E·XFL



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

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

Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	5MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, PWM, WDT
Number of I/O	16
Program Memory Size	1KB (1K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 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	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0113hh005ec

Email: info@E-XFL.COM

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

	07
Receiving Data using the Interrupt Driven Method	. 97
Clear To Sond (CTS) Operation	. 98
	. 99
	101
	101
	103
	100
	104
	104
UART Status 0 Register	105
UART Status 1 Register	106
UART Control 0 and Control 1 Registers	107
UART Address Compare Register	109
UART Baud Rate High and Low Byte Registers	110
Infrared Encoder/Decoder	113
Architecture	113
Operation	113
Transmitting IrDA Data	114
Receiving IrDA Data	115
Infrared Encoder/Decoder Control Register Definitions	116
Analog-to-Digital Converter	447
	117
Architecture	117 117
Architecture	117 117 118
Architecture	117 117 118 118
Architecture Operation Operation Data Format Automatic Powerdown Operation	 117 117 118 118 119
Architecture Operation Operation Data Format Automatic Powerdown Single-Shot Conversion	 117 117 118 118 119 119 119
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion	 117 117 118 118 119 119 120
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts	117 117 118 118 119 119 120 121
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation	 117 117 118 118 119 119 120 121 121
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation ADC Control Register Definitions	 117 117 118 118 119 119 120 121 121 122
Architecture Operation Operation Data Format Data Format Automatic Powerdown Single-Shot Conversion Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 Operation	 117 117 118 118 119 119 120 121 121 122 122 122
Architecture Operation Operation Data Format Automatic Powerdown Single-Shot Conversion Single-Shot Conversion Continuous Conversion Interrupts Interrupts Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 ADC Control/Status Register 1	 117 117 118 118 119 119 120 121 121 122 122 124
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 ADC Control/Status Register 1 ADC Data High Byte Register	 117 117 118 118 119 120 121 121 122 122 122 124 124
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 ADC Control/Status Register 1 ADC Data High Byte Register ADC Data Low Bits Register	 117 117 118 118 119 119 120 121 121 122 122 122 124 125
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 ADC Control/Status Register 1 ADC Data High Byte Register ADC Data Low Bits Register	117 117 118 118 119 120 121 121 122 122 122 124 124 125 127
Architecture Operation Data Format Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 ADC Control/Status Register 1 ADC Data High Byte Register ADC Data Low Bits Register Operation	117 117 118 118 119 120 121 121 122 122 122 122 124 124 125 127
Architecture Operation Data Format Automatic Powerdown Single-Shot Conversion Continuous Conversion Interrupts Calibration and Compensation ADC Control Register Definitions ADC Control Register 0 ADC Control/Status Register 1 ADC Data High Byte Register ADC Data Low Bits Register Comparator Operation Comparator Control Register Definitions	117 117 118 118 119 120 121 121 122 122 122 124 125 127 127

Z8 Encore! XP[®] F0823 Series Product Specification

Ordering Information	217
Part Number Suffix Designations	226
Index	227
Customer Support	237

Overview

Zilog's Z8 Encore! XP[®] microcontroller unit (MCU) family of products are the first Zilog[®] microcontroller products based on the 8-bit eZ8 CPU core. Z8 Encore! XP F0823 Series products expand upon Zilog's extensive line of 8-bit microcontrollers. The Flash in-circuit programming capability allows for faster development time and program changes in the field. The new eZ8 CPU is upward compatible with existing Z8[®] instructions. The rich peripheral set of Z8 Encore! XP F0823 Series makes it suitable for a variety of applications including motor control, security systems, home appliances, personal electronic devices, and sensors.

