



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

#### 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	23
Program Memory Size	1KB (1K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0130sj020sg

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong

# List of Tables

Table 1.	Z8 Encore! F0830 Series Family Part Selection Guide 2
Table 2.	Acronyms and Expansions 6
Table 3.	Z8 Encore! F0830 Series Package Options 7
Table 4.	Signal Descriptions 11
Table 5.	Pin Characteristics (20- and 28-pin Devices) 13
Table 6.	Z8 Encore! F0830 Series Program Memory Maps 15
Table 7.	Z8 Encore! F0830 Series Flash Memory Information Area Map 16
Table 8.	Register File Address Map 17
Table 9.	Reset and Stop Mode Recovery Characteristics and Latency 22
Table 10.	Reset Sources and Resulting Reset Type 23
Table 11.	Stop Mode Recovery Sources and Resulting Action 27
Table 12.	POR Indicator Values
Table 13.	Reset Status Register (RSTSTAT) 29
Table 14.	Power Control Register 0 (PWRCTL0) 32
Table 15.	Port Availability by Device and Package Type
Table 16.	Port Alternate Function Mapping
Table 17.	GPIO Port Registers and Subregisters
Table 18.	Port A–D GPIO Address Registers (PxADDR) 40
Table 19.	Port Control Subregister Access 40
Table 20.	Port A–D Control Registers (PxCTL) 41
Table 21.	Port A–D Data Direction Subregisters (PxDD) 41
Table 22.	Port A–D Alternate Function Subregisters (PxAF)
Table 23.	Port A–D Output Control Subregisters (PxOC) 43
Table 24.	Port A–D High Drive Enable Subregisters (PxHDE)
Table 25.	Port A–D Stop Mode Recovery Source Enable Subregisters (PxSMRE) 45
Table 26.	Port A–D Pull-Up Enable Subregisters (PxPUE)
Table 27.	Port A–D Alternate Function Set 1 Subregisters (PxAFS1) 47
Table 28.	Port A–D Alternate Function Set 2 Subregisters (PxAFS2)

# Z8 Encore!<sup>®</sup> F0830 Series Product Specification

## xvi

Table 119.	Flash Memory Electrical Characteristics and Timing 192
Table 120.	Watchdog Timer Electrical Characteristics and Timing 192
Table 121.	Nonvolatile Data Storage
Table 122.	Analog-to-Digital Converter Electrical Characteristics and Timing 193
Table 123.	Comparator Electrical Characteristics
Table 124.	GPIO Port Input Timing
Table 125.	GPIO Port Output Timing 196
Table 126.	On-Chip Debugger Timing 197
Table 127.	Power Consumption Reference Table
Table 128.	Z8 Encore! XP F0830 Series Ordering Matrix 200
Table 129.	Package and Pin Count Description 207
Table 130.	Timer 0 High Byte Register (T0H) 208
Table 131.	Timer 0 Low Byte Register (T0L) 209
Table 132.	Timer 0 Reload High Byte Register (T0RH) 209
Table 133.	Timer 0 Reload Low Byte Register (T0RL)
Table 134.	Timer 0 PWM High Byte Register (T0PWMH) 209
Table 135.	Timer 0 PWM Low Byte Register (T0PWML) 210
Table 136.	Timer 0 Control Register 0 (T0CTL0)
Table 137.	Timer 0 Control Register 1 (T0CTL1)
Table 138.	Timer 1 High Byte Register (T1H) 210
Table 139.	Timer 1 Low Byte Register (T1L) 211
Table 140.	Timer 1 Reload High Byte Register (T1RH) 211
Table 141.	Timer 1 Reload Low Byte Register (T1RL)
Table 142.	Timer 1 PWM High Byte Register (T1PWMH) 211
Table 143.	Timer 1 PWM Low Byte Register (T1PWML)
Table 144.	Timer 1 Control Register 0 (T1CTL0)
Table 145.	Timer 1 Control Register 1 (T1CTL1)
Table 146.	ADC Control Register 0 (ADCCTL0)
Table 147.	ADC Data High Byte Register (ADCD_H) 214
Table 148.	ADC Data Low Bits Register (ADCD_L) 214

The analog supply pins (AV<sub>DD</sub> and AV<sub>SS</sub>) are also not available on these parts and are replaced by PB6 and PB7.

