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, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	46
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	68-LCC (J-Lead)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f3202vs020ec

This publication is subject to replacement by a later edition. To determine whether a later edition exists, or to request copies of publications, contact:

ZiLOG Worldwide Headquarters
532 Race Street
San Jose, CA 95126
Telephone: 408.558.8500
Fax: 408.558.8300
www.ZiLOG.com

Document Disclaimer

ZiLOG is a registered trademark of ZiLOG Inc. in the United States and in other countries. All other products and/or service names mentioned herein may be trademarks of the companies with which they are associated.

©2004 by ZiLOG, Inc. All rights reserved. Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZiLOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZiLOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. Devices sold by ZiLOG, Inc. are covered by warranty and limitation of liability provisions appearing in the ZiLOG, Inc. Terms and Conditions of Sale. ZiLOG, Inc. makes no warranty of merchantability or fitness for any purpose Except with the express written approval of ZiLOG, use of information, devices, or technology as critical components of life support systems is not authorized. No licenses are conveyed, implicitly or otherwise, by this document under any intellectual property rights.

Table 3. Pin Characteristics of the Z8F640x family

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tri-State Output	Internal Pull-up or Pull-down	Schmitt Trigger Input	Open Drain Output
PF[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PG[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PH[3:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
$\overline{\text{RESET}}$	I	I	Low	N/A	Pull-up	Yes	N/A
VDD	N/A	N/A	N/A	N/A	No	No	N/A
XIN	I	I	N/A	N/A	No	No	N/A
XOUT	O	O	N/A	Yes, in Stop mode	No	No	No

x represents integer 0, 1,... to indicate multiple pins with symbol mnemonics that differ only by the integer

Table 4. Z8F640x Family Program Memory Maps (Continued)

Program Memory Address (Hex)	Function
Z8F480x Products	
0000-0001	Flash Option Bits
0002-0003	Reset Vector
0004-0005	WDT Interrupt Vector
0006-0007	Illegal Instruction Trap
0008-0037	Interrupt Vectors*
0038-BFFFH	Program Memory
Z8F640x Products	
0000-0001	Flash Option Bits
0002-0003	Reset Vector
0004-0005	WDT Interrupt Vector
0006-0007	Illegal Instruction Trap
0008-0037	Interrupt Vectors*
0038-FFFFH	Program Memory

*** See Table 22 on page 45 for a list of the interrupt vectors.**

Data Memory

The Z8F640x family devices contain 128 bytes of read-only memory at the top of the eZ8 CPU's 64KB Data Memory address space. The eZ8 CPU's LDE and LDEI instructions provide access to the Data Memory information. Table 5 describes the Z8F640x family's Data Memory Map.

Table 5. Z8F640x family Data Memory Maps

Data Memory Address (Hex)	Function
0000H-FFBFH	Reserved
FFC0H-FFD3H	Part Number 20-character ASCII alphanumeric code Left justified and filled with zeros
FFD4H-FFFFH	Reserved