Features

The key features of Z8 Encore! XP F0823 Series include:

- 5 MHz eZ8 CPU
- 1 KB, 2 KB, 4 KB, or 8 KB Flash memory with in-circuit programming capability
- 256 B, 512 B, or 1 KB register RAM
- 6 to 24 I/O pins depending upon package
- Internal precision oscillator (IPO)
- Full-duplex UART
- The universal asynchronous receiver/transmitter (UART) baud rate generator (BRG) can be configured and used as a basic 16-bit timer
- Infrared data association (IrDA)-compliant infrared encoder/decoders, integrated with UART
- Two enhanced 16-bit timers with capture, compare, and PWM capability
- Watchdog Timer (WDT) with dedicated internal RC oscillator
- On-Chip Debugger (OCD)
- Optional 8-channel, 10-bit Analog-to-Digital Converter (ADC)
- On-Chip analog comparator
- Up to 20 vectored interrupts
- Direct LED drive with programmable drive strengths
- Voltage Brownout (VBO) protection
- Power-On Reset (POR)

Reset Controller

Z8 Encore! $XP^{\mathbb{R}}$ F0823 Series products can be reset using the \overline{RESET} pin, POR, WDT time-out, STOP mode exit, or Voltage Brownout warning signal. The \overline{RESET} pin is bidirectional, that is, it functions as reset source as well as a reset indicator.

On-Chip Debugger

Z8 Encore! XP F0823 Series products feature an integrated On-Chip Debugger. The OCD provides a rich-set of debugging capabilities, such as reading and writing registers, programming Flash memory, setting breakpoints and executing code. A single-pin interface provides communication to the OCD.

Program Memory Address (Hex)	Function
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–003D	Oscillator Fail Traps*
003E-03FF	Program Memory
*See Table 33 on page 54 for a list of the int	terrupt vectors and traps.

Table 6. Z8 Encore! XP F0823 Series Program Memory Maps (Continued)

Data Memory

Z8 Encore! XP[®] F0823 Series does not use the eZ8 CPU's 64 KB Data Memory address space.

Flash Information Area

Table 7 lists the Z8 Encore! XP F0823 Series Flash Information Area. This 128 B Information Area is accessed by setting bit 7 of the Flash Page Select Register to 1. When access is enabled, the Flash Information Area is mapped into the Program Memory and overlays the 128 bytes at addresses FE00H to FF7FH. When the Information Area access is enabled, all reads from these Program Memory addresses return the Information Area data rather than the Program Memory data. Access to the Flash Information Area is read-only.

Program Memory Address	
(Hex)	Function
FE00–FE3F	Zilog Option Bits.
FE40–FE53	Part Number. 20-character ASCII alphanumeric code Left justified and filled with FH.
FE54–FE5F	Reserved.
FE60–FE7F	Zilog Calibration Data.
FE80–FFFF	Reserved.

Table 7. Z8 Encore! XP F0823 Series Flash Memory Information Area Map

Z8 Encore! XP[®] F0823 Series Product Specification

PSMRE[7:0]—Port Stop Mode Recovery Source Enabled.

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.

Port A–C Pull-up Enable Sub-Registers

The Port A–C Pull-up Enable sub-register (Table 25) is accessed through the Port A–C Control register by writing 06H to the Port A–C Address register. Setting the bits in the Port A–C Pull-up Enable sub-registers enables a weak internal resistive pull-up on the specified Port pins.

Table 25. Port A–C Pull-Up Enable Sub-Registers (PxPUE)

BITS	7	6	5	4	3	2	1	0
FIELD	PPUE7	PPUE6	PPUE5	PPUE4	PPUE3	PPUE2	PPUE1	PPUE0
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	If 06H in Port A–C Address Register, accessible through the Port A–C Control Register							

PPUE[7:0]—Port Pull-up Enabled

0 = The weak pull-up on the Port pin is disabled.

1 = The weak pull-up on the Port pin is enabled.

Port A–C Alternate Function Set 1 Sub-Registers

The Port A–C Alternate Function Set1 sub-register (Table 26) is accessed through the Port A–C Control register by writing 07H to the Port A–C Address register. The Alternate Function Set 1 sub-registers selects the alternate function available at a port pin. Alternate Functions selected by setting or clearing bits of this register are defined in GPIO Alternate Functions on page 36.

Note:

Alternate function selection on port pins must also be enabled as described in Port A–C Alternate Function Sub-Registers *on page 45*.

