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	Active
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	24KB (24K 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/z8f2422ar020sg

Table 177. UART Baud Rate High Byte Register (UxBRH) 258
Table 178. UART Baud Rate Low Byte Register (UxBRL) 259
Table 179. UART Transmit Data Register (UxTXD) 259
Table 180. UART Receive Data Register (UxRXD) 259
Table 181. UART Status 0 Register (UxSTAT0) 259
Table 182. UART Control 0 Register (UxCTL0) 260
Table 183. UART Control 1 Register (UxCTL1) 260
Table 184. UART Status 1 Register (UxSTAT1) 260
Table 185. UART Address Compare Register (UxADDR) 260
Table 186. UART Baud Rate High Byte Register (UxBRH) 261
Table 187. UART Baud Rate Low Byte Register (UxBRL) 261
Table 188. I^2C Data Register (I2CDATA) 261
Table 189. I^2C Status Register (I2CSTAT) 262
Table 190. I^2C Control Register (I2CCTL) 262
Table 191. I^2C Baud Rate High Byte Register (I2CBRH) 262
Table 192. I^2C Baud Rate Low Byte Register (I2CBRL) 262
Table 193. I^2C Diagnostic State Register (I2CDST) 263
Table 194. I^2C Diagnostic Control Register (I2CDIAG) 263
Table 195. SPI Data Register (SPIDATA) 263
Table 196. SPI Control Register (SPICTL) 264
Table 197. SPI Status Register (SPISTAT) 264
Table 198. SPI Mode Register (SPIMODE) 264
Table 199. SPI Diagnostic State Register (SPIDST) 265
Table 200. SPI Baud Rate High Byte Register (SPIBRH) 265
Table 201. SPI Baud Rate Low Byte Register (SPIBRL) 265
Table 202. ADC Data High Byte Register (ADCD_H) 266
Table 203. ADC Data Low Bits Register (ADCD_L) 266
Table 204. DMAx Control Register (DMAxCTL) 267
Table 205. DMAx I/O Address Register (DMAxIO) 267
Table 206. DMAx Address High Nibble Register (DMAxH) 267
Table 207. DMAx Start/Current Address Low Byte Register (DMAxSTART) 267
Table 208. DMAx End Address Low Byte Register (DMAxEND) 268
Table 209. DMAx Control Register (DMAxCTL) 268
Table 210. DMAx I/O Address Register (DMAxIO) 268
Table 211. DMAx Address High Nibble Register (DMAxH) 269
Table 212. DMAx Start/Current Address Low Byte Register (DMAxSTART) 269

Table 7. Z8 Encore! XP F64xx Series Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page
GPIO Port A (continued)				
FD3	Port A Output Data	PAOUT	00	46
GPIO Port B				
FD4	Port B Address	PBADDR	00	40
FD5	Port B Control	PBCTL	00	41
FD6	Port B Input Data	PBIN	XX	46
FD7	Port B Output Data	PBOUT	00	46
GPIO Port C				
FD8	Port C Address	PCADDR	00	40
FD9	Port C Control	PCCTL	00	41
FDA	Port C Input Data	PCIN	XX	46
FDB	Port C Output Data	PCOUT	00	46
GPIO Port D				
FDC	Port D Address	PDADDR	00	40
FDD	Port D Control	PDCTL	00	41
FDE	Port D Input Data	PDIN	XX	46
FDF	Port D Output Data	PDOUT	00	46
GPIO Port E				
FE0	Port E Address	PEADDR	00	40
FE1	Port E Control	PECTL	00	41
FE2	Port E Input Data	PEIN	XX	46
FE3	Port E Output Data	PEOUT	00	46
GPIO Port F				
FE4	Port F Address	PFADDR	00	40
FE5	Port F Control	PFCTL	00	41
FE6	Port F Input Data	PFIN	XX	46
FE7	Port F Output Data	PFOUT	00	46
GPIO Port G				
FE8	Port G Address	PGADDR	00	40
FE9	Port G Control	PGCTL	00	41
FEA	Port G Input Data	PGIN	XX	46
FEB	Port G Output Data	PGOUT	00	46

Note: XX = Undefined.

System Reset

During a system reset, the Z8 Encore! XP F64xx Series devices are held in Reset for 66 cycles of the Watchdog Timer oscillator followed by 16 cycles of the system clock. At the beginning of Reset, all GPIO pins are configured as inputs.

