



Welcome to [E-XFL.COM](#)

### What is "[Embedded - Microcontrollers](#)"?

"[Embedded - Microcontrollers](#)" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

### Applications of "[Embedded - Microcontrollers](#)"

#### Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, Temp Sensor, WDT
Number of I/O	23
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	64 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°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/z8f022asj020ec00tr">https://www.e-xfl.com/product-detail/zilog/z8f022asj020ec00tr</a>

# Overview

Zilog's Z8 Encore!<sup>®</sup> MCU family of products are the first in a line of Zilog<sup>®</sup> microcontroller products based upon the 8-bit eZ8 CPU. Zilog's Z8 Encore! XP<sup>®</sup> F082A Series products expand upon Zilog's extensive line of 8-bit microcontrollers. The Flash in-circuit programming capability allows for faster development time and program changes in the field. The new eZ8 CPU is upward compatible with existing Z8<sup>®</sup> instructions. The rich peripheral set of the Z8 Encore! XP F082A Series makes it suitable for a variety of applications including motor control, security systems, home appliances, personal electronic devices, and sensors.

## Features

The key features of Z8 Encore! XP F082A Series products include:

- 20 MHz eZ8 CPU
- 1 KB, 2 KB, 4 KB, or 8 KB Flash memory with in-circuit programming capability
- 256 B, 512 B, or 1 KB register RAM
- Up to 128 B non-volatile data storage (NVDS)
- Internal precision oscillator trimmed to  $\pm 1\%$  accuracy
- External crystal oscillator, operating up to 20 MHz
- Optional 8-channel, 10-bit analog-to-digital converter (ADC)
- Optional on-chip temperature sensor
- On-chip analog comparator
- Optional on-chip low-power operational amplifier (LPO)
- Full-duplex UART
- The UART baud rate generator (BRG) can be configured and used as a basic 16-bit timer
- Infrared Data Association (IrDA)-compliant infrared encoder/decoders, integrated with UART
- Two enhanced 16-bit timers with capture, compare, and PWM capability
- Watchdog Timer (WDT) with dedicated internal RC oscillator
- Up to 20 vectored interrupts
- 6 to 25 I/O pins depending upon package

## **Interrupt Controller**

The Z8 Encore! XP<sup>®</sup> F082A Series products support up to 20 interrupts. These interrupts consist of 8 internal peripheral interrupts and 12 general-purpose I/O pin interrupt sources. The interrupts have three levels of programmable interrupt priority.

## **Reset Controller**

The Z8 Encore! XP F082A Series products can be reset using the  $\overline{\text{RESET}}$  pin, Power-On Reset, Watchdog Timer (WDT) time-out, STOP mode exit, or Voltage Brownout (VBO) warning signal. The  $\overline{\text{RESET}}$  pin is bi-directional, that is, it functions as reset source as well as a reset indicator.

mation Area data rather than the Program Memory data. Access to the Flash Information Area is read-only.

**Table 6. Z8 Encore! XP F082A Series Flash Memory Information Area Map**

Program Memory Address (Hex)	Function
FE00–FE3F	Zilog Option Bits/Calibration Data
FE40–FE53	Part Number 20-character ASCII alphanumeric code Left justified and filled with FFH
FE54–FE5F	Reserved
FE60–FE7F	Zilog Calibration Data
FE80–FFFF	Reserved

