



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	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f4821an020sc00tr



Error Detection	135
SPI Interrupts	135
SPI Baud Rate Generator	136
SPI Control Register Definitions	137
SPI Data Register	137
SPI Control Register	137
SPI Status Register	139
SPI Mode Register	140
SPI Diagnostic State Register	141
SPI Baud Rate High and Low Byte Registers	142
I2C Controller	143
Overview	143
Architecture	144
Operation	144
SDA and SCL Signals	145
I ² C Interrupts	145
Software Control of I2C Transactions	146
Start and Stop Conditions	147
Master Write and Read Transactions	147
Address Only Transaction with a 7-bit Address	148
Write Transaction with a 7-Bit Address	149
Address Only Transaction with a 10-bit Address	150
Write Transaction with a 10-Bit Address	151
Read Transaction with a 7-Bit Address	153
Read Transaction with a 10-Bit Address	154
I2C Control Register Definitions	156
I2C Data Register	156
I2C Status Register	157
I2C Control Register	158
I2C Baud Rate High and Low Byte Registers	160
I2C Diagnostic State Register	161
I2C Diagnostic Control Register	163
Direct Memory Access Controller	165
Overview	165
Operation	165
DMA0 and DMA1 Operation	165
Configuring DMA0 and DMA1 for Data Transfer	166

Braces

The curly braces, { }, indicate a single register or bus created by concatenating some combination of smaller registers, buses, or individual bits.

- Example: The 12-bit register address {0H, RP[7:4], R1[3:0]} is composed of a 4-bit hexadecimal value (0H) and two 4-bit register values taken from the Register Pointer (RP) and Working Register R1. 0H is the most-significant nibble (4-bit value) of the 12-bit register, and R1[3:0] is the least significant nibble of the 12-bit register.

Parentheses

The parentheses, (), indicate an indirect register address lookup.

- Example: (R1) is the memory location referenced by the address contained in the Working Register R1.

Parentheses/Bracket Combinations

The parentheses, (), indicate an indirect register address lookup and the square brackets, [], indicate a register or bus.

- Example: Assume PC[15:0] contains the value 1234h. (PC[15:0]) then refers to the contents of the memory location at address 1234h.

Use of the Words *Set*, *Reset* and *Clear*

The word *set* implies that a register bit or a condition contains a logical 1. The words *reset* or *clear* imply that a register bit or a condition contains a logical 0. When either of these terms is followed by a number, the word *logical* may not be included; however, it is implied.

Notation for Bits and Similar Registers

A field of bits within a register is designated as: Register[n:n].

- Example: ADDR[15:0] refers to bits 15 through bit 0 of the Address.

Use of the Terms *LSB*, *MSB*, *lsb*, and *msb*

In this document, the terms *LSB* and *MSB*, when appearing in upper case, mean *least significant byte* and *most significant byte*, respectively. The lowercase forms, *lsb* and *msb*, mean *least significant bit* and *most significant bit*, respectively.

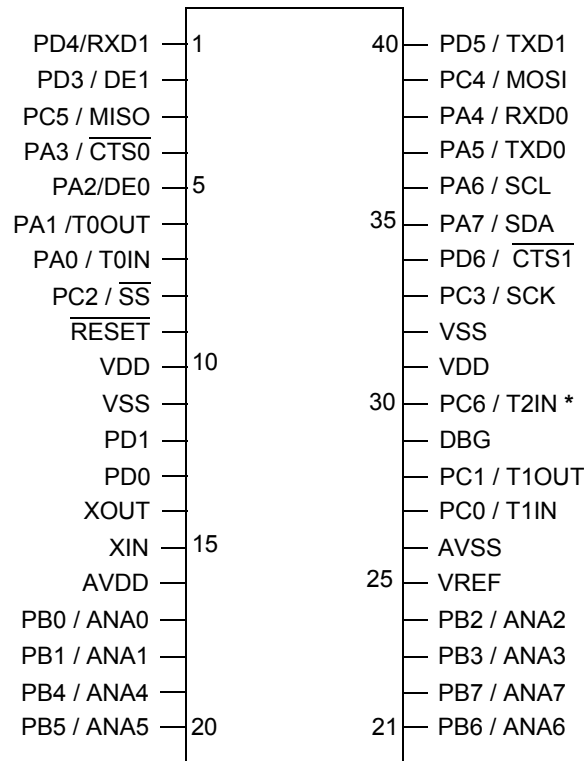
Use of Initial Uppercase Letters

Initial uppercase letters designate settings and conditions in general text.

