

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	Active
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	-
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	17
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	20-VQFN Exposed Pad
Supplier Device Package	-
Purchase URL	<a href="https://www.e-xfl.com/product-detail/zilog/z8f0431qh020eg">https://www.e-xfl.com/product-detail/zilog/z8f0431qh020eg</a>

Master Interrupt Enable .....	55
Interrupt Vectors and Priority .....	56
Interrupt Assertion .....	56
Software Interrupt Assertion .....	57
Interrupt Control Register Definitions .....	57
Interrupt Request 0 Register .....	58
Interrupt Request 1 Register .....	59
Interrupt Request 2 Register .....	60
IRQ0 Enable High and Low Bit Registers .....	60
IRQ1 Enable High and Low Bit Registers .....	62
IRQ2 Enable High and Low Bit Registers .....	63
Interrupt Edge Select Register .....	65
Shared Interrupt Select Register .....	66
Interrupt Control Register .....	67
Timers .....	68
Architecture .....	68
Operation .....	69
Timer Operating Modes .....	69
Reading the Timer Count Values .....	82
Timer Pin Signal Operation .....	82
Timer Control Register Definitions .....	83
Timer 0–1 High and Low Byte Registers .....	83
Timer Reload High and Low Byte Registers .....	85
Timer 0–1 PWM High and Low Byte Registers .....	86
Timer 0–1 Control Registers .....	87
Watchdog Timer .....	92
Operation .....	92
Watchdog Timer Refresh .....	93
Watchdog Timer Time-Out Response .....	93
Watchdog Timer Reload Unlock Sequence .....	94
Watchdog Timer Control Register Definitions .....	95
Watchdog Timer Control Register .....	95
Watchdog Timer Reload Upper, High and Low Byte Registers .....	96
Analog-to-Digital Converter .....	98
Architecture .....	98
Operation .....	99
ADC Timing .....	100
ADC Interrupt .....	101
Reference Buffer .....	101
Internal Voltage Reference Generator .....	101

## CPU and Peripheral Overview

The eZ8 CPU, Zilog's latest 8-bit CPU, meets the continuing demand for faster and more 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 CPU code
- Expanded internal register file allows access up to 4KB
- New instructions improve execution efficiency for code developed using high-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 about the eZ8 CPU, refer to the [eZ8 CPU Core User Manual \(UM0128\)](#), which is available for download on [www.zilog.com](http://www.zilog.com).

## General Purpose Input/Output

The Z8 Encore! F0830 Series features up to 25 port pins (Ports A–D) for general-purpose input/output (GPIO). The number of GPIO pins available is a function of package. Each pin is individually programmable.

## Flash Controller

The Flash Controller programs and erases the Flash memory. It also supports protection against accidental programming and erasure.

## Pin Description

The Z8 Encore! F0830 Series products are available in a variety of package styles and pin configurations. This chapter describes the signals and the pin configurations for each of the package styles. For information about the physical package specifications, see the [Packaging](#) chapter on page 199.

### Available Packages

Table 3 lists the package styles that are available for each device in the Z8 Encore! F0830 Series product line.

**Table 3. Z8 Encore! F0830 Series Package Options**

Part Number	ADC	20-pin QFN	20-pin SOIC	20-pin SSOP	20-pin PDIP	28-pin QFN	28-pin SOIC	28-pin SSOP	28-pin PDIP
Z8F1232	Yes	X	X	X	X	X	X	X	X
Z8F1233	No	X	X	X	X	X	X	X	X
Z8F0830	Yes	X	X	X	X	X	X	X	X
Z8F0831	No	X	X	X	X	X	X	X	X
Z8F0430	Yes	X	X	X	X	X	X	X	X
Z8F0431	No	X	X	X	X	X	X	X	X
Z8F0230	Yes	X	X	X	X	X	X	X	X
Z8F0231	No	X	X	X	X	X	X	X	X
Z8F0130	Yes	X	X	X	X	X	X	X	X
Z8F0131	No	X	X	X	X	X	X	X	X

### Pin Configurations

Figures 2 and 3 display the pin configurations of all of the packages available in the Z8 Encore! F0830 Series. See [Table 4](#) on page 11 for a description of the signals. Analog input alternate functions (ANAx) are not available on the following devices:

- Z8F0831
- Z8F0431
- Z8F0131
- Z8F0231
- Z8F1233

# Register Map

Table 8 provides an address map of the Z8 Encore! F0830 Series register file. Not all devices and package styles in the Z8 Encore! F0830 Series support the ADC or all of the GPIO ports. Consider registers for unimplemented peripherals as reserved.

