



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

Obsolete
eZ8
8-Bit
20MHz
IrDA, UART/USART
Brown-out Detect/Reset, LED, LVD, POR, PWM, Temp Sensor, WDT
6
8KB (8K x 8)
FLASH
-
1K x 8
2.7V ~ 3.6V
A/D 4x10b
Internal
0°C ~ 70°C (TA)
Surface Mount
8-SOIC (0.154", 3.90mm Width)
-
https://www.e-xfl.com/product-detail/zilog/z8f082asb020sc

Email: info@E-XFL.COM

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



Reset, Stop Mode Recovery, and Low Voltage Detection	23
Reset Types	23
Reset Sources	25
Power-On Reset	25
Voltage Brownout Reset	26
External Reset Input	27
External Reset Indicator	28
On-Chip Debugger Initiated Reset	28
Stop Mode Recovery	28
Stop Mode Recovery Using Watchdog Timer Time-Out	29
Stop Mode Recovery Using a GPIO Port Pin Transition	29
Stop Mode Recovery Using the External RESET Pin	30
	30
Reset Register Definitions	30
Low-Power Modes	33
STOP Mode	33
HALT Mode	34
Peripheral-Level Power Control	34
Power Control Register Definitions	34
General-Purpose Input/Output	37
GPIO Port Availability By Device	37
Architecture	38
GPIO Alternate Functions	38
Direct LED Drive	39
Shared Reset Pin	39
Shared Debug Pin	39
Crystal Oscillator Override	40
5 V Tolerance	40
External Clock Setup	40
GPIO Interrupts	45
GPIO Control Register Definitions	45
Port A–D Address Registers	46
Port A–D Control Registers	46
Port A D Alternate Function Sub-Registers	47
Poil A-D Alternate Function Sub-Registers	41



# **Universal Asynchronous Receiver/Transmitter**

The full-duplex universal asynchronous receiver/transmitter (UART) is included in all Z8 Encore! XP package types. The UART supports 8- and 9-bit data modes and selectable parity. The UART also supports multi-drop address processing in hardware. The UART baud rate generator (BRG) can be configured and used as a basic 16-bit timer.

### Timers

Two enhanced 16-bit reloadable timers can be used for timing/counting events or for motor control operations. These timers provide a 16-bit programmable reload counter and operate in ONE-SHOT, CONTINUOUS, GATED, CAPTURE, CAPTURE RESTART, COMPARE, CAPTURE and COMPARE, PWM SINGLE OUTPUT and PWM DUAL OUTPUT modes.

### General-Purpose Input/Output

The Z8 Encore! XP F082A Series features 6 to 25 port pins (Ports A–D) for general- purpose input/output (GPIO). The number of GPIO pins available is a function of package, and each pin is individually programmable. 5 V tolerant input pins are available on all I/Os on 8-pin devices and most I/Os on other package types.

# **Direct LED Drive**

The 20- and 28-pin devices support controlled current sinking output pins capable of driving LEDs without the need for a current limiting resistor. These LED drivers are independently programmable to four different intensity levels.

### **Flash Controller**

The Flash Controller programs and erases Flash memory. The Flash Controller supports several protection mechanisms against accidental program and erasure, as well as factory serialization and read protection.

### Non-Volatile Data Storage

The non-volatile data storage (NVDS) uses a hybrid hardware/software scheme to implement a byte programmable data memory and is capable of over 100,000 write cycles.

**Note:** Devices with 8 KB Flash memory do not include the NVDS feature.

Zilog <sub>17</sub>

Program Memory Address (Hex)	Function
Z8F022A and Z8F021A Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–0039	Reserved
003A-003D	Oscillator Fail Trap Vectors
003E-07FF	Program Memory
Z8F012A and Z8F011A Products	
0000–0001	Flash Option Bits
0002–0003	Reset Vector
0004–0005	WDT Interrupt Vector
0006–0007	Illegal Instruction Trap
0008–0037	Interrupt Vectors*
0038–0039	Reserved
003A-003D	Oscillator Fail Trap Vectors
003E-03FF	Program Memory
* See Table 32 on page 56 for a list of the in	nterrupt vectors.

Table 5. Z8 Encore! XP F082A Series Program Memory Maps (Continued)

# **Data Memory**

The Z8 Encore! XP F082A Series does not use the eZ8 CPU's 64 KB Data Memory address space.