- Example 1: The receiver forces the SCL line to Low.
- Example 2: The Master can generate a Stop condition to abort the transfer.

Pin Configurations

Figure 2 through Figure 7 on page 13 display the pin configurations for all of the packages available in the Z8 Encore! XP 64K Series Flash Microcontrollers. For description of the signals, see Table 3 on page 14. Timer 3 is not available in the 40-pin and 44-pin packages.



Note: Timer 3 is not supported.

* T2OUT is not supported.

Figure 2. Z8 Encore! XP 64K Series Flash Microcontrollers in 40-Pin Dual Inline Package (PDIP)

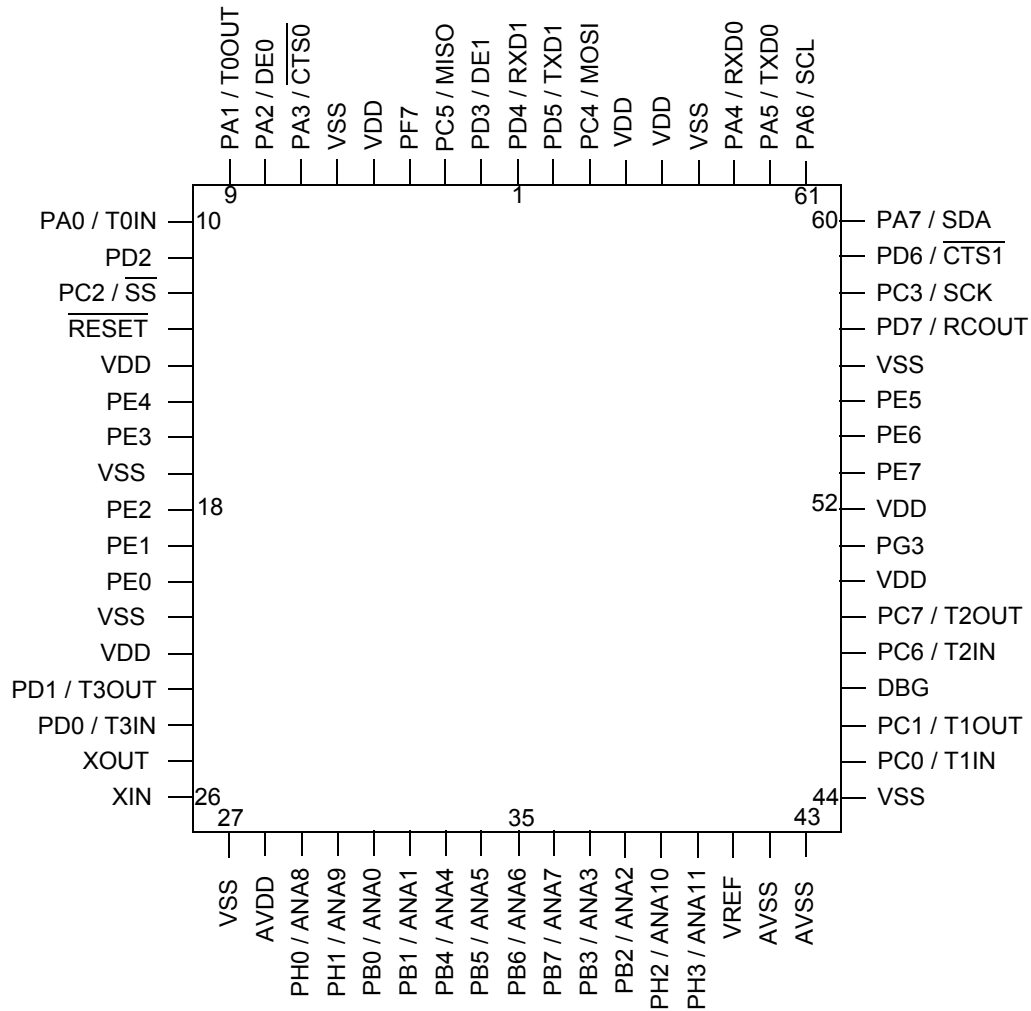


Figure 6. Z8 Encore! XP 64K Series Flash Microcontrollers in 68-Pin Plastic Leaded Chip Carrier (PLCC)

Signal Descriptions

Table 3 describes the Z8 Encore! XP signals. To determine the signals available for the specific package styles, see [Pin Configurations](#) on page 8.

