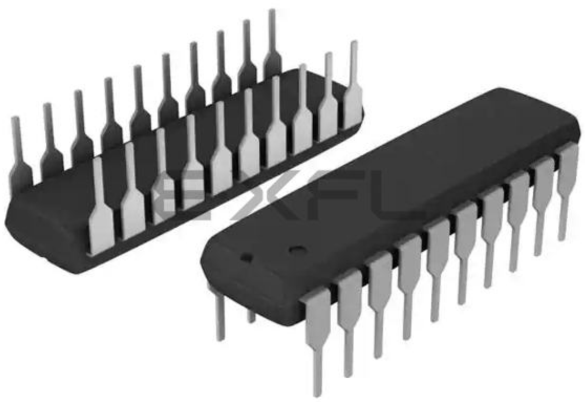

Welcome to E-XFL.COM

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, WDT
Number of I/O	17
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	128 x 8
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Through Hole
Package / Case	20-DIP (0.300", 7.62mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f041aph020ec

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.*

Table 14. Port Alternate Function Mapping (Non 8-Pin Parts)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port A	PA0	T0IN/$\overline{\text{T0OUT}}$*	Timer 0 Input/Timer 0 Output Complement	N/A
		Reserved		
	PA1	T0OUT	Timer 0 Output	
		Reserved		
	PA2	DE0	UART 0 Driver Enable	
		Reserved		
	PA3	$\overline{\text{CTS0}}$	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/$\overline{\text{T1OUT}}$*	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–D Alternate Function Sub-Registers on page 47 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 82.

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

PAFS1[7:0]—Port Alternate Function Set 1
0 = Port Alternate Function selected as defined in Table 14 and Table 15 on page 44.
1 = Port Alternate Function selected as defined in Table 14 and Table 15 on page 44.

Port A–D Alternate Function Set 2 Sub-Registers

The Port A–D Alternate Function Set 2 sub-register (Table 26) is accessed through the Port A–D Control register by writing 08H to the Port A–D Address register. The Alternate Function Set 2 sub-registers selects the alternate function available at a port pin. Alternate Functions selected by setting or clearing bits of this register is defined in Table 15.

Note: *Alternate function selection on port pins must also be enabled as described in Port A–D Alternate Function Sub-Registers on page 47.*

Table 26. Port A–D Alternate Function Set 2 Sub-Registers (PxAFS2)

BITS	7	6	5	4	3	2	1	0
FIELD	PAFS27	PAFS26	PAFS25	PAFS24	PAFS23	PAFS22	PAFS21	PAFS20
RESET	00H (all ports of 20/28 pin devices); 04H (Port A of 8-pin device)							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	If 08H in Port A–D Address Register, accessible through the Port A–D Control Register							

PAFS2[7:0]—Port Alternate Function Set 2
0 = Port Alternate Function selected as defined in Table 15.
1 = Port Alternate Function selected as defined in Table 15.

Port A–C Input Data Registers

Reading from the Port A–C Input Data registers (Table 27) returns the sampled values from the corresponding port pins. The Port A–C Input Data registers are read-only. The value returned for any unused ports is 0. Unused ports include those missing on the 8- and 28-pin packages, as well as those missing on the ADC-enabled 28-pin packages.

Table 27. 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							
X = Undefined.								

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.

UART Address Compare Register

The UART Address Compare (UxADDR) register stores the multi-node network address of the UART (see Table 67). When the MPMD[1] bit of UART Control Register 0 is set, all incoming address bytes are compared to the value stored in the Address Compare register. Receive interrupts and RDA assertions only occur in the event of a match.

Table 67. UART Address Compare Register (U0ADDR)

BITS	7	6	5	4	3	2	1	0
FIELD	COMP_ADDR							
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	F45H							

COMP_ADDR—Compare Address
This 8-bit value is compared to incoming address bytes.

UART Baud Rate High and Low Byte Registers

The UART Baud Rate High (UxBRH) and Low Byte (UxBRL) registers (Table 68 and Table 69) combine to create a 16-bit baud rate divisor value (BRG[15:0]) that sets the data transmission rate (baud rate) of the UART.

