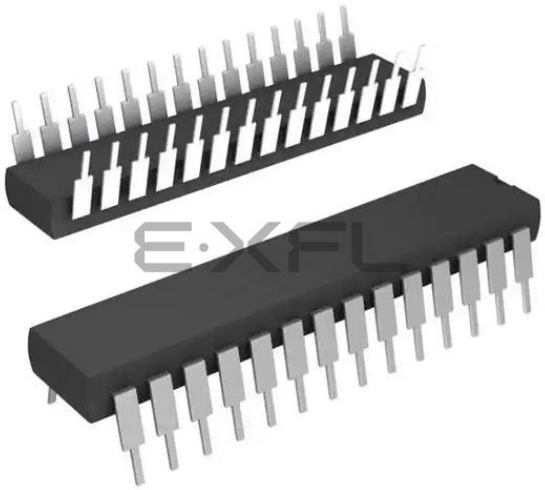


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, Temp Sensor, WDT
Number of I/O	23
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	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.600", 15.24mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f042apj020ec

Table 2. Signal Descriptions (Continued)

Signal Mnemonic	I/O	Description
Analog		
ANA[7:0]	I	Analog Port. These signals are used as inputs to the analog-to-digital converter (ADC).
VREF	I/O	Analog-to-digital converter reference voltage input, or buffered output for internal reference.
Low-Power Operational Amplifier (LPO)		
AMPINP/AMPINN	I	LPO inputs. If enabled, these pins drive the positive and negative amplifier inputs respectively.
AMPOUT	O	LPO output. If enabled, this pin is driven by the on-chip LPO.
Oscillators		
XIN	I	External Crystal Input. This is the input pin to the crystal oscillator. A crystal can be connected between it and the **XOUT** pin to form the oscillator. In addition, this pin is used with external RC networks or external clock drivers to provide the system clock.
XOUT	O	External Crystal Output. This pin is the output of the crystal oscillator. A crystal can be connected between it and the **XIN** pin to form the oscillator.
Clock Input		
CLKIN	I	Clock Input Signal. This pin may be used to input a TTL-level signal to be used as the system clock.
LED Drivers		
LED	O	Direct LED drive capability. All port C pins have the capability to drive an LED without any other external components. These pins have programmable drive strengths set by the GPIO block.
On-Chip Debugger		
DBG	I/O	Debug. This signal is the control and data input and output to and from the On-Chip Debugger. **Caution:** *The DBG pin is open-drain and requires a pull-up resistor to ensure proper operation.*
Reset		
$\overline{\text{RESET}}$	I/O	RESET. Generates a Reset when asserted (driven Low). Also serves as a reset indicator; the Z8 Encore! XP forces this pin low when in reset. This pin is open-drain and features an enabled internal pull-up resistor.

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	If 04H in Port A–D Address Register, accessible through the Port A–D Control 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 Recovery 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	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

Also, if the Timer Output alternate function is enabled, the Timer Output pin changes state for one system clock cycle (from Low to High or from High to Low) upon timer Reload. If it is appropriate to have the Timer Output make a state change at a One-Shot time-out (rather than a single cycle pulse), first set the TPOL bit in the Timer Control Register to the start value before enabling ONE-SHOT mode. After starting the timer, set TPOL to the opposite bit value.

Follow the steps below for configuring a timer for ONE-SHOT mode and initiating the count:

1. Write to the Timer Control register to:
   - Disable the timer
   - Configure the timer for ONE-SHOT mode.
   - Set the prescale value.
   - Set the initial output level (High or Low) if using the Timer Output alternate function.
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 Reload value.
4. If appropriate, enable the timer interrupt 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 ONE-SHOT mode, the system clock always provides the timer input. The timer period is given by the following equation:

$$\text{ONE-SHOT Mode Time-Out Period (s)} = \frac{\text{Reload Value} - \text{Start Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

CONTINUOUS Mode

In CONTINUOUS mode, 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. Upon 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. 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 CONTINUOUS mode and initiating the count:

1. Write to the Timer Control register to:
   - Disable the timer
   - Configure the timer for CONTINUOUS mode.

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:

COUNTER Mode Timer Input Transitions = Current Count Value-Start Value

COMPARATOR COUNTER Mode

In COMPARATOR COUNTER mode, the timer counts input transitions from the analog comparator output. The TPOL bit in the Timer Control Register selects whether the count occurs on the rising edge or the falling edge of the comparator output signal. In COMPARATOR COUNTER mode, the prescaler is disabled.

