



Welcome to [E-XFL.COM](http://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	5MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	24
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	-
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	<a href="https://www.e-xfl.com/product-detail/zilog/z8f0113sj005sg">https://www.e-xfl.com/product-detail/zilog/z8f0113sj005sg</a>

Table 34.	LED Drive Level Low Register (LEDLVLL) . . . . .	53
Table 35.	Trap and Interrupt Vectors in Order of Priority . . . . .	55
Table 36.	Interrupt Request 0 Register (IRQ0) . . . . .	59
Table 37.	Interrupt Request 1 Register (IRQ1) . . . . .	60
Table 38.	Interrupt Request 2 Register (IRQ2) . . . . .	61
Table 39.	IRQ0 Enable and Priority Encoding . . . . .	61
Table 40.	IRQ0 Enable High Bit Register (IRQ0ENH) . . . . .	62
Table 41.	IRQ0 Enable Low Bit Register (IRQ0ENL) . . . . .	62
Table 42.	IRQ1 Enable and Priority Encoding . . . . .	63
Table 43.	IRQ1 Enable Low Bit Register (IRQ1ENL) . . . . .	64
Table 44.	IRQ1 Enable High Bit Register (IRQ1ENH) . . . . .	64
Table 45.	IRQ2 Enable and Priority Encoding . . . . .	65
Table 46.	IRQ2 Enable High Bit Register (IRQ2ENH) . . . . .	65
Table 47.	IRQ2 Enable Low Bit Register (IRQ2ENL) . . . . .	66
Table 48.	Interrupt Edge Select Register (IRQES) . . . . .	66
Table 49.	Shared Interrupt Select Register (IRQSS) . . . . .	67
Table 50.	Interrupt Control Register (IRQCTL) . . . . .	68
Table 51.	Timer 0–1 High Byte Register (TxH) . . . . .	84
Table 52.	Timer 0–1 Low Byte Register (TxL) . . . . .	84
Table 53.	Timer 0–1 Reload High Byte Register (TxRH) . . . . .	85
Table 54.	Timer 0–1 Reload Low Byte Register (TxRL) . . . . .	85
Table 55.	Timer 0–1 PWM High Byte Register (TxPWMH) . . . . .	86
Table 56.	Timer 0–1 PWM Low Byte Register (TxPWML) . . . . .	86
Table 57.	Timer 0–1 Control Register 0 (TxCTL0) . . . . .	87
Table 58.	Timer 0–1 Control Register 1 (TxCTL1) . . . . .	88
Table 59.	Watchdog Timer Approximate Time-Out Delays . . . . .	91
Table 60.	Watchdog Timer Control Register (WDTCTL) . . . . .	94
Table 61.	Watchdog Timer Reload Upper Byte Register (WDTU) . . . . .	95
Table 62.	Watchdog Timer Reload High Byte Register (WDTH) . . . . .	95
Table 63.	Watchdog Timer Reload Low Byte Register (WDTL) . . . . .	95
Table 64.	UART Transmit Data Register (U0TXD) . . . . .	109
Table 65.	UART Receive Data Register (U0RXD) . . . . .	109
Table 66.	UART Status 0 Register (U0STAT0) . . . . .	110
Table 67.	UART Status 1 Register (U0STAT1) . . . . .	111
Table 68.	UART Control 0 Register (U0CTL0) . . . . .	112
Table 69.	UART Control 1 Register (U0CTL1) . . . . .	113

► **Note:** \*Analog input alternate functions (ANA) are not available on Z8F0x13 devices.

## Signal Descriptions

Table 3 lists the Z8 Encore! XP F0823 Series signals. To determine the signals available for the specific package styles, see the [Pin Configurations](#) section on page 7.

