



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	Active
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, WDT
Number of I/O	6
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	8-VDFN Exposed Pad
Supplier Device Package	8-QFN (5x6)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f041aqb020eg

Email: info@E-XFL.COM

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

Pin Description

The Z8 Encore! XP F082A Series products are available in a variety of packages styles and pin configurations. This chapter describes the signals and available pin configurations for each of the package styles. For information about physical package specifications, see the <u>Packaging</u> chapter on page 245.

Available Packages

The following package styles are available for each device in the Z8 Encore! XP F082A Series product line:

- SOIC: 8-, 20- and 28-pin
- PDIP: 8-, 20- and 28-pin
- SSOP: 20- and 28- pin
- QFN 8-pin (MLF-S, a QFN-style package with an 8-pin SOIC footprint)

In addition, the Z8 Encore! XP F082A Series devices are available both with and without advanced analog capability (ADC, temperature sensor and op amp). Devices Z8F082A, Z8F042A, Z8F022A and Z8F012A contain the advanced analog, while devices Z8F081A, Z8F041A, Z8F021A and Z8F011A do not have the advanced analog capability.

Pin Configurations

Figure 2 through Figure 4 display the pin configurations for all the packages available in the Z8 Encore! XP F082A Series. See <u>Table 2</u> on page 10 for a description of the signals. The analog input alternate functions (ANA*x*) are not available on the Z8F081A, Z8F041A, Z8F021A and Z8F011A devices. The analog supply pins (AV_{DD} and AV_{SS}) are also not available on these parts and are replaced by PB6 and PB7.

At reset, all Port A, B and C pins default to an input state. In addition, any alternate functionality is not enabled, so the pins function as general purpose input ports until programmed otherwise. At powerup, the PD0 pin defaults to the **RESET** alternate function.

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

Reset, Stop Mode Recovery and Low Voltage Detection

The Reset Controller within the Z8 Encore! XP F082A Series controls Reset and Stop Mode Recovery operation and provides indication of low supply voltage conditions. In typical operation, the following events cause a Reset:

- Power-On Reset (POR)
- Voltage Brown-Out (VBO)
- Watchdog Timer time-out (when configured by the WDT_RES Flash option bit to initiate a reset)
- External **RESET** pin assertion (when the alternate **RESET** function is enabled by the GPIO Register)
- On-chip debugger initiated Reset (OCDCTL[0] set to 1)

When the device is in STOP Mode, a Stop Mode Recovery is initiated by either of the following occurrences:

- Watchdog Timer time-out
- GPIO Port input pin transition on an enabled Stop Mode Recovery source

The low voltage detection circuitry on the device (available on the 8-pin product versions only) performs the following functions:

- Generates the VBO reset when the supply voltage drops below a minimum safe level.
- Generates an interrupt when the supply voltage drops below a user-defined level (8-pin devices only).

Reset Types

The Z8 Encore! XP F082A Series provides several different types of Reset operation. Stop Mode Recovery is considered as a form of Reset. Table 8 lists the types of Reset and their operating characteristics. The System Reset is longer if the external crystal oscillator is enabled by the Flash option bits, allowing additional time for oscillator start-up.

without initiating an interrupt (if enabled for that pin).

Stop Mode Recovery Using the External RESET Pin

When the Z8 Encore! XP F082A Series device is in STOP Mode and the external **RESET** pin is driven Low, a system reset occurs. Because of a glitch filter operating on the **RESET** pin, the Low pulse must be greater than the minimum width specified, or it is ignored. See the <u>Electrical Characteristics</u> chapter on page 226 for details.

Low Voltage Detection

In addition to the Voltage Brown-Out (VBO) Reset described above, it is also possible to generate an interrupt when the supply voltage drops below a user-selected value. For details about configuring the Low Voltage Detection (LVD) and the threshold levels available, see the <u>Trim Option Bits at Address 0003H (TLVD) Register</u> on page 166. The LVD function is available on the 8-pin product versions only.

