INC** R1	2.3 **INC** IR1	2.3 **SUB** r1,r2	2.4 **SUB** r1,Ir2	3.3 **SUB** R2,R1	3.4 **SUB** IR2,R1	3.3 **SUB** R1,IM	3.4 **SUB** IR1,IM	4.3 **SUBX** ER2,ER1	4.3 **SUBX** IM,ER1						1, 2 ATM
3	2.2 **DEC** R1	2.3 **DEC** IR1	2.3 **SBC** r1,r2	2.4 **SBC** r1,Ir2	3.3 **SBC** R2,R1	3.4 **SBC** IR2,R1	3.3 **SBC** R1,IM	3.4 **SBC** IR1,IM	4.3 **SBCX** ER2,ER1	4.3 **SBCX** IM,ER1						
4	2.2 **DA** R1	2.3 **DA** IR1	2.3 **OR** r1,r2	2.4 **OR** r1,Ir2	3.3 **OR** R2,R1	3.4 **OR** IR2,R1	3.3 **OR** R1,IM	3.4 **OR** IR1,IM	4.3 **ORX** ER2,ER1	4.3 **ORX** IM,ER1						
5	2.2 **POP** R1	2.3 **POP** IR1	2.3 **AND** r1,r2	2.4 **AND** r1,Ir2	3.3 **AND** R2,R1	3.4 **AND** IR2,R1	3.3 **AND** R1,IM	3.4 **AND** IR1,IM	4.3 **ANDX** ER2,ER1	4.3 **ANDX** IM,ER1						1.2 **WDT**
6	2.2 **COM** R1	2.3 **COM** IR1	2.3 **TCM** r1,r2	2.4 **TCM** r1,Ir2	3.3 **TCM** R2,R1	3.4 **TCM** IR2,R1	3.3 **TCM** R1,IM	3.4 **TCM** IR1,IM	4.3 **TCMX** ER2,ER1	4.3 **TCMX** IM,ER1						1.2 **STOP**
7	2.2 **PUSH** R2	2.3 **PUSH** IR2	2.3 **TM** r1,r2	2.4 **TM** r1,Ir2	3.3 **TM** R2,R1	3.4 **TM** IR2,R1	3.3 **TM** R1,IM	3.4 **TM** IR1,IM	4.3 **TMX** ER2,ER1	4.3 **TMX** IM,ER1						1.2 **HALT**
8	2.5 **DECW** RR1	2.6 **DECW** IRR1	2.5 **LDE** r1,Irr2	2.9 **LDEI** Ir1,Irr2	3.2 **LDX** r1,ER2	3.3 **LDX** Ir1,ER2	3.4 **LDX** IRR2,R1	3.5 **LDX** IRR2,IR1	3.4 **LDX** r1,rr2,X	3.4 **LDX** rr1,r2,X						1.2 **DI**
9	2.2 **RL** R1	2.3 **RL** IR1	2.5 **LDE** r2,Irr1	2.9 **LDEI** Ir2,Irr1	3.2 **LDX** r2,ER1	3.3 **LDX** Ir2,ER1	3.4 **LDX** R2,IRR1	3.5 **LDX** IR2,IRR1	3.3 **LEA** r1,r2,X	3.5 **LEA** rr1,rr2,X						1.2 **EI**
A	2.5 **INCW** RR1	2.6 **INCW** IRR1	2.3 **CP** r1,r2	2.4 **CP** r1,Ir2	3.3 **CP** R2,R1	3.4 **CP** IR2,R1	3.3 **CP** R1,IM	3.4 **CP** IR1,IM	4.3 **CPX** ER2,ER1	4.3 **CPX** IM,ER1						1.4 **RET**
B	2.2 **CLR** R1	2.3 **CLR** IR1	2.3 **XOR** r1,r2	2.4 **XOR** r1,Ir2	3.3 **XOR** R2,R1	3.4 **XOR** IR2,R1	3.3 **XOR** R1,IM	3.4 **XOR** IR1,IM	4.3 **XORX** ER2,ER1	4.3 **XORX** IM,ER1						1.5 **IRET**
C	2.2 **RRC** R1	2.3 **RRC** IR1	2.5 **LDC** r1,Irr2	2.9 **LDCI** Ir1,Irr2	2.3 **JP** IRR1	2.9 **LDC** Ir1,Irr2		3.4 **LD** r1,r2,X	3.2 **PUSHX** ER2							1.2 **RCF**
D	2.2 **SRA** R1	2.3 **SRA** IR1	2.5 **LDC** r2,Irr1	2.9 **LDCI** Ir2,Irr1	2.6 **CALL** IRR1	2.2 **BSWAP** R1	3.3 **CALL** DA	3.4 **LD** r2,r1,X	3.2 **POPX** ER1							1.2 **SCF**
E	2.2 **RR** R1	2.3 **RR** IR1	2.2 **BIT** p,b,r1	2.3 **LD** r1,Ir2	3.2 **LD** R2,R1	3.3 **LD** IR2,R1	3.2 **LD** R1,IM	3.3 **LD** IR1,IM	4.2 **LDX** ER2,ER1	4.2 **LDX** IM,ER1						1.2 **CCF**
F	2.2 **SWAP** R1	2.3 **SWAP** IR1	2.6 **TRAP** Vector	2.3 **LD** Ir1,r2	2.8 **MULT** RR1	3.3 **LD** R2,IR1	3.3 **BTJ** p,b,r1,X	3.4 **BTJ** p,b,Ir1,X								