During Reset, the eZ8 CPU and on-chip peripherals are idle; however, the on-chip crystal oscillator and Watchdog Timer oscillator continue to run. The system clock begins operating following the Watchdog Timer oscillator cycle count. The eZ8 CPU and on-chip peripherals remain idle through the 16 cycles of the system clock.

Upon Reset, control registers within the Register File that have a defined Reset value are loaded with their reset values. Other control registers (including the Stack Pointer, Register Pointer, and Flags) and general-purpose RAM are undefined following Reset. The eZ8 CPU fetches the Reset vector at program memory addresses 0002H and 0003H and loads that value into the Program Counter. Program execution begins at the Reset vector address.

Reset Sources

Table 9 lists the reset sources as a function of the operating mode. The text following provides more detailed information about the individual Reset sources. A Power-On Reset/Voltage Brown-Out event always takes priority over all other possible reset sources to ensure a full system reset occurs.

Table 9. Reset Sources and Resulting Reset Type

Operating Mode	Reset Source	Reset Type
NORMAL or HALT modes	Power-On Reset/Voltage Brown-Out	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 Brown-Out	system reset
	$\overline{\text{RESET}}$ pin assertion	system reset
	DBG pin driven Low	system reset

Architecture

Figure 10 displays a simplified block diagram of a GPIO port pin. In this figure, the ability to accommodate alternate functions and variable port current drive strength are not illustrated.

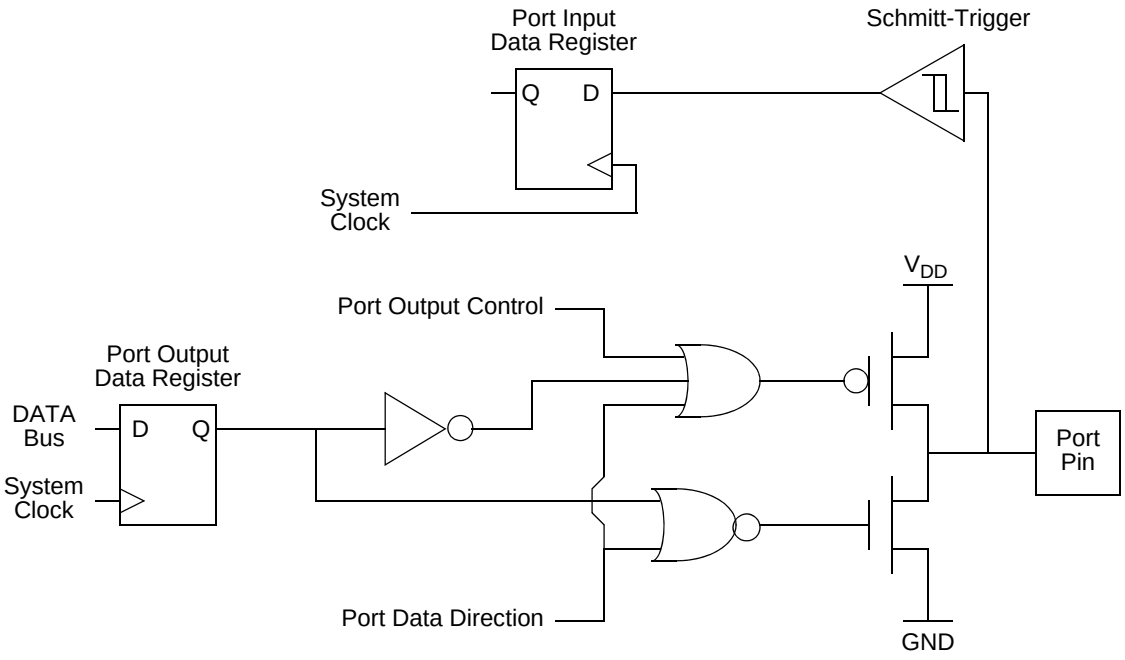

Figure 10. GPIO Port Pin Block Diagram

GPIO Alternate Functions

Many of the GPIO port pins can be used as both general-purpose I/O and to provide access to on-chip peripheral functions such as the timers and serial communication devices. The Port A–H Alternate Function subregisters configure these pins for either general-purpose I/O or alternate function operation. When a pin is configured for alternate function, control of the port pin direction (input/output) is passed from the Port A–H Data Direction registers to the alternate function assigned to this pin. Table 12 lists the alternate functions associated with each port pin.

