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	64KB (64K 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/z8f6422ar020sg

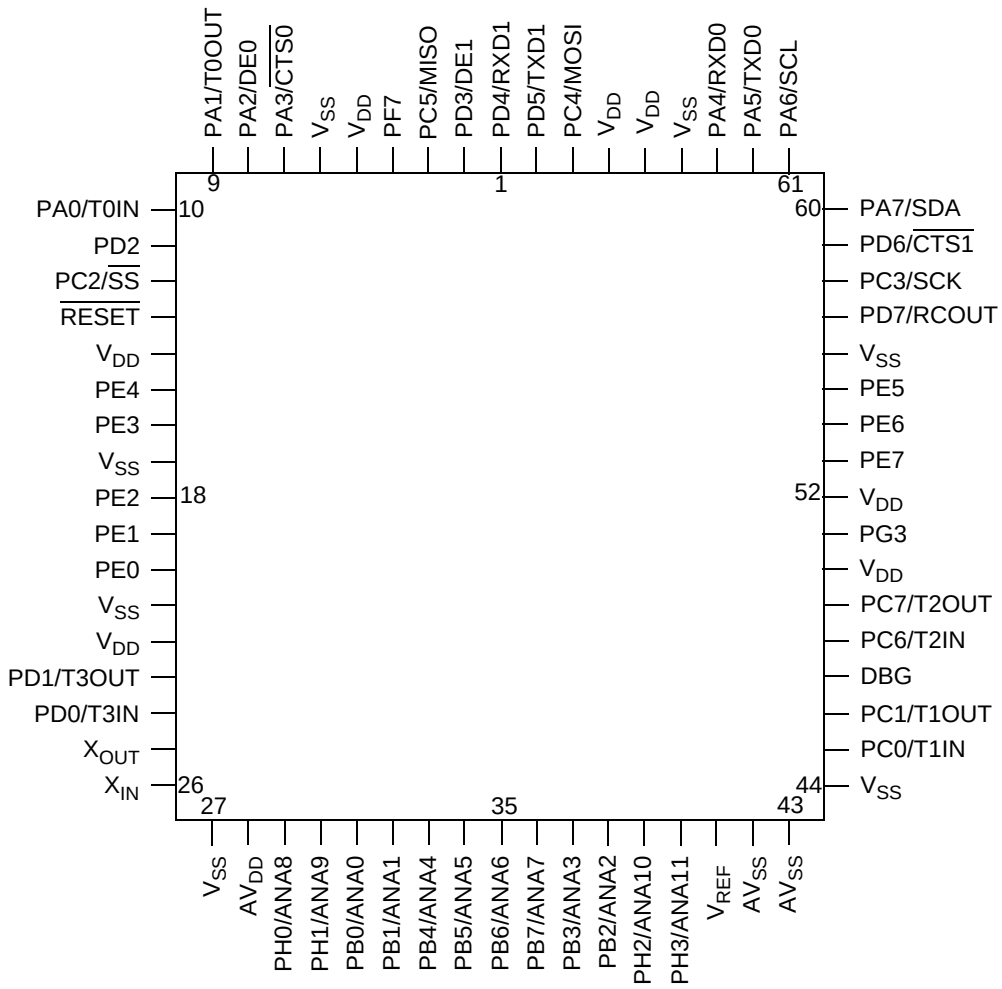

Figure 6. Z8 Encore! XP F64xx Series in 68-Pin Plastic Leaded Chip Carrier (PLCC)

Address Space

The eZ8 CPU can access three distinct address spaces:

- The Register File contains addresses for the general-purpose registers and the eZ8 CPU, peripheral and general-purpose I/O port control registers
- The program memory contains addresses for all memory locations having executable code and/or data
- The Data Memory consists of the addresses for all memory locations that hold only data

These three address spaces are covered briefly in the following sections. For more information about the eZ8 CPU and its address space, refer to the eZ8 CPU Core User Manual (UM0128), which is available for download on www.zilog.com.

