



Welcome to [E-XFL.COM](https://www.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	16
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 7x10b
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/z8f0423hh005ec

Operation	152
OCD Interface	152
DEBUG Mode	153
OCD Data Format	154
OCD Auto-Baud Detector/Generator	154
OCD Serial Errors	155
OCD Unlock Sequence (8-Pin Devices Only)	156
Breakpoints	156
Runtime Counter	156
On-Chip Debugger Commands	157
On-Chip Debugger Control Register Definitions	161
OCD Control Register	161
OCD Status Register	163
Oscillator Control	165
Operation	165
System Clock Selection	165
Clock Failure Detection and Recovery	166
Oscillator Control Register Definitions	167
Internal Precision Oscillator	169
Operation	169
eZ8 CPU Instruction Set	171
Assembly Language Programming Introduction	171
Assembly Language Syntax	172
eZ8 CPU Instruction Notation	172
eZ8 CPU Instruction Classes	174
eZ8 CPU Instruction Summary	179
Opcode Maps	188
Electrical Characteristics	193
Absolute Maximum Ratings	193
DC Characteristics	194
AC Characteristics	197
On-Chip Peripheral AC and DC Electrical Characteristics	199
General Purpose I/O Port Input Data Sample Timing	202
General Purpose I/O Port Output Timing	204
On-Chip Debugger Timing	205
UART Timing	206
Packaging	209

Table 5 provides detailed information about the characteristics for each pin available on Z8 Encore! XP[®] F0823 Series 8-pin devices.

► **Note:** *All six I/O pins on the 8-pin packages are 5 V-tolerant (unless the pull-up devices are enabled). The column in Table 4 below describes 5 V-tolerance for the 20- and 28-pin packages only.*

Table 4. Pin Characteristics (20- and 28-pin Devices)

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tristate Output	Internal Pull-up or Pull-down	Schmitt-Trigger Input	Open Drain Output	5 V Tolerance
AVDD	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
AVSS	N/A	N/A	N/A	N/A	N/A	N/A	N/A	NA
DBG	I/O	I	N/A	Yes	No	Yes	Yes	Yes
PA[7:0]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	PA[7:2] only
PB[7:0]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	PB[7:6] only
PC[7:0]	I/O	I	N/A	Yes	Programmable Pull-up	Yes	Yes, Programmable	PC[7:3] only
RESET	I/O	I/O (defaults to RESET)	Low (in Reset mode)	Yes (PD0 only)	Always on for RESET	Yes	Always on for RESET	Yes
VDD	N/A	N/A	N/A	N/A			N/A	N/A
VSS	N/A	N/A	N/A	N/A			N/A	N/A

► **Note:** *PB6 and PB7 are available only in the devices without ADC.*

Register Map

Table 8 lists the address map for the Register File of the Z8 Encore! XP[®] F0823 Series devices. Not all devices and package styles in the Z8 Encore! XP F0823 Series support the ADC, or all GPIO ports. Consider registers for unimplemented peripherals as reserved.

Table 8. Register File Address Map

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page No
General-Purpose RAM				
Z8F0823/Z8F0813 Devices				
000–3FF	General-Purpose Register File RAM	—	XX	
400–EFF	Reserved	—	XX	
Z8F0423/Z8F0413 Devices				
000–3FF	General-Purpose Register File RAM	—	XX	
400–EFF	Reserved	—	XX	
Z8F0223/Z8F0213 Devices				
000–1FF	General-Purpose Register File RAM	—	XX	
200–EFF	Reserved	—	XX	
Z8F0123/Z8F0113 Devices				
000–0FF	General-Purpose Register File RAM	—	XX	
100–EFF	Reserved	—	XX	
Timer 0				
F00	Timer 0 High Byte	T0H	00	80
F01	Timer 0 Low Byte	T0L	01	80
F02	Timer 0 Reload High Byte	T0RH	FF	81
F03	Timer 0 Reload Low Byte	T0RL	FF	81
F04	Timer 0 PWM High Byte	T0PWMH	00	81
F05	Timer 0 PWM Low Byte	T0PWML	00	82
F06	Timer 0 Control 0	T0CTL0	00	82
F07	Timer 0 Control 1	T0CTL1	00	83
Timer 1				
F08	Timer 1 High Byte	T1H	00	80
F09	Timer 1 Low Byte	T1L	01	80
F0A	Timer 1 Reload High Byte	T1RH	FF	81
F0B	Timer 1 Reload Low Byte	T1RL	FF	81

General-Purpose Input/Output

Z8 Encore! XP[®] F0823 Series products support a maximum of 24 port pins (Ports A–C) for general-purpose input/output (GPIO) operations. Each port contains control and data registers. The GPIO control registers determine data direction, open-drain, output drive current, programmable pull-ups, Stop Mode Recovery functionality, and alternate pin functions. Each port pin is individually programmable. In addition, the Port C pins are capable of direct LED drive at programmable drive strengths.