**Table 8. Register File Address Map**

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page No.
<b>General Purpose RAM</b>				
000–0FF	General purpose register file RAM	—	XX	
100–EFF	Reserved	—	XX	
<b>Timer 0</b>				
F00	Timer 0 high byte	T0H	00	83
F01	Timer 0 low byte	T0L	01	83
F02	Timer 0 reload high byte	T0RH	FF	85
F03	Timer 0 reload low byte	T0RL	FF	85
F04	Timer 0 PWM high byte	T0PWMH	00	86
F05	Timer 0 PWM low byte	T0PWML	00	86
F06	Timer 0 control 0	T0CTL0	00	87
F07	Timer 0 control 1	T0CTL1	00	88
<b>Timer 1</b>				
F08	Timer 1 high byte	T1H	00	83
F09	Timer 1 low byte	T1L	01	83
F0A	Timer 1 reload high byte	T1RH	FF	85
F0B	Timer 1 reload low byte	T1RL	FF	85
F0C	Timer 1 PWM high byte	T1PWMH	00	86
F0D	Timer 1 PWM low byte	T1PWML	00	86
F0E	Timer 1 control 0	T1CTL0	00	87
F0F	Timer 1 control 1	T1CTL1	00	83
F10–F6F	Reserved	—	XX	
<b>Analog-to-Digital Converter (ADC)</b>				
F70	ADC control 0	ADCCTL0	00	102
F71	Reserved	—	XX	
F72	ADC data high byte	ADCD_H	XX	103

Note: XX = Undefined.

## ***Reset and Stop Mode Recovery***

The reset controller in the Z8 Encore! F0830 Series controls RESET and Stop Mode Recovery operations. In a typical operation, the following events can cause a reset:

- Power-On Reset (POR)
- Voltage Brown-Out (VBO)
- Watchdog Timer time-out (when configured by the WDT\_RES Flash option bit to initiate a reset)
- External  $\overline{\text{RESET}}$  pin assertion (when the alternate RESET function is enabled by the GPIO register)
- On-Chip Debugger initiated reset (OCDCTL[0] set to 1)

When the device is in STOP Mode, a Stop Mode Recovery event is initiated by either of the following occurrences:

- A Watchdog Timer time-out
- A GPIO port input pin transition on an enabled Stop Mode Recovery source

The VBO circuitry on the device generates a VBO reset when the supply voltage drops below a minimum safe level.

### **Reset Types**

The Z8 Encore! F0830 Series provides different types of Reset operations. Stop Mode Recovery is considered a form of reset. Table 9 lists the types of resets and their operating characteristics. The duration of a system reset is longer if the external crystal oscillator is enabled by the Flash option bits; the result is additional time for oscillator startup.

## HALT Mode

Executing the eZ8 CPU HALT instruction places the device into HALT Mode. In HALT Mode, the operating characteristics are:

- Primary oscillator is enabled and continues to operate
- System clock is enabled and continues to operate
- eZ8 CPU is stopped
- Program counter (PC) stops incrementing
- Watchdog Timer's internal RC oscillator continues to operate
- If enabled, the Watchdog Timer continues to operate
- All other on-chip peripherals continue to operate

The eZ8 CPU can be brought out of HALT Mode by any one of the following operations:

- Interrupt
- Watchdog Timer time-out (interrupt or reset)
- Power-On Reset
- Voltage Brown-Out reset
- External  $\overline{\text{RESET}}$  pin assertion

To minimize current in HALT Mode, all GPIO pins that are configured as digital inputs must be driven to  $V_{DD}$  when pull-up register bit is enabled or to one of power rail ( $V_{DD}$  or GND) when pull-up register bit is disabled.

## Peripheral Level Power Control

In addition to the STOP and HALT modes, it is possible to disable each peripheral on each of the Z8 Encore! F0830 Series devices. Disabling a given peripheral minimizes its power consumption.

## Power Control Register Definitions

### Power Control Register 0

Each bit of the following registers disables a peripheral block, either by gating its system clock input or by removing power from the block.

## Port A–D Control Registers

The Port A–D Control registers, shown in Table 20, set the GPIO port operation. The value in the corresponding Port A–D Address Register determines which subregister is read from or written to by a Port A–D Control Register transaction.