Figure 27. First Opcode Map

Table 119. Power Consumption

Symbol	Parameter	V_{DD} = 2.7 V to 3.6 V			Units	Conditions
		Typical[1]	Maximum[2] Std Temp	Maximum[3] Ext Temp		
I_{DD} Stop	Supply Current in STOP Mode	0.1	2	7.5	µA	No peripherals enabled. All pins driven to V_{DD} or V_{SS}.
I_{DD} Halt	Supply Current in HALT Mode (with all peripherals disabled)	35	55	65	µA	32 kHz
		520	630	700	µA	5.5 MHz
I_{DD}	Supply Current in ACTIVE Mode (with all peripherals disabled)	2.8	4.5	4.8	mA	32 kHz
		4.5	5.2	5.2	mA	5.5 MHz
I_{DD} WDT	Watchdog Timer Supply Current	0.9	1.0	1.1	µA	
I_{DD} IPO	Internal Precision Oscillator Supply Current	350	500	550	µA	
I_{DD} VBO	Voltage Brownout Supply Current	50			µA	For 20-/28-pin devices (VBO only); see Note 4
						For 8-pin devices; See Note 4
I_{DD} ADC	Analog-to-Digital Converter Supply Current (with External Reference)	2.8	3.1	3.2	mA	32 kHz
		3.1	3.6	3.7	mA	5.5 MHz
		3.3	3.7	3.8	mA	10 MHz
		3.7	4.2	4.3	mA	20 MHz
I_{DD} ADCRef	ADC Internal Reference Supply Current	0			µA	See Note 4
I_{DD} CMP	Comparator supply Current	150	180	190	µA	See Note 4

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 8 KB Flash								
Standard Temperature: 0 °C to 70 °C								
Z8F0813PB005SC	8 KB	1 KB	6	12	2	0	1	PDIP 8-pin package
Z8F0813QB005SC	8 KB	1 KB	6	12	2	0	1	QFN 8-pin package
Z8F0813SB005SC	8 KB	1 KB	6	12	2	0	1	SOIC 8-pin package
Z8F0813SH005SC	8 KB	1 KB	16	18	2	0	1	SOIC 20-pin package
Z8F0813HH005SC	8 KB	1 KB	16	18	2	0	1	SSOP 20-pin package
Z8F0813PH005SC	8 KB	1 KB	16	18	2	0	1	PDIP 20-pin package
Z8F0813SJ005SC	8 KB	1 KB	24	18	2	0	1	SOIC 28-pin package
Z8F0813HJ005SC	8 KB	1 KB	24	18	2	0	1	SSOP 28-pin package
Z8F0813PJ005SC	8 KB	1 KB	24	18	2	0	1	PDIP 28-pin package
Extended Temperature: -40 °C to 105 °C								
Z8F0813PB005EC	8 KB	1 KB	6	12	2	0	1	PDIP 8-pin package
Z8F0813QB005EC	8 KB	1 KB	6	12	2	0	1	QFN 8-pin package
Z8F0813SB005EC	8 KB	1 KB	6	12	2	0	1	SOIC 8-pin package
Z8F0813SH005EC	8 KB	1 KB	16	18	2	0	1	SOIC 20-pin package
Z8F0813HH005EC	8 KB	1 KB	16	18	2	0	1	SSOP 20-pin package
Z8F0813PH005EC	8 KB	1 KB	16	18	2	0	1	PDIP 20-pin package
Z8F0813SJ005EC	8 KB	1 KB	24	18	2	0	1	SOIC 28-pin package
Z8F0813HJ005EC	8 KB	1 KB	24	18	2	0	1	SSOP 28-pin package
Z8F0813PJ005EC	8 KB	1 KB	24	18	2	0	1	PDIP 28-pin package
Replace C with G for Lead-Free Packaging								

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 1 KB Flash, 10-Bit Analog-to-Digital Converter								
Standard Temperature: 0 °C to 70 °C								
Z8F0123PB005SC	1 KB	256 B	6	12	2	4	1	PDIP 8-pin package
Z8F0123QB005SC	1 KB	256 B	6	12	2	4	1	QFN 8-pin package
Z8F0123SB005SC	1 KB	256 B	6	12	2	4	1	SOIC 8-pin package
Z8F0123SH005SC	1 KB	256 B	16	18	2	7	1	SOIC 20-pin package
Z8F0123HH005SC	1 KB	256 B	16	18	2	7	1	SSOP 20-pin package
Z8F0123PH005SC	1 KB	256 B	16	18	2	7	1	PDIP 20-pin package
Z8F0123SJ005SC	1 KB	256 B	22	18	2	8	1	SOIC 28-pin package
Z8F0123HJ005SC	1 KB	256 B	22	18	2	8	1	SSOP 28-pin package
Z8F0123PJ005SC	1 KB	256 B	22	18	2	8	1	PDIP 28-pin package
Extended Temperature: -40 °C to 105 °C								
Z8F0123PB005EC	1 KB	256 B	6	12	2	4	1	PDIP 8-pin package
Z8F0123QB005EC	1 KB	256 B	6	12	2	4	1	QFN 8-pin package
Z8F0123SB005EC	1 KB	256 B	6	12	2	4	1	SOIC 8-pin package
Z8F0123SH005EC	1 KB	256 B	16	18	2	7	1	SOIC 20-pin package
Z8F0123HH005EC	1 KB	256 B	16	18	2	7	1	SSOP 20-pin package
Z8F0123PH005EC	1 KB	256 B	16	18	2	7	1	PDIP 20-pin package
Z8F0123SJ005EC	1 KB	256 B	22	18	2	8	1	SOIC 28-pin package
Z8F0123HJ005EC	1 KB	256 B	22	18	2	8	1	SSOP 28-pin package
Z8F0123PJ005EC	1 KB	256 B	22	18	2	8	1	PDIP 28-pin package
Replace C with G for Lead-Free Packaging								

P

R