GPIO Port Availability By Device

Table 14 lists the port pins available with each device and package type.

Table 14. Port Availability by Device and Package Type

Devices	Package	10-Bit ADC	Port A	Port B	Port C	Total I/O
Z8F0823SB, Z8F0823PB Z8F0423SB, Z8F0423PB Z8F0223SB, Z8F0223PB Z8F0123SB, Z8F0123PB	8-pin	Yes	[5:0]	No	No	6
Z8F0813SB, Z8F0813PB Z8F0413SB, Z8F0413PB Z8F0213SB, Z8F0213PB Z8F0113SB, Z8F011vPB	8-pin	No	[5:0]	No	No	6
Z8F0823PH, Z8F0823HH Z8F0423PH, Z8F0423HH Z8F0223PH, Z8F0223HH Z8F0123PH, Z8F0123HH	20-pin	Yes	[7:0]	[3:0]	[3:0]	16
Z8F0813PH, Z8F0813HH Z8F0413PH, Z8F0413HH Z8F0213PH, Z8F0213HH Z8F0113PH, Z8F0113HH	20-pin	No	[7:0]	[3:0]	[3:0]	16
Z8F0823PJ, Z8F0823SJ Z8F0423PJ, Z8F0423SJ Z8F0223PJ, Z8F0223SJ Z8F0123PJ, Z8F0123SJ	28-pin	Yes	[7:0]	[5:0]	[7:0]	22
Z8F0813PJ, Z8F0813SJ Z8F0413PJ, Z8F0413SJ Z8F0213PJ, Z8F0213SJ Z8F0113PJ, Z8F0113SJ	28-pin	No	[7:0]	[7:0]	[7:0]	24

Table 15. Port Alternate Function Mapping (Non 8-Pin Parts)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port A	PA0	T0IN/T0OUT*	Timer 0 Input/Timer 0 Output Complement	N/A
		Reserved		
	PA1	T0OUT	Timer 0 Output	
		Reserved		
	PA2	DE0	UART 0 Driver Enable	
		Reserved		
	PA3	CTS0	UART 0 Clear to Send	
		Reserved		
	PA4	RXD0/IRRX0	UART 0 / IrDA 0 Receive Data	
		Reserved		
	PA5	TXD0/IRTX0	UART 0 / IrDA 0 Transmit Data	
		Reserved		
	PA6	T1IN/T1OUT*	Timer 1 Input/Timer 1 Output Complement	
		Reserved		
	PA7	T1OUT	Timer 1 Output	
		Reserved		

Note: Because there is only a single alternate function for each Port A pin, the Alternate Function Set registers are not implemented for Port A. Enabling alternate function selections as described in Port A–C Alternate Function Sub-Registers automatically enables the associated alternate function.

* Whether PA0/PA6 take on the timer input or timer output complement function depends on the timer configuration as described in Timer Pin Signal Operation on page 79.

