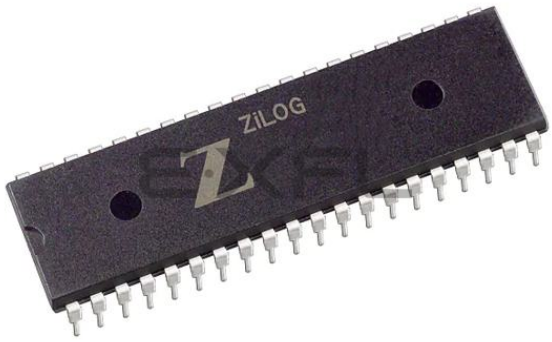


Welcome to E-XFL.COM

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	29
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Through Hole
Package / Case	40-DIP (0.620", 15.75mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f4821pm020ec

Revision History

Each instance in the Revision History reflects a change to this document from its previous revision. For more details, refer to the corresponding pages or appropriate links given in the table below.

Date	Revision Level	Description	Page No
December 2007	19	Updated Zilog logo, Disclaimer section, and implemented style guide. Updated Table 112. Changed Z8 Encore! 64K Series to Z8 Encore! XP 64K Series Flash Microcontrollers throughout the document.	All
December 2006	18	Updated Table 110 and Ordering Information.	228, 270
November 2006	17	Updated Part Number Suffix Designations.	275
June 2006	16	Updated Timer 0-3 Control 1 Registers.	94
October 2005	15	The paragraph tag for Ordering Information has been changed from H1 Heading to Chapter Title.	270

Table 7. Z8 Encore! XP 64K Series Flash Microcontrollers Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page No
Timer 3 (unavailable in the 44-pin packages)				
F18	Timer 3 High Byte	T3H	00	90
F19	Timer 3 Low Byte	T3L	01	90
F1A	Timer 3 Reload High Byte	T3RH	FF	91
F1B	Timer 3 Reload Low Byte	T3RL	FF	91
F1C	Timer 3 PWM High Byte	T3PWMH	00	92
F1D	Timer 3 PWM Low Byte	T3PWML	00	92
F1E	Timer 3 Control 0	T3CTL0	00	93
F1F	Timer 3 Control 1	T3CTL1	00	94
20-3F	Reserved	—	XX	
UART 0				
F40	UART0 Transmit Data	U0TXD	XX	114
	UART0 Receive Data	U0RXD	XX	115
F41	UART0 Status 0	U0STAT0	0000011Xb	115
F42	UART0 Control 0	U0CTL0	00	117
F43	UART0 Control 1	U0CTL1	00	117
F44	UART0 Status 1	U0STAT1	00	115
F45	UART0 Address Compare Register	U0ADDR	00	120
F46	UART0 Baud Rate High Byte	U0BRH	FF	120
F47	UART0 Baud Rate Low Byte	U0BRL	FF	120
UART 1				
F48	UART1 Transmit Data	U1TXD	XX	114
	UART1 Receive Data	U1RXD	XX	115
F49	UART1 Status 0	U1STAT0	0000011Xb	115
F4A	UART1 Control 0	U1CTL0	00	117
F4B	UART1 Control 1	U1CTL1	00	117
F4C	UART1 Status 1	U1STAT1	00	115
F4D	UART1 Address Compare Register	U1ADDR	00	120
F4E	UART1 Baud Rate High Byte	U1BRH	FF	120
F4F	UART1 Baud Rate Low Byte	U1BRL	FF	120
I^2C				
F50	I^2C Data	I2CDATA	00	156
F51	I^2C Status	I2CSTAT	80	157
F52	I^2C Control	I2CCTL	00	158
F53	I^2C Baud Rate High Byte	I2CBRH	FF	160
F54	I^2C Baud Rate Low Byte	I2CBRL	FF	160
F55	I^2C Diagnostic State	I2CDST	C0	161
F56	I^2C Diagnostic Control	I2CDIAG	00	163
F57-F5F	Reserved	—	XX	
Serial Peripheral Interface (SPI)				
F60	SPI Data	SPIDATA	XX	137

GPIO Control Register Definitions

Four registers for each Port provide access to GPIO control, input data, and output data. Table 13 lists these Port registers. Use the Port A–H Address and Control registers together to provide access to sub-registers for Port configuration and control.

Table 13. GPIO Port Registers and Sub-Registers

Port Register Mnemonic	Port Register Name
PxADDR	Port A–H Address Register (Selects sub-registers)
PxCTL	Port A–H Control Register (Provides access to sub-registers)
PxIN	Port A–H Input Data Register
PxOUT	Port A–H Output Data Register
Port Sub-Register Mnemonic	**Port Register Name**
PxDD	Data Direction
PxAF	Alternate Function
PxOC	Output Control (Open-Drain)
PxDD	High Drive Enable
PxSMRE	Stop Mode Recovery Source Enable

Port A–H Address Registers

The Port A–H Address registers select the GPIO Port functionality accessible through the Port A–H Control registers. The Port A–H Address and Control registers combine to provide access to all GPIO Port control (Table 14).

