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	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f3222ar020sc

Table 9. Reset Sources and Resulting Reset Type

Operating Mode	Reset Source	Reset Type
NORMAL or HALT modes	Power-On Reset/Voltage Brownout	system reset
	Watchdog Timer time-out when configured for Reset	system reset
	$\overline{\text{RESET}}$ pin assertion	system reset
	On-Chip Debugger initiated Reset (OCDCTL[0] set to 1)	system reset except the On-Chip Debugger is unaffected by the reset
STOP mode	Power-On Reset/Voltage Brownout	system reset
	$\overline{\text{RESET}}$ pin assertion	system reset
	DBG pin driven Low	system reset

Power-On Reset

Each device in the 64K Series contains an internal Power-On Reset circuit. The POR circuit monitors the supply voltage and holds the device in the Reset state until the supply voltage reaches a safe operating level. After the supply voltage exceeds the POR voltage threshold (V_{POR}), the POR Counter is enabled and counts 66 cycles of the Watchdog Timer oscillator. After the POR counter times out, the XTAL Counter is enabled to count a total of 16 system clock pulses. The devices are held in the Reset state until both the POR Counter and XTAL counter have timed out. After the 64K Series devices exit the Power-On Reset state, the eZ8 CPU fetches the Reset vector. Following Power-On Reset, the POR status bit in the Watchdog Timer Control (WDTCTL) register is set to 1.

Figure 8 displays Power-On Reset operation. For the POR threshold voltage (V_{POR}), see Electrical Characteristics on page 215.

Port A–H Data Direction Sub-Registers

The Port A–H Data Direction sub-register is accessed through the Port A–H Control register by writing 01H to the Port A–H Address register (Table 16).

Table 16. Port A–H Data Direction Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
RESET	1							
R/W	R/W							
ADDR	If 01H in Port A–H Address Register, accessible through Port A–H Control Register							

DD[7:0]—Data Direction
These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction register setting.
0 = Output. Data in the Port A–H Output Data register is driven onto the port pin.
1 = Input. The port pin is sampled and the value written into the Port A–H Input Data Register. The output driver is tri-stated.

Port A–H Alternate Function Sub-Registers

The Port A–H Alternate Function sub-register (Table 17) is accessed through the Port A–H Control register by writing 02H to the Port A–H Address register. The Port A–H Alternate Function sub-registers select the alternate functions for the selected pins. To determine the alternate function associated with each port pin, see GPIO Alternate Functions on page 59.

Caution: *Do not enable alternate function for GPIO port pins which do not have an associated alternate function. Failure to follow this guideline may result in unpredictable operation.*

Table 17. Port A–H Alternate Function Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	AF7	AF6	AF5	AF4	AF3	AF2	AF1	AF0
RESET	0							
R/W	R/W							
ADDR	If 02H in Port A–H Address Register, accessible through Port A–H Control Register							

Port A–H Input Data Registers

Reading from the Port A–H Input Data registers (Table 21) returns the sampled values from the corresponding port pins. The Port A–H Input Data registers are Read-only.

Table 21. 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							
R/W	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).

Port A–H Output Data Register

The Port A–H Output Data register (Table 22) writes output data to the pins.

Table 22. Port A–H Output Data Register (P*x*OUT)

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, FDFH, FE3H, FE7H, FEBH, FEFH							

POUT[7:0]—Port Output Data
These bits contain the data to be driven out from 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 Port Output Control register bit to 1.