Interrupt Controller

The interrupt controller on the Z8 Encore! XP F64xx Series products prioritizes the interrupt requests from the on-chip peripherals and the GPIO port pins. The features of the interrupt controller include:

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

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

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

Interrupt Vector Listing

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

unlock the Watchdog Timer Reload Byte registers (WDTU, WDTH, and WDTL) for write access.

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

All steps of the Watchdog Timer reload unlock sequence must be written in the sequence described above; there must be no other register writes between each of these operations. If a register write occurs, the lock state machine resets and no further writes can occur, unless the sequence is restarted. The value in the Watchdog Timer Reload registers is loaded into the counter when the Watchdog Timer is first enabled and every time a WDT instruction is executed.

Watchdog Timer Control Register Definitions

This section defines the features of the following Watchdog Timer Control registers.

Watchdog Timer Control Register: see page 83

Watchdog Timer Reload Upper, High and Low Byte Registers: see page 85

Watchdog Timer Control Register

The Watchdog Timer Control (WDTCTL) Register, shown in Table 48, is a read-only register that indicates the source of the most recent Reset event, indicates a Stop Mode Recovery event, and indicates a Watchdog Timer time-out. Reading this register resets the upper four bits to 0.

Writing the 55H, AAH unlock sequence to the Watchdog Timer Control (WDTCTL) Register address unlocks the three Watchdog Timer Reload Byte registers (WDTU, WDTH, and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL Register address produce no effect on the bits in the WDTCTL Register. The locking mechanism prevents spurious writes to the Reload registers.

Table 48. Watchdog Timer Control Register (WDTCTL)

Bit	7	6	5	4	3	2	1	0
Field	POR	STOP	WDT	EXT	Reserved			SM
RESET	See Table 49.			0				
R/W	R							
Address	FF0H							

Bit	Description
[7] POR	**Power-On Reset Indicator** If this bit is set to 1, a Power-On Reset event occurred. This bit is reset to 0 if a WDT time-out or Stop Mode Recovery occurs. This bit is also reset to 0 when the register is read.
[6] STOP	**Stop Mode Recovery Indicator** If this bit is set to 1, a Stop Mode Recovery occurred. If the stop and WDT bits are both set to 1, the Stop Mode Recovery occurred due to a WDT time-out. If the stop bit is 1 and the WDT bit is 0, the Stop Mode Recovery was not caused by a WDT time-out. This bit is reset by a Power-On Reset or a WDT time-out that occurred while not in STOP Mode. Reading this register also resets this bit.
[5] WDT	**Watchdog Timer Time-Out Indicator** If this bit is set to 1, a WDT time-out occurred. A Power-On Reset resets this pin. A Stop Mode Recovery from a change in an input pin also resets this bit. Reading this register resets this bit.
[4] EXT	**External Reset Indicator** If this bit is set to 1, a Reset initiated by the external $\overline{\text{RESET}}$ pin occurred. A Power-On Reset or a Stop Mode Recovery from a change in an input pin resets this bit. Reading this register resets this bit.
[3:1]	**Reserved** These bits are reserved and must be programmed to 000.
[0] SM	**STOP Mode Configuration Indicator** 0 = Watchdog Timer and its internal RC oscillator will continue to operate in STOP Mode. 1 = Watchdog Timer and its internal RC oscillator will be disabled in STOP Mode.

Table 51. Watchdog Timer Reload High Byte Register (WDTH)

Bit	7	6	5	4	3	2	1	0
Field	WDTH							
RESET	1							
R/W	R/W*							
Address	FF2H							
Note: *R/W = Read returns the current WDT count value; write sets the appropriate reload value.								

Bit	Description
[7:0] WDTH	**WDT Reload High Byte** Middle byte, bits[15:8] of the 24-bit WDT reload value.

Table 52. Watchdog Timer Reload Low Byte Register (WDTL)

Bit	7	6	5	4	3	2	1	0
Field	WDTL							
RESET	1							
R/W	R/W*							
Address	FF3H							
Note: *R/W = Read returns the current WDT count value; write sets the appropriate reload value.								

Bit	Description
[7:0] WDTL	**WDT Reload Low** Least significant byte, bits[7:0] of the 24-bit WDT reload value.

