



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	Active
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	6
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Through Hole
Package / Case	8-DIP (0.300", 7.62mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f081apb020sg

Operating Modes	198
Crystal Oscillator Operation	198
Oscillator Operation with an External RC Network	201
Internal Precision Oscillator	203
Operation	203
eZ8 CPU Instruction Set	204
Assembly Language Programming Introduction	204
Assembly Language Syntax	205
eZ8 CPU Instruction Notation	206
eZ8 CPU Instruction Classes	207
eZ8 CPU Instruction Summary	212
Opcode Maps	222
Electrical Characteristics	226
Absolute Maximum Ratings	226
DC Characteristics	227
AC Characteristics	232
On-Chip Peripheral AC and DC Electrical Characteristics	233
General Purpose I/O Port Input Data Sample Timing	240
General Purpose I/O Port Output Timing	241
On-Chip Debugger Timing	242
UART Timing	243
Packaging	245
Ordering Information	246
Part Number Suffix Designations	255
Index	256
Customer Support	265

List of Figures

Figure 1.	Z8 Encore! XP F082A Series Block Diagram	3
Figure 2.	Z8F08xA, Z8F04xA, Z8F02xA and Z8F01xA in 8-Pin SOIC, QFN/MLF-S, or PDIP Package	9
Figure 3.	Z8F08xA, Z8F04xA, Z8F02xA and Z8F01xA in 20-Pin SOIC, SSOP or PDIP Package	9
Figure 4.	Z8F08xA, Z8F04xA, Z8F02xA and Z8F01xA in 28-Pin SOIC, SSOP or PDIP Package	9
Figure 5.	Power-On Reset Operation	25
Figure 6.	Voltage Brown-Out Reset Operation	26
Figure 7.	GPIO Port Pin Block Diagram	37
Figure 8.	Interrupt Controller Block Diagram	57
Figure 9.	Timer Block Diagram	71
Figure 10.	UART Block Diagram	100
Figure 11.	UART Asynchronous Data Format without Parity	101
Figure 12.	UART Asynchronous Data Format with Parity	101
Figure 13.	UART Asynchronous MULTIPROCESSOR Mode Data Format	105
Figure 14.	UART Driver Enable Signal Timing (shown with 1 Stop Bit and Parity)	107
Figure 15.	UART Receiver Interrupt Service Routine Flow	109
Figure 16.	Infrared Data Communication System Block Diagram	120
Figure 17.	Infrared Data Transmission	121
Figure 18.	IrDA Data Reception	122
Figure 19.	Analog-to-Digital Converter Block Diagram	125
Figure 20.	Comparator Block Diagram	140
Figure 21.	Flash Memory Arrangement	147
Figure 22.	Flash Controller Operation Flow Chart	148
Figure 23.	On-Chip Debugger Block Diagram	180
Figure 24.	Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface; #1 of 2	181

Figure 25. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface; #2 of 2	182
Figure 26. OCD Data Format	183
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 Ω Resistor	202
Figure 30. Opcode Map Cell Description	222
Figure 31. First Opcode Map	224
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	242
Figure 37. UART Timing With CTS	243
Figure 38. UART Timing Without CTS	244

Table 59.	Watchdog Timer Control Register (WDTCTL)	96
Table 60.	Watchdog Timer Reload Upper Byte Register (WDTU)	97
Table 61.	Watchdog Timer Reload High Byte Register (WDTH)	97
Table 62.	Watchdog Timer Reload Low Byte Register (WDTL)	98
Table 63.	UART Control 0 Register (U0CTL0)	111
Table 64.	UART Control 1 Register (U0CTL1)	112
Table 65.	UART Status 0 Register (U0STAT0)	114
Table 66.	UART Status 1 Register (U0STAT1)	115
Table 67.	UART Transmit Data Register (U0TXD)	116
Table 68.	UART Receive Data Register (U0RXD)	116
Table 69.	UART Address Compare Register (U0ADDR)	117
Table 70.	UART Baud Rate High Byte Register (U0BRH)	117
Table 71.	UART Baud Rate Low Byte Register (U0BRL)	117
Table 72.	UART Baud Rates	118
Table 73.	ADC Control Register 0 (ADCCTL0)	134
Table 74.	ADC Control/Status Register 1 (ADCCTL1)	136
Table 75.	ADC Data High Byte Register (ADCD_H)	137
Table 76.	ADC Data Low Byte Register (ADCD_L)	137
Table 77.	Comparator Control Register (CMP0)	141
Table 78.	Z8 Encore! XP F082A Series Flash Memory Configurations	146
Table 79.	Flash Code Protection Using the Flash Option Bits	150
Table 80.	Flash Status Register (FSTAT)	155
Table 81.	Flash Control Register (FCTL)	155
Table 82.	Flash Page Select Register (FPS)	156
Table 83.	Flash Sector Protect Register (FPROT)	157
Table 84.	Flash Frequency High Byte Register (FFREQH)	158
Table 85.	Flash Frequency Low Byte Register (FFREQL)	158
Table 86.	Trim Bit Address Register (TRMADR)	161
Table 87.	Trim Bit Data Register (TRMDR)	162
Table 88.	Flash Option Bits at Program Memory Address 0000H	162

without initiating an interrupt (if enabled for that pin).