Receiving Data using the Interrupt-Driven Method

The UART Receiver interrupt indicates the availability of new data (as well as error conditions). Follow the steps below to configure the UART receiver for interrupt-driven 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. Execute a `DI` instruction to disable interrupts.
4. Write to the Interrupt control registers to enable the UART Receiver interrupt and set the desired priority.
5. Clear the UART Receiver interrupt in the applicable Interrupt Request register.
6. Write to the UART Control 1 Register to enable MULTIPROCESSOR (9-bit) mode functions, if desired.
   - Set the MULTIPROCESSOR Mode Select (`MPEN`) to Enable MULTIPROCESSOR mode.
   - Set the MULTIPROCESSOR Mode Bits, `MPMD[1:0],` to select the desired address matching scheme.
   - Configure the UART to interrupt on received data and errors or errors only (interrupt on errors only is unlikely to be useful for Z8 Encore! devices without a DMA block).
7. Write the device address to the Address Compare Register (automatic multiprocessor modes only).
8. Write to the UART Control 0 register to:
   - Set the receive enable bit (`REN`) to enable the UART for data reception.
   - Enable parity, if desired and if MULTIPROCESSOR mode is not enabled, and select either even or odd parity.
9. Execute an `EI` instruction to enable interrupts.

The UART is now configured for interrupt-driven data reception. When the UART Receiver interrupt is detected, the associated interrupt service routine performs the following:

1. Check the UART Status 0 register to determine the source of the interrupt - error, break, or received data.
2. If the interrupt was caused by data available, read the data from the UART Receive Data register. If operating in MULTIPROCESSOR (9-bit) mode, further actions may be required depending on the MULTIPROCESSOR Mode bits `MPMD[1:0]`.

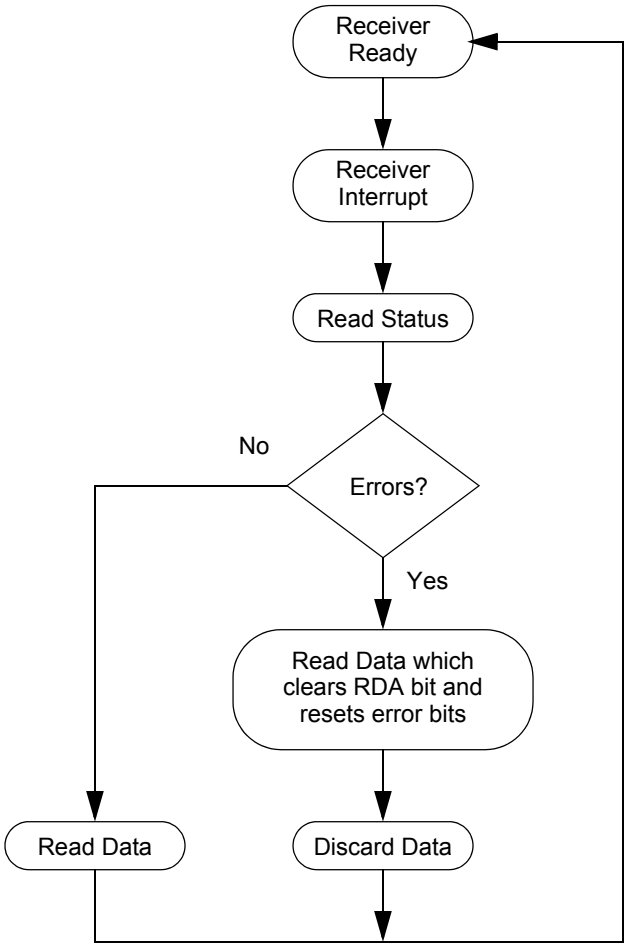


Figure 18. UART Receiver Interrupt Service Routine Flow

Baud Rate Generator Interrupts

If the Baud Rate Generator interrupt enable is set, the UART Receiver interrupt asserts when the UART Baud Rate Generator reloads. This action allows the Baud Rate Generator to function as an additional counter if the UART functionality is not employed.

UART Baud Rate Generator

The UART Baud Rate Generator creates a lower frequency baud rate clock for data transmission. The input to the Baud Rate Generator is the system clock. The UART Baud Rate High and Low Byte registers combine to create a 16-bit baud rate divisor value

PE—Parity Error
This bit indicates that a parity error has occurred. Reading the UART Receive Data register clears this bit.
0 = No parity error occurred.
1 = A parity error occurred.

