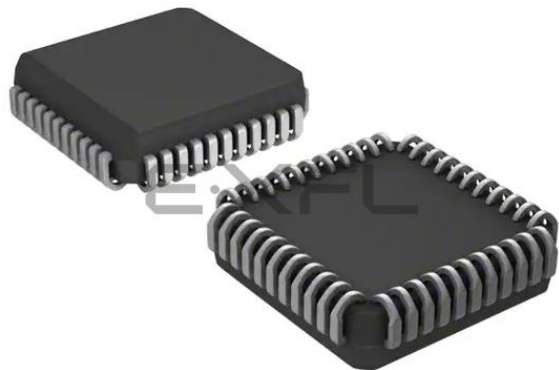




Welcome to [E-XFL.COM](https://www.e-xfl.com)

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	I ² C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	31
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 8x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f3201vn020sc00tr



Manual Objectives

This Product Specification provides detailed operating information for the Z8F640x, Z8F480x, Z8F320x, Z8F240x, and Z8F160x devices within the Z8 Encore!™ Microcontroller (MCU) family of products. Within this document, the Z8F640x, Z8F480x, Z8F320x, Z8F240x, and Z8F160x are referred to collectively as Z8 Encore!™ or the Z8F640x family unless specifically stated otherwise.

About This Manual

ZiLOG recommends that the user read and understand everything in this manual before setting up and using the product. However, we recognize that there are different styles of learning. Therefore, we have designed this Product Specification to be used either as a *how to* procedural manual or a reference guide to important data.

Intended Audience

This document is written for ZiLOG customers who are experienced at working with microcontrollers, integrated circuits, or printed circuit assemblies.

Manual Conventions

The following assumptions and conventions are adopted to provide clarity and ease of use:

Courier Typeface

Commands, code lines and fragments, bits, equations, hexadecimal addresses, and various executable items are distinguished from general text by the use of the *Courier* typeface. Where the use of the font is not indicated, as in the Index, the name of the entity is presented in upper case.

- Example: `FLAGS[1]` is `smrf`.

Hexadecimal Values

Hexadecimal values are designated by uppercase *H* suffix and appear in the *Courier* typeface.

- Example: R1 is set to `F8H`.

Brackets

The square brackets, `[ ]`, indicate a register or bus.

- Example: for the register `R1[7:0]`, R1 is an 8-bit register, `R1[7]` is the most significant bit, and `R1[0]` is the least significant bit.