Stop Mode Recovery Using the External $\overline{\text{RESET}}$ Pin

When the Z8 Encore! XP F082A Series device is in STOP Mode and the external $\overline{\text{RESET}}$ pin is driven Low, a system reset occurs. Because of a glitch filter operating on the $\overline{\text{RESET}}$ pin, the Low pulse must be greater than the minimum width specified, or it is ignored. See the [Electrical Characteristics](#) chapter on page 226 for details.

Low Voltage Detection

In addition to the Voltage Brown-Out (VBO) Reset described above, it is also possible to generate an interrupt when the supply voltage drops below a user-selected value. For details about configuring the Low Voltage Detection (LVD) and the threshold levels available, see the [Trim Option Bits at Address 0003H \(TLVD\) Register](#) on page 166. The LVD function is available on the 8-pin product versions only.

When the supply voltage drops below the LVD threshold, the LVD bit of the Reset Status (RSTSTAT) Register is set to one. This bit remains one until the low-voltage condition goes away. Reading or writing this bit does not clear it. The LVD circuit can also generate an interrupt when so enabled, see the [GPIO Mode Interrupt Controller](#) chapter on page 55. The LVD bit is not latched; therefore, enabling the interrupt is the only way to guarantee detection of a transient low voltage event.

The LVD functionality depends on circuitry shared with the VBO block; therefore, disabling the VBO also disables the LVD.

Reset Register Definitions

The following sections define the Reset registers.

Reset Status Register

The read-only Reset Status (RSTSTAT) Register, shown in Table 11, indicates the source of the most recent Reset event, indicates a Stop Mode Recovery event and indicates a Watchdog Timer time-out. Reading this register resets the upper four bits to 0. This register shares its address with the write-only Watchdog Timer Control Register.

Table 12 lists the bit settings for Reset and Stop Mode Recovery events.

Architecture

Figure 7 displays a simplified block diagram of a GPIO port pin. In this figure, the ability to accommodate alternate functions and variable port current drive strength is not displayed.