**Table 20. Port A–D Control Registers (PxCTL)**

Bit	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
Address	FD1H, FD5H, FD9H, FDDH							

Bit	Description
[7:0] PCTL	<b>Port Control</b> The Port Control Register provides access to all subregisters that configure the GPIO port operation.

## Port A–D Data Direction Subregisters

The Port A–D Data Direction Subregister, shown in Table 21, is accessed through the Port A–D Control Register by writing 01H to the Port A–D Address Register.

**Table 21. Port A–D Data Direction Subregisters (PxDD)**

Bit	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
Address	If 01H in Port A–D Address Register, accessible through the Port A–D Control Register							

Bit	Description
[7:0] DDx	<b>Data Direction</b> 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–D Output Data Register is driven onto the port pin. 1 = Input. The port pin is sampled and the value written into the Port A–D Input Data Register. The output driver is tristated.

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



## Port A–D High Drive Enable Subregisters

The Port A–D High Drive Enable Subregister, shown in Table 24, is accessed through the Port A–D Control Register by writing 04H to the Port A–D Address Register. Setting the bits in the Port A–D High Drive Enable subregisters to 1 configures the specified port pins for high-output current drive operation. The Port A–D High Drive Enable Subregister affects the pins directly and, as a result, alternate functions are also affected.

**Table 24. Port A–D High Drive Enable Subregisters (PxHDE)**

Bit	7	6	5	4	3	2	1	0
Field	PHDE7	PHDE6	PHDE5	PHDE4	PHDE3	PHDE2	PHDE1	PHDE0
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 04H in Port A–D Address Register, accessible through the Port A–D Control Register							

Bit	Description
[7:0]	<b>Port High Drive Enable</b>
PHDEx	0 = The port pin is configured for standard output current drive. 1 = The port pin is configured for high output current drive.
Note: x indicates the specific GPIO port pin number (7–0).	

reload. For the timer output to make a state change at a ONE-SHOT time-out (rather than a single cycle pulse), first set the TPOL bit in the Timer Control Register to the start value before enabling ONE-SHOT Mode. After starting the timer, set TPOL to the opposite bit value.

Observe the following steps for configuring a timer for ONE-SHOT Mode and for initiating the count:

1. Write to the Timer Control Register to:
  - Disable the timer
  - Configure the timer for ONE-SHOT Mode
  - Set the prescale value
  - Set the initial output level (High or Low) if using the timer output Alternate function
2. Write to the Timer High and Low Byte registers to set the starting count value.
3. Write to the Timer Reload High and Low Byte registers to set the reload value.
4. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
5. If using the timer output function, configure the associated GPIO port pin for the timer output alternate function.
6. Write to the Timer Control Register to enable the timer and initiate counting.

In ONE-SHOT Mode, the system clock always provides the timer input. The timer period is calculated with the following equation:

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

## **CONTINUOUS Mode**

In CONTINUOUS 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, the count value in the Timer High and Low Byte registers is reset to 0001H and the counting resumes. Additionally, 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.

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

1. Write to the Timer Control Register to:

## PWM SINGLE OUTPUT Mode

In PWM SINGLE OUTPUT Mode, the timer outputs a pulse width modulated (PWM) output signal through a GPIO port pin. The timer input is the system clock. The timer first counts up to 16-bit PWM match value stored in the timer PWM High and Low Byte registers. When the timer count value matches the PWM value, the timer output toggles. The timer continues counting until it reaches the reload value stored in the Timer Reload High and Low Byte registers. Upon reaching the reload value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes.

If the TPOL bit in the Timer Control Register is set to 1, the timer output signal begins as a High (1) and transitions to a Low (0) when the timer value matches the PWM value. The timer output signal returns to a High (1) after the timer reaches the reload value and is reset to 0001H.

If the TPOL bit in the Timer Control Register is set to 0, the timer output signal begins as a Low (0) and transitions to a High (1) when the timer value matches the PWM value. The timer output signal returns to a Low (0) after the timer reaches the reload value and is reset to 0001H.

Observe the following steps for configuring a timer for PWM SINGLE OUTPUT Mode and for initiating PWM operation:

1. Write to the Timer Control Register to:
  - Disable the timer
  - Configure the timer for PWM Mode
  - Set the prescale value
  - Set the initial logic level (High or Low) and PWM High/Low transition for the timer output alternate function
2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H). This value only affects the first pass in PWM Mode. After the first timer reset in PWM Mode, counting always begins at the reset value of 0001H.
3. Write to the PWM High and Low Byte registers to set the PWM value.
4. 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.
5. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
6. Configure the associated GPIO port pin for the timer output alternate function.
7. Write to the Timer Control Register to enable the timer and initiate counting.

