



Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

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

Details

Product Status	Active
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	6
Program Memory Size	1KB (1K x 8)
Program Memory Type	FLASH
EEPROM Size	16 x 8
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	8-VDFN Exposed Pad
Supplier Device Package	8-QFN (5x6)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f012aqb020eg

Email: info@E-XFL.COM

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

Z8 Encore! XP[®] F082A Series Product Specification

Figure 25	Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface
1 18410 201	#2 of 2
Figure 26.	OCD Data Format
Figure 27.	Recommended 20MHz Crystal Oscillator Configuration 199
Figure 28.	Connecting the On-Chip Oscillator to an External RC Network 201
Figure 29.	Typical RC Oscillator Frequency as a Function of the External Capacitance with a $45k\Omega$ Resistor
Figure 30.	Opcode Map Cell Description 222
Figure 31.	First Opcode Map
Figure 32.	Second Opcode Map after 1FH 225
Figure 33.	Typical Active Mode IDD Versus System Clock Frequency 231
Figure 34.	Port Input Sample Timing 240
Figure 35.	GPIO Port Output Timing 241
Figure 36.	On-Chip Debugger Timing
Figure 37.	UART Timing With CTS
Figure 38.	UART Timing Without CTS 244

List of Tables

Table 1.	Z8 Encore! XP F082A Series Family Part Selection Guide 2
Table 2.	Signal Descriptions
Table 3.	Pin Characteristics (20- and 28-pin Devices) 13
Table 4.	Pin Characteristics (8-Pin Devices) 14
Table 5.	Z8 Encore! XP F082A Series Program Memory Maps 16
Table 6.	Z8 Encore! XP F082A Series Flash Memory Information Area Map 17
Table 7.	Register File Address Map 18
Table 8.	Reset and Stop Mode Recovery Characteristics and Latency 23
Table 9.	Reset Sources and Resulting Reset Type
Table 10.	Stop Mode Recovery Sources and Resulting Action
Table 11.	Reset Status Register (RSTSTAT)
Table 12.	Reset and Stop Mode Recovery Bit Descriptions
Table 13.	Power Control Register 0 (PWRCTL0)
Table 14.	Port Availability by Device and Package Type
Table 15.	Port Alternate Function Mapping (Non 8-Pin Parts) 40
Table 16.	Port Alternate Function Mapping (8-Pin Parts)
Table 17.	GPIO Port Registers and Subregisters
Table 18.	Port A–D GPIO Address Registers (PxADDR)
Table 19.	Port A–D GPIO Address Registers by Bit Description
Table 20.	Port A–D Control Registers (PxCTL)
Table 21.	Port A–D Data Direction Subregisters (PxDD) 46
Table 22.	Port A–D Alternate Function Subregisters (PxAF)
Table 23.	Port A–D Output Control Subregisters (PxOC)
Table 24.	Port A–D High Drive Enable Subregisters (PxHDE)
Table 25.	Port A–D Stop Mode Recovery Source Enable Subregisters (PxSMRE) 49
Table 26.	Port A–D Pull-Up Enable Subregisters (PxPUE)
Table 27.	Port A–D Alternate Function Set 2 Subregisters (PxAFS2) 51
Table 28.	Port A–D Alternate Function Set 1 Subregisters (PxAFS1) 51

addresses outside the available Flash memory addresses returns FFH. Writing to these unimplemented Program Memory addresses produces no effect. Table 5 describes the Program Memory Maps for the Z8 Encore! XP F082A Series products.

Program Memory Address (Hex)	Function
Z8F082A and Z8F081A 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-1FFF	Program Memory
Z8F042A and Z8F041A 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-0FFF	Program Memory
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

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

Note: *See Table 32 on page 56 for a list of the interrupt vectors.



Figure 6. Voltage Brown-Out Reset Operation

The POR level is greater than the VBO level by the specified hysteresis value. This ensures that the device undergoes a Power-On Reset after recovering from a VBO condition.

