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	20MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, Temp Sensor, WDT
Number of I/O	23
Program Memory Size	1KB (1K x 8)
Program Memory Type	FLASH
EEPROM Size	16 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SSOP (0.173", 4.40mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f012ahj020eg

Table 7. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page
F85	Reserved	—	XX	
Oscillator Control				
F86	Oscillator Control	OSCCTL	A0	196
F87–F8F	Reserved	—	XX	
Comparator 0				
F90	Comparator 0 Control	CMP0	14	141
F91–FBF	Reserved	—	XX	
Interrupt Controller				
FC0	Interrupt Request 0	IRQ0	00	60
FC1	IRQ0 Enable High Bit	IRQ0ENH	00	63
FC2	IRQ0 Enable Low Bit	IRQ0ENL	00	63
FC3	Interrupt Request 1	IRQ1	00	61
FC4	IRQ1 Enable High Bit	IRQ1ENH	00	65
FC5	IRQ1 Enable Low Bit	IRQ1ENL	00	65
FC6	Interrupt Request 2	IRQ2	00	62
FC7	IRQ2 Enable High Bit	IRQ2ENH	00	66
FC8	IRQ2 Enable Low Bit	IRQ2ENL	00	67
FC9–FCC	Reserved	—	XX	
FCD	Interrupt Edge Select	IRQES	00	68
FCE	Shared Interrupt Select	IRQSS	00	68
FCF	Interrupt Control	IRQCTL	00	69
GPIO Port A				
FD0	Port A Address	PAADDR	00	44
FD1	Port A Control	PACTL	00	46
FD2	Port A Input Data	PAIN	XX	46
FD3	Port A Output Data	PAOUT	00	46
GPIO Port B				
FD4	Port B Address	PBADDR	00	44
FD5	Port B Control	PBCTL	00	46
FD6	Port B Input Data	PBIN	XX	46
FD7	Port B Output Data	PBOUT	00	46
GPIO Port C				
FD8	Port C Address	PCADDR	00	44

Notes:
1. XX = Undefined.
2. Refer to the eZ8 CPU Core User Manual (UM0128).

Table 7. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page
FD9	Port C Control	PCCTL	00	46
FDA	Port C Input Data	PCIN	XX	46
FDB	Port C Output Data	PCOUT	00	46
GPIO Port D				
FDC	Port D Address	PDADDR	00	44
FDD	Port D Control	PDCTL	00	46
FDE	Reserved	—	XX	
FDF	Port D Output Data	PDOUT	00	46
FE0–FEF	Reserved	—	XX	
Watchdog Timer (WDT)				
FF0	Reset Status (Read-only)	RSTSTAT	X0	29
	Watchdog Timer Control (Write-only)	WDTCTL	N/A	96
FF1	Watchdog Timer Reload Upper Byte	WDTU	00	97
FF2	Watchdog Timer Reload High Byte	WDTH	04	97
FF3	Watchdog Timer Reload Low Byte	WDTL	00	98
FF4–FF5	Reserved	—	XX	
Trim Bit Control				
FF6	Trim Bit Address	TRMADR	00	161
FF7	Trim Bit Data	TRMDR	00	162
Flash Memory Controller				
FF8	Flash Control	FCTL	00	155
FF8	Flash Status	FSTAT	00	155
FF9	Flash Page Select	FPS	00	156
	Flash Sector Protect	FPROT	00	157
FFA	Flash Programming Frequency High Byte	FFREQH	00	158
FFB	Flash Programming Frequency Low Byte	FFREQL	00	158
eZ8 CPU				
FFC	Flags	—	XX	See footnote 2.
FFD	Register Pointer	RP	XX	
FFE	Stack Pointer High Byte	SPH	XX	
FFF	Stack Pointer Low Byte	SPL	XX	

Notes:
1. XX = Undefined.
2. Refer to the eZ8 CPU Core User Manual (UM0128).