The PWM period is represented by the following equation:

## Timer Reload High and Low Byte Registers

The Timer 0–1 Reload High and Low Byte (TxRH and TxRL) registers, shown in Tables 52 and 53, 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 52. Timer 0–1 Reload High Byte Register (TxRH)**

Bit	7	6	5	4	3	2	1	0
Field	TRH							
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
Address	F02H, F0AH							

**Table 53. Timer 0–1 Reload Low Byte Register (TxRL)**

Bit	7	6	5	4	3	2	1	0
Field	TRL							
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
Address	F03H, F0BH							

Bit	Description
[7:0] TRH, TRL	<b>Timer Reload Register High and Low</b> These two bytes form the 16-bit reload value, {TRH[7:0], TRL[7:0]}. This value sets the maximum count value, which initiates a timer reload to 0001H. In COMPARE Mode, these two bytes form the 16-bit compare value.

## Watchdog Timer Reload Upper, High and Low Byte Registers

The Watchdog Timer Reload Upper, High and Low Byte (WDTU, WDTH, WDTL) registers, shown in Tables 60 through 62, form the 24-bit reload value that is loaded into the Watchdog Timer when a WDT instruction executes. This 24-bit value ranges across bits [23:0] to encompass the three bytes {WDTU[7:0], WDTH[7:0], WDTL[7:0]}. Writing to these registers sets the appropriate reload value; reading from these registers returns the current Watchdog Timer count value.

**! Caution:** The 24-bit WDT reload value must not be set to a value less than 000004H.

**Table 60. Watchdog Timer Reload Upper Byte Register (WDTU)**

Bit	7	6	5	4	3	2	1	0
Field	WDTU							
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	FF1H							
Note: *A read returns the current WDT count value; a write sets the appropriate reload value.								

Bit	Description
[7:0]	<b>WDT Reload Upper Byte</b>
WDTU	Most significant byte (MSB), Bits[23:16], of the 24-bit WDT reload value.

**Table 61. Watchdog Timer Reload High Byte Register (WDTH)**

Bit	7	6	5	4	3	2	1	0
Field	WDTH							
RESET	0	0	0	0	0	1	0	0
R/W	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*
Address	FF2H							
Note: *A read returns the current WDT count value; a write sets the appropriate reload value.								

Bit	Description
[7:0]	<b>WDT Reload High Byte</b>
WDTH	Middle byte, bits[15:8] of the 24-bit WDT reload value.

## ADC Interrupt

The ADC can generate an interrupt request when a conversion has been completed. An interrupt request that is pending when the ADC is disabled is not cleared automatically.

## Reference Buffer

The reference buffer, RBUF, supplies the reference voltage for the ADC. When enabled, the internal voltage reference generator supplies the ADC. When RBUF is disabled, the ADC must have the reference voltage supplied externally through the  $V_{REF}$  pin in 28-pin package. RBUF is controlled by the REFEN bit in the ADC Control Register.

## Internal Voltage Reference Generator

The internal voltage reference generator provides the voltage  $VR_2$ , for the RBUF.  $VR_2$  is 2V.

## Calibration and Compensation

A user can perform calibration and store the values into Flash or the user code can perform a manual offset calibration. There is no provision for manual gain calibration.

## ADC Control Register Definitions

The ADC Control registers are defined in this section.

- 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.

---

## NVDS Operational Requirements

The device uses a 12KB Flash memory space, despite the maximum specified Flash size of 8KB (with the exception of 12KB mode with non-NVDS). User code accesses the lower 8KB of Flash, leaving the upper 4KB for proprietary (for Zilog-only) memory. The NVDS is implemented by using this proprietary memory space for special-purpose routines and for the data required by these routines, which are factory-programmed and cannot be altered by the user. The NVDS operation is described in detail in *the [Nonvolatile Data Storage](#) chapter on page 134*.

The NVDS routines are triggered by a user code: CALL into proprietary memory. Code executing from this proprietary memory must be able to read and write other locations within proprietary memory. User code must not be able to read or write proprietary memory.

## Flash Control Register Definitions

This section defines the features of the following Flash Control registers.

