



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 "<u>Embedded -</u> <u>Microcontrollers</u>"

Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	60
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	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	80-BQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f4803ft020ec

Email: info@E-XFL.COM

Address: Room A, 16/F, Full Win Commercial Centre, 573 Nathan Road, Mongkok, Hong Kong



Signal Descriptions

Table 2 describes the Z8F640x family signals. Refer to the section **Pin Configurations on page 7** to determine the signals available for the specific package styles.

Signal Mnemonic	I/O	Description							
General-Purpose I/O	General-Purpose I/O Ports A-H								
PA[7:0]	I/O	Port A[7:0]. These pins are used for general-purpose I/O.							
PB[7:0]	I/O	Port B[7:0]. These pins are used for general-purpose I/O.							
PC[7:0]	I/O	Port C[7:0]. These pins are used for general-purpose I/O.							
PD[7:0]	I/O	Port D[7:0]. These pins are used for general-purpose I/O.							
PE[7:0]	I/O	Port E[7:0]. These pins are used for general-purpose I/O.							
PF[7:0]	I/O	Port F[7:0]. These pins are used for general-purpose I/O.							
PG[7:0]	I/O	Port G[7:0]. These pins are used for general-purpose I/O.							
PH[3:0]	I/O	Port H[3:0]. These pins are used for general-purpose I/O.							
I ² C Controller									
SCL	0	Serial Clock. This is the output clock for the I^2C . This pin is multiplexed with a general-purpose I/O pin. When the general-purpose I/O pin is configured for alternate function to enable the SCL function, this pin is open-drain.							
SDA	I/O	Serial Data. This open-drain pin is used to transfer data between the I^2C and a slave. This pin is multiplexed with a general-purpose I/O pin. When the general-purpose I/O pin is configured for alternate function to enable the SDA function, this pin is open-drain.							
SPI Controller									
SS	I/O	Slave Select. This signal can be an output or an input. If the Z8 Encore! is the SPI master, this pin may be configured as the Slave Select output. If the Z8 Encore! is the SPI slave, this pin is the input slave select. It is multiplexed with a general-purpose I/O pin.							
SCK	I/O	SPI Serial Clock. The SPI master supplies this pin. If the Z8 Encore! is the SPI master, this pin is an output. If the Z8 Encore! is the SPI slave, this pin is an input. It is multiplexed with a general-purpose I/O pin.							
MOSI	I/O	Master Out Slave In. This signal is the data output from the SPI master device and the data input to the SPI slave device. It is multiplexed with a general-purpose I/O pin.							
MISO	I/O	Master In Slave Out. This pin is the data input to the SPI master device and the data output from the SPI slave device. It is multiplexed with a general-purpose I/O pin.							

Table 2. Signal Descriptions



Program Memory Address (Hex)	Function
Z8F480x Products	
0000-0001	Flash Option Bits
0002-0003	Reset Vector
0004-0005	WDT Interrupt Vector
0006-0007	Illegal Instruction Trap
0008-0037	Interrupt Vectors*
0038-BFFFH	Program Memory
Z8F640x Products	
0000-0001	Flash Option Bits
0002-0003	Reset Vector
0004-0005	WDT Interrupt Vector
0006-0007	Illegal Instruction Trap
0008-0037	Interrupt Vectors*
0038-FFFFH	Program Memory
* See Table 22 on page 45 for a list of	of the interrupt vectors.

Table 4. Z8F640x Family Program Memory Maps (Continued)

Data Memory

The Z8F640x family devices contain 128 bytes of read-only memory at the top of the eZ8 CPU's 64KB Data Memory address space. The eZ8 CPU's LDE and LDEI instructions provide access to the Data Memory information. Table 5 describes the Z8F640x family's Data Memory Map.