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						

Table 48. Interrupt Control Register (IRQCTL)

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

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. Follow the steps below to configure a timer for COMPARE mode and to initiate 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 initial logic level (High or Low) for the Timer Output alternate function, if appropriate.
- 2. Write to the Timer High and Low Byte registers to set the starting count value.
- 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.
- 5. If using the Timer Output function, configure the associated GPIO port pin for the Timer Output alternate function.
- 6. Write to the Timer Control register to enable the timer and initiate counting.

In COMPARE mode, the system clock always provides the timer input. The Compare time can be calculated by the following equation:

COMPARE Mode Time (s) = (Compare Value – Start Value) × Prescale System Clock Frequency (Hz)

GATED Mode

In GATED mode, the timer counts only when the Timer Input signal is in its active state (asserted), as determined by the TPOL bit in the Timer Control register. When the Timer Input signal is asserted, counting begins. A timer interrupt is generated when the Timer Input signal is deasserted or a timer Reload occurs. To determine if a Timer Input signal deassertion generated the interrupt, read the associated GPIO input value and compare to the value stored in the TPOL bit.

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. When reaching the Reload value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes (assuming the Timer Input signal remains asserted). Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state (from Low to High or from High to Low) at timer reset.

BITS	7	6	5	4	3	2	1	0
FIELD				PW	/ML			
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				F05H,	F0DH			

Table 54. Timer 0–1 PWM Low Byte Register (TxPWML)

PWMH and PWML—Pulse-Width Modulator High and Low Bytes These two bytes, {PWMH[7:0], PWML[7:0]}, form a 16-bit value that is compared to the current 16-bit timer count. When a match occurs, the PWM output changes state. The PWM output value is set by the TPOL bit in the Timer Control Register (TxCTL1) register.

The TxPWMH and TxPWML registers also store the 16-bit captured timer value when operating in CAPTURE or CAPTURE/COMPARE modes.

Timer 0–1 Control Registers

Time 0–1 Control Register 0

The Timer Control Register 0 (TxCTL0) and Timer Control Register 1 (TxCTL1) determine the timer operating mode. It also includes a programmable PWM deadband delay, two bits to configure timer interrupt definition, and a status bit to identify if the most recent timer interrupt is caused by an input Capture event.

Table 55. Timer 0–1 Control Register 0 (TxCTL0)

BITS	7	6	5	4	3	2	1	0
FIELD	TMODEHI	TICO	NFIG	Reserved		PWMD		INPCAP
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	F06H, F0EH							

TMODEHI—Timer Mode High Bit

This bit along with the TMODE field in TxCTL1 register determines the operating mode of the timer. This is the most-significant bit of the Timer mode selection value. See the TxCTL1 register description for more details.

TICONFIG—Timer Interrupt Configuration

This field configures timer interrupt definition.

CAPTURE/COMPARE Mode

0 = Counting is started on the first rising edge of the Timer Input signal. The current count is captured on subsequent rising edges of the Timer Input signal.

1 = Counting is started on the first falling edge of the Timer Input signal. The current count is captured on subsequent falling edges of the Timer Input signal.

PWM DUAL OUTPUT Mode

0 = Timer Output is forced Low (0) and Timer Output Complement is forced High (1) when the timer is disabled. When enabled, the Timer Output is forced High (1) upon PWM count match and forced Low (0) upon Reload. When enabled, the Timer Output Complement is forced Low (0) upon PWM count match and forced High (1) upon Reload. The PWMD field in TxCTL0 register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to High (1).

1 = Timer Output is forced High (1) and Timer Output Complement is forced Low (0) when the timer is disabled. When enabled, the Timer Output is forced Low (0) upon PWM count match and forced High (1) upon Reload. When enabled, the Timer Output Complement is forced High (1) upon PWM count match and forced Low (0) upon Reload. The PWMD field in TxCTL0 register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to Low (0).

CAPTURE RESTART Mode

0 = Count is captured on the rising edge of the Timer Input signal

1 = Count is captured on the falling edge of the Timer Input signal

COMPARATOR COUNTER Mode

When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer Reload.

