



Welcome to [E-XFL.COM](https://www.e-xfl.com)

What is "[Embedded - Microcontrollers](#)"?

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

Applications of "[Embedded - Microcontrollers](#)"

Details

Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	46
Program Memory Size	16KB (16K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 12x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f1622ar020sc



Master Interrupt Enable	69
Interrupt Vectors and Priority	70
Interrupt Assertion	70
Software Interrupt Assertion	70
Interrupt Control Register Definitions	71
Interrupt Request 0 Register	71
Interrupt Request 1 Register	72
Interrupt Request 2 Register	73
IRQ0 Enable High and Low Bit Registers	74
IRQ1 Enable High and Low Bit Registers	75
IRQ2 Enable High and Low Bit Registers	76
Interrupt Edge Select Register	78
Interrupt Port Select Register	78
Interrupt Control Register	79
Timers	81
Overview	81
Architecture	81
Operation	82
Timer Operating Modes	82
Reading the Timer Count Values	90
Timer Output Signal Operation	90
Timer Control Register Definitions	90
Timer 0-3 High and Low Byte Registers	90
Timer Reload High and Low Byte Registers	91
Timer 0-3 PWM High and Low Byte Registers	92
Timer 0-3 Control 0 Registers	93
Timer 0-3 Control 1 Registers	94
Watchdog Timer	97
Overview	97
Operation	97
Watchdog Timer Refresh	98
Watchdog Timer Time-Out Response	98
Watchdog Timer Reload Unlock Sequence	99
Watchdog Timer Control Register Definitions	100
Watchdog Timer Control Register	100
Watchdog Timer Reload Upper, High and Low Byte Registers	101

Table 4. Pin Characteristics of the Z8 Encore! XP 64K Series Flash Microcontrollers

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tri-State Output	Internal Pull-up or Pull-down	Schmitt- Trigger Input	Open Drain Output
AVSS	N/A	N/A	N/A	N/A	No	No	N/A
AVDD	N/A	N/A	N/A	N/A	No	No	N/A
DBG	I/O	I	N/A	Yes	No	Yes	Yes
VSS	N/A	N/A	N/A	N/A	No	No	N/A
PA[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PB[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PC[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PD[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PE[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PF[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PG[7:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
PH[3:0]	I/O	I	N/A	Yes	No	Yes	Yes, Programmable
RESET	I	I	Low	N/A	Pull-up	Yes	N/A
VDD	N/A	N/A	N/A	N/A	No	No	N/A
XIN	I	I	N/A	N/A	No	No	N/A
XOUT	O	O	N/A	Yes, in STOP mode	No	No	No

Note: x represents integer 0, 1,... to indicate multiple pins with symbol mnemonics that differ only by the integer.

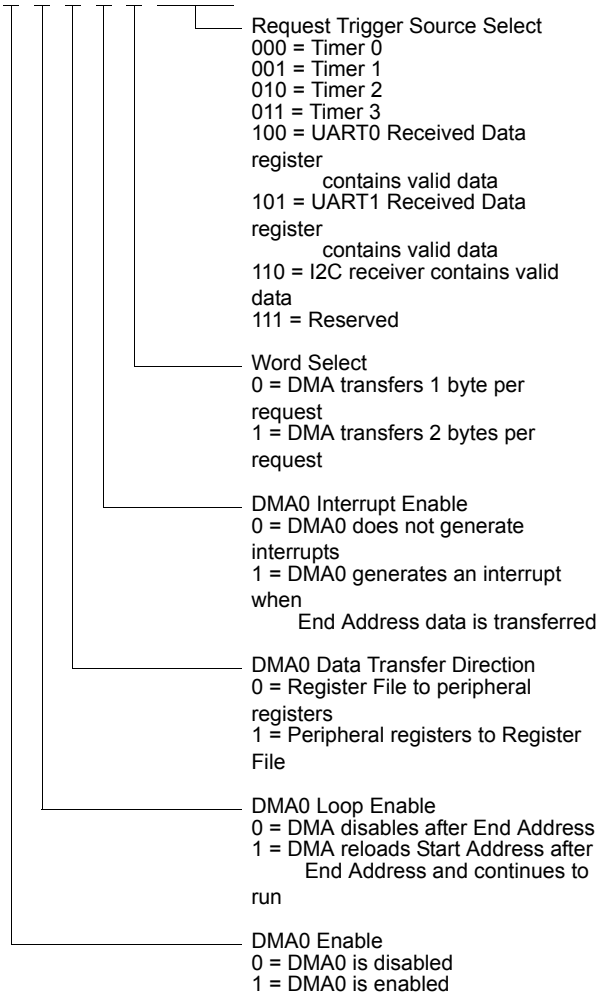
Table 6. Z8 Encore! XP 64K Series Flash Microcontrollers Information Area Map

