



Welcome to [E-XFL.COM](http://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 <sup>2</sup> C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	46
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	<a href="https://www.e-xfl.com/product-detail/zilog/z8f3202ar020ec">https://www.e-xfl.com/product-detail/zilog/z8f3202ar020ec</a>



Transmitting IrDA Data .....	96
Receiving IrDA Data .....	97
Jitter .....	98
Infrared Encoder/Decoder Control Register Definitions .....	98
<b>Serial Peripheral Interface .....</b>	<b>99</b>
Overview .....	99
Architecture .....	99
Operation .....	100
SPI Signals .....	101
SPI Clock Phase and Polarity Control .....	102
Multi-Master Operation .....	104
Error Detection .....	105
SPI Interrupts .....	105
SPI Baud Rate Generator .....	105
SPI Control Register Definitions .....	106
SPI Data Register .....	106
SPI Control Register .....	107
SPI Status Register .....	108
SPI Mode Register .....	109
SPI Baud Rate High and Low Byte Registers .....	110
<b>I2C Controller .....</b>	<b>111</b>
Overview .....	111
Operation .....	111
SDA and SCL Signals .....	111
I <sup>2</sup> C Interrupts .....	112
Start and Stop Conditions .....	112
Writing a Transaction with a 7-Bit Address .....	112
Writing a Transaction with a 10-Bit Address .....	114
Reading a Transaction with a 7-Bit Address .....	115
Reading a Transaction with a 10-Bit Address .....	116
I2C Control Register Definitions .....	118
I2C Data Register .....	118
I2C Status Register .....	118
I2C Control Register .....	119
I2C Baud Rate High and Low Byte Registers .....	121
<b>Direct Memory Access Controller .....</b>	<b>122</b>
Overview .....	122
Operation .....	122
DMA0 and DMA1 Operation .....	122
Configuring DMA0 and DMA1 for Data Transfer .....	123



DMA_ADC Operation .....	123
Configuring DMA_ADC for Data Transfer .....	124
DMA Control Register Definitions .....	124
DMAx Control Register .....	124
DMAx I/O Address Register .....	125
DMAx Address High Nibble Register .....	126
DMAx Start/Current Address Low Byte Register .....	127
DMAx End Address Low Byte Register .....	128
DMA_ADC Address Register .....	128
DMA_ADC Control Register .....	130
DMA Status Register .....	131
<b>Analog-to-Digital Converter .....</b>	<b>132</b>
Overview .....	132
Architecture .....	132
Operation .....	133
Automatic Power-Down .....	133
Single-Shot Conversion .....	133
Continuous Conversion .....	134
DMA Control of the ADC .....	135
ADC Control Register Definitions .....	135
ADC Control Register .....	135
ADC Data High Byte Register .....	137
ADC Data Low Bits Register .....	137
<b>Flash Memory .....</b>	<b>138</b>
Overview .....	138
Operation .....	139
Flash Operation Timing Using the Flash Frequency Registers ..	141
Flash Code Protection Against External Access .....	141
Flash Code Protection Against Accidental Program and Erasure	141
Byte Programming .....	142
Page Erase .....	143
Mass Erase .....	143
Flash Controller Bypass .....	143
Flash Control Register Definitions .....	144
Flash Control Register .....	144
Flash Status Register .....	145
Flash Page Select Register .....	146
Flash Frequency High and Low Byte Registers .....	147



## *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<sup>2</sup>C
- 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)



- Power-On Reset (POR)
- 3.0-3.6V operating voltage with 5V-tolerant inputs
- 0° to +70°C standard temperature and -40° to +105°C extended temperature operating ranges

## Part Selection Guide

Table 1 identifies the basic features and package styles available for each device within the Z8F640x family product line.

