Zilog - Z8F4802AR020SC Datasheet

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	46
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 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/z8f4802ar020sc

Introduction

The Z8 Encore!® MCU family of products are the first in a line of ZiLOG microcontroller products based upon the new 8-bit eZ8 CPU. The Z8F640x/Z8F480x/Z8F320x/Z8F240x/Z8F160x products are referred to collectively as either Z8 Encore!® or the Z8F640x family. The Z8F640x family of products introduce Flash memory to ZiLOG's extensive line of 8-bit microcontrollers. The Flash in-circuit programming capability allows for faster development time and program changes in the field. The new eZ8 CPU is upward compatible with existing Z8 instructions. The rich peripheral set of the Z8F640x family makes it suitable for a variety of applications including motor control, security systems, home appliances, personal electronic devices, and sensors.

Features

- eZ8 CPU, 20 MHz operation
- 12-channel, 10-bit analog-to-digital converter (ADC)
- 3-channel DMA
- Up to 64KB Flash memory with in-circuit programming capability
- Up to 4KB register RAM
- Serial communication protocols
  - Serial Peripheral Interface
  - I^2C
- Two full-duplex 9-bit UARTs
- 24 interrupts with programmable priority
- Three or four 16-bit timers with capture, compare, and PWM capability
- Single-pin On-Chip Debugger
- Two Infrared Data Association (IrDA)-compliant infrared encoder/decoders integrated with the UARTs
- Watch-Dog Timer (WDT) with internal RC oscillator
- Up to 60 I/O pins
- Voltage Brown-out Protection (VBO)

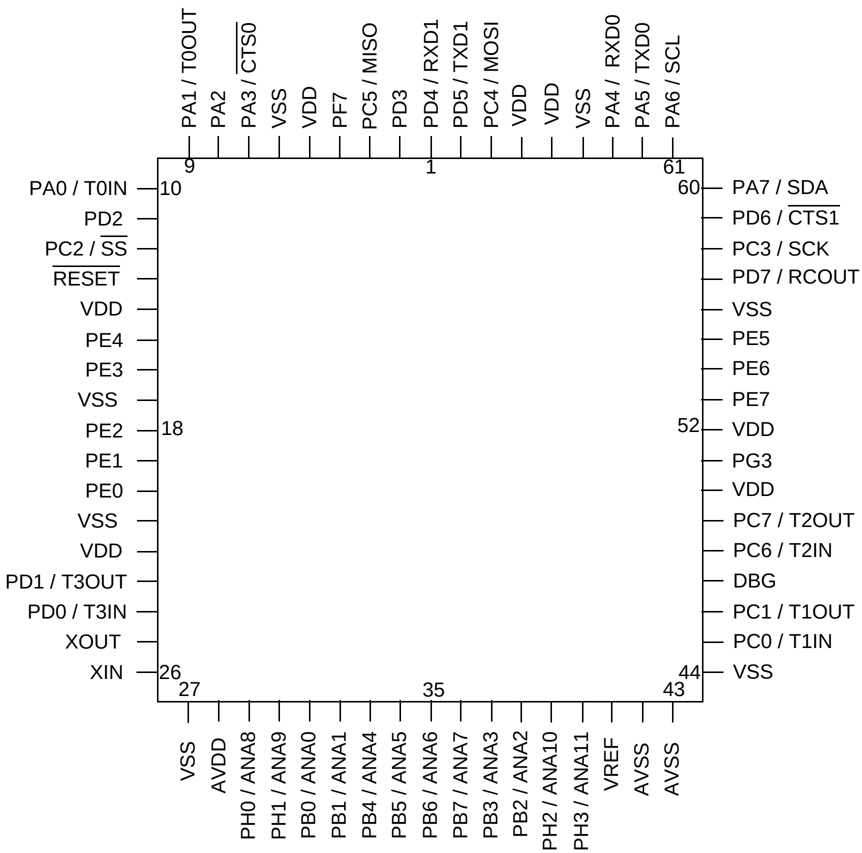


Figure 60. Z8Fxx02 in 68-Pin Plastic Leaded Chip Carrier (PLCC)

Table 12. GPIO Port Registers and Sub-Registers

Port Register Mnemonic	Port Register Name
P*x*ADDR	Port A-H Address Register (Selects sub-registers)
P*x*CTL	Port A-H Control Register (Provides access to sub-registers)
P*x*IN	Port A-H Input Data Register
P*x*OUT	Port A-H Output Data Register
Port Sub-Register Mnemonic	**Port Register Name**
P*x*DD	Data Direction
P*x*AF	Alternate Function
P*x*OC	Output Control (Open-Drain)
P*x*HDE	High Drive Enable
P*x*SMRE	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 13).