Register File

The Register File address space in the Z8 Encore! XP F64xx Series is 4KB (4096 bytes). The Register File is composed of two sections: control registers and general-purpose registers. When instructions are executed, registers are read from when defined as sources and written to when defined as destinations. The architecture of the eZ8 CPU allows all general-purpose registers to function as accumulators, address pointers, index registers, stack areas, or scratch pad memory.

The upper 256 bytes of the 4KB Register File address space are reserved for control of the eZ8 CPU, the on-chip peripherals, and the I/O ports. These registers are located at addresses from F00H to FFFH. Some of the addresses within the 256-byte control register section are reserved (unavailable). Reading from an reserved Register File addresses returns an undefined value. Writing to reserved Register File addresses is not recommended and can produce unpredictable results.

The on-chip RAM always begins at address 000H in the Register File address space. The Z8 Encore! XP F64xx Series provide 2KB to 4KB of on-chip RAM depending upon the device. Reading from Register File addresses outside the available RAM addresses (and not within the control register address space) returns an undefined value. Writing to these Register File addresses produces no effect. To determine the amount of RAM available for the specific Z8 Encore! XP F64xx Series device, see the Part Selection Guide section on page 2.

Low-Power Modes

The Z8 Encore! XP F64xx Series products contain power-saving features. The highest level of power reduction is provided by STOP Mode. The next level of power reduction is provided by HALT Mode.

STOP Mode

Execution of the eZ8 CPU's stop instruction places the device into STOP Mode. In STOP Mode, the operating characteristics are:

- Primary crystal oscillator is stopped; the X_{IN} pin is driven High and the X_{OUT} pin is driven Low
- System clock is stopped
- eZ8 CPU is stopped
- Program counter (PC) stops incrementing
- The Watchdog Timer and its internal RC oscillator continue to operate, if enabled for operation during STOP Mode
- The Voltage Brown-Out protection circuit continues to operate, if enabled for operation in STOP Mode using the associated option bit
- All other on-chip peripherals are idle

To minimize current in STOP Mode, all GPIO pins that are configured as digital inputs must be driven to one of the supply rails (V_{CC} or GND), the Voltage Brown-Out protection must be disabled, and the Watchdog Timer must be disabled. The devices can be brought out of STOP Mode using Stop Mode Recovery. For more information about Stop Mode Recovery, see the Reset and Stop Mode Recovery chapter on page 28.

Caution: STOP Mode must not be used when driving the Z8 Encore! XP F64xx Series devices with an external clock driver source.

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.