**Table 1. Z8F640x Family Part Selection Guide**

Part Number	Flash (KB)	RAM (KB)	I/O	16-bit Timers with PWM	ADC Inputs	UARTs with IrDA	I <sup>2</sup> C	SPI	40/44-pin packages	64/68-pin packages	80-pin package
Z8F1601	16	2	31	3	8	2	1	1	X		
Z8F1602	16	2	46	4	12	2	1	1		X	
Z8F2401	24	2	31	3	8	2	1	1	X		
Z8F2402	24	2	46	4	12	2	1	1		X	
Z8F3201	32	2	31	3	8	2	1	1	X		
Z8F3202	32	2	46	4	12	2	1	1		X	
Z8F4801	48	4	31	3	8	2	1	1	X		
Z8F4802	48	4	46	4	12	2	1	1		X	
Z8F4803	48	4	60	4	12	2	1	1			X
Z8F6401	64	4	31	3	8	2	1	1	X		
Z8F6402	64	4	46	4	12	2	1	1		X	
Z8F6403	64	4	60	4	12	2	1	1			X



## Serial Peripheral Interface

The serial peripheral interface (SPI) allows the Z8 Encore!® to exchange data between other peripheral devices such as EEPROMs, A/D converters and ISDN devices. The SPI is a full-duplex, synchronous, character-oriented channel that supports a four-wire interface.

## Timers

Up to four 16-bit reloadable timers can be used for timing/counting events or for motor control operations. These timers provide a 16-bit programmable reload counter and operate in One-Shot, Continuous, Gated, Capture, Compare, Capture and Compare, and PWM modes. Only 3 timers (Timers 0-2) are available in the 40- and 44-pin packages.

## Interrupt Controller

The Z8F640x family products support up to 24 interrupts. These interrupts consist of 12 internal and 12 general-purpose I/O pins. The interrupts have 3 levels of programmable interrupt priority.

## Reset Controller

The Z8F640x family can be reset using the  $\overline{\text{RESET}}$  pin, power-on reset, Watch-Dog Timer (WDT), Stop mode exit, or Voltage Brown-Out (VBO) warning signal.

## On-Chip Debugger

The Z8 Encore!® features an integrated On-Chip Debugger (OCD). The OCD provides a rich set of debugging capabilities, such as reading and writing registers, programming the Flash, setting breakpoints and executing code. A single-pin interface provides communication to the OCD.

## DMA Controller

The Z8F640x family features three channels of DMA. Two of the channels are for register RAM to and from I/O operations. The third channel automatically controls the transfer of data from the ADC to the memory.



**Table 3. Pin Characteristics of the Z8F640x family**

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

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

Table 11. Port Alternate Function Mapping (Continued)

Port	Pin	Mnemonic	Alternate Function Description
<b>Port D</b>	PD0	T3IN	Timer 3 In (not available in 40- and 44-pin packages)
	PD1	T3OUT	Timer 3 Out (not available in 40- and 44-pin packages)
	PD2	N/A	No alternate function
	PD3	N/A	No alternate function
	PD4	RXD1 / IRRX1	UART 1 / IrDA 1 Receive Data
	PD5	TXD1 / IRTX1	UART 1 / IrDA 1 Transmit Data
	PD6	$\overline{\text{CTS1}}$	UART 1 Clear to Send
	PD7	RCOUT	Watch-Dog Timer RC Oscillator Output
<b>Port E</b>	PE[7:0]	N/A	No alternate functions
<b>Port F</b>	PF[7:0]	N/A	No alternate functions
<b>Port G</b>	PG[7:0]	N/A	No alternate functions
<b>Port H</b>	PH0	ANA8	ADC Analog Input 8
	PH1	ANA9	ADC Analog Input 9
	PH2	ANA10	ADC Analog Input 10
	PH3	ANA11	ADC Analog Input 11

## GPIO Interrupts

Many of the GPIO port pins can be used as interrupt sources. Some port pins may be configured to generate an interrupt request on either the rising edge or falling edge of the pin input signal. Other port pin interrupts generate an interrupt when any edge occurs (both rising and falling). Refer to the **Interrupt Controller** chapter for more information on interrupts using the GPIO pins.