Table 13. Port A-H GPIO Address Registers (P*x*ADDR)

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							

I^2CI— I^2C Interrupt Request
0 = No interrupt request is pending for the I^2C.
1 = An interrupt request from the I^2C is awaiting service.

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

ADCI—ADC Interrupt Request
0 = No interrupt request is pending for the Analog-to-Digital Converter.
1 = An interrupt request from the Analog-to-Digital Converter is awaiting service.

Interrupt Request 1 Register

The Interrupt Request 1 (IRQ1) register (Table 24) stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ1 register becomes 1. If interrupts are globally enabled (vectored interrupts), the interrupt controller passes an interrupt request to the eZ8 CPU. If interrupts are globally disabled (polled interrupts), the eZ8 CPU can read the Interrupt Request 1 register to determine if any interrupt requests are pending.

Table 24. Interrupt Request 1 Register (IRQ1)

BITS	7	6	5	4	3	2	1	0
FIELD	PAD7I	PAD6I	PAD5I	PAD4I	PAD3I	PAD2I	PAD1I	PAD0I
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	FC3H							

PAD*x*I—Port A or Port D Pin *x* Interrupt Request
0 = No interrupt request is pending for GPIO Port A or Port D pin *x*.
1 = An interrupt request from GPIO Port A or Port D pin *x* is awaiting service.

where *x* indicates the specific GPIO Port pin number (0 through 7). For each pin, only 1 of either Port A or Port D can be enabled for interrupts at any one time. Port selection (A or D) is determined by the values in the Interrupt Port Select Register.

6. 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 select either even or odd parity.

The UART and DMA are now configured for data reception and automatic data transfer to the Register File. When a valid data byte is received by the UART the following occurs:

7. The UART notifies the DMA Controller that a data byte is available in the UART Receive Data register.
8. The DMA Controller requests control of the system bus from the eZ8 CPU.
9. The eZ8 CPU acknowledges the bus request.
10. The DMA Controller transfers the data from the UART Receive Data register to another location in RAM and then return bus control back to the eZ8 CPU.

The UART and DMA can continue to transfer incoming data bytes without eZ8 CPU intervention. When a UART error is detected, the UART Receiver interrupt is generated. The associated interrupt service routine (ISR) should perform the following:

11. Check the UART Status 0 register to determine the source of the UART error or break condition and then respond appropriately.

Multiprocessor (9-bit) mode

The UART has a Multiprocessor mode that uses an extra (9th) bit for selective communication when a number of processors share a common UART bus. In Multiprocessor (9-bit) 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 illustrated in Figure 70. The character format is:

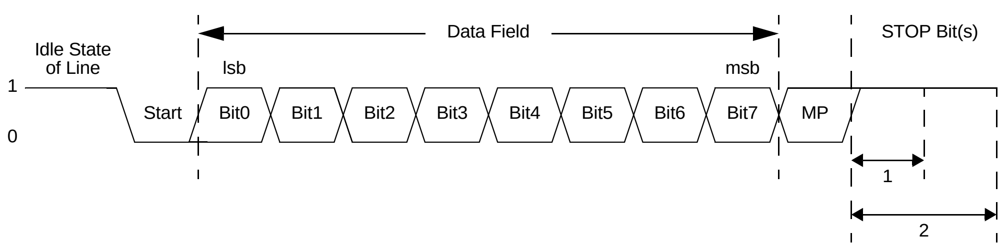


Figure 70. UART Asynchronous Multiprocessor (9-bit) Mode Data Format

In Multiprocessor (9-bit) mode, parity is not an option as 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.

SPI Control Register

The SPI Control register configures the SPI for transmit and receive operations.