Table 14. Port A–H GPIO Address Registers (PxADDR)

BITS	7	6	5	4	3	2	1	0
FIELD	PADDR[7:0]							
RESET	00H							
R/W	R/W							
ADDR	FD0H, FD4H, FD8H, FDCH, FE0H, FE4H, FE8H, FECH							

Interrupt Controller

Overview

The interrupt controller on the 64K Series products prioritizes the interrupt requests from the on-chip peripherals and the GPIO port pins. The features of the interrupt controller include the following:

- 24 unique interrupt vectors:
  - 12 GPIO port pin interrupt sources
  - 12 on-chip peripheral interrupt sources
- Flexible GPIO interrupts
  - Eight selectable rising and falling edge GPIO interrupts
  - Four dual-edge interrupts
- Three levels of individually programmable interrupt priority
- Watchdog Timer can be configured to generate an interrupt

Interrupt requests (IRQs) allow peripheral devices to suspend CPU operation in an orderly manner and force the CPU to start an interrupt service routine (ISR). Usually this interrupt service routine is involved with the exchange of data, status information, or control information between the CPU and the interrupting peripheral. When the service routine is completed, the CPU returns to the operation from which it was interrupted.

The eZ8 CPU supports both vectored and polled interrupt handling. For polled interrupts, the interrupt control has no effect on operation. For more information on interrupt servicing by the eZ8 CPU, refer to *eZ8™ CPU Core User Manual (UM0128)* available for download at www.zilog.com.

Interrupt Vector Listing

Table 23 lists all of the interrupts available in order of priority. The interrupt vector is stored with the most-significant byte (MSB) at the even Program Memory address and the least-significant byte (LSB) at the following odd Program Memory address.

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 36) determines whether an interrupt is generated for the rising edge or falling edge on the selected GPIO Port input pin. The Interrupt Port Select register selects between Port A and Port D for the individual interrupts.

Table 36. Interrupt Edge Select Register (IRQES)

BITS	7	6	5	4	3	2	1	0
FIELD	IES7	IES6	IES5	IES4	IES3	IES2	IES1	IES0
RESET	0							
R/W	R/W							
ADDR	FCDH							

IES*x*—Interrupt Edge Select *x*
The minimum pulse width should be greater than 1 system clock to guarantee capture of the edge triggered interrupt. Shorter pulses may be captured but not guaranteed.
0 = An interrupt request is generated on the falling edge of the PA*x*/PD*x* input.
1 = An interrupt request is generated on the rising edge of the PA*x*/PD*x* input.
where *x* indicates the specific GPIO Port pin number (0 through 7).

Interrupt Port Select Register

The Port Select (IRQPS) register (Table 37) determines the port pin that generates the PAx/PDx interrupts. This register allows either Port A or Port D pins to be used as interrupts. The Interrupt Edge Select register controls the active interrupt edge.

Table 37. Interrupt Port Select Register (IRQPS)

BITS	7	6	5	4	3	2	1	0
FIELD	PAD7S	PAD6S	PAD5S	PAD4S	PAD3S	PAD2S	PAD1S	PAD0S
RESET	0							
R/W	R/W							
ADDR	FCEH							

Timers

Overview

The 64K Series products contain up to four 16-bit reloadable timers that can be used for timing, event counting, or generation of pulse width modulated signals. The timers' features include:

- 16-bit reload counter
- Programmable prescaler with prescale values from 1 to 128
- PWM output generation
- Capture and compare capability
- External input pin for timer input, clock gating, or capture signal. External input pin signal frequency is limited to a maximum of one-fourth the system clock frequency.
- Timer output pin
- Timer interrupt

In addition to the timers described in this chapter, the Baud Rate Generators for any unused UART, SPI, or I^2C peripherals may also be used to provide basic timing functionality. For information on using the Baud Rate Generators as timers, see the respective serial communication peripheral. Timer 3 is unavailable in the 44-pin package devices.

Architecture

Figure 12 displays the architecture of the timers.

Table 42. Timer 0-3 Reload Low Byte Register (TxRL)

BITS	7	6	5	4	3	2	1	0
FIELD	TRL							
RESET	1							
R/W	R/W							
ADDR	F03H, F0BH, F13H, F1BH							

TRH and TRL—Timer Reload Register High and Low
These two bytes form the 16-bit Reload value, {TRH[7:0], TRL[7:0]}. This value sets the maximum count value which initiates a timer reload to 0001H. In COMPARE mode, these two byte form the 16-bit Compare value.