When the supply voltage drops below the LVD threshold, the LVD bit of the Reset Status (RSTSTAT) Register is set to one. This bit remains one until the low-voltage condition goes away. Reading or writing this bit does not clear it. The LVD circuit can also generate an interrupt when so enabled, see the <u>GPIO Mode Interrupt Controller</u> chapter on page 55. The LVD bit is not latched; therefore, enabling the interrupt is the only way to guarantee detection of a transient low voltage event.

The LVD functionality depends on circuitry shared with the VBO block; therefore, disabling the VBO also disables the LVD.

Reset Register Definitions

The following sections define the Reset registers.

Reset Status Register

The read-only Reset Status (RSTSTAT) Register, shown in Table 11, 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 write-only Watchdog Timer Control Register.

Table 12 lists the bit settings for Reset and Stop Mode Recovery events.

Port A–D Control Registers

The Port A–D Control registers set the GPIO port operation. The value in the corresponding Port A–D Address Register determines which subregister is read from or written to by a Port A–D Control Register transaction; see Table 20.

Bit	7	6	5	4	3	2	1	0
Field		PCTL						
RESET				00)H			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address			FI	D1H, FD5H,	FD9H, FDD	Н		

Table 20. Port A–D Control Registers (PxCTL)

Bit	Description
[7:0]	Port Control
PCTLx	The Port Control Register provides access to all subregisters that configure the GPIO port operation.

Note: x indicates the specific GPIO port pin number (7–0).

Port A–D Data Direction Subregisters

The Port A–D Data Direction subregister is accessed through the Port A–D Control Register by writing 01H to the Port A–D Address Register; see Table 21.

Bit	7	6	5	4	3	2	1	0
Field	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
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
Address	If 01H ir	n Port A–D A	Address Reg	gister, acces	sible throug	h the Port A	–D Control I	Register

Table 21. Port A–D Data Direction Subregisters (PxDD)

Bit	Description
[7:0]	Data Direction
DDx	These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction Register setting.
	1 = Input. The port pin is sampled and the value written into the Port A–D Input Data Register.
	The output driver is tristated.
Note:	x indicates the specific GPIO port pin number (7–0).

49

Port A–D Stop Mode Recovery Source Enable Subregisters

The Port A–D Stop Mode Recovery Source Enable Subregister, shown in Table 25, is accessed through the Port A–D Control Register by writing 05H to the Port A–D Address Register. Setting the bits in the Port A–D Stop Mode Recovery Source Enable subregisters to 1 configures the specified port pins as a Stop Mode Recovery source. During STOP Mode, any logic transition on a port pin enabled as a Stop Mode Recovery source initiates Stop Mode Recovery.

Table 25. Port A–D Stop Mode Recovery Source Enable Subregisters (PxSMRE)

Bit	7	6	5	4	3	2	1	0
Field	PSMRE7	PSMRE6	PSMRE5	PSMRE4	PSMRE3	PSMRE2	PSMRE1	PSMRE0
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	If 05H i	n Port A–D A	Address Reg	gister, acces	sible throug	h the Port A	–D Control I	Register

Bit Description

[7:0] **Port Stop Mode Recovery Source Enabled**

PSMREx 0 = The port pin is not configured as a Stop Mode Recovery source. Transitions on this pin during STOP Mode do not initiate Stop Mode Recovery.

1 = The port pin is configured as a Stop Mode Recovery source. Any logic transition on this pin during STOP Mode initiates Stop Mode Recovery.

Note: x indicates the specific GPIO port pin number (7-0).

Shared Interrupt Select Register

The Shared Interrupt Select (IRQSS) Register, shown in Table 48, determines the source of the PADxS interrupts. The Shared Interrupt Select Register selects between Port A and alternate sources for the individual interrupts.

Because these shared interrupts are edge-triggered, it is possible to generate an interrupt just by switching from one shared source to another. For this reason, an interrupt must be disabled before switching between sources.