Table 6. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page #
Timer 3 (not available in 40- and 44- Pin Packages)				
F18	Timer 3 High Byte	T3H	00	66
F19	Timer 3 Low Byte	T3L	01	66
F1A	Timer 3 Reload High Byte	T3RH	FF	67
F1B	Timer 3 Reload Low Byte	T3RL	FF	67
F1C	Timer 3 PWM High Byte	T3PWMH	00	69
F1D	Timer 3 PWM Low Byte	T3PWML	00	69
F1E	Reserved	—	XX	
F1F	Timer 3 Control	T3CTL	00	70
F20-F3F	Reserved	—	XX	
UART 0				
F40	UART0 Transmit Data	U0TXD	XX	86
	UART0 Receive Data	U0RXD	XX	87
F41	UART0 Status 0	U0STAT0	0000011Xb	87
F42	UART0 Control 0	U0CTL0	00	89
F43	UART0 Control 1	U0CTL1	00	89
F44	UART0 Status 1	U0STAT1	00	87
F45	Reserved	—	XX	
F46	UART0 Baud Rate High Byte	U0BRH	FF	91
F47	UART0 Baud Rate Low Byte	U0BRL	FF	91
UART 1				
F48	UART1 Transmit Data	U1TXD	XX	86
	UART1 Receive Data	U1RXD	XX	87
F49	UART1 Status 0	U1STAT0	0000011Xb	87
F4A	UART1 Control 0	U1CTL0	00	89
F4B	UART1 Control 1	U1CTL1	00	89
F4C	UART1 Status 1	U1STAT1	00	87
F4D	Reserved	—	XX	
F4E	UART1 Baud Rate High Byte	U1BRH	FF	91
F4F	UART1 Baud Rate Low Byte	U1BRL	FF	91
I^2C				
F50	I^2C Data	I2CDATA	00	118
F51	I^2C Status	I2CSTAT	80	118
F52	I^2C Control	I2CCTL	00	119
F53	I^2C Baud Rate High Byte	I2CBRH	FF	121
F54	I^2C Baud Rate Low Byte	I2CBRL	FF	121
F55-F5F	Reserved	—	XX	
Serial Peripheral Interface (SPI)				
F60	SPI Data	SPIDATA	XX	106
F61	SPI Control	SPICTL	00	107

XX=Undefined

Table 6. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page #
FED	Port H Control	PHCTL	00	38
FEE	Port H Input Data	PHIN	XX	42
FEF	Port H Output Data	PHOUT	00	43
Watch-Dog Timer (WDT)				
FF0	Watch-Dog Timer Control	WDTCTL	XXX00000b	75
FF1	Watch-Dog Timer Reload Upper Byte	WDTU	FF	76
FF2	Watch-Dog Timer Reload High Byte	WDTH	FF	76
FF3	Watch-Dog Timer Reload Low Byte	WDTL	FF	76
FF4--FF7	Reserved	—	XX	
Flash Memory Controller				
FF8	Flash Control	FCTL	00	144
FF8	Flash Status	FSTAT	00	145
FF9	Flash Page Select	FPS	00	146
FFA	Flash Programming Frequency High Byte	FFREQH	00	147
FFB	Flash Programming Frequency Low Byte	FFREQL	00	147
eZ8 CPU				
FFC	Flags	—	XX	Refer to the *eZ8 CPU User Manual*
FFD	Register Pointer	RP	XX	
FFE	Stack Pointer High Byte	SPH	XX	
FFF	Stack Pointer Low Byte	SPL	XX	

XX=Undefined

External Pin Reset

The $\overline{\text{RESET}}$ pin has a Schmitt-triggered input and an internal pull-up. Once the $\overline{\text{RESET}}$ pin is asserted, the device progresses through the Short Reset sequence. While the $\overline{\text{RESET}}$ input pin is asserted Low, the Z8F640x family device continues to be held in the Reset state. If the $\overline{\text{RESET}}$ pin is held Low beyond the Short Reset time-out, the device exits the Reset state immediately following $\overline{\text{RESET}}$ pin deassertion. Following a Short Reset initiated by the external $\overline{\text{RESET}}$ pin, the EXT status bit in the Watch-Dog Timer Control (WDTCTL) register is set to 1.

Stop Mode Recovery

Stop mode is entered by execution of a STOP instruction by the eZ8 CPU. Refer to the **Low-Power Modes** chapter for detailed Stop mode information. During Stop Mode Recovery, the Z8F640x family device is held in reset for 514 cycles of the Watch-Dog Timer oscillator followed by 16 cycles of the system clock (crystal oscillator). Stop Mode Recovery does not affect any values in the Register File, including the Stack Pointer, Register Pointer, Flags and general-purpose RAM.

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 Watch-Dog Timer Control Register is set to 1. Table 9 lists the Stop Mode Recovery sources and resulting actions. The text following provides more detailed information on each of the Stop Mode Recovery sources.