**Table 3. Signal Descriptions**

Signal Mnemonic	I/O	Description
<b>General-Purpose I/O Ports A–D</b>		
PA[7:0]	I/O	Port A. These pins are used for general-purpose I/O.
PB[7:0] <sup>1</sup>	I/O	Port B. These pins are used for general-purpose I/O. PB6 and PB7 are available only in those devices without an ADC.
PC[7:0]	I/O	Port C. These pins are used for general-purpose I/O.
<b>UART Controllers</b>		
TXD0	O	Transmit Data. This signal is the transmit output from the UART and IrDA.
RXD0	I	Receive Data. This signal is the receive input for the UART and IrDA.
CTS0	I	Clear To Send. This signal is the flow control input for the UART.
DE	O	Driver Enable. This signal allows automatic control of external RS-485 drivers. This signal is approximately the inverse of the TXE (Transmit Empty) bit in the UART Status 0 Register. The DE signal can be used to ensure the external RS-485 driver is enabled when data is transmitted by the UART.
<b>Timers</b>		
T0OUT/T1OUT	O	Timer Output 0–1. These signals are output from the timers.
T0OUT/T1OUT	O	Timer Complement Output 0–1. These signals are output from the timers in PWM DUAL OUTPUT Mode.
T0IN/T1IN	I	Timer Input 0–1. These signals are used as the capture, gating and counter inputs. The T0IN signal is multiplexed T0OUT signals.
<b>Comparator</b>		
CINP/CINN	I	Comparator Inputs. These signals are the positive and negative inputs to the comparator.

**Notes:**

1. PB6 and PB7 are only available in 28-pin packages without ADC. In 28-pin packages with ADC, they are replaced by AV<sub>DD</sub> and AV<sub>SS</sub>.
2. The AV<sub>DD</sub> and AV<sub>SS</sub> signals are available only in 28-pin packages with ADC. They are replaced by PB6 and PB7 on 28-pin packages without ADC.

Table 5 provides detailed information about the characteristics for each pin available on Z8 Encore! XP F0823 Series 8-pin devices.

**Table 5. Pin Characteristics (8-Pin Devices)**

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tristate Output	Internal Pull-up or Pull- down	Schmitt- Trigger Input	Open Drain Output	5V Tolerance
PA0/DBG	I/O	I (but can change during reset if key sequence detected)	N/A	Yes	Program- mable Pull-up	Yes	Yes, Programmable	Yes, unless pull-ups enabled
PA1	I/O	I	N/A	Yes	Program- mable Pull-up	Yes	Yes, Programmable	Yes, unless pull-ups enabled
RESET/PA2	I/O	I/O (defaults to RESET)	N/A	Yes	Program- mable for PA2; always on for RESET	Yes	Programma- ble for PA2; always on for RESET	Yes, unless pull-ups enabled
PA[5:3]	I/O	I	N/A	Yes	Program- mable Pull-up	Yes	Yes, Programmable	Yes, unless pull-ups enabled
VDD	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
VSS	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A



**Table 23. Port A–C Alternate Function Subregisters (PxAF)**

Bit	7	6	5	4	3	2	1	0
Field	AF7	AF6	AF5	AF4	AF3	AF2	AF1	AF0
RESET	00H (Ports A–C); 04H (Port A of 8-pin device)							
R/W	R/W							
Address	If 02H in Port A–C Address Register, accessible through the Port A–C Control Register							

Bit	Description
[7:0] AFx	<b>Port Alternate Function enabled</b> 0 = The port pin is in NORMAL Mode and the DDx bit in the Port A–C Data Direction Subregister determines the direction of the pin. 1 = The alternate function selected through Alternate Function Set subregisters is enabled. Port pin operation is controlled by the alternate function.

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

## Port A–C Output Control Subregisters

The Port A–C Output Control Subregister (Table 24) is accessed through the Port A–C Control Register by writing 03H to the Port A–C Address Register. Setting the bits in the Port A–C Output Control subregisters to 1 configures the specified port pins for open-drain operation. These subregisters affect the pins directly and, as a result, alternate functions are also affected.