Caution: When the Timer Output alternate function TxOUT on a GPIO port pin is enabled, Tx-OUT changes to whatever state the TPOL bit is in. The timer does not need to be enabled for that to happen. Also, the port data direction sub register is not needed to be set to output on TxOUT. Changing the TPOL bit with the timer enabled and running does not immediately change the TxOUT.

PRES—Prescale value.

The timer input clock is divided by 2^{PRES} , where PRES can be set from 0 to 7. The prescaler is reset each time the Timer is disabled. This reset ensures proper clock division each time the Timer is restarted.

000 = Divide by 1001 = Divide by 2

Z8 Encore! XP[®] F0823 Series Product Specification



No

If the Baud Rate Generator (BRG) interrupt enable is set, the UART Receiver interrupt asserts when the UART Baud Rate Generator reloads. This condition allows the Baud Rate Generator to function as an additional counter if the UART functionality is not employed.

Receiver Ready

Receiver Interrupt

Read Status

Errors?

UART Baud Rate Generator

The UART Baud Rate Generator creates a lower frequency baud rate clock for data transmission. The input to the Baud Rate Generator is the system clock. The UART Baud Rate High and Low Byte registers combine to create a 16-bit baud rate divisor value



Infrared Encoder/Decoder

Z8 Encore! XP[®] F0823 Series products contain a fully-functional, high-performance UART with Infrared Encoder/Decoder (Endec). The Infrared Endec is integrated with an on-chip UART to allow easy communication between the Z8 Encore! XP and IrDA Physical Layer Specification, Version 1.3-compliant infrared transceivers. Infrared communication provides secure, reliable, low-cost, point-to-point communication between PCs, PDAs, cell phones, printers and other infrared enabled devices.

Architecture



Figure 16 displays the architecture of the Infrared Endec.

Figure 16. Infrared Data Communication System Block Diagram

Operation

When the Infrared Endec is enabled, the transmit data from the associated on-chip UART is encoded as digital signals in accordance with the IrDA standard and output to the infrared transceiver through the TXD pin. Similarly, data received from the infrared transceiver is passed to the Infrared Endec through the RXD pin, decoded by the Infrared

REFSELL—Voltage Reference Level Select Low Bit; in conjunction with the High bit (REFSELH) in ADC Control/Status Register 1, this determines the level of the internal voltage reference; the following details the effects of {REFSELH, REFSELL};

Note:

This reference is independent of the Comparator reference.

00= Internal Reference Disabled, reference comes from external pin.

01 = Internal Reference set to 1.0 V

10= Internal Reference set to 2.0 V (default)

REFEXT—External Reference Select

0 = External reference buffer is disabled; V_{ref} pin is available for GPIO functions

1 = The internal ADC reference is buffered and connected to the V_{ref} pin

CONT

0 = Single-shot conversion. ADC data is output once at completion of the 5129 system clock cycles.

1 = Continuous conversion. ADC data updated every 256 system clock cycles.

ANAIN[3:0]—Analog Input Select

These bits select the analog input for conversion. Not all port pins in this list are available in all packages for Z8 Encore! XP[®] F0823 Series. For information on the port pins available with each package style, see Pin Description on page 7. Do not enable unavailable analog inputs. Usage of these bits changes depending on the buffer mode selected in ADC Control/Status Register 1.

For the reserved values, all input switches are disabled to avoid leakage or other undesirable operation. ADC samples taken with reserved bit settings are undefined.

Single-Ended:

0000 = ANA00001 = ANA10010 = ANA20011 = ANA3 0100 = ANA40101 = ANA50110 = ANA60111 = ANA71000 = Reserved1001 = Reserved1010 = Reserved1011 = Reserved1100 = Reserved1101 = Reserved1110 = Reserved1111 = Reserved

configurations. The information contained here is lost when page 0 of the Program Memory is erased.

Trim Option Bits

The trim option bits are contained in the information page of the Flash memory. These bits are factory programmed values required to optimize the operation of onboard analog circuitry and cannot be permanently altered. Program Memory may be erased without endangering these values. It is possible to alter working values of these bits by accessing the Trim Bit Address and Data Registers, but these working values are lost after a power loss or any other reset event.