Table 9. Stop Mode Recovery Sources and Resulting Action

Operating Mode	Stop Mode Recovery Source	Action
Stop mode	Watch-Dog Timer time-out when configured for Reset	Stop Mode Recovery
	Watch-Dog 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

Stop Mode Recovery Using Watch-Dog Timer Time-Out

If the Watch-Dog Timer times out during Stop mode, the Z8F640x family device undergoes a STOP Mode Recovery sequence. In the Watch-Dog Timer Control register, the WDT and STOP bits are set to 1. If the Watch-Dog Timer is configured to generate an interrupt upon time-out and the device is configured to respond to interrupts, the Z8F640x family device services the Watch-Dog Timer interrupt request following the normal Stop Mode Recovery sequence.

Architecture

Figure 65 illustrates a block diagram of the interrupt controller.

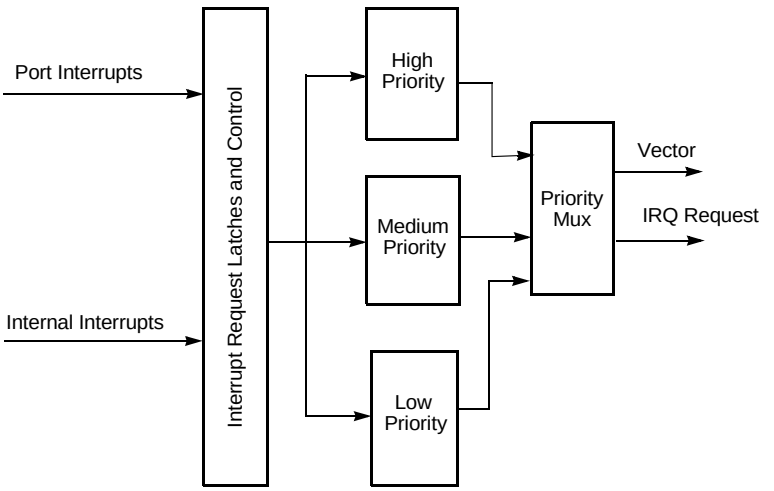

Figure 65. Interrupt Controller Block Diagram

Operation

Master Interrupt Enable

The master interrupt enable bit (IRQE) in the Interrupt Control register globally enables and disables interrupts.

Interrupts are globally enabled by any of the following actions:

- Execution of an EI (Enable Interrupt) instruction
- Execution of an IRET (Return from Interrupt) instruction
- Writing a 1 to the IRQE bit in the Interrupt Control register

Interrupts are globally disabled by any of the following actions:

- Execution of a DI (Disable Interrupt) instruction
- eZ8 CPU acknowledgement of an interrupt service request from the interrupt controller
- Writing a 0 to the IRQE bit in the Interrupt Control register
- Reset

written during counting, the 8-bit written value is placed in the counter (High or Low Byte) at the next clock edge. The counter continues counting from the new value.

Table 38. Timer 0-3 High Byte Register (TxH)

BITS	7	6	5	4	3	2	1	0
FIELD	TH							
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	F00H, F08H, F10H, F18H							

Table 39>. Timer 0-3 Low Byte Register (TxL)

BITS	7	6	5	4	3	2	1	0
FIELD	TL							
RESET	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F01H, F09H, F11H, F19H							

TH and TL—Timer High and Low Bytes
These 2 bytes, {TMRH[7:0], TMRL[7:0]}, contain the current 16-bit timer count value.

Timer Reload High and Low Byte Registers

The Timer 0-3 Reload High and Low Byte (TxRH and TxRL) registers (Tables 40 and 41) store a 16-bit reload value, {TRH[7:0], TRL[7:0]}. Values written to the Timer Reload High Byte register are stored in a temporary holding register. When a write to the Timer Reload Low Byte register occurs, the temporary holding register value is written to the Timer High Byte register. This operation allows simultaneous updates of the 16-bit Timer Reload value.