**Table 24. Port A–C Output Control Subregisters (PxOC)**

Bit	7	6	5	4	3	2	1	0
Field	POC7	POC6	POC5	POC4	POC3	POC2	POC1	POC0
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 03H in Port A–C Address Register, accessible through the Port A–C Control Register							

Bit	Description
[7:0] POCx	<b>Port Output Control</b> These bits function independently of the alternate function bit and always disable the drains if set to 1. 0 = The drains are enabled for any output mode (unless overridden by the alternate function). 1 = The drain of the associated pin is disabled (open-drain mode).

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

## Port A–C Pull-up Enable Subregisters

The Port A–C Pull-up Enable Subregister (Table 27) is accessed through the Port A–C Control Register by writing 06H to the Port A–C Address Register. Setting the bits in the Port A–C Pull-up Enable subregisters enables a weak internal resistive pull-up on the specified Port pins.

**Table 27. Port A–C Pull-Up Enable Subregisters (PPUE<sub>x</sub>)**

Bit	7	6	5	4	3	2	1	0
Field	PPUE7	PPUE6	PPUE5	PPUE4	PPUE3	PPUE2	PPUE1	PPUE0
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 06H in Port A–C Address Register, accessible through the Port A–C Control Register							

Bit	Description
[7:0]	<b>Port Pull-up Enabled</b>
PPUE <sub>x</sub>	0 = The weak pull-up on the Port pin is disabled. 1 = The weak pull-up on the Port pin is enabled.

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

## Shared Interrupt Select Register

The Shared Interrupt Select (IRQSS) register (Table 49) determines the source of the PADxS interrupts. The Shared Interrupt Select register selects between Port A and alternate sources for the individual interrupts.

Because these shared interrupts are edge-triggered, it is possible to generate an interrupt just by switching from one shared source to another. For this reason, an interrupt must be disabled before switching between sources.

**Table 49. Shared Interrupt Select Register (IRQSS)**

Bit	7	6	5	4	3	2	1	0
Field	Reserved	PA6CS	Reserved					
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	FCEH							

Bit	Description
[7]	<b>Reserved</b> This bit is reserved and must be programmed to 0.
[6] PA6CS	<b>PA6/Comparator Selection</b> 0 = PA6 is used for the interrupt for PA6CS interrupt request. 1 = The comparator is used as an interrupt for PA6CS interrupt requests.
[5:0]	<b>Reserved</b> These bits are reserved and must be programmed to 000000.

# Watchdog Timer

The Watchdog Timer (WDT) protects against corrupt or unreliable software, power faults, and other system-level problems which can place Z8 Encore! XP F0823 Series devices into unsuitable operating states. The features of Watchdog Timer include:

- On-chip RC oscillator
- A selectable time-out response: reset or interrupt
- 24-bit programmable time-out value

## Operation

The WDT is a retriggerable one-shot timer that resets or interrupts F0823 Series devices when the WDT reaches its terminal count. The Watchdog Timer uses a dedicated on-chip RC oscillator as its clock source. The Watchdog Timer operates in only two modes: ON and OFF. Once enabled, it always counts and must be refreshed to prevent a time-out. Perform an enable by executing the WDT instruction or by setting the WDT\_AO Flash Option Bit. The WDT\_AO bit forces the Watchdog Timer to operate immediately upon reset, even if a WDT instruction has not been executed.

The Watchdog Timer is a 24-bit reloadable down counter that uses three 8-bit registers in the eZ8 CPU register space to set the reload value. The nominal WDT time-out period is described by the following equation:

$$\text{WDT Time-out Period (ms)} = \frac{\text{WDT Reload Value}}{10}$$

where the WDT reload value is the decimal value of the 24-bit value given by {WDTU[7:0], WDTM[7:0], WDTL[7:0]} and the typical Watchdog Timer RC oscillator frequency is 10kHz. The Watchdog Timer cannot be refreshed after it reaches 000002H. The WDT Reload Value must not be set to values below 000004H. Table 59 provides information about approximate time-out delays for the minimum and maximum WDT reload values.