Table 3. Signal Descriptions

Signal Mnemonic	I/O	Description
General-Purpose I/O Ports A-H		
PA[7:0]	I/O	Port A[7:0]. These pins are used for general-purpose I/O and support 5 V-tolerant inputs.
PB[7:0]	I/O	Port B[7:0]. These pins are used for general-purpose I/O.
PC[7:0]	I/O	Port C[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5 V-tolerant inputs
PD[7:0]	I/O	Port D[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5 V-tolerant inputs
PE[7:0]	I/O	Port E[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5 V-tolerant inputs.
PF[7:0]	I/O	Port F[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5 V-tolerant inputs.
PG[7:0]	I/O	Port G[7:0]. These pins are used for general-purpose I/O. These pins are used for general-purpose I/O and support 5 V-tolerant inputs.
PH[3:0]	I/O	Port H[3:0]. These pins are used for general-purpose I/O.
I²C Controller		
SCL	O	Serial Clock. This is the output clock for the I ² C. This pin is multiplexed with a general-purpose I/O pin. When the general-purpose I/O pin is configured for alternate function to enable the SCL function, this pin is open-drain.
SDA	I/O	Serial Data. This open-drain pin transfers data between the I ² C and a slave. This pin is multiplexed with a general-purpose I/O pin. When the general-purpose I/O pin is configured for alternate function to enable the SDA function, this pin is open-drain.
SPI Controller		
SS	I/O	Slave Select. This signal can be an output or an input. If the Z8 Encore! XP 64K Series Flash Microcontrollers is the SPI master, this pin may be configured as the Slave Select output. If the Z8 Encore! XP 64K Series Flash Microcontrollers is the SPI slave, this pin is the input slave select. It is multiplexed with a general-purpose I/O pin.

UART0 Control 1

U0CTL1 (F43H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- └ Infrared Encoder/Decoder Enable
0 = Infrared endec is disabled
1 = Infrared endec is enabled
- └ Received Data Interrupt Enable
0 = Received data and errors
generate interrupt requests
1 = Only errors generate interrupt
requests. Received data does
not.
- └ Baud Rate Registers Control
Refer to UART chapter for operation
- └ Driver Enable Polarity
0 = DE signal is active High
1 = DE signal is active Low
- └ Multiprocessor Bit Transmit
0 = Send a 0 as the multiprocessor
bit
1 = Send a 1 as the multiprocessor
bit
- └ Multiprocessor Mode [0]
See Multiprocessor Mode [1] below
- └ Multiprocessor (9-bit) Enable
0 = Multiprocessor mode is disabled
1 = Multiprocessor mode is enabled
- └ Multiprocessor Mode [1]
with Multiprocess Mode bit 0:
00 = Interrupt on all received bytes
01 = Interrupt only on address bytes
10 = Interrupt on address match and
following data
11 = Interrupt on data following an
address match

UART0 Status 1

U0STAT1 (F44H - Read Only)

D7 D6 D5 D4 D3 D2 D1 D0

- └ Multitprocessor Receive
Returns value of last multiprocessor
bit
- └ New Frame
0 = Current byte is not start of frame
1 = Current byte is start of new
frame
- └ Reserved

UART0 Address Compare

U0ADDR (F45H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

UART0 Address Compare [7:0]

UART0 Baud Rate Generator High Byte

U0BRH (F46H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

UART0 Baud Rate divisor [15:8]

UART0 Baud Rate Generator Low Byte

U0BRL (F47H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

UART0 Baud Rate divisor [7:0]

UART1 Transmit Data

U1TXD (F48H - Write Only)

D7 D6 D5 D4 D3 D2 D1 D0

UART1 transmitter data byte[7:0]

UART1 Receive Data

U1RXD (F48H - Read Only)

D7 D6 D5 D4 D3 D2 D1 D0

UART receiver data byte [7:0]

General-Purpose I/O

Overview

The 64K Series products support a maximum of seven 8-bit ports (Ports A–G) and one 4-bit port (Port H) for general-purpose input/output (GPIO) operations. Each port consists of control and data registers. The GPIO control registers are used to determine data direction, open-drain, output drive current and alternate pin functions. Each port pin is individually programmable. All ports (except B and H) support 5 V-tolerant inputs.

