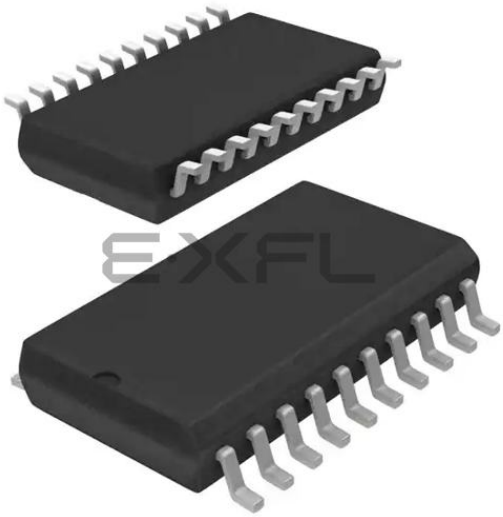

Welcome to E-XFL.COM

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	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	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 7x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0130sh020sg

Data Memory

The Z8 Encore! F0830 Series does not use the eZ8 CPU's 64KB data memory address space.

Flash Information Area

Table 7 maps the Z8 Encore! F0830 Series Flash information area. The 128-byte information area is accessed, by setting bit 7 of the Flash Page Select Register to 1. When access is enabled, the Flash information area is mapped into program memory and overlays these 128 bytes at addresses FE00H to FE7FH. When information area access is enabled, all reads from these program memory addresses return information area data rather than program memory data. Access to the Flash information area is read-only.

Table 7. Z8 Encore! F0830 Series Flash Memory Information Area Map

Program Memory Address (Hex)	Function
FE00–FE3F	Zilog option bits
FE40–FE53	Part Number 20-character ASCII alphanumeric code Left-justified and filled with FH
FE54–FE5F	Reserved
FE60–FE7F	Reserved
FE80–FFFF	Reserved

The Voltage Brown-Out circuit can be either enabled or disabled during STOP Mode. Operations during STOP Mode is set by the VBO_AO Flash option bit. See the Flash Option Bits chapter on page 124 for information about configuring VBO_AO.