Table 61. SPI Control Register (SPICTL)

BITS	7	6	5	4	3	2	1	0
FIELD	IRQE	STR	BIRQ	PHASE	CLKPOL	WOR	MMEN	SPIEN
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	F61H							

IRQE—Interrupt Request Enable
0 = SPI interrupts are disabled. No interrupt requests are sent to the Interrupt Controller.
1 = SPI interrupts are enabled. Interrupt requests are sent to the Interrupt Controller.

STR—Start an SPI Interrupt Request
0 = No effect.
1 = Setting this bit to 1 also sets the `IRQ` bit in the SPI Status register to 1. Setting this bit forces the SPI to send an interrupt request to the Interrupt Control. This bit can be used by software for a function similar to transmit buffer empty in a UART.

BIRQ—BRG Timer Interrupt Request
If the SPI is enabled, this bit has no effect. If the SPI is disabled:
0 = The Baud Rate Generator timer function is disabled.
1 = The Baud Rate Generator timer function and time-out interrupt are enabled.

PHASE—Phase Select
Sets the phase relationship of the data to the clock. Refer to the **SPI Clock Phase and Polarity Control** section for more information on operation of the `PHASE` bit.

CLKPOL—Clock Polarity
0 = SCK idles Low (0).
1 = SCK idle High (1).

WOR—Wire-OR (Open-Drain) Mode Enabled
0 = SPI signal pins not configured for open-drain.
1 = All four SPI signal pins (SCK, $\overline{SS}$, MISO, MOSI) configured for open-drain function. This setting is typically used for multi-master and/or multi-slave configurations.

MMEN—SPI Master Mode Enable
0 = SPI configured in Slave mode.
1 = SPI configured in Master mode.

START—Send Start Condition
This bit sends the Start condition. Once asserted, it is cleared by the I^2C Controller after it sends the START condition or by deasserting the IEN bit. After this bit is set, the Start condition is sent if there is data in the I^2C Data or I^2C Shift register. If there is no data in one of these registers, the I^2C Controller waits until data is loaded. If this bit is set while the I^2C Controller is shifting out data, it generates a START condition after the byte shifts and the acknowledge phase completed. If the STOP bit is also set, it also waits until the STOP condition is sent before the START condition. If this bit is 1, it cannot be cleared to 0 by writing to the register.This bit clears when the I^2C is disabled.

STOP—Send Stop Condition
This bit causes the I^2C Controller to issue a Stop condition after the byte in the I^2C Shift register has completed transmission or after a byte has been received in a receive operation. Once set, this bit is reset by the I^2C Controller after a Stop condition has been sent or by deasserting the IEN bit. If this bit is 1, it cannot be cleared to 0 by writing to the register.This bit clears when the I^2C is disabled.

BIRQ—Baud Rate Generator Interrupt Request
This bit causes an interrupt to occur every time the baud rate generator counts down to zero. This bit allows the I^2C Controller to be used as an additional counter when it is not being used elsewhere. This bit must only be set when the I^2C Controller is disabled.

TXI—Enable TDRE interrupts
This bit enables interrupts when the I^2C Data register is empty on the I^2C Controller.

NAK—Send NAK
This bit sends a Not Acknowledge condition after the next byte of data has been read from the I^2C slave. Once asserted, it is deasserted after a Not Acknowledge is sent or the IEN bit is deasserted.

FLUSH—Flush Data
Setting this bit to 1 clears the I^2C Data register and sets the TDRE bit to 1. This bit allows flushing of the I^2C Data register when an NAK is received after the data has been sent to the I^2C Data register. Reading this bit always returns 0.

FILTEN—I^2C Signal Filter Enable
Setting this bit to 1 enables low-pass digital filters on the SDA and SCL input signals. These filters reject any input pulse with periods less than a full system clock cycle. The filters introduce a 3-system clock cycle latency on the inputs.

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.

ADC Data High Byte Register

The ADC Data High Byte register contains the upper eight bits of the 10-bit ADC output. During a conversion, this value is invalid. Access to the ADC Data High Byte register is read-only. The full 10-bit ADC result is given by {ADCD_H[7:0], ADCD_L[7:6]}.