Watchdog Timer Reset

If the device is operating in NORMAL or HALT Mode, the Watchdog Timer can initiate a System Reset at time-out if the WDT_RES Flash option bit is programmed to 1, i.e., the unprogrammed state of the WDT_RES Flash option bit. If the bit is programmed to 0, it configures the Watchdog Timer to cause an interrupt, not a System Reset, at time-out.

The WDT bit in the Reset Status (RSTSTAT) Register is set to signify that the reset was initiated by the Watchdog Timer.

External Reset Input

The $\overline{\text{RESET}}$ pin has a Schmitt-Triggered input and an internal pull-up resistor. Once the $\overline{\text{RESET}}$ pin is asserted for a minimum of four system clock cycles, the device progresses through the System Reset sequence. Because of the possible asynchronicity of the system clock and reset signals, the required reset duration may be as short as three clock periods

Shared Debug Pin

On the 8-pin version of this device only, the Debug pin shares function with the PA0 GPIO pin. This pin performs as a general purpose input pin on power-up, but the debug logic monitors this pin during the reset sequence to determine if the unlock sequence occurs. If the unlock sequence is present, the debug function is unlocked and the pin no longer functions as a GPIO pin. If it is not present, the debug feature is disabled until/unless another reset event occurs. For more details, see the <u>On-Chip Debugger</u> chapter on page 180.

Crystal Oscillator Override

For systems using a crystal oscillator, PA0 and PA1 are used to connect the crystal. When the crystal oscillator is enabled, the GPIO settings are overridden and PA0 and PA1 are disabled. See the <u>Oscillator Control Register Definitions section on page 196</u> for details.

5V Tolerance

All six I/O pins on the 8-pin devices are 5V-tolerant, unless the programmable pull-ups are enabled. If the pull-ups are enabled and inputs higher than V_{DD} are applied to these parts, excessive current flows through those pull-up devices and can damage the chip.

Note: In the 20- and 28-pin versions of this device, any pin which shares functionality with an ADC, crystal or comparator port is not 5 V-tolerant, including PA[1:0], PB[5:0] and PC[2:0]. All other signal pins are 5 V-tolerant and can safely handle inputs higher than V_{DD} except when the programmable pull-ups are enabled.

External Clock Setup

For systems using an external TTL drive, PB3 is the clock source for 20- and 28-pin devices. In this case, configure PB3 for alternate function CLKIN. Write the Oscillator Control (OSCCTL) Register such that the external oscillator is selected as the system clock. See the <u>Oscillator Control Register Definitions section on page 196</u> for details. For 8-pin devices, use PA1 instead of PB3.

40

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port A ^{1,2}	PA0	T0IN/T0OUT	Timer 0 Input/Timer 0 Output Complement	N/A
		Reserved		-
	PA1	TOOUT	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		-

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

Notes:

- 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 automatically enables the associated alternate function. See the <u>Port A–D Alternate Function Subregisters (PxAF)</u> section on page 47 for details.
- 2. Whether PA0/PA6 takes on the timer input or timer output complement function depends on the timer configuration. See the <u>Timer Pin Signal Operation</u> section on page 84 for details.
- Because there are at most two choices of alternate function for any pin of Port B, the Alternate Function Set Register AFS2 is not used to select the function. Alternate function selection must also be enabled. See the Port <u>A–D Alternate Function Subregisters (PxAF)</u> section on page 47 for details.
- 4. V_{REF} is available on PB5 in 28-pin products and on PC2 in 20-pin parts.
- Because there are at most two choices of alternate function for any pin of Port C, the Alternate Function Set Register AFS2 is not used to select the function. Alternate function selection must also be enabled. See the Port <u>A–D Alternate Function Subregisters (PxAF)</u> section on page 47 for details.
- Because there is only a single alternate function for the Port PD0 pin, the Alternate Function Set registers are not implemented for Port D. Enabling alternate function selections automatically enables the associated alternate function. See the <u>Port A–D Alternate Function Subregisters (PxAF)</u> section on page 47 for details.

