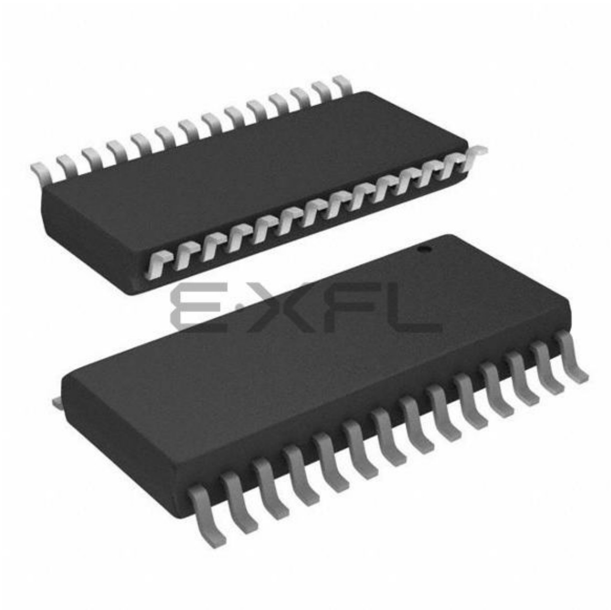

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	5MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	22
Program Memory Size	2KB (2K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	28-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0223sj005sc

Warning: DO NOT USE IN LIFE SUPPORT

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer

 Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

Z8, Z8 Encore!, Z8 Encore! XP, Z8 Encore! MC, Crimzon, eZ80, and ZNEO are trademarks or registered trademarks of Zilog, Inc. All other product or service names are the property of their respective owners.

The pin configurations listed are preliminary and subject to change based on manufacturing limitations.

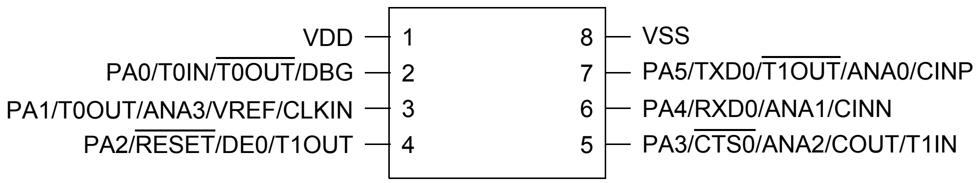

Figure 2. Z8F08x3, Z8F04x3, F02x3 and Z8F01x3 in 8-Pin SOIC, QFN/MLF-S, or PDIP Package*

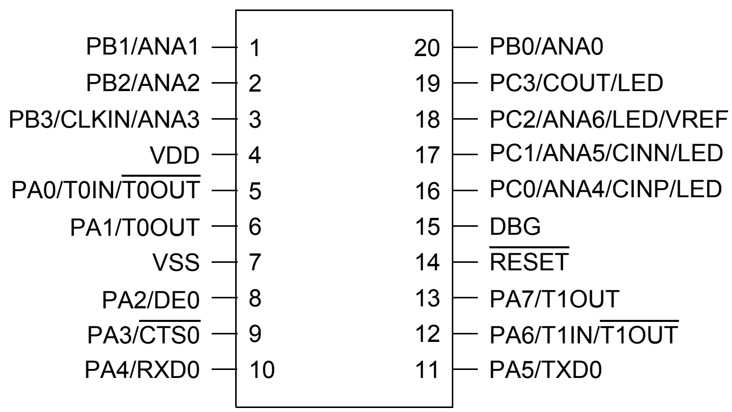

Figure 3. Z8F08x3, Z8F04x3, F02x3 and Z8F01x3 in 20-Pin SOIC, SSOP or PDIP Package*

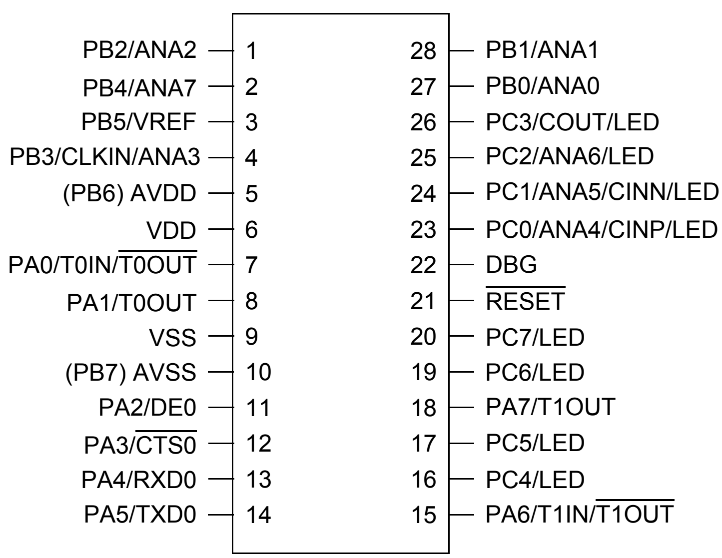

Figure 4. Z8F08x3, Z8F04x3, F02x3 and Z8F01x3 in 28-Pin SOIC, SSOP or PDIP Package*