**Table 15. Port Alternate Function Mapping (8-Pin Parts)**

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Select Register AFS1	Alternate Function Select Register AFS2
<b>Port A</b>	PA0	T0IN	Timer 0 Input	AFS1[0]: 0	AFS2[0]: 0
		Reserved		AFS1[0]: 0	AFS2[0]: 1
		Reserved		AFS1[0]: 1	AFS2[0]: 0
		$\overline{T0OUT}$	Timer 0 Output Complement	AFS1[0]: 1	AFS2[0]: 1
	PA1	T0OUT	Timer 0 Output	AFS1[1]: 0	AFS2[1]: 0
		Reserved		AFS1[1]: 0	AFS2[1]: 1
		CLKIN	External Clock Input	AFS1[1]: 1	AFS2[1]: 0
		Analog Functions*	ADC Analog Input/VREF	AFS1[1]: 1	AFS2[1]: 1
	PA2	DE0	UART 0 Driver Enable	AFS1[2]: 0	AFS2[2]: 0
		$\overline{RESET}$	External Reset	AFS1[2]: 0	AFS2[2]: 1
		T1OUT	Timer 1 Output	AFS1[2]: 1	AFS2[2]: 0
		Reserved		AFS1[2]: 1	AFS2[2]: 1
	PA3	$\overline{CTS0}$	UART 0 Clear to Send	AFS1[3]: 0	AFS2[3]: 0
		COUT	Comparator Output	AFS1[3]: 0	AFS2[3]: 1
		T1IN	Timer 1 Input	AFS1[3]: 1	AFS2[3]: 0
		Analog Functions*	ADC Analog Input/LPO Input (P)	AFS1[3]: 1	AFS2[3]: 1
	PA4	RXD0	UART 0 Receive Data	AFS1[4]: 0	AFS2[4]: 0
		Reserved		AFS1[4]: 0	AFS2[4]: 1
		Reserved		AFS1[4]: 1	AFS2[4]: 0
		Analog Functions*	ADC/Comparator Input (N)/LPO Input (N)	AFS1[4]: 1	AFS2[4]: 1
	PA5	TXD0	UART 0 Transmit Data	AFS1[5]: 0	AFS2[5]: 0
		$\overline{T1OUT}$	Timer 1 Output Complement	AFS1[5]: 0	AFS2[5]: 1
		Reserved		AFS1[5]: 1	AFS2[5]: 0
		Analog Functions*	ADC/Comparator Input (P) LPO Output	AFS1[5]: 1	AFS2[5]: 1

\*Analog Functions include ADC inputs, ADC reference, comparator inputs and LPO ports.

**Note:** Also, alternate function selection as described in [Port A–D Alternate Function Sub-Registers](#) on page 47 must be enabled.

## Port A–D Address Registers

The Port A–D Address registers select the GPIO Port functionality accessible through the Port A–D Control registers. The Port A–D Address and Control registers combine to provide access to all GPIO Port controls ([Table 17](#)).

**Table 17. Port A–D GPIO Address Registers (PxADDR)**

BITS	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
ADDR	FD0H, FD4H, FD8H, FDCH							

PADDR[7:0]—Port Address

The Port Address selects one of the sub-registers accessible through the Port Control register.

PADDR[7:0] Port Control sub-register accessible using the Port A–D Control Registers	
00H	No function. Provides some protection against accidental Port reconfiguration.
01H	Data Direction.
02H	Alternate Function.
03H	Output Control (Open-Drain).
04H	High Drive Enable.
05H	Stop Mode Recovery Source Enable.
06H	Pull-up Enable.
07H	Alternate Function Set 1.
08H	Alternate Function Set 2.
09H–FFH	No function.

## Port A–D Control Registers

The Port A–D Control registers set the GPIO port operation. The value in the corresponding Port A–D Address register determines which sub-register is read from or written to by a Port A–D Control register transaction ([Table 18](#)).

U0RXI—UART 0 Receiver Interrupt Request

0 = No interrupt request is pending for the UART 0 receiver.

1 = An interrupt request from the UART 0 receiver is awaiting service.

U0TXI—UART 0 Transmitter Interrupt Request

0 = No interrupt request is pending for the UART 0 transmitter.

1 = An interrupt request from the UART 0 transmitter is awaiting service.

ADCI—ADC Interrupt Request

0 = No interrupt request is pending for the analog-to-digital Converter.

1 = An interrupt request from the Analog-to-Digital Converter is awaiting service.

## Interrupt Request 1 Register

The Interrupt Request 1 (IRQ1) register ([Table 34](#)) stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ1 register becomes 1. If interrupts are globally enabled (vectored interrupts), the interrupt controller passes an interrupt request to the eZ8 CPU. If interrupts are globally disabled (polled interrupts), the eZ8 CPU can read the Interrupt Request 1 register to determine if any interrupt requests are pending.

**Table 34. Interrupt Request 1 Register (IRQ1)**

BITS	7	6	5	4	3	2	1	0
FIELD	PA7VI	PA6CI	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FC3H							

PA7VI—Port A Pin 7 or LVD Interrupt Request

0 = No interrupt request is pending for GPIO Port A or LVD.

1 = An interrupt request from GPIO Port A or LVD.

PA6CI—Port A Pin 6 or Comparator Interrupt Request

0 = No interrupt request is pending for GPIO Port A or Comparator.

1 = An interrupt request from GPIO Port A or Comparator.

PAxI—Port A Pin x Interrupt Request

0 = No interrupt request is pending for GPIO Port A pin x.