Caution: *The frequency of the comparator output signal must not exceed one-fourth the system clock frequency. Further, the high or low state of the comparator output signal pulse must be no less than twice the system clock period. A shorter pulse may not be captured.*

After 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.

If the TPOL bit in the Timer Control register is set to 0, the Timer Output signal begins as a Low (0) and transitions to a High (1) when the timer value matches the PWM value. The Timer Output signal returns to a Low (0) after the timer reaches the Reload value and is reset to 0001H.

Follow the steps below for configuring a timer for PWM SINGLE OUTPUT mode and initiating the PWM operation:

1. Write to the Timer Control register to:
   - Disable the timer.
   - Configure the timer for PWM SINGLE OUTPUT mode.
   - Set the prescale value.
   - Set the initial logic level (High or Low) and PWM High/Low transition for the Timer Output alternate function.
2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H). This only affects the first pass in PWM mode. After the first timer reset in PWM mode, counting always begins at the reset value of 0001H.
3. Write to the PWM High and Low Byte registers to set the PWM value.
4. Write to the Timer Reload High and Low Byte registers to set the Reload value (PWM period). The Reload value must be greater than the PWM value.
5. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
6. Configure the associated GPIO port pin for the Timer Output alternate function.
7. Write to the Timer Control register to enable the timer and initiate counting.

The PWM period is represented by the following equation:

$$\text{PWM Period (s)} = \frac{\text{Reload Value} \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

If an initial starting value other than 0001H is loaded into the Timer High and Low Byte registers, use the ONE-SHOT mode equation to determine the first PWM time-out period.

If TPOL is set to 0, the ratio of the PWM output High time to the total period is represented by:

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{Reload Value} - \text{PWM Value}}{\text{Reload Value}} \times 100$$

If TPOL is set to 1, the ratio of the PWM output High time to the total period is represented by:

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{PWM Value}}{\text{Reload Value}} \times 100$$

2. Write to the Timer High and Low Byte registers to set the starting count value. Writing these registers only affects the first pass in GATED mode. After the first timer reset in GATED mode, counting always begins at the reset value of 0001H.
3. Write to the Timer Reload High and Low Byte registers to set the Reload 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 is generated for both input deassertion and reload events. If appropriate, configure the timer interrupt to be generated only at the input deassertion 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. Assert the Timer Input signal to initiate the counting.

CAPTURE/COMPARE Mode

In CAPTURE/COMPARE mode, the timer begins counting on the first external Timer Input transition. The acceptable transition (rising edge or falling edge) is set by the TPOL bit in the Timer Control Register. The timer input is the system clock.

Every subsequent acceptable transition (after the first) of the Timer Input signal captures the current count value. The Capture value is written to the Timer PWM High and Low Byte Registers. When the Capture event occurs, an interrupt is generated, the count value in the Timer High and Low Byte registers is reset to 0001H, and counting resumes. The INPCAP bit in TxCTL0 register is set to indicate the timer interrupt is caused by an input capture event.

If no Capture event occurs, the timer counts up to the 16-bit Compare value stored in the Timer Reload High and Low Byte registers. Upon reaching the Compare value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. The INPCAP bit in TxCTL0 register is cleared to indicate the timer interrupt is not because of an input capture event.

Follow the steps below for configuring a timer for CAPTURE/COMPARE mode and initiating the count:

1. Write to the Timer Control register to:
   - Disable the timer.
   - Configure the timer for CAPTURE/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.

The UART is now configured for interrupt-driven data transmission. Because the UART Transmit Data register is empty, an interrupt is generated immediately. When the UART Transmit interrupt is detected, the associated interrupt service routine (ISR) performs the following:

1. Write the UART Control 1 register to select the multiprocessor bit for the byte to be transmitted:
2. Set the Multiprocessor Bit Transmitter (MPBT) if sending an address byte, clear it if sending a data byte.
3. Write the data byte to the UART Transmit Data register. The transmitter automatically transfers the data to the Transmit Shift register and transmits the data.
4. Clear the UART Transmit interrupt bit in the applicable Interrupt Request register.
5. Execute the IRET instruction to return from the interrupt-service routine and wait for the Transmit Data register to again become empty.