OE—Overrun Error
This bit indicates that an overrun error has occurred. An overrun occurs when new data is received and the UART Receive Data register has not been read. If the RDA bit is reset to 0, then reading the UART Receive Data register clears this bit.
0 = No overrun error occurred.
1 = An overrun error occurred.

FE—Framing Error
This bit indicates that a framing error (no Stop bit following data reception) was detected. Reading the UART Receive Data register clears this bit.
0 = No framing error occurred.
1 = A framing error occurred.

BRKD—Break Detect
This bit indicates that a break occurred. If the data bits, parity/multiprocessor bit, and Stop bit(s) are all zeros then this bit is set to 1. Reading the UART Receive Data register clears this bit.
0 = No break occurred.
1 = A break occurred.

TDRE—Transmitter Data Register Empty
This bit indicates that the UART Transmit Data register is empty and ready for additional data. Writing to the UART Transmit Data register resets this bit.
0 = Do not write to the UART Transmit Data register.
1 = The UART Transmit Data register is ready to receive an additional byte to be transmitted.

TXE—Transmitter Empty
This bit indicates that the transmit shift register is empty and character transmission is finished.
0 = Data is currently transmitting.
1 = Transmission is complete.

CTS—$\overline{\text{CTS}}$ signal
When this bit is read it returns the level of the $\overline{\text{CTS}}$ signal.

UART Status 1 Register

This register contains multiprocessor control and status bits.

During an SPI transfer, data is sent and received simultaneously by both the Master and the Slave SPI devices. Separate signals are required for data and the serial clock. When an SPI transfer occurs, a multi-bit (typically 8-bit) character is shifted out one data pin and an multi-bit character is simultaneously shifted in on a second data pin. An 8-bit shift register in the Master and another 8-bit shift register in the Slave are connected as a circular buffer. The SPI shift register is single-buffered in the transmit and receive directions. New data to be transmitted cannot be written into the shift register until the previous transmission is complete and receive data (if valid) has been read.

SPI Signals

The four basic SPI signals are:

- Master-In/Slave-Out
- Master-Out/Slave-In
- Serial Clock
- Slave Select

Each signal is described in both Master and Slave modes.

Master-In/Slave-Out

The Master-In/Slave-Out (MISO) pin is configured as an input in a Master device and as an output in a Slave device. It is one of the two lines that transfer serial data, with the most significant bit sent first. The MISO pin of a Slave device is placed in a high-impedance state if the Slave is not selected. When the SPI is not enabled, this signal is in a high-impedance state.

Master-Out/Slave-In

The Master-Out/Slave-In (MOSI) pin is configured as an output in a Master device and as an input in a Slave device. It is one of the two lines that transfer serial data, with the most significant bit sent first. When the SPI is not enabled, this signal is in a high-impedance state.

Serial Clock

The Serial Clock (SCK) synchronizes data movement both in and out of the device through its MOSI and MISO pins. In MASTER mode, the SPI's Baud Rate Generator creates the serial clock. The Master drives the serial clock out its own SCK pin to the Slave's SCK pin. When the SPI is configured as a Slave, the SCK pin is an input and the clock signal from the Master synchronizes the data transfer between the Master and Slave devices. Slave devices ignore the SCK signal, unless the $\overline{SS}$ pin is asserted. When configured as a slave, the SPI block requires a minimum SCK period of greater than or equal to 8 times the system (XIN) clock period.

Table 70. I²C Data Register (I2CDATA)

BITS	7	6	5	4	3	2	1	0
FIELD	DATA							
RESET	0							
R/W	R/W							
ADDR	F50H							

I²C Status Register

The Read-only I²C Status register (Table 71) indicates the status of the I²C Controller.