**Table 59. Watchdog Timer Approximate Time-Out Delays**

WDT Reload Value (Hex)	WDT Reload Value (Decimal)	Approximate Time-Out Delay (with 10kHz typical WDT oscillator frequency)	
		Typical	Description
000004	4	400 μs	Minimum time-out delay
FFFFFF	16,777,215	28 minutes	Maximum time-out delay

- Set or clear CTSE to enable or disable control from the remote receiver using the CTS pin.

8. Execute an EI instruction to enable interrupts.

The UART is now configured for interrupt-driven data transmission. Because the UART Transmit Data Register is empty, an interrupt is generated immediately. When the UART Transmit interrupt is detected, the associated interrupt service routine (ISR) performs the following:

1. Write the UART Control 1 Register to select the multiprocessor bit for the byte to be transmitted:  
 Set the Multiprocessor Bit Transmitter (MPBT) if sending an address byte, clear it if sending a data byte.
2. Write the data byte to the UART Transmit Data Register. The transmitter automatically transfers the data to the Transmit Shift register and transmits the data.
3. Clear the UART Transmit interrupt bit in the applicable Interrupt Request register.
4. Execute the IRET instruction to return from the interrupt-service routine and wait for the Transmit Data Register to again become empty.

## Receiving Data Using the Polled Method

Observe the following steps to configure the UART for polled data reception:

1. Write to the UART Baud Rate High and Low Byte registers to set an acceptable baud rate for the incoming data stream.
2. Enable the UART pin functions by configuring the associated GPIO port pins for alternate function operation.
3. Write to the UART Control 1 Register to enable MULTIPROCESSOR Mode functions, if appropriate.
4. Write to the UART Control 0 Register to:
  - Set the receive enable bit (REN) to enable the UART for data reception
  - Enable parity, if appropriate and if Multiprocessor mode is not enabled, and select either even or odd parity
5. Check the RDA bit in the UART Status 0 Register to determine if the Receive Data Register contains a valid data byte (indicated by a 1). If RDA is set to 1 to indicate available data, continue to [Step 6](#). If the Receive Data Register is empty (indicated by a 0), continue to monitor the RDA bit awaiting reception of the valid data.
6. Read data from the UART Receive Data Register. If operating in MULTIPROCESSOR (9-bit) Mode, further actions may be required depending on the MULTIPROCESSOR Mode bits MPMD[1:0].

scheme is enabled, the UART Address Compare register holds the network address of the device.

### **MULTIPROCESSOR (9-bit) Mode Receive Interrupts**

When MULTIPROCESSOR Mode is enabled, the UART only processes frames addressed to it. The determination of whether a frame of data is addressed to the UART can be made in hardware, software or some combination of the two, depending on the multiprocessor configuration bits. In general, the address compare feature reduces the load on the CPU, because it does not require access to the UART when it receives data directed to other devices on the multi-node network. The following three MULTIPROCESSOR modes are available in hardware:

- Interrupt on all address bytes
- Interrupt on matched address bytes and correctly framed data bytes
- Interrupt only on correctly framed data bytes

These modes are selected with `MPMD[1:0]` in the UART Control 1 Register. For all multiprocessor modes, bit `MPEN` of the UART Control 1 Register must be set to 1.

The first scheme is enabled by writing `01b` to `MPMD[1:0]`. In this mode, all incoming address bytes cause an interrupt, while data bytes never cause an interrupt. The interrupt service routine must manually check the address byte that caused triggered the interrupt. If it matches the UART address, the software clears `MPMD[0]`. Each new incoming byte interrupts the CPU. The software is responsible for determining the end of the frame. It checks for the end-of-frame by reading the `MPRX` bit of the UART Status 1 Register for each incoming byte. If `MPRX=1`, a new frame has begun. If the address of this new frame is different from the UART's address, `MPMD[0]` must be set to 1 causing the UART interrupts to go inactive until the next address byte. If the new frame's address matches the UART's, the data in the new frame is processed as well.