Receiving Data using the Polled Method

Follow the steps below to configure the UART for polled data reception:

1. Write to the UART Baud Rate High and Low Byte registers to set an acceptable baud rate for the incoming data stream.
2. Enable the UART pin functions by configuring the associated GPIO Port pins for alternate function operation.
3. Write to the UART Control 1 register to enable MULTIPROCESSOR mode functions, if appropriate.
4. 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.
5. Check the RDA bit in the UART Status 0 register to determine if the Receive Data register contains a valid data byte (indicated by a 1). If RDA is set to 1 to indicate available data, continue to Step 5. If the Receive Data register is empty (indicated by a 0), continue to monitor the RDA bit awaiting reception of the valid data.
6. Read data from the UART Receive Data register. If operating in MULTIPROCESSOR (9-bit) mode, further actions may be required depending on the MULTIPROCESSOR mode bits MPMD[1:0].
7. Return to Step 4 to receive additional data.

Comparator

The Z8 Encore! XP® F082A Series devices feature a general purpose comparator that compares two analog input signals. These analog signals may be external stimulus from a pin (CINP and/or CINN) or internally generated signals. Both a programmable voltage reference and the temperature sensor output voltage are available internally. The output is available as an interrupt source or can be routed to an external pin.

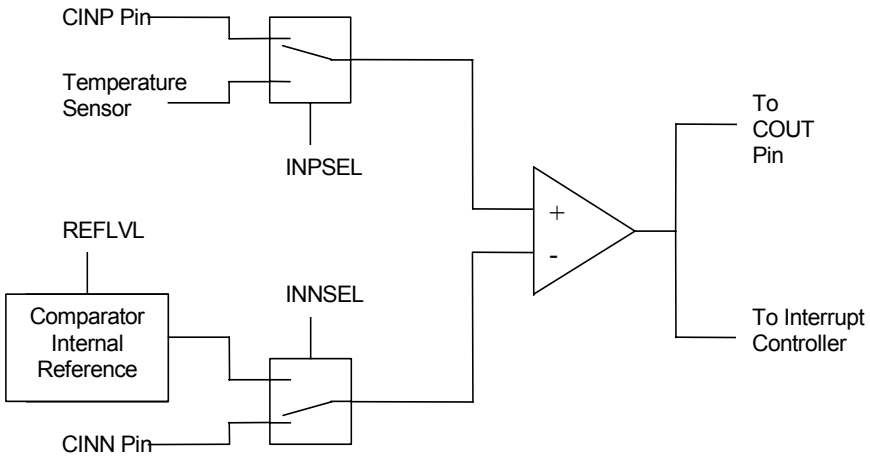


Figure 20. Comparator Block Diagram

Operation

When the positive comparator input exceeds the negative input by more than the specified hysteresis, the output is a logic HIGH. When the negative input exceeds the positive by more than the hysteresis, the output is a logic LOW. Otherwise, the comparator output retains its present value. See Table 137 on page 233 for details.

The comparator may be powered down to reduce supply current. See Power Control Register 0 on page 34 for details.

Caution: *Because of the propagation delay of the comparator, it is not recommended to enable or reconfigure the comparator without first disabling interrupts and waiting for the comparator output to settle. Doing so can result in spurious interrupts. The following example describes how to safely enable the comparator:*

```
di
ld cmp0, r0 ; load some new configuration
nop
```