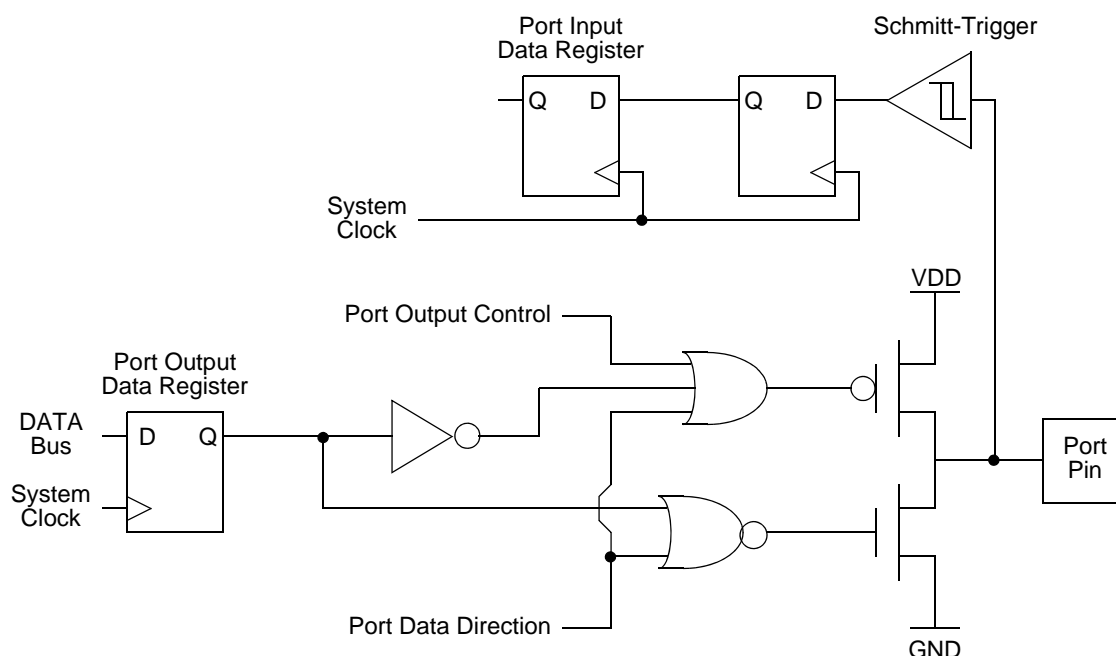


Figure 7. GPIO Port Pin Block Diagram

GPIO Alternate Functions

Many of the GPIO port pins can be used for general-purpose I/O and access to on-chip peripheral functions such as the timers and serial communication devices. The Port A–D Alternate Function subregisters configure these pins for either General-Purpose I/O or alternate function operation. When a pin is configured for alternate function, control of the port pin direction (input/output) is passed from the Port A–D Data Direction registers to the alternate function assigned to this pin. [Table 15](#) on page 40 lists the alternate functions possible with each port pin. For those pins with more one alternate function, the alternate function is defined through Alternate Function Sets subregisters AFS1 and AFS2.

The crystal oscillator functionality is not controlled by the GPIO block. When the crystal oscillator is enabled in the oscillator control block, the GPIO functionality of PA0 and PA1 is overridden. In that case, those pins function as input and output for the crystal oscillator.

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

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port C ⁵	PC0	Reserved		AFS1[0]: 0
		ANA4/CINP	ADC or Comparator Input	AFS1[0]: 1
	PC1	Reserved		AFS1[1]: 0
		ANA5/CINN	ADC or Comparator Input	AFS1[1]: 1
	PC2	Reserved		AFS1[2]: 0
		ANA6/V _{REF} ⁴	ADC Analog Input or ADC Voltage Reference	AFS1[2]: 1
	PC3	COUT	Comparator Output	AFS1[3]: 0
		Reserved		AFS1[3]: 1
	PC4	Reserved		AFS1[4]: 0
				AFS1[4]: 1
	PC5	Reserved		AFS1[5]: 0
				AFS1[5]: 1
	PC6	Reserved		AFS1[6]: 0
				AFS1[6]: 1
	PC7	Reserved		AFS1[7]: 0
				AFS1[7]: 1
Port D ⁶	PD0	RESET	External Reset	N/A

Notes:

1. 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 [Port A–D Alternate Function Subregisters \(PxAF\)](#) 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 [Timer Pin Signal Operation](#) section on page 84 for details.
3. 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 A–D Alternate Function Subregisters \(PxAF\)](#) section on page 47 for details.
4. V_{REF} is available on PB5 in 28-pin products and on PC2 in 20-pin parts.
5. 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 A–D Alternate Function Subregisters \(PxAF\)](#) section on page 47 for details.
6. 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 [Port A–D Alternate Function Subregisters \(PxAF\)](#) section on page 47 for details.

- Writing a 1 to the IRQE bit in the Interrupt Control Register

Interrupts are globally disabled by any of the following actions:

- Execution of a Disable Interrupt (DI) instruction
- eZ8 CPU acknowledgement of an interrupt service request from the interrupt controller
- Writing a 0 to the IRQE bit in the Interrupt Control Register
- Reset
- Execution of a Trap instruction
- Illegal Instruction Trap
- Primary Oscillator Fail Trap
- Watchdog Oscillator Fail Trap

Interrupt Vectors and Priority

The interrupt controller supports three levels of interrupt priority. Level 3 is the highest priority, Level 2 is the second highest priority and Level 1 is the lowest priority. If all of the interrupts are enabled with identical interrupt priority (all as Level 2 interrupts, for example), the interrupt priority is assigned from highest to lowest as specified in [Table 34 on page 56](#). Level 3 interrupts are always assigned higher priority than Level 2 interrupts which, in turn, always are assigned higher priority than Level 1 interrupts. Within each interrupt priority level (Level 1, Level 2, or Level 3), priority is assigned as specified in Table 34, above. Reset, Watchdog Timer interrupt (if enabled), Primary Oscillator Fail Trap, Watchdog Oscillator Fail Trap and Illegal Instruction Trap always have highest (level 3) priority.

Interrupt Assertion

Interrupt sources assert their interrupt requests for only a single system clock period (single pulse). When the interrupt request is acknowledged by the eZ8 CPU, the corresponding bit in the Interrupt Request Register is cleared until the next interrupt occurs. Writing a 0 to the corresponding bit in the Interrupt Request Register likewise clears the interrupt request.

! Caution: Zilog recommends not using a coding style that clears bits in the Interrupt Request registers. All incoming interrupts received between execution of the first LDX command and the final LDX command are lost. See Example 1, which follows.

- Configure the timer for GATED Mode
 - Set the prescale value
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.