Program Memory Address (Hex)	Function
FE00H-FE3FH	Reserved
FE40H-FE53H	Part Number 20-character ASCII alphanumeric code Left justified and filled with zeros (ASCII Null character)
FE54H-FFFFH	Reserved

DMA0 Control

DMA0CTL (FB0H - Read/Write)

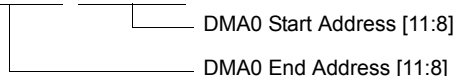
D7 D6 D5 D4 D3 D2 D1 D0



DMA0 Address High Nibble

DMA0H (FB2H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



DMA0 Start/Current Address Low Byte

DMA0START (FB3H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



DMA0 End Address Low Byte

DMA0END (FB4H - Read/Write)

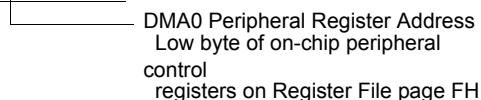
D7 D6 D5 D4 D3 D2 D1 D0



DMA0 I/O Address

DMA0IO (FB1H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



On-Chip Debugger Initiated Reset

A Power-On Reset can be initiated using the On-Chip Debugger by setting the RST bit in the OCD Control register. The On-Chip Debugger block is not reset but the rest of the chip goes through a normal system reset. The RST bit automatically clears during the system reset. Following the system reset the POR bit in the WDT Control register is set.

Stop Mode Recovery

STOP mode is entered by the eZ8 executing a STOP instruction. For detailed STOP mode information, see [Low-Power Modes](#) on page 47. During Stop Mode Recovery, the devices are held in reset for 66 cycles of the Watchdog Timer oscillator followed by 16 cycles of the system clock. Stop Mode Recovery only affects the contents of the Watchdog Timer Control register. Stop Mode Recovery does not affect any other values in the Register File, including the Stack Pointer, Register Pointer, Flags, peripheral control registers, and general-purpose RAM.

The eZ8[™] CPU fetches the Reset vector at Program Memory addresses 0002H and 0003H and loads that value into the Program Counter. Program execution begins at the Reset vector address. Following Stop Mode Recovery, the STOP bit in the Watchdog Timer Control Register is set to 1. [Table 10](#) lists the Stop Mode Recovery sources and resulting actions.

Table 10. Stop Mode Recovery Sources and Resulting Action

Operating Mode	Stop Mode Recovery Source	Action
STOP mode	Watchdog Timer time-out when configured for Reset	Stop Mode Recovery
	Watchdog Timer time-out when configured for interrupt	Stop Mode Recovery followed by interrupt (if interrupts are enabled)
	Data transition on any GPIO Port pin enabled as a Stop Mode Recovery source	Stop Mode Recovery

Stop Mode Recovery Using Watchdog Timer Time-Out

If the Watchdog Timer times out during STOP mode, the device undergoes a Stop Mode Recovery sequence. In the Watchdog Timer Control register, the WDT and STOP bits are set to 1. If the Watchdog Timer is configured to generate an interrupt upon time-out and the 64K Series devices are configured to respond to interrupts, the eZ8 CPU services the Watchdog Timer interrupt request following the normal Stop Mode Recovery sequence.