At reset, by default, all pins of Port A, B and C are in Input state. The alternate functionality is also disabled, so the pins function as general purpose input ports until programmed otherwise. At power-up, the Port D0 pin defaults to the RESET Alternate function.

The pin configurations listed are preliminary and subject to change based on manufacturing limitations.



Figure 2. Z8F0830 Series in 20-Pin SOIC, SSOP, PDIP Package



Figure 3. Z8F0830 Series in 28-Pin SOIC, SSOP, PDIP Package

clock and reset signals, the required reset duration may be three or four clock periods. A reset pulse of three clock cycles in duration might trigger a reset and a reset pulse of four cycles in duration always triggers a reset.

While the  $\overline{\text{RESET}}$  input pin is asserted low, the Z8 Encore! F0830 Series devices remain in the Reset state. If the  $\overline{\text{RESET}}$  pin is held low beyond the system reset time-out, the device exits the Reset state on the system clock rising edge following  $\overline{\text{RESET}}$  pin deassertion. Following a system reset initiated by the external  $\overline{\text{RESET}}$  pin, the EXT status bit in the Reset Status (RSTSTAT) Register is set to 1.

## **External Reset Indicator**

During system reset or when enabled by the GPIO logic, the RESET pin functions as an open-drain (active low) RESET mode indicator in addition to the input functionality. This reset output feature allows an Z8 Encore! F0830 Series device to reset other components to which it is connected, even if that reset is caused by internal sources such as POR, VBO or WDT events. See the <u>Port A–D Control Registers</u> section on page 41.

After an internal Reset event occurs, the internal circuitry begins driving the RESET pin low. The  $\overrightarrow{\text{RESET}}$  pin is held low by the internal circuitry until the appropriate delay listed in <u>Table 9</u> (see page 22) has elapsed.

# **On-Chip Debugger Initiated Reset**

A Power-On Reset can be initiated using the On-Chip Debugger by setting the RST bit in the OCD Control Register. The OCD block is not reset, but the remainder of the chip goes through a normal system reset. The RST bit automatically clears during the system reset. Following the system reset, the POR bit in the Reset Status (RSTSTAT) Register is set.

# Stop Mode Recovery

The device enters the STOP Mode when the STOP instruction is executed by the eZ8 CPU. See the <u>Low-Power Modes</u> chapter on page 30 for detailed STOP Mode information. During Stop Mode Recovery, the CPU is held in reset for about 66 IPO cycles if the crystal oscillator is disabled or about 5000 cycles if it is enabled.

Stop Mode Recovery does not affect the on-chip registers other than the Reset Status (RSTSTAT) Register and the Oscillator Control Register (OSCCTL). After any Stop Mode Recovery, the IPO is enabled and selected as the system clock. If another system clock source is required or IPO disabling is required, the Stop Mode Recovery code must reconfigure the oscillator control block such that the correct system clock source is enabled and selected.

# Architecture

Figure 8 displays a simplified block diagram of a GPIO port pin. In this figure, the ability to accommodate alternate functions and variable port current drive strength is not displayed.





# **GPIO Alternate Functions**

Many of the GPIO port pins can be used for general purpose input/output and access to onchip peripheral functions such as the timers and serial communication devices. The Port A–D Alternate Function subregisters configure these pins for either GPIO or Alternate function operation. When a pin is configured for Alternate function, control of the port pin direction (input/output) is passed from the Port A–D data direction registers to the Alternate function assigned to this pin. <u>Table 16</u> on page 36 lists the alternate functions possible with each port pin. The alternate function associated at a pin is defined through Alternate Function subregisters AFS1 and AFS2.

The crystal oscillator functionality is not controlled by the GPIO block. When the crystal oscillator is enabled in the oscillator control block, the GPIO functionality of PA0 and PA1 is overridden. In that case, pins PA0 and PA1 functions as input and output for the crystal oscillator.