Timer 0-3 PWM High and Low Byte Registers

The Timer 0-3 PWM High and Low Byte (TxPWMH and TxPWML) registers (see Table 43 and Table 44 on page 92) are used for Pulse-Width Modulator (PWM) operations. These registers also store the Capture values for the Capture and Capture/COMPARE modes.

Table 43. Timer 0-3 PWM High Byte Register (TxPWMH)

BITS	7	6	5	4	3	2	1	0
FIELD	PWMH							
RESET	0							
R/W	R/W							
ADDR	F04H, F0CH, F14H, F1CH							

Table 44. Timer 0-3 PWM Low Byte Register (TxPWML)

BITS	7	6	5	4	3	2	1	0
FIELD	PWML							
RESET	0							
R/W	R/W							
ADDR	F05H, F0DH, F15H, F1DH							

If interrupts are enabled, following completion of the Stop Mode Recovery the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executing code from the vector address.

WDT Reset in Normal Operation

If configured to generate a Reset when a time-out occurs, the Watchdog Timer forces the device into the Reset state. The WDT status bit in the Watchdog Timer Control register is set to 1. For more information on Reset, see Reset and Stop Mode Recovery on page 47.

WDT Reset in STOP Mode

If enabled in STOP mode and configured to generate a Reset when a time-out occurs and the device is in STOP mode, the Watchdog Timer initiates a Stop Mode Recovery. Both the WDT status bit and the STOP bit in the Watchdog Timer Control register are set to 1 following WDT time-out in STOP mode. Default operation is for the WDT and its RC oscillator to be enabled during STOP mode.

WDT RC Disable in STOP Mode

To minimize power consumption in STOP Mode, the WDT and its RC oscillator can be disabled in STOP mode. The following sequence configures the WDT to be disabled when the 64K Series devices enter STOP Mode following execution of a STOP instruction:

1. Write 55H to the Watchdog Timer Control register (WDTCTL).
2. Write AAH to the Watchdog Timer Control register (WDTCTL).
3. Write 81H to the Watchdog Timer Control register (WDTCTL) to configure the WDT and its oscillator to be disabled during STOP Mode. Alternatively, write 00H to the Watchdog Timer Control register (WDTCTL) as the third step in this sequence to reconfigure the WDT and its oscillator to be enabled during STOP mode.

This sequence only affects WDT operation in STOP mode.

Watchdog Timer Reload Unlock Sequence

Writing the unlock sequence to the Watchdog Timer (WDTCTL) Control register address unlocks the three Watchdog Timer Reload Byte registers (WDTU, WDTH, and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL register address produce no effect on the bits in the WDTCTL register. The locking mechanism prevents spurious writes to the Reload registers. Follow the steps below to unlock the Watchdog Timer Reload Byte registers (WDTU, WDTH, and WDTL) for write access.

1. Write 55H to the Watchdog Timer Control register (WDTCTL).
2. Write AAH to the Watchdog Timer Control register (WDTCTL).
3. Write the Watchdog Timer Reload Upper Byte register (WDTU).
4. Write the Watchdog Timer Reload High Byte register (WDTH).

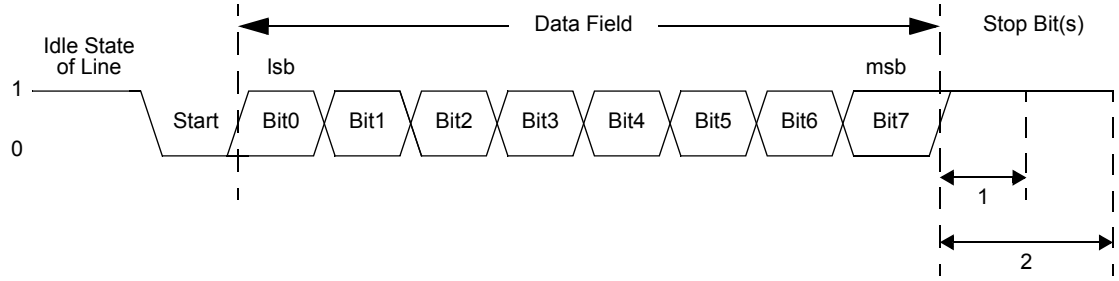


Figure 14. UART Asynchronous Data Format without Parity

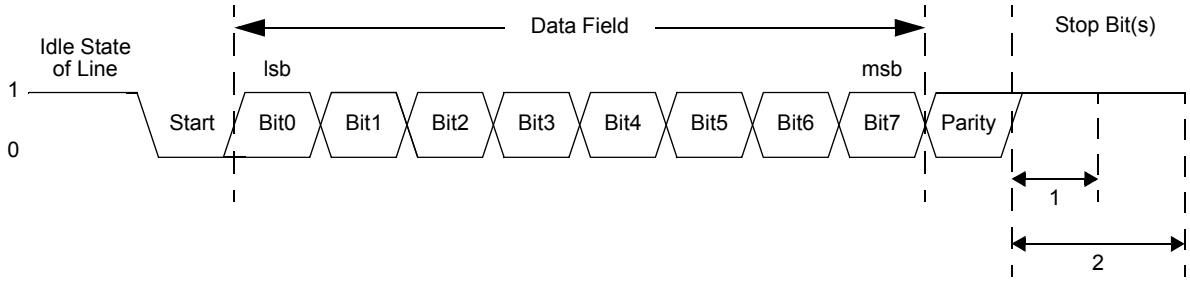


Figure 15. UART Asynchronous Data Format with Parity

Transmitting Data using the Polled Method

Follow the steps below to transmit data using the polled method of operation:

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. If MULTIPROCESSOR mode is desired, write to the UART Control 1 register to enable MULTIPROCESSOR (9-bit) mode functions.
   - Set the MULTIPROCESSOR Mode Select (MPEN) to Enable MULTIPROCESSOR mode.
4. Write to the UART Control 0 register to:
   - Set the transmit enable bit (TEN) to enable the UART for data transmission
   - If parity is desired and MULTIPROCESSOR mode is not enabled, set the parity enable bit (PEN) and select either Even or Odd parity (PSEL).

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

Clear To Send ($\overline{CTS}$) 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 ($\overline{CTS}$) 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 $\overline{CTS}$ at least one system clock cycle before a new data transmission begins. For multiple character transmissions, this would typically be done during Stop Bit transmission. If $\overline{CTS}$ 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 16. The character format is:

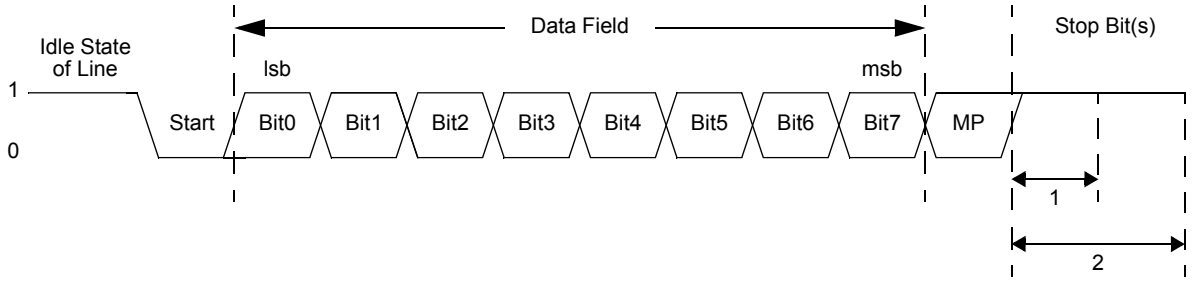


Figure 16. UART Asynchronous MULTIPROCESSOR Mode Data Format

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

Table 61. UART Baud Rates

20.0 MHz System Clock

Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	1	1250.0	0.00
625.0	2	625.0	0.00
250.0	5	250.0	0.00
115.2	11	113.6	-1.36
57.6	22	56.8	-1.36
38.4	33	37.9	-1.36
19.2	65	19.2	0.16
9.60	130	9.62	0.16
4.80	260	4.81	0.16
2.40	521	2.40	-0.03
1.20	1042	1.20	-0.03
0.60	2083	0.60	0.02
0.30	4167	0.30	-0.01

18.432 MHz System Clock

Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	1	1152.0	-7.84%
625.0	2	576.0	-7.84%
250.0	5	230.4	-7.84%
115.2	10	115.2	0.00
57.6	20	57.6	0.00
38.4	30	38.4	0.00
19.2	60	19.2	0.00
9.60	120	9.60	0.00
4.80	240	4.80	0.00
2.40	480	2.40	0.00
1.20	960	1.20	0.00
0.60	1920	0.60	0.00
0.30	3840	0.30	0.00

16.667 MHz System Clock

Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	1	1041.69	-16.67
625.0	2	520.8	-16.67
250.0	4	260.4	4.17
115.2	9	115.7	0.47
57.6	18	57.87	0.47
38.4	27	38.6	0.47
19.2	54	19.3	0.47
9.60	109	9.56	-0.45
4.80	217	4.80	-0.83
2.40	434	2.40	0.01

11.0592 MHz System Clock

Desired Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	N/A	N/A	N/A
625.0	1	691.2	10.59
250.0	3	230.4	-7.84
115.2	6	115.2	0.00
57.6	12	57.6	0.00
38.4	18	38.4	0.00
19.2	36	19.2	0.00
9.60	72	9.60	0.00
4.80	144	4.80	0.00
2.40	288	2.40	0.00

Operation

When the Infrared Endec is enabled, the transmit data from the associated on-chip UART is encoded as digital signals in accordance with the IrDA standard and output to the infrared transceiver via the TXD pin. Likewise, data received from the infrared transceiver is passed to the Infrared Endec via the RXD pin, decoded by the Infrared Endec, and then passed to the UART. Communication is half-duplex, which means simultaneous data transmission and reception is not allowed.