1 = An interrupt request from GPIO Port A pin x is awaiting service.

where x indicates the specific GPIO Port pin number (0–5).

Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state for one system clock cycle (from Low to High or from High to Low) upon timer Reload. If it is appropriate to have the Timer Output 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.

Follow the steps below for configuring a timer for ONE-SHOT mode and 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 given by 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 counting resumes. Also, 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.

Follow the steps below for configuring a timer for CONTINUOUS mode and initiating the count:

1. Write to the Timer Control register to:
  - Disable the timer
  - Configure the timer for CONTINUOUS mode.



6. Check the TDRE bit in the UART Status 0 register to determine if the Transmit Data register is empty (indicated by a 1). If empty, continue to [Step 7](#). If the Transmit Data register is full (indicated by a 0), continue to monitor the TDRE bit until the Transmit Data register becomes available to receive new data.
7. Write the UART Control 1 register to select the outgoing address bit.
8. Set the Multiprocessor Bit Transmitter (MPBT) if sending an address byte, clear it if sending a data byte.
9. 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.
10. Make any changes to the Multiprocessor Bit Transmitter (MPBT) value, if appropriate and MULTIPROCESSOR mode is enabled.
11. To transmit additional bytes, return to [Step 5](#).

### **Transmitting Data using the Interrupt-Driven Method**

The UART Transmitter interrupt indicates the availability of the Transmit Data register to accept new data for transmission. Follow the steps below to configure the UART for interrupt-driven data transmission:

1. Write to the UART Baud Rate High and Low Byte registers to set the appropriate baud rate.
2. Enable the UART pin functions by configuring the associated GPIO Port pins for alternate function operation.
3. Execute a DI instruction to disable interrupts.
4. Write to the Interrupt control registers to enable the UART Transmitter interrupt and set the acceptable priority.
5. Write to the UART Control 1 register to enable MULTIPROCESSOR (9-bit) mode functions, if MULTIPROCESSOR mode is appropriate.
6. Set the MULTIPROCESSOR Mode Select (MPEN) to Enable MULTIPROCESSOR mode.
7. Write to the UART Control 0 register to:
  - Set the transmit enable bit (TEN) to enable the UART for data transmission.
  - Enable parity, if appropriate and if MULTIPROCESSOR mode is not enabled, and select either even or odd parity.
  - Set or clear CTSE to enable or disable control from the remote receiver using the  $\overline{\text{CTS}}$  pin.
8. Execute an EI instruction to enable interrupts.

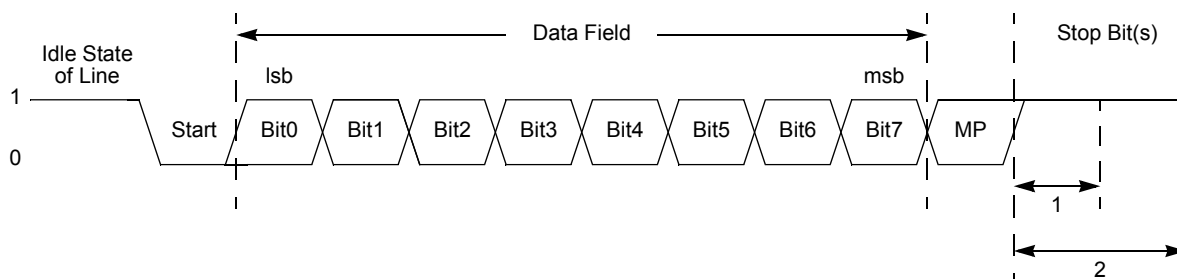
3. Clears the UART Receiver interrupt in the applicable Interrupt Request register.
4. Executes the IRET instruction to return from the interrupt-service routine and await more data.

### Clear To Send (CTS) Operation

The CTS pin, if enabled by the CTSE bit of the UART Control 0 register, performs flow control on the outgoing transmit datastream. The Clear To Send (CTS) input pin is sampled one system clock before beginning any new character transmission. To delay transmission of the next data character, an external receiver must deassert CTS at least one system clock cycle before a new data transmission begins. For multiple character transmissions, this action is typically performed during Stop Bit transmission. If CTS deasserts in the middle of a character transmission, the current character is sent completely.