GPIO Port Availability By Device

[Table 11](#) lists the port pins available with each device and package type.

Table 11. Port Availability by Device and Package Type

Device	Packages	Port A	Port B	Port C	Port D	Port E	Port F	Port G	Port H
Z8X1621	40-pin	[7:0]	[7:0]	[6:0]	[6:3, 1:0]	–	–	–	–
Z8X1621	44-pin	[7:0]	[7:0]	[7:0]	[6:0]	–	–	–	–
Z8X1622	64- and 68-pin	[7:0]	[7:0]	[7:0]	[7:0]	[7:0]	[7]	[3]	[3:0]
Z8X2421	40-pin	[7:0]	[7:0]	[6:0]	[6:3, 1:0]	–	–	–	–
Z8X2421	44-pin	[7:0]	[7:0]	[7:0]	[6:0]	–	–	–	–
Z8X2422	64- and 68-pin	[7:0]	[7:0]	[7:0]	[7:0]	[7:0]	[7]	[3]	[3:0]
Z8X3221	40-pin	[7:0]	[7:0]	[6:0]	[6:3, 1:0]	–	–	–	–
Z8X3221	44-pin	[7:0]	[7:0]	[7:0]	[6:0]	–	–	–	–
Z8X3222	64- and 68-pin	[7:0]	[7:0]	[7:0]	[7:0]	[7:0]	[7]	[3]	[3:0]
Z8X4821	40-pin	[7:0]	[7:0]	[6:0]	[6:3, 1:0]	–	–	–	–
Z8X4821	44-pin	[7:0]	[7:0]	[7:0]	[6:0]	–	–	–	–
Z8X4822	64- and 68-pin	[7:0]	[7:0]	[7:0]	[7:0]	[7:0]	[7]	[3]	[3:0]

Port A–H Data Direction Sub-Registers

The Port A–H Data Direction sub-register is accessed through the Port A–H Control register by writing 01H to the Port A–H Address register ([Table 16](#)).

Table 16. Port A–H Data Direction Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
RESET	1							
R/W	R/W							
ADDR	If 01H in Port A–H Address Register, accessible through Port A–H 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–H Output Data register is driven onto the port pin.

1 = Input. The port pin is sampled and the value written into the Port A–H Input Data Register. The output driver is tri-stated.

Port A–H Alternate Function Sub-Registers

The Port A–H Alternate Function sub-register ([Table 17](#)) is accessed through the Port A–H Control register by writing 02H to the Port A–H Address register. The Port A–H Alternate Function sub-registers select the alternate functions for the selected pins. To determine the alternate function associated with each port pin, see [GPIO Alternate Functions](#) on page 59.



Caution: Do not enable alternate function for GPIO port pins which do not have an associated alternate function. Failure to follow this guideline may result in unpredictable operation.

Table 17. Port A–H Alternate Function Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	AF7	AF6	AF5	AF4	AF3	AF2	AF1	AF0
RESET	0							
R/W	R/W							
ADDR	If 02H in Port A–H Address Register, accessible through Port A–H Control Register							

- Executing a Trap instruction.
- Illegal Instruction trap.

Interrupt Vectors and Priority

The interrupt controller supports three levels of interrupt priority. Level 3 is the highest priority, Level 2 is the second highest priority, and Level 1 is the lowest priority. If all of the interrupts were enabled with identical interrupt priority (all as Level 2 interrupts, for example), then interrupt priority would be assigned from highest to lowest as specified in [Table 23](#) on page 68. Level 3 interrupts always have higher priority than Level 2 interrupts which, in turn, always have higher priority than Level 1 interrupts. Within each interrupt priority level (Level 1, Level 2, or Level 3), priority is assigned as specified in [Table 23](#) on page 68. Reset, Watchdog Timer interrupt (if enabled), and Illegal Instruction Trap always have highest priority.