Observe the following steps 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

into the Watchdog Timer Reload registers results in a one-second time-out at room temperature and 3.3 V supply voltage. Time-outs other than one second may be obtained by scaling the calibration values up or down as required.

► **Note:** The Watchdog Timer accuracy still degrades as temperature and supply voltage vary. See [Table 137](#) on page 235 for details.

Watchdog Timer Control Register Definitions

This section defines the features of the following Watchdog Timer Control registers.

Watchdog Timer Control Register (WDTCTL): see page 96

Watchdog Timer Reload Upper Byte Register (WDTU): see page 97

Watchdog Timer Reload High Byte Register (WDTH): see page 97

Watchdog Timer Reload Low Byte Register (WDTL): see page 98

Watchdog Timer Control Register

The Watchdog Timer Control (WDTCTL) Register is a write-only control register. Writing the 55H, AAH unlock sequence to the WDTCTL Register address unlocks the three Watchdog Timer Reload Byte registers (WDTU, WDTH and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL Register address produce no effect on the bits in the WDTCTL Register. The locking mechanism prevents spurious writes to the reload registers. This register address is shared with the read-only Reset Status Register.

Table 59. Watchdog Timer Control Register (WDTCTL)

Bit	7	6	5	4	3	2	1	0
Field	WDTUNLK							
RESET	X	X	X	X	X	X	X	X
R/W	W	W	W	W	W	W	W	W
Address	FF0H							
Note: X = Undefined.								

Bit	Description
[7:0]	Watchdog Timer Unlock
WDTUNLK	The software must write the correct unlocking sequence to this register before it is allowed to modify the contents of the Watchdog Timer reload registers.

The third scheme is enabled by setting MPMD[1:0] to 11b and by writing the UART's address into the UART Address Compare Register. This mode is identical to the second scheme, except that there are no interrupts on address bytes. The first data byte of each frame remains accompanied by a NEWFRM assertion.

External Driver Enable

The UART provides a Driver Enable (DE) signal for off-chip bus transceivers. This feature reduces the software overhead associated with using a GPIO pin to control the transceiver when communicating on a multi-transceiver bus, such as RS-485.

Driver Enable is an active High signal that envelopes the entire transmitted data frame including parity and Stop bits as displayed in Figure 14. The Driver Enable signal asserts when a byte is written to the UART Transmit Data Register. The Driver Enable signal asserts at least one UART bit period and no greater than two UART bit periods before the Start bit is transmitted. This allows a setup time to enable the transceiver. The Driver Enable signal deasserts one system clock period after the final Stop bit is transmitted. This one system clock delay allows both time for data to clear the transceiver before disabling it, plus the ability to determine if another character follows the current character. In the event of back to back characters (new data must be written to the Transmit Data Register before the previous character is completely transmitted) the DE signal is not deasserted between characters. The DEPOL bit in the UART Control Register 1 sets the polarity of the Driver Enable signal.

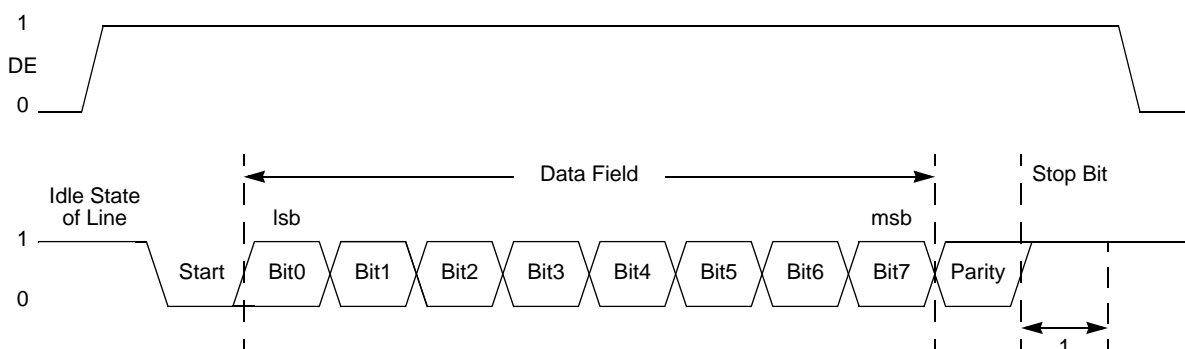


Figure 14. UART Driver Enable Signal Timing (shown with 1 Stop Bit and Parity)

The Driver Enable-to-Start bit setup time is calculated as follows:

$$\left(\frac{1}{\text{Baud Rate (Hz)}} \right) \leq \text{DE to Start Bit Setup Time (s)} \leq \left(\frac{2}{\text{Baud Rate (Hz)}} \right)$$