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Select Register AFS1	Alternate Function Select Register AFS2
Port A	PA0	TOIN	Timer 0 Input	AFS1[0]: 0	AFS2[0]: 0
		Reserved		AFS1[0]: 0	AFS2[0]: 1
		Reserved		AFS1[0]: 1	AFS2[0]: 0
		TOOUT	Timer 0 Output Complement	AFS1[0]: 1	AFS2[0]: 1
	PA1	TOOUT	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 ¹	ADC Analog Input/V _{REF}	AFS1[1]: 1	AFS2[1]: 1
	PA2	DE0	UART 0 Driver Enable	AFS1[2]: 0	AFS2[2]: 0
		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	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 ²	ADC Analog Input/LPO Input (P)	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 ²	ADC/Comparator Input (N)/LPO Input (N)	AFS1[4]: 1	AFS2[4]: 1
	PA5	TXD0	UART 0 Transmit Data	AFS1[5]: 0	AFS2[5]: 0
		T1OUT	Timer 1 Output Complement	AFS1[5]: 0	AFS2[5]: 1
		Reserved		AFS1[5]: 1	AFS2[5]: 0
		Analog Functions ²	ADC/Comparator Input (P) LPO Output	AFS1[5]: 1	AFS2[5]: 1

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

Notes:

1. Analog functions include ADC inputs, ADC reference, comparator inputs and LPO ports.

2. The alternate function selection must be enabled; see the Port A–D Alternate Function Subregisters (PxAF) section on page 47 for details.

Priority	Program Memory Vector Address	Interrupt or Trap Source
Highest	0002H	Reset (not an interrupt)
	0004H	Watchdog Timer (see Watchdog Timer)
	003AH	Primary Oscillator Fail Trap (not an interrupt)
	003CH	Watchdog Oscillator Fail Trap (not an interrupt)
	0006H	Illegal Instruction Trap (not an interrupt)
	0008H	Reserved
	000AH	Timer 1
	000CH	Timer 0
	000EH	UART 0 receiver
	0010H	UART 0 transmitter
	0012H	Reserved
	0014H	Reserved
	0016H	ADC
	0018H	Port A Pin 7, selectable rising or falling input edge or LVD (see Reset, Stop Mode Recovery and Low Voltage Detection)
	001AH	Port A Pin 6, selectable rising or falling input edge or Comparator Output
	001CH	Port A Pin 5, selectable rising or falling input edge
	001EH	Port A Pin 4, selectable rising or falling input edge
	0020H	Port A Pin 3, selectable rising or falling input edge
	0022H	Port A Pin 2, selectable rising or falling input edge
	0024H	Port A Pin 1, selectable rising or falling input edge
	0026H	Port A Pin 0, selectable rising or falling input edge
	0028H	Reserved
	002AH	Reserved
	002CH	Reserved
	002EH	Reserved
	0030H	Port C Pin 3, both input edges
	0032H	Port C Pin 2, both input edges
	0034H	Port C Pin 1, both input edges
Lowest	0036H	Port C Pin 0, both input edges
	0038H	Reserved

Table 34. Trap and Interrupt Vectors in Order of Priority

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.

Observe the following steps for configuring a timer for COMPARATOR COUNTER Mode and initiating the count:

- 1. Write to the Timer Control Register to:
 - Disable the timer.
 - Configure the timer for COMPARATOR COUNTER Mode.
 - Select either the rising edge or falling edge of the comparator output signal for the count. This 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 action only affects the first pass in COMPARATOR COUNTER Mode. After the first timer Reload in COMPARATOR COUNTER Mode, counting always begins at the reset value of 0001H. Generally, in COMPARATOR 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. 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.

In COMPARATOR COUNTER Mode, the number of comparator output transitions since the timer start is computed via the following equation:

Comparator Output Transitions = Current Count Value – Start Value

- 6. Read data from the UART Receive Data Register. If operating in MULTIPROCES-SOR (9-bit) Mode, further actions may be required depending on the MULTIPRO-CESSOR Mode bits MPMD[1:0].
- 7. Return to <u>Step 4</u> to receive additional data.

Receiving Data using the Interrupt-Driven Method

The UART Receiver interrupt indicates the availability of new data (and error conditions). Observe the following steps 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! 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.