Table 68. UART Baud Rate High Byte Register (U0BRH)

BITS	7	6	5	4	3	2	1	0
FIELD	BRH							
RESET	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F46H							

Table 69. UART Baud Rate Low Byte Register (U0BRL)

BITS	7	6	5	4	3	2	1	0
FIELD	BRL							
RESET	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADDR	F47H							

The UART data rate is calculated using the following equation:

$$\text{UART Baud Rate (bits/s)} = \frac{\text{System Clock Frequency (Hz)}}{16 \times \text{UART Baud Rate Divisor Value}}$$

For a given UART data rate, calculate the integer baud rate divisor value using the following equation:

$$\text{UART Baud Rate Divisor Value (BRG)} = \text{Round}\left(\frac{\text{System Clock Frequency (Hz)}}{16 \times \text{UART Data Rate (bits/s)}}\right)$$

The baud rate error relative to the acceptable baud rate is calculated using the following equation:

$$\text{UART Baud Rate Error (\%)} = 100 \times \left(\frac{\text{Actual Data Rate} - \text{Desired Data Rate}}{\text{Desired Data Rate}}\right)$$

For reliable communication, the UART baud rate error must never exceed 5 percent. Table 70 provides information on the data rate errors for popular baud rates and commonly used crystal oscillator frequencies.

Table 70. UART Baud Rates

10.0 MHz System Clock				5.5296 MHz System Clock			
Acceptable Rate (kHz)	**BRG Divisor (Decimal)**	**Actual Rate (kHz)**	**Error (%)**	**Acceptable Rate (kHz)**	**BRG Divisor (Decimal)**	**Actual Rate (kHz)**	**Error (%)**
1250.0	N/A	N/A	N/A	1250.0	N/A	N/A	N/A
625.0	1	625.0	0.00	625.0	N/A	N/A	N/A
250.0	3	208.33	-16.67	250.0	1	345.6	38.24
115.2	5	125.0	8.51	115.2	3	115.2	0.00
57.6	11	56.8	-1.36	57.6	6	57.6	0.00
38.4	16	39.1	1.73	38.4	9	38.4	0.00
19.2	33	18.9	0.16	19.2	18	19.2	0.00
9.60	65	9.62	0.16	9.60	36	9.60	0.00
4.80	130	4.81	0.16	4.80	72	4.80	0.00
2.40	260	2.40	-0.03	2.40	144	2.40	0.00
1.20	521	1.20	-0.03	1.20	288	1.20	0.00
0.60	1042	0.60	-0.03	0.60	576	0.60	0.00
0.30	2083	0.30	0.2	0.30	1152	0.30	0.00