The baud rate is set by the UART's Baud Rate Generator and supports IrDA standard baud rates from 9600 baud to 115.2 Kbaud. Higher baud rates are possible, but do not meet IrDA specifications. The UART must be enabled to use the Infrared Endec. The Infrared Endec data rate is calculated using the following equation:

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

Transmitting IrDA Data

The data to be transmitted using the infrared transceiver is first sent to the UART. The UART's transmit signal (TXD) and baud rate clock are used by the IrDA to generate the modulation signal (IR_TXD) that drives the infrared transceiver. Each UART/Infrared data bit is 16-clock wide. If the data to be transmitted is 1, the IR_TXD signal remains low for the full 16-clock period. If the data to be transmitted is 0, a 3-clock high pulse is output following a 7-clock low period. After the 3-clock high pulse, a 6-clock low pulse is output to complete the full 16-clock data period. Figure 20 displays IrDA data transmission. When the Infrared Endec is enabled, the UART's TXD signal is internal to the 64K Series products while the IR_TXD signal is output through the TXD pin.

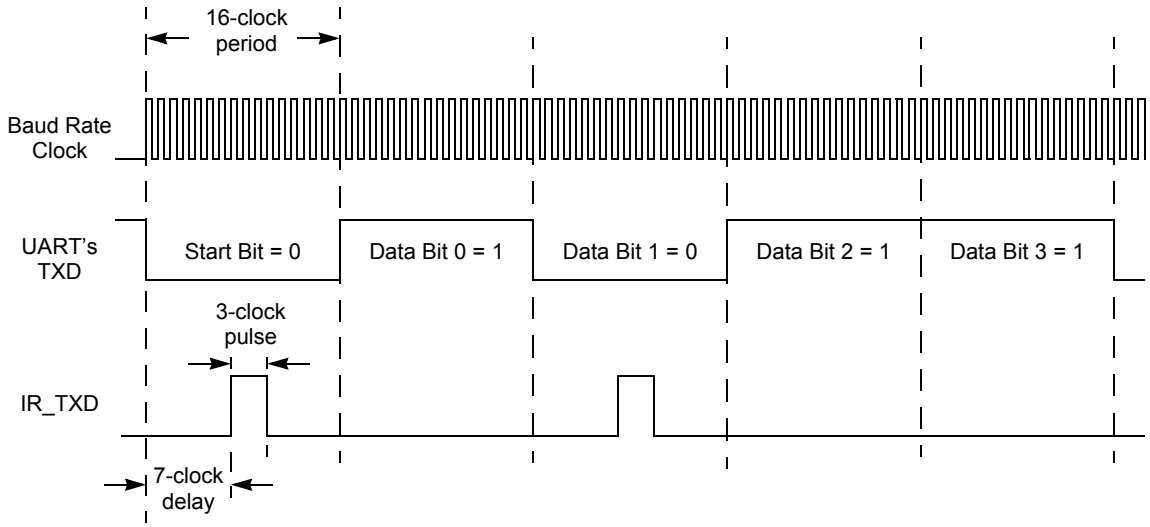


Figure 20. Infrared Data Transmission

SPI Control Register Definitions

SPI Data Register

The SPI Data register (Table 63) stores both the outgoing (transmit) data and the incoming (receive) data. Reads from the SPI Data register always return the current contents of the 8-bit shift register. Data is shifted out starting with bit 7. The last bit received resides in bit position 0.

With the SPI configured as a Master, writing a data byte to this register initiates the data transmission. With the SPI configured as a Slave, writing a data byte to this register loads the shift register in preparation for the next data transfer with the external Master. In either the Master or Slave modes, if a transmission is already in progress, writes to this register are ignored and the Overrun error Flag, OVR, is set in the SPI Status register.

When the character length is less than 8 bits (as set by the NUMBITS field in the SPI Mode register), the transmit character must be left justified in the SPI Data register. A received character of less than 8 bits is right justified (last bit received is in bit position 0). For example, if the SPI is configured for 4-bit characters, the transmit characters must be written to SPIDATA[7:4] and the received characters are read from SPIDATA[3:0].

Table 63. SPI Data Register (SPIDATA)

BITS	7	6	5	4	3	2	1	0
FIELD	DATA							
RESET	X							
R/W	R/W							
ADDR	F60H							

DATA—Data
Transmit and/or receive data.

SPI Control Register

The SPI Control register (see Table 64 on page 138) configures the SPI for transmit and receive operations.