Interrupt Assertion

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



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

Poor coding style that can result in lost interrupt requests:

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

To avoid missing interrupts, the following style of coding to clear bits in the Interrupt Request 0 register is recommended:

Good coding style that avoids lost interrupt requests:

```
ANDX IRQ0, MASK
```

Software Interrupt Assertion

Program code can generate interrupts directly. Writing a 1 to the desired 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.

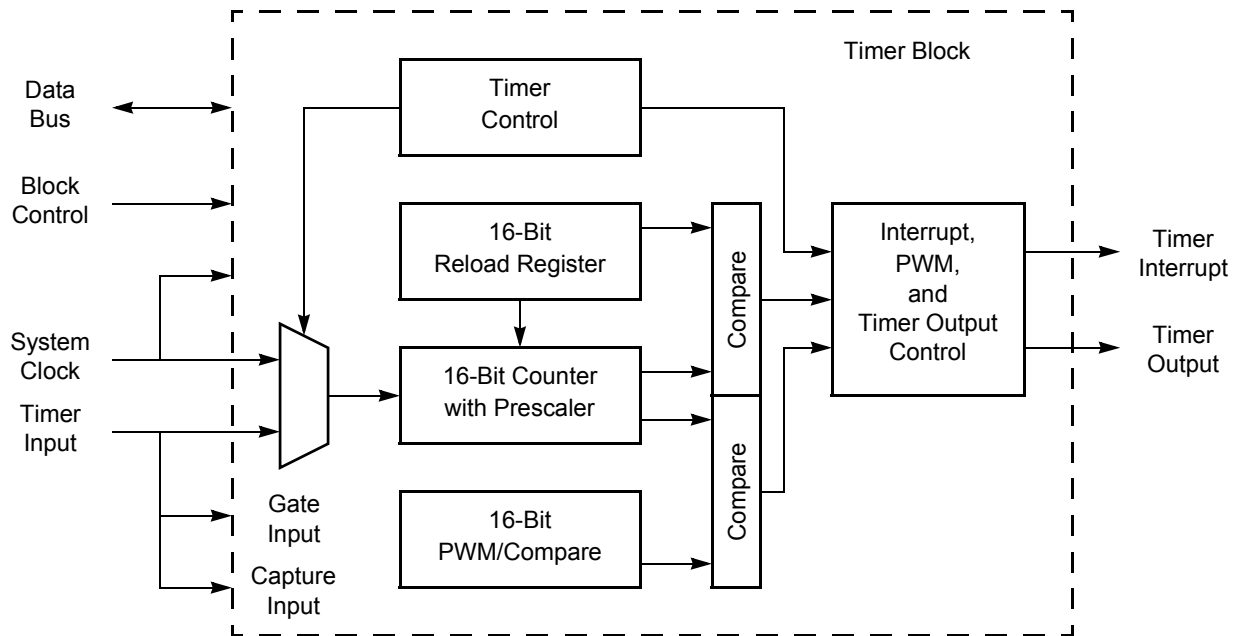


Figure 12. Timer Block Diagram

Operation

The timers are 16-bit up-counters. Minimum time-out delay is set by loading the value 0001H into the Timer Reload High and Low Byte registers and setting the prescale value to 1. Maximum time-out delay is set by loading the value 0000H into the Timer Reload High and Low Byte registers and setting the prescale value to 128. If the Timer reaches FFFFH, the timer rolls over to 0000H and continues counting.

Timer Operating Modes

The timers can be configured to operate in the following modes:

ONE-SHOT Mode

In ONE-SHOT mode, the timer counts up to the 16-bit Reload value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the Reload value, the timer generates an interrupt and the count value in the Timer High and Low Byte registers is reset to 0001H. Then, the timer is automatically disabled and stops counting.

Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state for one system clock cycle (from Low to High or from High to Low) upon timer Reload. If it is desired to have the Timer Output make a permanent state change upon

2. Write to the Timer High and Low Byte registers to set the starting count value (usually 0001H), affecting only the first pass in CONTINUOUS mode. After the first timer Reload in CONTINUOUS mode, counting always begins at the reset value of 0001H.
3. Write to the Timer Reload High and Low Byte registers to set the Reload value.
4. If desired, 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 1 register to enable the timer and initiate counting.