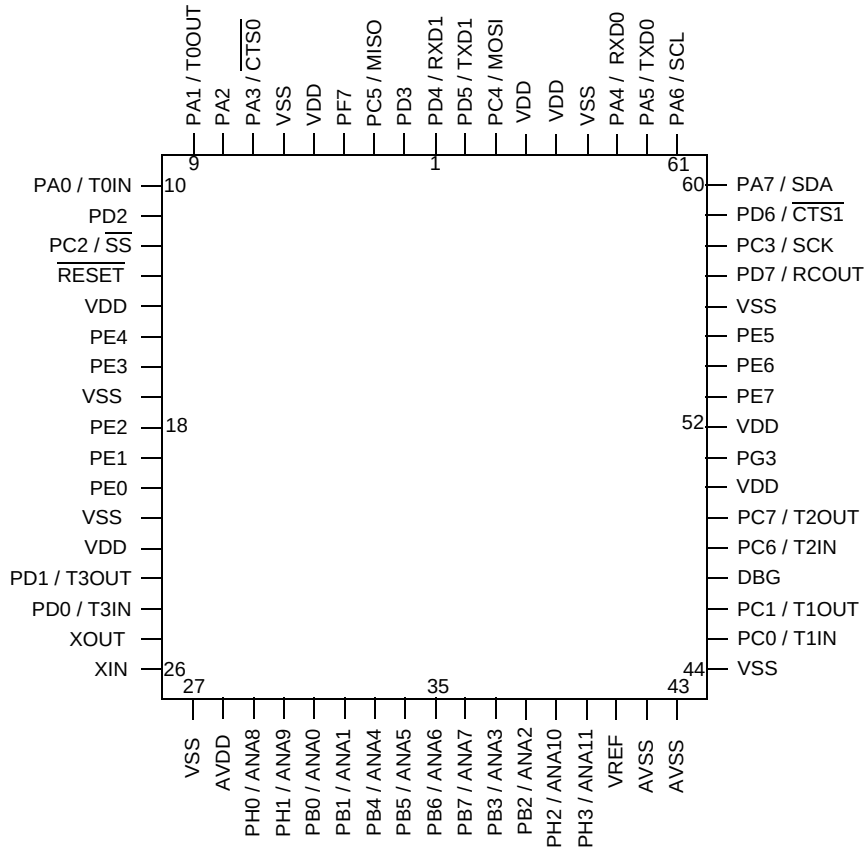


Figure 60. Z8Fxx02 in 68-Pin Plastic Leaded Chip Carrier (PLCC)



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
I²C				
F50	I ² C Data	I2CDATA	00	118
F51	I ² C Status	I2CSTAT	80	118
F52	I ² C Control	I2CCTL	00	119
F53	I ² C Baud Rate High Byte	I2CBRH	FF	121
F54	I ² C 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 19. Port A-H STOP Mode Recovery Source Enable Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	PSMRE7	PSMRE6	PSMRE5	PSMRE4	PSMRE3	PSMRE2	PSMRE1	PSMRE0
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	If 05H in Port A-H Address Register, accessible via Port A-H Control Register							

PSMRE[7:0]—Port STOP Mode Recovery Source Enabled

0 = The Port pin is not configured as a STOP Mode Recovery source. Transitions on this pin during Stop mode do not initiate STOP Mode Recovery.

1 = The Port pin is configured as a STOP Mode Recovery source. Any logic transition on this pin during Stop mode initiates STOP Mode Recovery.

Port A-H Input Data Registers

Reading from the Port A-H Input Data registers (Table 20) returns the sampled values from the corresponding port pins. The Port A-H Input Data registers are Read-only.

Table 20. Port A-H Input Data Registers (PxIN)

BITS	7	6	5	4	3	2	1	0
FIELD	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X	X	X	X	X	X	X	X
R/W	R	R	R	R	R	R	R	R
ADDR	FD2H, FD6H, FDAH, FDEH, FE2H, FE6H, FEAH, FEEH							

PIN[7:0]—Port Input Data

Sampled data from the corresponding port pin input.

0 = Input data is logical 0 (Low).

1 = Input data is logical 1 (High).

Table 33. IRQ2 Enable High Bit Register (IRQ2ENH)

BITS	7	6	5	4	3	2	1	0
FIELD	T3ENH	U1RENH	U1TENH	DMAENH	C3ENH	C2ENH	C1ENH	C0ENH
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	FC7H							

T3ENH—Timer 3 Interrupt Request Enable High Bit
 U1RENH—UART 1 Receive Interrupt Request Enable High Bit
 U1TENH—UART 1 Transmit Interrupt Request Enable High Bit
 DMAENH—DMA Interrupt Request Enable High Bit
 C3ENH—Port C3 Interrupt Request Enable High Bit
 C2ENH—Port C2 Interrupt Request Enable High Bit
 C1ENH—Port C1 Interrupt Request Enable High Bit
 C0ENH—Port C0 Interrupt Request Enable High Bit

Table 34. IRQ2 Enable Low Bit Register (IRQ2ENL)

BITS	7	6	5	4	3	2	1	0
FIELD	T3ENL	U1RENL	U1TENL	DMAENL	C3ENL	C2ENL	C1ENL	C0ENL
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	FC8H							

T3ENL—Timer 3 Interrupt Request Enable Low Bit
 U1RENL—UART 1 Receive Interrupt Request Enable Low Bit
 U1TENL—UART 1 Transmit Interrupt Request Enable Low Bit
 DMAENL—DMA Interrupt Request Enable Low Bit
 C3ENL—Port C3 Interrupt Request Enable Low Bit
 C2ENL—Port C2 Interrupt Request Enable Low Bit
 C1ENL—Port C1 Interrupt Request Enable Low Bit
 C0ENL—Port C0 Interrupt Request Enable Low Bit

Interrupt Edge Select Register

The Interrupt Edge Select (IRQES) register (Table 35) determines whether an interrupt is generated for the rising edge or falling edge on the selected GPIO Port input pin. The



- Configure the timer for Gated mode.
 - Set the prescale value.
2. Write to the Timer High and Low Byte registers to set the starting count value. This only affects the first pass in Gated mode. After the first timer reset in Gated mode, counting always begins at the reset value of 0001H.
 3. Write to the Timer Reload High and Low Byte registers to set the Reload value.
 4. If desired, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
 5. Configure the associated GPIO port pin for the Timer Input alternate function.
 6. Write to the Timer Control register to enable the timer.
 7. Assert the Timer Input signal to initiate the counting.