### MULTIPROCESSOR (9-bit) Mode

The UART has a MULTIPROCESSOR (9-bit) mode that uses an extra (9th) bit for selective communication when a number of processors share a common UART bus. In MULTIPROCESSOR mode (also referred to as 9-bit mode), the multiprocessor bit (MP) is transmitted immediately following the 8-bits of data and immediately preceding the Stop bit(s) as displayed in Figure 13. The character format is:



**Figure 13. UART Asynchronous MULTIPROCESSOR Mode Data Format**

In MULTIPROCESSOR (9-bit) mode, the Parity bit location (9th bit) becomes the Multiprocessor control bit. The UART Control 1 and Status 1 registers provide MULTIPROCESSOR (9-bit) mode control and status information. If an automatic address matching 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

and the CTSE bit. If the  $\overline{\text{CTS}}$  signal is Low and the CTSE bit is 1, the transmitter is enabled.

0 = Transmitter disabled.

1 = Transmitter enabled.

REN—Receive Enable

This bit enables or disables the receiver.

0 = Receiver disabled.

1 = Receiver enabled.

CTSE—CTS Enable

0 = The  $\overline{\text{CTS}}$  signal has no effect on the transmitter.

1 = The UART recognizes the  $\overline{\text{CTS}}$  signal as an enable control from the transmitter.

PEN—Parity Enable

This bit enables or disables parity. Even or odd is determined by the PSEL bit.

0 = Parity is disabled.

1 = The transmitter sends data with an additional parity bit and the receiver receives an additional parity bit.

PSEL—Parity Select

0 = Even parity is transmitted and expected on all received data.

1 = Odd parity is transmitted and expected on all received data.

SBRK—Send Break

This bit pauses or breaks data transmission. Sending a break interrupts any transmission in progress, so ensure that the transmitter has finished sending data before setting this bit.

0 = No break is sent.

1 = Forces a break condition by setting the output of the transmitter to zero.

STOP—Stop Bit Select

0 = The transmitter sends one stop bit.

1 = The transmitter sends two stop bits.

LBEN—Loop Back Enable

0 = Normal operation.

1 = All transmitted data is looped back to the receiver.

**Table 62. UART Control 1 Register (U0CTL1)**

BITS	7	6	5	4	3	2	1	0
FIELD	MPMD[1]	MPEN	MPMD[0]	MPBT	DEPOL	BRGCTL	RDAIRQ	IREN
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F43H							

## UART Address Compare Register

The UART Address Compare (UxADDR) register stores the multi-node network address of the UART (see [Table 67](#)). When the MPMD[1] bit of UART Control Register 0 is set, all incoming address bytes are compared to the value stored in the Address Compare register. Receive interrupts and RDA assertions only occur in the event of a match.

**Table 67. UART Address Compare Register (U0ADDR)**

BITS	7	6	5	4	3	2	1	0
FIELD	COMP_ADDR							
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F45H							

COMP\_ADDR—Compare Address  
This 8-bit value is compared to incoming address bytes.

## UART Baud Rate High and Low Byte Registers

The UART Baud Rate High (UxBRH) and Low Byte (UxBRL) registers ([Table 68](#) and [Table 69](#)) combine to create a 16-bit baud rate divisor value (BRG[15:0]) that sets the data transmission rate (baud rate) of the UART.

**Table 68. UART Baud Rate High Byte Register (U0BRH)**

BITS	7	6	5	4	3	2	1	0
FIELD	BRH							
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
ADDR	F46H							

**Table 69. UART Baud Rate Low Byte Register (U0BRL)**

BITS	7	6	5	4	3	2	1	0
FIELD	BRL							
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
ADDR	F47H							

The UART data rate is calculated using the following equation:

$$\text{UART Baud Rate (bits/s)} = \frac{\text{System Clock Frequency (Hz)}}{16 \times \text{UART Baud Rate Divisor Value}}$$

For a given UART data rate, calculate the integer baud rate divisor value using the following equation:

$$\text{UART Baud Rate Divisor Value (BRG)} = \text{Round}\left(\frac{\text{System Clock Frequency (Hz)}}{16 \times \text{UART Data Rate (bits/s)}}\right)$$

The baud rate error relative to the acceptable baud rate is calculated using the following equation:

$$\text{UART Baud Rate Error (\%)} = 100 \times \left( \frac{\text{Actual Data Rate} - \text{Desired Data Rate}}{\text{Desired Data Rate}} \right)$$

For reliable communication, the UART baud rate error must never exceed 5 percent.

[Table 70](#) provides information on the data rate errors for popular baud rates and commonly used crystal oscillator frequencies.