# 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.

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							

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

Bit	Description
[7:0]	<b>Port Control</b>
PCTL	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.

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	lf 01H ii	n Port A–D A	Address Reg	gister, acces	sible throug	h the Port A	–D Control I	Register

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

Bit	Description
[7:0]	Data Direction
DDx	<ul> <li>These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction Register setting.</li> <li>0 = Output. Data in the Port A–D Output Data Register is driven onto the port pin.</li> <li>1 = Input. The port pin is sampled and the value written into the Port A–D Input Data Register. The output driver is tristated.</li> </ul>

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

• Writing 1 to the IRQE bit in the Interrupt Control Register

Interrupts are globally disabled by any of the following actions:

- Execution of a DI (disable interrupt) instruction
- eZ8 CPU acknowledgement of an interrupt service request from the Interrupt Controller
- Writing a 0 to the IRQE bit in the Interrupt Control Register
- Reset
- Execution of a trap instruction
- Illegal instruction Trap
- Primary oscillator fail trap
- Watchdog Oscillator fail 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 are enabled with identical interrupt priority (all as level 2 interrupts, for example), the interrupt priority is assigned from highest to lowest as specified in <u>Table 34</u> on page 54. Level 3 interrupts are always assigned higher priority than level 2 interrupts and level 2 interrupts are assigned 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 34, above. Reset, Watchdog Timer interrupt (if enabled), primary oscillator fail trap, Watchdog Oscillator fail trap and illegal instruction trap always have highest (level 3) 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. Writing 0 to the corresponding bit in the interrupt request register clears the interrupt request.

**Caution:** Zilog recommends not using a coding style that clears bits in the Interrupt Request registers. All incoming interrupts received between execution of the first LDX command and the final LDX command are lost. See Example 1, which follows.

**Example 1.** A poor coding style that can result in lost interrupt requests:

# **Timers**

The Z8 Encore! F0830 Series products contain up to 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 ranging 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

# Architecture

Figure 10 displays the architecture of the timers.

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:

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:

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 COUNTER Mode and for initiating the count:

- 1. Write to the Timer Control Register to:
  - Disable the timer
  - Configure the timer for COUNTER Mode
  - Select either the rising edge or falling edge of the timer input signal for the count. This selection also sets the initial logic level (High or Low) for the timer output alternate function. However, the timer output function is not required to be enabled.
- 2. Write to the Timer High and Low Byte registers to set the starting count value. This only affects the first pass in COUNTER Mode. After the first timer reload in COUNTER Mode, counting always begins at the reset value 0001H. In COUNTER Mode, the Timer High and Low Byte registers must be written with the value 0001H.
- 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. Configure the associated GPIO port pin for the timer input alternate function.
- 6. If using the timer output function, configure the associated GPIO port pin for the timer output alternate function.
- 7. Write to the Timer Control Register to enable the timer.

In COUNTER Mode, the number of timer input transitions is calculated with the following equation:

Counter Mode Timer Input Transitions = Current Count Value – Start Value

#### **COMPARATOR COUNTER Mode**

In COMPARATOR COUNTER Mode, the timer counts the input transitions from the analog comparator output. The TPOL bit in the Timer Control Register determines whether the count occurs on the rising edge or the falling edge of the comparator output signal. In COMPARATOR COUNTER Mode, the prescaler is disabled.

- 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 user software to determine if interrupts are 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 were 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. The user can configure the timer interrupt to be generated only at the input capture event or the reload event by setting the TICONFIG field of the TxCTL1 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 between the timer start and the capture event can be calculated using the following equation:

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

#### **COMPARE Mode**

In COMPARE Mode, the timer counts up to 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). Additionally, 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 resets to 0000H and continues counting.

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

- 1. Write to the Timer Control Register to:
  - Disable the timer
  - Configure the timer for COMPARE Mode
  - Set the prescale value
  - Set the initial logic level (High or Low) for 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 compare value.