Bit	7	6	5	4	3	2	1	0		
Field	PA7VS	PA6CS			Rese	erved				
RESET	0	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				FC	EH					
Bit	Descriptio	Description								
[7] PA7VS	PA7/LVD S 0 = PA7 is 0 1 = The LV	Selection used for the D is used fo	interrupt for r the interrup	PA7VS inte ot for PA7VS	rrupt reques interrupt re	st. quest.				
[6] PA6CS	 PA6/Comparator Selection 0 = PA6 is used for the interrupt for PA6CS interrupt request. 1 = The Comparator is used for the interrupt for PA6CS interrupt request. 									
[5:0]	Reserved									

Table 48. Shared Interrupt Select Register (IRQSS)

These bits are reserved and must be programmed to 000000.

- 3. Write to the Timer Reload High and Low Byte registers to set the reload value.
- 4. Clear the Timer PWM High and Low Byte registers to 0000H. This allows the software to determine if interrupts were generated by either a capture event or a reload. If the PWM High and Low Byte registers still contain 0000H after the interrupt, the interrupt was generated by a Reload.
- 5. Enable the timer interrupt, if appropriate and set the timer interrupt priority by writing to the relevant interrupt registers. By default, the timer interrupt is generated for both input capture and reload events. If appropriate, configure the timer interrupt to be generated only at the input capture event or the reload event by setting TICONFIG field of the TxCTL0 Register.
- 6. Configure the associated GPIO port pin for the Timer Input alternate function.
- 7. Write to the Timer Control Register to enable the timer and initiate counting.

In CAPTURE Mode, the elapsed time from timer start to Capture event can be calculated using the following equation:

Capture Elapsed Time (s) = $\frac{(Capture Value - Start Value) \times Prescale}{System Clock Frequency (Hz)}$

COMPARE Mode

In COMPARE Mode, the timer counts up to the 16-bit maximum Compare value stored in the Timer Reload High and Low Byte registers. The timer input is the system clock. Upon reaching the Compare value, the timer generates an interrupt and counting continues (the timer value is not reset to 0001H). Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) upon Compare.

If the Timer reaches FFFFH, the timer rolls over to 0000H and continue counting.

Observe the following steps for configuring a timer for COMPARE Mode and initiating the count:

- 1. Write to the Timer Control Register to:
 - Disable the timer
 - Configure the timer for COMPARE Mode
 - Set the prescale value

- Set the Capture edge (rising or falling) for the Timer Input
- 2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H).
- 3. Write to the Timer Reload High and Low Byte registers to set the Compare value.
- 4. Enable the timer interrupt, if appropriate and set the timer interrupt priority by writing to the relevant interrupt registers.By default, the timer interrupt are generated for both input capture and reload events. If appropriate, configure the timer interrupt to be generated only at the input capture event or the reload event by setting TICONFIG field of the TxCTL0 Register.
- 5. Configure the associated GPIO port pin for the Timer Input alternate function.
- 6. Write to the Timer Control Register to enable the timer.
- 7. Counting begins on the first appropriate transition of the Timer Input signal. No interrupt is generated by this first edge.

In CAPTURE/COMPARE Mode, the elapsed time from timer start to Capture event can be calculated using the following equation:

Capture Elapsed Time (s) = $\frac{(Capture Value - Start Value) \times Prescale}{System Clock Frequency (Hz)}$

Reading the Timer Count Values

The current count value in the timers can be read while counting (enabled). This capability has no effect on timer operation. When the timer is enabled and the Timer High Byte Register is read, the contents of the Timer Low Byte Register are placed in a holding register. A subsequent read from the Timer Low Byte Register returns the value in the holding register. This operation allows accurate reads of the full 16-bit timer count value while enabled. When the timers are not enabled, a read from the Timer Low Byte Register returns the actual value in the counter.

Timer Pin Signal Operation

The timer output function is a GPIO port pin alternate function. The Timer Output is toggled every time the counter is reloaded.