## GPIO Control Register Definitions

Four registers for each Port provide access to GPIO control, input data, and output data. Table 12 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 30. IRQ1 Enable High Bit Register (IRQ1ENH)

BITS	7	6	5	4	3	2	1	0
FIELD	PAD7ENH	PAD6ENH	PAD5ENH	PAD4ENH	PAD3ENH	PAD2ENH	PAD1ENH	PAD0ENH
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	FC4H							

PADxENH—Port A or Port D Bit[x] Interrupt Request Enable High Bit  
Refer to the Interrupt Port Select register for selection of either Port A or Port D as the interrupt source.

Table 31. IRQ1 Enable Low Bit Register (IRQ1ENL)

BITS	7	6	5	4	3	2	1	0
FIELD	PAD7ENL	PAD6ENL	PAD5ENL	PAD4ENL	PAD3ENL	PAD2ENL	PAD1ENL	PAD0ENL
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	FC5H							

PADxENL—Port A or Port D Bit[x] Interrupt Request Enable Low Bit  
Refer to the Interrupt Port Select register for selection of either Port A or Port D as the interrupt source.

## IRQ2 Enable High and Low Bit Registers

The IRQ2 Enable High and Low Bit registers (Tables 33 and 34) form a priority encoded enabling for interrupts in the Interrupt Request 2 register. Priority is generated by setting bits in each register. Table 32 describes the priority control for IRQ2.

Table 32. IRQ2 Enable and Priority Encoding

IRQ2ENH[x]	IRQ2ENL[x]	Priority	Description
0	0	Disabled	Disabled
0	1	Level 1	Low
1	0	Level 2	Nominal
1	1	Level 3	High

where  $x$  indicates the register bits from 0 through 7.



0 = Disable Multiprocessor mode.

1 = Enable Multiprocessor mode.

MPE—Multiprocessor Enable

0 = The UART processes all received data bytes.

1 = The UART processes only data bytes in which the multiprocessor data bit (9th bit) is set to 1.

MPBT—Multiprocessor Bit Transmitter

This bit is applicable only when Multiprocessor (9-bit) mode is enabled.

0 = Send a 0 in the multiprocessor bit location of the data stream (9th bit).

1 = Send a 1 in the multiprocessor bit location of the data stream (9th bit).

Reserved

These bits are reserved and must be 0.

$\overline{\text{RDAIRQ}}$ —Receive Data Interrupt  $\overline{\text{Enable}}$

0 = Received data and receiver errors generates an interrupt request to the Interrupt Controller.

1 = Received data does not generate an interrupt request to the Interrupt Controller. Only receiver errors generate an interrupt request. The associated DMA will still be notified that received data is available.

IREN—Infrared Encoder/Decoder Enable

0 = Infrared Encoder/Decoder is disabled. UART operates normally operation.

1 = Infrared Encoder/Decoder is enabled. The UART transmits and receives data through the Infrared Encoder/Decoder.

## UARTx Baud Rate High and Low Byte Registers

The UARTx Baud Rate High and Low Byte registers (Tables 56 and 57) combine to create a 16-bit baud rate divisor value (BRG[15:0]) that sets the data transmission rate (baud rate) of the UART.

**Table 56. UARTx Baud Rate High Byte Register (UxBRH)**

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	F46H and F4EH							

**Table 58. UART Baud Rates**

**20.0 MHz System Clock**

Desired Rate	BRG Divisor	Actual Rate	Error
(kHz)	(Decimal)	(kHz)	(%)
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	BRG Divisor	Actual Rate	Error
(kHz)	(Decimal)	(kHz)	(%)
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	BRG Divisor	Actual Rate	Error
(kHz)	(Decimal)	(kHz)	(%)
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
1.20	868	1.20	0.01
0.60	1736	0.60	0.01
0.30	3472	0.30	0.01

**11.0592 MHz System Clock**

Desired Rate	BRG Divisor	Actual Rate	Error
(kHz)	(Decimal)	(kHz)	(%)
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
1.20	576	1.20	0.00
0.60	1152	0.60	0.00
0.30	2304	0.30	0.00