clock and reset signals, the required reset duration can be as short as three clock periods and as long as four. A reset pulse three clock cycles in duration might trigger a reset; a pulse four cycles in duration always triggers a reset.

While the $\overline{\text{RESET}}$ input pin is asserted Low, the Z8 Encore! XP F0823 Series devices remain in the Reset state. If the $\overline{\text{RESET}}$ pin is held Low beyond the System Reset time-out, the device exits the Reset state on the system clock rising edge following $\overline{\text{RESET}}$ pin deassertion. Following a System Reset initiated by the external $\overline{\text{RESET}}$ pin, the EXT status bit in the WDT Control (WDTCTL) register is set to 1.

External Reset Indicator

During System Reset or when enabled by the GPIO logic (see Port A–C Control Registers on page 44), the $\overline{\text{RESET}}$ pin functions as an open-drain (active Low) reset mode indicator in addition to the input functionality. This reset output feature allows an Z8 Encore! XP F0823 Series device to reset other components to which it is connected, even if that reset is caused by internal sources such as POR, VBO, or WDT events.

After an internal reset event occurs, the internal circuitry begins driving the $\overline{\text{RESET}}$ pin Low. The $\overline{\text{RESET}}$ pin is held Low by the internal circuitry until the appropriate delay listed in Table 9 has elapsed.

On-Chip Debugger Initiated Reset

A POR is initiated using the On-Chip Debugger by setting the RST bit in the OCD Control register. The OCD block is not reset but the rest of the chip goes through a normal system reset. The RST bit automatically clears during the System Reset. Following the System Reset, the POR bit in the Reset Status (RSTSTAT) register is set.

Stop Mode Recovery

The device enters into STOP mode when eZ8 CPU executes a `STOP` instruction. For more details on STOP mode, see Low-Power Modes on page 31. During Stop Mode Recovery, the CPU is held in reset for 66 IPO cycles if the crystal oscillator is disabled or 5000 cycles if it is enabled. The SMR delay also included the time required to start up the IPO.

Stop Mode Recovery does not affect on-chip registers other than the Watchdog Timer Control register (WDTCTL) and the Oscillator Control register (OSCCTL). After any Stop Mode Recovery, the IPO is enabled and selected as the system clock. If another system clock source is required or IPO disabling is required, the Stop Mode Recovery code must reconfigure the oscillator control block such that the correct system clock source is enabled and selected.

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

Stop Mode Recovery Using the External $\overline{\text{RESET}}$ Pin

When the Z8 Encore! XP F0823 Series device is in STOP mode and the external $\overline{\text{RESET}}$ pin is driven Low, a system reset occurs. Because of a glitch filter operating on the $\overline{\text{RESET}}$ pin, the Low pulse must be greater than the minimum width specified, or it is ignored. For more details, see Electrical Characteristics on page 193.

Reset Register Definitions

Reset Status Register

The Reset Status (RSTSTAT) register 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.

This register shares its address with the Watchdog Timer control register, which is write-only (Table 12).

Table 12. Reset Status Register (RSTSTAT)

BITS	7	6	5	4	3	2	1	0
FIELD	POR	STOP	WDT	EXT	Reserved			
RESET	See descriptions below			0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
ADDR	FF0H							

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 WDT time-out	0	0	1	0
Reset using the OCD (OCTCTL[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 WDT time-out	0	1	1	0

POR—Power-On Reset Indicator
If this bit is set to 1, a Power-On Reset event is 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.

tions as a GPIO pin. If it is not present, the debug feature is disabled until/unless another reset event occurs. For more details, see On-Chip Debugger on page 151.

Crystal Oscillator Override

For systems using a crystal oscillator, PA0 and PA1 are used to connect the crystal. When the crystal oscillator is enabled (see Oscillator Control Register Definitions on page 167), the GPIO settings are overridden and PA0 and PA1 are disabled.