Table 71. I²C Status Register (I2CSTAT)

BITS	7	6	5	4	3	2	1	0
FIELD	TDRE	RDRF	ACK	10B	RD	TAS	DSS	NCKI
RESET	1	0						
R/W	R							
ADDR	F51H							

TDRE—Transmit Data Register Empty
When the I²C Controller is enabled, this bit is 1 when the I²C Data register is empty. When this bit is set, an interrupt is generated if the TXI bit is set, except when the I²C Controller is shifting in data during the reception of a byte or when shifting an address and the RD bit is set. This bit is cleared by writing to the I2CDATA register.

RDRF—Receive Data Register Full
This bit is set = 1 when the I²C Controller is enabled and the I²C Controller has received a byte of data. When asserted, this bit causes the I²C Controller to generate an interrupt. This bit is cleared by reading the I²C Data register (unless the read is performed using execution of the On-Chip Debugger's Read Register command).

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 occurred for the last byte transmitted or received. This bit is cleared when IEN = 0 or when a Not Acknowledge occurred for the last byte transmitted or received. It is not reset at the beginning of each transaction and is not reset when this register is read.

Direct Memory Access Controller

Overview

The 64K Series Direct Memory Access (DMA) Controller provides three independent Direct Memory Access channels. Two of the channels (DMA0 and DMA1) transfer data between the on-chip peripherals and the Register File. The third channel (DMA_ADC) controls the ADC operation and transfers SINGLE-SHOT mode ADC output data to the Register File.

Operation

DMA0 and DMA1 Operation

DMA0 and DMA1, referred to collectively as DMA*x*, transfer data either from the on-chip peripheral control registers to the Register File, or from the Register File to the on-chip peripheral control registers. The sequence of operations in a DMA*x* data transfer is:

1. DMA*x* trigger source requests a DMA data transfer.
2. DMA*x* requests control of the system bus (address and data) from the eZ8 CPU.
3. After the eZ8 CPU acknowledges the bus request, DMA*x* transfers either a single byte or a two-byte word (depending upon configuration) and then returns system bus control back to the eZ8 CPU.
4. If Current Address equals End Address:
   – DMA*x* reloads the original Start Address
   – If configured to generate an interrupt, DMA*x* sends an interrupt request to the Interrupt Controller
   – If configured for single-pass operation, DMA*x* resets the DEN bit in the DMA*x* Control register to 0 and the DMA is disabled.

   If Current Address does not equal End Address, the Current Address increments by 1 (single-byte transfer) or 2 (two-byte word transfer).

Follow the steps below for setting up the ADC and initiating continuous conversion:

1. Enable the desired analog input by configuring the general-purpose I/O pins for alternate function. This disables the digital input and output driver.
2. Write to the ADC Control register to configure the ADC for continuous conversion. The bit fields in the ADC Control register may be written simultaneously:
   – Write to the ANAIN[3:0] field to select one of the 12 analog input sources.
   – Set CONT to 1 to select continuous conversion.
   – Write to the ~~VREF~~ bit to enable or disable the internal voltage reference generator.
   – Set CEN to 1 to start the conversions.
3. When the first conversion in continuous operation is complete (after 5129 system clock cycles, plus the 40 cycles for power-up, if necessary), the ADC control logic performs the following operations:
   – CEN resets to 0 to indicate the first conversion is complete. CEN remains 0 for all subsequent conversions in continuous operation.
   – An interrupt request is sent to the Interrupt Controller to indicate the conversion is complete.
4. Thereafter, the ADC writes a new 10-bit data result to {ADCD_H[7:0], ADCD_L[7:6]} every 256 system clock cycles. An interrupt request is sent to the Interrupt Controller when each conversion is complete.
5. To disable continuous conversion, clear the CONT bit in the ADC Control register to 0.

DMA Control of the ADC

The Direct Memory Access (DMA) Controller can control operation of the ADC including analog input selection and conversion enable. For more information on the DMA and configuring for ADC operations, see Direct Memory Access Controller on page 165.