The second scheme requires the following: set `MPMD[1:0]` to `10B` and write the UART's address into the UART Address Compare register. This mode introduces additional hardware control, interrupting only on frames that match the UART's address. When an incoming address byte does not match the UART's address, it is ignored. All successive data bytes in this frame are also ignored. When a matching address byte occurs, an interrupt is issued and further interrupts now occur on each successive data byte. When the first data byte in the frame is read, the `NEWFRM` bit of the UART Status 1 Register is asserted. All successive data bytes have `NEWFRM=0`. When the next address byte occurs, the hardware compares it to the UART's address. If there is a match, the interrupts continues and the `NEWFRM` bit is set for the first byte of the new frame. If there is no match, the UART ignores all incoming bytes until the next address match.

The third scheme is enabled by setting `MPMD[1:0]` to `11b` and by writing the UART's address into the UART Address Compare Register. This mode is identical to the second

# ***Analog-to-Digital Converter***

The Analog-to-Digital Converter (ADC) converts an analog input signal to its digital representation. The features of this sigma-delta ADC include:

- 10-bit resolution
- Eight single-ended analog input sources are multiplexed with general-purpose I/O ports
- Interrupt upon conversion complete
- Bandgap generated internal voltage reference generator with two selectable levels
- Factory offset and gain calibration

## **Architecture**

Figure 19 displays the major functional blocks of the ADC. An analog multiplexer network selects the ADC input from the available analog pins, ANA0 through ANA7.

**Figure 20. Flash Memory Arrangement**

## Flash Information Area

The Flash information area is separate from program memory and is mapped to the address range `FE00H` to `FFFFH`. Not all these addresses are accessible. Factory trim values for the analog peripherals are stored here. Factory calibration data for the ADC is also stored here.

## Operation

The Flash Controller programs and erases Flash memory. The Flash Controller provides the proper Flash controls and timing for Byte Programming, Page Erase, and Mass Erase of Flash memory.

The Flash Controller contains several protection mechanisms to prevent accidental programming or erasure. These mechanism operate on the page, sector and full-memory levels.

Figure 21 displays a basic Flash Controller flow. The following subsections provide details about the various operations (Lock, Unlock, Byte Programming, Page Protect, Page Unprotect, Page Select Page Erase, and Mass Erase) displayed in Figure 21.



- 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, the Flash Controller allows only a single page to be opened for write/erase. When writing multiple Flash pages, the Flash controller must repeat the unlock sequence to select another page.

---

## Flash Control Register Definitions

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

Flash Control Register: see page 141

Flash Status Register: see page 143

Flash Page Select Register: see page 143

Flash Sector Protect Register: see page 145

Flash Frequency High and Low Byte Registers: see page 145

## Flash Control Register

The Flash Controller must be unlocked using the Flash Control (FTCTL) Register before programming or erasing the Flash memory. Writing the sequence 73H 8CH, sequentially, to the Flash Control Register unlocks the Flash Controller. When the Flash Controller is unlocked, the Flash memory can be enabled for Mass Erase or Page Erase by writing the appropriate enable command to the FCTL. Page Erase applies only to the active page selected in Flash Page Select register. Mass Erase is enabled only through the On-Chip Debugger. Writing an invalid value or an invalid sequence returns the Flash Controller to its locked state. The Write-only Flash Control Register shares its Register File address with the read-only Flash Status Register.

**Stuff Instruction (11H).** The Stuff command steps one assembly instruction and allows specification of the first byte of the instruction. The remaining 0–4 bytes of the instruction are read from Program Memory. This command is useful for stepping over instructions where the first byte of the instruction has been overwritten by a Breakpoint. If the device is not in DEBUG Mode or the Flash Read Protect Option bit is enabled, the OCD ignores this command.