Table 81. ADC Data High Byte Register (ADCD_H)

BITS	7	6	5	4	3	2	1	0
FIELD	ADCD_H							
RESET	X							
R/W	R							
ADDR	F72H							

ADCD_H—ADC Data High Byte
This byte contains the upper eight bits of the 10-bit ADC output. These bits are not valid during a conversion. These bits are undefined after a Reset.

ADC Data Low Bits Register

The ADC Data Low Bits register contains the lower two bits of the conversion value. During a conversion this value is invalid. Access to the ADC Data Low Bits register is read-only. The full 10-bit ADC result is given by {ADCD_H[7:0], ADCD_L[7:6]}.

Table 82. ADC Data Low Bits Register (ADCD_L)

BITS	7	6	5	4	3	2	1	0
FIELD	ADCD_L		Reserved					
RESET	X		X					
R/W	R		R					
ADDR	F73H							

ADCD_L—ADC Data Low Bits
These are the least significant two bits of the 10-bit ADC output. During a conversion, this value is invalid. These bits are undefined after a Reset.

Reserved
These bits are reserved and are always undefined.

Option Bits

Overview

Option Bits allow user configuration of certain aspects of Z8F640x family device operation. The feature configuration data is stored in the Program Memory and read during Reset. The features available for control via the Option Bits are:

- Watch-Dog Timer time-out response selection–interrupt or Short Reset.
- Watch-Dog Timer enabled at Reset.
- The ability to prevent unwanted read access to user code in Program Memory.
- The ability to prevent accidental programming and erasure of all or a portion of the user code in Program Memory.

Operation

Option Bit Configuration By Reset

Each time the Option Bits are programmed or erased, the Z8F640x family device must be Reset for the change to take place. During any reset operation (System Reset, Short Reset, or Stop Mode Recovery), the Option Bits are automatically read from the Program Memory and written to Option Configuration registers. The Option Configuration registers control operation of the Z8F640x family device. Option Bit control of the Z8F640x family device is established before the device exits Reset and the eZ8 CPU begins code execution. The Option Configuration registers are not part of the Register File and are not accessible for read or write access.

Option Bit Address Space

The first two bytes of Program Memory at addresses `0000H` and `0001H` are reserved for the user Option Bits. The byte at Program Memory address `0000H` is used to configure user options. The byte at Program Memory address `0001H` is reserved for future use and must be left in its unprogrammed state.

Table 93. On-Chip Debugger Commands

Debug Command	Command Byte	Enabled when NOT in Debug mode?	Disabled by Read Protect Option Bit
Write Program Memory	0AH	-	Disabled
Read Program Memory	0BH	-	Disabled
Write Data Memory	0CH	-	Yes
Read Data Memory	0DH	-	-
Read Program Memory CRC	0EH	-	-
Reserved	0FH	-	-
Step Instruction	10H	-	Disabled
Stuff Instruction	11H	-	Disabled
Execute Instruction	12H	-	Disabled
Reserved	13H - 1FH	-	-
Write Watchpoint	20H	-	Disabled
Read Watchpoint	21H	-	-
Reserved	22H - FFH	-	-

In the following bulleted list of OCD Commands, data and commands sent from the host to the On-Chip Debugger are identified by ’`DBG <-- Command/Data`’. Data sent from the On-Chip Debugger back to the host is identified by ’`DBG --> Data`’

- **Read OCD Revision (00H)**—The Read OCD Revision command is used to determine the version of the On-Chip Debugger. If OCD commands are added, removed, or changed, this revision number changes.

```
DBG <-- 00H
DBG --> OCDREV[15:8] (Major revision number)
DBG --> OCDREV[7:0] (Minor revision number)
```

- **Read OCD Status Register (02H)**—The Read OCD Status Register command is used to read the OCDSTAT register.

```
DBG <-- 02H
DBG --> OCDSTAT[7:0]
```

- **Read Runtime Counter (03H)**—The Runtime Counter is used to count Z8 Encore! system clock cycles in between Breakpoints. The 16-bit Runtime Counter counts up from `0000H` and stops at the maximum count of `FFFFH`. The Runtime Counter is overwritten during the Write Memory, Read Memory, Write Register, Read Register, Read Memory CRC, Step Instruction, Stuff Instruction, and Execute Instruction commands.

