Zilog - Z8F0411HH020SC00TR Datasheet

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	I²C, IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	11
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K 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	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0411hh020sc00tr

Signal and Pin Descriptions

Z8 Encore! XP fi F0822 Series products are available in a variety of packages, styles, and pin configurations. This chapter describes the signals and available pin configurations for each of the package styles. For information regarding the physical package specifications, see Packaging on page 233.

Available Packages

Table 2 identifies the package styles available for each device within the Z8 Encore! XP F0822 Series product line.

Table 2. Z8 Encore! XP F0822 Series Package Options

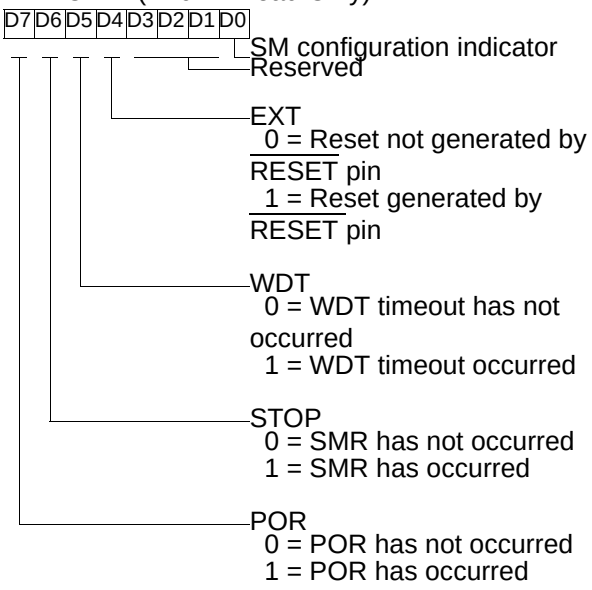
Part Number	10-Bit ADC	20-Pin SSOP	and PDIP	28-Pin SOIC and PDIP	
Z8F0822	Yes				X
Z8F0821	Yes		X		
Z8F0812	No				X
Z8F0811	No		X		
Z8F0422	Yes				X
Z8F0421	Yes		X		
Z8F0412	No				X
Z8F0411	No		X		

Pin Configurations

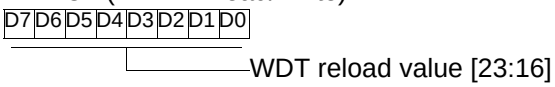
Figure 2 through Figure 5 display the pin configurations for all of the packages available in Z8 Encore! XP F0822 Series. See Table 4 for a description of the signals.

Note: The analog input alternate functions (ANAx) are not available on Z8 Encore! XP fi F0822 Series devices.

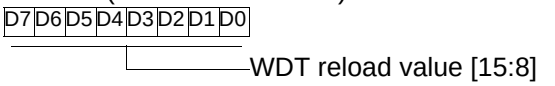
Watchdog Timer Control
WDTCTL (FF0H - Read Only)



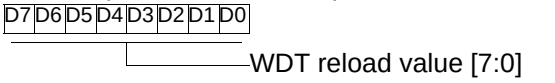
Watchdog Timer Reload Upper Byte
WDTU (FF1H - Read/Write)



Watchdog Timer Reload Middle Byte
WDTH (FF2H - Read/Write)



Watchdog Timer Reload Low Byte
WDTL (FF3H - Read/Write)



Port A C Output Data Register

The Port A C Output Data Register (Table 23) controls the output data to the pins.

Table 23. Port A C Output Data Register (P xOUT)

BITS	7	6	5	4	3	2	1	0
FIELD	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
RESET	0							
R/W	R/W							
ADDR	FD3H, FD7H, FDBH							

POUT[7:0] Port Output Data

These bits contain the data to be driven to the port pins. The values are only driven if the corresponding pin is configured as an output and the pin is not configured for alternate function operation.

0 = Drive a logical 0 (Low).

1 = Drive a logical 1 (High). High value is not driven if the drain has been disabled by setting the corresponding PC Output Control Register bit to 1.

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.

I2CI I2C Interrupt Request
0 = No interrupt request is pending for the I2C.
1 = An interrupt request from the I2C is awaiting service.

SPII SPI Interrupt Request
0 = No interrupt request is pending for the SPI.
1 = An interrupt request from the SPI is awaiting service.