Capture/Compare Mode

In Capture/Compare mode, the timer begins counting on the *first* external Timer Input transition. The desired transition (rising edge or falling edge) is set by the TPOL bit in the Timer Control Register. The timer input is the system clock.

Every subsequent desired transition (after the first) of the Timer Input signal captures the current count value. The Capture value is written to the Timer PWM High and Low Byte Registers. When the Capture event occurs, an interrupt is generated, the count value in the Timer High and Low Byte registers is reset to 0001H, and counting resumes.

If no Capture event occurs, the timer counts up to the 16-bit Compare value stored in the Timer Reload High and Low Byte registers. Upon reaching the Compare value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes.

The steps for configuring a timer for Capture/Compare mode and initiating the count are as follows:

1. Write to the Timer Control register to:
 - Disable the timer
 - Configure the timer for Capture/Compare mode.
 - Set the prescale value.
 - Set the Capture edge (rising or falling) for the Timer Input.
2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H).
3. Write to the Timer Reload High and Low Byte registers to set the Compare value.
4. If desired, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.



5. 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 6. 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.
6. Write the data byte to the UART Transmit Data register. The transmitter automatically transfers the data to the Transmit Shift register and transmit the data.
7. To transmit additional bits, 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 these steps to configure the UART for interrupt-driven data transmission:

1. Write to the UART Baud Rate High and Low Byte registers to set the desired 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 desired priority.
5. Write to the UART Control 1 register to enable Multiprocessor (9-bit) mode functions, if desired.
6. Write to the UART Control 0 register to:
 - Set the transmit enable bit (TEN) to enable the UART for data transmission
 - Enable parity, if desired, and select either even or odd parity.
 - Set or clear the CTSE bit to enable or disable control from the receiver via the CTS pin.
7. Execute an EI instruction to enable interrupts.

The UART is now configured for interrupt-driven data transmission. When the UART Transmit interrupt is detected, the associated interrupt service routine (ISR) should perform the following:

8. Write the data byte to the UART Transmit Data register. The transmitter will automatically transfer the data to the Transmit Shift register and transmit the data.
9. Clear the UART Transmit interrupt bit in the applicable Interrupt Request register.
10. Execute the IRET instruction to return from the interrupt-service routine and wait for the Transmit Data register to again become empty.

Table 57. UARTx Baud Rate Low Byte Register (UxBRL)

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 and F4FH							

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, the integer baud rate divisor value is calculated 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 desired 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 58 provides information on data rate errors for popular baud rates and commonly used crystal oscillator frequencies.



Infrared Encoder/Decoder

Overview

The Z8F640x family products contain two fully-functional, high-performance UART to Infrared Encoder/Decoders (Endecs). Each Infrared Endec is integrated with an on-chip UART to allow easy communication between the Z8F640x family device and IrDA Physical Layer Specification Version 1.3-compliant infrared transceivers. Infrared communication provides secure, reliable, low-cost, point-to-point communication between PCs, PDAs, cell phones, printers and other infrared enabled devices.

Architecture

Figure 71 illustrates the architecture of the Infrared Endec.