# **Flash Information Area**

Table 6 on page 18 describes the Z8 Encore! XP F082A 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 Infor-



function). (Push-pull output)

1 = The source current for the associated pin is disabled (open-drain mode).

## Port A–D High Drive Enable Sub-Registers

The Port A–D High Drive Enable sub-register (Table 22) is accessed through the Port A–D Control register by writing 04H to the Port A–D Address register. Setting the bits in the Port A–D High Drive Enable sub-registers to 1 configures the specified port pins for high current output drive operation. The Port A–D High Drive Enable sub-register affects the pins directly and, as a result, alternate functions are also affected.

Table 22. Port A–D High Drive Enable Sub-Registers (PxHDE)

BITS	7	6	5	4	3	2	1	0
FIELD	PHDE7	PHDE6	PHDE5	PHDE4	PHDE3	PHDE2	PHDE1	PHDE0
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	lf 04H i	n Port A–D /	Address Reg	gister, acces	sible throug	h the Port A-	-D Control F	Register

PHDE[7:0]—Port High Drive Enabled

0 = The Port pin is configured for standard output current drive.

1 = The Port pin is configured for high output current drive.

### Port A–D Stop Mode Recovery Source Enable Sub-Registers

The Port A–D Stop Mode Recovery Source Enable sub-register (Table 23) 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 sub-registers 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 23. Port A–D Stop Mode Recove	ry Source Enable Sub-Registers (PxSMRE)
-------------------------------------	---

BITS	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							
ADDR	lf 05H i	If 05H in Port A–D Address Register, accessible through the Port A–D Control Register							

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



PIN[7:0]—Port Input Data
Sampled data from the corresponding port pin input.
0 = Input data is logical 0 (Low).
1 = Input data is logical 1 (High).

# Port A–D Output Data Register

The Port A–D Output Data register (Table 28) controls the output data to the pins.

BITS	7	6	5	4	3	2	1	0
FIELD	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
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			FI	D3H, FD7H,	FDBH, FDF	Ή		

# Table 28. Port A–D Output Data Register (PxOUT)

POUT[7:0]—Port Output Data

These bits contain the data to be driven to the port pins. The values are only driven if the corresponding pin is configured as an output and the pin is not configured for alternate function operation.

0 = Drive a logical 0 (Low).

1= Drive a logical 1 (High). High value is not driven if the drain has been disabled by setting the corresponding Port Output Control register bit to 1.

# LED Drive Enable Register

The LED Drive Enable register (Table 29) activates the controlled current drive. The Port C pin must first be enabled by setting the Alternate Function register to select the LED function.

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				F8	2H				

## Table 29. LED Drive Enable (LEDEN)



# **Timer Control Register Definitions**

# 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 (Table 48). 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 48. 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
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 details of the full timer mode decoding.

TICONFIG—Timer Interrupt Configuration

This field configures timer interrupt definition.

- 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



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

# Table 58. Watchdog Timer Reload Upper Byte Register (WDTU)

BITS	7	6	5	4	3	2	1	0	
FIELD	WDTU								
RESET		00H							
R/W				R/	W*				
ADDR	FF1H								
R/W/* - Rea	PAW* - Pad returns the current WDT count value. Write sets the appropriate Peload Value								

R/W\* - Read returns the current WDT count value. Write sets the appropriate Reload Value.

WDTU—WDT Reload Upper Byte

Most-significant byte (MSB), Bits[23:16], of the 24-bit WDT reload value.

### Table 59. Watchdog Timer Reload High Byte Register (WDTH)

BITS	7	6	5	4	3	2	1	0		
FIELD	WDTH									
RESET	04H									
R/W				R/	W*					
ADDR	FF2H									
R/W* - Rea	R/W* - Read returns the current WDT count value. Write sets the appropriate Reload Value.									

WDTH—WDT Reload High Byte

Middle byte, Bits[15:8], of the 24-bit WDT reload value.

### Table 60. Watchdog Timer Reload Low Byte Register (WDTL)

BITS	7	6	5	4	3	2	1	0		
FIELD	WDTL									
RESET		00H								
R/W				R/	W*					
ADDR	FF3H									
R/W* - Rea	R/W* - Read returns the current WDT count value. Write sets the appropriate Reload Value.									

WDTL—WDT Reload Low

Least significant byte (LSB), Bits[7:0], of the 24-bit WDT reload value.