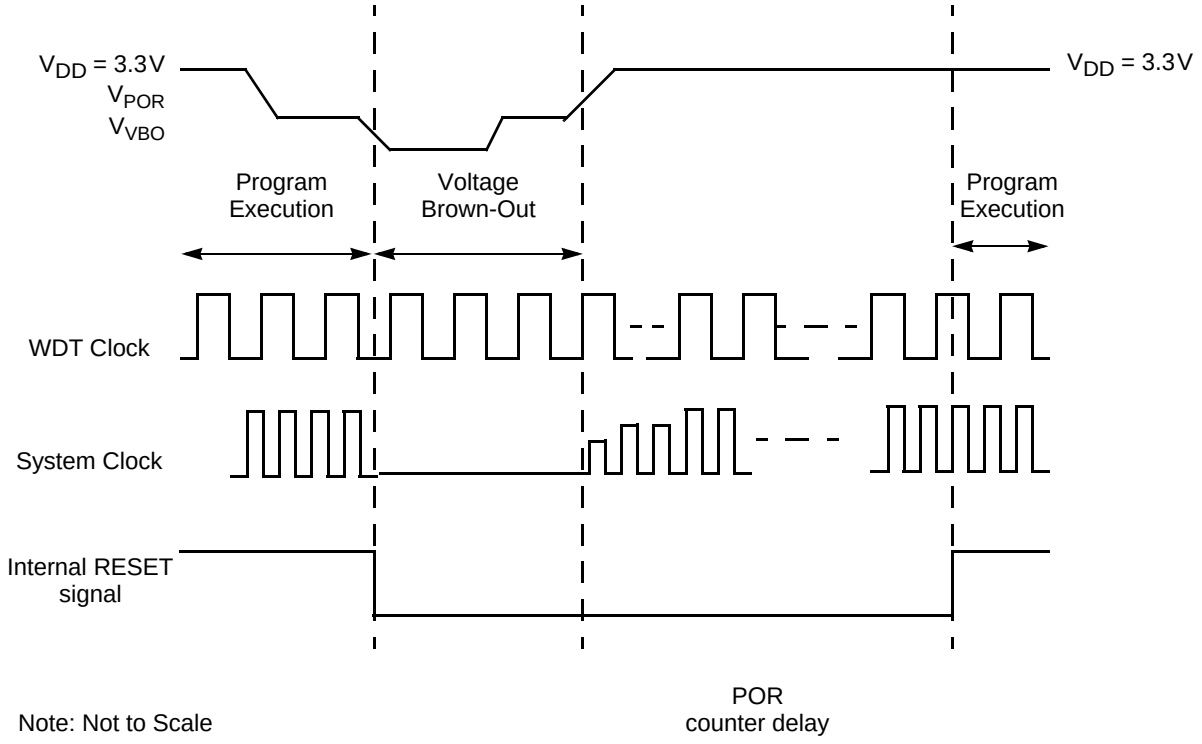

Figure 7. Voltage Brown-Out Reset Operation

Watchdog Timer Reset

If the device is operating in NORMAL or STOP Mode, the Watchdog Timer can initiate a system reset at time-out if the WDT_RES Flash option bit is programmed to 1; this state 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 status bit in the Reset Status (RSTSTAT) Register is set to 1 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. After 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

The eZ8 CPU fetches the reset vector at program memory addresses 0002H and 0003H and loads that value into the program counter. Program execution begins at the reset vector address. Following Stop Mode Recovery, the STOP bit in the Reset Status (RSTSTAT) Register is set to 1. Table 11 lists the Stop Mode Recovery sources and resulting actions. The following sections provide more details about each of the Stop Mode Recovery sources.

Table 11. Stop Mode Recovery Sources and Resulting Action

Operating Mode	Stop Mode Recovery Source	Action
STOP Mode	Watchdog Timer time-out when configured for Reset	Stop Mode Recovery
	Watchdog Timer time-out when configured for interrupt	Stop Mode Recovery followed by interrupt (if interrupts are enabled)
	Data transition on any GPIO port pin enabled as a Stop Mode Recovery source	Stop Mode Recovery
	Assertion of external $\overline{\text{RESET}}$ Pin	System reset
	Debug pin driven Low	System reset

Stop Mode Recovery using WDT Time-Out

If the Watchdog Timer times out during STOP Mode, the device undergoes a Stop Mode Recovery sequence. In the Reset Status (RSTSTAT) Register, the WDT and STOP bits are set to 1. If the Watchdog Timer is configured to generate an interrupt upon time-out and the Z8 Encore! F0830 Series device is configured to respond to interrupts, the eZ8 CPU services the WDT interrupt request following the normal Stop Mode Recovery sequence.

Stop Mode Recovery using GPIO Port Pin Transition

Each of the GPIO port pins may be configured as a Stop Mode Recovery input source. If any GPIO pin is enabled as a Stop Mode Recovery source, a change in the input pin value (from High to Low or from Low to High) initiates Stop Mode Recovery. In the Reset Status (RSTSTAT) Register, the STOP bit is set to 1.

Caution: In STOP Mode, the GPIO Port Input Data registers (PxIN) are disabled. These Port Input Data registers record the port transition only if the signal stays on the port pin through the end of the Stop Mode Recovery delay. As a result, short pulses on the port pin can initiate Stop Mode Recovery without being written to the Port Input Data Register or without initiating an interrupt (if enabled for that pin).

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.

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

To avoid missing interrupts, use the coding style in Example 2 to clear bits in the Interrupt Request 0 Register:

Example 2. A good coding style that avoids lost interrupt requests:

```
ANDX IRQ0, MASK
```

Software Interrupt Assertion

Program code can generate interrupts directly. Writing 1 to the correct bit in the interrupt request register triggers an interrupt (assuming that interrupt is enabled). When the interrupt request is acknowledged by the eZ8 CPU, the bit in the interrupt request register is automatically cleared to 0.

