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, WDT
Number of I/O	17
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	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f081ahh020sc

Table of Contents

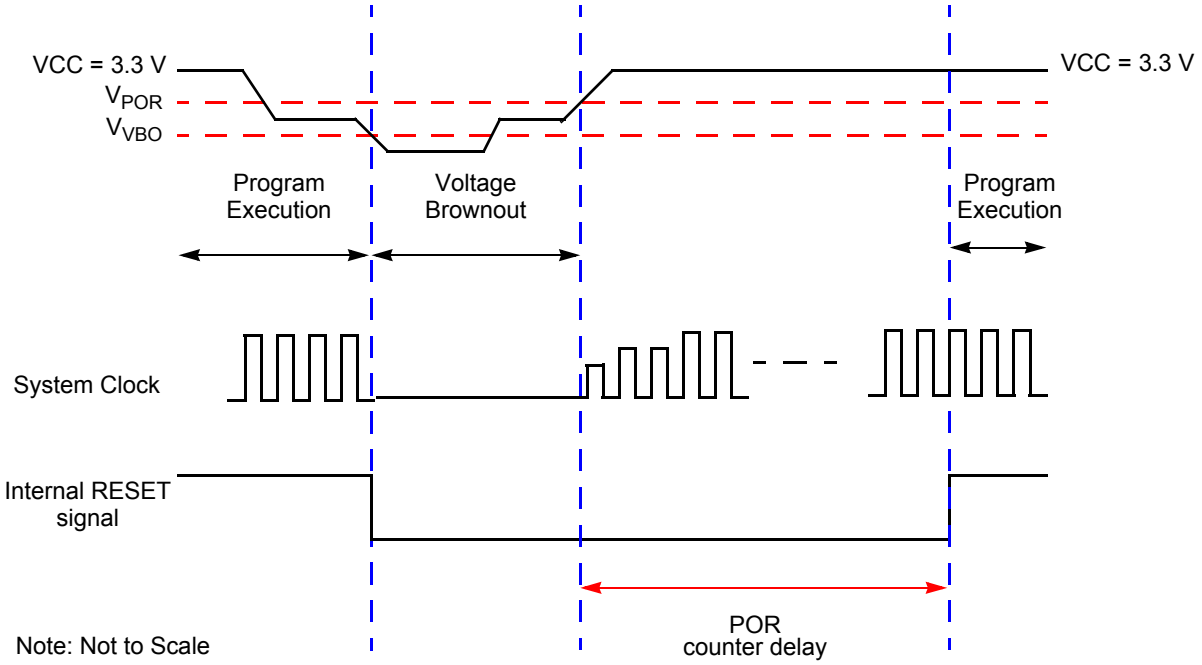


Figure 6. Voltage Brownout Reset Operation

The POR level is greater than the VBO level by the specified hysteresis value. This ensures that the device undergoes a Power-On Reset after recovering from a VBO condition.

Watchdog Timer Reset

If the device is in NORMAL or HALT mode, the Watchdog Timer can initiate a System Reset at time-out if the WDT_RES Flash Option Bit is programmed to 1. This is the unprogrammed state of the WDT_RES Flash Option Bit. If the bit is programmed to 0, it configures the Watchdog Timer to cause an interrupt, not a System Reset, at time-out.

The WDT bit in the Reset Status (RSTSTAT) register is set to signify that the reset was initiated by the Watchdog Timer.

External Reset Input

The $\overline{\text{RESET}}$ pin has a Schmitt-Triggered input and an internal pull-up resistor. Once the $\overline{\text{RESET}}$ pin is asserted for a minimum of four system clock cycles, the device progresses through the System Reset sequence. Because of the possible asynchronicity of the system clock and reset signals, the required reset duration may be as short as three clock periods

initiate Stop Mode Recovery without being written to the Port Input Data register or without initiating an interrupt (if enabled for that pin).

Stop Mode Recovery Using the External $\overline{\text{RESET}}$ Pin

When the Z8 Encore! XP F082A Series device is in STOP mode and the external $\overline{\text{RESET}}$ pin is driven Low, a system reset occurs. Because of a glitch filter operating on the $\overline{\text{RESET}}$ pin, the Low pulse must be greater than the minimum width specified, or it is ignored. See Electrical Characteristics on page 221 for details.