Stop Mode Recovery Using a GPIO Port Pin Transition HALT

Each of the GPIO Port pins may be configured as a Stop Mode Recovery input source. On any GPIO pin enabled as a Stop Mode Recovery source, a change in the input pin value (from High to Low or from Low to High) initiates Stop Mode Recovery. The GPIO Stop Mode Recovery signals are filtered to reject pulses less than 10 ns (typical) in duration. In the Watchdog Timer Control register, the STOP bit is set to 1.



Caution: *In STOP mode, the GPIO Port Input Data registers (PxIN) are disabled. The Port Input Data registers record the Port transition only if the signal stays on the Port pin through the end of the Stop Mode Recovery delay. Thus, short pulses on the Port pin can initiate Stop Mode Recovery without being written to the Port Input Data register or without initiating an interrupt (if enabled for that pin).*

Port A–H Input Data Registers

Reading from the Port A–H Input Data registers ([Table 21](#)) returns the sampled values from the corresponding port pins. The Port A–H Input Data registers are Read-only.

Table 21. Port A–H Input Data Registers (PxIN)

BITS	7	6	5	4	3	2	1	0
FIELD	PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0
RESET	X							
R/W	R							
ADDR	FD2H, FD6H, FDAH, FDEH, FE2H, FE6H, FEAH, FEEH							

PIN[7:0]—Port Input Data

Sampled data from the corresponding port pin input.

0 = Input data is logical 0 (Low).

1 = Input data is logical 1 (High).

Port A–H Output Data Register

The Port A–H Output Data register ([Table 22](#)) writes output data to the pins.

Table 22. Port A–H Output Data Register (PxOUT)

BITS	7	6	5	4	3	2	1	0
FIELD	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
RESET	0							
R/W	R/W							
ADDR	FD3H, FD7H, FDBH, FDFH, FE3H, FE7H, FEBH, FEFH							

POUT[7:0]—Port Output Data

These bits contain the data to be driven out from the port pins. The values are only driven if the corresponding pin is configured as an output and the pin is not configured for alternate function operation.

0 = Drive a logical 0 (Low).

1 = Drive a logical 1 (High). High value is not driven if the drain has been disabled by setting the corresponding Port Output Control register bit to 1.

Timers

Overview

The 64K Series products contain up to four 16-bit reloadable timers that can be used for timing, event counting, or generation of pulse width modulated signals. The timers' features include:

- 16-bit reload counter
- Programmable prescaler with prescale values from 1 to 128
- PWM output generation
- Capture and compare capability
- External input pin for timer input, clock gating, or capture signal. External input pin signal frequency is limited to a maximum of one-fourth the system clock frequency.
- Timer output pin
- Timer interrupt

In addition to the timers described in this chapter, the Baud Rate Generators for any unused UART, SPI, or I²C peripherals may also be used to provide basic timing functionality. For information on using the Baud Rate Generators as timers, see the respective serial communication peripheral. Timer 3 is unavailable in the 44-pin package devices.

Architecture

Figure 12 displays the architecture of the timers.

6. Write to the Timer Control 1 register to enable the timer.
7. Counting begins on the first appropriate transition of the Timer Input signal. No interrupt is generated by this first edge.

In m/COMPARE mode, the elapsed time from timer start to Capture event can be calculated using the following equation:

$$\text{Capture Elapsed Time (s)} = \frac{(\text{Capture Value} - \text{Start Value}) \times \text{Prescale}}{\text{System Clock Frequency (Hz)}}$$

Reading the Timer Count Values

The current count value in the timers can be read while counting (enabled). This capability has no effect on timer operation. When the timer is enabled and the Timer High Byte register is read, the contents of the Timer Low Byte register are placed in a holding register. A subsequent read from the Timer Low Byte register returns the value in the holding register. This operation allows accurate reads of the full 16-bit timer count value while enabled. When the timers are not enabled, a read from the Timer Low Byte register returns the actual value in the counter.