UART Interrupts

The UART features separate interrupts for the transmitter and the receiver. In addition, when the UART primary functionality is disabled, the Baud Rate Generator can also function as a basic timer with interrupt capability.

Transmitter Interrupts

The transmitter generates a single interrupt when the Transmit Data Register Empty bit (TDRE) is set to 1. This indicates that the transmitter is ready to accept new data for transmission. The TDRE interrupt occurs after the Transmit Shift Register has shifted the first bit of data out. At this point, the Transmit Data Register can be written with the next character to send. This provides 7 bit-periods of latency to load the Transmit Data Register before the Transmit Shift Register completes shifting the current character. Writing to the UART Transmit Data Register clears the TDRE bit to 0.

Receiver Interrupts

The receiver generates an interrupt when any of the following events occurs:

- A data byte has been received and is available in the UART Receive Data Register. This interrupt can be disabled independent of the other receiver interrupt sources. The received data interrupt occurs once the receive character has been received and placed in the Receive Data Register. Software must respond to this received data available condition before the next character is completely received to avoid an overrun error.

Note: In MULTIPROCESSOR Mode (MPEN = 1), the receive data interrupts are dependent on the multiprocessor configuration and the most recent address byte.

- A break is received
- An overrun is detected
- A data framing error is detected

UART Overrun Errors

When an overrun error condition occurs the UART prevents overwriting of the valid data currently in the Receive Data Register. The Break Detect and Overrun status bits are not displayed until after the valid data has been read.

After the valid data has been read, the UART Status 0 Register is updated to indicate the overrun condition (and Break Detect, if applicable). The RDA bit is set to 1 to indicate that the Receive Data Register contains a data byte. However, because the overrun error occurred, this byte may not contain valid data and should be ignored. The BRKD bit indicates if the overrun was caused by a break condition on the line. After reading the status

Bit	Description (Continued)
[2] 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.
[1] 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.
[0] CTS	**$\overline{\text{CTS}}$ Signal** When this bit is read, it returns the level of the $\overline{\text{CTS}}$ signal.

UART Status 1 Register

The UART Status 1 Register, shown in Table 56, contains multiprocessor control and UART status bits.

Table 56. UART Status 1 Register (U*x*STAT1)

Bit	7	6	5	4	3	2	1	0
Field	Reserved						NEWFRM	MPRX
RESET	0							
R/W	R				R/W		R	
Address	F44H and F4CH							

Bit	Description
[7:2]	**Reserved** These bits are reserved and must be programmed to 000000.
[1] NEWFRM	**New Frame** Status bit denoting the start of a new frame. Reading the UART Receive Data Register resets this bit to 0. 0 = The current byte is not the first data byte of a new frame. 1 = The current byte is the first data byte of a new frame.
[0] MPRX	**Multiprocessor Receive** Returns the value of the last multiprocessor bit received. Reading from the UART Receive Data Register resets this bit to 0.

Table 69. SPI Baud Rate High Byte Register (SPIBRH)

Bit	7	6	5	4	3	2	1	0
Field	BRH							
RESET	1							
R/W	R/W							
Address	F66H							

Bit	Description
[7:0] BRH	**SPI Baud Rate High Byte** Most significant byte, BRG[15:8], of the SPI Baud Rate Generator's reload value.

Table 70. SPI Baud Rate Low Byte Register (SPIBRL)

Bit	7	6	5	4	3	2	1	0
Field	BRL							
RESET	1							
R/W	R/W							
Address	F67H							

Bit	Description
[7:0] BRL	**SPI Baud Rate Low Byte** Least significant byte, BRG[7:0], of the SPI Baud Rate Generator's reload value.

5. After the first bit has been shifted out, a transmit interrupt is asserted.
6. Software responds by writing the lower eight bits of address to the I^2C Data Register.
7. The I^2C Controller completes shifting of the two address bits and a 0 (write).
8. If the I^2C slave acknowledges the first address byte by pulling the SDA signal Low during the next High period of SCL, the I^2C Controller sets the ACK bit in the I^2C Status Register. Continue with Step 9.