In CONTINUOUS mode, the system clock always provides the timer input. The timer period is given by the following equation:

$$\text{CONTINUOUS Mode Time-Out Period (s)} = \frac{\text{Reload Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

If an initial starting value other than 0001H is loaded into the Timer High and Low Byte registers, the ONE-SHOT mode equation must be used to determine the first time-out period.

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 1 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 1 register to:
 - Disable the timer
 - Configure the timer for COUNTER mode

Table 39. Timer 0-3 High Byte Register (TxH)

BITS	7	6	5	4	3	2	1	0
FIELD	TH							
RESET	0							
R/W	R/W							
ADDR	F00H, F08H, F10H, F18H							

Table 40. Timer 0-3 Low Byte Register (TxL)

BITS	7	6	5	4	3	2	1	0
FIELD	TL							
RESET	0							1
R/W	R/W							
ADDR	F01H, F09H, F11H, F19H							

TH and TL—Timer High and Low Bytes

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

Timer Reload High and Low Byte Registers

The Timer 0-3 Reload High and Low Byte (TxRH and TxRL) registers (see [Table 41](#) and [Table 42](#) on page 92) 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.

In COMPARE mode, the Timer Reload High and Low Byte registers store the 16-bit Compare value.

Table 41. Timer 0-3 Reload High Byte Register (TxRH)

BITS	7	6	5	4	3	2	1	0
FIELD	TRH							
RESET	1							
R/W	R/W							
ADDR	F02H, F0AH, F12H, F1AH							

Table 47. Watchdog Timer Approximate Time-Out Delays

WDT Reload Value (Hex)	WDT Reload Value (Decimal)	Approximate Time-Out Delay (with 10 kHz typical WDT oscillator frequency)	
		Typical	Description
000004	4	400 μ s	Minimum time-out delay
FFFFFF	16,777,215	1677.5 s	Maximum time-out delay

Watchdog Timer Refresh

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

When the 64K Series devices are operating in DEBUG Mode (through the On-Chip Debugger), the Watchdog Timer is continuously refreshed to prevent spurious 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 Reset. The WDT_RES Option Bit determines the time-out response of the Watchdog Timer. For information on programming of the WDT_RES Option Bit, see [Option Bits](#) on page 195.

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 FFFFFFFH and continues counting. The Watchdog Timer counter is not automatically returned to its Reload Value.

WDT Interrupt in STOP Mode

If configured to generate an interrupt when a time-out occurs and the 64K Series devices 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 WDT time-out in STOP mode. For more information on Stop Mode Recovery, see [Reset and Stop Mode Recovery](#) on page 47.

ADC Control Register Definitions

ADC Control Register

The ADC Control register selects the analog input channel and initiates the analog-to-digital conversion.

Table 86. ADC Control Register (ADCCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	CEN	Reserved	VREF	CONT	ANAIN[3:0]			
RESET	0		1	0				
R/W	R/W							
ADDR	F70H							

CEN—Conversion Enable

0 = Conversion is complete. Writing a 0 produces no effect. The ADC automatically clears this bit to 0 when a conversion has been completed.

1 = Begin conversion. Writing a 1 to this bit starts a conversion. If a conversion is already in progress, the conversion restarts. This bit remains 1 until the conversion is complete.

Reserved—Must be 0.

$\overline{\text{VREF}}$

0 = Internal voltage reference generator enabled. The VREF pin should be left unconnected (or capacitively coupled to analog ground) if the internal voltage reference is selected as the ADC reference voltage.

1 = Internal voltage reference generator disabled. An external voltage reference must be provided through the VREF pin.

CONT

0 = Single-shot conversion. ADC data is output once at completion of the 5129 system clock cycles.

1 = Continuous conversion. ADC data updated every 256 system clock cycles.

ANAIN—Analog Input Select

These bits select the analog input for conversion. Not all Port pins in this list are available in all packages for the Z8F642x family Z8R642x family of products. For information on the Port pins available with each package style, see [Signal and Pin Descriptions](#) on page 7. Do not enable unavailable analog inputs.