Table 108. Oscillator Configuration and Selection

Clock Source	Characteristics	Required Setup
Internal Precision RC Oscillator	• 32.8 kHz or 5.53 MHz • High accuracy • No external components required	• Unlock and write Oscillator Control Register (OSCCTL) to enable and select oscillator at either 5.53 MHz or 32.8 kHz
External Crystal/ Resonator	• 32 kHz to 20 MHz • Very high accuracy (dependent on crystal or resonator used) • Requires external components	• Configure Flash option bits for correct external oscillator mode • Unlock and write OSCCTL to enable crystal oscillator, wait for it to stabilize and select as system clock (if the XTLDIS option bit has been de-asserted, no waiting is required)
External RC Oscillator	• 32 kHz to 4 MHz • Accuracy dependent on external components	• Configure Flash option bits for correct external oscillator mode • Unlock and write OSCCTL to enable crystal oscillator and select as system clock
External Clock Drive	• 0 to 20 MHz • Accuracy dependent on external clock source	• Write GPIO registers to configure PB3 pin for external clock function • Unlock and write OSCCTL to select external system clock • Apply external clock signal to GPIO
Internal Watchdog Timer Oscillator	• 10 kHz nominal • Low accuracy; no external components required • Very low power consumption	• Enable WDT if not enabled and wait until WDT Oscillator is operating. • Unlock and write Oscillator Control Register (OSCCTL) to enable and select oscillator

Caution: *Unintentional accesses to the oscillator control register can actually stop the chip by switching to a non-functioning oscillator. To prevent this condition, the oscillator control block employs a register unlocking/locking scheme.*

OSC Control Register Unlocking/Locking

To write the oscillator control register, unlock it by making two writes to the OSCCTL register with the values E7H followed by 18H. A third write to the OSCCTL register changes the value of the actual register and returns the register to a locked state. Any other sequence of oscillator control register writes has no effect. The values written to unlock the register must be ordered correctly, but are not necessarily consecutive. It is possible to write to or read from other registers within the unlocking/locking operation.

Caution: *It is possible to disable the clock failure detection circuitry as well as all functioning clock sources. In this case, the Z8 Encore! XP F082A Series device ceases functioning and can only be recovered by Power-On-Reset.*

Oscillator Control Register Definitions

Oscillator Control Register

The Oscillator Control Register (OSCCTL) enables/disables the various oscillator circuits, enables/disables the failure detection/recovery circuitry and selects the primary oscillator, which becomes the system clock.

The Oscillator Control Register must be unlocked before writing. Writing the two step sequence E7H followed by 18H to the Oscillator Control Register unlocks it. The register is locked at successful completion of a register write to the OSCCTL.

Table 109. Oscillator Control Register (OSCCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	INTEN	XTLEN	WDTEN	SOFEN	WDFEN	SCKSEL		
RESET	1	0	1	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F86H							

INTEN—Internal Precision Oscillator Enable
1 = Internal precision oscillator is enabled
0 = Internal precision oscillator is disabled

XTLEN—Crystal Oscillator Enable; this setting overrides the GPIO register control for PA0 and PA1
1 = Crystal oscillator is enabled
0 = Crystal oscillator is disabled

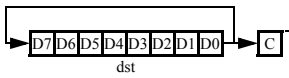
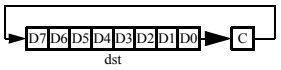
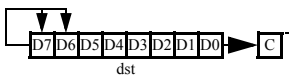
WDTEN—Watchdog Timer Oscillator Enable
1 = Watchdog Timer oscillator is enabled
0 = Watchdog Timer oscillator is disabled

SOFEN—System Clock Oscillator Failure Detection Enable
1 = Failure detection and recovery of system clock oscillator is enabled
0 = Failure detection and recovery of system clock oscillator is disabled

WDFEN—Watchdog Timer Oscillator Failure Detection Enable
1 = Failure detection of Watchdog Timer oscillator is enabled
0 = Failure detection of Watchdog Timer oscillator is disabled

SCKSEL—System Clock Oscillator Select
000 = Internal precision oscillator functions as system clock at 5.53 MHz
001 = Internal precision oscillator functions as system clock at 32 kHz
010 = Crystal oscillator or external RC oscillator functions as system clock
011 = Watchdog Timer oscillator functions as system
100 = External clock signal on PB3 functions as system clock
101 = Reserved
110 = Reserved
111 = Reserved

Table 124. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
RR dst		R		E0	*	*	*	*	–	–	2	2
		IR		E1							2	3
RRC dst		R		C0	*	*	*	*	–	–	2	2
		IR		C1							2	3
SBC dst, src	dst ← dst – src - C	r	r	32	*	*	*	*	1	*	2	3
		r	Ir	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 ← 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		R		1F C0	*	*	0	*	–	–	3	2
		IR		1F C1							3	3
SRP src	RP ← src		IM	01	–	–	–	–	–	–	2	2
STOP	STOP Mode			6F	–	–	–	–	–	–	1	2
SUB dst, src	dst ← dst – src	r	r	22	*	*	*	*	1	*	2	3
		r	Ir	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 the result of the operation.
– = Unaffected
X = Undefined
0 = Reset to 0
1 = Set to 1

Table 124. eZ8 CPU Instruction Summary (Continued)

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

Figure 33 displays the typical current consumption while operating with all peripherals disabled, at 30 ºC, versus the system clock frequency.