DC Characteristics

Table 101 lists the DC characteristics of the Z8F640x family devices. All voltages are referenced to V_{SS}, the primary system ground.

Table 101. DC Characteristics

Symbol	Parameter	$T_A = -40^0C$ to 105^0C			Units	Conditions
		Minimum	Typical	Maximum		
V_{DD}	Supply Voltage	3.0	–	3.6	V	
V_{IL1}	Low Level Input Voltage	-0.3	–	$0.3*V_{DD}$	V	For all input pins except $\overline{\text{RESET}}$, DBG, and XIN.
V_{IL2}	Low Level Input Voltage	-0.3	–	$0.2*V_{DD}$	V	For $\overline{\text{RESET}}$, DBG, and XIN.
V_{IH1}	High Level Input Voltage	$0.7*V_{DD}$	–	5.5	V	Port A, C, D, E, F, and G pins.
V_{IH2}	High Level Input Voltage	$0.7*V_{DD}$	–	$V_{DD}+0.3$	V	Port B and H pins.
V_{IH3}	High Level Input Voltage	$0.8*V_{DD}$	–	$V_{DD}+0.3$	V	$\overline{\text{RESET}}$, DBG, and XIN pins.
V_{OL1}	Low Level Output Voltage	–	–	0.4	V	$V_{DD} = 3.0V$; $I_{OL} = 2mA$ High Output Drive disabled.
V_{OH1}	High Level Output Voltage	2.4	–	–	V	$V_{DD} = 3.0V$; $I_{OH} = -2mA$ High Output Drive disabled.
V_{OL2}	Low Level Output Voltage	–	–	0.6	V	$V_{DD} = 3.3V$; $I_{OL} = 20mA$ High Output Drive enabled. $T_A = -40^0C$ to $+70^0C$
V_{OL3}	Low Level Output Voltage	–	–	0.6	V	$V_{DD} = 3.3V$; $I_{OL} = 15mA$ High Output Drive enabled. $T_A = 70^0C$ to $+105^0C$
V_{OH2}	High Level Output Voltage	2.4	–	–	V	$V_{DD} = 3.3V$; $I_{OH} = -20mA$ High Output Drive enabled. $T_A = -40^0C$ to $+70^0C$
V_{OH3}	High Level Output Voltage	2.4	–	–	V	$V_{DD} = 3.3V$; $I_{OH} = -15mA$ High Output Drive enabled. $T_A = 70^0C$ to $+105^0C$
I_{IL}	Input Leakage Current	-5	–	+5	µA	$V_{DD} = 3.6V$; V_{IN} = VDD or VSS[1]
I_{TL}	Tri-State Leakage Current	-5	–	+5	µA	$V_{DD} = 3.6V$
C_{PAD}	GPIO Port Pad Capacitance	–	8.0[2]	–	pF	
C_{XIN}	XIN Pad Capacitance	–	8.0[2]	–	pF	
C_{XOUT}	XOUT Pad Capacitance	–	9.5[2]	–	pF	

Table 125. Rotate and Shift Instructions

Mnemonic	Operands	Instruction
BSWAP	dst	Bit Swap
RL	dst	Rotate Left
RLC	dst	Rotate Left through Carry
RR	dst	Rotate Right
RRC	dst	Rotate Right through Carry
SRA	dst	Shift Right Arithmetic
SRL	dst	Shift Right Logical
SWAP	dst	Swap Nibbles

eZ8 CPU Instruction Summary

Table 126 summarizes the eZ8 CPU instructions. The table identifies the addressing modes employed by the instruction, the effect upon the Flags register, the number of CPU clock cycles required for the instruction fetch, and the number of CPU clock cycles required for the instruction execution.