ADCI ADC Interrupt Request
0 = No interrupt request is pending for the ADC.
1 = An interrupt request from the ADC is awaiting service.

Interrupt Request 1 Register

The Interrupt Request 1 (IRQ1) Register (Table 26) 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 reads the IRQ1 Register to determine if any interrupt requests are pending.

Table 26. Interrupt Request 1 Register (IRQ1)

BITS	7	6	5	4	3	2	1	0
FIELD	PA7I	PA6I	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
RESET	0							
R/W	R/W							
ADDR	FC3H							

PA*x*I 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 A pin number (0 through 7).

6. Write to the Timer Control Register to enable the timer and initiate counting.

In CONTINUOUS mode, the system clock always provides the timer input. The timer period is given by the following equation:

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

If an initial starting value other than 0001H is loaded into the Timer High and Low Byte Registers, the ONE-SHOT mode equation must be used to determine the first time-out period.

COUNTER Mode

In COUNTER mode, the timer counts input transitions from a GPIO port pin. The timer input is taken from the GPIO Port pin Timer Input alternate function. The TPOL bit in the Timer Control Register selects whether the count occurs on the rising edge or the falling edge of the Timer Input signal. In COUNTER mode, the prescaler is disabled.

! Caution: The input frequency of the Timer Input signal must not exceed one-fourth system clock frequency.

Upon reaching the Reload value stored in the Timer Reload High and Low Byte Registers, 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 COUNTER mode and 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 also sets the initial logic level (High or Low) for the Timer Output alternate function. However, the Timer Output function does not have 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 of 0001H. Generally, 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.

3. Write to the PWM High and Low Byte registers to set the PWM value.
4. Write to the Timer Reload High and Low Byte Registers to set the Reload value (PWM period). The Reload value must be greater than the PWM value.
5. If required, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
6. Configure the associated GPIO port pin for the Timer Output alternate function.
7. Write to the Timer Control Register to enable the timer and initiate counting.

The PWM period is given 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 is used to determine 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 given by

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{Reload Value} - \text{PWM Value}}{\text{Reload Value}} \times 100$$

If TPOL is set to 1, the ratio of the PWM output High time to the total period is given by

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{PWM Value}}{\text{Reload Value}} \times 100$$

CAPTURE Mode

In CAPTURE mode, the current timer count value is recorded when the desired external Timer Input transition occurs. The Capture count value is written to the Timer PWM High and Low Byte Registers. The timer input is the system clock. The TPOL bit in the Timer Control Register determines if the Capture occurs on a rising edge or a falling edge of the Timer Input signal. When the Capture event occurs, an interrupt is generated and the timer continues counting.

The timer continues counting up to the 16-bit Reload value stored in the Timer Reload High and Low Byte registers. Upon reaching the Reload value, the timer generates an interrupt and continues counting.

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

1. Write to the Timer Control Register to:

   Disable the timer

Enable signal asserts at least one UART bit period and no greater than two UART bit periods before the Start bit is transmitted. This format allows a setup time to enable the transceiver. The Driver Enable signal deasserts one system clock period after the last STOP bit is transmitted. This one system clock delay allows both time for data to clear the transceiver before disabling it, as well as the ability to determine if another character follows the current character. In the event of back to back characters (new data must be written to the Transmit Data Register before the previous character is completely transmitted) the DE signal is not deasserted between characters. The DEPOL bit in the UART Control Register 1 sets the polarity of the Driver Enable signal.