0000 = ANA0

0001 = ANA1

0010 = ANA2

0011 = ANA3



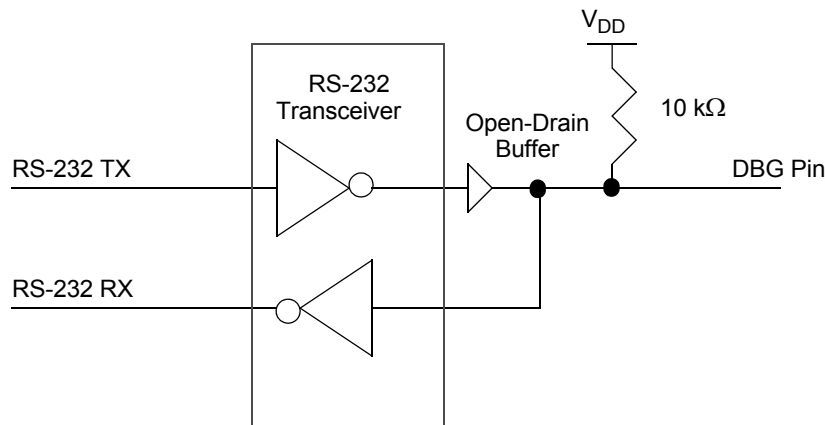


Figure 38. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface (2)

DEBUG Mode

The operating characteristics of the 64K Series 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 any of the following operations:

- Writing the DBGMODE bit in the OCD Control Register to 1 using the OCD interface.
- eZ8 CPU execution of a BRK (Breakpoint) instruction (when enabled).
- If the DBG pin is Low when the device exits Reset, the On-Chip Debugger automatically puts the device into DEBUG mode.

Exiting DEBUG Mode

The device exits DEBUG mode following any of the following operations:

- Clearing the DBGMODE bit in the OCD Control Register to 0.
- Power-On Reset
- Voltage Brownout reset

OCD Serial Errors

The On-Chip Debugger can detect any of the following error conditions on the DBG pin:

- Serial Break (a minimum of nine continuous bits Low).
- Framing Error (received Stop bit is Low).
- Transmit Collision (OCD and host simultaneous transmission detected by the OCD).

When the OCD detects one of these errors, it aborts any command currently in progress, transmits a Serial Break 4096 system clock cycles long back to the host, and resets the Auto-Baud Detector/Generator. A Framing Error or Transmit Collision may be caused by the host sending a Serial Break to the OCD. Because of the open-drain nature of the interface, returning a Serial Break back to the host only extends the length of the Serial Break if the host releases the Serial Break early.

The host transmits a Serial Break on the DBG pin when first connecting to the 64K Series devices or when recovering from an error. A Serial Break from the host resets the Auto-Baud Generator/Detector but does not reset the OCD Control register. A Serial Break leaves the device in DEBUG mode if that is the current mode. The OCD is held in Reset until the end of the Serial Break when the DBG pin returns High. Because of the open-drain nature of the DBG pin, the host can send a Serial Break to the OCD even if the OCD is transmitting a character.

Breakpoints

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

If breakpoints are enabled, the OCD can be configured to automatically enter DEBUG mode, or to loop on the break instruction. If the OCD is configured to loop on the BRK instruction, then the CPU is still enabled to service DMA and interrupt requests.

The loop on BRK instruction can be used to service interrupts in the background. For interrupts to be serviced in the background, there cannot be any breakpoints in the interrupt service routine. Otherwise, the CPU stops on the breakpoint in the interrupt routine. For interrupts to be serviced in the background, interrupts must also be enabled. Debugging software should not automatically enable interrupts when using this feature, since interrupts are typically disabled during critical sections of code where interrupts should not occur (such as adjusting the stack pointer or modifying shared data).

Software can poll the IDLE bit of the OCDSTAT register to determine if the OCD is looping on a BRK instruction. When software wants to stop the CPU on the BRK instruction it is looping on, software should not set the DBGMODE bit of the OCDCTL register. The CPU may have vectored to and be in the middle of an interrupt service routine when this bit gets set. Instead, software must clear the BRKLP bit. This action allows the CPU to

must add no more than 4 pF of stray capacitance to either the X_{IN} or X_{OUT} pins. If oscillation does not occur, reduce the values of capacitors C1 and C2 to decrease loading.