General-Purpose Input/Output

The Z8 Encore! XP F082A Series products support a maximum of 25 port pins (Ports A–D) for general-purpose input/output (GPIO) operations. Each port contains control and data registers. The GPIO control registers determine data direction, open-drain, output drive current, programmable pull-ups, Stop Mode Recovery functionality and alternate pin functions. Each port pin is individually programmable. In addition, the Port C pins are capable of direct LED drive at programmable drive strengths.

GPIO Port Availability By Device

Table 14 lists the port pins available with each device and package type.

Table 14. Port Availability by Device and Package Type

Devices	Package	ADC	Port A	Port B	Port C	Port D	Total I/O
Z8F082ASB, Z8F082APB, Z8F082AQB Z8F042ASB, Z8F042APB, Z8F042AQB Z8F022ASB, Z8F022APB, Z8F022AQB Z8F012ASB, Z8F012APB, Z8F012AQB	8-pin	Yes	[5:0]	No	No	No	6
Z8F081ASB, Z8F081APB, Z8F081AQB Z8F041ASB, Z8F041APB, Z8F041AQB Z8F021ASB, Z8F021APB, Z8F021AQB Z8F011ASB, Z8F011APB, Z8F011AQB	8-pin	No	[5:0]	No	No	No	6
Z8F082APH, Z8F082AHH, Z8F082ASH Z8F042APH, Z8F042AHH, Z8F042ASH Z8F022APH, Z8F022AHH, Z8F022ASH Z8F012APH, Z8F012AHH, Z8F012ASH	20-pin	Yes	[7:0]	[3:0]	[3:0]	[0]	17
Z8F081APH, Z8F081AHH, Z8F081ASH Z8F041APH, Z8F041AHH, Z8F041ASH Z8F021APH, Z8F021AHH, Z8F021ASH Z8F011APH, Z8F011AHH, Z8F011ASH	20-pin	No	[7:0]	[3:0]	[3:0]	[0]	17
Z8F082APJ, Z8F082ASJ, Z8F082AHJ Z8F042APJ, Z8F042ASJ, Z8F042AHJ Z8F022APJ, Z8F022ASJ, Z8F022AHJ Z8F012APJ, Z8F012ASJ, Z8F012AHJ	28-pin	Yes	[7:0]	[5:0]	[7:0]	[0]	23
Z8F081APJ, Z8F081ASJ, Z8F081AHJ Z8F041APJ, Z8F041ASJ, Z8F041AHJ Z8F021APJ, Z8F021ASJ, Z8F021AHJ Z8F011APJ, Z8F011ASJ, Z8F011AHJ	28-pin	No	[7:0]	[7:0]	[7:0]	[0]	25

Port A–D Pull-up Enable Subregisters

The Port A–D Pull-up Enable Subregister, shown in Table 26, is accessed through the Port A–D Control Register by writing 06H to the Port A–D Address Register. Setting the bits in the Port A–D Pull-up Enable subregisters enables a weak internal resistive pull-up on the specified port pins.

Table 26. Port A–D Pull-Up Enable Subregisters (PxPUE)

Bit	7	6	5	4	3	2	1	0
Field	PPUE7	PPUE6	PPUE5	PPUE4	PPUE3	PPUE2	PPUE1	PPUE0
RESET	00H (Ports A-C); 01H (Port D); 04H (Port A of 8-pin device)							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	If 06H in Port A–D Address Register, accessible through the Port A–D Control Register							

Bit	Description
[7:0] PPUEx	**Port Pull-up Enabled** 0 = The weak pull-up on the port pin is disabled. 1 = The weak pull-up on the port pin is enabled.

Note: x indicates the specific GPIO port pin number (7–0).

Port A–D Alternate Function Set 1 Subregisters

The Port A–D Alternate Function Set1 Subregister, shown in Table 27, is accessed through the Port A–D Control Register by writing 07H to the Port A–D Address Register. The Alternate Function Set 1 subregisters selects the alternate function available at a port pin. Alternate Functions selected by setting or clearing bits of this register are defined in the GPIO Alternate Functions section on page 37.