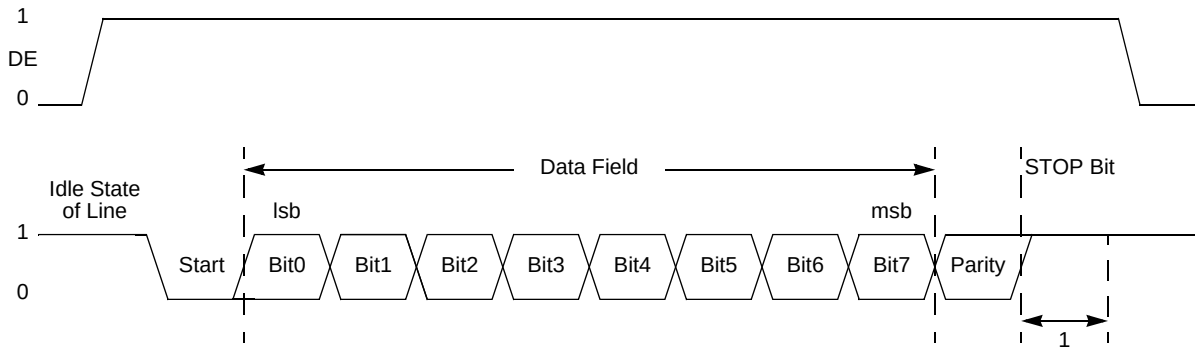


Figure 15. UART Driver Enable Signal Timing (with 1 STOP Bit and Parity)

The Driver Enable to Start bit setup time is calculated as follows:

$$\left(\frac{1}{\text{Baud Rate (Hz)}}\right) \leq \text{DE to Start Bit Setup Time (s)} \leq \left(\frac{2}{\text{Baud Rate (Hz)}}\right)$$

UART Interrupts

The UART features separate interrupts for the transmitter and the receiver. In addition, when the UART primary functionality is disabled, the BRG also functions as a basic timer with interrupt capability.

Transmitter Interrupts

The transmitter generates a single interrupt when the Transmit Data Register Empty bit (TDRE) is set to 1. This indicates that the transmitter is ready to accept new data for transmission. The TDRE interrupt occurs after the Transmit shift register has shifted the first bit of data out. At this point, the Transmit Data Register can be written with the next character to send. This provides 7 bit periods of latency to load the Transmit Data Register before the Transmit shift register completes shifting the current character. Writing to the UART Transmit Data Register clears the TDRE bit to 0.

Serial Peripheral Interface

The Serial Peripheral Interface (SPI) is a synchronous interface allowing several SPI-type devices to be interconnected. SPI-compatible devices include EEPROMs, Analog-to-Digital Converters, and ISDN devices. Features of the SPI include:

- Full-duplex, synchronous, and character-oriented communication
- Four-wire interface
- Data transfers rates up to a maximum of one-half the system clock frequency
- Error detection
- Dedicated Baud Rate Generator

The SPI is not available in 20-pin package devices.

Architecture

The SPI is be configured as either a Master (in single or multi-master systems) or a Slave as displayed in Figure 20 through Figure 22