There are 32 bytes of trim data. To modify one of these values the user code must first write a value between 00H and 1FH into the Trim Bit Address Register. The next write to the Trim Bit Data register changes the working value of the target trim data byte.

Reading the trim data requires the user code to write a value between 00H and 1FH into the Trim Bit Address Register. The next read from the Trim Bit Data register returns the working value of the target trim data byte.

The trim address range is from information address 20-3F only. The remainder of the information page is not accessible through the trim bit address and data registers.

Calibration Option Bits

The calibration option bits are also contained in the information page. These bits are factory programmed values intended for use in software correcting the device's analog performance. To read these values, the user code must employ the LDC instruction to access the information area of the address space as defined in Flash Information Area on page 15

Serialization Bits

As an optional feature, Zilog[®] is able to provide factory-programmed serialization. For serialized products, the individual devices are programmed with unique serial numbers. These serial numbers are binary values, four bytes in length. The numbers increase in size with each device, but gaps in the serial sequence may exist.

These serial numbers are stored in the Flash information page (for more details, see Reading the Flash Information Page on page 143 and Serialization Data on page 148) and are unaffected by mass erasure of the device's Flash memory.

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.

Note:

Table 105. Notational Shorthand

Notation	Description	Operand	Range
b	Bit	b	b represents a value from 0 to 7 (000B to 111B).
СС	Condition Code	—	See Condition Codes overview in the eZ8 CPU User Manual.
DA	Direct Address	Addrs	Addrs represents a number in the range of 0000H to FFFFH.
ER	Extended Addressing Register	Reg	Reg represents a number in the range of 000H to FFFH.
IM	Immediate Data	#Data	Data is a number between 00H to FFH.
lr	Indirect Working Register	@Rn	n = 0–15.
IR	Indirect Register	@Reg	Reg. represents a number in the range of 00H to FFH.
Irr	Indirect Working Register Pair	@RRp	p = 0, 2, 4, 6, 8, 10, 12, or 14.
IRR	Indirect Register Pair	@Reg	Reg represents an even number in the range 00H to FEH
р	Polarity	р	Polarity is a single bit binary value of either 0B or 1B.
r	Working Register	Rn	n = 0–15.
R	Register	Reg	Reg. represents a number in the range of 00H to FFH.
RA	Relative Address	Х	X represents an index in the range of +127 to – 128 which is an offset relative to the address of the next instruction
rr	Working Register Pair	RRp	p = 0, 2, 4, 6, 8, 10, 12, or 14.
RR	Register Pair	Reg	Reg. represents an even number in the range of 00H to FEH.
Vector	Vector Address	Vector	Vector represents a number in the range of 00H to FFH.
X	Indexed	#Index	The register or register pair to be indexed is offset by the signed Index value (#Index) in a +127 to -128 range.

Table 110. CPU Control Instructions (Continued)

Mnemonic	Operands	Instruction
SCF	_	Set Carry Flag
SRP	SIC	Set Register Pointer
STOP	—	STOP Mode
WDT	—	Watchdog Timer Refresh

Table 111. Load Instructions

Mnemonic	Operands	Instruction
CLR	dst	Clear
LD	dst, src	Load
LDC	dst, src	Load Constant to/from Program Memory
LDCI	dst, src	Load Constant to/from Program Memory and Auto-Increment Addresses
LDE	dst, src	Load External Data to/from Data Memory
LDEI	dst, src	Load External Data to/from Data Memory and Auto-Increment Addresses
LDWX	dst, src	Load Word using Extended Addressing
LDX	dst, src	Load using Extended Addressing
LEA	dst, X(src)	Load Effective Address
POP	dst	Рор
POPX	dst	Pop using Extended Addressing
PUSH	SrC	Push
PUSHX	src	Push using Extended Addressing

Table 112. Logical Instructions

Mnemonic	Operands	Instruction
AND	dst, src	Logical AND
ANDX	dst, src	Logical AND using Extended Addressing
COM	dst	Complement
OR	dst, src	Logical OR