Note: Alternate function selection on port pins must also be enabled as described in the Port A–D Alternate Function Subregisters section on page 47.

Table 27. Port A–D Alternate Function Set 1 Subregisters (PxAFS1)

Bit	7	6	5	4	3	2	1	0
Field	PAFS17	PAFS16	PAFS15	PAFS14	PAFS13	PAFS12	PAFS11	PAFS10
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	If 07H in Port A–D Address Register, accessible through the Port A–D Control Register							

Bit	Description
[7:0] PAFSx	**Port Alternate Function Set 1** 0 = Port Alternate Function selected, as defined in Tables 15 and 16 on page 43. 1 = Port Alternate Function selected, as defined in Tables 15 and 16 on page 43.

Note: x indicates the specific GPIO port pin number (7–0).

Port A–D Alternate Function Set 2 Subregisters

The Port A–D Alternate Function Set 2 Subregister, shown in Table 28, is accessed through the Port A–D Control Register by writing `08H` to the Port A–D Address Register. The Alternate Function Set 2 subregisters selects the alternate function available at a port pin. Alternate Functions selected by setting or clearing bits of this register is defined in Table 16 on page 43.

Note: Alternate function selection on the port pins must also be enabled. See the Port A–D Alternate Function Subregisters section on page 47 for details.

Table 28. Port A–D Alternate Function Set 2 Subregisters (PxAFS2)

Bit	7	6	5	4	3	2	1	0
Field	PAFS27	PAFS26	PAFS25	PAFS24	PAFS23	PAFS22	PAFS21	PAFS20
RESET	00H (all ports of 20/28 pin devices); 04H (Port A of 8-pin device)							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	If 08H in Port A–D Address Register, accessible through the Port A–D Control Register							

Bit	Description
[7] PAFS2x	**Port Alternate Function Set 2** 0 = Port Alternate Function selected, as defined in Table 16. 1 = Port Alternate Function selected, as defined in Table 16.

Note: x indicates the specific GPIO port pin number (7–0).

Timers

These Z8 Encore! XP F082A Series products contain two 16-bit reloadable timers that can be used for timing, event counting, or generation of pulse-width modulated (PWM) signals. The timers' feature 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) may also provide basic timing functionality. For information about using the Baud Rate Generator as an additional timer, see the Universal Asynchronous Receiver/ Transmitter chapter on page 99.

Architecture

Figure 9 displays the architecture of the timers.

3. Write to the Timer Reload High and Low Byte registers to set the reload value.
4. Clear the Timer PWM High and Low Byte registers to 0000H. This allows the software to determine if interrupts were generated by either a capture event or a reload. If the PWM High and Low Byte registers still contain 0000H after the interrupt, the interrupt was generated by a Reload.
5. Enable the timer interrupt, if appropriate and set the timer interrupt priority by writing to the relevant interrupt registers. By default, the timer interrupt is generated for both input capture and reload events. If appropriate, configure the timer interrupt to be generated only at the input capture event or the reload event by setting TICONFIG field of the TxCTL0 Register.
6. Configure the associated GPIO port pin for the Timer Input alternate function.
7. Write to the Timer Control Register to enable the timer and initiate counting.

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

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

COMPARE Mode

In COMPARE Mode, the timer counts up to the 16-bit maximum Compare value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the Compare value, the timer generates an interrupt and counting continues (the timer value is not reset to 0001H). Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) upon Compare.

If the Timer reaches FFFFH, the timer rolls over to 0000H and continue counting.