The baud rate is set by the UART's Baud Rate Generator and supports IrDA standard baud rates from 9600 baud to 115.2 kbaud. Higher baud rates are possible, but do not meet IrDA specifications. The UART must be enabled to use the infrared endec. The infrared endec data rate is calculated using the following equation:

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

Transmitting IrDA Data

The data to be transmitted using the infrared transceiver is first sent to the UART. The UART's transmit signal (TXD) and baud rate clock are used by the IrDA to generate the modulation signal (IR_TXD) that drives the infrared transceiver. Each UART/Infrared data bit is 16 clocks wide. If the data to be transmitted is 1, the IR_TXD signal remains low for the full 16 clock period. If the data to be transmitted is 0, the transmitter first outputs a 7 clock low period, followed by a 3 clock high pulse. Finally, a 6 clock low pulse is output to complete the full 16 clock data period. Figure 17 displays IrDA data transmission. When the infrared endec is enabled, the UART's TXD signal is internal to the Z8 Encore! XP F082A Series products while the IR_TXD signal is output through the TXD pin.

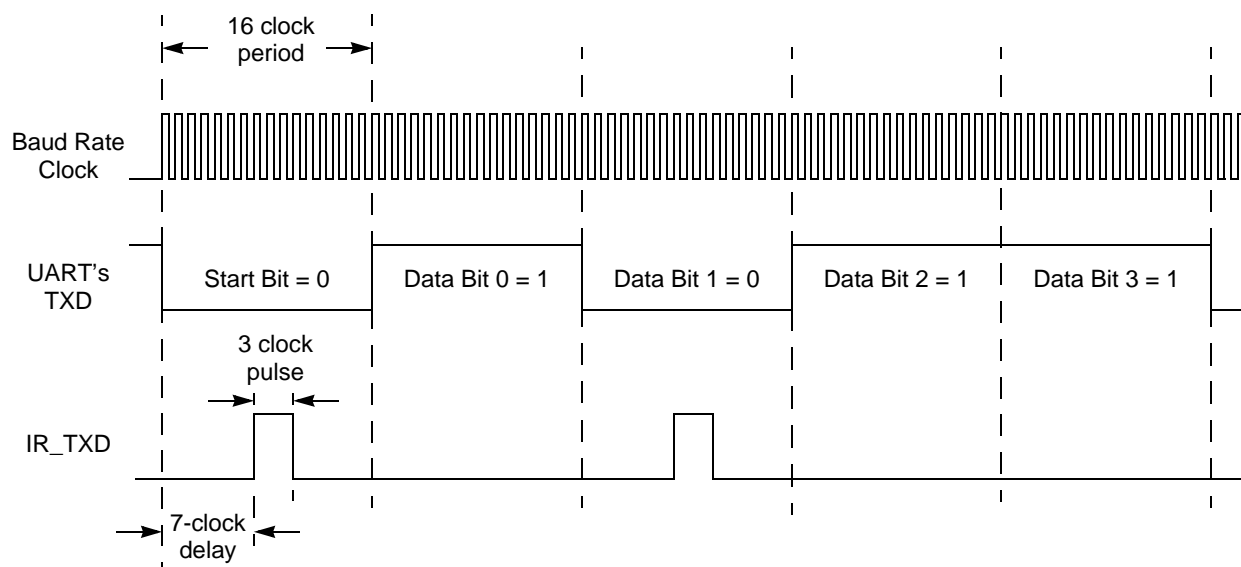


Figure 17. Infrared Data Transmission

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. Observe the following steps for setting up the ADC and initiating a single-shot conversion:

1. Enable the appropriate 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, plus 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.

Flash Memory

The products in the Z8 Encore! XP F082A Series feature a nonvolatile Flash memory of 8 KB (8192), 4 KB (4096), 2 KB (2048 bytes), or 1 KB (1024) with read/write/erase capability. The Flash Memory can be programmed and erased in-circuit by user code or through the On-Chip Debugger. The features include:

- User controlled read and write protect capability
- Sector-based write protection scheme
- Additional protection schemes against accidental program and erasure

Architecture

The Flash memory array is arranged in pages with 512 bytes per page. The 512-byte page is the minimum Flash block size that can be erased. Each page is divided into 8 rows of 64 bytes.

For program or data protection, the Flash memory is also divided into sectors. In the Z8 Encore! XP F082A Series, these sectors are either 1024 bytes (in the 8KB devices) or 512 bytes (all other memory sizes) in size. Page and sector sizes are not generally equal.

The first 2 bytes of Flash Program memory are used as Flash option bits. For more information about their operation, see the [Flash Option Bits](#) chapter on page 159.