Low Voltage Detection

In addition to the Voltage Brownout (VBO) Reset described above, it is also possible to generate an interrupt when the supply voltage drops below a user-selected value. For details about configuring the Low Voltage Detection (LVD) and the threshold levels available, see Trim Bit Address 0003H on page 159. The LVD function is available on the 8-pin product versions only.

When the supply voltage drops below the LVD threshold, the LVD bit of the Reset Status (RSTSTAT) register is set to one. This bit remains one until the low-voltage condition goes away. Reading or writing this bit does not clear it. The LVD circuit can also generate an interrupt when so enabled, see Interrupt Vectors and Priority on page 58. The LVD bit is NOT latched, so enabling the interrupt is the only way to guarantee detection of a transient low voltage event.

The LVD functionality depends on circuitry shared with the VBO block; therefore, disabling the VBO also disables the LVD.

Reset Register Definitions

The following sections define the Reset registers.

Reset Status Register

The Reset Status (RSTSTAT) register is a read-only register that indicates the source of the most recent Reset event, indicates a Stop Mode Recovery event, and indicates a Watchdog Timer time-out. Reading this register resets the upper four bits to 0.

This register shares its address with the Watchdog Timer control register, which is write-only (see Table 11 on page 31).

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

Table 13. 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

during STOP mode do not initiate Stop Mode Recovery.
1 = The Port pin is configured as a Stop Mode Recovery source. Any logic transition on this pin during STOP mode initiates Stop Mode Recovery.

Port A–D Pull-up Enable Sub-Registers

The Port A–D Pull-up Enable sub-register (Table 24) 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 sub-registers enables a weak internal resistive pull-up on the specified Port pins.

Table 24. Port A–D Pull-Up Enable Sub-Registers (PxPUE)

BITS	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
ADDR	If 06H in Port A–D Address Register, accessible through the Port A–D Control Register							

PPUE[7:0]—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.

Port A–D Alternate Function Set 1 Sub-Registers

The Port A–D Alternate Function Set1 sub-register (Table 25) is accessed through the Port A–D Control register by writing 07H to the Port A–D Address register. The Alternate Function Set 1 sub-registers selects the alternate function available at a port pin. Alternate Functions selected by setting or clearing bits of this register are defined in GPIO Alternate Functions on page 38.

Note: *Alternate function selection on port pins must also be enabled as described in* Port A–D Alternate Function Sub-Registers *on page 47.*

Table 25. Port A–D Alternate Function Set 1 Sub-Registers (PxAFS1)

BITS	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
ADDR	If 07H in Port A–D Address Register, accessible through the Port A–D Control Register							