Observe the following steps for configuring a timer for COMPARE Mode and initiating the count:

1. Write to the Timer Control Register to:
   - Disable the timer
   - Configure the timer for COMPARE Mode
   - Set the prescale value

Bit	Description (Continued)
[6:5] TICONFIG	**Timer Interrupt Configuration** This field configures timer interrupt definition. 0x = Timer Interrupt occurs on all defined Reload, Compare and Input Events. 10 = Timer Interrupt only on defined Input Capture/Deassertion Events. 11 = Timer Interrupt only on defined Reload/Compare Events.
[4]	**Reserved** This bit is reserved and must be programmed to 0.
[3:1] PWMD	**PWM Delay Value** This field is a programmable delay to control the number of system clock cycles delay before the Timer Output and the Timer Output Complement are forced to their active state. 000 = No delay. 001 = 2 cycles delay. 010 = 4 cycles delay. 011 = 8 cycles delay. 100 = 16 cycles delay. 101 = 32 cycles delay. 110 = 64 cycles delay. 111 = 128 cycles delay.
[0] INPCAP	**Input Capture Event** This bit indicates if the most recent timer interrupt is caused by a Timer Input Capture Event. 0 = Previous timer interrupt is not a result of Timer Input Capture Event. 1 = Previous timer interrupt is a result of Timer Input Capture Event.

Timer 0–1 Control Register 1

The Timer 0–1 Control (TxCTL1) registers, shown in Table 51, enable and disable the timers, set the prescaler value and determine the timer operating mode.

Table 51. Timer 0–1 Control Register 1 (TxCTL1)

Bit	7	6	5	4	3	2	1	0
Field	TEN	TPOL	PRES			TMODE		
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F07H, F0FH							

Bit	Description
[7] TEN	**Timer Enable** 0 = Timer is disabled. 1 = Timer enabled to count.

Table 52. Timer 0–1 High Byte Register (TxH)

Bit	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
Address	F00H, F08H							

Table 53. Timer 0–1 Low Byte Register (TxL)

Bit	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
Address	F01H, F09H							

Bit	Description
[7:0] TH, TL	**Timer High and Low Bytes** These 2 bytes, {TH[7:0], TL[7:0]}, contain the current 16-bit timer count value.

In MULTIPROCESSOR (9-bit) Mode, the Parity (9th) bit location becomes the multiprocessor control bit. The UART Control 1 and Status 1 registers provide MULTIPROCESSOR (9-bit) Mode control and status information. If an automatic address matching scheme is enabled, the UART Address Compare Register holds the network address of the device.

MULTIPROCESSOR (9-bit) Mode Receive Interrupts

When MULTIPROCESSOR Mode is enabled, the UART only processes frames addressed to it. The determination of whether a frame of data is addressed to the UART can be made 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.

Table 64. UART Control 1 Register (U0CTL1)

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

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

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 baud rate clocks to plus eight baud rate clocks around the expected time of an incoming pulse. If an incoming pulse is detected inside this window this process is repeated. If the incoming data is a logical 1 (no pulse), the Endec returns to the initial state and waits for the next falling edge. As each falling edge is detected, the Endec clock counter is reset, resynchronizing the Endec to the incoming signal, allowing the Endec to tolerate jitter and baud rate errors in the incoming datastream. Resynchronizing the Endec does not alter the operation of the UART, which ultimately receives the data. The UART is only synchronized to the incoming data stream when a Start bit is received.

Infrared Encoder/Decoder Control Register Definitions

All infrared endec configuration and status information is set by the UART Control registers as defined in the Universal Asynchronous Receiver/Transmitter section on page 99.

! Caution: To prevent spurious signals during IrDA data transmission, set the IREN bit in the UART Control 1 Register to 1 to enable the Infrared Encoder/Decoder before enabling the GPIO Port alternate function for the corresponding pin.

Trim Bit Address 0002H

Table 92. Trim Option Bits at 0002H (TIPO)

Bit	7	6	5	4	3	2	1	0
Field	IPO_TRIM							
RESET	U							
R/W	R/W							
Address	Information Page Memory 0022H							

Note: U = Unchanged by Reset. R/W = Read/Write.

Bit	Description
[7:0] IPO_TRIM	**Internal Precision Oscillator Trim Byte** Contains trimming bits for the Internal Precision Oscillator.