In Compare mode, the Timer Reload High and Low Byte registers store the 16-bit Compare value.

In single-byte DMA transactions to the Timer Reload High Byte register, the temporary holding register is bypassed and the value is written directly to the register. If the DMA is

CTSE—CTS Enable
0 = The CTS signal has no effect on the transmitter.
1 = The UART recognizes the 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 insure that the transmitter has finished sending data before setting this bit.
0 = No break is sent.
1 = The output of the transmitter is 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 55. UART*x* Control 1 Register (U*x*CTL1)

BITS	7	6	5	4	3	2	1	0
FIELD	BIRQ	MPM	MPE	MPBT	Reserved		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 and F4BH							

BIRQ—Baud Rate Generator Interrupt Request
This bit sets an interrupt request when the Baud Rate Generator times out and is only set if a UART is not enabled. The is bit produces no effect when the UART is enabled.
0 = Interrupts behave as set by UART control.
1 = The Baud Rate Generator generates a receive interrupt when it counts down to zero.

MPM—Multiprocessor (9-bit) mode Select
This bit is used to enable Multiprocessor (9-bit) mode.

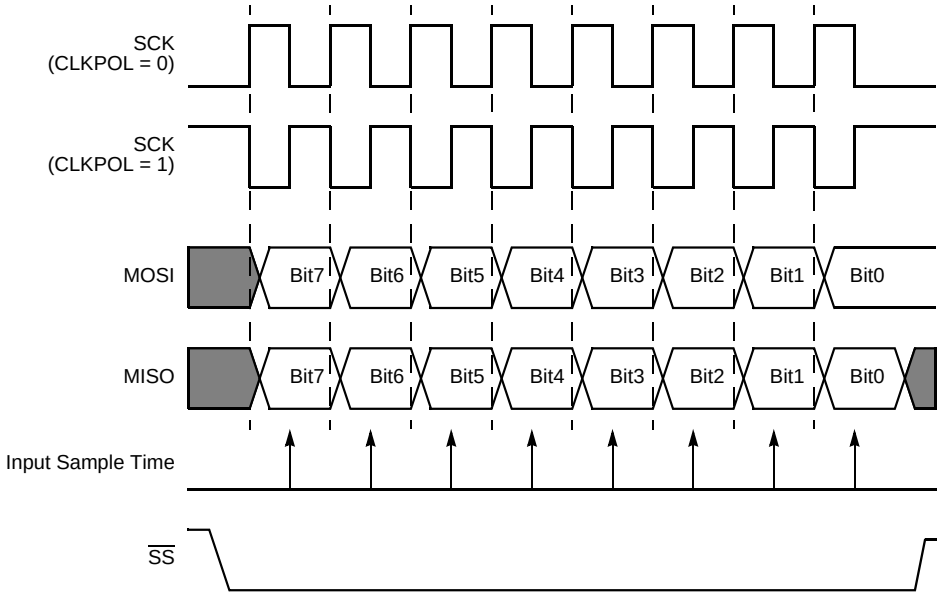

Figure 78. SPI Timing When PHASE is 1

Multi-Master Operation

In a multi-master SPI system, all SCK pins are tied together, all MOSI pins are tied together and all MISO pins are tied together. All SPI pins must then be configured in open-drain mode to prevent bus contention. At any one time, only one SPI device is configured as the Master and all other SPI devices on the bus are configured as Slaves. The Master enables a single Slave by asserting the $\overline{SS}$ pin on that Slave only. Then, the single Master drives data out its SCK and MOSI pins to the SCK and MOSI pins on the Slaves (including those which are not enabled). The enabled Slave drives data out its MISO pin to the MISO Master pin.

For a Master device operating in a multi-master system, if the $\overline{SS}$ pin is configured as an input and is driven Low by another Master, the COL bit is set to 1 in the SPI Status Register. The COL bit indicates the occurrence of a multi-master collision (mode fault error condition).