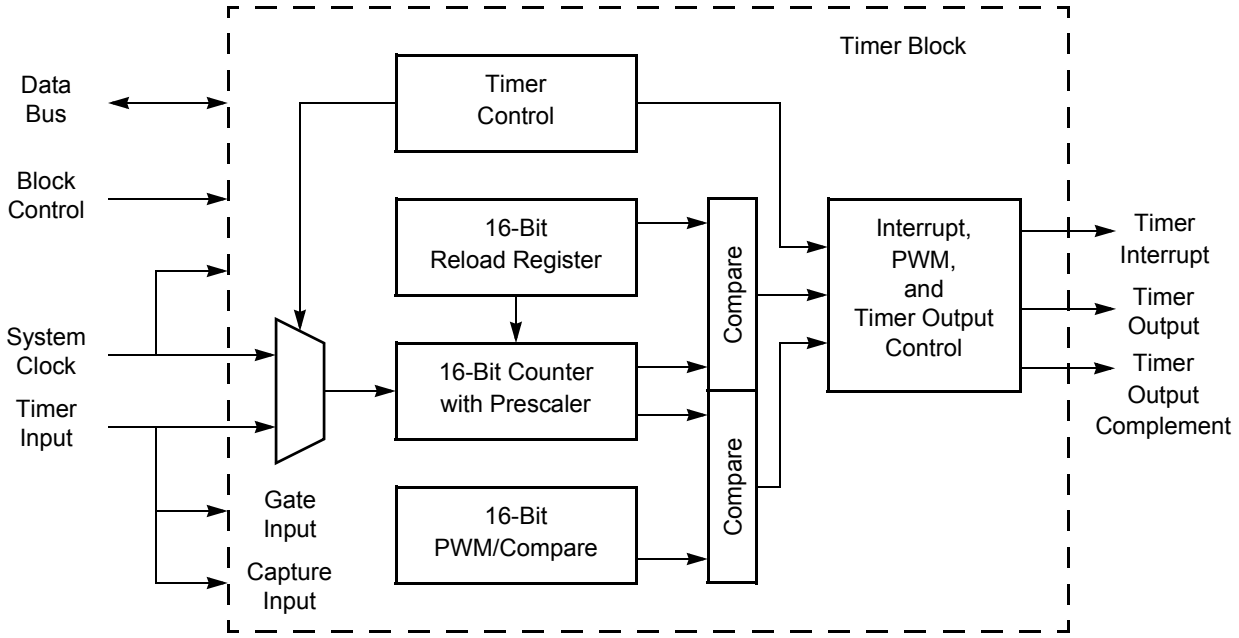


Figure 9. Timer Block Diagram

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. The timer is automatically disabled and stops counting.

duration of the negative phase of the PWM signal (as defined by the difference between the PWM registers and the Timer Reload registers).

5. Write to the Timer Reload High and Low Byte registers to set the Reload value (PWM period). The Reload value must be greater than the PWM value.
6. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
7. Configure the associated GPIO port pin for the Timer Output and Timer Output Complement alternate functions. The Timer Output Complement function is shared with the Timer Input function for both timers. Setting the timer mode to Dual PWM automatically switches the function from Timer In to Timer Out Complement.
8. Write to the Timer Control register to enable the timer and initiate counting.

The PWM period is represented by the following equation:

$$\text{PWM Period (s)} = \frac{\text{Reload Value x Prescale}}{\text{System Clock Frequency (Hz)}}$$

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

If TPOL is set to 0, the ratio of the PWM output High time to the total period is represented by:

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{Reload Value} - \text{PWM Value}}{\text{Reload Value}} \times 100$$

If TPOL is set to 1, the ratio of the PWM output High time to the total period is represented by:

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{PWM Value}}{\text{Reload Value}} \times 100$$

CAPTURE Mode

In CAPTURE mode, the current timer count value is recorded when the appropriate external Timer Input transition occurs. The Capture count value is written to the Timer PWM High and Low Byte Registers. The timer input is the system clock. The TPOL bit in the Timer Control register determines if the Capture occurs on a rising edge or a falling edge of the Timer Input signal. When the Capture event occurs, an interrupt is generated and the timer continues counting. The INPCAP bit in TxCTL0 register is set to indicate the timer interrupt is because of an input capture event.

The timer continues counting up to the 16-bit Reload value stored in the Timer Reload High and Low Byte registers. Upon reaching the Reload value, the timer generates an interrupt and continues counting. The INPCAP bit in TxCTL0 register clears indicating the timer interrupt is not because of an input capture event.

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® F082A Series products while the IR_RXD signal is received through the RXD pin.