Table 126. eZ8 CPU Instruction Summary

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
ADC dst, src	dst ← dst + src + C	r	r	12	*	*	*	*	0	*	2	3
		r	Ir	13							2	4
		R	R	14							3	3
		R	IR	15							3	4
		R	IM	16							3	3
		IR	IM	17							3	4
ADCX dst, src	dst ← dst + src + C	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							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

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
ADD dst, src	dst ← dst + src	r	r	02	*	*	*	*	0	*	2	3
		r	Ir	03							2	4
		R	R	04							3	3
		R	IR	05							3	4
		R	IM	06							3	3
		IR	IM	07							3	4
ADDX dst, src	dst ← dst + src	ER	ER	08	*	*	*	*	0	*	4	3
		ER	IM	09							4	3
AND dst, src	dst ← dst AND src	r	r	52	-	*	*	0	-	-	2	3
		r	Ir	53							2	4
		R	R	54							3	3
		R	IR	55							3	4
		R	IM	56							3	3
		IR	IM	57							3	4
ANDX dst, src	dst ← dst AND src	ER	ER	58	-	*	*	0	-	-	4	3
		ER	IM	59							4	3
BCLR bit, dst	dst[bit] ← 0	r		E2	-	*	*	0	-	-	2	2
BIT p, bit, dst	dst[bit] ← p	r		E2	-	*	*	0	-	-	2	2
BRK	Debugger Break			00	-	-	-	-	-	-	1	1
BSET bit, dst	dst[bit] ← 1	r		E2	-	*	*	0	-	-	2	2
BSWAP dst	dst[7:0] ← dst[0:7]	R		D5	X	*	*	0	-	-	2	2
BTJ p, bit, src, dst	if src[bit] = p PC ← PC + X		r	F6	-	-	-	-	-	-	3	3
			Ir	F7							3	4
BTJNZ bit, src, dst	if src[bit] = 1 PC ← PC + X		r	F6	-	-	-	-	-	-	3	3
			Ir	F7							3	4

Flags Notation: * = Value is a function of the result of the operation. - = Unaffected X = Undefined 0 = Reset to 0 1 = Set to 1

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
DA dst	dst ← DA(dst)	R		40	*	*	*	X	-	-	2	2
		IR		41							2	3
DEC dst	dst ← dst - 1	R		30	-	*	*	*	-	-	2	2
		IR		31							2	3
DECW dst	dst ← dst - 1	RR		80	-	*	*	*	-	-	2	5
		IRR		81							2	6
DI	IRQCTL[7] ← 0			8F	-	-	-	-	-	-	1	2
DJNZ dst, RA	dst ← dst – 1 if dst ≠ 0 PC ← PC + X	r		0A-FA	-	-	-	-	-	-	2	3
EI	IRQCTL[7] ← 1			9F	-	-	-	-	-	-	1	2
HALT	Halt Mode			7F	-	-	-	-	-	-	1	2
INC dst	dst ← dst + 1	R		20	-	*	*	*	-	-	2	2
		IR		21							2	3
		r		0E-FE							1	2
INCW dst	dst ← dst + 1	RR		A0	-	*	*	*	-	-	2	5
		IRR		A1							2	6
IRET	FLAGS ← @SP SP ← SP + 1 PC ← @SP SP ← SP + 2 IRQCTL[7] ← 1			BF	*	*	*	*	*	*	1	5
JP dst	PC ← dst	DA		8D	-	-	-	-	-	-	3	2
		IRR		C4							2	3
JP cc, dst	if cc is true PC ← dst	DA		0D-FD	-	-	-	-	-	-	3	2
JR dst	PC ← PC + X	DA		8B	-	-	-	-	-	-	2	2
JR cc, dst	if cc is true PC ← PC + X	DA		0B-FB	-	-	-	-	-	-	2	2

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.