**Table 70. UART Baud Rates**

10.0 MHz System Clock				5.5296 MHz System Clock			
Acceptable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Acceptable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	N/A	N/A	N/A	1250.0	N/A	N/A	N/A
625.0	1	625.0	0.00	625.0	N/A	N/A	N/A
250.0	3	208.33	-16.67	250.0	1	345.6	38.24
115.2	5	125.0	8.51	115.2	3	115.2	0.00
57.6	11	56.8	-1.36	57.6	6	57.6	0.00
38.4	16	39.1	1.73	38.4	9	38.4	0.00
19.2	33	18.9	0.16	19.2	18	19.2	0.00
9.60	65	9.62	0.16	9.60	36	9.60	0.00
4.80	130	4.81	0.16	4.80	72	4.80	0.00
2.40	260	2.40	-0.03	2.40	144	2.40	0.00
1.20	521	1.20	-0.03	1.20	288	1.20	0.00
0.60	1042	0.60	-0.03	0.60	576	0.60	0.00
0.30	2083	0.30	0.2	0.30	1152	0.30	0.00

```

nop      ; wait for output to settle
clr irq0 ; clear any spurious interrupts pending
ei

```

## Comparator Control Register Definitions

### Comparator Control Register

The Comparator Control Register (CMP0) configures the comparator inputs and sets the value of the internal voltage reference.

**Table 75. Comparator Control Register (CMP0)**

BITS	7	6	5	4	3	2	1	0
FIELD	INPSEL	INNSEL	REFLVL				Reserved (20-/28-pin) REFLVL (8-pin)	
RESET	0	0	0	1	0	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F90H							

INPSEL—Signal Select for Positive Input

0 = GPIO pin used as positive comparator input

1 = temperature sensor used as positive comparator input

INNSEL—Signal Select for Negative Input

0 = internal reference disabled, GPIO pin used as negative comparator input

1 = internal reference enabled as negative comparator input

REFLVL—Internal Reference Voltage Level (this reference is independent of the ADC voltage reference). Note that the 8-pin devices contain two additional LSBs for increased resolution.

For 20-/28-pin devices:

0000 = 0.0 V

0001 = 0.2 V

0010 = 0.4 V

0011 = 0.6 V

0100 = 0.8 V

0101 = 1.0 V (Default)

0110 = 1.2 V

0111 = 1.4 V

1000 = 1.6 V

value 63H to the Flash Control register initiates the Mass Erase operation. While the Flash Controller executes the Mass Erase operation, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate. Using the On-Chip Debugger, poll the Flash Status register to determine when the Mass Erase operation is complete. When the Mass Erase is complete, the Flash Controller returns to its locked state.

## Flash Controller Bypass

The Flash Controller can be bypassed and the control signals for the Flash memory brought out to the GPIO pins. Bypassing the Flash Controller allows faster Row Programming algorithms by controlling the Flash programming signals directly.

Row programming is recommended for gang programming applications and large volume customers who do not require in-circuit initial programming of the Flash memory. Page Erase operations are also supported when the Flash Controller is bypassed.

For more information on bypassing the Flash Controller, refer to *Third-Party Flash Programming Support for Z8 Encore!<sup>®</sup> MCU Application Note (AN0117)* available for download at [www.zilog.com](http://www.zilog.com).

## Flash Controller Behavior in DEBUG Mode

The following changes in behavior of the Flash Controller occur when the Flash Controller is accessed using the On-Chip Debugger:

- The Flash Write Protect option bit is ignored.
- 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, the 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.*

## Flash Control Register Definitions

### Flash Control Register

The Flash Controller must be unlocked using the Flash Control (FCTL) 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.

**Table 78. Flash Control Register (FCTL)**

BITS	7	6	5	4	3	2	1	0
FIELD	FCMD							
RESET	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W
ADDR	FF8H							

FCMD—Flash Command

73H = First unlock command.

8CH = Second unlock command.

95H = Page Erase command (must be third command in sequence to initiate Page Erase).

63H = Mass Erase command (must be third command in sequence to initiate Mass Erase).