5 V Tolerance

All six I/O pins on the 8-pin devices are 5 V-tolerant, unless the programmable pull-ups are enabled. If the pull-ups are enabled and inputs higher than V_{DD} are applied to these parts, excessive current flows through those pull-up devices and can damage the chip.

Note: *In the 20- and 28-pin versions of this device, any pin which shares functionality with an ADC, crystal or comparator port is not 5 V-tolerant, including PA[1:0], PB[5:0], and PC[2:0]. All other signal pins are 5 V-tolerant, and can safely handle inputs higher than* V_{DD} *even with the pull-ups enabled.*

External Clock Setup

For systems using an external TTL drive, PB3 is the clock source for 20- and 28-pin devices. In this case, configure PB3 for alternate function CLKIN. Write the Oscillator Control Register (see Oscillator Control Register Definitions on page 167) such that the external oscillator is selected as the system clock. For 8-pin devices use PA1 instead of PB3.

Port A–C Address Registers

The Port A–C Address registers select the GPIO Port functionality accessible through the Port A–C Control registers. The Port A–C Address and Control registers combine to provide access to all GPIO Port controls (Table 18).

Table 18. Port A–C GPIO Address Registers (PxADDR)

BITS	7	6	5	4	3	2	1	0
FIELD	PADDR[7:0]							
RESET	00H							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	FD0H, FD4H, FD8H							

PADDR[7:0]—Port Address
The Port Address selects one of the sub-registers accessible through the Port Control register.

PADDR[7:0]	Port Control Sub-register Accessible Using the Port A–C Control Registers
00H	No function. Provides some protection against accidental Port reconfiguration.
01H	Data Direction.
02H	Alternate Function.
03H	Output Control (Open-Drain).
04H	High Drive Enable.
05H	Stop Mode Recovery Source Enable.
06H	Pull-up Enable.
07H	Alternate Function Set 1.
08H	Alternate Function Set 2.
09H–FFH	No function.

Port A–C Control Registers

The Port A–C Control registers set the GPIO port operation. The value in the corresponding Port A–C Address register determines which sub-register is read from or written to by a Port A–C Control register transaction (Table 19).

Table 30. LED Drive Enable (LEDEN)

BITS	7	6	5	4	3	2	1	0
FIELD	LEDEN[7:0]							
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	F82H							

LEDEN[7:0]—LED Drive Enable
These bits determine which Port C pins are connected to an internal current sink.
0 = Tristate the Port C pin.
1= Connect controlled current sink to the Port C pin.

LED Drive Level High Register

The LED Drive Level registers contain two control bits for each Port C pin (Table 31). These two bits select between four programmable drive levels. Each pin is individually programmable.

Table 31. LED Drive Level High Register (LEDLVLH)

BITS	7	6	5	4	3	2	1	0
FIELD	LEDLVLH[7:0]							
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	F83H							

LEDLVLH[7:0]—LED Level High Bit
{LEDLVLH, LEDLVLL} select one of four programmable current drive levels for each Port C pin.
00 = 3 mA
01= 7 mA
10= 13 mA
11= 20 mA

LED Drive Level Low Register

The LED Drive Level registers contain two control bits for each Port C pin (Table 32). These two bits select between four programmable drive levels. Each pin is individually programmable.

Table 33. Trap and Interrupt Vectors in Order of Priority

Priority	Program Memory Vector Address	Interrupt or Trap Source
Highest	0002H	Reset (not an interrupt)
	0004H	Watchdog Timer (see Watchdog Timer on page 87)
	003AH	Primary Oscillator Fail Trap (not an interrupt)
	003CH	Watchdog Timer Oscillator Fail Trap (not an interrupt)
	0006H	Illegal Instruction Trap (not an interrupt)
	0008H	Reserved
	000AH	Timer 1
	000CH	Timer 0
	000EH	UART 0 receiver
	0010H	UART 0 transmitter
	0012H	Reserved
	0014H	Reserved
	0016H	ADC
	0018H	Port A Pin 7, selectable rising or falling input edge
	001AH	Port A Pin 6, selectable rising or falling input edge or Comparator Output
	001CH	Port A Pin 5, selectable rising or falling input edge
	001EH	Port A Pin 4, selectable rising or falling input edge
	0020H	Port A Pin 3 or Port D Pin 3, selectable rising or falling input edge
	0022H	Port A Pin 2 or Port D Pin 2, selectable rising or falling input edge
	0024H	Port A Pin 1, selectable rising or falling input edge
	0026H	Port A Pin 0, selectable rising or falling input edge
	0028H	Reserved
	002AH	Reserved
	002CH	Reserved
	002EH	Reserved
	0030H	Port C Pin 3, both input edges
	0032H	Port C Pin 2, both input edges
	0034H	Port C Pin 1, both input edges

Table 33. Trap and Interrupt Vectors in Order of Priority (Continued)

Priority	Program Memory Vector Address	Interrupt or Trap Source
Lowest	0036H	Port C Pin 0, both input edges
	0038H	Reserved

Architecture

Figure 8 displays the interrupt controller block diagram.