Table 16. Port Alternate Function Mapping (8-Pin Parts)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Select Register AFS1	Alternate Function Select Register AFS2
Port A	PA0	T0IN	Timer 0 Input	AFS1[0]: 0	AFS2[0]: 0
		Reserved		AFS1[0]: 0	AFS2[0]: 1
		Reserved		AFS1[0]: 1	AFS2[0]: 0
		$\overline{T0OUT}$	Timer 0 Output Complement	AFS1[0]: 1	AFS2[0]: 1
	PA1	T0OUT	Timer 0 Output	AFS1[1]: 0	AFS2[1]: 0
		Reserved		AFS1[1]: 0	AFS2[1]: 1
		CLKIN	External Clock Input	AFS1[1]: 1	AFS2[1]: 0
		Analog Functions*	ADC Analog Input/VREF	AFS1[1]: 1	AFS2[1]: 1
	PA2	DE0	UART 0 Driver Enable	AFS1[2]: 0	AFS2[2]: 0
		\overline{RESET}	External Reset	AFS1[2]: 0	AFS2[2]: 1
		T1OUT	Timer 1 Output	AFS1[2]: 1	AFS2[2]: 0
		Reserved		AFS1[2]: 1	AFS2[2]: 1
	PA3	$\overline{CTS0}$	UART 0 Clear to Send	AFS1[3]: 0	AFS2[3]: 0
		COUT	Comparator Output	AFS1[3]: 0	AFS2[3]: 1
		T1IN	Timer 1 Input	AFS1[3]: 1	AFS2[3]: 0
		Analog Functions*	ADC Analog Input	AFS1[3]: 1	AFS2[3]: 1
	PA4	RXD0	UART 0 Receive Data	AFS1[4]: 0	AFS2[4]: 0
		Reserved		AFS1[4]: 0	AFS2[4]: 1
		Reserved		AFS1[4]: 1	AFS2[4]: 0
		Analog Functions*	ADC/Comparator Input (N)	AFS1[4]: 1	AFS2[4]: 1
	PA5	TXD0	UART 0 Transmit Data	AFS1[5]: 0	AFS2[5]: 0
		$\overline{T1OUT}$	Timer 1 Output Complement	AFS1[5]: 0	AFS2[5]: 1
		Reserved		AFS1[5]: 1	AFS2[5]: 0
		Analog Functions*	ADC/Comparator Input (P)	AFS1[5]: 1	AFS2[5]: 1

Note: * Analog Functions include ADC inputs, ADC reference and comparator inputs. Also, alternate function selection as described in Port A–C Alternate Function Sub-Registers must be enabled.

Table 28. Port A–C Input Data Registers (PxIN)

BITS	7	6	5	4	3	2	1	0
FIELD	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X	X	X	X	X	X	X	X
R/W	R	R	R	R	R	R	R	R
ADDR	FD2H, FD6H, FDAH							

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–C Output Data Register

The Port A–C Output Data register (Table 29) controls the output data to the pins.

Table 29. Port A–C Output Data Register (PxOUT)

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	FD3H, FD7H, FDBH							

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

Interrupt Request 1 Register

The Interrupt Request 1 (IRQ1) register (Table 35) stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ1 register becomes 1. If interrupts are globally enabled (vectored interrupts), the interrupt controller passes an interrupt request to the eZ8 CPU. If interrupts are globally disabled (polled interrupts), the eZ8 CPU reads the Interrupt Request 1 register to determine if any interrupt requests are pending.

Table 35. Interrupt Request 1 Register (IRQ1)

BITS	7	6	5	4	3	2	1	0
FIELD	PA7VI	PA6CI	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
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	FC3H							

PA7VI—Port A7 Interrupt Request

0 = No interrupt request is pending for GPIO Port A

1 = An interrupt request from GPIO Port A

PA6CI—Port A6 or Comparator Interrupt Request

0 = No interrupt request is pending for GPIO Port A or Comparator

1 = An interrupt request from GPIO Port A or Comparator

PAxI—Port A Pin *x* Interrupt Request

0 = No interrupt request is pending for GPIO Port A pin *x*

1 = An interrupt request from GPIO Port A pin *x* is awaiting service

where *x* indicates the specific GPIO Port pin number (0–5)

Interrupt Request 2 Register

The Interrupt Request 2 (IRQ2) register (Table 36) stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ2 register becomes 1. If interrupts are globally enabled (vectored interrupts), the interrupt controller passes an interrupt request to the eZ8 CPU. If interrupts are globally disabled (polled interrupts), the eZ8 CPU can read the Interrupt Request 2 register to determine if any interrupt requests are pending.

COUNTER Mode

In COUNTER mode, the timer counts input transitions from a GPIO port pin. The timer input is taken from the GPIO port pin Timer Input alternate function. The TPOL bit in the Timer Control register selects whether the count occurs on the rising edge or the falling edge of the timer input signal. In COUNTER mode, the prescaler is disabled.

! Caution: *The input frequency of the timer input signal must not exceed one-fourth the system clock frequency.*

Upon reaching the Reload value stored in the Timer Reload High and Low Byte registers, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. 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 Reload.