   If the slave does not acknowledge the first address byte, the I^2C Controller sets the NCKI bit and clears the ACK bit in the I^2C Status Register. Software responds to the Not Acknowledge interrupt by setting the stop and flush bits and clearing the TXI bit. The I^2C Controller sends the stop condition on the bus and clears the stop and NCKI bits. The transaction is complete (ignore following steps).
9. The I^2C Controller loads the I^2C Shift Register with the contents of the I^2C Data Register (second address byte).
10. The I^2C Controller shifts out the second address byte. After the first bit is shifted, the I^2C Controller generates a transmit interrupt.
11. Software responds by setting the start bit of the I^2C Control Register to generate a repeated start by clearing the TXI bit.
12. Software responds by writing `11110B` followed by the 2-bit slave address and a 1 (read) to the I^2C Data Register.
13. If only one byte is to be read, software sets the NAK bit of the I^2C Control Register.
14. After the I^2C Controller shifts out the 2nd address byte, the I^2C slave sends an acknowledge by pulling the SDA signal Low during the next High period of SCL, the I^2C Controller sets the ACK bit in the I^2C Status Register. Continue with Step 15.

    If the slave does not acknowledge the second address byte, the I^2C Controller sets the NCKI bit and clears the ACK bit in the I^2C Status Register. Software responds to the Not Acknowledge interrupt by setting the stop and flush bits and clearing the TXI bit. The I^2C Controller sends the stop condition on the bus and clears the stop and NCKI bits. The transaction is complete (ignore the following steps).
15. The I^2C Controller sends the repeated start condition.
16. The I^2C Controller loads the I^2C Shift Register with the contents of the I^2C Data Register (third address transfer).
17. The I^2C Controller sends `11110B` followed by the two most significant bits of the slave read address and a 1 (read).
18. The I^2C slave sends an acknowledge by pulling the SDA signal Low during the next High period of SCL

Configuring DMA_ADC for Data Transfer

Observe the following procedure to configure and enable the DMA_ADC:

1. Write the DMA_ADC Address Register with the 7 most significant bits of the Register File address for data transfers.
2. Write to the DMA_ADC Control Register to complete the following operations:
   - Enable the DMA_ADC interrupt request, if appropriate
   - Select the number of ADC analog inputs to convert
   - Enable the DMA_ADC channel

Caution: When using the DMA_ADC to perform conversions on multiple ADC inputs, the Analog-to-Digital Converter must be configured for SINGLE-SHOT Mode. If the ADC_IN field in the DMA_ADC Control Register is greater than 000b, the ADC must be in SINGLE-SHOT Mode.

CONTINUOUS Mode operation of the ADC can only be used in conjunction with the DMA_ADC if the ADC_IN field in the DMA_ADC Control Register is reset to 000b to enable conversion on ADC analog input 0 only.

DMA Control Register Definitions

This section defines the features of the following DMA Control registers.

DMAx Control Register: see page 153

DMAx I/O Address Register: see page 154

DMAx Address High Nibble Register: see page 155

DMAx Start/Current Address Low Byte Register: see page 156

DMAx End Address Low Byte Register: see page 156

DMA_ADC Address Register: see page 157

DMA_ADC Control Register: see page 158

DMA_ADC Status Register: see page 159

Flash Memory

The products in the Z8 Encore! XP F64xx Series feature 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 512 byte per page. The 512 byte page is the minimum Flash block size that can be erased. The Flash memory is also divided into 8 sectors which can be protected from programming and erase operations on a per sector basis.

Table 90 describes the Flash memory configuration for each device in the Z8 Encore! XP F64xx Series. Table 91 lists the sector address ranges. Figure 35 displays the Flash memory arrangement.

Table 90. Flash Memory Configurations

Part Number	Flash Size	Number of Pages	Flash Memory Addresses	Sector Size	Number of Sectors	Pages per Sector
Z8F162x	16K (16,384)	32	0000H–3FFFH	2K (2048)	8	4
Z8F242x	24K (24,576)	48	0000H–5FFFH	4K (4096)	6	8
Z8F322x	32K (32,768)	64	0000H–7FFFH	4K (4096)	8	8
Z8F482x	48K (49,152)	96	0000H–BFFFH	8K (8192)	6	16
Z8F642x	64K (65,536)	128	0000H–FFFFH	8K (8192)	8	16