4. The I2C Controller loads the I2C Shift register with the contents of the I2C Data register.
5. After the first bit has been shifted out, a Transmit interrupt is asserted.
6. Software responds by writing eight bits of address to the I2C Data register.
7. The I2C Controller completes shifting of the two address bits and a 0 (write).
8. The I2C slave sends an acknowledge by pulling the SDA signal Low during the next high period of SCL.
9. The I2C Controller loads the I2C Shift register with the contents of the I2C Data register.
10. The I2C Controller shifts out the next eight bits of address. After the first bits are shifted, the I2C Controller generates a Transmit interrupt.
11. Software responds by setting the START bit of the I2C Control register to generate a repeated START.
12. Software responds by writing 11110B followed by the 2-bit slave address and a 1 (read).
13. Software responds by setting the NAK bit of the I2C Control register, so that a Not Acknowledge is sent after the first byte of data has been read. If you want to read only one byte, software responds by setting the NAK bit of the I2C Control register.
14. After the I2C Controller shifts out the address bits mentioned in step 9, the I2C slave sends an acknowledge by pulling the SDA signal Low during the next high period of SCL.
15. The I2C Controller sends the repeated START condition.
16. The I2C Controller loads the I2C Shift register with the contents of the I2C Data register.
17. The I2C Controller sends 11110B followed by the 2-bit slave read and a 1 (read).
18. The I2C slave sends an acknowledge by pulling the SDA signal Low during the next high period of SCL.
19. The I2C slave sends a byte of data.
20. A Receive interrupt is generated.
21. Software responds by reading the I2C Data register.
22. Software responds by setting the STOP bit of the I2C Control register.
23. A NAK condition is sent to the I2C slave.
24. A STOP condition is sent to the I2C slave.

received a byte of data. When active, this bit causes the I2C Controller to generate an interrupt. This bit is cleared by reading the I2C Data register.

ACK—Acknowledge
This bit indicates the status of the Acknowledge for the last byte transmitted or received. When set, this bit indicates that an Acknowledge was received for the last byte transmitted or received.

10B—10-Bit Address
This bit indicates whether a 10- or 7-bit address is being transmitted. After the START bit is set, if the five most-significant bits of the address are 11110B, this bit is set. When set, it is reset once the first byte of the address has been sent.

RD—Read
This bit indicates the direction of transfer of the data. It is active high during a read. The status of this bit is determined by the least-significant bit of the I2C Shift register after the START bit is set.

TAS—Transmit Address State
This bit is active high while the address is being shifted out of the I2C Shift register.

DSS—Data Shift State
This bit is active high while data is being transmitted to or from the I2C Shift register.

NCKI—NACK Interrupt
This bit is set high when a Not Acknowledge condition is received or sent and neither the START nor the STOP bit is active. When set, this bit generates an interrupt that can only be cleared by setting the START or STOP bit, allowing the user to specify whether he wants to perform a STOP or a repeated START.

I2C Control Register

The I2C Control register enables the I2C operation.

Table 68. I2C Control Register (I2CCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	IEN	START	STOP	BIRQ	TXI	NAK	FLUSH	FILTEN
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	F52H							

IEN—I2C Enable
This bit enables the I2C transmitter and receiver.

DMA*x* Start/Current Address Low Byte Register

The DMA*x* Start/Current Address Low register, in conjunction with the DMA*x* Address High Nibble register, forms a 12-bit Start/Current Address. Writes to this register set the Start Address for DMA operations. Each time the DMA completes a data transfer, the 12-bit Start/Current Address increments by either 1 (single-byte transfer) or 2 (two-byte word transfer). Reads from this register return the low byte of the Current Address to be used for the next DMA data transfer.