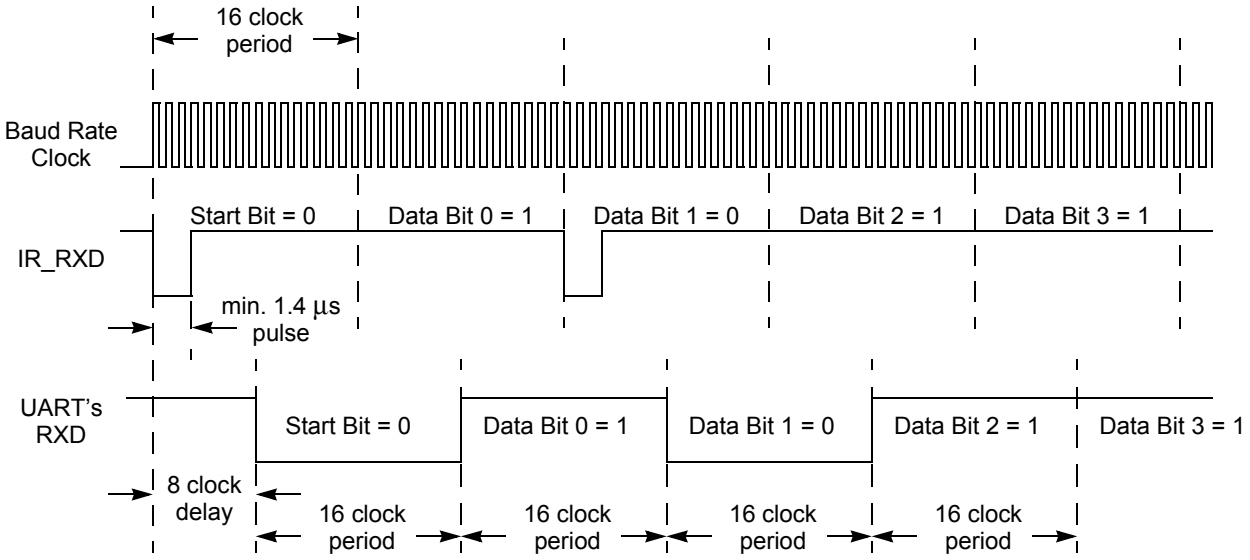


Figure 18. IrDA Data Reception

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

can output values across the entire 11-bit range, from -1024 to +1023. In SINGLE-ENDED mode, the output generally ranges from 0 to +1023, but offset errors can cause small negative values.

The ADC registers actually return 13 bits of data, but the two LSBs are intended for compensation use only. When the software compensation routine is performed on the 13 bit raw ADC value, two bits of resolution are lost because of a rounding error. As a result, the final value is an 11-bit number.

Hardware Overflow

When the hardware overflow bit (OVF) is set in ADC Data Low Byte (ADCD_L) register, all other data bits are invalid. The hardware overflow bit is set for values greater than V_{ref} and less than $-V_{ref}$ (DIFFERENTIAL mode).

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 power up. 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 desired analog inputs by configuring the general-purpose I/O pins for alternate analog function. This configuration disables the digital input and output drivers.
2. Write the ADC Control/Status Register 1 to configure the ADC.
   - Write to BUFMODE[2:0] to select SINGLE-ENDED or DIFFERENTIAL mode, as well as unbuffered or buffered mode.
   - Write the REFSELH 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.
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 (the ADC can be configured and enabled with the same write instruction):
   - 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.

Flash Operation Timing Using the Flash Frequency Registers

Before performing either a program or erase operation on Flash memory, you must first configure the Flash Frequency High and Low Byte registers. The Flash Frequency registers allow programming and erasing of the Flash with system clock frequencies ranging from 32 kHz (32768 Hz) through 20 MHz.

The Flash Frequency High and Low Byte registers combine to form a 16-bit value, FFREQ, to control timing for Flash program and erase operations. The 16-bit binary Flash Frequency value must contain the system clock frequency (in kHz). This value is calculated using the following equation:

$$\text{FFREQ[15:0]} = \frac{\text{System Clock Frequency (Hz)}}{1000}$$

Caution: *Flash programming and erasure are not supported for system clock frequencies below 32 kHz (32768 Hz) or above 20 MHz. The Flash Frequency High and Low Byte registers must be loaded with the correct value to ensure operation of the Z8 Encore! XP® F082A Series devices.*

Flash Code Protection Against External Access

The user code contained within the Flash memory can be protected against external access by the on-chip debugger. Programming the FRP Flash Option Bit prevents reading of the user code with the On-Chip Debugger. See Flash Option Bits on page 153 and On-Chip Debugger on page 173 for more information.

Flash Code Protection Against Accidental Program and Erasure

The Z8 Encore! XP F082A Series provides several levels of protection against accidental program and erasure of the Flash memory contents. This protection is provided by a combination of the Flash Option bits, the register locking mechanism, the page select redundancy and the sector level protection control of the Flash Controller.