Flash Control Register: see page 119

Flash Status Register: see page 120

Flash Page Select Register: see page 121

Flash Sector Protect Register: see page 122

Flash Frequency High and Low Byte Registers: see page 123

## Option Bit Types

This section describes the two types of Flash option bits offered in the F0830 Series.

### User Option Bits

The user option bits are contained in the first two bytes of program memory. User access to these bits is provided because these locations contain application specific device configurations. The information contained here is lost when page 0 of program memory is erased.

### Trim Option Bits

The trim option bits are contained in the information page of the Flash memory. These bits are factory programmed values required to optimize the operation of onboard analog circuitry and cannot be permanently altered by the user. Program memory can be erased without endangering these values. It is possible to alter working values of these bits by accessing the trim bit address and data registers, but these working values are lost after a power loss.

There are 32 bytes of trim data. To modify one of these values, the user code must first write a value between 00H and 1FH into the Trim Bit Address Register. The next write to the Trim Bit Data Register changes the working value of the target trim data byte.

Reading the trim data requires the user code to write a value between 00H and 1FH into the Trim Bit Address Register. The next read from the Trim Bit Data Register returns the working value of the target trim data byte.

---

► **Note:** The trim address range is from information address 20–3F only. The remaining information page is not accessible via the Trim Bit Address and Data registers.

---

During reset, the first 43 system clock cycles perform 43 Flash accesses. The six bits of the counter provide the lower six bits of the Flash memory address. All other address bits are set to 0. The option bit registers use the 6-bit address from the counter as an address and latch the data from the Flash on the positive edge of the IPO clock, allowing for a maximum of 344-bits (43 bytes) of option information to be read from Flash.

Because option information is stored in both the first two bytes of program memory and in the information area of Flash memory, the data must be placed in specific locations to be read correctly. In this case, the first two bytes at addresses 0 and 1 in program memory are read out and the remainder of the bytes are read out of the Flash information area.



► **Note:** The bit values used in Table 85 are set at the factory; no calibration is required.

**Table 86. 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]	<b>Internal Precision Oscillator Trim Byte</b>
IPO_TRIM	Contains trimming bits for the Internal Precision Oscillator.

► **Note:** The bit values used in Table 86 are set at the factory; no calibration is required.

**Table 87. Trim Option Bits at 0003H (TVBO)**

Bit	7	6	5	4	3	2	1	0
Field	Reserved				Reserved	VBO_TRIM		
RESET	U				U	1	0	0
R/W	R/W				R/W	R/W		
Address	Information Page Memory 0023H							
Note: U = Unchanged by Reset. R/W = Read/Write.								

Bit	Description
[7:3]	<b>Reserved</b> These bits are reserved and must be programmed to 11111.
[2]	<b>VBO Trim Values</b>
VBO_TRIM	Contains factory-trimmed values for the oscillator and the VBO.

Table 113. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Op Code(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
AND dst, src	$\text{dst} \leftarrow \text{dst AND src}$	r	r	52	–	*	*	0	–	–	2	3
		r	lr	53							2	4
		R	R	54							3	3
		R	IR	55							3	4
		R	IM	56							3	3
		IR	IM	57							3	4
ANDX dst, src	$\text{dst} \leftarrow \text{dst AND src}$	ER	ER	58	–	*	*	0	–	–	4	3
		ER	IM	59							4	3
ATM	Block all interrupt and DMA requests during execution of the next 3 instructions			2F	–	–	–	–	–	–	1	2
BCLR bit, dst	$\text{dst}[\text{bit}] \leftarrow 0$	r		E2	–	*	*	0	–	–	2	2
BIT p, bit, dst	$\text{dst}[\text{bit}] \leftarrow \text{p}$	r		E2	–	*	*	0	–	–	2	2
BRK	Debugger Break			00	–	–	–	–	–	–	1	1
BSET bit, dst	$\text{dst}[\text{bit}] \leftarrow 1$	r		E2	–	*	*	0	–	–	2	2
BSWAP dst	$\text{dst}[7:0] \leftarrow \text{dst}[0:7]$	R		D5	X	*	*	0	–	–	2	2
BTJ p, bit, src, dst	if $\text{src}[\text{bit}] = \text{p}$ $\text{PC} \leftarrow \text{PC} + \text{X}$		r	F6	–	–	–	–	–	–	3	3
			lr	F7							3	4
BTJNZ bit, src, dst	if $\text{src}[\text{bit}] = 1$ $\text{PC} \leftarrow \text{PC} + \text{X}$		r	F6	–	–	–	–	–	–	3	3
			lr	F7							3	4
BTJZ bit, src, dst	if $\text{src}[\text{bit}] = 0$ $\text{PC} \leftarrow \text{PC} + \text{X}$		r	F6	–	–	–	–	–	–	3	3
			lr	F7							3	4
CALL dst	$\text{SP} \leftarrow \text{SP} - 2$	IRR		D4	–	–	–	–	–	–	2	6
	@SP $\leftarrow$ PC PC $\leftarrow$ dst	DA		D6							3	3
CCF	$\text{C} \leftarrow \sim \text{C}$			EF	*	–	–	–	–	–	1	2