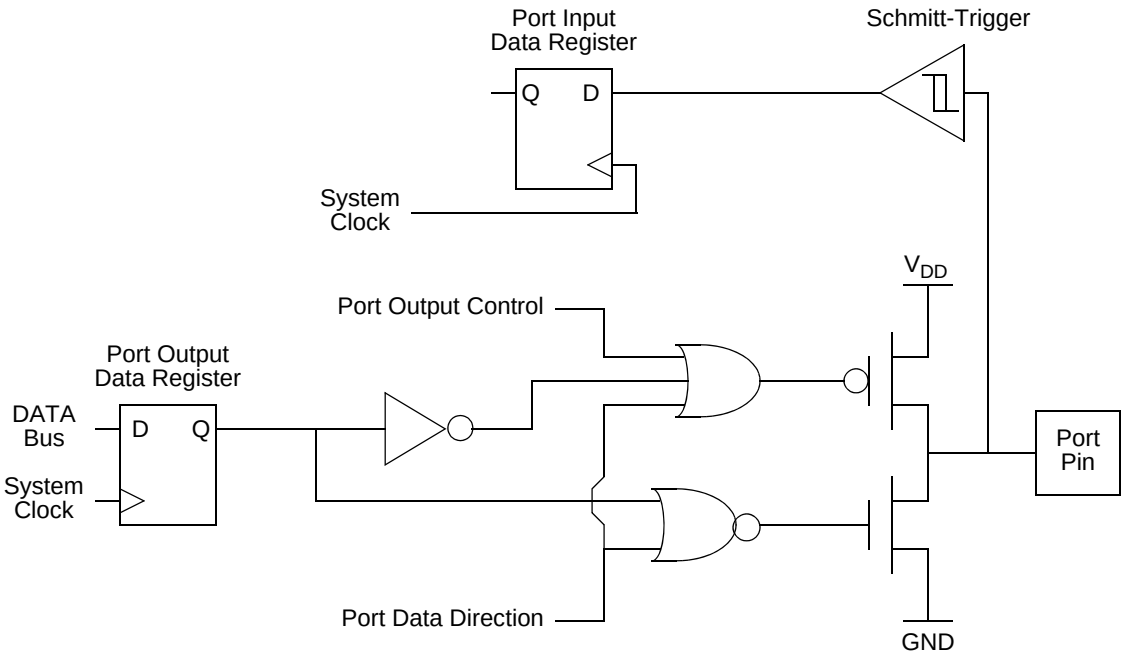

Figure 10. GPIO Port Pin Block Diagram

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.

Table 24. Interrupt Request 0 Register (IRQ0)

Bit	7	6	5	4	3	2	1	0
Field	T2I	T1I	T0I	U0RXI	U0TXI	I2CI	SPII	ADCI
RESET	0							
R/W	R/W							
Address	FC0H							

Bit	Description
[7] T2I	**Timer 2 Interrupt Request** 0 = No interrupt request is pending for Timer 2. 1 = An interrupt request from Timer 2 is awaiting service.
[6] T1I	**Timer 1 Interrupt Request** 0 = No interrupt request is pending for Timer 1. 1 = An interrupt request from Timer 1 is awaiting service.
[5] T0I	**Timer 0 Interrupt Request** 0 = No interrupt request is pending for Timer 0. 1 = An interrupt request from Timer 0 is awaiting service.
[4] U0RXI	**UART 0 Receiver Interrupt Request** 0 = No interrupt request is pending for the UART 0 receiver. 1 = An interrupt request from the UART 0 receiver is awaiting service.
[3] U0TXI	**UART 0 Transmitter Interrupt Request** 0 = No interrupt request is pending for the UART 0 transmitter. 1 = An interrupt request from the UART 0 transmitter is awaiting service.
[2] I2CI	**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.
[1] SPII	**SPI Interrupt Request** 0 = No interrupt request is pending for the SPI. 1 = An interrupt request from the SPI is awaiting service.
[0] 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.

Table 31. IRQ1 Enable High Bit Register (IRQ1ENH)

Bit	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
Address	FC4H							

Bit	Description
[7:0] PADxENH	**Port A or Port D Bit[*x*] Interrupt Request Enable High Bit** To select either Port A or Port D as the interrupt source, see the Interrupt Port Select Register on page 60.

Note: *x* indicates register bits in the range [7:0].

Table 32. IRQ1 Enable Low Bit Register (IRQ1ENL)

Bit	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
Address	FC5H							

Bit	Description
[7:0] PADxENL	**Port A or Port D Bit[*x*] Interrupt Request Enable Low Bit** To select either Port A or Port D as the interrupt source, see the Interrupt Port Select Register on page 60.

Note: *x* indicates register bits in the range [7:0].