Flash Code Protection Using the Flash Option Bits

The FRP and FWP Flash Option Bits combine to provide three levels of Flash Program Memory protection as listed in Table 77. See Flash Option Bits on page 153 for more information.

Trim Bit Address 0001H

Table 89. Trim Option Bits at 0001H

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved							
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
ADDR	Information Page Memory 0021H							

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

Reserved—Altering this register may result in incorrect device operation.

Trim Bit Address 0002H

Table 90. Trim Option Bits at 0002H (TIPO)

BITS	7	6	5	4	3	2	1	0
FIELD	IPO_TRIM							
RESET	U							
R/W	R/W							
ADDR	Information Page Memory 0022H							

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

IPO_TRIM—Internal Precision Oscillator Trim Byte
Contains trimming bits for Internal Precision Oscillator.

Trim Bit Address 0003H

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

Table 91. Trim Option Bits at Address 0003H (TLVD)

BITS	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
ADDR	Information Page Memory 0023H							

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

Table 100. Serialization Data Locations

Info Page Address	Memory Address	Usage
1C	FE1C	Serial Number Byte 3 (most significant)
1D	FE1D	Serial Number Byte 2
1E	FE1E	Serial Number Byte 1
1F	FE1F	Serial Number Byte 0 (least significant)

Randomized Lot Identifier

Table 101. Lot Identification Number (RAND_LOT)

BITS	7	6	5	4	3	2	1	0
FIELD	RAND_LOT							
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
ADDR	Interspersed throughout Information Page Memory							

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

RAND_LOT—Randomized Lot ID
The randomized lot ID is a 32 byte binary value that changes for each production lot.

Table 102. Randomized Lot ID Locations

Info Page Address	Memory Address	Usage
3C	FE3C	Randomized Lot ID Byte 31 (most significant)
3D	FE3D	Randomized Lot ID Byte 30
3E	FE3E	Randomized Lot ID Byte 29
3F	FE3F	Randomized Lot ID Byte 28
58	FE58	Randomized Lot ID Byte 27
59	FE59	Randomized Lot ID Byte 26
5A	FE5A	Randomized Lot ID Byte 25
5B	FE5B	Randomized Lot ID Byte 24

Non-Volatile Data Storage

The Z8 Encore! XP® F082A Series devices contain a non-volatile data storage (NVDS) element of up to 128 bytes. This memory can perform over 100,000 write cycles.

Operation

The NVDS is implemented by special purpose Zilog® software stored in areas of program memory, which are not user-accessible. These special-purpose routines use the Flash memory to store the data. The routines incorporate a dynamic addressing scheme to maximize the write/erase endurance of the Flash.

Note: *Different members of the Z8 Encore! XP F082A Series feature multiple NVDS array sizes. See Z8 Encore! XP® F082A Series Family Part Selection Guide on page 3 for details. Also the members containing 8 KB of Flash memory do not include the NVDS feature.*

NVDS Code Interface

Two routines are required to access the NVDS: a write routine and a read routine. Both of these routines are accessed with a CALL instruction to a pre-defined address outside of the user-accessible program memory. Both the NVDS address and data are single-byte values. Because these routines disturb the working register set, user code must ensure that any required working register values are preserved by pushing them onto the stack or by changing the working register pointer just prior to NVDS execution.

During both read and write accesses to the NVDS, interrupt service is NOT disabled. Any interrupts that occur during the NVDS execution must take care not to disturb the working register and existing stack contents or else the array may become corrupted. Disabling interrupts before executing NVDS operations is recommended.

Use of the NVDS requires 15 bytes of available stack space. Also, the contents of the working register set are overwritten.

For correct NVDS operation, the Flash Frequency Registers must be programmed based on the system clock frequency (see Flash Operation Timing Using the Flash Frequency Registers on page 145).

Byte Write

To write a byte to the NVDS array, the user code must first push the address, then the data byte onto the stack. The user code issues a `CALL` instruction to the address of the byte-write routine (0x10B3). At the return from the sub-routine, the write status byte

If the device is not in DEBUG mode or the Flash Read Protect Option bit is enabled, this command reads and discards one byte.

```
DBG ← 12H
DBG ← 1-5 byte opcode
```

On-Chip Debugger Control Register Definitions

OCD Control Register

The OCD Control register controls the state of the On-Chip Debugger. This register is used to enter or exit DEBUG mode and to enable the BRK instruction. It can also reset the Z8 Encore! XP® F082A Series device.