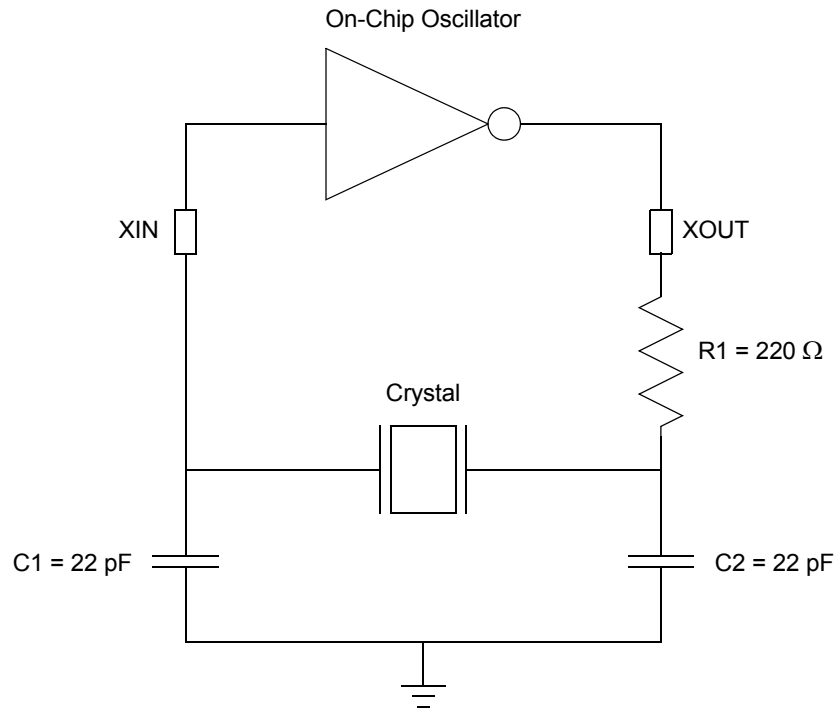


Figure 40. Recommended 20 MHz Crystal Oscillator Configuration

Table 104. Recommended Crystal Oscillator Specifications (20 MHz Operation)

Parameter	Value	Units	Comments
Frequency	20	MHz	
Resonance	Parallel		
Mode	Fundamental		
Series Resistance (R _S)	25	Ω	Maximum
Load Capacitance (C _L)	20	pF	Maximum
Shunt Capacitance (C ₀)	7	pF	Maximum
Drive Level	1	mW	Maximum

Ordering Information

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	I ² C	SPI	UARTs with IrDA	Description
Z8F642x with 64 KB Flash, 10-Bit Analog-to-Digital Converter										
Standard Temperature: 0 °C to 70 °C										
Z8F6421PM020SC	64 KB	4 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F6421AN020SC	64 KB	4 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F6421VN020SC	64 KB	4 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F6422AR020SC	64 KB	4 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F6422VS020SC	64 KB	4 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F6423FT020SC	64 KB	4 KB	60	24	4	12	1	1	2	QFP 80-pin package
Extended Temperature: –40 °C to +105 °C										
Z8F6421PM020EC	64 KB	4 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F6421AN020EC	64 KB	4 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F6421VN020EC	64 KB	4 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F6422AR020EC	64 KB	4 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F6422VS020EC	64 KB	4 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F6423FT020EC	64 KB	4 KB	60	24	4	12	1	1	2	QFP 80-pin package
Automotive/Industrial Temperature: –40 °C to +125 °C										
Z8F6421PM020AC	64 KB	4 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F6421AN020AC	64 KB	4 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F6421VN020AC	64 KB	4 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F6422AR020AC	64 KB	4 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F6422VS020AC	64 KB	4 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F6423FT020AC	64 KB	4 KB	60	24	4	12	1	1	2	QFP 80-pin package



Example: Part number Z8F6421AN020SC is an 8-bit microcontroller product in an LQFP package, using 44 pins, operating with a maximum 20 MHz external clock frequency over a 0 °C to +70 °C temperature range and built using the Plastic-Standard environmental flow.