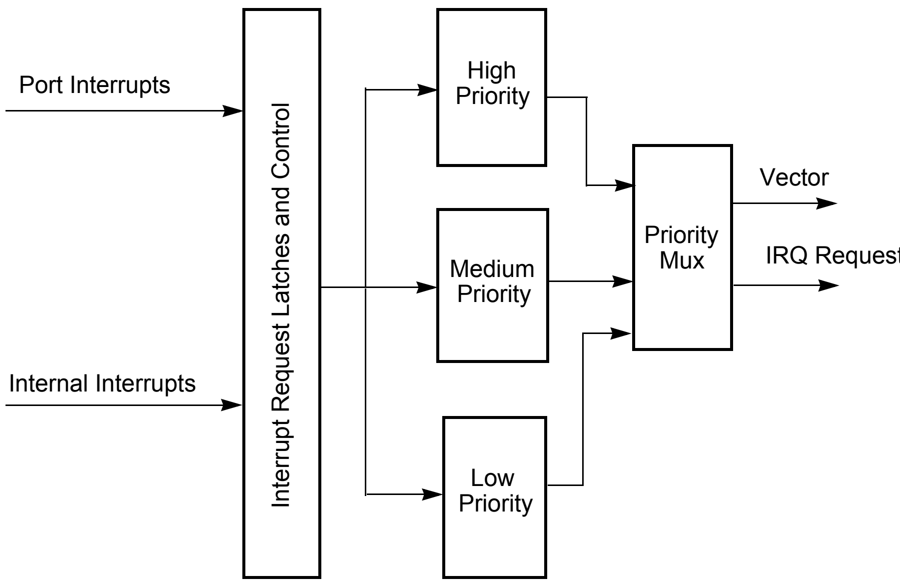

Figure 8. Interrupt Controller Block Diagram

Operation

Master Interrupt Enable

The master interrupt enable bit (`IRQE`) in the Interrupt Control register globally enables and disables interrupts.

Interrupts are globally enabled by any of the following actions:

- Execution of an Enable Interrupt (`EI`) instruction
- Execution of an Return from Interrupt (`IRET`) instruction

! Caution: *To avoid missing interrupts, use the following coding style to clear bits in the Interrupt Request 0 register:*

Good coding style that avoids lost interrupt requests:

```
ANDX IRQ0, MASK
```

Software Interrupt Assertion

Program code generates interrupts directly. Writing a 1 to the correct bit in the Interrupt Request register triggers an interrupt (assuming that interrupt is enabled). When the interrupt request is acknowledged by the eZ8 CPU, the bit in the Interrupt Request register is automatically cleared to 0.

! Caution: *The following coding style used to generate software interrupts by setting bits in the Interrupt Request registers is not recommended. All incoming interrupts received between execution of the first LDX command and the final LDX command are lost.*

Poor coding style that can result in lost interrupt requests:

```
LDX r0, IRQ0
OR r0, MASK
LDX IRQ0, r0
```

! Caution: *To avoid missing interrupts, use the following coding style to set bits in the Interrupt Request registers:*

Good coding style that avoids lost interrupt requests:

```
ORX IRQ0, MASK
```

Watchdog Timer Interrupt Assertion

The Watchdog Timer interrupt behavior is different from interrupts generated by other sources. The Watchdog Timer continues to assert an interrupt as long as the timeout condition continues. As it operates on a different (and usually slower) clock domain than the rest of the device, the Watchdog Timer continues to assert this interrupt for many system clocks until the counter rolls over.