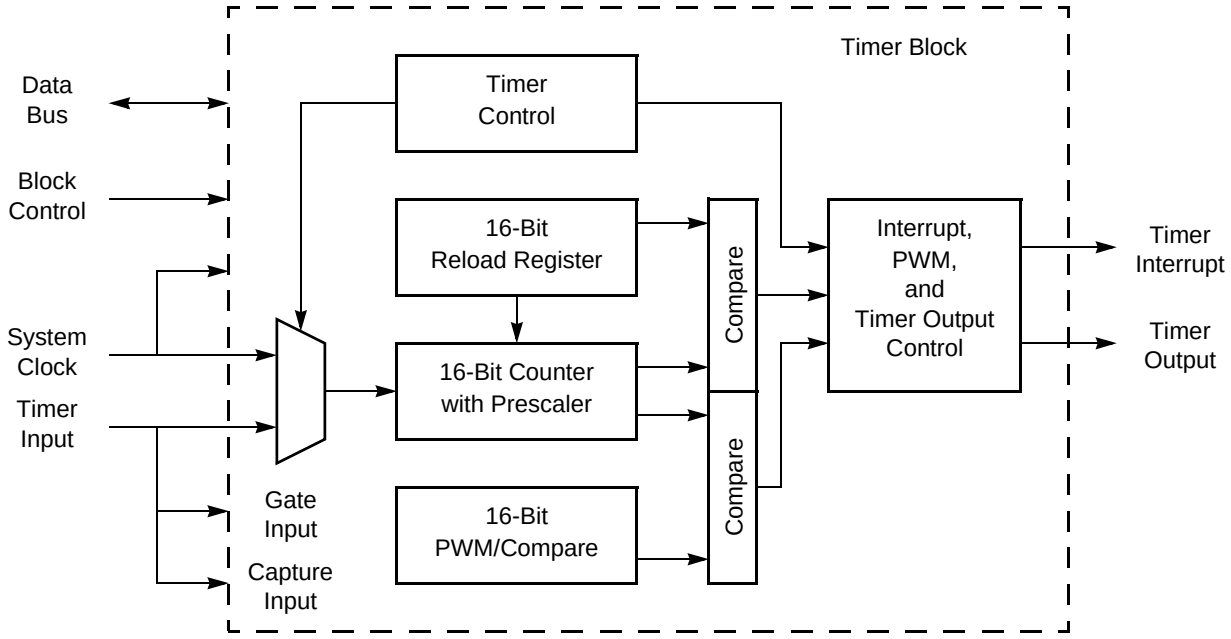

Figure 12. Timer Block Diagram

Operation

The timers are 16-bit up-counters. Minimum time-out delay is set by loading the value 0001H into the Timer Reload High and Low Byte registers and setting the prescale value to 1. Maximum time-out delay is set by loading the value 0000H into the Timer Reload High and Low Byte registers and setting the prescale value to 128. If the Timer reaches FFFFH, the timer rolls over to 0000H and continues counting.

Timer Operating Modes

The timers can be configured to operate in the following modes:

ONE-SHOT Mode

In ONE-SHOT Mode, the timer counts up to the 16-bit reload value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the reload value, the timer generates an interrupt and the count value in the Timer High and Low Byte registers is reset to 0001H. Then, the timer is automatically disabled and stops counting.

Also, if the timer output alternate function is enabled, the timer output pin changes state for one system clock cycle (from Low to High or from High to Low) upon timer reload. If

Table 49. Watchdog Timer Events

Reset or Stop Mode Recovery Event	POR	STOP	WDT	EXT
Power-On Reset	1	0	0	0
Reset using $\overline{\text{RESET}}$ pin assertion	0	0	0	1
Reset using Watchdog Timer time-out	0	0	1	0
Reset using the On-Chip Debugger (OCDCTL[1] set to 1)	1	0	0	0
Reset from STOP Mode using DBG Pin driven Low	1	0	0	0
Stop Mode Recovery using GPIO pin transition	0	1	0	0
Stop Mode Recovery using Watchdog Timer time-out	0	1	1	0

Watchdog Timer Reload Upper, High and Low Byte Registers

The Watchdog Timer Reload Upper, High and Low Byte (WDTU, WDTH, WDTL) registers, shown in Tables 50 through 52, form the 24-bit reload value that is loaded into the Watchdog Timer when a WDT instruction executes. The 24-bit reload value is {WDTU[7:0], WDTH[7:0], WDTL[7:0]}. Writing to these registers sets the appropriate reload value. Reading from these registers returns the current Watchdog Timer count value.

! **Caution:** The 24-bit WDT reload value must not be set to a value less than `000004H`.

Table 50. Watchdog Timer Reload Upper Byte Register (WDTU)

Bit	7	6	5	4	3	2	1	0
Field	WDTU							
RESET	1							
R/W	R/W*							
Address	FF1H							

Note: *R/W = Read returns the current WDT count value; write sets the appropriate reload value.

Bit	Description
[7:0] WDTU	**WDT Reload Upper Byte** Most significant byte, bits[23:16] of the 24-bit WDT reload value.