## SPI Baud Rate High and Low Byte Registers

The SPI Baud Rate High and Low Byte registers combine to form a 16-bit reload value, BRG[15:0], for the SPI Baud Rate Generator. The reload value must be greater than or equal to 0002H for proper SPI operation (maximum baud rate is system clock frequency divided by 4). The SPI baud rate is calculated using the following equation:

$$\text{SPI Baud Rate (bits/s)} = \frac{\text{System Clock Frequency (Hz)}}{2 \times \text{BRG}[15:0]}$$

**Table 64. SPI Baud Rate High Byte Register (SPIBRH)**

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	F66H							

BRH = SPI Baud Rate High Byte

Most significant byte, BRG[15:8], of the SPI Baud Rate Generator's reload value.

**Table 65. SPI Baud Rate Low Byte Register (SPIBRL)**

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	F67H							

BRL = SPI Baud Rate Low Byte

Least significant byte, BRG[7:0], of the SPI Baud Rate Generator's reload value.

Table 74. DMAx Start/Current Address Low Byte Register (DMAxSTART)

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

DMA\_START—DMAx Start/Current Address Low

These bits, with the four lower bits of the DMAx\_H register, form the 12-bit Start/Current address. The full 12-bit address is given by {DMA\_START\_H[3:0], DMA\_START[7:0]}.

### DMAx End Address Low Byte Register

The DMAx End Address Low Byte register, in conjunction with the DMAx\_H register, forms a 12-bit End Address.

Table 75. DMAx End Address Low Byte Register (DMAxEND)

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

DMA\_END—DMAx End Address Low

These bits, with the four upper bits of the DMAx\_H register, form a 12-bit address. This address is the ending location of the DMAx transfer. The full 12-bit address is given by {DMA\_END\_H[3:0], DMA\_END[7:0]}.

### DMA\_ADC Address Register

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



this bit to 0 when a conversion has been completed.

1 = Begin conversion. Writing a 1 to this bit starts a conversion. If a conversion is already in progress, the conversion restarts. This bit remains 1 until the conversion is complete.

Reserved

This bit is reserved and must be 0.

#### VREF

0 = Internal voltage reference generator enabled. The VREF pin should be left unconnected (or capacitively coupled to analog ground).

1 = Internal voltage reference generator disabled. An external voltage reference must be provided through the VREF pin.

#### CONT

0 = Single-shot conversion. ADC data is output once at completion of the 5129 system clock cycles.

1 = Continuous conversion. ADC data updated every 256 system clock cycles.

#### ANAIN—Analog Input Select

These bits select the analog input for conversion. Not all Port pins in this list are available in all packages for the Z8F640x family of products. Refer to the **Signal and Pin Descriptions** chapter for information regarding the Port pins available with each package style.

Do not enable unavailable analog inputs.

0000 = ANA0

0001 = ANA1

0010 = ANA2

0011 = ANA3

0100 = ANA4

0101 = ANA5

0110 = ANA6

0111 = ANA7

1000 = ANA8

1001 = ANA9

1010 = ANA10

1011 = ANA11

11XX = Reserved.

## 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.



## Program Memory Address 0000H

**Table 90. Option Bits At Program Memory Address 0000H**

BITS	7	6	5	4	3	2	1	0
FIELD	WDT_RES	WDT_AO	Reserved			RP	FHSWP	FWP
RESET	U	U	U	U	U	U	U	U
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	Program Memory 0000H							
Note: U = Unchanged by Reset. R/W = Read/Write.								

### WDT\_RES—Watch-Dog Timer Reset

0 = Watch-Dog Timer time-out generates an interrupt request. Interrupts must be globally enabled for the eZ8 CPU to acknowledge the interrupt request.

1 = Watch-Dog Timer time-out causes a Short Reset. This setting is the default for unprogrammed (erased) Flash.

### WDT\_AO—Watch-Dog Timer Always On

0 = Watch-Dog Timer is automatically enabled upon application of system power. Watch-Dog Timer can not be disabled.