Timer Output Signal Operation

Timer Output is a GPIO Port pin alternate function. Generally, the Timer Output is toggled every time the counter is reloaded.

Timer Control Register Definitions

Timers 0-2 are available in all packages. Timer 3 is only available in the 64-, 68-, and 80-pin packages.

Timer 0-3 High and Low Byte Registers

The Timer 0-3 High and Low Byte (TxH and TxL) registers (see [Table 39](#) and [Table 40](#) on page 91) contain the current 16-bit timer count value. When the timer is enabled, a read from TxH causes the value in TxL to be stored in a temporary holding register. A read from TMRL always returns this temporary register when the timers are enabled. When the timer is disabled, reads from the TMRL reads the register directly.

Writing to the Timer High and Low Byte registers while the timer is enabled is not recommended. There are no temporary holding registers available for write operations, so simultaneous 16-bit writes are not possible. If either the Timer High or Low Byte registers are written during counting, the 8-bit written value is placed in the counter (High or Low Byte) at the next clock edge. The counter continues counting from the new value.

Timer 3 is unavailable in the 40- and 44-pin packages.



Table 61. UART Baud Rates (Continued)

9.60	23	9.73	1.32	9.60	12	9.60	0.00
4.80	47	4.76	-0.83	4.80	24	4.80	0.00
2.40	93	2.41	0.23	2.40	48	2.40	0.00
1.20	186	1.20	0.23	1.20	96	1.20	0.00
0.60	373	0.60	-0.04	0.60	192	0.60	0.00
0.30	746	0.30	-0.04	0.30	384	0.30	0.00

0 = \overline{SS} input pin is asserted (Low).
1 = \overline{SS} input is not asserted (High).
If SPI enabled as a Master, this bit is not applicable.

SPI Mode Register

The SPI Mode register (Table 66) configures the character bit width and the direction and value of the \overline{SS} pin.

Table 66. SPI Mode Register (SPIMODE)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved		DIAG	NUMBITS[2:0]			SSIO	SSV
RESET	0							
R/W	R		R/W					
ADDR	F63H							

Reserved—Must be 0.

DIAG—Diagnostic Mode Control bit

This bit is for SPI diagnostics. Setting this bit allows the Baud Rate Generator value to be read using the SPIBRH and SPIBRL register locations.

0 = Reading SPIBRH, SPIBRL returns the value in the SPIBRH and SPIBRL registers

1 = Reading SPIBRH returns bits [15:8] of the SPI Baud Rate Generator; and reading SPIBRL returns bits [7:0] of the SPI Baud Rate Counter. The Baud Rate Counter High and Low byte values are not buffered.



Caution: Exercise caution if reading the values while the BRG is counting.

NUMBITS[2:0]—Number of Data Bits Per Character to Transfer

This field contains the number of bits to shift for each character transfer. For information on valid bit positions when the character length is less than 8-bits, see SPI Data Register description.

000 = 8 bits

001 = 1 bit

010 = 2 bits

011 = 3 bits

100 = 4 bits

101 = 5 bits

110 = 6 bits

111 = 7 bits

7. The I²C Controller loads the I²C Shift register with the contents of the I²C Data register.
8. After one bit of address is shifted out by the SDA signal, the Transmit interrupt is asserted.
9. Software responds by writing the second byte of address into the contents of the I²C Data register.
10. The I²C Controller shifts the rest of the first byte of address and write bit out the SDA signal.
11. If the I²C slave sends an acknowledge by pulling the SDA signal low during the next high period of SCL the I²C Controller sets the ACK bit in the I²C Status register. Continue with [step 12](#).

If the slave does not acknowledge the first address byte, the I²C Controller sets the NCKI bit and clears the ACK bit in the I²C Status register. Software responds to the Not Acknowledge interrupt by setting the STOP and FLUSH bits and clearing the TXI bit. The I²C Controller sends the STOP condition on the bus and clears the STOP and NCKI bits. The transaction is complete (ignore following steps).