Follow the steps below for configuring a timer for COUNTER mode and initiating the count:

1. Write to the Timer Control register to:
 - Disable the timer.
 - Configure the timer for COUNTER mode.
 - Select either the rising edge or falling edge of the Timer Input signal for the count. This selection also sets the initial logic level (High or Low) for the Timer Output alternate function. However, the Timer Output function is not required to be enabled.
2. Write to the Timer High and Low Byte registers to set the starting count value. This only affects the first pass in COUNTER mode. After the first timer Reload in COUNTER mode, counting always begins at the reset value of 0001H. In COUNTER mode the Timer High and Low Byte registers must be written with the value 0001H.
3. Write to the Timer Reload High and Low Byte registers to set the Reload value.
4. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
5. Configure the associated GPIO port pin for the Timer Input alternate function.
6. If using the Timer Output function, configure the associated GPIO port pin for the Timer Output alternate function.
7. Write to the Timer Control register to enable the timer.

In COUNTER mode, the number of Timer Input transitions since the timer start is given by the following equation:

$$\text{COUNTER Mode Timer Input Transitions} = \text{Current Count Value} - \text{Start Value}$$

Watchdog Timer Refresh

When first enabled, the WDT is loaded with the value in the Watchdog Timer Reload registers. The Watchdog Timer counts down to 000000H unless a WDT instruction is executed by the eZ8 CPU. Execution of the WDT instruction causes the down counter to be reloaded with the WDT Reload value stored in the Watchdog Timer Reload registers. Counting resumes following the reload operation.

When Z8 Encore! XP[®] F0823 Series devices are operating in DEBUG Mode (using the OCD), the Watchdog Timer is continuously refreshed to prevent any Watchdog Timer time-outs.

Watchdog Timer Time-Out Response

The Watchdog Timer times out when the counter reaches 000000H. A time-out of the Watchdog Timer generates either an interrupt or a system reset. The WDT_RES Flash Option Bit determines the time-out response of the Watchdog Timer. For information on programming of the WDT_RES Flash Option Bit, see Flash Option Bits on page 141.

WDT Interrupt in Normal Operation

If configured to generate an interrupt when a time-out occurs, the Watchdog Timer issues an interrupt request to the interrupt controller and sets the WDT status bit in the Watchdog Timer Control register. If interrupts are enabled, the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executing code from the vector address. After time-out and interrupt generation, the Watchdog Timer counter rolls over to its maximum value of FFFFFFFH and continues counting. The Watchdog Timer counter is not automatically returned to its Reload Value.

The Reset Status Register (see Reset Status Register on page 28) must be read before clearing the WDT interrupt. This read clears the WDT time-out Flag and prevents further WDT interrupts for immediately occurring.

WDT Interrupt in STOP Mode

If configured to generate an interrupt when a time-out occurs and Z8 Encore! XP F0823 Series are in STOP mode, the Watchdog Timer automatically initiates a Stop Mode Recovery and generates an interrupt request. Both the WDT status bit and the STOP bit in the Watchdog Timer Control register are set to 1 following a WDT time-out in STOP mode. For more information on Stop Mode Recovery, see Reset and Stop Mode Recovery on page 21.

If interrupts are enabled, following completion of the Stop Mode Recovery the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executing code from the vector address.

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! XP devices without a DMA block)
7. Write the device address to the Address Compare Register (automatic `MULTIPROCESSOR` 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]`.

Reserved—R/W bits must be 0 during writes; 0 when read.

NEWFRM—Status bit denoting the start of a new frame. Reading the UART Receive Data register resets this bit to 0.

0 = The current byte is not the first data byte of a new frame

1 = The current byte is the first data byte of a new frame

MPRX—Multiprocessor Receive

Returns the value of the most recent multiprocessor bit received. Reading from the UART Receive Data register resets this bit to 0.

UART Control 0 and Control 1 Registers

The UART Control 0 and Control 1 registers (Table 66 and Table 67) configure the properties of the UART's transmit and receive operations. The UART Control registers must not be written while the UART is enabled.

Table 66. UART Control 0 Register (U0CTL0)

BITS	7	6	5	4	3	2	1	0
FIELD	TEN	REN	CTSE	PEN	PSEL	SBRK	STOP	LBEN
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	F42H							

TEN—Transmit Enable

This bit enables or disables the transmitter. The enable is also controlled by the $\overline{\text{CTS}}$ signal and the CTSE bit. If the $\overline{\text{CTS}}$ signal is low and the CTSE bit is 1, the transmitter is enabled.