! Caution: *To avoid re-triggerings of the Watchdog Timer interrupt after exiting the associated interrupt service routine, it is recommended that the service routine continues to read from the RSTSTAT register until the* `WDT` *bit is cleared as given in the following coding sample:*

```
CLEARWDT:
LDX r0, RSTSTAT ; read reset status register to clear wdt
   bit
BTJNZ 5, r0, CLEARWDT ; loop until bit is cleared
```

Table 48. 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 executing 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, reset or by a direct register write of a 0 to this bit.
0 = Interrupts are disabled
1 = Interrupts are enabled

Reserved—0 when read

0x = Timer Interrupt occurs on all defined Reload, Compare and Input Events
10 = Timer Interrupt only on defined Input Capture/Deassertion Events
11 = Timer Interrupt only on defined Reload/Compare Events

Reserved—Must be 0

PWMD—PWM Delay value
This field is a programmable delay to control the number of system clock cycles delay before the Timer Output and the Timer Output Complement are forced to their active state.

000 = No delay
001 = 2 cycles delay
010 = 4 cycles delay
011 = 8 cycles delay
100 = 16 cycles delay
101 = 32 cycles delay
110 = 64 cycles delay
111 = 128 cycles delay

INPCAP—Input Capture Event
This bit indicates if the most recent timer interrupt is caused by a Timer Input Capture Event.

0 = Previous timer interrupt is not a result of Timer Input Capture Event
1 = Previous timer interrupt is a result of Timer Input Capture Event

Timer 0–1 Control Register 1

The Timer 0–1 Control (TxCTL1) registers enable/disable the timers, set the prescaler value, and determine the timer operating mode.

Table 56. Timer 0–1 Control Register 1 (TxCTL1)

BITS	7	6	5	4	3	2	1	0
FIELD	TEN	TPOL	PRES			TMODE		
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	F07H, F0FH							

TEN—Timer Enable
0 = Timer is disabled
1 = Timer enabled to count

TPOL—Timer Input/Output Polarity
Operation of this bit is a function of the current operating mode of the timer

Watchdog Timer

The Watchdog Timer (WDT) protects against corrupt or unreliable software, power faults, and other system-level problems which can place Z8 Encore! XP® F0823 Series devices into unsuitable operating states. The features of Watchdog Timer include:

- On-chip RC oscillator
- A selectable time-out response: reset or interrupt
- 24-bit programmable time-out value

Operation

The WDT is a retriggerable one-shot timer that resets or interrupts Z8 Encore! XP F0823 Series devices when the WDT reaches its terminal count. The Watchdog Timer uses a dedicated on-chip RC oscillator as its clock source. The Watchdog Timer operates in only two modes: ON and OFF. Once enabled, it always counts and must be refreshed to prevent a time-out. Perform an enable by executing the `WDT` instruction or by setting the WDT_AO Flash Option Bit. The `WDT_AO` bit forces the Watchdog Timer to operate immediately upon reset, even if a `WDT` instruction has not been executed.

The Watchdog Timer is a 24-bit reloadable down counter that uses three 8-bit registers in the eZ8 CPU register space to set the reload value. The nominal WDT time-out period is described by the following equation:

$$\text{WDT Time-out Period (ms)} = \frac{\text{WDT Reload Value}}{10}$$

where the WDT reload value is the decimal value of the 24-bit value given by {WDTU[7:0], WDTH[7:0], WDTL[7:0]} and the typical Watchdog Timer RC oscillator frequency is 10 kHz. The Watchdog Timer cannot be refreshed after it reaches `000002H`. The WDT Reload Value must not be set to values below `000004H`. Table 57 provides information about approximate time-out delays for the minimum and maximum WDT reload values.

Table 57. Watchdog Timer Approximate Time-Out Delays

WDT Reload Value (Hex)	WDT Reload Value (Decimal)	Approximate Time-Out Delay (with 10 kHz typical WDT oscillator frequency)	
		Typical	Description
000004	4	400 μs	Minimum time-out delay
FFFFFF	16,777,215	28 minutes	Maximum time-out delay

Analog-to-Digital Converter

The Analog-to-Digital Converter (ADC) converts an analog input signal to its digital representation. The features of this sigma-delta ADC include:

- 10-bit resolution
- Eight single-ended analog input sources are multiplexed with general-purpose I/O ports
- Interrupt upon conversion complete
- Bandgap generated internal voltage reference generator with two selectable levels
- Factory offset and gain calibration

Architecture

Figure 19 displays the major functional blocks of the ADC. An analog multiplexer network selects the ADC input from the available analog pins, ANA0 through ANA7.

is not in DEBUG mode or if the Flash 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 0). If the device is not in DEBUG mode or if the Flash 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 0). If the device is not in DEBUG mode or if the Flash 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
```

- **Read Data Memory (0DH)**—The Read Data Memory command reads from Data Memory. This command is equivalent to the `LDE` and `LDEI` instructions. Data can be read 1 to 65536 bytes at a time (65536 bytes can be read by setting size to 0). If the device is not in DEBUG mode, this command returns `FFH` for the data.