Table 78 describes the Flash memory configuration for each device in the Z8 Encore! XP F082A Series. Figure 21 displays the Flash memory arrangement.

Table 78. Z8 Encore! XP F082A Series Flash Memory Configurations

Part Number	Flash Size KB (Bytes)	Flash Pages	Program Memory Addresses	Flash Sector Size (Bytes)
Z8F08xA	8 (8192)	16	0000H–1FFFFH	1024
Z8F04xA	4 (4096)	8	0000H–0FFFFH	512
Z8F02xA	2 (2048)	4	0000H–07FFFH	512
Z8F01xA	1 (1024)	2	0000H–03FFFH	512

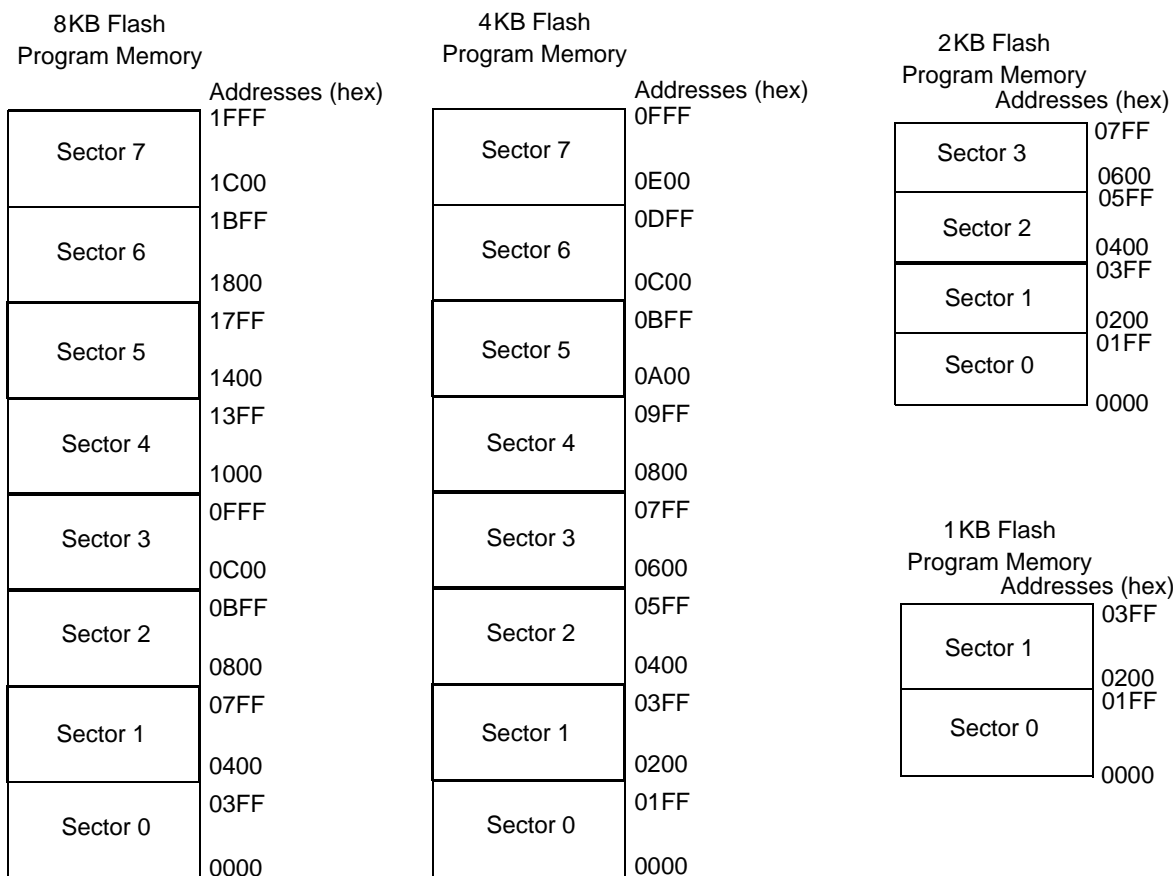


Figure 21. Flash Memory Arrangement

Flash Information Area

The Flash information area is separate from Program Memory and is mapped to the address range FE00H to FFFFH. This area is readable but cannot be erased or overwritten. Factory trim values for the analog peripherals are stored here. Factory calibration data for the ADC is also stored here.

Operation

The Flash Controller programs and erases Flash memory. The Flash Controller provides the proper Flash controls and timing for Byte Programming, Page Erase and Mass Erase of Flash memory.

The Flash Controller contains several protection mechanisms to prevent accidental programming or erasure. These mechanism operate on the page, sector and full-memory levels.