Table 74. DMA*x* Start/Current Address Low Byte Register (DMA*x*START)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_START							
RESET	X	X	X	X	X	X	X	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FB3H, FHBH							

DMA_START—DMA*x* Start/Current Address Low
These bits, with the four lower bits of the DMA*x*_H register, form the 12-bit Start/Current address. The full 12-bit address is given by {DMA_START_H[3:0], DMA_START[7:0]}.

DMA*x* End Address Low Byte Register

The DMA*x* End Address Low Byte register, in conjunction with the DMA*x*_H register, forms a 12-bit End Address.

Table 75. DMA*x* End Address Low Byte Register (DMA*x*END)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_END							
RESET	X	X	X	X	X	X	X	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FB4H, FBCH							

DMA_END—DMA*x* End Address Low
These bits, with the four upper bits of the DMA*x*_H register, form a 12-bit address. This address is the ending location of the DMA*x* transfer. The full 12-bit address is given by {DMA_END_H[3:0], DMA_END[7:0]}.

DMA_ADC Address Register

The DMA_ADC Address register points to a block of the Register File to store ADC conversion values as illustrated in Table 76. This register contains the seven most-significant bits of the 12-bit Register File addresses. The five least-significant bits are calculated from the ADC Analog Input number (5-bit base address is equal to twice the ADC Analog Input number). The 10-bit ADC conversion data is stored as two bytes with the most significant byte of the ADC data stored at the even numbered Register File address.

On-Chip Debugger

Overview

The Z8F640x family devices have an integrated On-Chip Debugger (OCD) that provides advanced debugging features including:

- Reading and writing of the Register File
- Reading and writing of Program and Data Memory
- Setting of Breakpoints and Watchpoints
- Execution of eZ8 CPU instructions.

Architecture

The On-Chip Debugger consists of four primary functional blocks: transmitter, receiver, auto-baud generator, and debug controller. Figure 86 illustrates the architecture of the On-Chip Debugger

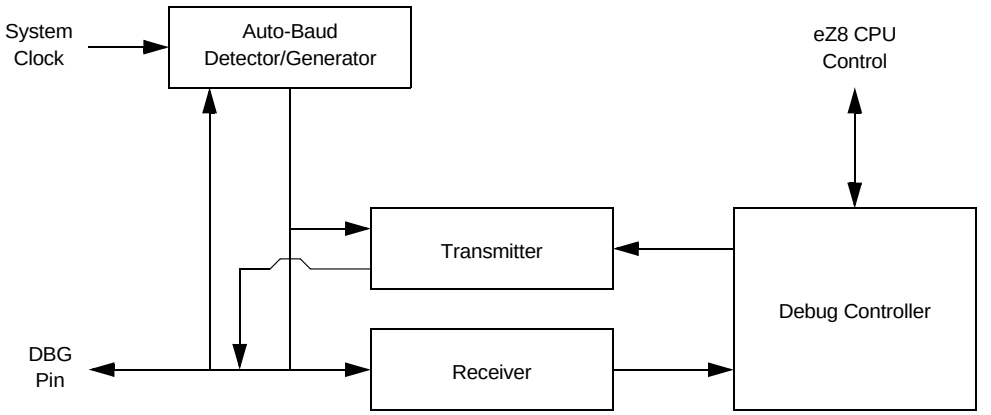

Figure 86. On-Chip Debugger Block Diagram

Table 106. Analog-to-Digital Converter Electrical Characteristics and Timing (Continued)

Symbol	Parameter	V_{DD} = 3.0 - 3.6V T_A = -40^{0}C to 105^{0}C			Units	Conditions
		Minimum	Typical	Maximum		
	DC Offset Error	-50	–	25	mV	40-pin PDIP, 44-pin LQFP, 44-pin PLCC, and 68-pin PLCC packages.
V_{REF}	Internal Reference Voltage	–	2.0	–	V	
	Single-Shot Conversion Time	–	5129	–	cycles	System clock cycles
	Continuous Conversion Time	–	256	–	cycles	System clock cycles
	Sampling Rate	System Clock / 256			Hz	
	Signal Input Bandwidth	–	–	3.5	kHz	
R_S	Analog Source Impedance	–	–	10[1]	kΩ	
Zin	Input Impedance		150		kΩ	20MHz system clock. Input impedance increases with lower system clock frequency.
V_{REF}	External Reference Voltage			AVDD	V	AVDD <= VDD. When using an external reference voltage, decoupling capacitance should be placed from VREF to AVSS.