0 = Transmitter disabled

1 = Transmitter enabled

REN—Receive Enable

This bit enables or disables the receiver.

0 = Receiver disabled

1 = Receiver enabled

CTSE—CTS Enable

0 = The $\overline{\text{CTS}}$ signal has no effect on the transmitter

1 = The UART recognizes the $\overline{\text{CTS}}$ signal as an enable control from the transmitter

PEN—Parity Enable

This bit enables or disables parity. Even or odd is determined by the PSEL bit.

0 = Parity is disabled

1 = The transmitter sends data with an additional parity bit and the receiver receives an additional parity bit

baud rate clocks to plus eight baud rate clocks around the expected time of an incoming pulse. If an incoming pulse is detected inside this window this process is repeated. If the incoming data is a logical 1 (no pulse), the Endec returns to the initial state and waits for the next falling edge. As each falling edge is detected, the Endec clock counter is reset, resynchronizing the Endec to the incoming signal, allowing the Endec to tolerate jitter and baud rate errors in the incoming datastream. Resynchronizing the Endec does not alter the operation of the UART, which ultimately receives the data. The UART is only synchronized to the incoming data stream when a Start bit is received.

Infrared Encoder/Decoder Control Register Definitions

All Infrared Endec configuration and status information is set by the UART control registers as defined in Universal Asynchronous Receiver/Transmitter on page 93.

! **Caution:** *To prevent spurious signals during IrDA data transmission, set the IREN bit in the UART Control 1 register to 1 to enable the Infrared Encoder/Decoder before enabling the GPIO port alternate function for the corresponding pin.*