Table 5	. Z8F640x	family	Data	Memory	Maps
---------	-----------	--------	------	--------	------

Data Memory Address (Hex)	Function
0000H-FFBFH	Reserved
FFC0H-FFD3H	Part Number 20-character ASCII alphanumeric code Left justified and filled with zeros
FFD4H-FFFFH	Reserved



Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page #
F62	SPI Status	SPISTAT	01	108
F63	SPI Mode	SPIMODE	00	109
F64-F65	Reserved	—	XX	
F66	SPI Baud Rate High Byte	SPIBRH	FF	110
F67	SPI Baud Rate Low Byte	SPIBRL	FF	110
F68-F69	Reserved	—	XX	
Analog-to-Digit	al Converter (ADC)			
F70	ADC Control	ADCCTL	20	135
F71	Reserved	_	XX	
F72	ADC Data High Byte	ADCD_H	XX	137
F73	ADC Data Low Bits	ADCD_L	XX	137
F74-FAF	Reserved	_	XX	
DMA 0				
FB0	DMA0 Control	DMA0CTL	00	124
FB1	DMA0 I/O Address	DMA0IO	XX	125
FB2	DMA0 End/Start Address High Nibble	DMA0H	XX	126
FB3	DMA0 Start Address Low Byte	DMA0START	XX	127
FB4	DMA0 End Address Low Byte	DMA0END	XX	128
DMA 1	-			
FB8	DMA1 Control	DMA1CTL	00	124
FB9	DMA1 I/O Address	DMA1IO	XX	125
FBA	DMA1 End/Start Address High Nibble	DMA1H	XX	126
FBB	DMA1 Start Address Low Byte	DMA1START	XX	127
FBC	DMA1 End Address Low Byte	DMA1END	XX	128
DMA ADC	· · · · · · · · · · · · · · · · · · ·			
FBD	DMA_ADC Address	DMAA_ADDR	XX	128
FBE	DMA_ADC Control	DMAACTL	00	130
FBF	DMA_ADC Status	DMAASTAT	00	131
Interrupt Conti	oller			
FC0	Interrupt Request 0	IRQ0	00	48
FC1	IRQ0 Enable High Bit	IRQOENH	00	51
FC2	IRQ0 Enable Low Bit	IRQ0ENL	00	51
FC3	Interrupt Request 1	IRQ1	00	49
FC4	IRQ1 Enable High Bit	IRQ1ENH	00	52
FC5	IRQ1 Enable Low Bit	IRQ1ENL	00	52
FC6	Interrupt Request 2	IRQ2	00	50
FC7	IRQ2 Enable High Bit	IRQ2ENH	00	53
FC8	IRQ2 Enable Low Bit	IRQ2ENL	00	53
FC9-FCC	Reserved	_	XX	
FCD	Interrupt Edge Select	IRQES	00	54
XX=Undefined				

Table 6. Register File Address Map (Continued)



Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page #
FCE	Interrupt Port Select	IRQPS	00	55
FCF	Interrupt Control	IRQCTL	00	56
GPIO Port A				
FD0	Port A Address	PAADDR	00	37
FD1	Port A Control	PACTL	00	38
FD2	Port A Input Data	PAIN	XX	42
FD3	Port A Output Data	PAOUT	00	43
GPIO Port B				
FD4	Port B Address	PBADDR	00	37
FD5	Port B Control	PBCTL	00	38
FD6	Port B Input Data	PBIN	XX	42
FD7	Port B Output Data	PBOUT	00	43
GPIO Port C	*			
FD8	Port C Address	PCADDR	00	37
FD9	Port C Control	PCCTL	00	38
FDA	Port C Input Data	PCIN	XX	42
FDB	Port C Output Data	PCOUT	00	43
GPIO Port D	*			
FDC	Port D Address	PDADDR	00	37
FDD	Port D Control	PDCTL	00	38
FDE	Port D Input Data	PDIN	XX	42
FDF	Port D Output Data	PDOUT	00	43
GPIO Port E				
FE0	Port E Address	PEADDR	00	37
FE1	Port E Control	PECTL	00	38
FE2	Port E Input Data	PEIN	XX	42
FE3	Port E Output Data	PEOUT	00	43
GPIO Port F				
FE4	Port F Address	PFADDR	00	37
FE5	Port F Control	PFCTL	00	38
FE6	Port F Input Data	PFIN	XX	42
FE7	Port F Output Data	PFOUT	00	43
GPIO Port G				
FE8	Port G Address	PGADDR	00	37
FE9	Port G Control	PGCTL	00	38
FEA	Port G Input Data	PGIN	XX	42
FEB	Port G Output Data	PGOUT	00	43
GPIO Port H				
FEC	Port H Address	PHADDR	00	37
XX-Undefined				

Table 6. Register File Address Map (Continued)



IRQ0 Enable High and Low Bit Registers

The IRQ0 Enable High and Low Bit registers (Tables 27 and 28) form a priority encoded enabling for interrupts in the Interrupt Request 0 register. Priority is generated by setting bits in each register. Table 26 describes the priority control for IRQ0.