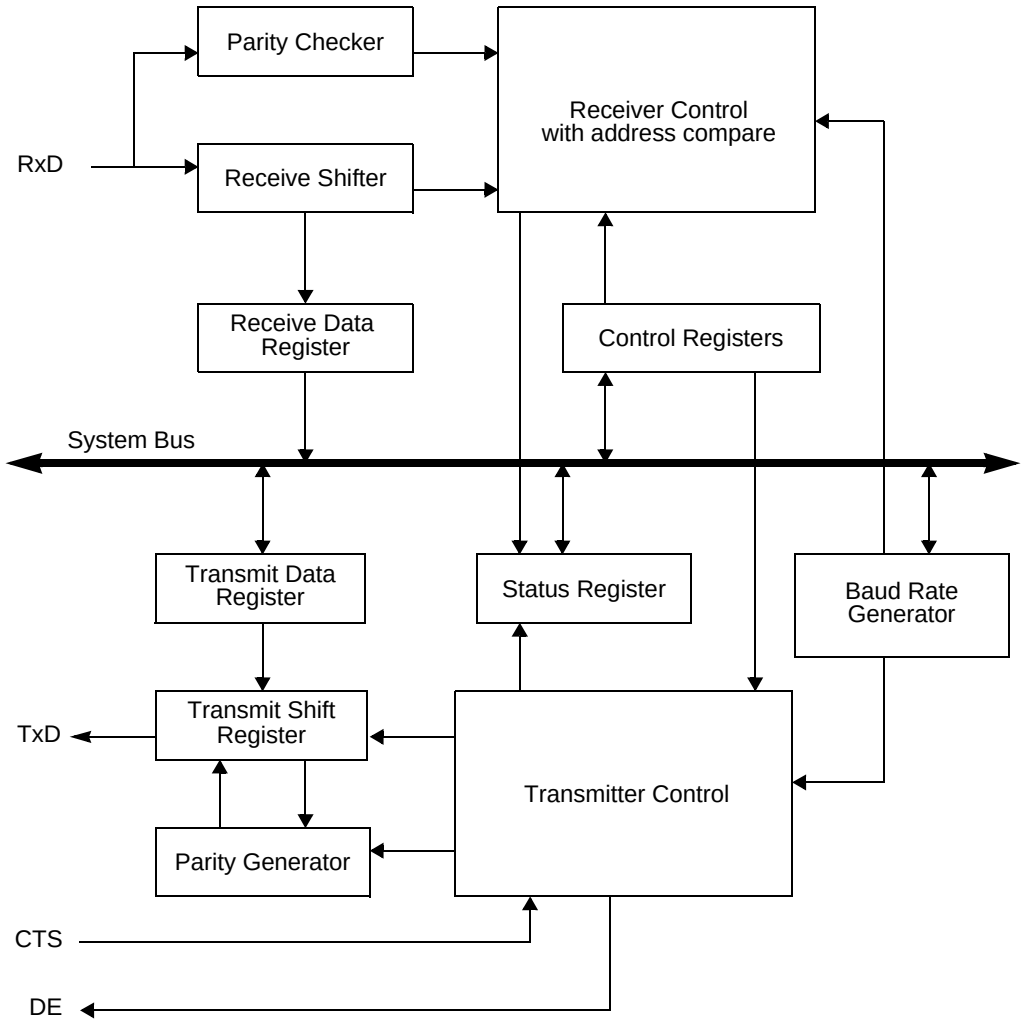

Figure 13. UART Block Diagram

Operation

The UART always transmits and receives data in an 8-bit data format, least significant bit first. An even or odd parity bit can be optionally added to the data stream. Each character begins with an active Low start bit and ends with either 1 or 2 active High stop bits. Figures 14 and 15 display the asynchronous data format employed by the UART without parity and with parity, respectively.