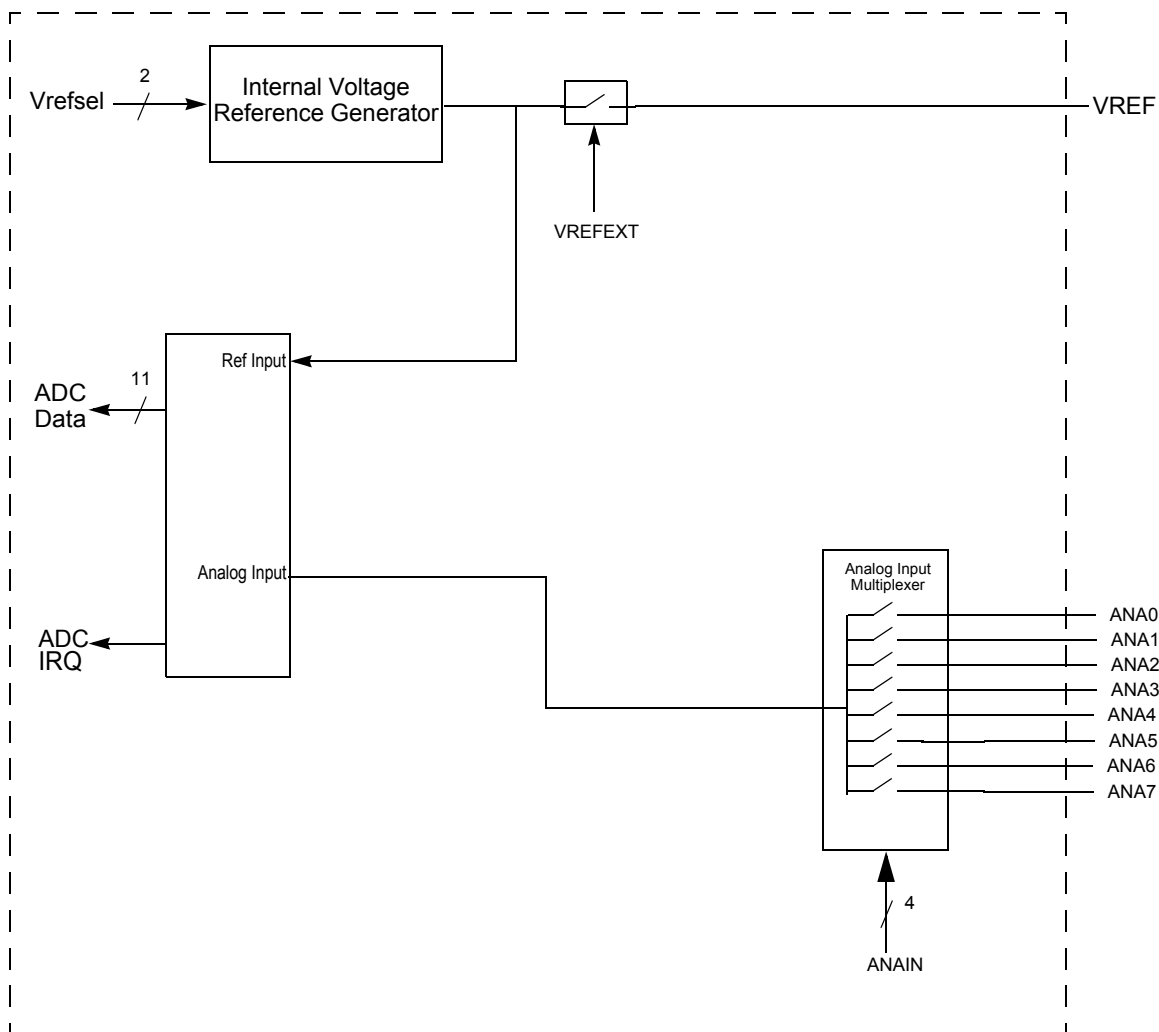


Figure 19. Analog-to-Digital Converter Block Diagram

Operation

Data Format

The output of the ADC is an 11-bit, signed, two's complement digital value. The output generally ranges from 0 to +1023, but offset errors can cause small negative values.

The ADC registers return 13 bits of data, but the two LSBs are intended for compensation use only. When the compensation routine is performed on the 13 bit raw ADC value, two

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 = ANA0

0001 = ANA1

0010 = ANA2

0011 = ANA3

0100 = ANA4

0101 = ANA5

0110 = ANA6

0111 = ANA7

1000 = Reserved

1001 = Reserved

1010 = Reserved

1011 = Reserved

1100 = Reserved

1101 = Reserved

1110 = Reserved

1111 = Reserved

- Rotate and Shift

Tables 107 through Table 114 contain the instructions belonging to each group and the number of operands required for each instruction. Some instructions appear in more than one table as these instruction can be considered as a subset of more than one category. Within these tables, the source operand is identified as ‘src’, the destination operand is ‘dst’ and a condition code is ‘cc’.

Table 107. Arithmetic Instructions

Mnemonic	Operands	Instruction
ADC	dst, src	Add with Carry
ADCX	dst, src	Add with Carry using Extended Addressing
ADD	dst, src	Add
ADDX	dst, src	Add using Extended Addressing
CP	dst, src	Compare
CPC	dst, src	Compare with Carry
CPCX	dst, src	Compare with Carry using Extended Addressing
CPX	dst, src	Compare using Extended Addressing
DA	dst	Decimal Adjust
DEC	dst	Decrement
DECW	dst	Decrement Word
INC	dst	Increment
INCW	dst	Increment Word
MULT	dst	Multiply
SBC	dst, src	Subtract with Carry
SBCX	dst, src	Subtract with Carry using Extended Addressing
SUB	dst, src	Subtract
SUBX	dst, src	Subtract using Extended Addressing