5EH = Enable Flash Sector Protect Register Access







**Table 126. Absolute Maximum Ratings (Continued)**

Parameter	Minimum	Maximum	Units	Notes
<b>28-pin Packages Maximum Ratings at 0 °C to 70 °C</b>				
Total power dissipation		450	mW	
Maximum current into $V_{DD}$ or out of $V_{SS}$		125	mA	
Operating temperature is specified in DC Characteristics.				
1. This voltage applies to all pins except the following: $V_{DD}$ , $AV_{DD}$ , pins supporting analog input (Port B[5:0], Port C[2:0]) and pins supporting the crystal oscillator (PA0 and PA1). On the 8-pin packages, this applies to all pins but $V_{DD}$ .				
2. This voltage applies to pins on the 20-/28-pin packages supporting analog input (Port B[5:0], Port C[2:0]) and pins supporting the crystal oscillator (PA0 and PA1).				

## DC Characteristics

Table 127 lists the DC characteristics of the Z8 Encore! XP<sup>®</sup> F082A Series products. All voltages are referenced to  $V_{SS}$ , the primary system ground.

**Table 127. DC Characteristics**

Symbol	Parameter	$T_A = -40\text{ °C to }+105\text{ °C}$ (unless otherwise specified)			Units	Conditions
		Minimum	Typical	Maximum		
$V_{DD}$	Supply Voltage	2.7	–	3.6	V	
$V_{IL1}$	Low Level Input Voltage	-0.3	–	$0.3 \cdot V_{DD}$	V	
$V_{IH1}$	High Level Input Voltage	$0.7 \cdot V_{DD}$	–	5.5	V	For all input pins without analog or oscillator function. For all signal pins on the 8-pin devices. Programmable pull-ups must also be disabled.
$V_{IH2}$	High Level Input Voltage	$0.7 \cdot V_{DD}$	–	$V_{DD} + 0.3$	V	For those pins with analog or oscillator function (20-/28-pin devices only), or when programmable pull-ups are enabled.
$V_{OL1}$	Low Level Output Voltage	–	–	0.4	V	$I_{OL} = 2\text{ mA}$ ; $V_{DD} = 3.0\text{ V}$ High Output Drive disabled.
$V_{OH1}$	High Level Output Voltage	2.4	–	–	V	$I_{OH} = -2\text{ mA}$ ; $V_{DD} = 3.0\text{ V}$ High Output Drive disabled.

Part Number	Flash	RAM	NVDS	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Comparator	Temperature Sensor	Description
<b>Z8 Encore! XP<sup>®</sup> F082A Series with 4 KB Flash</b>											
<b>Standard Temperature: 0 °C to 70 °C</b>											
Z8F041APB020SC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	PDIP 8-pin package
Z8F041AQB020SC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	QFN 8-pin package
Z8F041ASB020SC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	SOIC 8-pin package
Z8F041ASH020SC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SOIC 20-pin package
Z8F041AHH020SC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SSOP 20-pin package
Z8F041APH020SC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	PDIP 20-pin package
Z8F041ASJ020SC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SOIC 28-pin package
Z8F041AHJ020SC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SSOP 28-pin package
Z8F041APJ020SC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	PDIP 28-pin package
<b>Extended Temperature: -40 °C to 105 °C</b>											
Z8F041APB020EC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	PDIP 8-pin package
Z8F041AQB020EC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	QFN 8-pin package
Z8F041ASB020EC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	SOIC 8-pin package
Z8F041ASH020EC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SOIC 20-pin package
Z8F041AHH020EC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SSOP 20-pin package
Z8F041APH020EC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	PDIP 20-pin package
Z8F041ASJ020EC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SOIC 28-pin package
Z8F041AHJ020EC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SSOP 28-pin package
Z8F041APJ020EC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	PDIP 28-pin package
Replace C with G for Lead-Free Packaging											

## Part Number Suffix Designations

Z8 F 04 2A S H 020 S C

### Environmental Flow

C = Standard Plastic Packaging Compound  
G = Green Plastic Packaging Compound

### Temperature Range

S = Standard, 0 °C to 70 °C  
E = Extended, -40 °C to +105 °C

### Speed

020 = 20 MHz

### Pin Count

B = 8  
H = 20  
J = 28

### Package

H = SSOP  
P = PDIP  
Q = QFN  
S = SOIC

### Device Type

2A = Contains Advanced Analog Peripherals  
1A = Does Not Contain Advanced Analog Peripherals

### Memory Size

08 = 8 KB Flash, 1 KB RAM, 0 B NVDS  
04 = 4 KB Flash, 1 KB RAM, 128 B NVDS  
02 = 2 KB Flash, 512 B RAM, 64 B NVDS  
01 = 1 KB Flash, 256 B RAM, 16 B NVDS

### Memory Type

F = Flash

### Device Family

Z8 = Zilog's 8-Bit Microcontroller