Timing Using the Flash Frequency Registers

Before performing a program or erase operation on the Flash memory, you must first configure the Flash Frequency High and Low Byte registers. The Flash Frequency registers allow programming and erasure of the Flash with system clock frequencies ranging from 20 kHz through 20 MHz (the valid range is limited to the device operating frequencies).

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 Flash Frequency value must contain the system clock frequency in kHz. This value is calculated using the following equation:.

$$\text{FFREQ}[15:0] = \frac{\text{System Clock Frequency (Hz)}}{1000}$$

Caution: *Flash programming and erasure are not supported for system clock frequencies below 20 kHz, above 20 MHz, or outside of the device operating frequency range. The Flash Frequency High and Low Byte registers must be loaded with the correct value to insure proper Flash programming and erase operations.*

Flash Read Protection

The user code contained within the Flash memory can be protected from external access. Programming the Flash Read Protect Option Bit prevents reading of user code by the On-Chip Debugger or by using the Flash Controller Bypass mode. For more information, see Option Bits on page 195 and On-Chip Debugger on page 199.

Flash Write/Erase Protection

The 64K Series provides several levels of protection against accidental program and erasure of the Flash memory contents. This protection is provided by the Flash Controller unlock mechanism, the Flash Sector Protect register, and the Flash Write Protect option bit.

Flash Controller Unlock Mechanism

At Reset, the Flash Controller locks to prevent accidental program or erasure of the Flash memory. To program or erase the Flash memory, the Flash controller must be unlocked. After unlocking the Flash Controller, the Flash can be programmed or erased. Any value written by user code to the Flash Control register or Page Select Register out of sequence will lock the Flash Controller.

Follow the steps below to unlock the Flash Controller from user code:

1. Write 00H to the Flash Control register to reset the Flash Controller.
2. Write the page to be programmed or erased to the Page Select register.

Table 133. eZ8 CPU Instruction Summary (Continued)

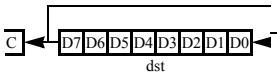
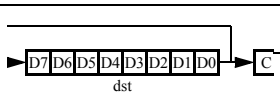
Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
OR dst, src	dst ← dst OR src	r	r	42	-	*	*	0	-	-	2	3
		r	Ir	43							2	4
		R	R	44							3	3
		R	IR	45							3	4
		R	IM	46							3	3
		IR	IM	47							3	4
ORX dst, src	dst ← 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

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
SWAP dst	dst[7:4] ↔ dst[3:0]	R		F0	X	*	*	X	-	-	2	2
		IR		F1							2	3
TCM dst, src	(NOT dst) AND src	r	r	62	-	*	*	0	-	-	2	3
		r	Ir	63							2	4
		R	R	64							3	3
		R	IR	65							3	4
		R	IM	66							3	3
		IR	IM	67							3	4
TCMX dst, src	(NOT dst) AND src	ER	ER	68	-	*	*	0	-	-	4	3
		ER	IM	69							4	3
TM dst, src	dst AND src	r	r	72	-	*	*	0	-	-	2	3
		r	Ir	73							2	4
		R	R	74							3	3
		R	IR	75							3	4
		R	IM	76							3	3
		IR	IM	77							3	4
TMX dst, src	dst AND src	ER	ER	78	-	*	*	0	-	-	4	3
		ER	IM	79							4	3
TRAP Vector	SP ← SP – 2 @SP ← PC SP ← SP – 1 @SP ← FLAGS PC ← @Vector		Vector	F2	-	-	-	-	-	-	2	6
WDT				5F	-	-	-	-	-	-	1	2

Opcode Maps

A description of the opcode map data and the abbreviations are provided in Figure 59 and Table 134 on page 262. Figure 60 on page 263 and Figure 61 on page 264 provide information on each of the eZ8™ CPU instructions.