IRQ0ENH[x]	IRQ0ENL[x]	Priority	Description
0	0	Disabled	Disabled
0	1	Level 1	Low
1	0	Level 2	Nominal
1	1	Level 3	High

Table 26. IRQ0 Enable and Priority Encoding

where x indicates the register bits from 0 through 7.

Table 27. IRQ0 Enable High Bit Register (IRQ0ENH)

BITS	7	6	5	4	3	2	1	0
FIELD	T2ENH	T1ENH	T0ENH	UORENH	U0TENH	I2CENH	SPIENH	ADCENH
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	FC1H							

T2ENH—Timer 2 Interrupt Request Enable High Bit T1ENH—Timer 1 Interrupt Request Enable High Bit T0ENH—Timer 0 Interrupt Request Enable High Bit U0RENH—UART 0 Receive Interrupt Request Enable High Bit U0TENH—UART 0 Transmit Interrupt Request Enable High Bit I2CENH—I²C Interrupt Request Enable High Bit SPIENH—SPI Interrupt Request Enable High Bit ADCENH—ADC Interrupt Request Enable High Bit



Interrupt Control Register

The Interrupt Control (IRQCTL) register (Table 37) contains the master enable bit for all interrupts.

Table 37. Interrupt Control Register (IRQCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	IRQE	Reserved						
RESET	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R
ADDR	FCFH							

IRQE—Interrupt Request Enable

This bit is set to 1 by execution of an EI (Enable Interrupts) or IRET (Interrupt Return) instruction, or by a direct register write of a 1 to this bit. It is reset to 0 by executing a DI instruction, eZ8 CPU acknowledgement of an interrupt request, or Reset.

0 = Interrupts are disabled.

1 = Interrupts are enabled.

Reserved

These bits must be 0.



set to 2-byte transfers, the temporary holding register for the Timer Reload High Byte is not bypassed.

BITS 7 6 5 4 3 2 1 0 TRH FIELD 1 1 1 1 1 1 1 1 RESET R/W R/W R/W R/W R/W R/W R/W R/W R/W F02H, F0AH, F12H, F1AH ADDR

Table 40. Timer 0-3 Reload High Byte Register (TxRH)

Table 41. Timer 0-3 Reload Low Byte Register (TxRL)

BITS	7	6	5	4	3	2	1	0
FIELD	TRL							
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	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 is used to set the maximum count value which initiates a timer reload to 0001H. In Compare mode, these two byte form the 16-bit Compare value.



Serial Peripheral Interface

Overview

The Serial Peripheral Interface[™] (SPI) is a synchronous interface allowing several SPItype devices to be interconnected. SPI-compatible devices include EEPROMs, Analog-to-Digital Converters, and ISDN devices. Features of the SPI include:

- Full-duplex, synchronous, character-oriented communication
- Four-wire interface
- Data transfers rates up to a maximum of one-fourth the system clock frequency
- Error detection
- Write and mode collision detection
- Dedicated Baud Rate Generator

Architecture

The SPI may be configured as either a Master (in single or multi-master systems) or a Slave as illustrated in Figures 74 through 76.



Figure 74. SPI Configured as a Master in a Single Master, Single Slave System



104



Figure 78. SPI Timing When PHASE is 1

Multi-Master Operation

In a multi-master SPI system, all SCK pins are tied together, all MOSI pins are tied together and all MISO pins are tied together. All SPI pins must then be configured in open-drain mode to prevent bus contention. At any one time, only one SPI device is configured as the Master and all other SPI devices on the bus are configured as Slaves. The Master enables a single Slave by asserting the SS pin on that Slave only. Then, the single Master drives data out its SCK and MOSI pins to the SCK and MOSI pins on the Slaves (including those which are not enabled). The enabled Slave drives data out its MISO pin to the MISO Master pin.