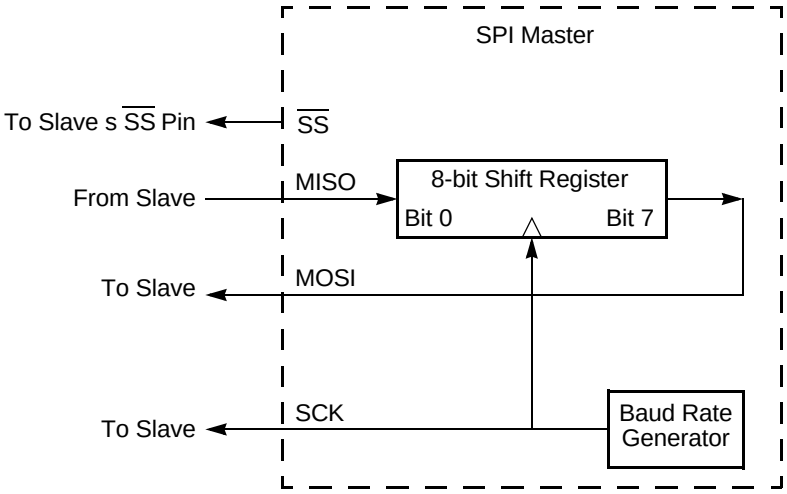


Figure 20. SPI Configured as a Master in a Single Master, Single Slave System

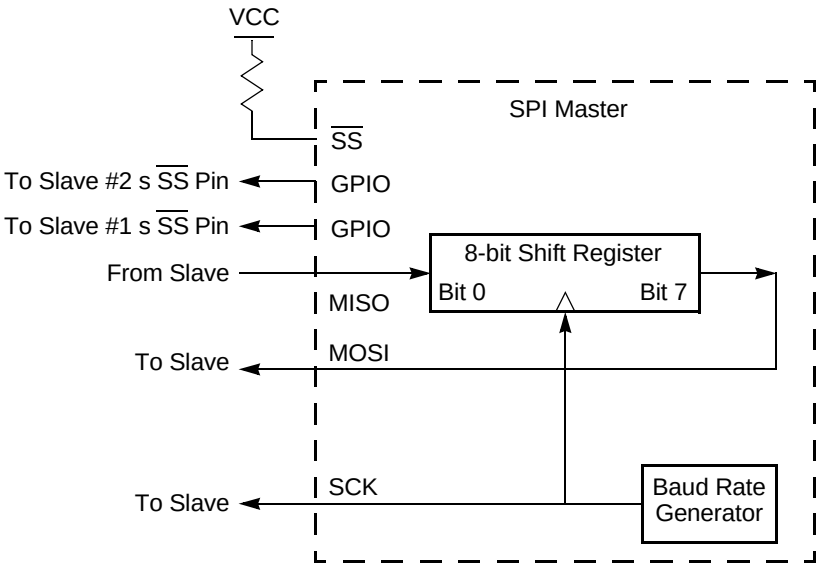


Figure 21. SPI Configured as a Master in a Single Master, Multiple Slave System

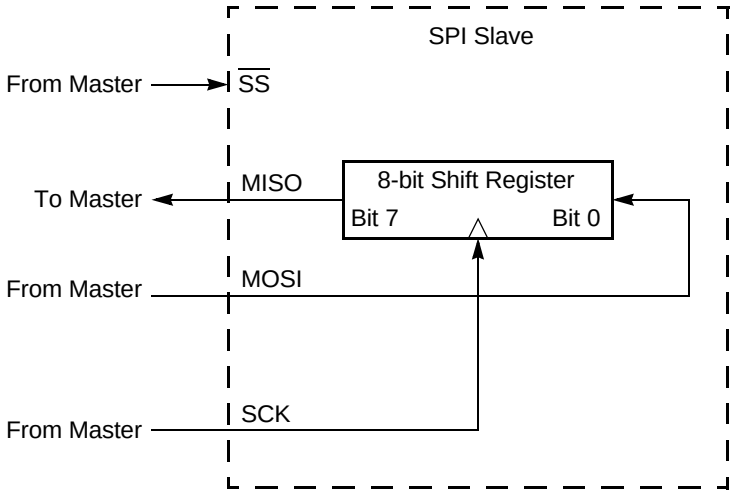


Figure 22. SPI Configured as a Slave

Operation

The SPI is a full-duplex, synchronous, and character-oriented channel that supports a four-wire interface (serial clock, transmit, receive and Slave select). The SPI block consists of a transmit/receive shift register, a Baud Rate (clock) Generator and a control unit.

TXRXSTATE	State Description
1_0111	10-bit addressing: Bit 0 (R/W) of 1st address byte
1_1000	10-bit addressing: Acknowledge state for 1st address byte
1_1001	10-bit addressing: Bit 7 of 2nd address byte 7-bit addressing: Bit 7 of address byte
1_1010	10-bit addressing: Bit 6 of 2nd address byte 7-bit addressing: Bit 6 of address byte
1_1011	10-bit addressing: Bit 5 of 2nd address byte 7-bit addressing: Bit 5 of address byte
1_1100	10-bit addressing: Bit 4 of 2nd address byte 7-bit addressing: Bit 4 of address byte
1_1101	10-bit addressing: Bit 3 of 2nd address byte 7-bit addressing: Bit 3 of address byte
1_1110	10-bit addressing: Bit 2 of 2nd address byte 7-bit addressing: Bit 2 of address byte
1_1111	10-bit addressing: Bit 1 of 2nd address byte 7-bit addressing: Bit 1 of address byte

I2C Diagnostic Control Register

The I2C Diagnostic register (Table 76) provides control over diagnostic modes. This register is a read/write register used for I2C diagnostics.

Table 76. I2C Diagnostic Control Register (I2CDIAG)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved							DIAG
RESET	0							
R/W	R							R/W
ADDR	F56H							

DIAG = Diagnostic Control Bit Selects read back value of the Baud Rate Reload registers.

0 = Normal mode. Reading the Baud Rate High and Low Byte registers returns the baud rate reload value.