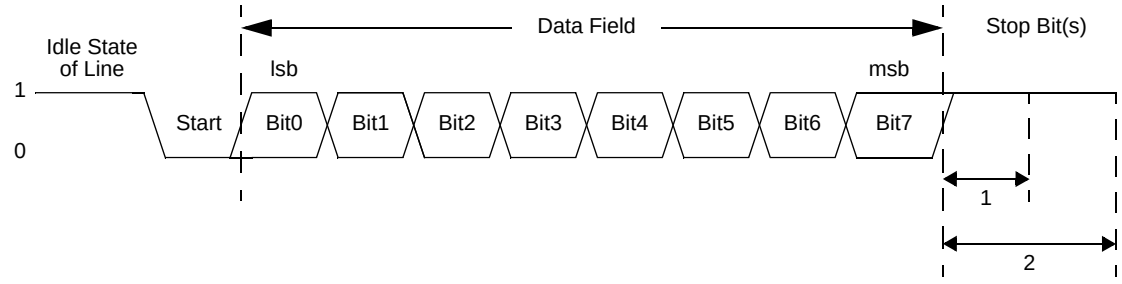

Figure 14. UART Asynchronous Data Format without Parity

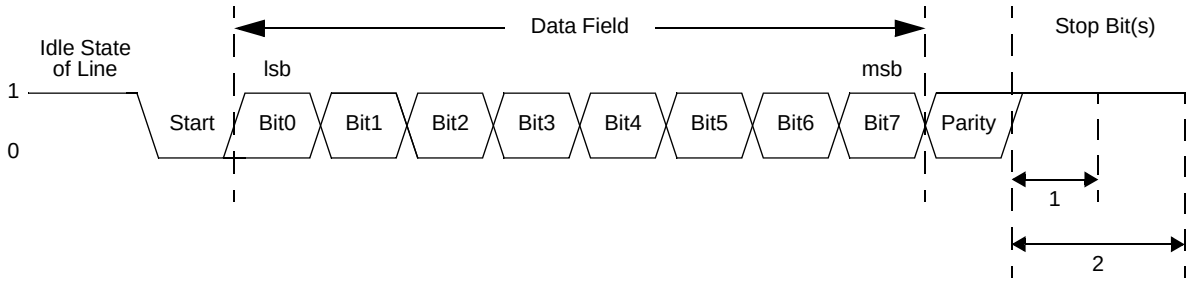

Figure 15. UART Asynchronous Data Format with Parity

Transmitting Data using the Polled Method

Observe the following procedure to transmit data using the polled method of operation:

1. Write to the UART Baud Rate High and Low Byte registers to set the appropriate baud rate.
2. Enable the UART pin functions by configuring the associated GPIO port pins for alternate function operation.
3. If MULTIPROCESSOR Mode is appropriate, write to the UART Control 1 Register to enable MULTIPROCESSOR (9-Bit) Mode functions.
   - Set the MULTIPROCESSOR Mode Select (MPEN) to Enable MULTIPROCESSOR Mode
4. Write to the UART Control 0 Register to:
   - Set the transmit enable bit (TEN) to enable the UART for data transmission
   - If parity is appropriate and MULTIPROCESSOR Mode is not enabled, set the parity enable bit (PEN) and select either Even or Odd parity (PSEL)

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.

Bit	Description (Continued)
[3] WSEL	**Word Select** 0 = DMA*x* transfers a single byte per request. 1 = DMA*x* transfers a two-byte word per request. The address for the on-chip peripheral control register must be an even address.
[2:0] RSS	**Request Trigger Source Select** The Request Trigger Source Select field determines the peripheral that can initiate a DMA transfer. The corresponding interrupts do not need to be enabled within the Interrupt Controller to initiate a DMA transfer. However, if the Request Trigger Source can enable or disable the interrupt request sent to the Interrupt Controller, the interrupt request must be enabled within the Request Trigger Source block. 000 = Timer 0. 001 = Timer 1. 010 = Timer 2. 011 = Timer 3. 100 = DMA0 Control Register: UART0 Received Data Register contains valid data. DMA1 Control Register: UART0 Transmit Data Register empty. 101 = DMA0 Control Register: UART1 Received Data Register contains valid data. DMA1 Control Register: UART1 Transmit Data Register empty. 110 = DMA0 Control Register: I^2C Receiver Interrupt. DMA1 Control Register: I^2C Transmitter Interrupt Register empty. 111 = Reserved.