For a Master device operating in a multi-master system, if the \overline{SS} pin is configured as an input and is driven Low by another Master, the COLbit is set to 1 in the SPI Status Register. The COLbit indicates the occurrence of a multi-master collision (mode fault error condition).



I²C Control Register Definitions

I²C Data Register

The I²C Data register holds the data that is to be loaded into the I²C Shift register during a write to a slave. This register also holds data that is loaded from the I²C Shift register during a read from a slave. The I²C Shift is not accessible in the Register File address space, but is used only to buffer incoming and outgoing data.

 Table 66. I²C Data Register (I2CDATA)

BITS	7	6	5	4	3	2	1	0
FIELD	DATA							
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	F50H							

I²C Status Register

The Read-only I²C Status register indicates the status of the I²C Controller.

BITS	7	6	5	4	3	2	1	0
FIELD	TDRE	RDRF	ACK	10B	RD	TAS	DSS	NCKI
RESET	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
ADDR				F5	1H			

Table 67. I²C Status Register (I2CSTAT)

TDRE—Transmit Data Register Empty

When the I²C Controller is enabled, this bit is 1 when the I²C Data register is empty. When active, this bit causes the I²C Controller to generate an interrupt, except when the I²C Controller is shifting in data during the reception of a byte or when shifting an address and the RDbit is set. This bit and the interrupt are cleared by writing to the I²CD register.

RDRF—Receive Data Register Full

This bit is set active high when the I²C Controller is enabled and the I²C Controller has



Configuring DMA0 and DMA1 for Data Transfer

Follow these steps to configure and enable DMA0 or DMA1:

- Write to the DMAx I/O Address register to set the Register File address identifying the on-chip peripheral control register. The upper nibble of the 12-bit address for on-chip peripheral control registers is always FH. The full address is {FH, DMAx_IO[7:0]}
- Determine the 12-bit Start and End Register File addresses. The 12-bit Start Address is given by {DMAx_H[3:0], DMA_START[7:0]}. The 12-bit End Address is given by {DMAx_H[7:4], DMA_END[7:0]}.
- 3. Write the Start and End Register File address high nibbles to the DMAx End/Start Address High Nibble register.
- 4. Write the lower byte of the Start Address to the DMAx Start/Current Address register.
- 5. Write the lower byte of the End Address to the DMAx End Address register.
- 6. Write to the DMAx Control register to complete the following:
 - Select loop or single-pass mode operation
 - Select the data transfer direction (either from the Register File RAM to the onchip peripheral control register; or from the on-chip peripheral control register to the Register File RAM)
 - Enable the DMAx interrupt request, if desired
 - Select Word or Byte mode
 - Select the DMAx request trigger
 - Enable the DMAx channel

DMA_ADC Operation

DMA_ADC transfers data from the ADC to the Register File. The sequence of operations in a DMA_ADC data transfer is:

- 1. ADC completes conversion on the current ADC input channel and signals the DMA controller that two-bytes of ADC data are ready for transfer.
- 2. DMA_ADC requests control of the system bus (address and data) from the eZ8 CPU.
- 3. After the eZ8 CPU acknowledges the bus request, DMA_ADC transfers the two-byte ADC output value to the Register File and then returns system bus control back to the eZ8 CPU.
- 4. If the current ADC Analog Input is the highest numbered input to be converted:
 - DMA_ADC resets the ADC Analog Input number to 0 and initiates data conversion on ADC Analog Input 0.
 - If configured to generate an interrupt, DMA_ADC sends an interrupt request to the Interrupt Controller



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

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

Table 76. DMA_ADC Register File Address Example

¹ DMAA_ADDR set to 72H.

Table 77. DMA_ADC Address Register (DMAA_ADDR)

BITS	7	6	5	4	3	2	1	0
FIELD			Ľ	MAA_ADD	R			Reserved
RESET	Х	Х	Х	Х	Х	Х	Х	Х
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR				FB	DH			

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.