[1] Analog source impedance affects the ADC offset voltage (because of pin leakage) and input settling time.

SPI Master Mode Timing

Figure 96 and Table 110 provide timing information for SPI Master mode pins. Timing is shown with SCK rising edge used to source MOSI output data, SCK falling edge used to sample MISO input data. Timing on the SS output pin(s) is controlled by software.

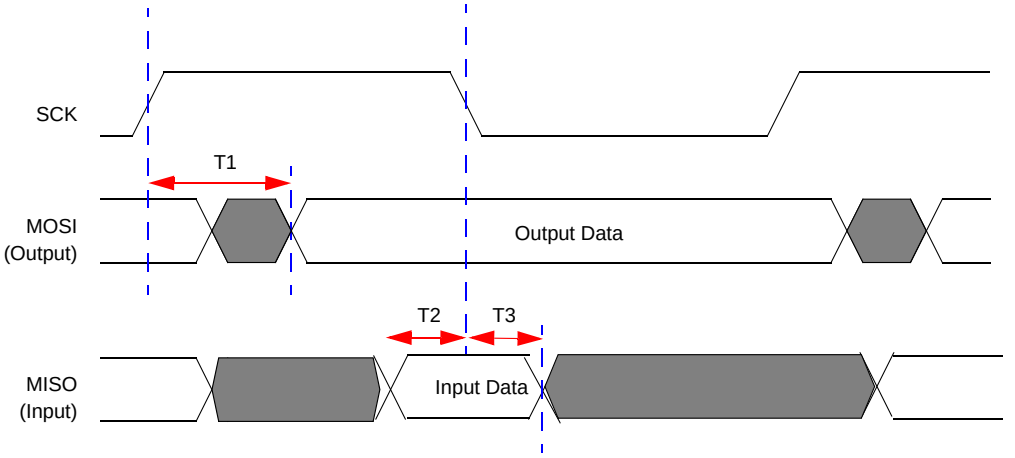

Figure 96. SPI Master Mode Timing

Table 110. SPI Master Mode Timing

		Delay (ns)	
Parameter	**Abbreviation**	**Minimum**	**Maximum**
T_1	SCK Rise to MOSI output Valid Delay	-5	+5
T_2	MISO input to SCK (receive edge) Setup Time	20	
T_3	MISO input to SCK (receive edge) Hold Time	0	

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
XOR dst, src	dst ← dst XOR src	r	r	B2	-	*	*	0	-	-	2	3
		r	Ir	B3							2	4
		R	R	B4							3	3
		R	IR	B5							3	4
		R	IM	B6							3	3
		IR	IM	B7							3	4
XORX dst, src	dst ← dst XOR src	ER	ER	B8	-	*	*	0	-	-	4	3
		ER	IM	B9							4	3
Flags Notation:	* = Value is a function of the result of the operation. - = Unaffected X = Undefined				0 = Reset to 0 1 = Set to 1							

Opcode Maps

Figures 101 and 102 provide information on each of the eZ8 CPU instructions. A description of the opcode map data and the abbreviations are provided in Figure 100 and Table 127.