# Watchdog Timer Refresh

Upon first enable, the Watchdog Timer is loaded with the value in the Watchdog Timer Reload registers. The Watchdog Timer 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 Z8 Encore! F0830 Series devices are operating in DEBUG Mode (using the On-Chip Debugger), the Watchdog Timer must be continuously refreshed to prevent any WDT 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 system reset. The WDT\_RES Flash option bit determines the time-out response of the Watchdog Timer. See *the* <u>Flash Option</u> <u>Bits</u> chapter on page 124 for information about programming the WDT\_RES Flash option bit.

### **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 Reset Status 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 resets to its maximum value of FFFFFH and continues counting. The Watchdog Timer counter will not automatically return to its reload value.

The Reset Status Register (see <u>Table 12</u> on page 29) must be read before clearing the WDT interrupt. This read clears the WDT time-out flag and prevents further WDT interrupts occurring immediately.

### WDT Interrupt in STOP Mode

If configured to generate an interrupt when a time-out occurs and the Z8 Encore! F0830 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 a WDT time-out in STOP Mode. See *the* <u>Reset and Stop Mode Recovery</u> *chapter on page 21* for more information about Stop Mode Recovery operations.

If interrupts are enabled, following completion of the Stop Mode Recovery, the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executes the code from the vector address.

# ADC Data High Byte Register

The ADC Data High Byte Register, listed in Table 64, contains the upper eight bits of the ADC output. Access to the ADC Data High Byte Register is read-only. Reading the ADC Data High Byte Register latches data in the ADC Low Bits Register.

#### Table 64. ADC Data High Byte Register (ADCD\_H)

Bit	7	6	5	4	3	2	1	0
Field				ADC	CDH			
RESET		Х						
R/W	R							
Address	F72H							

Bit	Description
[7:0]	ADC High Byte
ADCDH	00h–FFh = The last conversion output is held in the data registers until the next ADC conversion is completed.

# ADC Data Low Bits Register

The ADC Data Low Bits Register, shown in Table 65, contains the lower bits of the ADC output. Access to the ADC Data Low Bits Register is read-only. Reading the ADC Data High Byte Register latches lower bits of the ADC in the ADC Data Low Bits Register.

Bit	7	6	5	4	3	2	1	0
Field	ADO	CDL	Reserved					
RESET	Х		Х					
R/W	F	२	R					
Address				F7	3H			

Table 65.	. ADC Data	Low Bits	Register	(ADCD_	<u>L)</u>
-----------	------------	----------	----------	--------	-----------

Bit	Description
[7:6] ADCDL	ADC Low Bits 00–11b = These bits are the two least-significant bits of the 10-bit ADC output. These bits are undefined after a reset. The low bits are latched into this register whenever the ADC Data High Byte Register is read.
[5:0]	Reserved These bits are reserved and must be programmed to 000000.

Debug Command	Command Byte	Enabled when not in DEBUG Mode?	Disabled by Flash Read Protect Option Bit
Read Program Memory CRC	0EH	-	_
Reserved	0FH	-	_
Step Instruction	10H	-	Disabled
Stuff Instruction	11H	-	Disabled
Execute Instruction	12H	_	Disabled
Reserved	13H–FFH	-	-

#### Table 95. On-Chip Debugger Command Summary (Continued)

In the following bulleted list of OCD commands, data and commands sent from the host to the OCD are identified by DBG  $\leftarrow$  Command/Data. Data sent from the OCD back to the host is identified by DBG Data.

**Read OCD Revision (00H).** The read OCD revision command determines the version of the On-Chip Debugger. If OCD commands are added, removed or changed this revision number changes.

```
DBG \leftarrow 00H
DBG \rightarrow OCDRev[15:8] (Major revision number)
DBG \rightarrow OCDRev[7:0] (Minor revision number)
```

**Read OCD Status Register (02H).** The read OCD Status Register command reads the OCDSTAT register.

```
DBG \leftarrow 02H
DBG \rightarrow OCDSTAT[7:0]
```