Table 91. Flash Memory Sector Addresses

Sector Number	Flash Sector Address Ranges				
	Z8F162x	Z8F242x	Z8F322x	Z8F482x	Z8F642x
0	0000H–07FFH	0000H–0FFFH	0000H–0FFFH	0000H–1FFFH	0000H–1FFFH
1	0800H–0FFFH	1000H–1FFFH	1000H–1FFFH	2000H–3FFFH	2000H–3FFFH
2	1000H–17FFH	2000H–2FFFH	2000H–2FFFH	4000H–5FFFH	4000H–5FFFH
3	1800H–1FFFH	3000H–3FFFH	3000H–3FFFH	6000H–7FFFH	6000H–7FFFH
4	2000H–27FFH	4000H–4FFFH	4000H–4FFFH	8000H–9FFFH	8000H–9FFFH
5	2800H–2FFFH	5000H–5FFFH	5000H–5FFFH	A000H–BFFFH	A000H–BFFFH
6	3000H–37FFH	N/A	6000H–6FFFH	N/A	C000H–DFFFH
7	3800H–3FFFH	N/A	7000H–7FFFH	N/A	E000H–FFFFH

Table 92. Z8 Encore! XP F64xx Series Information Area Map

Flash Memory Address (Hex)	Function
FE00H–FE3FH	Reserved
FE40H–FE53H	Part Number 20-character ASCII alphanumeric code Left-justified and filled with zeros
FE54H–FFFFH	Reserved

Operation

The Flash Controller provides the proper signals and timing for the Byte Programming, Page Erase, and Mass Erase operations within Flash memory. The Flash Controller contains a protection mechanism, via the Flash Control Register (FCTL), to prevent accidental programming or erasure. The following subsections provide details about the Lock, Unlock, Sector Protect, Byte Programming, Page Erase and Mass Erase operations.

Timing Using the Flash Frequency Registers

Before performing a program or erase operation in 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 20kHz through 20MHz (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 20kHz, above 20MHz, or outside of the devices' operating frequency range. The Flash Frequency High and Low Byte registers must be loaded with the correct value to ensure proper Flash programming and erase operations.

Electrical Characteristics

The data in this chapter represents all known data prior to qualification and characterization of the Z8 Encore! XP F64xx Series of products, and is therefore subject to change. Additional electrical characteristics may be found in the individual chapters of this document.

Absolute Maximum Ratings

Stresses greater than those listed in Table 106 may cause permanent damage to the device. These ratings are stress ratings only. Operation of the device at any condition outside those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. For improved reliability, unused inputs must be tied to one of the supply voltages (V_{DD} or V_{SS}).

Table 106. Absolute Maximum Ratings

Parameter	Minimum	Maximum	Units	Notes
Ambient temperature under bias	–40	+125	C	
Storage temperature	–65	+150	C	
Voltage on any pin with respect to V_{SS}	–0.3	+5.5	V	1
Voltage on V_{DD} pin with respect to V_{SS}	–0.3	+3.6	V	
Maximum current on input and/or inactive output pin	–5	+5	µA	
Maximum output current from active output pin	–25	+25	mA	
80-pin QFP maximum ratings at –40°C to 70°C				
Total power dissipation		550	mW	
Maximum current into V_{DD} or out of V_{SS}		150	mA	
80-pin QFP maximum ratings at 70°C to 125°C				
Total power dissipation		200	mW	
Maximum current into V_{DD} or out of V_{SS}		56	mA	
68-pin PLCC maximum ratings at –40°C to 70°C				
Total power dissipation		1000	mW	
Maximum current into V_{DD} or out of V_{SS}		275	mA	
68-pin PLCC maximum ratings at 70°C to 125°C				
Total power dissipation		500	mW	
Maximum current into V_{DD} or out of V_{SS}		140	mA	

Appendix B. Register Tables

For the reader's convenience, this appendix lists all F64xx Series registers numerically by hexadecimal address.

General Purpose RAM

In the F64xx Series, the 000–FFF hexadecimal address range is partitioned for general-purpose random access memory, as follows.

Hex Addresses: 000–7FF

This address range is reserved for 2KB general-purpose register file RAM devices. For more details, see the Register File section on page 18.

Hex Addresses: 000–FFF

This address range is reserved for 4KB general-purpose register file RAM devices. For more details, see the Register File section on page 18.

Timer 0

For more information about these Timer Control registers, see the Timer Control Register Definitions section on page 72.