Receiving IrDA Data

Data received from the infrared transceiver using the IR_RXD signal through the RXD pin is decoded by the infrared endec and passed to the UART. The UART's baud rate clock is used by the infrared endec to generate the demodulated signal (RXD) that drives the UART. Each UART/Infrared data bit is 16-clocks wide. Figure 18 displays data reception. When the infrared endec is enabled, the UART's RXD signal is internal to the Z8 Encore! XP F082A Series products while the IR_RXD signal is received through the RXD pin.



Infrared Data Reception

Caution: The system clock frequency must be at least 1.0MHz to ensure proper reception of the 1.4µs minimum width pulses allowed by the IrDA standard.

Endec Receiver Synchronization

The IrDA receiver uses a local baud rate clock counter (0 to 15 clock periods) to generate an input stream for the UART and to create a sampling window for detection of incoming pulses. The generated UART input (UART RXD) is delayed by 8 baud rate clock periods with respect to the incoming IrDA data stream. When a falling edge in the input data stream is detected, the Endec counter is reset. When the count reaches a value of 8, the UART RXD value is updated to reflect the value of the decoded data. When the count reaches 12 baud clock periods, the sampling window for the next incoming pulse opens.

Analog-to-Digital Converter

The analog-to-digital converter (ADC) converts an analog input signal to its digital representation. The features of this sigma-delta ADC include:

- 11-bit resolution in DIFFERENTIAL Mode
- 10-bit resolution in SINGLE-ENDED Mode
- Eight single-ended analog input sources are multiplexed with general-purpose I/O ports
- 9th analog input obtained from temperature sensor peripheral
- 11 pairs of differential inputs also multiplexed with general-purpose I/O ports
- Low-power operational amplifier (LPO)
- Interrupt on conversion complete
- Bandgap generated internal voltage reference with two selectable levels
- Manual in-circuit calibration is possible employing user code (offset calibration)
- Factory calibrated for in-circuit error compensation

Architecture

Figure 19 displays the major functional blocks of the ADC. An analog multiplexer network selects the ADC input from the available analog pins, ANA0 through ANA7.

The input stage of the ADC allows both differential gain and buffering. The following input options are available:

- Unbuffered input (SINGLE-ENDED and DIFFERENTIAL modes)
- Buffered input with unity gain (SINGLE-ENDED and DIFFERENTIAL modes)
- LPO output with full pin access to the feedback path

148

Figure 22 displays a basic Flash Controller flow. The following subsections provide details about the various operations displayed in Figure 22.



Figure 22. Flash Controller Operation Flow Chart

Caution: The byte at each address of the Flash memory cannot be programmed (any bits written to 0) more than twice before an erase cycle occurs. Doing so may result in corrupted data at the target byte.

Page Erase

The Flash memory can be erased one page (512 bytes) at a time. Page Erasing the Flash memory sets all bytes in that page to the value FFH. The Flash Page Select Register identifies the page to be erased. Only a page residing in an unprotected sector can be erased. With the Flash Controller unlocked and the active page set, writing the value 95h to the Flash Control Register initiates the Page Erase operation. While the Flash Controller executes the Page Erase operation, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate. The eZ8 CPU resumes operation after the Page Erase operation completes. If the Page Erase operation is performed using the On-Chip Debugger, poll the Flash Status Register to determine when the Page Erase operation is complete. When the Page Erase is complete, the Flash Controller returns to its locked state.

Mass Erase

The Flash memory can also be Mass Erased using the Flash Controller, but only by using the On-Chip Debugger. Mass Erasing the Flash memory sets all bytes to the value FFH. With the Flash Controller unlocked and the Mass Erase successfully enabled, writing the value 63H to the Flash Control Register initiates the Mass Erase operation. While the Flash Controller executes the Mass Erase operation, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate. Using the On-Chip Debugger, poll the Flash Status Register to determine when the Mass Erase operation is complete. When the Mass Erase is complete, the Flash Controller returns to its locked state.

Flash Controller Bypass

The Flash Controller can be bypassed and the control signals for the Flash memory brought out to the GPIO pins. Bypassing the Flash Controller allows faster Row Programming algorithms by controlling the Flash programming signals directly.