DMA*x* I/O Address Register

The DMA*x* I/O Address Register, shown in Table 79, contains the low byte of the on-chip peripheral address for data transfer. The full 12-bit Register File address is provided by {FH, DMA*x*_IO[7:0]}. When the DMA is configured for two-byte word transfers, the DMA*x* I/O Address Register must contain an even-numbered address.

Table 79. DMA*x* I/O Address Register (DMA*x*IO)

Bit	7	6	5	4	3	2	1	0
Field	DMA_IO							
RESET	X							
R/W	R/W							
Address	FB1H, FB9H							

Bit	Description
[7:0] DMA_IO	**DMA On-Chip Peripheral Control Register Address** This byte sets the low byte of the on-chip peripheral control register address on Register File Page FH (addresses F00H to FFFH).

Table 84. DMA_ADC Address Register (DMAA_ADDR)

Bit	7	6	5	4	3	2	1	0
Field	DMAA_ADDR							Reserved
RESET	X							
R/W	R/W							
Address	FBDH							

Bit	Description
[7:1] 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}.
0	**Reserved** This bit is reserved and must be programmed to 0.

DMA_ADC Control Register

The DMA_ADC Control Register, shown in Table 85, enables and sets options (DMA enable and interrupt enable) for ADC operation.

Table 85. DMA_ADC Control Register (DMAACTL)

Bit	7	6	5	4	3	2	1	0
Field	DAEN	IRQEN	Reserved		ADC_IN			
RESET	0							
R/W	R/W							
Address	FBEH							

Bit	Description
[7] DAEN	**DMA_ADC Enable** 0 = DMA_ADC is disabled and the ADC analog input Number (ADC_IN) is reset to 0. 1 = DMA_ADC is enabled.
[6] IRQEN	**Interrupt Enable** 0 = DMA_ADC does not generate any interrupts. 1 = DMA_ADC generates an interrupt after transferring data from the last ADC analog input specified by the ADC_IN field.

Continuous Conversion

When configured for continuous conversion, the ADC continuously performs an analog-to-digital conversion on the selected analog input. Each new data value over-writes the previous value stored in the ADC Data registers. An interrupt is generated after each conversion.

Caution: In CONTINUOUS Mode, you must be aware that ADC updates are limited by the input signal bandwidth of the ADC and the latency of the ADC and its digital filter. Step changes at the input are not seen at the next output from the ADC. The response of the ADC (in all modes) is limited by the input signal bandwidth and the latency.

Observe the following procedure for setting up the ADC and initiating continuous conversion:

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

The write-only Flash Control Register shares its Register File address with the read-only Flash Status Register.

Table 93. Flash Control Register (FCTL)

Bit	7	6	5	4	3	2	1	0
Field	FCMD							
RESET	0							
R/W	W							
Address	FF8H							

Bit	Description
[7:0] FCMD	**Flash Command*** 73H = First unlock command. 8CH = Second unlock command. 95H = Page erase command. 63H = Mass erase command 5EH = Flash Sector Protect Register select.

Note: *All other commands, or any command out of sequence, lock the Flash Controller.

Table 107. DC Characteristics (Continued)

		T_A = –40°C to 125°C				
Symbol	**Parameter**	**Minimum**	**Typical**	**Maximum**	**Units**	**Conditions**
I_{DDS}	Stop Mode Supply Current; GPIO pins configured as outputs (see Figure 47 on page 209 and Figure 48 on page 210)	–	520		µA	VBO and WDT enabled
				700		V_{DD} = 3.6V
				650		V_{DD} = 3.3V
		–	10		µA	VBO disabled, WDT enabled, T_A = 0 to 70ºC
				25		V_{DD} = 3.6V
				20		V_{DD} = 3.3V
		–	–		µA	VBO disabled, WDT enabled, T_A = –40 to +105ºC
				80		V_{DD} = 3.6V
				70		V_{DD} = 3.3V
		–	–		µA	VBO disabled, WDT enabled, T_A = –40 to +125ºC
				250		V_{DD} = 3.6V
				150		V_{DD} = 3.3V

Notes:
1. This condition excludes all pins that have on-chip pull-ups, when driven Low.
2. These values are provided for design guidance only and are not tested in production.