**Read Runtime Counter (03H).** The runtime counter counts system clock cycles in between breakpoints. The 16-bit runtime counter counts from 0000H and stops at the maximum count of FFFFH. The runtime counter is overwritten during the write memory, read memory, write register, read register, read memory CRC, step instruction, stuff instruction and execute instruction commands.

```
DBG \leftarrow 03H
DBG \rightarrow RuntimeCounter[15:8]
DBG \rightarrow RuntimeCounter[7:0]
```

Write OCD Control Register (04H). The write OCD Control Register command writes the data that follows to the OCDCTL register. When the Flash read protect option bit is enabled, the DBGMODE bit (OCDCTL[7]) can only be set to 1, it cannot be cleared to 0. To return the device to normal operating mode, the device must be reset.

```
DBG \leftarrow 04H
DBG \leftarrow OCDCTL[7:0]
```

# **AC Characteristics**

The section provides information about the AC characteristics and timing. All AC timing information assumes a standard load of 50pF on all outputs.

		V <sub>DD</sub> = 2.7 to 3.6V T <sub>A</sub> = 0°C to +70°C		$V_{DD} = 2.7 \text{ to } 3.6 \text{V}$ $T_A = -40^{\circ}\text{C} \text{ to}$ +105°C				
Symbol	Parameter	Min	Max	Min	Max	Units	Conditions	
F <sub>SYSCLK</sub>	System Clock Fre- quency			-	20.0	MHz	Read-only from Flash memory	
				0.03276 8	20.0	MHz	Program or erasure of the Flash memory	
F <sub>XTAL</sub>	Crystal Oscillator Frequency			1.0	20.0	MHz	System clock frequen- cies below the crystal oscillator minimum require an external	
F <sub>IPO</sub>	Internal Precision Oscillator Frequency			0.03276 8	5.5296	MHz	Oscillator is <b>not</b> adjust- able over the entire range. User may select Min or Max value only.	
F <sub>IPO</sub>	Internal Precision Oscillator Frequency			5.31	5.75	MHz	High speed with trim- ming	
F <sub>IPO</sub>	Internal Precision Oscillator Frequency			4.15	6.91	MHz	High speed without trimming	
F <sub>IPO</sub>	Internal Precision Oscillator Frequency			30.7	33.3	KHz	Low speed with trim- ming	
F <sub>IPO</sub>	Internal Precision Oscillator Frequency			24	40	KHz	Low speed without trimming	
T <sub>XIN</sub>	System Clock Period			50	-	ns	T <sub>CLK</sub> = 1/F <sub>syscik</sub>	
T <sub>XINH</sub>	System Clock High Time			20	30	ns	T <sub>CLK</sub> = 50 ns	
T <sub>XINL</sub>	System Clock Low Time			20	30	ns	T <sub>CLK</sub> = 50 ns	

#### **Table 117. AC Characteristics**

Part Number	Flash	RAM	NVDS	ADC Channels	Description
Z8F0831HH020EG	8KB	256	Yes	0	SSOP 20-pin
Z8F0831PH020EG	8KB	256	Yes	0	PDIP 20-pin
Z8F0831QH020EG	8KB	256	Yes	0	QFN 20-pin
Z8F0830SJ020EG	8KB	256	Yes	8	SOIC 28-pin
Z8F0830HJ020EG	8KB	256	Yes	8	SSOP 28-pin
Z8F0830PJ020EG	8KB	256	Yes	8	PDIP 28-pin
Z8F0830QJ020EG	8KB	256	Yes	8	QFN 28-pin
Z8F0831SJ020EG	8KB	256	Yes	0	SOIC 28-pin
Z8F0831HJ020EG	8KB	256	Yes	0	SSOP 28-pin
Z8F0831PJ020EG	8KB	256	Yes	0	PDIP 28-pin
Z8F0831QJ020EG	8KB	256	Yes	0	QFN 28-pin
Z8 Encore! F0830 with	4KB Flash	1			
Standard Temperature	: 0°C to 70	°C			
Z8F0430SH020SG	4KB	256	Yes	7	SOIC 20-pin
Z8F0430HH020SG	4KB	256	Yes	7	SSOP 20-pin
Z8F0430PH020SG	4KB	256	Yes	7	PDIP 20-pin
Z8F0430QH020SG	4KB	256	Yes	7	QFN 20-pin
Z8F0431SH020SG	4KB	256	Yes	0	SOIC 20-pin
Z8F0431HH020SG	4KB	256	Yes	0	SSOP 20-pin
Z8F0431PH020SG	4KB	256	Yes	0	PDIP 20-pin
Z8F0431QH020SG	4KB	256	Yes	0	QFN 20-pin
Z8F0430SJ020SG	4KB	256	Yes	8	SOIC 28-pin
Z8F0430HJ020SG	4KB	256	Yes	8	SSOP 28-pin
Z8F0430PJ020SG	4KB	256	Yes	8	PDIP 28-pin
Z8F0430QJ020SG	4KB	256	Yes	8	QFN 28-pin
Z8F0431SJ020SG	4KB	256	Yes	0	SOIC 28-pin
Z8F0431HJ020SG	4KB	256	Yes	0	SSOP 28-pin
Z8F0431PJ020SG	4KB	256	Yes	0	PDIP 28-pin
Z8F0431QJ020SG	4KB	256	Yes	0	QFN 28-pin
Extended Temperature	: -40°C to	105°C			
Z8F0430SH020EG	4KB	256	Yes	7	SOIC 20-pin
Z8F0430HH020EG	4KB	256	Yes	7	SSOP 20-pin
Z8F0430PH020EG	4KB	256	Yes	7	PDIP 20-pin