Row programming is recommended for gang programming applications and large volume customers who do not require in-circuit initial programming of the Flash memory. Page Erase operations are also supported when the Flash Controller is bypassed.

For more information about bypassing the Flash Controller, refer to the <u>Third-Party Flash</u> <u>Programming Support for Z8 Encore! MCUs Application Note (AN0117)</u>, which is available for download on <u>www.zilog.com</u>.

Randomized Lot Identifier

Table 104. Lot Identification Number (RAND_LOT)

Bit	7	6	5	4	3	2	1	0
Field	RAND_LOT							
RESET	U	U	U	U	U	U	U	U
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	s Interspersed throughout Information Page Memory							
Note: $U = l$	te: U = Unchanged by Reset. R/W = Read/Write.							

Bit Description [7] Randomized Lot ID RAND_LOT The randomized lot ID is a 32-byte binary value that changes for each production lot. See Table 105.

Table 105. Randomized Lot ID Locations

Info Page	Memory	
Address	Address	Usage
3C	FE3C	Randomized Lot ID Byte 31 (most significant).
3D	FE3D	Randomized Lot ID Byte 30.
3E	FE3E	Randomized Lot ID Byte 29.
3F	FE3F	Randomized Lot ID Byte 28.
58	FE58	Randomized Lot ID Byte 27.
59	FE59	Randomized Lot ID Byte 26.
5A	FE5A	Randomized Lot ID Byte 25.
5B	FE5B	Randomized Lot ID Byte 24.
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.

Byte Read

To read a byte from the NVDS array, user code must first push the address onto the stack. User code issues a CALL instruction to the address of the byte-read routine (0×1000) . At the return from the sub-routine, the read byte resides in working register R0 and the read status byte resides in working register R1. The contents of the status byte are undefined for 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 nonuniform execution time. A read operation takes between 44 μ s and 489 μ s (assuming a 20MHz 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 μ s 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 nonzero, 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

NVDS read time can vary drastically. This discrepancy is a trade-off for minimizing the frequency of writes that require post-write page erases, as indicated in Table 107. 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, plus 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.

Operation	Minimum Latency	Maximum Latency
Read (16 byte array)	875	9961
Read (64 byte array)	876	8952
Read (128 byte array)	883	7609
Write (16 byte array)	4973	5009
Write (64 byte array)	4971	5013
Write (128 byte array)	4984	5023
Illegal Read	43	43
Illegal Write	31	31

Table 107. NVDS Read Time

If NVDS read performance is critical to your software architecture, you can optimize your code for speed. Try the first suggestion below before attempting the second.

- 1. Periodically refresh all addresses that are used. The optimal use of NVDS in terms of speed is to rotate the writes evenly among all addresses planned to use, bringing all reads closer to the minimum read time. Because the minimum read time is much less than the write time, however, actual speed benefits are not always realized.
- 2. Use as few unique addresses as possible to optimize the impact of refreshing, plus minimize the requirement for it.

205

Assembly Language Source Program Example

JP START	; Everything after the semicolon is a comment.
START:	; A label called 'START'. The first instruction (JP START) in this ; example causes program execution to jump to the point within the ; program where the START label occurs.
LD R4, R7	; A Load (LD) instruction with two operands. The first operand, ; Working Register R4, is the destination. The second operand, ; Working Register R7, is the source. The contents of R7 is ; written into R4.
LD 234H, #%01	; Another Load (LD) instruction with two operands. ; The first operand, Extended Mode Register Address 234H, ; identifies the destination. The second operand, Immediate Data ; value 01H, is the source. The value 01H is written into the ; Register at address 234H.

Assembly Language Syntax

For proper instruction execution, eZ8 CPU assembly language syntax requires that the operands be written as 'destination, source'. After assembly, the object code usually has the operands in the order 'source, destination', but ordering is opcode-dependent. The following instruction examples illustrate the format of some basic assembly instructions and the resulting object code produced by the assembler. This binary format must be followed if manual program coding is preferred or if you intend to implement your own assembler.