### **Receiving Data using the Interrupt-Driven Method**

The UART Receiver interrupt indicates the availability of new data (as well as error conditions). Follow the steps below to configure the UART receiver for interrupt-driven operation:

- 1. Write to the UART Baud Rate High and Low Byte registers to set the acceptable baud rate.
- 2. Enable the UART pin functions by configuring the associated GPIO Port pins for alternate function operation.
- 3. Execute a DI instruction to disable interrupts.
- 4. Write to the Interrupt control registers to enable the UART Receiver interrupt and set the acceptable priority.
- 5. Clear the UART Receiver interrupt in the applicable Interrupt Request register.
- 6. Write to the UART Control 1 Register to enable Multiprocessor (9-bit) mode functions, if appropriate.
  - Set the Multiprocessor Mode Select (MPEN) to Enable MULTIPROCESSOR mode.
  - Set the Multiprocessor Mode Bits, MPMD[1:0], to select the acceptable address matching scheme.
  - Configure the UART to interrupt on received data and errors or errors only (interrupt on errors only is unlikely to be useful for Z8 Encore!<sup>®</sup> devices without a DMA block)
- 7. Write the device address to the Address Compare Register (automatic MULTIPRO-CESSOR modes only).
- 8. Write to the UART Control 0 register to:
  - Set the receive enable bit (REN) to enable the UART for data reception
  - Enable parity, if appropriate and if multiprocessor mode is not enabled, and select either even or odd parity.
- 9. Execute an EI instruction to enable interrupts.

The UART is now configured for interrupt-driven data reception. When the UART Receiver interrupt is detected, the associated interrupt service routine (ISR) performs the following:

- 1. Checks the UART Status 0 register to determine the source of the interrupt error, break, or received data.
- 2. Reads the data from the UART Receive Data register if the interrupt was because of data available. If operating in MULTIPROCESSOR (9-bit) mode, further actions may be required depending on the MULTIPROCESSOR mode bits MPMD[1:0].



138

# zilog

# **Temperature Sensor**

The on-chip Temperature Sensor allows you to measure temperature on the die with either the on-board ADC or on-board comparator. This block is factory calibrated for in-circuit software correction. Uncalibrated accuracy is significantly worse, therefore the temperature sensor is not recommended for uncalibrated use.

# **Temperature Sensor Operation**

The on-chip temperature sensor is a Proportional to Absolute Temperature (PTAT) topology. A pair of Flash option bytes contain the calibration data. The temperature sensor can be disabled by a bit in the Power Control Register 0 on page 34 to reduce power consumption.

The temperature sensor can be directly read by the ADC to determine the absolute value of its output. The temperature sensor output is also available as an input to the comparator for threshold type measurement determination. The accuracy of the sensor when used with the comparator is substantially less than when measured by the ADC.

If the temperature sensor is routed to the ADC, the ADC must be configured in unity-gain buffered mode (see Input Buffer Stage on page 129) The value read back from the ADC is a signed number, although it is always positive.

The sensor is factory-trimmed through the ADC using the external 2.0 V reference. Unless the sensor is re-trimmed for use with a different reference, it is most accurate when used with the external 2.0 V reference.

Because this sensor is an on-chip sensor it is recommended that the user account for the difference between ambient and die temperature when inferring ambient temperature conditions.

During normal operation, the die undergoes heating that causes a mismatch between the ambient temperature and that measured by the sensor. For best results, the Z8 Encore! XP<sup>®</sup> device must be placed into STOP mode for sufficient time such that the die and ambient temperatures converge (this time is dependent on the thermal design of the system). The temperature sensor measurement must then be made immediately after recovery from STOP mode.

The following equation defines the transfer function between the temperature sensor output voltage and the die temperature. This is needed for comparator threshold measurements.

 $V = 0.01 \times T + 0.65$ 

where, T is the temperature in °C; V is the sensor output in volts.

# zilog <sub>14</sub>

# Flash Operation Timing Using the Flash Frequency Registers

Before performing either a program or erase operation on Flash memory, you must first configure the Flash Frequency High and Low Byte registers. The Flash Frequency registers allow programming and erasing of the Flash with system clock frequencies ranging from 32 kHz (32768 Hz) through 20 MHz.