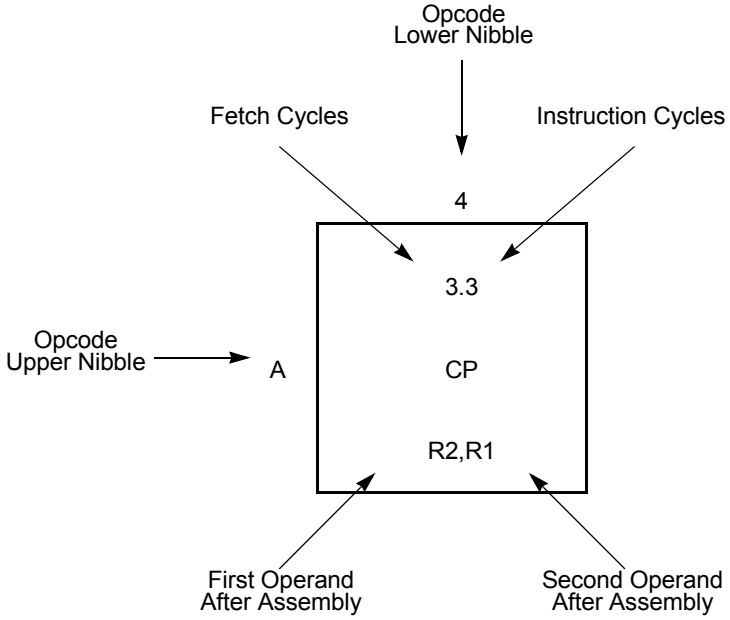
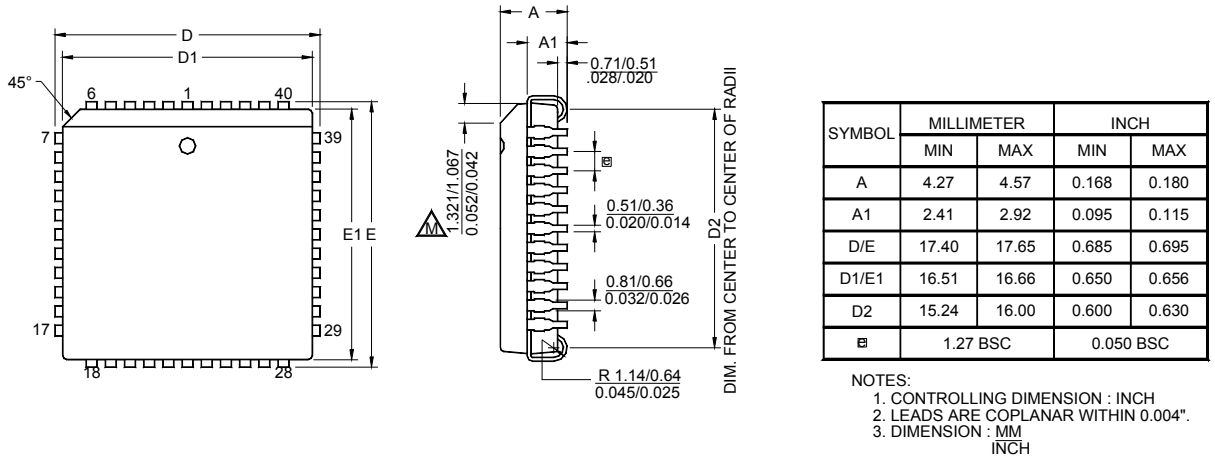