1 = Diagnostic mode. Reading the Baud Rate High and Low Byte registers returns the baud rate counter value.

ADCD_L ADC Data Low Bits
These are the least significant two bits of the 10-bit ADC output. These bits are undefined after a Reset.

Reserved
These bits are reserved and are always undefined.

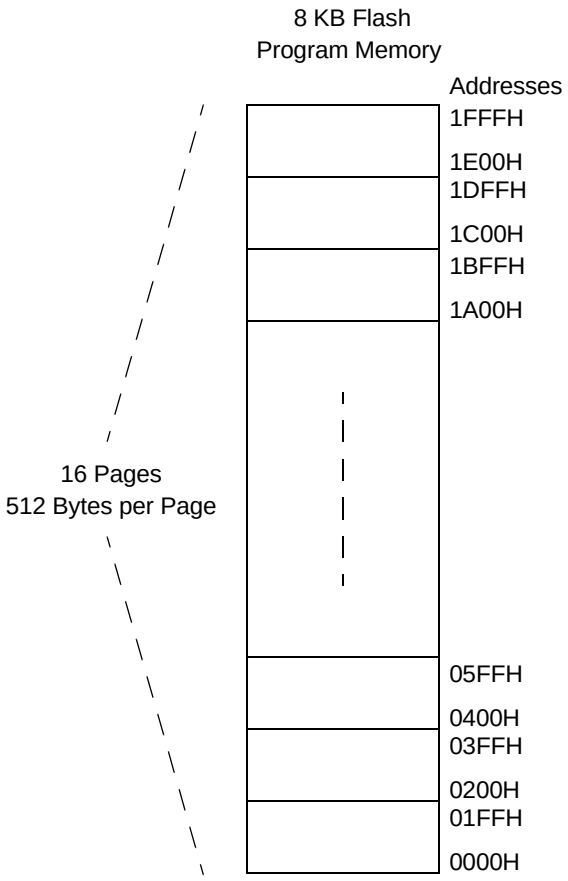


Figure 33. Flash Memory Arrangement

Information Area

Table 82 on page 155 describes the Z8 Encore! XP fi F0822 Series Information Area. This 512-byte Information Area is accessed by setting bit 7 of the Page Select Register to 1. When access is enabled, the Information Area is mapped into Flash Memory and overlays the 512 bytes at address FE00H to FFFFH. When the Information Area access is enabled, LDC instructions return data from the Information Area. CPU instruction fetches always comes from Flash Memory regardless of the Information Area access bit. Access to the Information Area is read-only.

Debug Mode

The operating characteristics of the Z8 Encore! XP fi F0822 Series devices in DEBUG mode are:

- The eZ8 CPU fetch unit stops, idling the eZ8 CPU, unless directed by the OCD to execute specific instructions.
- The system clock operates unless in STOP mode.
- All enabled on-chip peripherals operate unless in STOP mode.
- Automatically exits HALT mode.
- Constantly refreshes the Watchdog Timer, if enabled.

Entering Debug Mode

The device enters DEBUG mode following any of the following operations:

- Writing the DBGMODE bit in the OCD Control Register to 1 using the OCD interface.
- eZ8 CPU execution of a BRK (Breakpoint) instruction.
- Match of PC to OCDCNTR register (when enabled)
- OCDCNTR register decrements to 0000H (when enabled)
- If the DBG pin is Low when the device exits Reset, the OCD automatically puts the device into DEBUG mode.

Exiting Debug Mode

The device exits DEBUG mode following any of the following operations:

- Clearing the DBGMODE bit in the OCD Control Register to 0.
- Power-On Reset
- Voltage Brownout reset
- Asserting the $\overline{\text{RESET}}$ pin Low to initiate a Reset.
- Driving the DBG pin Low while the device is in STOP mode initiates a System Reset.

OCD Data Format

The OCD interface uses the asynchronous data format defined for RS-232. Each character is transmitted as 1 Start bit, 8 data bits (least-significant bit first), and 1 STOP bit (see Figure 40).