7. The I2C Controller loads the I2C Shift register with the contents of the I2C Data register.
8. After one bit of address is shifted out by the SDA signal, the Transmit interrupt is asserted.
9. Software responds by writing the second byte of address into the contents of the I2C Data register.
10. The I2C Controller shifts the rest of the first byte of address and write bit out the SDA signal.
11. If the I2C slave sends an acknowledge by pulling the SDA signal low during the next high period of SCL the I2C Controller sets the ACK bit in the I2C Status register. Continue with step 12.

    If the slave does not acknowledge the first address byte, the I2C Controller sets the NCKI bit and clears the ACK bit in the I2C Status register. Software responds to the Not Acknowledge interrupt by setting the STOP and FLUSH bits and clearing the TXI bit. The I2C Controller sends the STOP condition on the bus and clears the STOP and NCKI bits. The transaction is complete (ignore following steps).
12. The I2C Controller loads the I2C Shift register with the contents of the I2C Data register (2nd byte of address).
13. The I2C Controller shifts the second address byte out the SDA signal. After the first bit has been sent, the Transmit interrupt is asserted.
14. Software responds by setting the STOP bit in the I2C Control register. The TXI bit can be cleared at the same time.
15. Software polls the STOP bit of the I2C Control register. Hardware deasserts the STOP bit when the transaction is completed (STOP condition has been sent).
16. Software checks the ACK bit of the I2C Status register. If the slave acknowledged, the ACK bit is = 1. If the slave does not acknowledge, the ACK bit is = 0. The NCKI interrupt do not occur because the STOP bit was set.

Write Transaction with a 10-Bit Address

Figure 31 displays the data transfer format for a 10-bit addressed slave. Shaded regions indicate data transferred from the I2C Controller to slaves and unshaded regions indicate data transferred from the slaves to the I2C Controller.

S	Slave Address 1st 7 bits	W = 0	A	Slave Address 2nd Byte	A	Data	A	Data	$\overline{A}$/A	P/S

Figure 31. 10-Bit Addressed Slave Data Transfer Format

0101 = ADC Analog Inputs 0-5 updated.
0110 = ADC Analog Inputs 0-6 updated.
0111 = ADC Analog Inputs 0-7 updated.
1000 = ADC Analog Inputs 0-8 updated.
1001 = ADC Analog Inputs 0-9 updated.
1010 = ADC Analog Inputs 0-10 updated.
1011 = ADC Analog Inputs 0-11 updated.
1100-1111 = Reserved.

DMA Status Register

The DMA Status register (Table 85 on page 173) indicates the DMA channel that generated the interrupt and the ADC Analog Input that is currently undergoing conversion. Reads from this register reset the Interrupt Request Indicator bits (IRQA, IRQ1, and IRQ0) to 0. Therefore, software interrupt service routines that read this register must process all three interrupt sources from the DMA.

Table 85. DMA_ADC Status Register (DMAA_STAT)

BITS	7	6	5	4	3	2	1	0
FIELD	CADC[3:0]				Reserved	IRQA	IRQ1	IRQ0
RESET	0							
R/W	R							
ADDR	FBFH							

CADC[3:0]—Current ADC Analog Input
This field identifies the Analog Input that the ADC is currently converting.

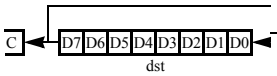
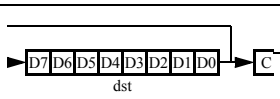
Reserved
This bit is reserved and must be 0.

IRQA—DMA_ADC Interrupt Request Indicator
This bit is automatically reset to 0 each time a read from this register occurs.
0 = DMA_ADC is not the source of the interrupt from the DMA Controller.
1 = DMA_ADC completed transfer of data from the last ADC Analog Input and generated an interrupt.

IRQ1—DMA1 Interrupt Request Indicator
This bit is automatically reset to 0 each time a read from this register occurs.
0 = DMA1 is not the source of the interrupt from the DMA Controller.
1 = DMA1 completed transfer of data to/from the End Address and generated an interrupt.

IRQ0—DMA0 Interrupt Request Indicator
This bit is automatically reset to 0 each time a read from this register occurs.

Table 133. 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
OR dst, src	dst ← dst OR src	r	r	42	-	*	*	0	-	-	2	3
		r	Ir	43							2	4
		R	R	44							3	3
		R	IR	45							3	4
		R	IM	46							3	3
		IR	IM	47							3	4
ORX dst, src	dst ← dst OR src	ER	ER	48	-	*	*	0	-	-	4	3
		ER	IM	49							4	3
POP dst	dst ← @SP SP ← SP + 1	R		50	-	-	-	-	-	-	2	2
		IR		51							2	3
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
		IM		1F 70							3	2
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