! Caution: Zilog recommends not using a coding style to generate software interrupts by setting 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 3, which follows.

Example 3. A poor coding style that can result in lost interrupt requests:

```
LDX r0, IRQ0
OR r0, MASK
LDX IRQ0, r0
```

To avoid missing interrupts, use the coding style in Example 4 to set bits in the Interrupt Request registers:

Example 4. A good coding style that avoids lost interrupt requests:

```
ORX IRQ0, MASK
```

Interrupt Control Register Definitions

The Interrupt Control registers enable individual interrupts, set interrupt priorities and indicate interrupt requests for all of the interrupts other than the Watchdog Timer interrupt, the primary oscillator fail trap and the Watchdog Oscillator fail trap interrupts.

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

1. Write to the Timer Control Register to:
   - Disable the timer
   - Configure the timer for PWM DUAL OUTPUT Mode; setting the mode also involves writing to TMODEHI bit in the TxCTL1 Register
   - 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 write 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 PWM Control Register to set the PWM deadband delay value. The deadband delay must be less than the duration of the positive phase of the PWM signal (as defined by the PWM High and Low Byte registers). It must also be less than the 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 will automatically switch 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:

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

If an initial starting value other than 0001H is loaded into the Timer High and Low Byte registers, the ONE-SHOT Mode equation 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:

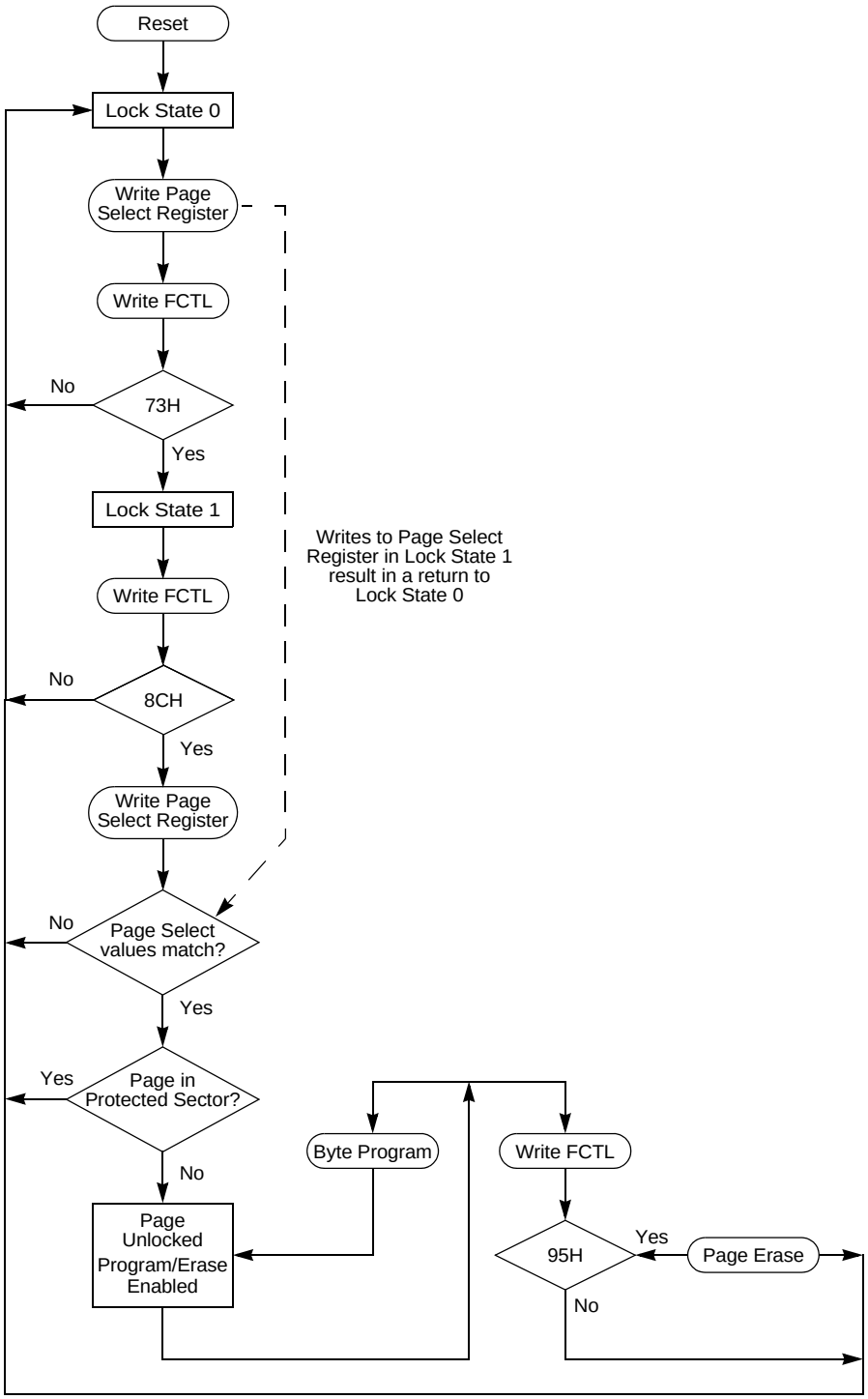

Figure 19. Flash Controller Operation Flow Chart

Flash Sector Protect Register

The Flash Sector Protect Register is shared with the Flash Page Select Register. When the Flash Control Register is locked and written with 5EH, the next write to this address targets the Flash Sector Protect Register. In all other cases, it targets the Flash Page Select Register.

This register selects one of the eight available Flash memory sectors to be protected. The Reset state of each sector protect bit is the zero (unprotected) state. After a sector is protected by setting its corresponding register bit, the register bit cannot be cleared by the user.

To determine the appropriate Flash memory sector address range and sector number for your F0830 Series product, please refer to Table 70 on page 112.

Table 75. Flash Sector Protect Register (FPROT)

Bit	7	6	5	4	3	2	1	0
Field	SPROT7	SPROT6	SPROT5	SPROT4	SPROT3	SPROT2	SPROT1	SPROT0
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	FF9H							

Bit	Description
[7:0] SPROT*x*	**Sector Protection** For Z8F12xx, Z8F08xx and Z8F04xx devices, all bits are used. For Z8F02xx devices, the upper four bits remain unused. For Z8F01xx devices, the upper six bits remain unused. To determine the appropriate Flash memory sector address range and sector number for your F0830 Series product, please refer to Table 69 and to Figures 14 through 18.

Note: *x* indicates bits in the range 7–0.

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.

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 (`0x20B3`). At the return from the subroutine, the write status byte resides in working register R0. The bit fields of this status byte are defined in Table 91. Additionally, user code should pop the address and data bytes off the stack.

The write routine uses 16 bytes of stack space in addition to the two bytes of address and data pushed by the user code. Sufficient memory must be available for this stack usage.

Because of the Flash memory architecture, NVDS writes exhibit a nonuniform execution time. In general, a write takes 136µs (assuming a 20MHz system clock). For every 200 writes, however, a maintenance operation is necessary. In this rare occurrence, the write takes up to 58ms to complete. Slower system clock speeds result in proportionally higher execution times.

NVDS byte writes to invalid addresses (those exceeding the NVDS array size) have no effect. Illegal write operations have a 7µs execution time.

Table 91. Write Status Byte

Bit	7	6	5	4	3	2	1	0
Field	Reserved					FE	IGADDR	WE
Default Value	0	0	0	0	0	0	0	0

Bit	Description
[7:3]	**Reserved** These bits are reserved and must be programmed to 00000.
[2] FE	**Flash Error** If a Flash error is detected, this bit is set to 1.
[1] IGADDR	**Illegal Address** When an NVDS byte writes to invalid addresses occur (those exceeding the NVDS array size), this bit is set to 1.
[0] WE	**Write Error** A failure occurs during data writes to Flash. When writing data into a certain address, a read-back operation is performed. If the read-back value is not the same as the value written, this bit is set to 1.