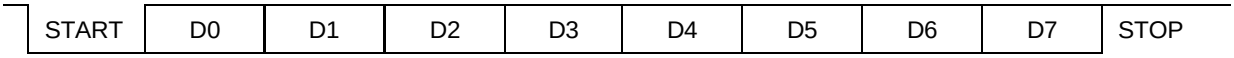


Figure 40. OCD Data Format

OCDCNTR Register

The OCD contains a multipurpose 16-bit Counter Register. It can be used for the following:

- Count system clock cycles between Breakpoints.
- Generate a BRK when it counts down to zero.
- Generate a BRK when its value matches the Program Counter.

When configured as a counter, the OCDCNTR register starts counting when the OCD leaves DEBUG mode and stops counting when it enters DEBUG mode again or when it reaches the maximum count of FFFFH. The OCDCNTR register automatically resets itself to 0000H when the OCD exits DEBUG mode if it is configured to count clock cycles between breakpoints.

! Caution: The OCDCNTR register is used by many of the OCD commands. It counts the number of bytes for the register and memory read/write commands. It holds the residual value when generating the CRC. Therefore, if the OCDCNTR is being used to generate a BRK, this value should be written as a last step before leaving DEBUG mode.

Since this register is overwritten by various OCD commands, it should only be used to generate temporary breakpoints, such as stepping over CALL instructions or running to a specific instruction and stopping.

On-Chip Debugger Commands

The host communicates to the OCD by sending OCD commands using the DBG interface. During normal operation, only a subset of the OCD commands are available. In DEBUG mode, all OCD commands become available unless the user code and control registers are protected by programming the Read Protect Option Bit (RPB). The Read Protect Option Bit prevents the code in memory from being read out of the Z8 Encore! XP F0822 Series products. When this option is enabled, several of the OCD commands are disabled. Table 93 on page 177 contains a summary of the OCD commands. Each OCD command is described further in the bulleted list. It also lists the commands that operate when the device is not in DEBUG mode (normal operation) and those commands that are disabled by programming the Read Protect Option Bit.

eZ8 CPU Instruction Classes

eZ8 CPU instructions are divided functionally into the following groups:

- Arithmetic
- Bit Manipulation
- Block Transfer
- CPU Control
- Load
- Logical
- Program Control
- Rotate and Shift

Tables 118 through Table 125 on page 218 contain the instructions belonging to each group and the number of operands required for each instruction. Some instructions appear in more than one table as these instructions can be considered as a subset of more than one category. Within these tables, the source operand is identified as src , the destination operand is dst and a condition code is cc .

Table 118. Arithmetic Instructions

Mnemonic	Operands	Instruction
ADC	dst, src	Add with Carry
ADCX	dst, src	Add with Carry using Extended Addressing
ADD	dst, src	Add
ADDX	dst, src	Add using Extended Addressing
CP	dst, src	Compare
CPC	dst, src	Compare with Carry
CPCX	dst, src	Compare with Carry using Extended Addressing
CPX	dst, src	Compare using Extended Addressing
DA	dst	Decimal Adjust
DEC	dst	Decrement
DECW	dst	Decrement Word
INC	dst	Increment
INCW	dst	Increment Word
MULT	dst	Multiply

Table 118. Arithmetic Instructions (Continued)

Mnemonic	Operands	Instruction
SBC	dst, src	Subtract with Carry
SBCX	dst, src	Subtract with Carry using Extended Addressing
SUB	dst, src	Subtract
SUBX	dst, src	Subtract using Extended Addressing

Table 119. Bit Manipulation Instructions

Mnemonic	Operands	Instruction
BCLR	bit, dst	Bit Clear
BIT	p, bit, dst	Bit Set or Clear
BSET	bit, dst	Bit Set
BSWAP	dst	Bit Swap
CCF		Complement Carry Flag
RCF		Reset Carry Flag
SCF		Set Carry Flag
TCM	dst, src	Test Complement Under Mask
TCMX	dst, src	Test Complement Under Mask using Extended Addressing
TM	dst, src	Test Under Mask
TMX	dst, src	Test Under Mask using Extended Addressing

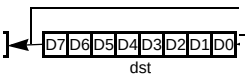
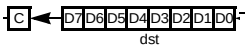
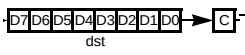
Table 120. Block Transfer Instructions

Mnemonic	Operands	Instruction
LDCI	dst, src	Load Constant to/from Program Memory and Auto-Increment Addresses
LDEI	dst, src	Load External Data to/from Data Memory and Auto-Increment Addresses