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 106. 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 = The Z8 Encore! XP F082A Series device is operating in NORMAL mode.
1 = The Z8 Encore! XP F082A 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.

Oscillator Control

The Z8 Encore! XP® F082A Series devices uses five possible clocking schemes, each user-selectable:

- Internal precision trimmed RC oscillator (IPO).
- On-chip oscillator using off-chip crystal or resonator.
- On-chip oscillator using external RC network.
- External clock drive.
- On-chip low power Watchdog Timer oscillator.
- Clock failure detection circuitry.

In addition, Z8 Encore! XP F082A Series devices contain clock failure detection and recovery circuitry, allowing continued operation despite a failure of the system clock oscillator.

Operation

This chapter discusses the logic used to select the system clock and handle primary oscillator failures.

System Clock Selection

The oscillator control block selects from the available clocks. Table 108 details each clock source and its usage.

Assembly Language Syntax

For proper instruction execution, eZ8 CPU assembly language syntax requires that the operands be written as ‘destination, source’. After assembly, the object code usually has the operands in the order ‘source, destination’, but ordering is opcode-dependent. The following instruction examples illustrate the format of some basic assembly instructions and the resulting object code produced by the assembler. This binary format must be followed if manual program coding is preferred or if you intend to implement your own assembler.

Example 1: If the contents of Registers 43H and 08H are added and the result is stored in 43H, the assembly syntax and resulting object code is:

Table 112. Assembly Language Syntax Example 1

Assembly Language Code	ADD	43H,	08H	(ADD dst, src)
Object Code	04	08	43	(OPC src, dst)

Example 2: In general, when an instruction format requires an 8-bit register address, that address can specify any register location in the range 0–255 or, using Escaped Mode Addressing, a Working Register R0–R15. If the contents of Register 43H and Working Register R8 are added and the result is stored in 43H, the assembly syntax and resulting object code is:

Table 113. Assembly Language Syntax Example 2

Assembly Language Code	ADD	43H,	R8	(ADD dst, src)
Object Code	04	E8	43	(OPC src, dst)

See the device-specific Product Specification to determine the exact register file range available. The register file size varies, depending on the device type.

eZ8 CPU Instruction Notation

In the eZ8 CPU Instruction Summary and Description sections, the operands, condition codes, status flags, and address modes are represented by a notational shorthand that is described in Table 114.

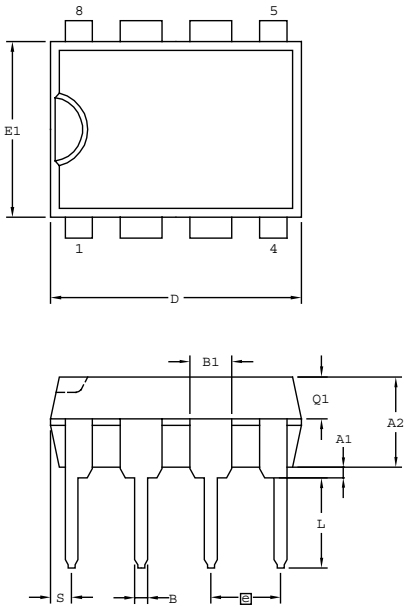
Table 124. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
HALT	Halt Mode			7F	–	–	–	–	–	–	1	2
INC dst	dst ← dst + 1	R		20	–	*	*	–	–	–	2	2
		IR		21							2	3
		r		0E-FE							1	2
INCW dst	dst ← dst + 1	RR		A0	–	*	*	*	–	–	2	5
		IRR		A1							2	6
IRET	FLAGS ← @SP SP ← SP + 1 PC ← @SP SP ← SP + 2 IRQCTL[7] ← 1			BF	*	*	*	*	*	*	1	5
JP dst	PC ← dst	DA		8D	–	–	–	–	–	–	3	2
		IRR		C4							2	3
JP cc, dst	if cc is true PC ← dst	DA		0D-FD	–	–	–	–	–	–	3	2
JR dst	PC ← PC + X	DA		8B	–	–	–	–	–	–	2	2
JR cc, dst	if cc is true PC ← PC + X	DA		0B-FB	–	–	–	–	–	–	2	2
LD dst, rc	dst ← src	r	IM	0C-FC	–	–	–	–	–	–	2	2
		r	X(r)	C7							3	3
		X(r)	r	D7							3	4
		r	Ir	E3							2	3
		R	R	E4							3	2
		R	IR	E5							3	4
		R	IM	E6							3	2
		IR	IM	E7							3	3
		Ir	r	F3							2	3
		IR	R	F5							3	3