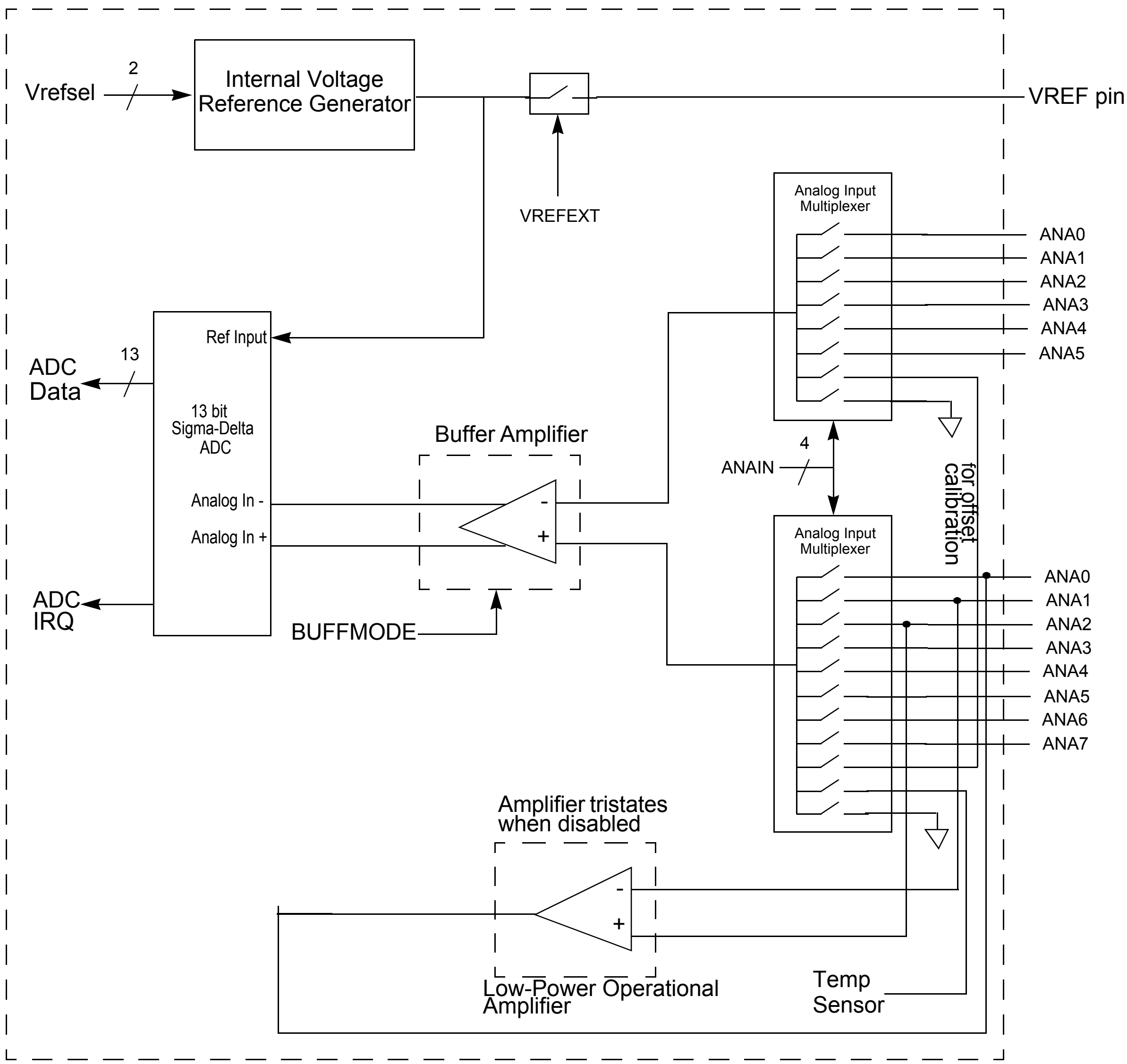

Figure 19. Analog-to-Digital Converter Block Diagram

Operation

Data Format

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

The ADC registers actually return 13 bits of data, but the two LSBs are intended for compensation use only. When the software compensation routine is performed on the 13 bit raw ADC value, two bits of resolution are lost because of a rounding error. As a result, the final value is an 11-bit number.

Hardware Overflow

When the hardware overflow bit (OVF) is set in ADC Data Low Byte (ADCD_L) register, all other data bits are invalid. The hardware overflow bit is set for values greater than V_{ref} and less than $-V_{ref}$ (DIFFERENTIAL mode).

Automatic Powerdown

If the ADC is idle (no conversions in progress) for 160 consecutive system clock cycles, portions of the ADC are automatically powered down. From this powerdown state, the ADC requires 40 system clock cycles to power up. The ADC powers up when a conversion is requested by the ADC Control register.

Single-Shot Conversion

When configured for single-shot conversion, the ADC performs a single analog-to-digital conversion on the selected analog input channel. After completion of the conversion, the ADC shuts down. Follow the steps below for setting up the ADC and initiating a single-shot conversion:

1. Enable the desired analog inputs by configuring the general-purpose I/O pins for alternate analog function. This configuration disables the digital input and output drivers.
2. Write the ADC Control/Status Register 1 to configure the ADC.
   - Write to `BUFMODE[2:0]` to select SINGLE-ENDED or DIFFERENTIAL mode, as well as unbuffered or buffered mode.
   - Write the `REFSELH` bit of the pair {`REFSELH`, `REFSELL`} to select the internal voltage reference level or to disable the internal reference. The `REFSELL` bit is. contained in the ADC Control Register 0.
3. Write to the ADC Control Register 0 to configure the ADC and begin the conversion. The bit fields in the ADC Control register can be written simultaneously (the ADC can be configured and enabled with the same write instruction):
   - Write to the `ANAIN[3:0]` field to select from the available analog input sources (different input pins available depending on the device).
   - Clear `CONT` to 0 to select a single-shot conversion.

1001 = 1.8 V
1010–1111 = Reserved

For 8-pin devices:

000000 = 0.00 V
000001 = 0.05 V
000010 = 0.10 V
000011 = 0.15 V
000100 = 0.20 V
000101 = 0.25 V
000110 = 0.30 V
000111 = 0.35 V
001000 = 0.40 V
001001 = 0.45 V
001010 = 0.50 V
001011 = 0.55 V
001100 = 0.60 V
001101 = 0.65 V
001110 = 0.70 V
001111 = 0.75 V
010000 = 0.80 V
010001 = 0.85 V
010010 = 0.90 V
010011 = 0.95 V
010100 = 1.00 V (Default)
010101 = 1.05 V
010110 = 1.10 V
010111 = 1.15 V
011000 = 1.20 V
011001 = 1.25 V
011010 = 1.30 V
011011 = 1.35 V
011100 = 1.40 V
011101 = 1.45 V
011110 = 1.50 V
011111 = 1.55 V
100000 = 1.60 V
100001 = 1.65 V
100010 = 1.70 V
100011 = 1.75 V
100100 = 1.80 V

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:

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

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® 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.