The Flash Frequency High and Low Byte registers combine to form a 16-bit value, FFREQ, to control timing for Flash program and erase operations. The 16-bit binary Flash Frequency value must contain the system clock frequency (in kHz). This value is calculated using the following equation:

 $FFREQ[15:0] = \frac{System Clock Frequency (Hz)}{1000}$ 



**Caution:** Flash programming and erasure are not supported for system clock frequencies below 32 kHz (32768 Hz) or above 20 MHz. The Flash Frequency High and Low Byte registers must be loaded with the correct value to ensure operation of the Z8 Encore! XP<sup>®</sup> F082A Series devices.

# Flash Code Protection Against External Access

The user code contained within the Flash memory can be protected against external access by the on-chip debugger. Programming the FRP Flash Option Bit prevents reading of the user code with the On-Chip Debugger. See Flash Option Bits on page 153 and On-Chip Debugger on page 173 for more information.

# Flash Code Protection Against Accidental Program and Erasure

The Z8 Encore! XP F082A Series provides several levels of protection against accidental program and erasure of the Flash memory contents. This protection is provided by a combination of the Flash Option bits, the register locking mechanism, the page select redundancy and the sector level protection control of the Flash Controller.

### Flash Code Protection Using the Flash Option Bits

The FRP and FWP Flash Option Bits combine to provide three levels of Flash Program Memory protection as listed in Table 77. See Flash Option Bits on page 153 for more information.

**z**ilog<sup>®</sup>

# Flash Status Register

The Flash Status (FSTAT) register indicates the current state of the Flash Controller. This register can be read at any time. The read-only Flash Status register shares its Register File address with the Write-only Flash Control register.

Table 79. Flash Status Register (FSTAT)

BITS	7	6	5	4	3	2	1	0			
FIELD	Rese	erved	FSTAT								
RESET	0	0	0	0	0	0	0	0			
R/W	R	R	R	R R R R F							
ADDR				FF	8H						

Reserved—Must be 0.

FSTAT—Flash Controller Status

000000 = Flash Controller locked

000001 = First unlock command received (73H written)

000010 = Second unlock command received (8CH written)

000011 = Flash Controller unlocked

000100 = Sector protect register selected

001xxx = Program operation in progress

010xxx = Page erase operation in progress

100xxx = Mass erase operation in progress

# Flash Page Select Register

The Flash Page Select (FPS) register shares address space with the Flash Sector Protect Register. Unless the Flash controller is unlocked and written with 5EH, writes to this address target the Flash Page Select Register.

The register is used to select one of the available Flash memory pages to be programmed or erased. Each Flash Page contains 512 bytes of Flash memory. During a Page Erase operation, all Flash memory having addresses with the most significant 7 bits given by FPS[6:0] are chosen for program/erase operation.



Info Page Address	Memory Address	Usage
5C	FE5C	Randomized Lot ID Byte 23
5D	FE5D	Randomized Lot ID Byte 22
5E	FE5E	Randomized Lot ID Byte 21
5F	FE5F	Randomized Lot ID Byte 20
61	FE61	Randomized Lot ID Byte 19
62	FE62	Randomized Lot ID Byte 18
64	FE64	Randomized Lot ID Byte 17
65	FE65	Randomized Lot ID Byte 16
67	FE67	Randomized Lot ID Byte 15
68	FE68	Randomized Lot ID Byte 14
6A	FE6A	Randomized Lot ID Byte 13
6B	FE6B	Randomized Lot ID Byte 12
6D	FE6D	Randomized Lot ID Byte 11
6E	FE6E	Randomized Lot ID Byte 10
70	FE70	Randomized Lot ID Byte 9
71	FE71	Randomized Lot ID Byte 8
73	FE73	Randomized Lot ID Byte 7
74	FE74	Randomized Lot ID Byte 6
76	FE76	Randomized Lot ID Byte 5
77	FE77	Randomized Lot ID Byte 4
79	FE79	Randomized Lot ID Byte 3
7A	FE7A	Randomized Lot ID Byte 2
7C	FE7C	Randomized Lot ID Byte 1
7D	FE7D	Randomized Lot ID Byte 0 (least significant)

# Table 102. Randomized Lot ID Locations (Continued)



If the device is not in DEBUG mode or the Flash Read Protect Option bit is enabled, this command reads and discards one byte.

DBG  $\leftarrow$  12H DBG  $\leftarrow$  1-5 byte opcode

# **On-Chip Debugger Control Register Definitions**

# **OCD Control Register**

The OCD Control register controls the state of the On-Chip Debugger. This register is used to enter or exit DEBUG mode and to enable the BRK instruction. It can also reset the Z8 Encore!  $XP^{\text{(B)}}$  F082A Series device.