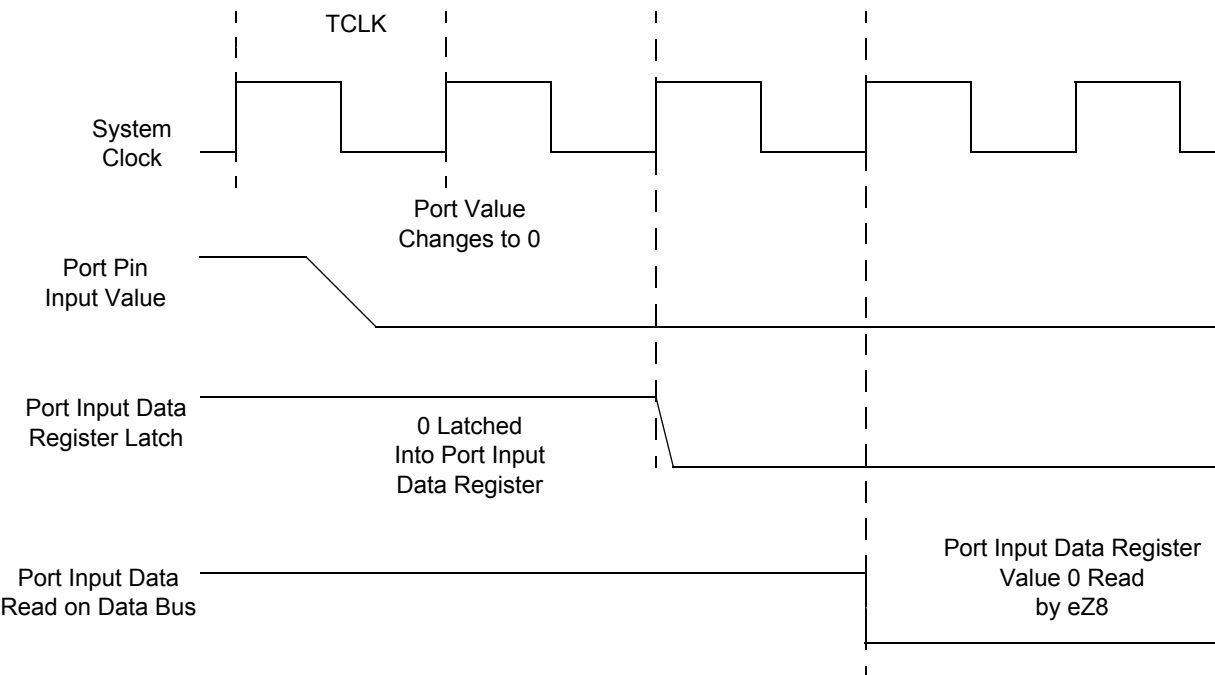


Figure 29. Port Input Sample Timing

Table 127. GPIO Port Input Timing

Parameter Abbreviation		Delay (ns)	
		Minimum	Maximum
T _{S_PORT}	Port Input Transition to XIN Rise Setup Time (Not pictured)	5	—
T _{H_PORT}	XIN Rise to Port Input Transition Hold Time (Not pictured)	0	—
T _{SMR}	GPIO Port Pin Pulse Width to ensure Stop Mode Recovery (for GPIO Port Pins enabled as SMR sources)	1 μ s	

page erase 135
page select register 138, 139
FPS register 138, 139
FSTAT register 137

G

GATED mode 84
general-purpose I/O 35
GPIO 4, 35
 alternate functions 36
 architecture 36
 control register definitions 43
 input data sample timing 202
 interrupts 43
 port A-C pull-up enable sub-registers 48, 49
 port A-H address registers 44
 port A-H alternate function sub-registers 45
 port A-H control registers 44
 port A-H data direction sub-registers 45
 port A-H high drive enable sub-registers 47
 port A-H input data registers 49
 port A-H output control sub-registers 46
 port A-H output data registers 50
 port A-H stop mode recovery sub-registers 47
 port availability by device 35
 port input timing 203
 port output timing 204

H

H 174
HALT 176
halt mode 32, 176
hexadecimal number prefix/suffix 174

I

I2C 4
IM 173
immediate data 173
immediate operand prefix 174

INC 175
increment 175
increment word 175
INCW 175
indexed 173
indirect address prefix 174
indirect register 173
indirect register pair 173
indirect working register 173
indirect working register pair 173
infrared encoder/decoder (IrDA) 113
Instruction Set 171
instruction set, eZ8 CPU 171
instructions
 ADC 175
 ADCX 175
 ADD 175
 ADDX 175
 AND 177
 ANDX 177
 arithmetic 175
 BCLR 176
 BIT 176
 bit manipulation 176
 block transfer 176
 BRK 178
 BSET 176
 BSWAP 176, 178
 BTJ 178
 BTJNZ 178
 BTJZ 178
 CALL 178
 CCF 176
 CLR 177
 COM 177
 CP 175
 CPC 175
 CPCX 175
 CPU control 176
 CPX 175
 DA 175
 DEC 175
 DECW 175
 DI 176