Z8 Encore! XP[®] F082A Series **Product Specification**



Figure 19. Analog-to-Digital Converter Block Diagram

Operation

In both SINGLE-ENDED and DIFFERENTIAL modes, the effective output of the ADC is an 11-bit, signed, two's complement digital value. In DIFFERENTIAL Mode, the ADC can output values across the entire 11-bit range, from -1024 to +1023. In SINGLE-ENDED Mode, the output generally ranges from 0 to +1023, but offset errors can cause small negative values.

The following code example illustrates how to safely enable the comparator:

Comparator Control Register Definition

The Comparator Control Register (CMP0) configures the comparator inputs and sets the value of the internal voltage reference.

Bit	7	6	5	4	3	2	1	0
Field	INPSEL	INNSEL		REF	LVL		Reserved (REFLVL	20-/28-pin) _ (8-pin)
RESET	0	0	0	1	0	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address				F9	0H			

Table 77. Comparator Control Register (CMP0)

Bit	Description
[7] INPSEL	Signal Select for Positive Input 0 = GPIO pin used as positive comparator input. 1 = Temperature sensor used as positive comparator input.
[6] INNSEL	Signal Select for Negative Input 0 = Internal reference disabled, GPIO pin used as negative comparator input. 1 = Internal reference enabled as negative comparator input.

Caution: The byte at each address of the Flash memory cannot be programmed (any bits written to 0) more than twice before an erase cycle occurs. Doing so may result in corrupted data at the target byte.

Page Erase

The Flash memory can be erased one page (512 bytes) at a time. Page Erasing the Flash memory sets all bytes in that page to the value FFH. The Flash Page Select Register identifies the page to be erased. Only a page residing in an unprotected sector can be erased. With the Flash Controller unlocked and the active page set, writing the value 95h to the Flash Control Register initiates the Page Erase operation. While the Flash Controller executes the Page Erase operation, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate. The eZ8 CPU resumes operation after the Page Erase operation completes. If the Page Erase operation is performed using the On-Chip Debugger, poll the Flash Status Register to determine when the Page Erase operation is complete. When the Page Erase is complete, the Flash Controller returns to its locked state.

Mass Erase

The Flash memory can also be Mass Erased using the Flash Controller, but only by using the On-Chip Debugger. Mass Erasing the Flash memory sets all bytes to the value FFH. With the Flash Controller unlocked and the Mass Erase successfully enabled, writing the value 63H to the Flash Control Register initiates the Mass Erase operation. While the Flash Controller executes the Mass Erase operation, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate. Using the On-Chip Debugger, poll the Flash Status Register to determine when the Mass Erase operation is complete. When the Mass Erase is complete, the Flash Controller returns to its locked state.

Flash Controller Bypass

The Flash Controller can be bypassed and the control signals for the Flash memory brought out to the GPIO pins. Bypassing the Flash Controller allows faster Row Programming algorithms by controlling the Flash programming signals directly.

Row programming is recommended for gang programming applications and large volume customers who do not require in-circuit initial programming of the Flash memory. Page Erase operations are also supported when the Flash Controller is bypassed.

For more information about bypassing the Flash Controller, refer to the <u>Third-Party Flash</u> <u>Programming Support for Z8 Encore! MCUs Application Note (AN0117)</u>, which is available for download on <u>www.zilog.com</u>. These serial numbers are stored in the Flash information page and are unaffected by mass erasure of the device's Flash memory. See the Reading the Flash Information Page section below and the <u>Serialization Data section on page 173</u> for more details.

Randomized Lot Identification Bits

As an optional feature, Zilog is able to provide a factory-programmed random lot identifier. With this feature, all devices in a given production lot are programmed with the same random number. This random number is uniquely regenerated for each successive production lot and is not likely to be repeated.

The randomized lot identifier is a 32 byte binary value, stored in the Flash information page and is unaffected by mass erasure of the device's Flash memory. See Reading the Flash Information Page, below, and the <u>Randomized Lot Identifier section on page 174</u> for more details.