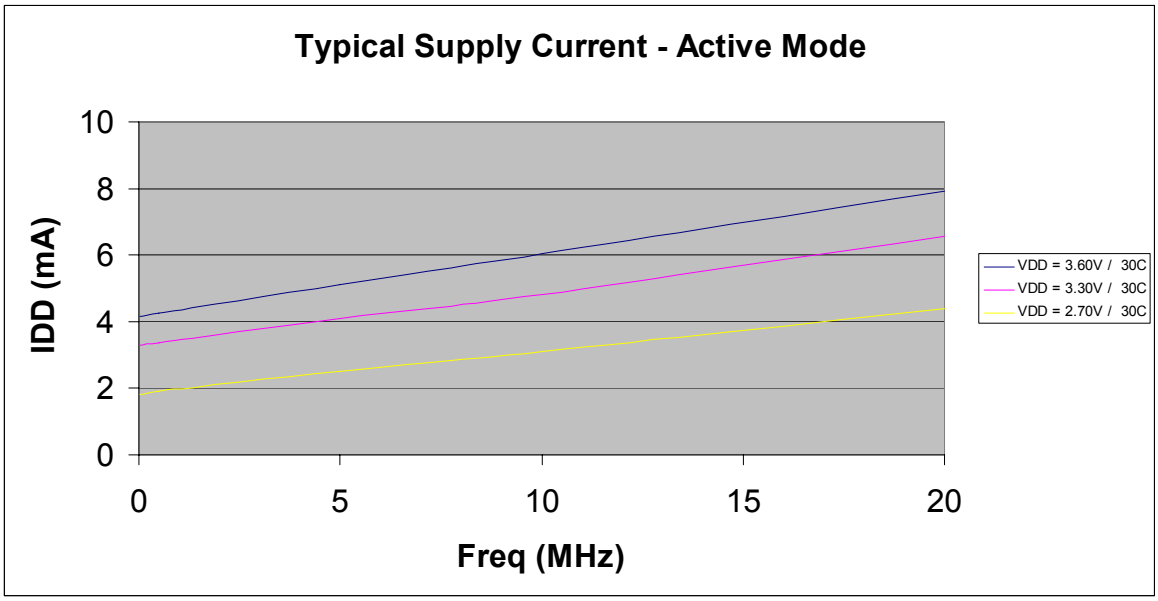


Figure 33. Typical Active Mode I_{DD} Versus System Clock Frequency

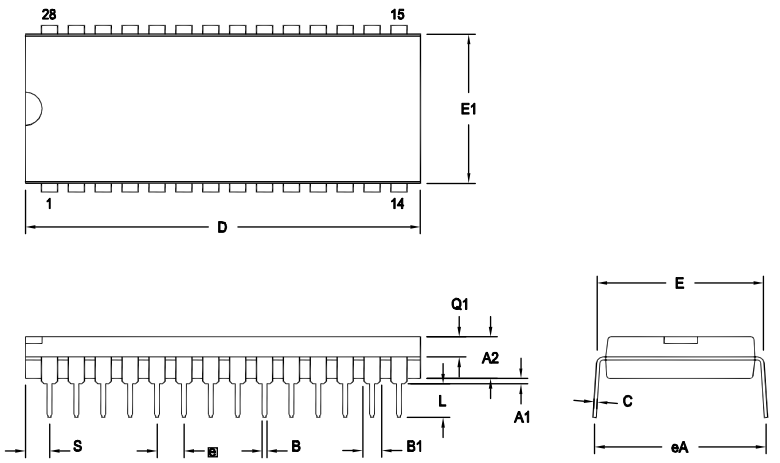
Table 134. Non-Volatile Data Storage

	V_{DD} = 2.7 V to 3.6 V T_A = -40 °C to +105 °C				
Parameter	**Minimum**	**Typical**	**Maximum**	**Units**	**Notes**
NVDS Byte Read Time	34	–	519	µs	With system clock at 20 MHz
NVDS Byte Program Time	0.171	–	39.7	ms	With system clock at 20 MHz
Data Retention	100	–	–	years	25 °C
Endurance	160,000	–	–	cycles	Cumulative write cycles for entire memory