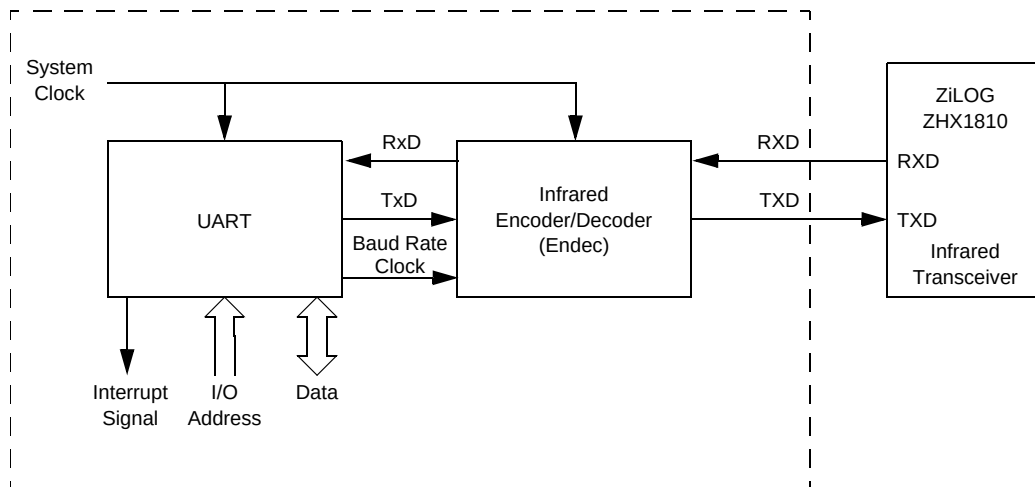


Figure 71. Infrared Data Communication System Block Diagram

I²C Baud Rate High and Low Byte Registers

The I²C Baud Rate High and Low Byte registers combine to form a 16-bit reload value, BRG[15:0], for the I²C Baud Rate Generator. The I²C baud rate is calculated using the following equation:

$$\text{I2C Baud Rate (bits/s)} = \frac{\text{System Clock Frequency (Hz)}}{4 \times \text{BRG}[15:0]}$$

Table 69. I²C Baud Rate High Byte Register (I2CBRH)

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	F53H							

BRH = I²C Baud Rate High Byte
Most significant byte, BRG[15:8], of the I²C Baud Rate Generator's reload value.

Table 70. I²C Baud Rate Low Byte Register (I2CBRL)

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	F54H							

BRL = I²C Baud Rate Low Byte
Least significant byte, BRG[7:0], of the I²C Baud Rate Generator's reload value.

Table 76 provides an example of the Register File addresses if the DMA_ADC Address register contains the value 72H.

Table 76. DMA_ADC Register File Address Example

ADC Analog Input	Register File Address (Hex) ¹
0	720H-721H
1	722H-723H
2	724H-725H
3	726H-727H
4	728H-729H
5	72AH-72BH
6	72CH-72DH
7	72EH-72FH
8	730H-731H
9	732H-733H
10	734H-735H
11	736H-737H

¹ DMAA_ADDR set to 72H.

Table 77. DMA_ADC Address Register (DMAA_ADDR)

BITS	7	6	5	4	3	2	1	0
FIELD	DMAA_ADDR							Reserved
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	FBDH							

DMAA_ADDR—DMA_ADC Address

These bits specify the seven most-significant bits of the 12-bit Register File addresses used for storing the ADC output data. The ADC Analog Input Number defines the five least-significant bits of the Register File address. Full 12-bit address is {DMAA_ADDR[7:1], 4-bit ADC Analog Input Number, 0}.

Reserved

This bit is reserved and must be 0.

Flash Frequency High and Low Byte Registers

The Flash Frequency High and Low Byte registers combine to form a 16-bit value, FFREQ, to control timing for Flash program and erase operations. The 16-bit binary Flash Frequency value must contain the system clock frequency (in kHz) and is calculated using the following equation:

$$\text{FFREQ}[15:0] = \{\text{FFREQH}[7:0], \text{FFREQL}[7:0]\} = \frac{\text{System Clock Frequency}}{1000}$$



Caution:

Flash programming and erasure is not supported for system clock frequencies below 32KHz (32768Hz) or above 20MHz. The Flash Frequency High and Low Byte registers must be loaded with the correct value to insure proper operation of the Z8F640x family device.

Table 88. Flash Frequency High Byte Register (FFREQH)

BITS	7	6	5	4	3	2	1	0
FIELD	FFREQH							
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	FFAH							

FFREQH—Flash Frequency High Byte
High byte of the 16-bit Flash Frequency value.