Reading the Flash Information Page

The following code example shows how to read data from the Flash information area.

; get value at info address 60 (FE60h) ldx FPS, #%80 ; enable access to flash info page ld R0, #%FE ld R1, #%60 ldc R2, @RR0 ; R2 now contains the calibration value

Flash Option Bit Control Register Definitions

This section briefly describes the features of the Trim Bit Address and Data registers.

Trim Bit Address Register

The Trim Bit Address (TRMADR) Register contains the target address for an access to the trim option bits (Table 86).

Bit	7	6	5	4	3	2	1	0
Field		TRMADR: Trim Bit Address (00H to 1FH)						
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				FF	6H			

Table 86. Trim Bit Address Register (TRMADR)

ADC Calibration Data

Table 96. ADC Calibration Bits

Bit	7	6	5	4	3	2	1	0
Field				ADC	_CAL			
RESET	U	U	U	U	U	U	U	U
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address			Informati	on Page Me	mory 0060H	1-007DH		
Noto: II -	Linchanged k		/ - Pood/Mrit	0				

Note: U = Unchanged by Reset. R/W = Read/Write.

Bit Description

[7:0] Analog-to-Digital Converter Calibration Values
 ADC_CAL Contains factory-calibrated values for ADC gain and offset compensation. Each of the ten supported modes has one byte of offset calibration and two bytes of gain calibration. These values are read by the software to compensate ADC measurements as described in the Software Compensation Procedure Using Factory Calibration Data section on page 129. The location of each calibration byte is provided in Table 97.

Info Page	Memory			
Address	Address	Compensation Usage	ADC Mode	Reference Type
60	FE60	Offset	Single-Ended Unbuffered	Internal 2.0 V
08	FE08	Gain High Byte	Single-Ended Unbuffered	Internal 2.0 V
09	FE09	Gain Low Byte	Single-Ended Unbuffered	Internal 2.0 V
63	FE63	Offset	Single-Ended Unbuffered	Internal 1.0 V
0A	FE0A	Gain High Byte	Single-Ended Unbuffered	Internal 1.0 V
0B	FE0B	Gain Low Byte	Single-Ended Unbuffered	Internal 1.0 V
66	FE66	Offset	Single-Ended Unbuffered	External 2.0 V
0C	FE0C	Gain High Byte	Single-Ended Unbuffered	External 2.0 V
0D	FE0D	Gain Low Byte	Single-Ended Unbuffered	External 2.0 V
69	FE69	Offset	Single-Ended 1x Buffered	Internal 2.0 V
0E	FE0E	Gain High Byte	Single-Ended 1x Buffered	Internal 2.0 V
0F	FE0F	Gain Low Byte	Single-Ended 1x Buffered	Internal 2.0 V
6C	FE6C	Offset	Single-Ended 1x Buffered	External 2.0 V
10	FE10	Gain High Byte	Single-Ended 1x Buffered	External 2.0 V
11	FE11	Gain Low Byte	Single-Ended 1x Buffered	External 2.0 V
6F	FE6F	Offset	Differential Unbuffered	Internal 2.0 V

Table 97. ADC Calibration Data Location

Temperature Sensor Calibration Data

Table 98. Temperature Sensor Calibration High Byte at 003A (TSCALH)

Bit	7	6	5	4	3	2	1	0	
Field	TSCALH								
RESET									
R/W	R/W	R/W R/W R/W R/W R/W R/W							
Address	Information Page Memory 003A								
Note: U =	Note: U = Unchanged by Reset. R/W = Read/Write.								

Bit Description [7:0] **Temperature Sensor Calibration High Byte** TSCALH The TSCALH and TSCALL bytes combine to form the 12-bit temperature sensor offset calibra-

tion value. For more details, see Temperature Sensor Operation on page 139.