DBG ← 11H  
DBG ← opcode[7:0]

**Execute Instruction (12H).** The Execute command allows sending an entire instruction to be executed to the eZ8 CPU. This command can also step over breakpoints. The number of bytes to send for the instruction depends on the opcode. 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

This section describes the features of the On-Chip Debugger Control and Status registers.

### OCD Control Register

The OCD Control Register controls the state of the OCD. This register is used to enter or exit DEBUG Mode and to enable the BRK instruction. It also resets Z8 Encore! XP F0823 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 102. OCD Control Register (OCDCTL)

Bit	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

Bit	Description
[7] DBGMODE	<p><b>DEBUG Mode</b></p> <p>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.</p> <p>0 = F0823 Series device is operating in NORMAL Mode. 1 = F0823 Series device is in DEBUG Mode.</p>
[6] BRKEN	<p><b>Breakpoint Enable</b></p> <p>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.</p> <p>0 = Breakpoints are disabled. 1 = Breakpoints are enabled.</p>
[5] DBGACK	<p><b>Debug Acknowledge</b></p> <p>This bit enables the debug acknowledge feature. If this bit is set to 1, the OCD sends a Debug Acknowledge character (FFH) to the host when a Breakpoint occurs.</p> <p>0 = Debug Acknowledge is disabled. 1 = Debug Acknowledge is enabled.</p>
[4:1]	<p><b>Reserved</b></p> <p>These bits are reserved and must be 00000 when read.</p>
[0] RST	<p><b>Reset</b></p> <p>Setting this bit to 1 resets the Z8F04xA family device. The device goes through a normal Power-On Reset sequence with the exception that the OCD is not reset. This bit is automatically cleared to 0 at the end of reset.</p> <p>0 = No effect. 1 = Reset the Flash Read Protect Option Bit device.</p>

## OCD Status Register

The OCD Status Register reports status information about the current state of the debugger and the system.

**Table 103. OCD Status Register (OCDSTAT)**

Bit	7	6	5	4	3	2	1	0
Field	DBG	HALT	FRPENB	Reserved				
RESET	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	Description
[7] DBG	<b>Debug Status</b> 0 = NORMAL Mode. 1 = DEBUG Mode.
[6] HALT	<b>HALT Mode</b> 0 = Not in HALT Mode. 1 = In HALT Mode.
[5] FRPENB	<b>Flash Read Protect Option Bit Enable</b> 0 = FRP bit enabled to allow disabling of many OCD commands. 1 = FRP bit has no effect.
[4:0]	<b>Reserved</b> These bits are reserved and must be 00000 when read.

## Internal Precision Oscillator

The internal precision oscillator (IPO) is designed for use without external components. You can either manually trim the oscillator for a non-standard frequency or use the automatic factory-trimmed version to achieve a 5.53MHz frequency. The features of IPO include:

- On-chip RC oscillator that does not require external components
- Output frequency of either 5.53MHz or 32.8kHz (contains both a fast and a slow mode)
- Trimming possible through Flash option bits with user override
- Elimination of crystals or ceramic resonators in applications where high timing accuracy is not required

### Operation

An 8-bit trimming register, incorporated into the design, compensates for absolute variation of oscillator frequency. Once trimmed the oscillator frequency is stable and does not require subsequent calibration. Trimming is performed during manufacturing and is not necessary for you to repeat unless a frequency other than 5.53MHz (fast mode) or 32.8kHz (slow mode) is required. This trimming is done at +30°C and a supply voltage of 3.3 V, so accuracy of this operating point is optimal.

Power down this block for minimum system power. By default, the oscillator is configured through the Flash Option bits. However, the user code can override these trim values, as described in the [Trim Bit Address Space](#) section on page 151.

Select one of the two frequencies for the oscillator: 5.53MHz and 32.8kHz, using the OSCSEL bits in the [Oscillator Control](#) chapter on page 169.