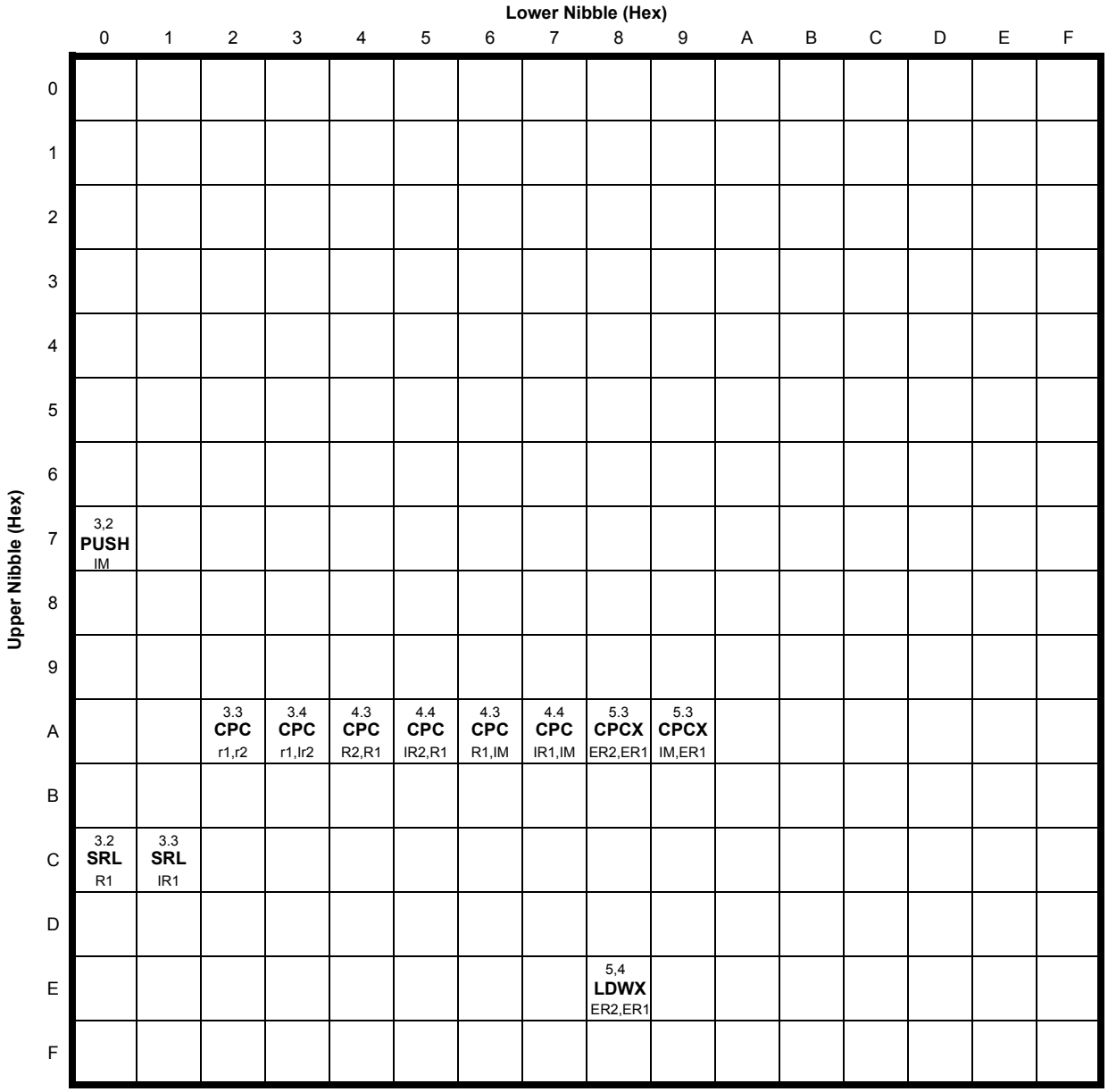
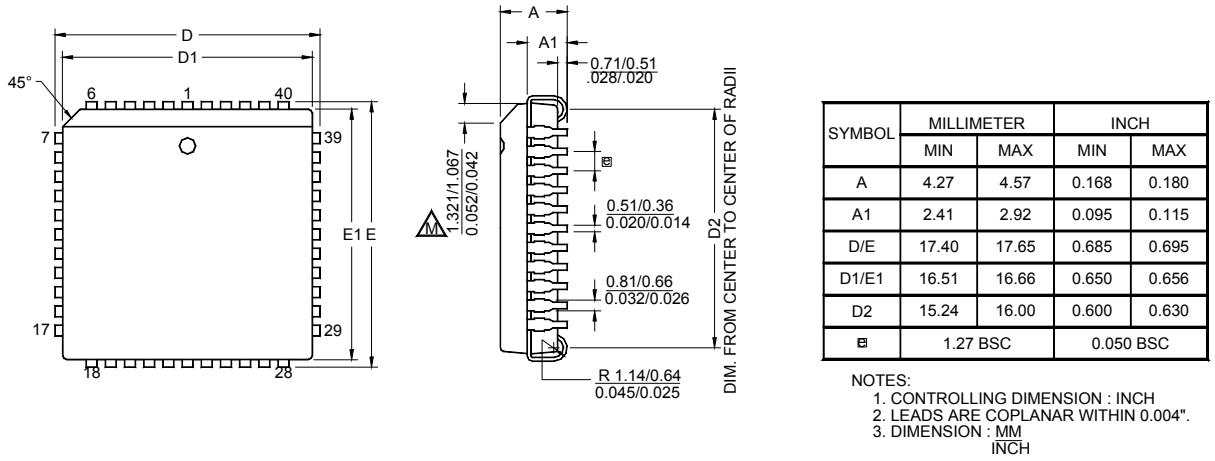