Table 99. Temperature Sensor Calibration Low Byte at 003B (TSCALL)

Bit	7	6	5	4	3	2	1	0	
Field	TSCALL								
RESET	U	U							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Address	Information Page Memory 003B								
Note: U =	Note: U = Unchanged by Reset. R/W = Read/Write.								

Bit Description

[7:0]	Temperature Sensor Calibration Low Byte
TSCALL	The TSCALH and TSCALL bytes combine to form the 12-bit temperature sensor offset calibra-
	tion value. For usage details, see the <u>Temperature Sensor Operation</u> section on page 144.

host is the character 80H. The character 80H has eight continuous bits Low (one Start bit plus 7 data bits), framed between High bits. The Auto-Baud Detector measures this period and sets the OCD Baud Rate Generator accordingly.

The Auto-Baud Detector/Generator is clocked by the system clock. The minimum baud rate is the system clock frequency divided by 512. For optimal operation with asynchronous datastreams, the maximum recommended baud rate is the system clock frequency divided by 8. The maximum possible baud rate for asynchronous datastreams is the system clock frequency divided by 4, but this theoretical maximum is possible only for low noise designs with clean signals. Table 108 lists minimum and recommended maximum baud rates for sample crystal frequencies.

System Clock Frequency (MHz)	Recommended Maximum Baud Rate (Kbps)	Recommended Standard PC Baud Rate (bps)	Minimum Baud Rate (Kbps)
20.0	2500.0	1,843,200	39
1.0	125.0	115,200	1.95
0.032768 (32kHz)	4.096	2,400	0.064

Table 108. OCD Baud-Rate Limits

If the OCD receives a Serial Break (nine or more continuous bits Low) the Auto-Baud Detector/Generator resets. Reconfigure the Auto-Baud Detector/Generator by sending 80H.

OCD Serial Errors

The On-Chip Debugger can detect any of the following error conditions on the DBG pin:

- Serial Break (a minimum of nine continuous bits Low)
- Framing Error (received Stop bit is Low)
- Transmit Collision (OCD and host simultaneous transmission detected by the OCD)

When the OCD detects one of these errors, it aborts any command currently in progress, transmits a four character long Serial Break back to the host and resets the Auto-Baud Detector/Generator. A Framing Error or Transmit Collision may be caused by the host sending a Serial Break to the OCD. Because of the open-drain nature of the interface, returning a Serial Break break back to the host only extends the length of the Serial Break if the host releases the Serial Break early.

The host transmits a Serial Break on the DBG pin when first connecting to the Z8 Encore! XP F082A Series devices or when recovering from an error. A Serial Break from the host resets the Auto-Baud Generator/Detector but does not reset the OCD Control Register. A

Z8 Encore! XP[®] F082A Series Product Specification



Figure 29. Typical RC Oscillator Frequency as a Function of the External Capacitance with a $45k\Omega$ Resistor

Caution: When using the external RC oscillator mode, the oscillator can stop oscillating if the power supply drops below 2.7 V, but before the power supply drops to the Voltage Brown-Out threshold. The oscillator resumes oscillation when the supply voltage exceeds 2.7 V.

202

Mnemonic	Operands	Instruction
BCLR	bit, dst	Bit Clear
BIT	p, bit, dst	Bit Set or Clear
BSET	bit, dst	Bit Set
BSWAP	dst	Bit Swap
CCF	—	Complement Carry Flag
RCF	_	Reset Carry Flag
SCF	—	Set Carry Flag
ТСМ	dst, src	Test Complement Under Mask
ТСМХ	dst, src	Test Complement Under Mask using Extended Addressing
ТМ	dst, src	Test Under Mask
TMX	dst, src	Test Under Mask using Extended Addressing

Table 121. Bit Manipulation Instructions

Table 122. Block Transfer Instructions

Mnemonic	Operands	Instruction
LDCI	dst, src	Load Constant to/from Program Memory and Auto- Increment Addresses
LDEI	dst, src	Load External Data to/from Data Memory and Auto- Increment Addresses