Table 102. Randomized Lot ID Locations (Continued)

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)

read operations to illegal addresses. Also, the user code must pop the address byte off the stack.

The read routine uses 9 bytes of stack space in addition to the one byte of address pushed by the user. Sufficient memory must be available for this stack usage.

Because of the Flash memory architecture, NVDS reads exhibit a non-uniform execution time. A read operation takes between 44 µs and 489 µs (assuming a 20 MHz system clock). Slower system clock speeds result in proportionally higher execution times.

NVDS byte reads from invalid addresses (those exceeding the NVDS array size) return 0xff. Illegal read operations have a 2 µs execution time.

The status byte returned by the NVDS read routine is zero for successful read, as determined by a CRC check. If the status byte is non-zero, there was a corrupted value in the NVDS array at the location being read. In this case, the value returned in R0 is the byte most recently written to the array that does not have a CRC error.

Power Failure Protection

The NVDS routines employ error checking mechanisms to ensure a power failure endangers only the most recently written byte. Bytes previously written to the array are not perturbed.

A system reset (such as a pin reset or Watchdog Timer reset) that occurs during a write operation also perturbs the byte currently being written. All other bytes in the array are unperturbed.

Optimizing NVDS Memory Usage for Execution Speed

The NVDS read time varies drastically, this discrepancy being a trade-off for minimizing the frequency of writes that require post-write page erases (see Table 104). The NVDS read time of address N is a function of the number of writes to addresses other than N since the most recent write to address N, as well as the number of writes since the most recent page erase. Neglecting effects caused by page erases and results caused by the initial condition in which the NVDS is blank, a rule of thumb is that every write since the most recent page erase causes read times of unwritten addresses to increase by 1 µs, up to a maximum of (511-NVDS_SIZE) µs.

Table 104. NVDS Read Time

Operation	Minimum Latency	Maximum Latency
Read (16 byte array)	875	9961
Read (64 byte array)	876	8952

Breakpoints in Flash Memory

The BRK instruction is opcode 00H, which corresponds to the fully programmed state of a byte in Flash memory. To implement a Breakpoint, write 00H to the required break address, overwriting the current instruction. To remove a Breakpoint, the corresponding page of Flash memory must be erased and reprogrammed with the original data.

Runtime Counter

The On-Chip Debugger contains a 16-bit Runtime Counter. It counts system clock cycles between Breakpoints. The counter starts counting when the On-Chip Debugger leaves DEBUG mode and stops counting when it enters DEBUG mode again or when it reaches the maximum count of FFFFH.

On-Chip Debugger Commands

The host communicates to the on-chip debugger by sending OCD commands using the DBG interface. During normal operation, only a subset of the OCD commands are available. In DEBUG mode, all OCD commands become available unless the user code and control registers are protected by programming the Flash Read Protect Option bit (FRP). The Flash Read Protect Option bit prevents the code in memory from being read out of the Z8 Encore! XP F082A Series products. When this option is enabled, several of the OCD commands are disabled. Table 106 on page 184 is a summary of the On-chip debugger commands. Each OCD command is described in further detail in the bulleted list following this table. Table 106 on page 184 also indicates those commands that operate when the device is not in DEBUG mode (normal operation) and those commands that are disabled by programming the Flash Read Protect Option bit.

Debug Command	Command Byte	Enabled when NOT in DEBUG mode?	Disabled by Flash Read Protect Option Bit
Read OCD Revision	00H	Yes	–
Reserved	01H	–	–
Read OCD Status Register	02H	Yes	–
Read Runtime Counter	03H	–	–
Write OCD Control Register	04H	Yes	Cannot clear DBGMODE bit
Read OCD Control Register	05H	Yes	–
Write Program Counter	06H	–	Disabled
Read Program Counter	07H	–	Disabled

Table 111. Transconductance Values for Low, Medium, and High Gain Operating Modes

Mode	Crystal Frequency Range	Function	Transconductance (mA/V) Use this range for calculations		
Low Gain*	32 kHz–1 MHz	Low Power/Frequency Applications	0.02	0.04	0.09
Medium Gain*	0.5 MHz–10 MHz	Medium Power/Frequency Applications	0.84	1.7	3.1
High Gain*	8 MHz–20 MHz	High Power/Frequency Applications	1.1	2.3	4.2

Note: *Printed circuit board layout must not add more than 4 pF of stray capacitance to either XIN or XOUT pins. if no Oscillation occurs, reduce the values of the capacitors C1 and C2 to decrease the loading.