Hex Address: F00

Table 138. Timer 0–3 High Byte Register (TxH)

Bit	7	6	5	4	3	2	1	0
Field	TH							
RESET	0							
R/W	R/W							
Address	F00H, F08H, F10H, F18H							

Hex Address: F43

Table 174. UART Control 1 Register (U*x*CTL1)

Bit	7	6	5	4	3	2	1	0
Field	MPMD[1]	MPEN	MPMD[0]	MPBT	DEPOL	BRGCTL	RDAIRQ	IREN
RESET	0							
R/W	R/W							
Address	F43H and F4BH							

Hex Address: F44

Table 175. UART Status 1 Register (U*x*STAT1)

Bit	7	6	5	4	3	2	1	0
Field	Reserved						NEWFRM	MPRX
RESET	0							
R/W	R				R/W		R	
Address	F44H and F4CH							

Hex Address: F45

Table 176. UART Address Compare Register (U*x*ADDR)

Bit	7	6	5	4	3	2	1	0
Field	COMP_ADDR							
RESET	0							
R/W	R/W							
Address	F45H and F4DH							

Hex Address: F46

Table 177. UART Baud Rate High Byte Register (U*x*BRH)

Bit	7	6	5	4	3	2	1	0
Field	BRH							
RESET	1							
R/W	R/W							
Address	F46H and F4EH							

Hex Address: F47

Table 178. UART Baud Rate Low Byte Register (U*x*BRL)

Bit7	7	6	5	4	3	2	1	0
Field	BRL							
RESET	1							
R/W	R/W							
Address	F47H and F4FH							

Hex Address: F48

Table 179. UART Transmit Data Register (U*x*TXD)

Bit	7	6	5	4	3	2	1	0
Field	TXD							
RESET	X							
R/W	W							
Address	F40H and F48H							

Table 180. UART Receive Data Register (U*x*RXD)

Bit	7	6	5	4	3	2	1	0
Field	RXD							
RESET	X							
R/W	R							
Address	F40H and F48H							

Hex Address: F49

Table 181. UART Status 0 Register (U*x*STAT0)

Bit	7	6	5	4	3	2	1	0
Field	RDA	PE	OE	FE	BRKD	TDRE	TXE	CTS
RESET	0						1	X
R/W	R							
Address	F41H and F49H							

first opcode map 247
FLAGS 229
flags register 229
flash
  controller 5
  option bit address space 181
  option bit configuration - reset 181
  program memory address 0001H 183
flash memory
  arrangement 171
  byte programming 174
  code protection 173
  configurations 170
  control register definitions 176
  controller bypass 175
  electrical characteristics and timing 214
  flash control register 177, 285
  flash status register 178
  frequency high and low byte registers 180
  mass erase 175
  operation 172
  operation timing 172
  page erase 175
  page select register 178
FPS register 178
FSTAT register 178

G

gated mode 79
general-purpose I/O 37
GPIO 5, 37
  alternate functions 38
  architecture 38
  control register definitions 40
  input data sample timing 218
  interrupts 40
  port A-H address registers 41
  port A-H alternate function sub-registers 43
  port A-H control registers 42
  port A-H data direction sub-registers 42
  port A-H high drive enable sub-registers 45
  port A-H input data registers 47
  port A-H output control sub-registers 44
  port A-H output data registers 47
  port A-H Stop Mode Recovery sub-registers 46
  port availability by device 37
  port input timing 218
  port output timing 219

H

H 229
HALT 233
halt mode 36, 233
hexadecimal number prefix/suffix 229

I

I2C 5
  10-bit address read transaction 140
  10-bit address transaction 137
  10-bit addressed slave data transfer format 137
  10-bit receive data format 140
  7-bit address transaction 134
  7-bit address, reading a transaction 139
  7-bit addressed slave data transfer format 134, 135, 136
  7-bit receive data transfer format 139
  baud high and low byte registers 146, 148, 150
  C status register 143, 263
  control register definitions 142
  controller 129
  controller signals 15
  interrupts 131
  operation 130
  SDA and SCL signals 131
  stop and start conditions 133
I2CBRH register 147, 148, 150, 263, 264
I2CBRL register 147, 263
I2CCTL register 145, 263
I2CDATA register 143, 262
I2CSTAT register 143, 263
IM 228
immediate data 228
immediate operand prefix 229
INC 231
increment 231