Table 89. Flash Frequency Low Byte Register (FFREQL)

BITS	7	6	5	4	3	2	1	0
FIELD	FFREQL							
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	FFBH							

FFREQL—Flash Frequency Low Byte
Low byte of the 16-bit Flash Frequency value.

**Table 100. Absolute Maximum Ratings**

Parameter	Minimum	Maximum	Units	Notes
68-Pin PLCC Maximum Ratings at 70°C to 105°C				
Total power dissipation		500	mW	
Maximum current into V_{DD} or out of V_{SS}		140	mA	
64-Pin LQFP Maximum Ratings at -40°C to 70°C				
Total power dissipation		1000	mW	
Maximum current into V_{DD} or out of V_{SS}		275	mA	
64-Pin LQFP Maximum Ratings at 70°C to 105°C				
Total power dissipation		540	mW	
Maximum current into V_{DD} or out of V_{SS}		150	mA	
44-Pin PLCC Maximum Ratings at -40°C to 70°C				
Total power dissipation		750	mW	
Maximum current into V_{DD} or out of V_{SS}		200	mA	
44-Pin PLCC Maximum Ratings at 70°C to 105°C				
Total power dissipation		295	mW	
Maximum current into V_{DD} or out of V_{SS}		83	mA	
44-pin LQFP Maximum Ratings at -40°C to 70°C				
Total power dissipation		750	mW	
Maximum current into V_{DD} or out of V_{SS}		200	mA	
44-pin LQFP Maximum Ratings at 70°C to 105°C				
Total power dissipation		410	mW	
Maximum current into V_{DD} or out of V_{SS}		114	mA	
40-Pin PDIP Maximum Ratings at -40°C to 70°C				
Total power dissipation		1000	mW	
Maximum current into V_{DD} or out of V_{SS}		275	mA	
40-Pin PDIP Maximum Ratings at 70°C to 105°C				
Total power dissipation		540	mW	
Maximum current into V_{DD} or out of V_{SS}		150	mA	

Notes:

1. This voltage applies to all pins except the following: V_{DD} , AV_{DD} , pins supporting analog input (Port B and Port H), RESET, and where noted otherwise.

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

Symbol	Parameter	V _{DD} = 3.0 - 3.6V T _A = -40°C to 105°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	
Z _{in}	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.						

On-Chip Debugger Timing

Figure 95 and Table 109 provide timing information for DBG pins. The timing specifications presume a rise and fall time on DBG of less than 4μs.

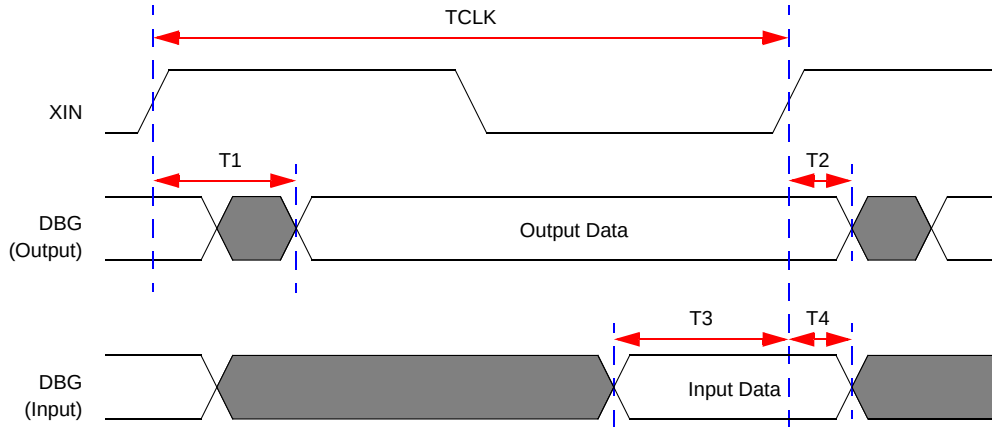


Figure 95. On-Chip Debugger Timing

Table 109. On-Chip Debugger Timing

Parameter	Abbreviation	Delay (ns)	
		Minimum	Maximum
DBG			
T ₁	XIN Rise to DBG Valid Delay	–	15
T ₂	XIN Rise to DBG Output Hold Time	2	–
T ₃	DBG to XIN Rise Input Setup Time	10	–
T ₄	DBG to XIN Rise Input Hold Time	5	–
	DBG frequency		System Clock / 4

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.