Table 136. 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
LDX dst, src	dst ← src	r	ER	84	–	–	–	–	–	–	3	2
		Ir	ER	85							3	3
		R	IRR	86							3	4
		IR	IRR	87							3	5
		r	X(rr)	88							3	4
		X(rr)	r	89							3	4
		ER	r	94							3	2
		ER	Ir	95							3	3
		IRR	R	96							3	4
		IRR	IR	97							3	5
		ER	ER	E8							4	2
		ER	IM	E9							4	2
LEA dst, X(src)	dst ← src + X	r	X(r)	98	–	–	–	–	–	–	3	3
		rr	X(rr)	99							3	5
MULT dst	dst[15:0] ← dst[15:8] * dst[7:0]	RR		F4	–	–	–	–	–	–	2	8
NOP	No operation			0F	–	–	–	–	–	–	1	2
OR dst, src	dst ← dst OR src	r	r	42	–	*	*	0	–	–	2	3
		r	Ir	43							2	4
		R	R	44							3	3
		R	IR	45							3	4
		R	IM	46							3	3
		IR	IM	47							3	4
ORX dst, src	dst ← dst OR src	ER	ER	48	–	*	*	0	–	–	4	3
		ER	IM	49							4	3
POP dst	dst ← @SP SP ← SP + 1	R		50	–	–	–	–	–	–	2	2
		IR		51							2	3

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

Hex Address: F05

Table 143. Timer 0–3 PWM Low Byte Register (TxPWML)

Bit	7	6	5	4	3	2	1	0
Field	PWML							
RESET	0							
R/W	R/W							
Address	F05H, F0DH, F15H, F1DH							

Hex Address: F06

Table 144. Timer 0–3 Control 0 Register (TxCTL0)

Bit	7	6	5	4	3	2	1	0
Field	Reserved			CSC	Reserved			
RESET	0							
R/W	R/W							
Address	F06H, F0EH, F16H, F1EH							

Hex Address: F07

Table 145. Timer 0–3 Control 1 Register (TxCTL1)

Bit	7	6	5	4	3	2	1	0
Field	TEN	TPOL	PRES			TMODE		
RESET	0							
R/W	R/W							
Address	F07H, F0FH, F17H, F1FH							

Hex Address: F08

Table 146. 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							

Index

Numerics

10-bit ADC 5

A

absolute maximum ratings 201
AC characteristics 217
ADC 231
  architecture 162
  automatic power-down 164
  block diagram 163
  continuous conversion 165
  control register 166
  control register definitions 166
  data high byte register 168
  data low bits register 169
  DMA control 166
  electrical characteristics and timing 215
  operation 164
  single-shot conversion 164
ADCCTL register 166
ADCDH register 168
ADCDL register 169
ADCX 231
ADD 231
add - extended addressing 231
add with carry 231
add with carry - extended addressing 231
additional symbols 229
address space 19
ADDX 231
analog signals 16
analog-to-digital converter (ADC) 162
AND 234
ANDX 234
arithmetic instructions 231
assembly language programming 226
assembly language syntax 227

B

baud rate generator, UART 99
BCLR 232
binary number suffix 229
BIT 232
bit 228
  clear 232
  manipulation instructions 232
  set 232
  set or clear 232
  swap 232
  test and jump 234
  test and jump if non-zero 234
  test and jump if zero 234
bit jump and test if non-zero 234
bit swap 235
block diagram 4
block transfer instructions 232
BRK 234
BSET 232
BSWAP 232, 235
BTJ 234
BTJNZ 234
BTJZ 234

C

CALL procedure 234
capture mode 79
capture/compare mode 79
cc 228
CCF 233
characteristics, electrical 201
clear 233
clock phase (SPI) 117
CLR 233
COM 234
compare 79
compare - extended addressing 231
compare mode 79

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