Figure 59. Opcode Map Cell Description

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

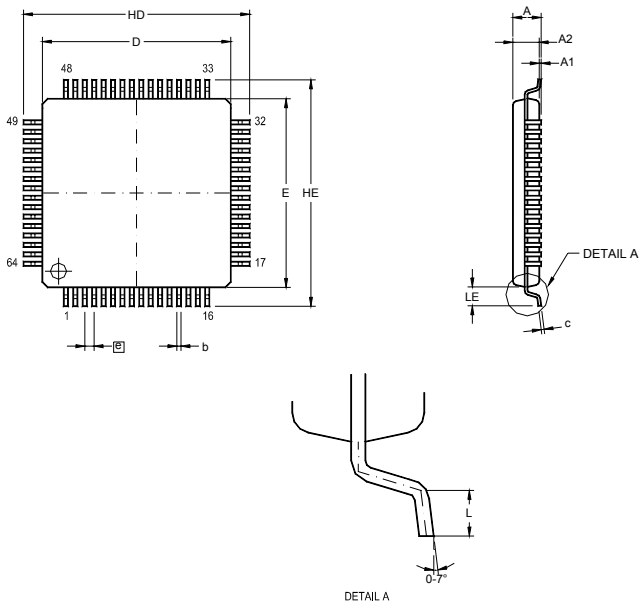


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.27	4.57	0.168	0.180
A1	2.41	2.92	0.095	0.115
D/E	17.40	17.65	0.685	0.695
D1/E1	16.51	16.66	0.650	0.656
D2	15.24	16.00	0.600	0.630
e	1.27 BSC		0.050 BSC	

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.

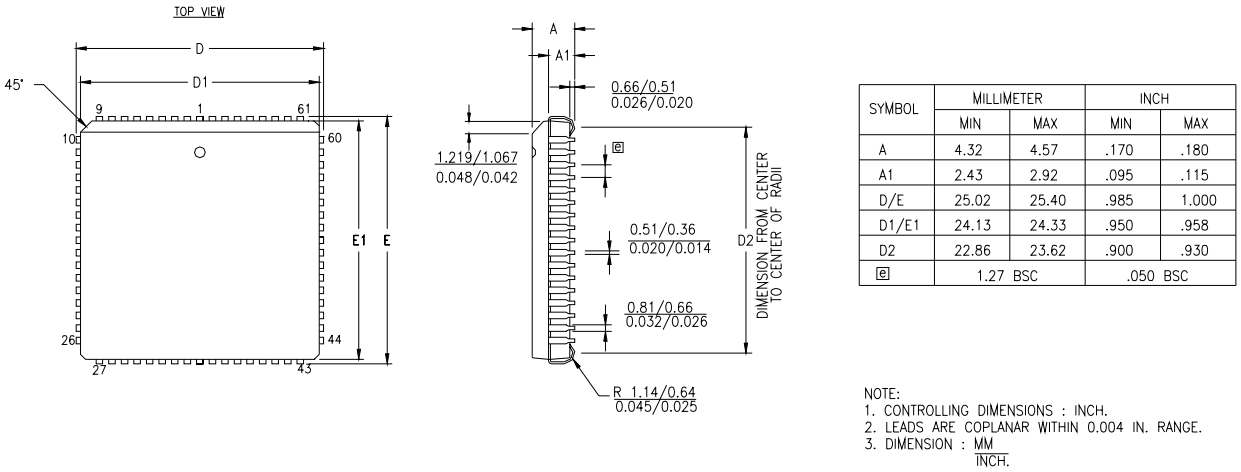


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	1.40	1.60	0.055	0.063
A1	0.05	0.15	0.002	0.006
A2	1.35	1.45	0.053	0.057
b	0.17	0.27	0.007	0.011
c	0.09	0.20	0.004	0.008
HD	11.75	12.25	0.463	0.482
D	9.90	10.10	0.390	0.398
HE	11.75	12.25	0.463	0.482
E	9.90	10.10	0.390	0.398
e	0.50 BSC		0.0197 BSC	
L	0.45	0.75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

Figure 66 displays the 68-pin Plastic Lead Chip Carrier (PLCC) package available for the Z8X1622, Z8X2422, Z8X3222, Z8X4822, and Z8X6422 devices.