## Table 128. Z8 Encore! XP F0830 Series Ordering Matrix

					-
Part Number	Flach	DAM		ADC Channala	Description
	FIDSII	RAIVI	NVD3	Channels	Description
Z8F0131PJ020SG	1KB	256	Yes	0	PDIP 28-pin
Z8F0131QJ020SG	1KB	256	Yes	0	QFN 28-pin
Extended Temperature	e: -40°C to	105°C			
Z8F0130SH020EG	1KB	256	Yes	7	SOIC 20-pin
Z8F0130HH020EG	1KB	256	Yes	7	SSOP 20-pin
Z8F0130PH020EG	1KB	256	Yes	7	PDIP 20-pin
Z8F0130QH020EG	1KB	256	Yes	7	QFN 20-pin
Z8F0131SH020EG	1KB	256	Yes	0	SOIC 20-pin
Z8F0131HH020EG	1KB	256	Yes	0	SSOP 20-pin
Z8F0131PH020EG	1KB	256	Yes	0	PDIP 20-pin
Z8F0131QH020EG	1KB	256	Yes	0	QFN 20-pin
Z8F0130SJ020EG	1KB	256	Yes	8	SOIC 28-pin
Z8F0130HJ020EG	1KB	256	Yes	8	SSOP 28-pin
Z8F0130PJ020EG	1KB	256	Yes	8	PDIP 28-pin
Z8F0130QJ020EG	1KB	256	Yes	8	QFN 28-pin
Z8F0131SJ020EG	1KB	256	Yes	0	SOIC 28-pin
Z8F0131HJ020EG	1KB	256	Yes	0	SSOP 28-pin
Z8F0131PJ020EG	1KB	256	Yes	0	PDIP 28-pin
Z8F0131QJ020EG	1KB	256	Yes	0	QFN 28-pin
ZUSBSC00100ZACG					USB Smart Cable Accessory Kit
ZUSBOPTSC01ZACG					Opto-Isolated USB Smart Cable Accessory Kit

Table 128. Z8 Encore! XP F0830 Series Ordering Matrix

# **Part Number Suffix Designations**

Zilog part numbers consist of a number of components, as indicated in the following example.

**Example.** Part number Z8F0830SH020SG is an 8-bit 20MHz Flash MCU with 8KB Program Memory and equipped with ADC and NVDS in a 20-pin SOIC package, operating within a 0°C to +70°C temperature range and built using lead-free solder.