Figure 61. Second Opcode Map after 1FH

Figure 64 displays the 44-pin Plastic Lead Chip Carrier (PLCC) package available for the Z8X1621, Z8X2421, Z8X3221, Z8X4821, and Z8X6421 devices.

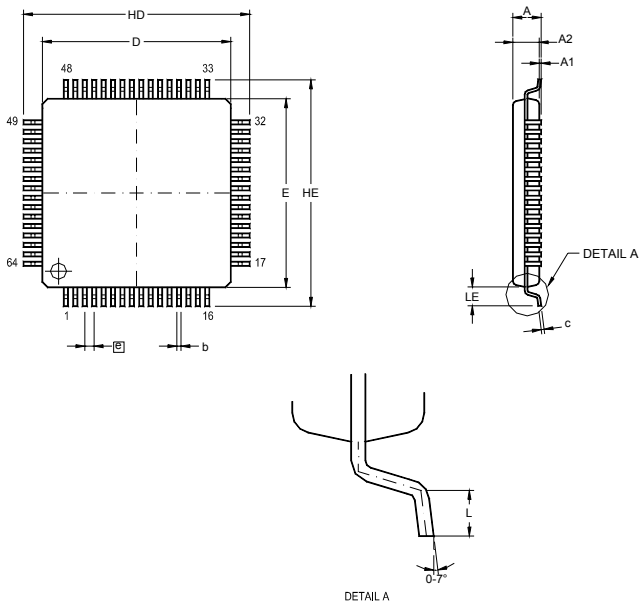


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.27	4.57	0.168	0.180
A1	2.41	2.92	0.095	0.115
D/E	17.40	17.65	0.685	0.695
D1/E1	16.51	16.66	0.650	0.656
D2	15.24	16.00	0.600	0.630
e	1.27 BSC		0.050 BSC	

NOTES:
1. CONTROLLING DIMENSION : INCH
2. LEADS ARE COPLANAR WITHIN 0.004".
3. DIMENSION : MM / INCH

Figure 64. 44-Lead Plastic Lead Chip Carrier Package (PLCC)

Figure 64 displays the 64-pin Low-Profile Quad Flat Package (LQFP) available for the Z8X1622, Z8X2422, Z8X3222, Z8X4822, and Z8X6422 devices.



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	1.40	1.60	0.055	0.063
A1	0.05	0.15	0.002	0.006
A2	1.35	1.45	0.053	0.057
b	0.17	0.27	0.007	0.011
c	0.09	0.20	0.004	0.008
HD	11.75	12.25	0.463	0.482
D	9.90	10.10	0.390	0.398
HE	11.75	12.25	0.463	0.482
E	9.90	10.10	0.390	0.398
e	0.50 BSC		0.0197 BSC	
L	0.45	0.75	0.018	0.030
LE	1.00 REF		0.039 REF	

1. CONTROLLING DIMENSIONS : mm
2. MAX. COPLANARITY : 0.10mm / 0.004"

Figure 65. 64-Lead Low-Profile Quad Flat Package (LQFP)

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
Z8F322x with 32 KB Flash, 10-Bit Analog-to-Digital Converter										
Standard Temperature: 0 °C to 70 °C										
Z8F3221PM020SC	32 KB	2 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F3221AN020SC	32 KB	2 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F3221VN020SC	32 KB	2 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F3222AR020SC	32 KB	2 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F3222VS020SC	32 KB	2 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Extended Temperature: –40 °C to 105 °C										
Z8F3221PM020EC	32 KB	2 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F3221AN020EC	32 KB	2 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F3221VN020EC	32 KB	2 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F3222AR020EC	32 KB	2 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F3222VS020EC	32 KB	2 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Automotive/Industrial Temperature: –40 °C to 125°C										
Z8F3221PM020AC	32 KB	2 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F3221AN020AC	32 KB	2 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F3221VN020AC	32 KB	2 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F3222AR020AC	32 KB	2 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F3222VS020AC	32 KB	2 KB	46	24	4	12	1	1	2	PLCC 68-pin package

For technical and customer support, hardware and software development tools, refer to the Zilog® website at www.zilog.com. The latest released version of ZDS can be downloaded from this website.

Part Number Suffix Designations