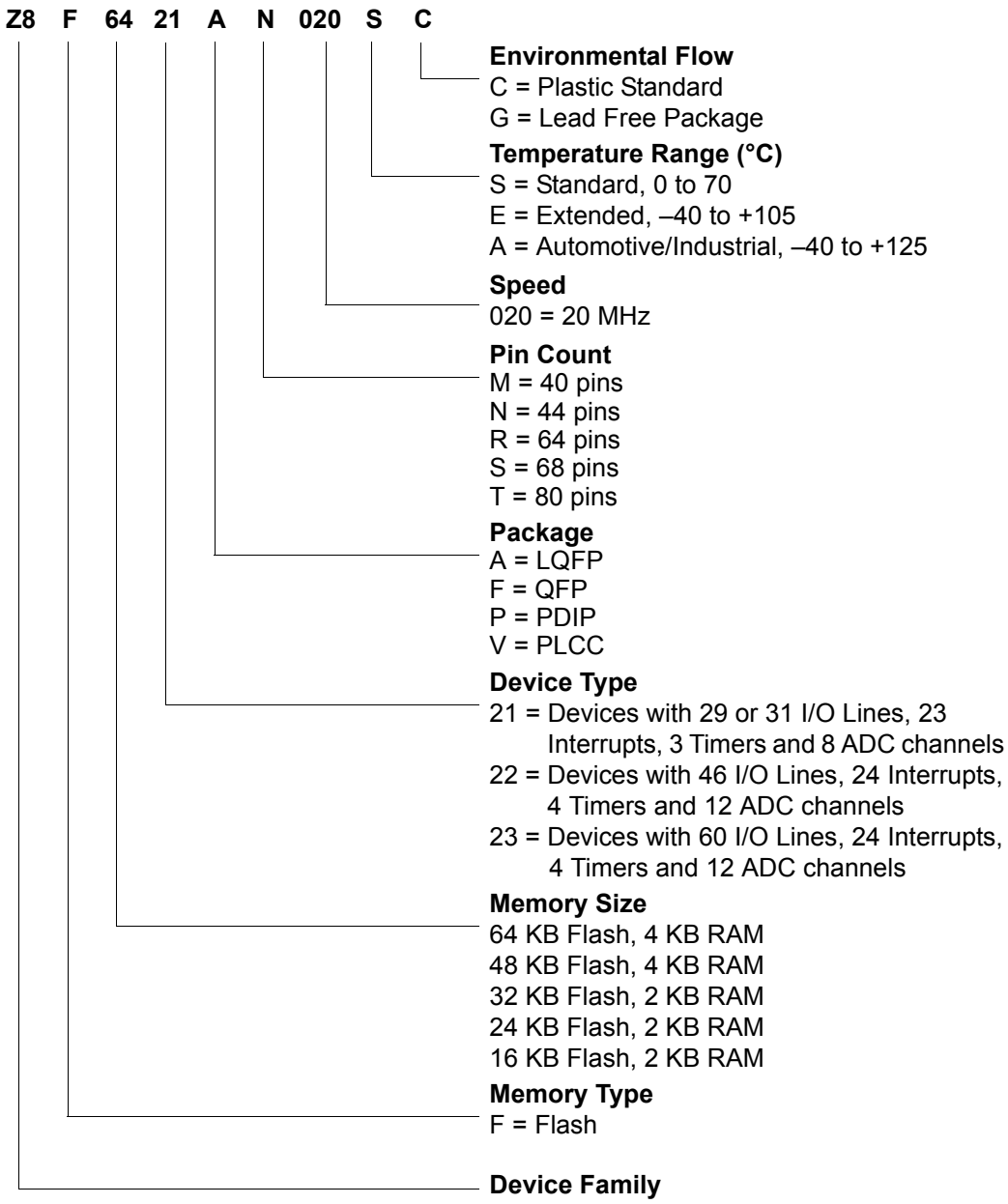


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	4.32	4.57	.170	.180
A1	2.43	2.92	.095	.115
D/E	25.02	25.40	.985	1.000
D1/E1	24.13	24.33	.950	.958
D2	22.86	23.62	.900	.930
e	1.27 BSC		.050 BSC	

Figure 66. 68-Lead Plastic Lead Chip Carrier Package (PLCC)

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



T

U