Flash Sector Protect Register

The Flash Sector Protect (FPROT) Register is shared with the Flash Page Select Register. When the Flash Control Register is written with 5EH, the next write to this address targets the Flash Sector Protect Register. In all other cases, it targets the Flash Page Select Register.

This register selects one of the 8 available Flash memory sectors to be protected. The reset state of each Sector Protect bit is an unprotected state. After a sector is protected by setting its corresponding register bit, it cannot be unprotected (the register bit cannot be cleared) without powering down the device.

Table 83. Flash Sector Protect Register (FPROT)

Bit	7	6	5	4	3	2	1	0
Field	SPROT7	SPROT6	SPROT5	SPROT4	SPROT3	SPROT2	SPROT1	SPROT0
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
Address	FF9H							

Bit	Description
[7:0]	Sector Protection
SPROT _n	Each bit corresponds to a 1024-byte Flash sector on devices in the 8K range, while the remaining devices correspond to a 512-byte Flash sector. To determine the appropriate Flash memory sector address range and sector number for your Z8F082A Series product, please refer to Table 78 on page 146 and to Figure 21, which follows the table. <ul style="list-style-type: none"> For Z8F08xA and Z8F04xA devices, all bits are used. For Z8F02xA devices, the upper 4 bits are unused. For Z8F01xA devices, the upper 6 bits are unused.

Flash Frequency High and Low Byte Registers

The Flash Frequency High (FFREQH) and Low Byte (FFREQL) 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) and is calculated using the following equation:

$$\text{FFREQ}[15:0] = \{\text{FFREQH}[7:0], \text{FFREQL}[7:0]\} = \frac{\text{System Clock Frequency}}{1000}$$

Read Register (09H). The Read Register command reads data from the Register File. Data can be read 1–256 bytes at a time (256 bytes can be read by setting size to 0). If the device is not in DEBUG Mode or if the Flash Read Protect Option bit is enabled, this command returns FFH for all the data values.

```
DBG ← 09H
DBG ← {4'h0, Register Address[11:8]}
DBG ← Register Address[7:0]
DBG ← Size[7:0]
DBG → 1-256 data bytes
```

Write Program Memory (0AH). The Write Program Memory command writes data to Program Memory. This command is equivalent to the LDC and LDCI instructions. Data can be written 1–65536 bytes at a time (65536 bytes can be written by setting size to 0). The on-chip Flash Controller must be written to and unlocked for the programming operation to occur. If the Flash Controller is not unlocked, the data is discarded. If the device is not in DEBUG Mode or if the Flash Read Protect Option bit is enabled, the data is discarded.

```
DBG ← 0AH
DBG ← Program Memory Address[15:8]
DBG ← Program Memory Address[7:0]
DBG ← Size[15:8]
DBG ← Size[7:0]
DBG ← 1-65536 data bytes
```

Read Program Memory (0BH). The Read Program Memory command reads data from Program Memory. This command is equivalent to the LDC and LDCI instructions. Data can be read 1–65536 bytes at a time (65536 bytes can be read by setting size to 0). If the device is not in DEBUG Mode or if the Flash Read Protect Option bit is enabled, this command returns FFH for the data.

```
DBG ← 0BH
DBG ← Program Memory Address[15:8]
DBG ← Program Memory Address[7:0]
DBG ← Size[15:8]
DBG ← Size[7:0]
DBG → 1-65536 data bytes
```

Write Data Memory (0CH). The Write Data Memory command writes data to Data Memory. This command is equivalent to the LDE and LDEI instructions. Data can be written 1–65536 bytes at a time (65536 bytes can be written by setting size to 0). If the device is not in DEBUG Mode or if the Flash Read Protect Option bit is enabled, the data is discarded.

```
DBG ← 0CH
DBG ← Data Memory Address[15:8]
DBG ← Data Memory Address[7:0]
```

Register file size varies depending on the device type. See the device-specific Z8 Encore! XP Product Specification to determine the exact register file range available.

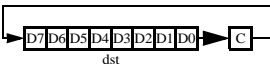
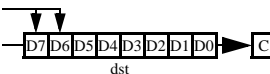
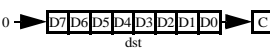
eZ8 CPU Instruction Notation

In the eZ8 CPU Instruction Summary and Description sections, the operands, condition codes, status flags and address modes are represented by a notational shorthand that is described in Table 118.