Operation

The following section describes the operation of the On-Chip Debugging function.

OCD Interface

The On-Chip Debugger uses the DBG pin for communication with an external host. This one-pin interface is a bidirectional open-drain interface that transmits and receives data. Data transmission is half-duplex, which means that transmission and data retrieval cannot occur simultaneously. The serial data on the DBG pin is sent using the standard asynchronous data format defined in RS-232.This pin creates an interface between the Z8 Encore! F0830 Series products and the serial port of a host PC using minimal external hardware. Two different methods for connecting the DBG pin to an RS-232 interface are displayed in Figures 21 and 22. The recommended method is the buffered implementation depicted in Figure 22. The DBG pin must always be connected to V_{DD} through an external pull-up resistor.

! Caution: For proper operation of the On-Chip Debugger, all power pins (V_{DD} and AV_{DD}) must be supplied with power and all ground pins (V_{SS} and AV_{SS}) must be properly grounded. The DBG pin is open-drain and must always be connected to V_{DD} through an external pull-up resistor to ensure proper operation.

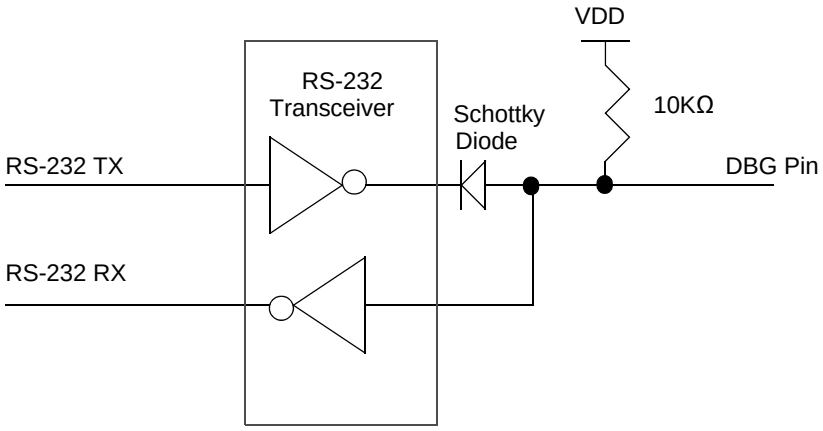

Figure 21. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface, #1 of 2

Crystal Oscillator

The products in the Z8 Encore! F0830 Series contain an on-chip crystal oscillator for use with external crystals with 32kHz to 20MHz frequencies. In addition, the oscillator supports external RC networks with oscillation frequencies up to 4MHz or ceramic resonators with frequencies up to 8MHz. The on-chip crystal oscillator can be used to generate the primary system clock for the internal eZ8 CPU and the majority of its on-chip peripherals. Alternatively, the X_{IN} input pin can also accept a CMOS-level clock input signal (32kHz–20MHz). If an external clock generator is used, the X_{OUT} pin must remain unconnected. The on-chip crystal oscillator also contains a clock filter function. To see the settings for this clock filter, see Table 90 on page 133. By default, however, this clock filter is disabled; therefore, no divide to the input clock (namely, the frequency of the signal on the X_{IN} input pin) can determine the frequency of the system clock when using the default settings.

Note: Although the X_{IN} pin can be used as an input for an external clock generator, the CLKIN pin is better suited for such use. See *the* System Clock Selection *section on page 151* for more information.

Operating Modes

The Z8 Encore! F0830 Series products support the following four OSCILLATOR Modes:

- Minimum power for use with very low frequency crystals (32kHz to 1MHz)
- Medium power for use with medium frequency crystals or ceramic resonators (0.5MHz to 8MHz)
- Maximum power for use with high frequency crystals (8MHz to 20MHz)
- On-chip oscillator configured for use with external RC networks (<4MHz)

The OSCILLATOR Mode is selected using user-programmable Flash option bits. See the Flash Option Bits chapter on page 124 for more information.