```
DBG ← 0DH
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
```

Table 114. Rotate and Shift Instructions (Continued)

Mnemonic	Operands	Instruction
SRA	dst	Shift Right Arithmetic
SRL	dst	Shift Right Logical
SWAP	dst	Swap Nibbles

eZ8 CPU Instruction Summary

Table 115 summarizes the eZ8 CPU instructions. The table identifies the addressing modes employed by the instruction, the effect upon the Flags register, the number of CPU clock cycles required for the instruction fetch, and the number of CPU clock cycles required for the instruction execution.

Table 115. eZ8 CPU Instruction Summary

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
ADC dst, src	dst ← dst + src + C	r	r	12	*	*	*	*	0	*	2	3
		r	Ir	13							2	4
		R	R	14							3	3
		R	IR	15							3	4
		R	IM	16							3	3
		IR	IM	17							3	4
ADCX dst, src	dst ← dst + src + C	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							4	3
ADD dst, src	dst ← dst + src	r	r	02	*	*	*	*	0	*	2	3
		r	Ir	03							2	4
		R	R	04							3	3
		R	IR	05							3	4
		R	IM	06							3	3
		IR	IM	07							3	4
ADDX dst, src	dst ← dst + src	ER	ER	08	*	*	*	*	0	*	4	3
		ER	IM	09							4	3
Flags Notation:	* = Value is a function of the result of the operation. – = Unaffected X = Undefined				0 = Reset to 0 1 = Set to 1							

Table 115. 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
SUBX dst, src	dst ← dst – src	ER	ER	28	*	*	*	*	1	*	4	3
		ER	IM	29							4	3
SWAP dst	dst[7:4] ↔ dst[3:0]	R		F0	X	*	*	X	–	–	2	2
		IR		F1							2	3
TCM dst, src	(NOT dst) AND src	r	r	62	–	*	*	0	–	–	2	3
		r	Ir	63							2	4
		R	R	64							3	3
		R	IR	65							3	4
		R	IM	66							3	3
		IR	IM	67							3	4
TCMX dst, src	(NOT dst) AND src	ER	ER	68	–	*	*	0	–	–	4	3
		ER	IM	69							4	3
TM dst, src	dst AND src	r	r	72	–	*	*	0	–	–	2	3
		r	Ir	73							2	4
		R	R	74							3	3
		R	IR	75							3	4
		R	IM	76							3	3
		IR	IM	77							3	4
TMX dst, src	dst AND src	ER	ER	78	–	*	*	0	–	–	4	3
		ER	IM	79							4	3
TRAP Vector	SP ← SP – 2 @SP ← PC SP ← SP – 1 @SP ← FLAGS PC ← @Vector		Vector	F2	–	–	–	–	–	–	2	6
WDT				5F	–	–	–	–	–	–	1	2
Flags Notation:	* = Value is a function of the result of the operation. – = Unaffected X = Undefined				0 = Reset to 0 1 = Set to 1							

Table 116. Opcode Map Abbreviations

Abbreviation	Description	Abbreviation	Description
b	Bit position	IRR	Indirect Register Pair
cc	Condition code	p	Polarity (0 or 1)
X	8-bit signed index or displacement	r	4-bit Working Register
DA	Destination address	R	8-bit register
ER	Extended Addressing register	r1, R1, Ir1, Irr1, IR1, rr1, RR1, IRR1, ER1	Destination address
IM	Immediate data value	r2, R2, Ir2, Irr2, IR2, rr2, RR2, IRR2, ER2	Source address
Ir	Indirect Working Register	RA	Relative
IR	Indirect register	rr	Working Register Pair
Irr	Indirect Working Register Pair	RR	Register Pair

Lower Nibble (Hex)

Upper Nibble (Hex)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7	3, 2 **PUSH** IM															
8																
9																
A			3.3 **CPC** r1,r2	3.4 **CPC** r1,Ir2	4.3 **CPC** R2,R1	4.4 **CPC** IR2,R1	4.3 **CPC** R1,IM	4.4 **CPC** IR1,IM	5.3 **CPCX** ER2,ER1	5.3 **CPCX** IM,ER1						
B																
C	3.2 **SRL** R1	3.3 **SRL** IR1														
D																
E									5, 4 **LDWX** ER2,ER1							
F																

Figure 28. Second Opcode Map after 1FH

T

U

V

W