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

Packaging

Figure 39 displays the 8-pin Plastic Dual Inline Package (PDIP) available for Z8 Encore! XP® F082A Series devices.

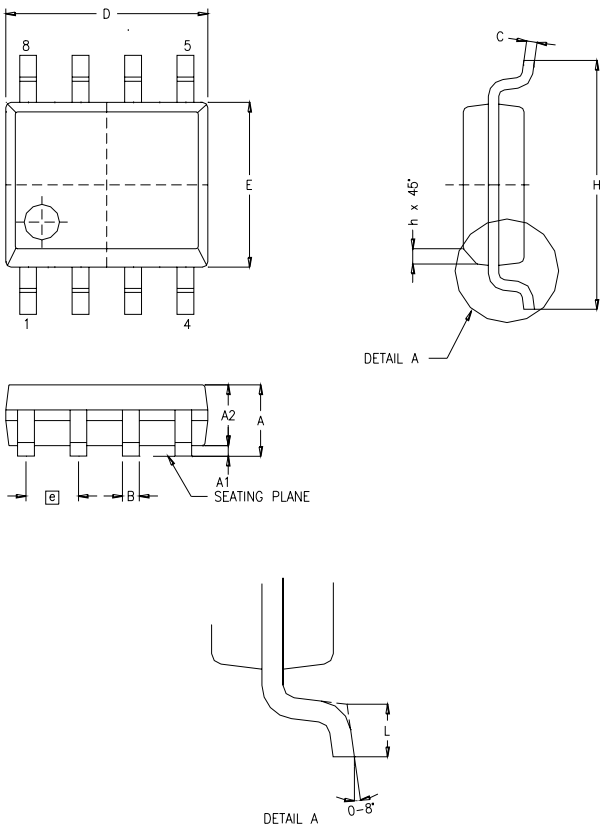


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A1	0.38	0.81	0.015	0.032
A2	3.25	3.81	0.128	0.150
B	0.38	0.53	0.015	0.021
B1	1.40	1.65	0.055	0.065
C	0.20	0.30	0.008	0.012
D	9.02	9.78	0.355	0.385
E	7.62	8.26	0.300	0.325
E1	6.10	6.60	0.240	0.260
e	2.54 BSC		0.100 BSC	
eA	7.87	9.14	0.310	0.360
L	3.18	3.43	0.125	0.135
Q1	1.40	1.65	0.055	0.065
S	0.64	0.89	0.025	0.035

CONTROLLING DIMENSIONS : MM.

Figure 39. 8-Pin Plastic Dual Inline Package (PDIP)

Figure 40 displays the 8-pin Small Outline Integrated Circuit package (SOIC) available for the Z8 Encore! XP® F082A Series devices.



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	1.55	1.73	0.061	0.068
A1	0.10	0.25	0.004	0.010
A2	1.40	1.55	0.055	0.061
B	0.36	0.48	0.014	0.019
C	0.18	0.25	0.007	0.010
D	4.80	4.98	0.189	0.196
E	3.81	3.99	0.150	0.157
e	1.27 BSC		.050 BSC	
H	5.84	6.15	0.230	0.242
h	0.25	0.40	0.010	0.016
L	0.46	0.81	0.018	0.032

CONTROLLING DIMENSIONS : MM
LEADS ARE COPLANAR WITHIN .004 INCH.

Figure 40. 8-Pin Small Outline Integrated Circuit Package (SOIC)