Example 1. If the contents of registers 43H and 08H are added and the result is stored in 43H, the assembly syntax and resulting object code is:

Assembly Language Code	ADD	43H,	08H	(ADD dst, src)
Object Code	04	08	43	(OPC src, dst)

Example 2. In general, when an instruction format requires an 8-bit register address, that address can specify any register location in the range 0–255 or, using Escaped Mode Addressing, a Working Register R0–R15. If the contents of Register 43H and Working Register R8 are added and the result is stored in 43H, the assembly syntax and resulting object code is:

Assembly Language Code	ADD	43H,	R8	(ADD dst, src)
Object Code	04	E8	43	(OPC src, dst)

Electrical Characteristics

The data in this chapter represents all known data prior to qualification and characterization of the F082A Series of products, and is therefore subject to change. Additional electrical characteristics may be found in the individual chapters of this document.

Absolute Maximum Ratings

Stresses greater than those listed in Table 130 may cause permanent damage to the device. These ratings are stress ratings only. Operation of the device at any condition outside those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability. For improved reliability, tie unused inputs to one of the supply voltages (V_{DD} or V_{SS}).

Parameter	Minimum	Maximum	Units	Notes
Ambient temperature under bias	-40	+105	°C	
Storage temperature	-65	+150	°C	
Voltage on any pin with respect to V _{SS}	-0.3	+5.5	V	1
	-0.3	+3.9	V	2
Voltage on V_{DD} pin with respect to V_{SS}	-0.3	+3.6	V	
Maximum current on input and/or inactive output pin	-5	+5	μA	
Maximum output current from active output pin	-25	+25	mA	
8-pin Packages Maximum Ratings at 0°C to 70°C				
Total power dissipation		220	mW	
Maximum current into V _{DD} or out of V _{SS}		60	mA	
20-pin Packages Maximum Ratings at 0°C to 70°C				
Total power dissipation		430	mW	

Table	130.	Absolute	Maximum	Ratings
Iabio		/	maximani	runigo

227

Table 130. Absolute Maximum Ratings (Continued)

Parameter	Minimum	Maximum	Units	Notes
Maximum current into V _{DD} or out of V _{SS}		120	mA	
28-pin Packages Maximum Ratings at 0°C to 70°C				
Total power dissipation		450	mW	
Maximum current into V _{DD} or out of V _{SS}		125	mA	

Notes: Operating temperature is specified in DC Characteristics.

This voltage applies to all pins except the following: V_{DD}, AV_{DD}, pins supporting analog input (Port B[5:0], Port C[2:0]) and pins supporting the crystal oscillator (PA0 and PA1). On the 8-pin packages, this applies to all pins but V_{DD}.

2. This voltage applies to pins on the 20-/28-pin packages supporting analog input (Port B[5:0], Port C[2:0]) and pins supporting the crystal oscillator (PA0 and PA1).

DC Characteristics

Table 131 lists the DC characteristics of the Z8 Encore! XP F082A Series products. All voltages are referenced to V_{SS} , the primary system ground.

		T _A = -40°C to +105°C (unless otherwise specified)				
Symbol	Parameter	Minimum	Typical	Maximum	Units	Conditions
V _{DD}	Supply Voltage	2.7	-	3.6	V	
V _{IL1}	Low Level Input Voltage	-0.3	-	0.3*V _{DD}	V	
V _{IH1}	High Level Input Voltage	0.7*V _{DD}	-	5.5	V	For all input pins without analog or oscillator function. For all sig- nal pins on the 8-pin devices. Programmable pull-ups must also be disabled.
V _{IH2}	High Level Input Voltage	0.7*V _{DD}	-	V _{DD} +0.3	V	For those pins with analog or oscillator function (20-/28-pin devices only), or when pro- grammable pull-ups are enabled.
V _{OL1}	Low Level Output Voltage	-	-	0.4	V	$I_{OL} = 2 \text{ mA}; V_{DD} = 3.0 \text{ V}$ High Output Drive disabled.

Table 131. DC Characteristics

Notes:

1. This condition excludes all pins that have on-chip pull-ups, when driven Low.

2. These values are provided for design guidance only and are not tested in production.