Crystal Oscillator Operation

The XTLDIS Flash option bit controls whether the crystal oscillator is enabled during reset. The crystal may later be disabled after reset if a new oscillator has been selected as the system clock. If the crystal is manually enabled after reset through the OSCCTL Reg-

DC Characteristics

Table 116 lists the DC characteristics of the Z8 Encore! F0830 Series products. All voltages are referenced to V_{SS}, the primary system ground.

Table 116. DC Characteristics

		T_A = 0°C to +70°C			T_A = –40°C to +105°C				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
V_{DD}	Supply Voltage				2.7	–	3.6	V	Power supply noise not to exceed 100mV peak to peak
V_{IL1}	Low Level Input Voltage				–0.3	–	0.3*V_{DD}	V	For all input pins except $\overline{\text{RESET}}$.
V_{IL2}	Low Level Input Voltage				–0.3	–	0.8	V	For $\overline{\text{RESET}}$.
V_{IH1}	High Level Input Voltage				2.0	–	5.5	V	For all input pins without analog or oscillator function.
V_{IH2}	High Level Input Voltage				2.0	–	V_{DD}+0.3	V	For those pins with analog or oscillator function.
V_{OL1}	Low Level Output Voltage				–	–	0.4	V	I_{OL} = 2mA; V_{DD} = 3.0V High Output Drive disabled.
V_{OH1}	High Level Output Voltage				2.4	–	–	V	I_{OH} = –2mA; V_{DD} = 3.0V High Output Drive disabled.
V_{OL2}	Low Level Output Voltage				–	–	0.6	V	I_{OL} = 20mA; V_{DD} = 3.3V High Output Drive enabled.
V_{OH2}	High Level Output Voltage				2.4	–	–	V	I_{OH} = –20mA; V_{DD} = 3.3V High Output Drive enabled.
I_{IL}	Input Leakage Current				–5	–	+5	µA	V_{DD} = 3.6V; V_{IN} = V_{DD} or V_{SS}[1]
I_{TL}	Tristate Leakage Current				–5	–	+5	µA	V_{DD} = 3.6V

Notes:
1. This condition excludes all pins that have on-chip pull-ups, when driven Low.
2. These values are provided for design guidance only and are not tested in production.
3. See Figure 31 for HALT Mode current.

Figure 32 displays the typical current consumption versus the system clock frequency in NORMAL Mode.

Figure 32. I_{CC} Versus System Clock Frequency (NORMAL Mode)

Low Power Control

For more information about the Power Control Register, see the Power Control Register Definitions section on page 31.

Hex Address: F80

Table 151. Power Control Register 0 (PWRCTL0)

Bit	7	6	5	4	3	2	1	0
Field	Reserved			VBO	Reserved	Reserved	COMP	Reserved
RESET	1	0	0	0	1	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F80H							

Hex Address: F81

This address range is reserved.

LED Controller

For more information about the LED Drive registers, see the GPIO Control Register Definitions section on page 39.

Hex Address: F82

Table 152. LED Drive Enable (LEDEN)

Bit	7	6	5	4	3	2	1	0
Field	LEDEN[7:0]							
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	F82H							

Hex Address: FD3

Table 172. Port A Output Data Register (PAOUT)

Bit	7	6	5	4	3	2	1	0
Field	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
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	FD3H							

Hex Address: FD4

Table 173. Port B GPIO Address Register (PBADDR)

Bit	7	6	5	4	3	2	1	0
Field	PADDR[7:0]							
RESET	00H							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	FD4H							

Hex Address: FD5

Table 174. Port B Control Registers (PBCTL)

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	FD5H							

Hex Address: FD6

Table 175. Port B Input Data Registers (PBIN)

Bit	7	6	5	4	3	2	1	0
Field	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X	X	X	X	X	X	X	X
R/W	R	R	R	R	R	R	R	R
Address	FD6H							

D

E

F

J

L