- ADC data high byte (ADCDH) 124
- ADC data low bits (ADCDL) 125
- flash control (FCTL) 137, 143, 144
- flash high and low byte (FFREQH and FREEQL) 139
- flash page select (FPS) 138, 139
- flash status (FSTAT) 137
- GPIO port A-H address (PxADDR) 44
- GPIO port A-H alternate function sub-registers 46
- GPIO port A-H control address (PxCTL) 45
- GPIO port A-H data direction sub-registers 45
- OCD control 161
- OCD status 163
- UARTx baud rate high byte (UxBRH) 110
- UARTx baud rate low byte (UxBRL) 110
- UARTx Control 0 (UxCTL0) 107, 110
- UARTx control 1 (UxCTL1) 108
- UARTx receive data (UxRXD) 105
- UARTx status 0 (UxSTAT0) 105
- UARTx status 1 (UxSTAT1) 106
- UARTx transmit data (UxTXD) 104
- Watchdog Timer control (WDTCTL) 90, 128
- watch-dog timer control (WDTCTL) 167
- Watchdog Timer reload high byte (WDTH) 91
- Watchdog Timer reload low byte (WDTL) 91
- Watchdog Timer reload upper byte (WD-TU) 91
- register file 13
- register pair 173
- register pointer 174
- reset
 - and stop mode characteristics 22
 - and stop mode recovery 21
 - carry flag 176
 - sources 22
- RET 178
- return 178
- RL 178
- RLC 178
- rotate and shift instructions 178
- rotate left 178
- rotate left through carry 178
- rotate right 178
- rotate right through carry 178
- RP 174
- RR 173, 178
- rr 173
- RRC 178
- S**
- SBC 175
- SCF 176, 177
- second opcode map after 1FH 191
- set carry flag 176, 177
- set register pointer 177
- shift right arithmetic 179
- shift right logical 179
- signal descriptions 9
- single-sho conversion (ADC) 119
- software trap 178
- source operand 174
- SP 174
- SRA 179
- src 174
- SRL 179
- SRP 177
- stack pointer 174
- STOP 177
- STOP mode 31, 177
- Stop Mode Recovery
 - sources 26
 - using a GPIO port pin transition 27, 28
 - using Watchdog Timer time-out 27
- SUB 175
- subtract 175
- subtract - extended addressing 175
- subtract with carry 175
- subtract with carry - extended addressing 175
- SUBX 175
- SWAP 179
- swap nibbles 179
- symbols, additional 174

T

- TCM 176
- TCMX 176
- test complement under mask 176
- test complement under mask - extended addressing 176
- test under mask 176
- test under mask - extended addressing 176
- timer signals 9
- timers 67
 - architecture 67
 - block diagram 67
 - CAPTURE mode 74, 75, 84, 85
 - CAPTURE/COMPARE mode 78, 85
 - COMPARE mode 76, 84
 - CONTINUOUS mode 69, 84
 - COUNTER mode 70, 71
 - COUNTER modes 84
 - GATED mode 77, 84
 - ONE-SHOT mode 68, 84
 - operating mode 68
 - PWM mode 72, 73, 84, 85
 - reading the timer count values 79
 - reload high and low byte registers 80
 - timer control register definitions 80
 - timer output signal operation 79
- timers 0-3
 - control registers 82, 83
 - high and low byte registers 80, 81
- TM 176
- TMX 176
- tools, hardware and software 226
- transmit
 - IrDA data 114
- transmitting UART data-polled method 95
- transmitting UART dat-interrupt-driven method 96
- TRAP 178

U

- UART 4
 - architecture 93
 - baud rate generator 103

- control register definitions 104
- controller signals 9
- data format 94
- interrupts 101
- MULTIPROCESSOR mode 99
- receiving data using interrupt-driven method 98
- receiving data using the polled method 97
- transmitting data using the interrupt-driven method 96
- transmitting data using the polled method 95
- x baud rate high and low registers 110
- x control 0 and control 1 registers 107
- x status 0 and status 1 registers 105, 106

- UxBRH register 110
- UxBRL register 110
- UxCTL0 register 107, 110
- UxCTL1 register 108
- UxRXD register 105
- UxSTAT0 register 105
- UxSTAT1 register 106
- UxTXD register 104

V

- vector 173
- Voltage Brownout reset (VBR) 24

W

- Watchdog Timer
 - approximate time-out delay 87
 - CNTL 24
 - control register 89, 127, 167
 - electrical characteristics and timing 200, 202
 - interrupt in normal operation 88
 - interrupt in STOP mode 88
 - refresh 88, 177
 - reload unlock sequence 89
 - reload upper, high and low registers 90
 - reset 25
 - reset in normal operation 89