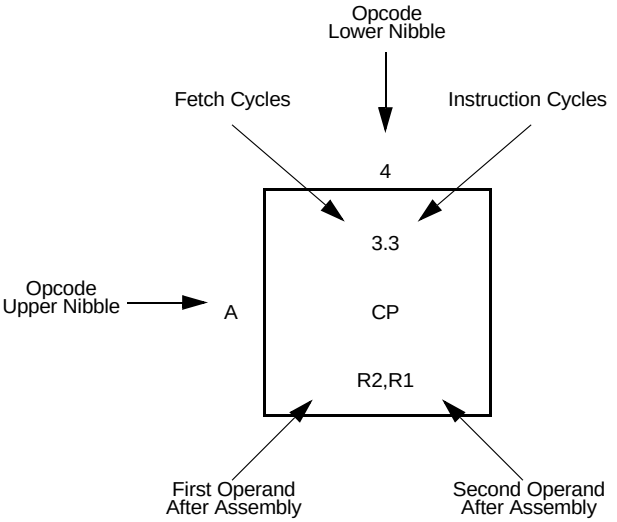

Figure 100. Opcode Map Cell Description

Table 128. Ordering Information (Continued)

Part	Flash KB (Bytes)	RAM KB (Bytes)	Max. Speed (MHz)	Temp (^{0}C)	Voltage (V)	Package	Part Number
Z8 Encore! ®with 48KB Flash, Extended Temperature							
Z8 Encore!®	48 (49,152)	4 (4096)	20	-40 to +105	3.0 - 3.6	PDIP-40	Z8F4801PM020EC
Z8 Encore!®	48 (49,152)	4 (4096)	20	-40 to +105	3.0 - 3.6	LQFP-44	Z8F4801AN020EC
Z8 Encore!®	48 (49,152)	4 (4096)	20	-40 to +105	3.0 - 3.6	PLCC-44	Z8F4801VN020EC
Z8 Encore!®	48 (49,152)	4 (4096)	20	-40 to +105	3.0 - 3.6	LQFP-64	Z8F4802AR020EC
Z8 Encore!®	48 (49,152)	4 (4096)	20	-40 to +105	3.0 - 3.6	PLCC-68	Z8F4802VS020EC
Z8 Encore!®	48 (49,152)	4 (4096)	20	-40 to +105	3.0 - 3.6	QFP-80	Z8F4803FT020EC
Z8 Encore!® with 64KB Flash, Extended Temperature							
Z8 Encore!®	64 (65,536)	4 (4096)	20	-40 to +105	3.0 - 3.6	PDIP-40	Z8F6401PM020EC
Z8 Encore!®	64 (65,536)	4 (4096)	20	-40 to +105	3.0 - 3.6	LQFP-44	Z8F6401AN020EC
Z8 Encore!®	64 (65,536)	4 (4096)	20	-40 to +105	3.0 - 3.6	PLCC-44	Z8F6401VN020EC
Z8 Encore!®	64 (65,536)	4 (4096)	20	-40 to +105	3.0 - 3.6	LQFP-64	Z8F6402AR020EC
Z8 Encore!v	64 (65,536)	4 (4096)	20	-40 to +105	3.0 - 3.6	PLCC-68	Z8F6402VS020EC
Z8 Encore!®	64 (65,536)	4 (4096)	20	-40 to +105	3.0 - 3.6	QFP-80	Z8F6403FT020EC
Z8 Encore! ®Development Tools							
Z8 Encore!® Developer Kit							Z8ENCORE000ZCO

Contact ZILOG's worldwide customer support center for more information on ordering the Z8 Encore!®. The customer support center is open from 7 a.m. to 7 p.m. Pacific Time.

The customer support toll-free number for ZiLOG is 1-877-ZiLOGCS (1-877-945-6427). For Z8 Encore!® the customer support toll-free number is 1-866-498-3636. The FAX number for the customer support center is 1-603-316-0345. Customers can also gain access to customer support using the ZiLOG website. Z8 Encore!® has its own web page at www.zilog.com/z8encore.

For customer service, navigate your browser to:

- http://register.zilog.com/login.asp?login = servicelogin

For technical support, navigate your browser to:

- http://register.zilog.com/login.asp?login = supportlogin