**Table 119. Opcode Map Abbreviations**

<b>Abbreviation</b>	<b>Description</b>	<b>Abbreviation</b>	<b>Description</b>
b	Bit position	IRR	Indirect Register Pair
cc	Condition code	p	Polarity (0 or 1)
X	8-bit signed index or displacement	r	4-bit Working Register
DA	Destination address	R	8-bit register
ER	Extended Addressing register	r1, R1, Ir1, Irr1, IR1, rr1, RR1, IRR1, ER1	Destination address
IM	Immediate data value	r2, R2, Ir2, Irr2, IR2, rr2, RR2, IRR2, ER2	Source address
Ir	Indirect Working Register	RA	Relative
IR	Indirect register	rr	Working Register Pair
Irr	Indirect Working Register Pair	RR	Register Pair

**Table 126. Flash Memory Electrical Characteristics and Timing**

$V_{DD} = 2.7V \text{ to } 3.6V$ $T_A = -40^{\circ}C \text{ to } +105^{\circ}C$ (unless otherwise stated)					
Parameter	Minimum	Typical	Maximum	Units	Notes
Flash Byte Read Time	100	–	–	ns	
Flash Byte Program Time	20	–	40	$\mu s$	
Flash Page Erase Time	10	–	–	ms	
Flash Mass Erase Time	200	–	–	ms	
Writes to Single Address Before Next Erase	–	–	2		
Flash Row Program Time	–	–	8	ms	Cumulative program time for single row cannot exceed limit before next erase. This parameter is only an issue when bypassing the Flash Controller.
Data Retention	100	–	–	years	25°C
Endurance	10,000	–	–	cycles	Program/erase cycles

**Table 127. Watchdog Timer Electrical Characteristics and Timing**

V <sub>DD</sub> = 2.7V to 3.6V T <sub>A</sub> = −40°C to +105°C (unless otherwise stated)						
Symbol	Parameter	Minimum	Typical	Maximum	Units	Conditions
F <sub>WDT</sub>	WDT Oscillator Frequency		10		kHz	
F <sub>WDT</sub>	WDT Oscillator Error			±50	%	
T <sub>WDT-CAL</sub>	WDT Calibrated Timeout	0.98	1	1.02	s	V <sub>DD</sub> = 3.3 V; T <sub>A</sub> = 30°C
		0.70	1	1.30	s	V <sub>DD</sub> = 2.7V to 3.6V T <sub>A</sub> = 0°C to 70°C
		0.50	1	1.50	s	V <sub>DD</sub> = 2.7V to 3.6V T <sub>A</sub> = −40°C to +105°C

## On-Chip Debugger Timing

Figure 31 and Table 132 provide timing information for the DBG pin. The DBG pin timing specifications assume a 4 ns maximum rise and fall time.