Table 126. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
DECW dst	dst ← dst - 1	RR		80	-	*	*	*	-	-	2	5
		IRR		81							2	6
DI	IRQCTL[7] ← 0			8F	-	-	-	-	-	-	1	2
DJNZ dst, RA	dst ← dst 1 if dst ≠ 0 PC ← PC + X	r		0A-FA	-	-	-	-	-	-	2	3
EI	IRQCTL[7] ← 1			9F	-	-	-	-	-	-	1	2
HALT	HALT Mode			7F	-	-	-	-	-	-	1	2
INC dst	dst ← dst + 1	R		20	-	*	*	*	-	-	2	2
		IR		21							2	3
		r		0E-FE							1	2
INCW dst	dst ← dst + 1	RR		A0	-	*	*	*	-	-	2	5
		IRR		A1							2	6
IRET	FLAGS ← @SP SP ← SP + 1 PC ← @SP SP ← SP + 2 IRQCTL[7] ← 1			BF	*	*	*	*	*	*	1	5
JP dst	PC ← dst	DA		8D	-	-	-	-	-	-	3	2
		IRR		C4							2	3
JP cc, dst	if cc is true PC ← dst	DA		0D-FD	-	-	-	-	-	-	3	2
JR dst	PC ← PC + X	DA		8B	-	-	-	-	-	-	2	2
JR cc, dst	if cc is true PC ← PC + X	DA		0B-FB	-	-	-	-	-	-	2	2

Table 126. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
POPX dst	dst ← @SP SP ← SP + 1	ER		D8	-	-	-	-	-	-	3	2
PUSH src	SP ← SP 1 @SP ← src	R		70	-	-	-	-	-	-	2	2
		IR		71							2	3
PUSHX src	SP ← SP 1 @SP ← src	ER		C8	-	-	-	-	-	-	3	2
RCF	C ← 0			CF	0	-	-	-	-	-	1	2
RET	PC ← @SP SP ← SP + 2			AF	-	-	-	-	-	-	1	4
RL dst		R		90	*	*	*	*	-	-	2	2
		IR		91							2	3
RLC dst		R		10	*	*	*	*	-	-	2	2
		IR		11							2	3
RR dst		R		E0	*	*	*	*	-	-	2	2
		IR		E1							2	3
RRC dst		R		C0	*	*	*	*	-	-	2	2
		IR		C1							2	3
SBC dst, src	dst ← dst src - C	r	r	32	*	*	*	*	1	*	2	3
		r	Ir	33							2	4
		R	R	34							3	3
		R	IR	35							3	4
		R	IM	36							3	3
		IR	IM	37							3	4
SBCX dst, src	dst ← dst src - C	ER	ER	38	*	*	*	*	1	*	4	3
		ER	IM	39							4	3

Ordering Information

Order Z8 Encore! XP fi F0822 Series from Zilog, using the following part numbers. For more information regarding ordering, consult your local Zilog sales office. Zilog website at www.zilog.com lists all regional offices and provides additional Z8 Encore! XP product information.

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	I^2C	SPI	UARTs with IrDA	Description
Z8F08xx with 8 KB Flash, 10-Bit Analog-to-Digital Converter										
Standard Temperature: 0 °C to 70 °C										
Z8F0821HH020SC	8 KB	1 KB	11	16	2	2	1	0	1	SSOP 20-pin package
Z8F0821PH020SC	8 KB	1 KB	11	16	2	2	1	0	1	PDIP 20-pin package
Z8F0822SJ020SC	8 KB	1 KB	19	19	2	5	1	1	1	SOIC 28-pin package
Z8F0822PJ020SC	8 KB	1 KB	19	19	2	5	1	1	1	PDIP 28-pin package
Extended Temperature: -40° to +105°C										
Z8F0821HH020EC	8 KB	1 KB	11	16	2	2	1	0	1	SSOP 20-pin package
Z8F0821PH020EC	8 KB	1 KB	11	16	2	2	1	0	1	PDIP 20-pin package
Z8F0822SJ020EC	8 KB	1 KB	19	19	2	5	1	1	1	SOIC 28-pin package
Z8F0822PJ020EC	8 KB	1 KB	19	19	2	5	1	1	1	PDIP 28-pin package

S

T