• · · · · · · · · · ·		Address Mode		Fla	igs							
Assembly Mnemonic	Symbolic Operation	dst	src	- Opcode(s) (Hex)	С	z	S	v	D	Н	- Fetch Cycles	Instr. Cycles
RR dst		R		E0	*	*	*	*	-	_	2	2
	► D7 D6 D5 D4 D3 D2 D1 D0 ► C	IR		E1							2	3
RRC dst		R		C0	*	*	*	*	_	_	2	2
	► D7 D6 D5 D4 D3 D2 D1 D0 ► C dst	IR		C1	-						2	3
SBC dst, src	dst ← dst – src - C	r	r	32	*	*	*	*	1	*	2	3
		r	lr	33	-						2	4
		R	R	34	-						3	3
		R	IR	35	-						3	4
		R	IM	36	-						3	3
		IR	IM	37	-						3	4
SBCX dst, src	$dst \gets dst - src - C$	ER	ER	38	*	*	*	*	1	*	4	3
		ER	IM	39	-						4	3
SCF	C ← 1			DF	1	-	-	-	-	_	1	2
SRA dst		R		D0	*	*	*	0	_	-	2	2
		IR		D1	-						2	3
SRL dst	0 - ▶ D7 D6 D5 D4 D3 D2 D1 D0 ▶ C	R		1F C0	*	*	0	*	-	_	3	2
	dst	IR		1F C1	_						3	3
SRP src	$RP \leftarrow src$		IM	01	_	-	-	_	-	_	2	2
STOP	STOP Mode			6F	_	_	_	_	_	_	1	2
SUB dst, src	$dst \gets dst - src$	r	r	22	*	*	*	*	1	*	2	3
		r	lr	23	-						2	4
		R	R	24	-						3	3
		R	IR	25	-						3	4
		R	IM	26	-						3	3
		IR	IM	27	-						3	4
Flags Notation:	* = Value is a function of t – = Unaffected X = Undefined	he resu	It of the o	peration.	0 = 1 =	= Re = Se	eset et to	to 1	0			

Table 115. eZ8 CPU Instruction Summary (Continued)

Accombly		Address Mode			Fla	igs			Fatab	Inotr		
Mnemonic	Symbolic Operation	dst	src	(Hex)	С	z	S	V	D	Н	Cycles	Cycles
SUBX dst, src	$dst \gets dst - src$	ER	ER	28	*	*	*	*	1	*	4	3
		ER	IM	29	_						4	3
SWAP dst	$dst[7:4] \leftrightarrow dst[3:0]$	R		F0	Х	*	*	Х	_	_	2	2
		IR		F1	-						2	3
TCM dst, src	(NOT dst) AND src	r	r	62	_	*	*	0	_	-	2	3
		r	lr	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	lr	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	$SP \leftarrow SP - 2$ @SP \leftarrow PC SP \leftarrow SP - 1 @SP \leftarrow FLAGS PC \leftarrow @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 115. eZ8 CPU Instruction Summary (Continued)

Assembly	Symbolic Operation	Address Mode		Oncode(s)	Fla	ags			Fotch	Instr		
Mnemonic		dst	src	(Hex)	С	z	S	v	D	н	Cycles	Cycles
XOR dst, src	$dst \gets dst \ XOR \ src$	r	r	B2	-	*	*	0	_	_	2	3
		r	lr	B3	_	-					2	4
		R	R	B4	_						3	3
		R	IR	B5	_						3	4
		R	IM	B6	_						3	3
		IR	IM	B7	_						3	4
XORX dst, src	$dst \gets dst \: XOR \: src$	ER	ER	B8	_	*	*	0	_	_	4	3
		ER	IM	B9	_						4	3
Flags Notation:	* = Value is a function o – = Unaffected X = Undefined	f the resu	It of the o	peration.	0 = 1 =	= R(= S(ese et to	t to 0 1	0			

Table 115. eZ8 CPU Instruction Summary (Continued)