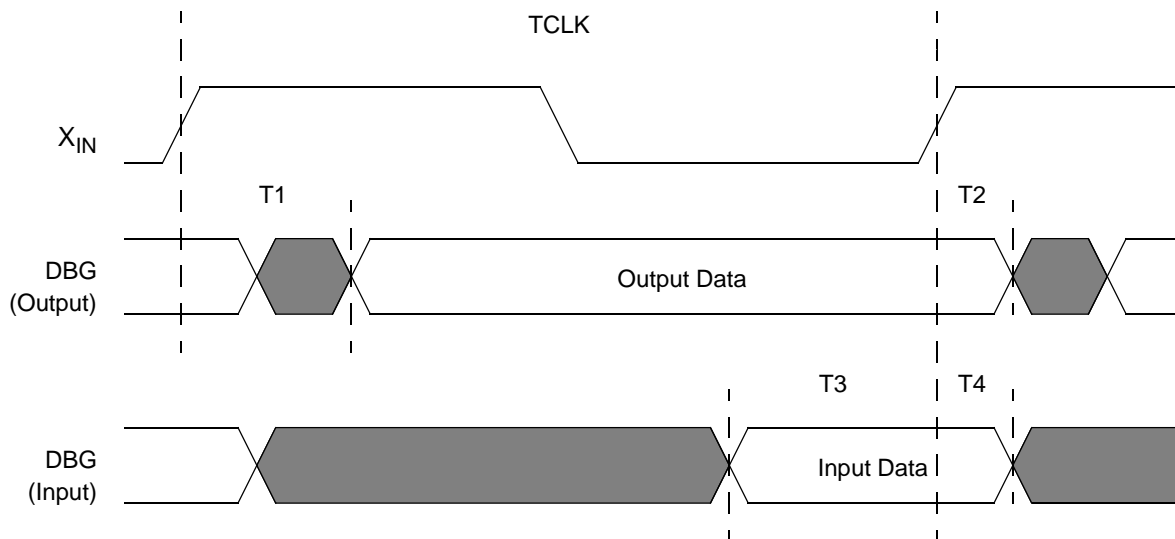


Figure 31. On-Chip Debugger Timing

Table 132. On-Chip Debugger Timing

Parameter	Abbreviation	Delay (ns)	
		Minimum	Maximum
DBG			
T <sub>1</sub>	X <sub>IN</sub> Rise to DBG Valid Delay	—	15
T <sub>2</sub>	X <sub>IN</sub> Rise to DBG Output Hold Time	2	—
T <sub>3</sub>	DBG to X <sub>IN</sub> Rise Input Setup Time	5	—
T <sub>4</sub>	DBG to X <sub>IN</sub> Rise Input Hold Time	5	—



**Table 135. Z8 Encore! XP F0823 Series Ordering Matrix (Continued)**

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Description
<b>Z8 Encore! XP F0823 Series with 1 KB Flash</b>								
<b>Standard Temperature: 0°C to 70°C</b>								
Z8F0113PB005SG	1 KB	256 B	6	12	2	0	1	PDIP 8-pin package
Z8F0113QB005SG	1 KB	256 B	6	12	2	0	1	QFN 8-pin package
Z8F0113SB005SG	1 KB	256 B	6	12	2	0	1	SOIC 8-pin package
Z8F0113SH005SG	1 KB	256 B	16	18	2	0	1	SOIC 20-pin package
Z8F0113HH005SG	1 KB	256 B	16	18	2	0	1	SSOP 20-pin package
Z8F0113PH005SG	1 KB	256 B	16	18	2	0	1	PDIP 20-pin package
Z8F0113SJ005SG	1 KB	256 B	24	18	2	0	1	SOIC 28-pin package
Z8F0113HJ005SG	1 KB	256 B	24	18	2	0	1	SSOP 28-pin package
Z8F0113PJ005SG	1 KB	256 B	24	18	2	0	1	PDIP 28-pin package
<b>Extended Temperature: -40°C to 105°C</b>								
Z8F0113PB005EG	1 KB	256 B	6	12	2	0	1	PDIP 8-pin package
Z8F0113QB005EG	1 KB	256 B	6	12	2	0	1	QFN 8-pin package
Z8F0113SB005EG	1 KB	256 B	6	12	2	0	1	SOIC 8-pin package
Z8F0113SH005EG	1 KB	256 B	16	18	2	0	1	SOIC 20-pin package
Z8F0113HH005EG	1 KB	256 B	16	18	2	0	1	SSOP 20-pin package
Z8F0113PH005EG	1 KB	256 B	16	18	2	0	1	PDIP 20-pin package
Z8F0113SJ005EG	1 KB	256 B	24	18	2	0	1	SOIC 28-pin package
Z8F0113HJ005EG	1 KB	256 B	24	18	2	0	1	SSOP 28-pin package
Z8F0113PJ005EG	1 KB	256 B	24	18	2	0	1	PDIP 28-pin package