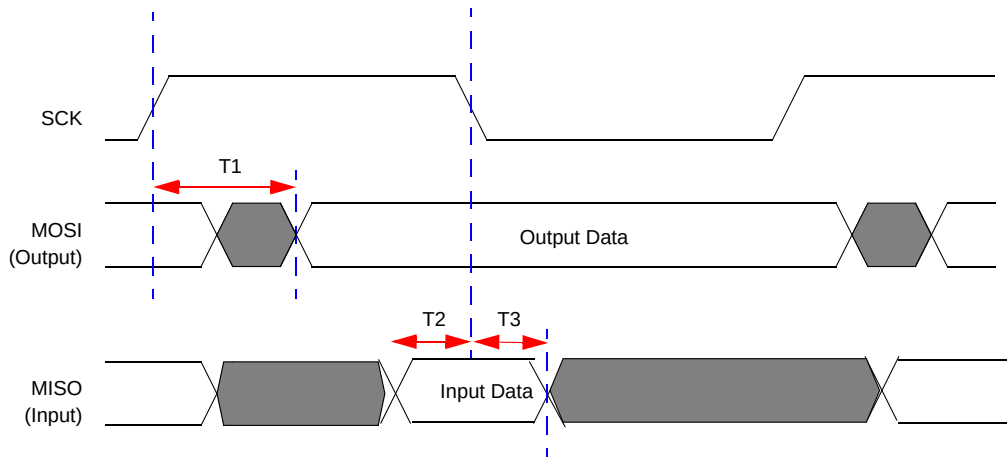


Figure 96. SPI Master Mode Timing

Table 110. SPI Master Mode Timing

Parameter	Abbreviation	Delay (ns)	
		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	

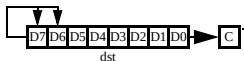


Table 126. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
BTJZ bit, src, dst	if src[bit] = 0		r	F6	-	-	-	-	-	-	3	3
	PC ← PC + X		Ir	F7							3	4
CALL dst	SP ← SP -2	IRR		D4	-	-	-	-	-	-	2	6
	@SP ← PC	DA		D6							3	3
	PC ← dst											
CCF	C ← ~C			EF	*	-	-	-	-	-	1	2
CLR dst	dst ← 00H	R		B0	-	-	-	-	-	-	2	2
		IR		B1							2	3
COM dst	dst ← ~dst	R		60	-	*	*	0	-	-	2	2
		IR		61							2	3
CP dst, src	dst - src	r	r	A2	*	*	*	*	-	-	2	3
		r	Ir	A3							2	4
		R	R	A4							3	3
		R	IR	A5							3	4
		R	IM	A6							3	3
		IR	IM	A7							3	4
CPC dst, src	dst - src - C	r	r	1F A2	*	*	*	*	-	-	3	3
		r	Ir	1F A3							3	4
		R	R	1F A4							4	3
		R	IR	1F A5							4	4
		R	IM	1F A6							4	3
		IR	IM	1F A7							4	4
CPCX dst, src	dst - src - C	ER	ER	1F A8	*	*	*	*	-	-	5	3
		ER	IM	1F A9							5	3
CPX dst, src	dst - src	ER	ER	A8	*	*	*	*	-	-	4	3
		ER	IM	A9							4	3
Flags Notation:	* = Value is a function of the result of the operation.				0 = Reset to 0							
	- = Unaffected				1 = Set to 1							
	X = Undefined											



Table 126. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
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
SCF	C ← 1			DF	1	-	-	-	-	-	1	2
SRA dst		R		D0	*	*	*	0	-	-	2	2
		IR		D1							2	3
SRL dst		R		1F C0	*	*	0	*	-	-	3	2
		IR		1F C1							3	3
SRP src	RP ← src		IM	01	-	-	-	-	-	-	2	2
STOP	Stop Mode			6F	-	-	-	-	-	-	1	2
SUB dst, src	dst ← dst − src	r	r	22	*	*	*	*	1	*	2	3
		r	Ir	23							2	4
		R	R	24							3	3
		R	IR	25							3	4
		R	IM	26							3	3
		IR	IM	27							3	4
SUBX dst, src	dst ← dst − src	ER	ER	28	*	*	*	*	1	*	4	3
		ER	IM	29							4	3
SWAP dst	dst[7:4] ↔ dst[3:0]	R		F0	X	*	*	X	-	-	2	2
		IR		F1							2	3
Flags Notation:	* = Value is a function of the result of the operation.				0 = Reset to 0							
	- = Unaffected				1 = Set to 1							
	X = Undefined											