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.

138

Flash Memory

Overview

The Z8F640x family features up to 64KB (65,536 bytes) of non-volatile Flash memory with read/write/erase capability. The Flash Memory can be programmed and erased in-circuit by either user code or through the On-Chip Debugger.

The Flash memory array is arranged in pages with 512 bytes per page. The 512-byte page is the minimum Flash block size that can be erased. Each page is divided into 8 rows of 64 bytes. The Flash memory also contains a High Sector that can be enabled for writes and erase separately from the rest of the Flash array. The first 2 bytes of the Flash Program memory are used as Option Bits. Refer to the Option Bits chapter for more information on their operation.

Table 83 describes the Flash memory configuration for each device in the Z8F640x family. Figure 84 illustrates the Flash memory arrangement.

Part Number	Flash Size KB (Bytes)	Flash Pages	Program Memory Addresses	Flash High Sector Size KB (Bytes)	High Sector Addresses
Z8F160x	16 (16,384)	32	0000H - 3FFFH	1 (1024)	3C00H - 3FFFH
Z8F240x	24 (24,576)	48	0000H - 5FFFH	2 (2048)	5800H - 5FFFH
Z8F320x	32 (32,768)	64	0000H - 7FFFH	2 (2048)	7800H - 7FFFH
Z8F480x	48 (49,152)	96	0000H - BFFFH	4 (4096)	B000H - BFFFH
Z8F640x	64 (65,536)	128	0000H - FFFFH	8 (8192)	E000H - FFFFH

Table 83. Z8F640x family Flash Memory Configurations



Flash Status Register

The Flash Status register indicates the current state of the Flash Controller. This register can be read at any time. The Read-only Flash Status Register shares its Register File address with the Write-only Flash Control Register.

Table	86.	Flash	Status	Register	(FSTAT)
-------	-----	-------	--------	----------	---------

BITS	7	6	5	4	3	2	1	0
FIELD	Rese	erved	FSTAT					
RESET	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
ADDR				FF	'8H			

Reserved

These bits are reserved and must be 0.

FSTAT—Flash Controller Status

000000 = Flash Controller locked.

000001 = First unlock command received.

000010 = Flash Controller unlocked (second unlock command received).

001xxx = Program operation in progress.

010xxx = Page erase operation in progress.

100xxx = Mass erase operation in progress.



146

Flash Page Select Register

The Flash Page Select register is used to select one of the 128 available Flash memory pages to be erased in a Page Erase operation. Each Flash Page contains 512 bytes of Flash memory. During a Page Erase operation, all Flash memory having addresses with the most significant 7-bits given by FPS[6:0] are erased (all bytes written to FFH).

Table 87. Flash Page Select Register (FPS)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved				PAGE			
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				FF	9H			

Reserved This bit is reserved and must be 0.

PAGE—Page Select

This 7-bit field identifies the Flash memory page for Page Erase operation. Program Memory Address[15:9] = PAGE[6:0]



156

ister. When the Watchpoint event occurs, the Z8F640x family device enters Debug mode and the DBGMODE 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.

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 DBGMODEt
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

Table 93. On-Chip Debugger Commands



Customer Feedback Form

The Z8 Encore!™ Product Specification

If you experience any problems while operating this product, or if you note any inaccuracies while reading this Product Specification, please copy and complete this form, then mail or fax it to ZiLOG (see Return Information below). We also welcome your suggestions!

Customer Information

Name	Country
Company	Phone
Address	Fax
City/State/Zip	E-Mail

Product Information

Part #, Serial #, Board Fab #, or Rev. #
Software Version
Document Number
Host Computer Description/Type

Return Information

ZiLOG 532 Race Street San Jose, CA 95126 Fax: (408) 558-8536 Email: tools@zilog.com



Problem Description or Suggestion

Provide a complete description of the problem or your suggestion. If you are reporting a specific problem, include all steps leading up to the occurrence of the problem. Attach additional pages as necessary.

Z8F640x/Z8F480x/Z8F320x/Z8F240x/Z8F160x Z8 Encore!®



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

F

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

G

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

H

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

I

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