A reset and stop function can be achieved by writing 81H to this register. A reset and go function can be achieved by writing 41H to this register. If the device is in DEBUG mode, a run function can be implemented by writing 40H to this register.

# Table 106. OCD Control Register (OCDCTL)

BITS	7	6	5	4	3	2	2 1					
FIELD	DBGMODE	BRKEN	DBGACK		Reserved							
RESET	0	0	0	0	0	0	0	0				
R/W	R/W	R/W	R/W	R	R	R	R	R/W				

### DBGMODE—DEBUG Mode

The device enters DEBUG mode when this bit is 1. When in DEBUG mode, the eZ8 CPU stops fetching new instructions. Clearing this bit causes the eZ8 CPU to restart. This bit is automatically set when a BRK instruction is decoded and Breakpoints are enabled. If the Flash Read Protect Option Bit is enabled, this bit can only be cleared by resetting the device. It cannot be written to 0.

0 = The Z8 Encore! XP F082A Series device is operating in NORMAL mode.

1 = The Z8 Encore! XP F082A Series device is in DEBUG mode.

### BRKEN—Breakpoint Enable

This bit controls the behavior of the BRK instruction (opcode 00H). By default, Breakpoints are disabled and the BRK instruction behaves similar to an NOP instruction. If this bit is 1, when a BRK instruction is decoded, the DBGMODE bit of the OCDCTL register is automatically set to 1.

0 = Breakpoints are disabled.

1 = Breakpoints are enabled.

zilog

189

When selecting a new clock source, the system clock oscillator failure detection circuitry and the Watchdog Timer oscillator failure circuitry must be disabled. If SOFEN and WOFEN are not disabled prior to a clock switch-over, it is possible to generate an interrupt for a failure of either oscillator. The Failure detection circuitry can be enabled anytime after a successful write of OSCSEL in the OSCCTL register.

The internal precision oscillator is enabled by default. If the user code changes to a different oscillator, it may be appropriate to disable the IPO for power savings. Disabling the IPO does not occur automatically.

# **Clock Failure Detection and Recovery**

### System Clock Oscillator Failure

The Z8F04xA family devices can generate non-maskable interrupt-like events when the primary oscillator fails. To maintain system function in this situation, the clock failure recovery circuitry automatically forces the Watchdog Timer oscillator to drive the system clock. The Watchdog Timer oscillator must be enabled to allow the recovery. Although this oscillator runs at a much slower speed than the original system clock, the CPU continues to operate, allowing execution of a clock failure vector and software routines that either remedy the oscillator failure or issue a failure alert. This automatic switch-over is not available if the Watchdog Timer oscillator is disabled, though it is not necessary to enable the Watchdog Timer reset function (see Watchdog Timer on page 91).

The primary oscillator failure detection circuitry asserts if the system clock frequency drops below 1 kHz  $\pm$ 50%. If an external signal is selected as the system oscillator, it is possible that a very slow but non-failing clock can generate a failure condition. Under these conditions, do not enable the clock failure circuitry (SOFEN must be deasserted in the OSCCTL register).

### Watchdog Timer Failure

In the event of a Watchdog Timer oscillator failure, a similar non-maskable interrupt-like event is issued. This event does not trigger an attendant clock switch-over, but alerts the CPU of the failure. After a Watchdog Timer failure, it is no longer possible to detect a primary oscillator failure. The failure detection circuitry does not function if the Watchdog Timer is used as the system clock oscillator or if the Watchdog Timer oscillator has been disabled. For either of these cases, it is necessary to disable the detection circuitry by deasserting the WDFEN bit of the OSCCTL register.

The Watchdog Timer oscillator failure detection circuit counts system clocks while looking for a Watchdog Timer clock. The logic counts 8004 system clock cycles before determining that a failure has occurred. The system clock rate determines the speed at which the Watchdog Timer failure can be detected. A very slow system clock results in very slow detection times.

zilog <sub>201</sub>

# Table 114. Notational Shorthand

Notation	Description	Operand	Range
b	Bit	b	b represents a value from 0 to 7 (000B to 111B).
CC	Condition Code	—	Refer to Condition Codes section in the <i>eZ8 CPU Core User Manual (UM0128)</i> .
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 115 lists additional symbols that are used throughout the Instruction Summary and Instruction Set Description sections.