Trim Bit Address 0003H

Note: The LVD is available on 8-pin devices only.

Table 93. Trim Option Bits at Address 0003H (TLVD)

Bit	7	6	5	4	3	2	1	0
Field	Reserved			LVD_TRIM				
RESET	U	U	U	U	U	U	U	U
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	Information Page Memory 0023H							

Note: U = Unchanged by Reset. R/W = Read/Write.

Bit	Description
[7:5]	**Reserved** These bits are reserved and must be programmed to 111.
[4:0] LVD_TRIM	**Low Voltage Detect Trimm** This trimming affects the low voltage detection threshold. Each LSB represents a 50mV change in the threshold level. Alternatively, the low voltage threshold may be computed from the options bit value by the following equation: $LVD\_LVL = 3.6\ V - LVD\_TRIM \times 0.05\ V$ These values are tabulated in Table 94.

Operation

This section describes the interface and modes of operation of the On-Chip Debugger.

OCD Interface

The on-chip debugger uses the DBG pin for communication with an external host. This one-pin interface is a bidirectional, open-drain interface that transmits and receives data. Data transmission is half-duplex, in that transmit and receive cannot occur simultaneously. The serial data on the DBG pin is sent using the standard asynchronous data format defined in RS-232. This pin creates an interface from the Z8 Encore! XP F082A Series products to the serial port of a host PC using minimal external hardware.Two different methods for connecting the DBG pin to an RS-232 interface are displayed in Figure 24 and Figure 25. The recommended method is the buffered implementation displayed in Figure 25. The DBG pin has a internal pull-up resistor which is sufficient for some applications (for more details about the pull-up current, see the Electrical Characteristics chapter on page 226). For OCD operation at higher data rates or in noisy systems, an external pull-up resistor is recommended.

! Caution: For operation of the on-chip debugger, all power pins (V_{DD} and AV_{DD}) must be supplied with power and all ground pins (V_{SS} and AV_{SS}) must be properly grounded. The DBG pin is open-drain and may require an external pull-up resistor to ensure proper operation.