Table 135. Analog-to-Digital Converter Electrical Characteristics and Timing

		V_{DD} = 3.0 V to 3.6 V T_A = 0 °C to +70 °C (unless otherwise stated)				
Symbol	**Parameter**	**Minimum**	**Typical**	**Maximum**	**Units**	**Conditions**
	Resolution	10		–	bits	
	Differential Nonlinearity (DNL)	-1.0	–	1.0	LSB[3]	External V_{REF} = 2.0 V; $R_S \leftarrow$ 3.0 kΩ
	Integral Nonlinearity (INL)	-3.0	–	3.0	LSB[3]	External V_{REF} = 2.0 V; $R_S \leftarrow$ 3.0 kΩ
	Offset Error with Calibration		±1		LSB[3]	
	Absolute Accuracy with Calibration		±3		LSB[3]	
V_{REF}	Internal Reference Voltage	1.0 2.0	1.1 2.2	1.2 2.4	V	REFSEL=01 REFSEL=10
V_{REF}	Internal Reference Variation with Temperature		±1.0		%	Temperature variation with V_{DD} = 3.0
V_{REF}	Internal Reference Voltage Variation with V_{DD}		±0.5		%	Supply voltage variation with T_A = 30 °C
R_{REFOUT}	Reference Buffer Output Impedance		850		Ω	When the internal reference is buffered and driven out to the VREF pin (REFOUT = 1)

Figure 45 displays the 28-pin Plastic Dual Inline Package (PDIP) available for the Z8 Encore! XP F082A Series devices.



SYMBOL	OPT #	MILLIMETER		INCH	
		MIN	MAX	MIN	MAX
A1		0.38	1.02	.015	.040
A2		3.18	4.19	.125	.165
B		0.38	0.53	.015	.021
B1	01	1.40	1.65	.055	.065
	02	1.14	1.40	.045	.055
C		0.23	0.38	.009	.015
D	01	36.58	37.34	1.440	1.470
	02	35.31	35.94	1.390	1.415
E		15.24	15.75	.600	.620
E1	01	13.59	14.10	.535	.555
	02	12.83	13.08	.505	.515
e		2.54 TYP		.100 BSC	
eA		15.49	16.76	.610	.660
L		3.05	3.81	.120	.150
Q1	01	1.40	1.91	.055	.075
	02	1.40	1.78	.055	.070
S	01	1.52	2.29	.060	.090
	02	1.02	1.52	.040	.060

CONTROLLING DIMENSIONS : INCH

OPTION TABLE	
OPTION #	PACKAGE
01	STANDARD
02	IDF

Note: ZiLOG supplies both options for production. Component layout PCB design should cover bigger option 01.

Figure 45. 28-Pin Plastic Dual Inline Package (PDIP)