<mark>z</mark>ilog<sup>°</sup>

21	1

Assembly	Symbolic	Addres	s Mode	Opcode(s)			Fla	ags			Fetch	Instr. Cycles
Mnemonic	Operation	dst	src	(Hex)	С	Ζ	S	۷	D	н	Cycles	
LDC dst, src	$dst \gets src$	r	Irr	C2	-	-	-	-	-	-	2	5
		Ir	Irr	C5	-						2	9
		Irr	r	D2	-						2	5
LDCI dst, src	$dst \gets src$	lr	Irr	C3	_	_	_	_	-	_	2	9
	$r \leftarrow r + 1$ $rr \leftarrow rr + 1$	Irr	lr	D3	_						2	9
LDE dst, src	$dst \gets src$	r	Irr	82	-	-	-	-	-	-	2	5
		Irr	r	92	-						2	5
LDEI dst, src	dst ← src	lr	Irr	83	_	_	_	_	-	_	2	9
	r ← r + 1 rr ← rr + 1	Irr	lr	93	_						2	9
LDWX dst, src	$dst \gets src$	ER	ER	1FE8	_	-	-	-	-	-	5	4
LDX dst, src	$dst \gets src$	r	ER	84	-	-	_	-	-	-	3	2
		lr	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	lr	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)	$dst \gets src + X$	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
Flags Notation:	* = Value is a function of – = Unaffected X = Undefined	the result	of the o	peration.	0 = 1 =	Re Se	set t to	to ( 1	)			

# Table 124. eZ8 CPU Instruction Summary (Continued)

**z**ilog<sup>°</sup>

21	Λ
<b>∠</b> I	4

Assembly	Symbolic	Addre	ss Mode	Opcode(s)	Flags						Fetch	Instr.
Mnemonic	Operation	dst	src	(Hex)	С	Ζ	S	۷	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 - = Unaffected X = Undefined	the resul	It of the o	peration.	0 = 1 =	Re Se	eset et to	to ( 1	)			

# Table 124. eZ8 CPU Instruction Summary (Continued)



Figure 34. Port Input Sample Timing

Table	139.	<b>GPIO</b>	Port	Input	Timing
-------	------	-------------	------	-------	--------

		Dela	y (ns)
Parameter	Abbreviation	Minimum	Maximum
T <sub>S_PORT</sub>	Port Input Transition to XIN Rise Setup Time (Not pictured)	5	-
T <sub>H_PORT</sub>	XIN Rise to Port Input Transition Hold Time (Not pictured)	0	-
T <sub>SMR</sub>	GPIO Port Pin Pulse Width to ensure Stop Mode Recovery (for GPIO Port Pins enabled as SMR sources)	1 μs	



Part Number	Flash	RAM	SDVN	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Comparator	Temperature Sensor	Description
Z8 Encore! XP <sup>®</sup> F082	A Serie	s with 4	KB Fla	sh							
Standard Temperatur	re: 0 °C 1	to 70 °C	)								
Z8F041APB020SC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	PDIP 8-pin package
Z8F041AQB020SC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	QFN 8-pin package
Z8F041ASB020SC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	SOIC 8-pin package
Z8F041ASH020SC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SOIC 20-pin package
Z8F041AHH020SC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SSOP 20-pin package
Z8F041APH020SC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	PDIP 20-pin package
Z8F041ASJ020SC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SOIC 28-pin package
Z8F041AHJ020SC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SSOP 28-pin package
Z8F041APJ020SC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	PDIP 28-pin package
Extended Temperatur	re: -40 °	C to 10	5 °C								
Z8F041APB020EC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	PDIP 8-pin package
Z8F041AQB020EC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	QFN 8-pin package
Z8F041ASB020EC	4 KB	1 KB	128 B	6	13	2	0	1	1	0	SOIC 8-pin package
Z8F041ASH020EC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SOIC 20-pin package
Z8F041AHH020EC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	SSOP 20-pin package
Z8F041APH020EC	4 KB	1 KB	128 B	17	19	2	0	1	1	0	PDIP 20-pin package
Z8F041ASJ020EC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SOIC 28-pin package
Z8F041AHJ020EC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	SSOP 28-pin package
Z8F041APJ020EC	4 KB	1 KB	128 B	25	19	2	0	1	1	0	PDIP 28-pin package
Replace C with G for Lea	d-Free P	ackagin	9								