Table 123. CPU Control Instructions

Mnemonic	Operands	Instruction
ATM		Atomic Execution
CCF		Complement Carry Flag
DI	_	Disable Interrupts
EI	_	Enable Interrupts
HALT	_	Halt Mode
NOP		No Operation

Mnemonic	Operands	Instruction
BRK	—	On-Chip Debugger Break
BTJ	p, bit, src, DA	Bit Test and Jump
BTJNZ	bit, src, DA	Bit Test and Jump if Non-Zero
BTJZ	bit, src, DA	Bit Test and Jump if Zero
CALL	dst	Call Procedure
DJNZ	dst, src, RA	Decrement and Jump Non-Zero
IRET	—	Interrupt Return
JP	dst	Jump
JP cc	dst	Jump Conditional
JR	DA	Jump Relative
JR cc	DA	Jump Relative Conditional
RET	—	Return
TRAP	vector	Software Trap

Table 126. Program Control Instructions

Table 127. Rotate and Shift Instructions

Mnemonic	Operands	Instruction
BSWAP	dst	Bit Swap
RL	dst	Rotate Left
RLC	dst	Rotate Left through Carry
RR	dst	Rotate Right
RRC	dst	Rotate Right through Carry
SRA	dst	Shift Right Arithmetic
SRL	dst	Shift Right Logical
SWAP	dst	Swap Nibbles

eZ8 CPU Instruction Summary

Table 128 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.

Assembly		Add Mc	ress ode	_ Opcode(s)			Fla	ags			Fetch Cvcle	Instr. Cvcle
Mnemonic	Symbolic Operation	dst	src	(Hex)	С	Ζ	S	۷	D	Н	S	S
ADC dst, src	$dst \gets dst + src + C$	r	r	12	*	*	*	*	0	*	2	3
		r	lr	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	$dst \gets dst + src + C$	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							4	3
ADD dst, src	$dst \leftarrow dst + src$	r	r	02	*	*	*	*	0	*	2	3
		r	lr	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	$dst \gets dst + src$	ER	ER	08	*	*	*	*	0	*	4	3
		ER	IM	09	-						4	3

Table 128. eZ8 CPU Instruction Summary

Note: Flags Notation: * = Value is a function of the result of the operation.

- = Unaffected.

X = Undefined.

0 = Reset to 0.

1 = Set to 1.

LD 210 LDC 210 LDCI 209, 210 LDE 210 **LDEI 209** LDX 210 LEA 210 logical 210 **MULT 208 NOP 209** OR 210 **ORX 210 POP 210** POPX 210 program control 211 **PUSH 210** PUSHX 210 RCF 209, 210 **RET 211** RL 211 **RLC 211** rotate and shift 211 RR 211 **RRC 211 SBC 208** SCF 209, 210 SRA 211 SRL 211 **SRP 210 STOP 210 SUB 208 SUBX 208 SWAP 211 TCM 209 TCMX 209** TM 209 TMX 209 **TRAP 211** Watchdog Timer refresh 210 **XOR 210 XORX 210** instructions, eZ8 classes of 207 interrupt control register 69 interrupt controller 55

architecture 55 interrupt assertion types 58 interrupt vectors and priority 58 operation 57 register definitions 60 software interrupt assertion 59 interrupt edge select register 67 interrupt request 0 register 60 interrupt request 1 register 61 interrupt request 2 register 62 interrupt return 211 interrupt vector listing 55 interrupts **UART 108** IR 206 Ir 206 IrDA architecture 120 block diagram 120 control register definitions 123 operation 120 receiving data 122 transmitting data 121 **IRET 211** IRQ0 enable high and low bit registers 62 IRQ1 enable high and low bit registers 64 IRQ2 enable high and low bit registers 65 **IRR 206** Irr 206

J

JP 211 jump, conditional, relative, and relative conditional 211

L

LD 210 LDC 210 LDCI 209, 210 LDE 210 LDEI 209, 210 LDX 210