1 = Watch-Dog Timer is enabled upon execution of the WDT instruction. Once enabled, the Watch-Dog Timer can only be disabled by a Reset or Stop Mode Recovery. This setting is the default for unprogrammed (erased) Flash.

### Reserved

These Option Bits are reserved for future use and must always be set to 1. This setting is the default for unprogrammed (erased) Flash.

### RP—Read Protect

0 = User program code is inaccessible. Limited control features are available through the On-Chip Debugger.

1 = User program code is accessible. All On-Chip Debugger commands are enabled. This setting is the default for unprogrammed (erased) Flash.



ister. When the Watchpoint event occurs, the Z8F640x family device enters Debug mode and the DBGMODE bit in the OCDCTL register becomes 1.

## Runtime Counter

The On-Chip Debugger contains a 16-bit Runtime Counter. It counts system clock cycles between Breakpoints. The counter starts counting when the On-Chip Debugger leaves Debug mode and stops counting when it enters Debug mode again or when it reaches the maximum count of FFFFH.

## On-Chip Debugger Commands

The host communicates to the On-Chip Debugger by sending OCD commands using the DBG interface. During normal operation of the Z8F640x family device, only a subset of the OCD commands are available. In Debug mode, all OCD commands become available unless the user code and control registers are protected by programming the Read Protect Option Bit (RP). The Read Protect Option Bit prevents the code in memory from being read out of the Z8F640x family device. When this option is enabled, several of the OCD commands are disabled. Table 93 contains a summary of the On-Chip Debugger commands. Each OCD command is described in further detail in the bulleted list following Table 93. Table 93 indicates those commands that operate when the Z8F640x family device is not in Debug mode (normal operation) and those commands that are disabled by programming the Read Protect Option Bit.

**Table 93. On-Chip Debugger Commands**

Debug Command	Command Byte	Enabled when NOT in Debug mode?	Disabled by Read Protect Option Bit
Read OCD Revision	00H	Yes	-
Reserved	01H	-	-
Read OCD Status Register	02H	Yes	-
Read Runtime Counter	03H	-	-
Write OCD Control Register	04H	Yes	Cannot clear DBGMODE bit
Read OCD Control Register	05H	Yes	-
Write Program Counter	06H	-	Disabled
Read Program Counter	07H	-	Disabled
Write Register	08H	-	Only writes of the Flash Memory Control registers are allowed. Additionally, only the Mass Erase command is allowed to be written to the Flash Control register.
Read Register	09H	-	Disabled



zero). If the Z8F640x family device is not in Debug mode or if the Read Protect Option Bit is enabled, this command returns FFH for all the data values.

```
DBG <-- 09H
DBG <-- {4'h0, Register Address[11:8]}
DBG <-- Register Address[7:0]
DBG <-- Size[7:0]
DBG --> 1-256 data bytes
```

- **Write Program Memory (0AH)**—The Write Program Memory command writes data to Program Memory. This command is equivalent to the LDC and LDCI instructions. Data can be written 1-65536 bytes at a time (65536 bytes can be written by setting size to zero). The on-chip Flash Controller must be written to and unlocked for the programming operation to occur. If the Flash Controller is not unlocked, the data is discarded. If the Z8F640x family device is not in Debug mode or if the Read Protect Option Bit is enabled, the data is discarded.

```
DBG <-- 0AH
DBG <-- Program Memory Address[15:8]
DBG <-- Program Memory Address[7:0]
DBG <-- Size[15:8]
DBG <-- Size[7:0]
DBG <-- 1-65536 data bytes
```

- **Read Program Memory (0BH)**—The Read Program Memory command reads data from Program Memory. This command is equivalent to the LDC and LDCI instructions. Data can be read 1-65536 bytes at a time (65536 bytes can be read by setting size to zero). If the Z8F640x family device is not in Debug mode or if the Read Protect Option Bit is enabled, this command returns FFH for the data.

```
DBG <-- 0BH
DBG <-- Program Memory Address[15:8]
DBG <-- Program Memory Address[7:0]
DBG <-- Size[15:8]
DBG <-- Size[7:0]
DBG --> 1-65536 data bytes
```