Note: Flags Notation:

\* = Value is a function of the result of the operation.

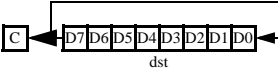
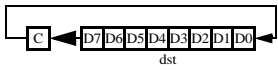
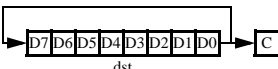
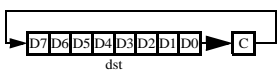
– = Unaffected.

X = Undefined.

0 = Reset to 0.

1 = Set to 1.

Table 113. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Op Code(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
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
RRC dst		R		C0	*	*	*	*	–	–	2	2
		IR		C1							2	3
SBC dst, src	dst ← dst – src – C	r	r	32	*	*	*	*	1	*	2	3
		r	lr	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

Note: Flags Notation:

\* = Value is a function of the result of the operation.

– = Unaffected.

X = Undefined.

0 = Reset to 0.

1 = Set to 1.

# ***Packaging***

Zilog's F0830 Series of MCUs includes the Z8F0130, Z8F0131, Z8F0230, Z8F0231, Z8F1232 and Z8F1233 devices, which are available in the following packages:

- 20-Pin Quad Flat No-Lead Package (QFN)
- 20-pin Small Outline Integrated Circuit Package (SOIC)
- 20-pin Plastic Dual-Inline Package (PDIP)
- 20-pin Small Shrink Outline Package (SSOP)
- 28-Pin Quad Flat No-Lead Package (QFN)
- 28-pin Small Outline Integrated Circuit Package (SOIC)
- 28-pin Plastic Dual-Inline Package (PDIP)
- 28-pin Small Shrink Outline Package (SSOP)

Current diagrams for each of these packages are published in Zilog's Packaging Product Specification (PS0072), which is available free for download from the Zilog website.

# ***Index***

## **Symbols**

@ 165  
 # 165  
 % 165

## **Numerics**

10-bit ADC 4

## **A**

absolute maximum ratings 184  
 AC characteristics 189  
 ADC 166  
     block diagram 99  
     overview 98  
 ADC Channel Register 1 (ADCCTL) 102  
 ADC Data High Byte Register (ADCDH) 103  
 ADC Data Low Bit Register (ADC DL) 103, 104, 105  
 ADCX 166  
 ADD 166  
 add - extended addressing 166  
 add with carry 166  
 add with carry - extended addressing 166  
 additional symbols 165  
 address space 14  
 ADDX 166  
 analog block/PWM signal synchronization 100  
 analog block/PWM signal zynchronization 100  
 analog signals 11  
 analog-to-digital converter  
     overview 98  
 AND 169  
 ANDX 169  
 architecture  
     voltage measurements 98  
 arithmetic instructions 166  
 assembly language programming 162  
 assembly language syntax 163

## **B**

B 165  
 b 164  
 BCLR 167  
 binary number suffix 165  
 BIT 167  
 bit 164  
     clear 167  
     manipulation instructions 167  
     set 167  
     set or clear 167  
     swap 167  
     test and jump 169  
     test and jump if non-zero 169  
     test and jump if zero 169  
 bit jump and test if non-zero 166  
 bit swap 169  
 block diagram 3  
 block transfer instructions 167  
 BRK 169  
 BSET 167  
 BSWAP 167, 169  
 BTJ 169  
 BTJNZ 166, 169  
 BTJZ 169

## **C**

calibration and compensation, motor control measurements 101  
 CALL procedure 169  
 capture mode 89, 90  
 capture/compare mode 89  
 cc 164  
 CCF 168  
 characteristics, electrical 184  
 clear 168  
 CLR 168  
 COM 169  
 compare 89