- extended addressing register 184
- external pin reset 29
- eZ8 CPU features 3
- eZ8 CPU instruction classes 187
- eZ8 CPU instruction notation 183
- eZ8 CPU instruction set 182
- eZ8 CPU instruction summary 191

F

- FCTL register 144
- features, Z8 Encore!® 1
- first opcode map 204
- FLAGS 185
- flags register 185
- flash
 - controller 4
 - option bit address space 148
 - option bit configuration - reset 148
 - program memory address 0000H 149
 - program memory address 0001H 150
- flash memory 138
 - arrangement 139
 - byte programming 142
 - code protection 141
 - configurations 138
 - control register definitions 144
 - controller bypass 143
 - electrical characteristics and timing 173
 - flash control register 144
 - flash option bits 142
 - flash status register 145
 - flow chart 140
 - frequency high and low byte registers 147
 - mass erase 143
 - operation 139
 - operation timing 141
 - page erase 143
 - page select register 146
- FPS register 146
- FSTAT register 145

G

- gated mode 71
- general-purpose I/O 33
- GPIO 4, 33
 - alternate functions 34
 - architecture 34
 - control register definitions 36
 - input data sample timing 176
 - interrupts 36
 - port A-H address registers 37
 - port A-H alternate function sub-registers 39
 - port A-H control registers 38
 - port A-H data direction sub-registers 39
 - port A-H high drive enable sub-registers 41
 - port A-H input data registers 42
 - port A-H output control sub-registers 40
 - port A-H output data registers 43
 - port A-H stop mode recovery sub-registers 41
 - port availability by device 33
 - port input timing 176
 - port output timing 177

H

- H 185
- HALT 189
- HALT mode 31, 189
- hexadecimal number prefix/suffix 185

I

- I²C 4
 - 10-bit address read transaction 116
 - 10-bit address transaction 114
 - 10-bit addressed slave data transfer format 114
 - 10-bit receive data format 116
 - 7-bit address transaction 112
 - 7-bit address, reading a transaction 115
 - 7-bit addressed slave data transfer format 113
 - 7-bit receive data transfer format 115
 - baud high and low byte registers 121
 - C status register 118
 - control register definitions 118



controller 111	CALL 190
controller signals 13	CCF 188, 189
interrupts 112	CLR 189
operation 111	COM 190
SDA and SCL signals 111	CP 187
stop and start conditions 112	CPC 187
I2CBRH register 121	CPCX 187
I2CBRL register 121	CPU control 189
I2CCTL register 119	CPX 187
I2CDATA register 118	DA 187
I2CSTAT register 118	DEC 187
IM 184	DECW 187
immediate data 184	DI 189
immediate operand prefix 185	DJNZ 190
INC 187	EI 189
increment 187	HALT 189
increment word 187	INC 187
INCW 187	INCW 187
indexed 184	IRET 190
indirect address prefix 185	JP 190
indirect register 184	LD 189
indirect register pair 184	LDC 189
indirect working register 184	LDCI 188, 189
indirect working register pair 184	LDE 189
infrared encoder/decoder (IrDA) 95	LDEI 188
instruction set, ez8 CPU 182	LDX 189
instructions	LEA 189
ADC 187	load 189
ADCX 187	logical 190
ADD 187	MULT 187
ADDX 187	NOP 189
AND 190	OR 190
ANDX 190	ORX 190
arithmetic 187	POP 189
BCLR 188	POPX 189
BIT 188	program control 190
bit manipulation 188	PUSH 189
block transfer 188	PUSHX 189
BRK 190	RCF 188, 189
BSET 188	RET 190
BSWAP 188, 191	RL 191
BTJ 190	RLC 191
BTJNZ 190	rotate and shift 191
BTJZ 190	RR 191