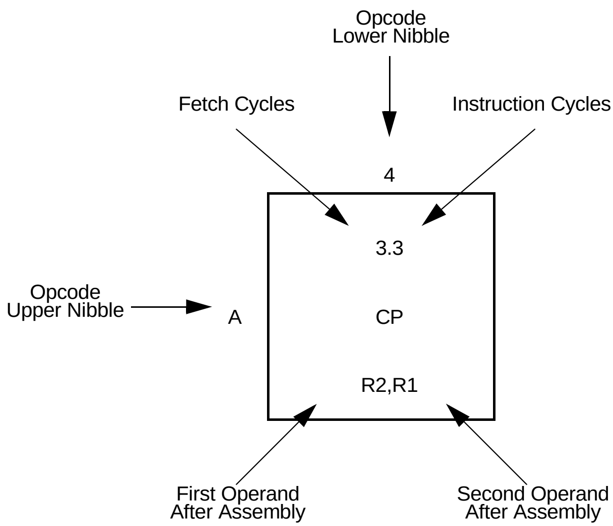
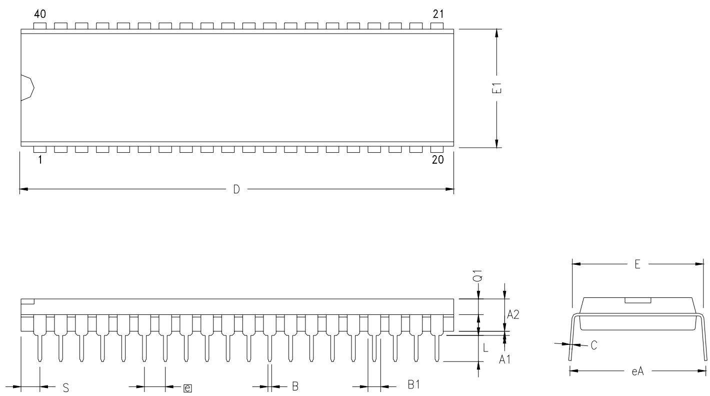


Figure 100. Opcode Map Cell Description

Packaging

Figure 103 illustrates the 40-pin PDIP (plastic dual-inline package) available for the Z8F1601, Z8F2401, Z8F3201, Z8F4801, and Z8F6401 devices.

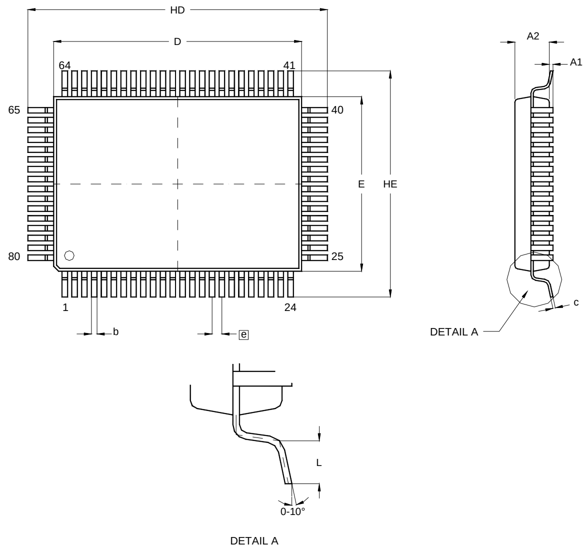


SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A1	0.51	1.02	.020	.040
A2	3.18	3.94	.125	.155
B	0.38	0.53	.015	.021
B1	1.02	1.52	.040	.060
C	0.23	0.38	.009	.015
D	52.07	52.58	2.050	2.070
E	15.24	15.75	.600	.620
E1	13.59	14.22	.535	.560
e	2.54 TYP		.100 TYP	
eA	15.49	16.76	.610	.660
L	3.05	3.81	.120	.150
Q1	1.40	1.91	.055	.075
S	1.52	2.29	.060	.090

CONTROLLING DIMENSIONS : INCH

Figure 103. 40-Lead Plastic Dual-Inline Package (PDIP)

Figure 108 illustrates the 80-pin QFP (quad flat package) available for the Z8F4803 and Z8F6403 devices.



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A1	0.10	0.38	.004	.015
A2	2.60	2.80	.102	.110
b	0.30	0.45	.012	.018
c	0.13	0.20	.005	.008
HD	23.70	24.15	.933	.951
D	19.90	20.10	.783	.791
HE	17.70	18.15	.697	.715
E	13.90	14.10	.547	.555
e	0.80 BSC		.0315 BSC	
L	0.70	1.10	.028	.043

NOTES:
1. CONTROLLING DIMENSIONS : MILLIMETER
2. LEAD COPLANARITY : MAX .10 / .004"

Figure 108. 80-Lead Quad-Flat Package (QFP)

U

V

W

X