Oscillator Operation with an External RC Network

Figure 28 displays a recommended configuration for connection with an external resistor-capacitor (RC) network.

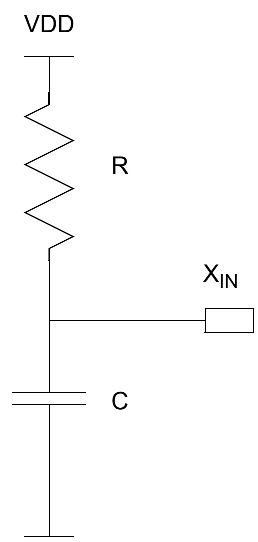

Figure 28. Connecting the On-Chip Oscillator to an External RC Network

An external resistance value of 45 kΩ is recommended for oscillator operation with an external RC network. The minimum resistance value to ensure operation is 40 kΩ. The typical oscillator frequency can be estimated from the values of the resistor (*R* in kΩ) and capacitor (*C* in pF) elements using the following equation:

$$\text{Oscillator Frequency (kHz)} = \frac{1\times 10^6}{(0.4 \times R \times C) + (4 \times C)}$$

Table 117. Bit Manipulation Instructions

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

Table 118. Block Transfer Instructions

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

Table 119. CPU Control Instructions

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

Table 121. Logical Instructions (Continued)

Mnemonic	Operands	Instruction
ORX	dst, src	Logical OR using Extended Addressing
XOR	dst, src	Logical Exclusive OR
XORX	dst, src	Logical Exclusive OR using Extended Addressing

Table 122. Program Control Instructions

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

Table 123. Rotate and Shift Instructions

Mnemonic	Operands	Instruction
BSWAP	dst	Bit Swap
RL	dst	Rotate Left
RLC	dst	Rotate Left through Carry
RR	dst	Rotate Right
RRC	dst	Rotate Right through Carry

Table 124. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Flags Z	Flags S	Flags V	Flags D	Flags H	Fetch Cycles	Instr. Cycles
COM dst	dst ← ~dst	R		60	–	*	*	0	–	–	2	2
		IR		61							2	3
CP dst, src	dst - src	r	r	A2	*	*	*	*	–	–	2	3
		r	Ir	A3							2	4
		R	R	A4							3	3
		R	IR	A5							3	4
		R	IM	A6							3	3
		IR	IM	A7							3	4
CPC dst, src	dst - src - C	r	r	1F A2	*	*	*	*	–	–	3	3
		r	Ir	1F A3							3	4
		R	R	1F A4							4	3
		R	IR	1F A5							4	4
		R	IM	1F A6							4	3
		IR	IM	1F A7							4	4
CPCX dst, src	dst - src - C	ER	ER	1F A8	*	*	*	*	–	–	5	3
		ER	IM	1F A9							5	3
CPX dst, src	dst - src	ER	ER	A8	*	*	*	*	–	–	4	3
		ER	IM	A9							4	3
DA dst	dst ← DA(dst)	R		40	*	*	*	X	–	–	2	2
		IR		41							2	3
DEC dst	dst ← dst - 1	R		30	–	*	*	*	–	–	2	2
		IR		31							2	3
DECW dst	dst ← dst - 1	RR		80	–	*	*	*	–	–	2	5
		IRR		81							2	6
DI	IRQCTL[7] ← 0			8F	–	–	–	–	–	–	1	2
DJNZ dst, RA	dst ← dst – 1 if dst ≠ 0 PC ← PC + X	r		0A-FA	–	–	–	–	–	–	2	3
EI	IRQCTL[7] ← 1			9F	–	–	–	–	–	–	1	2
Flags Notation:	* = Value is a function of the result of the operation. – = Unaffected X = Undefined				0 = Reset to 0 1 = Set to 1							

Table 130. Internal Precision Oscillator Electrical Characteristics

Symbol	Parameter	V_{DD} = 2.7 V to 3.6 V T_A = -40 °C to +105 °C (unless otherwise stated)			Units	Conditions
		Minimum	Typical	Maximum		
F_{IPO}	Internal Precision Oscillator Frequency (High Speed)		5.53		MHz	V_{DD} = 3.3 V T_A = 30 °C
F_{IPO}	Internal Precision Oscillator Frequency (Low Speed)		32.7		kHz	V_{DD} = 3.3 V T_A = 30 °C
F_{IPO}	Internal Precision Oscillator Error		±1	±4	%	
T_{IPOST}	Internal Precision Oscillator Startup Time		3		µs	