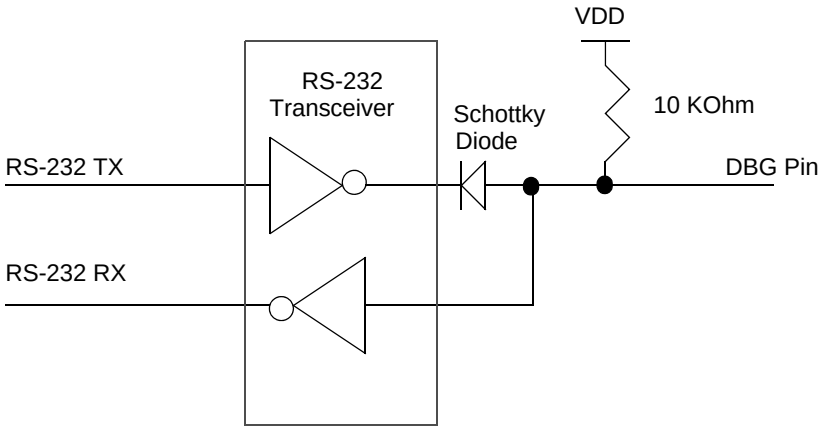


Figure 24. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface; #1 of 2

Table 112. Oscillator Configuration and Selection

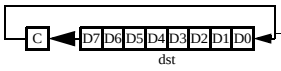
Clock Source	Characteristics	Required Setup
Internal Precision RC Oscillator	• 32.8kHz or 5.53MHz • High accuracy • No external components required	• Unlock and write Oscillator Control Register (OSCCTL) to enable and select oscillator at either 5.53MHz or 32.8kHz
External Crystal/ Resonator	• 32kHz to 20MHz • Very high accuracy (dependent on crystal or resonator used) • Requires external components	• Configure Flash option bits for correct external oscillator mode • Unlock and write OSCCTL to enable crystal oscillator, wait for it to stabilize and select as system clock (if the XTLDIS option bit has been deasserted, no waiting is required)
External RC Oscillator	• 32kHz to 4MHz • Accuracy dependent on external components	• Configure Flash option bits for correct external oscillator mode • Unlock and write OSCCTL to enable crystal oscillator and select as system clock
External Clock Drive	• 0 to 20MHz • Accuracy dependent on external clock source	• Write GPIO registers to configure PB3 pin for external clock function • Unlock and write OSCCTL to select external system clock • Apply external clock signal to GPIO
Internal Watchdog Timer Oscillator	• 10kHz nominal • Low accuracy; no external components required • Very low power consumption	• Enable WDT if not enabled and wait until WDT Oscillator is operating • Unlock and write Oscillator Control Register (OSCCTL) to enable and select oscillator

! **Caution:** Unintentional accesses to the Oscillator Control Register can actually stop the chip by switching to a nonfunctioning oscillator. To prevent this condition, the oscillator control block employs a register unlocking/locking scheme.

OSC Control Register Unlocking/Locking

To write the Oscillator Control Register, unlock it by making two writes to the OSCCTL Register with the values E7H followed by 18H. A third write to the OSCCTL Register changes the value of the actual register and returns the register to a locked state. Any other sequence of Oscillator Control Register writes has no effect. The values written to unlock the register must be ordered correctly, but are not necessarily consecutive. It is possible to write to or read from other registers within the unlocking/locking operation.

Table 128. 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
ORX dst, src	dst ← dst OR src	ER	ER	48	–	*	*	0	–	–	4	3
		ER	IM	49							4	3
POP dst	dst ← @SP SP ← SP + 1	R		50	–	–	–	–	–	–	2	2
		IR		51							2	3
POPX dst	dst ← @SP SP ← SP + 1	ER		D8	–	–	–	–	–	–	3	2
PUSH src	SP ← SP – 1 @SP ← src	R		70	–	–	–	–	–	–	2	2
		IR		71							2	3
		IM		IF70							3	2
PUSHX src	SP ← SP – 1 @SP ← src	ER		C8	–	–	–	–	–	–	3	2
RCF	C ← 0			CF	0	–	–	–	–	–	1	2
RET	PC ← @SP SP ← SP + 2			AF	–	–	–	–	–	–	1	4
RL dst		R		90	*	*	*	*	–	–	2	2
		IR		91							2	3
RLC dst		R		10	*	*	*	*	–	–	2	2
		IR		11							2	3
RR dst		R		E0	*	*	*	*	–	–	2	2
		IR		E1							2	3

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

Table 137. Watchdog Timer Electrical Characteristics and Timing

		V_{DD} = 2.7 V to 3.6 V T_A = –40°C to +105°C (unless otherwise stated)				
Symbol	**Parameter**	**Minimum**	**Typical**	**Maximum**	**Units**	**Conditions**
F_{WDT}	WDT Oscillator Frequency		10		kHz	
F_{WDT}	WDT Oscillator Error			±50	%	
T_{WDTCAL}	WDT Calibrated Time-out	0.98	1	1.02	s	V_{DD} = 3.3V; T_A = 30°C
		0.70	1	1.30	s	V_{DD} = 2.7 V to 3.6 V T_A = 0°C to 70°C
		0.50	1	1.50	s	V_{DD} = 2.7 V to 3.6 V T_A = –40°C to +105°C

Table 138. Non-Volatile Data Storage

	V_{DD} = 2.7 V to 3.6 V T_A = –40°C to +105°C				
Parameter	**Minimum**	**Typical**	**Maximum**	**Units**	**Notes**
NVDS Byte Read Time	34	–	519	µs	With system clock at 20 MHz
NVDS Byte Program Time	0.171	–	39.7	ms	With system clock at 20 MHz
Data Retention	100	–	–	years	25°C
Endurance	160,000	–	–	cycles	Cumulative write cycles for entire memory

M

N

O

S

T