## Hex Address: F0D

#### Table 143. Timer 1 PWM Low Byte Register (T1PWML)

Bit	7	6	5	4	3	2	1	0	
Field	PWML								
RESET	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Address	FODH								

## Hex Address: F0E

#### Table 144. Timer 1 Control Register 0 (T1CTL0)

Bit	7	6	5	4	3	2	1	0	
Field	TMODEHI	TICO	NFIG	Reserved	Reserved PWMD		INPCAP		
RESET	0	0	0	0	0	0	0	0	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Address	FOEH								

## Hex Address: F0F

#### Table 145. Timer 1 Control Register 1 (T1CTL1)

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	F0FH							

## Hex Addresses: F10–F6F

This address range is reserved.

#### Z8 Encore!<sup>®</sup> F0830 Series Product Specification

page select register 121, 122 FPS register 121, 122 FSTAT register 120

## G

gated mode 89 general-purpose I/O 33 GPIO 4.33 alternate functions 34 architecture 34 control register definitions 39 input data sample timing 195 interrupts 39 port A-C pull-up enable sub-registers 46, 47, 48 port A-H address registers 40 port A-H alternate function sub-registers 42 port A-H control registers 41 port A-H data direction sub-registers 41 port A-H high drive enable sub-registers 44 port A-H input data registers 49 port A-H output control sub-registers 43 port A-H output data registers 50, 51 port A-H stop mode recovery sub-registers 45 port availability by device 33 port input timing 195 port output timing 196

## Η

H 165 HALT 168 halt mode 31, 168 hexadecimal number prefix/suffix 165

# I

IM 164 immediate data 164 immediate operand prefix 165 INC 166 increment 166 increment word 167 INCW 167 indexed 165 indirect address prefix 165 indirect register 164 indirect register pair 164 indirect working register 164 indirect working register pair 164 instruction set, ez8 CPU 162 instructions ADC 166 **ADCX 166** ADD 166 **ADDX 166** AND 169 **ANDX 169** arithmetic 166 **BCLR 167** BIT 167 bit manipulation 167 block transfer 167 **BRK 169 BSET 167** BSWAP 167, 169 BTJ 169 BTJNZ 166, 169 **BTJZ 169 CALL 169** CCF 167, 168 CLR 168 COM 169 CP 166 **CPC 166 CPCX 166** CPU control 168 CPX 166 DA 166 **DEC 166 DECW 166** DI 168 **DJNZ 169** EI 168 **HALT 168 INC 166 INCW 167 IRET 169** 

#### 233

watchdog timer reload upper byte (WDTU) 96 register file 14 register pair 165 register pointer 165 registers ADC channel 1 102 ADC data high byte 103 ADC data low bit 103, 104, 105 reset and stop mode characteristics 22 and stop mode recovery 21 carry flag 167 sources 23 **RET 169** return 169 RL 169 **RLC 169** rotate and shift instuctions 169 rotate left 169 rotate left through carry 169 rotate right 170 rotate right through carry 170 RP 165 RR 165, 170 rr 165 **RRC** 170

# S

SBC 167 SCF 167, 168 second opcode map after 1FH 183 set carry flag 167, 168 set register pointer 168 shift right arithmatic 170 shift right logical 170 signal descriptions 11 software trap 169 source operand 165 SP 165 SRA 170 src 165 SRL 170 SRP 168 stack pointer 165 **STOP 168** stop mode 30, 168 stop mode recovery sources 26 using a GPIO port pin transition 27, 28 using watch-dog timer time-out 27 **SUB 167** subtract 167 subtract - extended addressing 167 subtract with carry 167 subtract with carry - extended addressing 167 **SUBX 167 SWAP 170** swap nibbles 170 symbols, additional 165

# Т

Table 134. Power Consumption Reference Table 197 **TCM 167 TCMX 167** test complement under mask 167 test complement under mask - extended addressing 167 test under mask 167 test under mask - extended addressing 167 tiing diagram, voltage measurement 100 timer signals 11 timers 68 architecture 68 block diagram 69 capture mode 77, 78, 89, 90 capture/compare mode 81, 89 compare mode 79, 89 continuous mode 70, 89 counter mode 71.72 counter modes 89 gated mode 80, 89 one-shot mode 69, 89 operating mode 69 PWM mode 74, 75, 89, 90 reading the timer count values 82