- **Write Data Memory (0CH)**—The Write Data Memory command writes data to Data Memory. This command is equivalent to the LDE and LDEI instructions. Data can be written 1-65536 bytes at a time (65536 bytes can be written by setting size to zero). If the Z8F640x family device is not in Debug mode or if the Read Protect Option Bit is enabled, the data is discarded.

```
DBG <-- 0CH
DBG <-- Data Memory Address[15:8]
DBG <-- Data Memory Address[7:0]
DBG <-- Size[15:8]
DBG <-- Size[7:0]
DBG <-- 1-65536 data bytes
```

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

## eZ8 CPU Instruction Classes

eZ8 CPU instructions can be divided functionally into the following groups:

- Arithmetic
- Bit Manipulation
- Block Transfer
- CPU Control
- Load
- Logical
- Program Control
- Rotate and Shift

Tables 118 through 125 contain the instructions belonging to each group and the number of operands required for each instruction. Some instructions appear in more than one table as these instruction can be considered as a subset of more than one category. Within these tables, the source operand is identified as 'src', the destination operand is 'dst' and a condition code is 'cc'.

**Table 118. Arithmetic Instructions**

Mnemonic	Operands	Instruction
ADC	dst, src	Add with Carry
ADCX	dst, src	Add with Carry using Extended Addressing
ADD	dst, src	Add
ADDX	dst, src	Add using Extended Addressing
CP	dst, src	Compare
CPC	dst, src	Compare with Carry
CPCX	dst, src	Compare with Carry using Extended Addressing
CPX	dst, src	Compare using Extended Addressing
DA	dst	Decimal Adjust
DEC	dst	Decrement
DECW	dst	Decrement Word
INC	dst	Increment
INCW	dst	Increment Word
MULT	dst	Multiply



- read watchpoint (21H) 161
- step instruction (10H) 160
- stuff instruction (11H) 160
- write data memory (0CH) 159
- write OCD control register (04H) 158
- write program counter (06H) 158
- write program memory (0AH) 159
- write register (08H) 158
- write watchpoint (20H) 161
- on-chip debugger 5
- on-chip debugger (OCD) 151
- on-chip debugger signals 14
- on-chip oscillator 165
- one-shot mode 70
- opcode map
  - abbreviations 203
  - cell description 202
  - first 204
  - second after 1FH 205
- OR 190
- ordering information 211
- ORX 190
- oscillator signals 14

## P

- p 184
- packaging
  - LQFP
    - 44 lead 207
    - 64 lead 208
  - PDIP 206
  - PLCC
    - 44 lead 207
    - 68 lead 209
  - QFP 210
- part number description 214
- part selection guide 2
- PC 185
- PDIP 206
- peripheral AC and DC electrical characteristics 173
- PHASE=0 timing (SPI) 103
- PHASE=1 timing (SPI) 104
- pin characteristics 15

## PLCC

- 44 lead 207
- 68-lead 209
- polarity 184
- POP 189
- pop using extended addressing 189
- POPX 189
- port availability, device 33
- port input timing (GPIO) 176
- port output timing, GPIO 177
- power supply signals 15
- power-down, automatic (ADC) 133
- power-on and voltage brown-out 173
- power-on reset (POR) 27
- problem description or suggestion 217
- product information 216
- program control instructions 190
- program counter 185
- program memory 18
- PUSH 189
- push using extended addressing 189
- PUSHX 189
- PWM mode 70
- PxADDR register 37
- PxCTL register 38

## Q

- QFP 210

## R

- R 184
- r 184
- RA, register address 184
- RCF 188, 189
- receive
  - 10-bit data format (I2C) 116
  - 7-bit data transfer format (I2C) 115
  - IrDA data 97
- receive interrupt 112
- receiving UART data-DMA controller 83
- receiving UART data-interrupt-driven method 82
- receiving UART data-pollled method 82