Table 118. Notational Shorthand

Notation	Description	Operand	Range
b	Bit	b	b represents a value from 0 to 7 (000B to 111B).
cc	Condition code	—	Refer to the Condition Codes section in the eZ8 CPU Core User Manual (UM0128) .
DA	Direct address	Addr	Represents a number in the range 0000H to FFFFH.
ER	Extended addressing register	Reg	Reg. represents a number in the range of 000H to FFFH.
IM	Immediate data	#Data	Data is a number between 00H to FFH.
Ir	Indirect working register	@Rn	n = 0–15.
IR	Indirect register	@Reg	Reg. represents a number in the range of 00H to FFH.
Irr	Indirect working register pair	@RRp	p = 0, 2, 4, 6, 8, 10, 12, or 14.
IRR	Indirect register pair	@Reg	Reg. represents an even number in the range 00H to FEH.
p	Polarity	p	Polarity is a single bit binary value of either 0B or 1B.
r	Working register	Rn	n = 0 – 15.
R	Register	Reg	Reg. represents a number in the range of 00H to FFH.
RA	Relative address	X	X represents an index in the range of +127 to –128 which is an offset relative to the address of the next instruction.
rr	Working register pair	RRp	p = 0, 2, 4, 6, 8, 10, 12, or 14.
RR	Register pair	Reg	Reg. represents an even number in the range of 00H to FEH.

Table 128. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycle s	Instr. Cycle s
		dst	src		C	Z	S	V	D	H		
RRC dst		R		C0	*	*	*	*	–	–	2	2
		IR		C1							2	3
SBC dst, src	$dst \leftarrow dst - src - C$	r	r	32	*	*	*	*	1	*	2	3
		r	lr	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	$dst \leftarrow dst - src - C$	ER	ER	38	*	*	*	*	1	*	4	3
		ER	IM	39							4	3
SCF	$C \leftarrow 1$			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	$RP \leftarrow src$		IM	01	–	–	–	–	–	–	2	2
STOP	STOP Mode			6F	–	–	–	–	–	–	1	2

Note: Flags Notation:

* = Value is a function of the result of the operation.

– = Unaffected.

X = Undefined.

0 = Reset to 0.

1 = Set to 1.

On-Chip Debugger Timing

Figure 36 and Table 145 provide timing information for the DBG pin. The DBG pin timing specifications assume a 4 ns maximum rise and fall time.

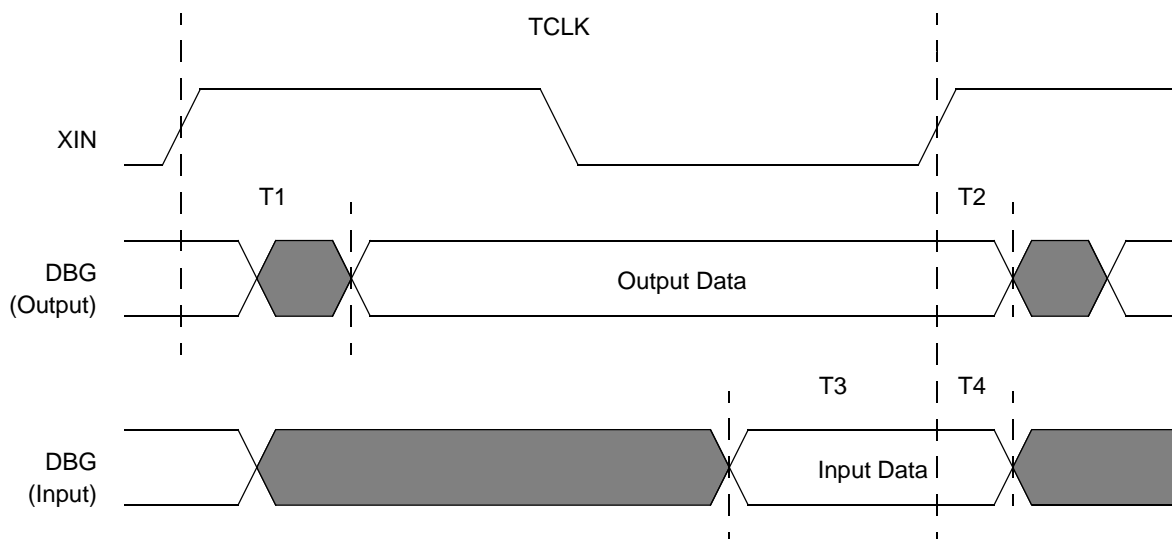


Figure 36. On-Chip Debugger Timing

Table 145. On-Chip Debugger Timing

Parameter	Abbreviation	Delay (ns)	
		Minimum	Maximum
DBG			
T ₁	X _{IN} Rise to DBG Valid Delay	–	15
T ₂	X _{IN} Rise to DBG Output Hold Time	2	–
T ₃	DBG to XIN Rise Input Setup Time	5	–
T ₄	DBG to XIN Rise Input Hold Time	5	–