Part Number	Flash	RAM	NVDS	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Comparator	Temperature Sensor	Description
Z8 Encore! XP® F082A Series with 4 KB Flash, 10-Bit Analog-to-Digital Converter											
Standard Temperature: 0 °C to 70 °C											
Z8F042APB020SC	4 KB	1 KB	128 B	6	14	2	4	1	1	1	PDIP 8-pin package
Z8F042AQB020SC	4 KB	1 KB	128 B	6	14	2	4	1	1	1	QFN 8-pin package
Z8F042ASB020SC	4 KB	1 KB	128 B	6	14	2	4	1	1	1	SOIC 8-pin package
Z8F042ASH020SC	4 KB	1 KB	128 B	17	20	2	7	1	1	1	SOIC 20-pin package
Z8F042AHH020SC	4 KB	1 KB	128 B	17	20	2	7	1	1	1	SSOP 20-pin package
Z8F042APH020SC	4 KB	1 KB	128 B	17	20	2	7	1	1	1	PDIP 20-pin package
Z8F042ASJ020SC	4 KB	1 KB	128 B	23	20	2	8	1	1	1	SOIC 28-pin package
Z8F042AHJ020SC	4 KB	1 KB	128 B	23	20	2	8	1	1	1	SSOP 28-pin package
Z8F042APJ020SC	4 KB	1 KB	128 B	23	20	2	8	1	1	1	PDIP 28-pin package
Extended Temperature: -40 °C to 105 °C											
Z8F042APB020EC	4 KB	1 KB	128 B	6	14	2	4	1	1	1	PDIP 8-pin package
Z8F042AQB020EC	4 KB	1 KB	128 B	6	14	2	4	1	1	1	QFN 8-pin package
Z8F042ASB020EC	4 KB	1 KB	128 B	6	14	2	4	1	1	1	SOIC 8-pin package
Z8F042ASH020EC	4 KB	1 KB	128 B	17	20	2	7	1	1	1	SOIC 20-pin package
Z8F042AHH020EC	4 KB	1 KB	128 B	17	20	2	7	1	1	1	SSOP 20-pin package
Z8F042APH020EC	4 KB	1 KB	128 B	17	20	2	7	1	1	1	PDIP 20-pin package
Z8F042ASJ020EC	4 KB	1 KB	128 B	23	20	2	8	1	1	1	SOIC 28-pin package
Z8F042AHJ020EC	4 KB	1 KB	128 B	23	20	2	8	1	1	1	SSOP 28-pin package
Z8F042APJ020EC	4 KB	1 KB	128 B	23	20	2	8	1	1	1	PDIP 28-pin package
Replace C with G for Lead-Free Packaging											

Part Number	Flash	RAM	NVDS	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	UART with IrDA	Comparator	Temperature Sensor	Description
Z8 Encore! XP® F082A Series with 2 KB Flash											
Standard Temperature: 0 °C to 70 °C											
Z8F021APB020SC	2 KB	512 B	64 B	6	13	2	0	1	1	0	PDIP 8-pin package
Z8F021AQB020SC	2 KB	512 B	64 B	6	13	2	0	1	1	0	QFN 8-pin package
Z8F021ASB020SC	2 KB	512 B	64 B	6	13	2	0	1	1	0	SOIC 8-pin package
Z8F021ASH020SC	2 KB	512 B	64 B	17	19	2	0	1	1	0	SOIC 20-pin package
Z8F021AHH020SC	2 KB	512 B	64 B	17	19	2	0	1	1	0	SSOP 20-pin package
Z8F021APH020SC	2 KB	512 B	64 B	17	19	2	0	1	1	0	PDIP 20-pin package
Z8F021ASJ020SC	2 KB	512 B	64 B	25	19	2	0	1	1	0	SOIC 28-pin package
Z8F021AHJ020SC	2 KB	512 B	64 B	25	19	2	0	1	1	0	SSOP 28-pin package
Z8F021APJ020SC	2 KB	512 B	64 B	25	19	2	0	1	1	0	PDIP 28-pin package
Extended Temperature: -40 °C to 105 °C											
Z8F021APB020EC	2 KB	512 B	64 B	6	13	2	0	1	1	0	PDIP 8-pin package
Z8F021AQB020EC	2 KB	512 B	64 B	6	13	2	0	1	1	0	QFN 8-pin package
Z8F021ASB020EC	2 KB	512 B	64 B	6	13	2	0	1	1	0	SOIC 8-pin package
Z8F021ASH020EC	2 KB	512 B	64 B	17	19	2	0	1	1	0	SOIC 20-pin package
Z8F021AHH020EC	2 KB	512 B	64 B	17	19	2	0	1	1	0	SSOP 20-pin package
Z8F021APH020EC	2 KB	512 B	64 B	17	19	2	0	1	1	0	PDIP 20-pin package
Z8F021ASJ020EC	2 KB	512 B	64 B	25	19	2	0	1	1	0	SOIC 28-pin package
Z8F021AHJ020EC	2 KB	512 B	64 B	25	19	2	0	1	1	0	SSOP 28-pin package
Z8F021APJ020EC	2 KB	512 B	64 B	25	19	2	0	1	1	0	PDIP 28-pin package
Replace C with G for Lead-Free Packaging											