12. The I²C Controller loads the I²C Shift register with the contents of the I²C Data register (2nd byte of address).
13. The I²C Controller shifts the second address byte out the SDA signal. After the first bit has been sent, the Transmit interrupt is asserted.
14. Software responds by setting the STOP bit in the I²C Control register. The TXI bit can be cleared at the same time.
15. Software polls the STOP bit of the I²C Control register. Hardware deasserts the STOP bit when the transaction is completed (STOP condition has been sent).
16. Software checks the ACK bit of the I²C Status register. If the slave acknowledged, the ACK bit is = 1. If the slave does not acknowledge, the ACK bit is = 0. The NCKI interrupt do not occur because the STOP bit was set.

Write Transaction with a 10-Bit Address

[Figure 31](#) displays the data transfer format for a 10-bit addressed slave. Shaded regions indicate data transferred from the I²C Controller to slaves and unshaded regions indicate data transferred from the slaves to the I²C Controller.

S	Slave Address 1st 7 bits	W = 0	A	Slave Address 2nd Byte	A	Data	A	Data	A/A	P/S
---	-----------------------------	-------	---	---------------------------	---	------	---	------	-----	-----

Figure 31. 10-Bit Addressed Slave Data Transfer Format

4. The I²C Controller sends the START condition.
5. The I²C Controller shifts the address and read bit out the SDA signal.
6. If the I²C slave acknowledges the address by pulling the SDA signal Low during the next high period of SCL, the I²C Controller sets the ACK bit in the I²C Status register. Continue with [step 7](#).

If the slave does not acknowledge, the Not Acknowledge interrupt occurs (NCKI bit is set in the Status register, ACK bit is cleared). Software responds to the Not Acknowledge interrupt by setting the STOP bit and clearing the TXI bit. The I²C Controller sends the STOP condition on the bus and clears the STOP and NCKI bits. The transaction is complete (ignore the following steps).

7. The I²C Controller shifts in the byte of data from the I²C slave on the SDA signal. The I²C Controller sends a Not Acknowledge to the I²C slave if the NAK bit is set (last byte), else it sends an Acknowledge.
8. The I²C Controller asserts the Receive interrupt (RDRF bit set in the Status register).
9. Software responds by reading the I²C Data register which clears the RDRF bit. If there is only one more byte to receive, set the NAK bit of the I²C Control register.
10. If there are more bytes to transfer, return to [step 7](#).
11. After the last byte is shifted in, a Not Acknowledge interrupt is generated by the I²C Controller.
12. Software responds by setting the STOP bit of the I²C Control register.
13. A STOP condition is sent to the I²C slave, the STOP and NCKI bits are cleared.

Read Transaction with a 10-Bit Address

[Figure 33](#) displays the read transaction format for a 10-bit addressed slave. The shaded regions indicate data transferred from the I²C Controller to slaves and unshaded regions indicate data transferred from the slaves to the I²C Controller.

S	Slave Address 1st 7 bits	W=0	A	Slave Address 2nd Byte	A	S	Slave Address 1st 7 bits	R=1	A	Data	A	Data	\bar{A}	P
---	-----------------------------	-----	---	---------------------------	---	---	-----------------------------	-----	---	------	---	------	-----------	---

Figure 33. Receive Data Format for a 10-Bit Addressed Slave

The first seven bits transmitted in the first byte are 11110xx. The two bits xx are the two most-significant bits of the 10-bit address. The lowest bit of the first byte transferred is the write control bit.

DMAx Start/Current Address Low Byte Register

The DMAx Start/Current Address Low register, in conjunction with the DMAx Address High Nibble register, forms a 12-bit Start/Current Address. Writes to this register set the Start Address for DMA operations. Each time the DMA completes a data transfer, the 12-bit Start/Current Address increments by either 1 (single-byte transfer) or 2 (two-byte word transfer). Reads from this register return the low byte of the Current Address to be used for the next DMA data transfer.

Table 80. DMAx Start/Current Address Low Byte Register (DMAxSTART)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_START							
RESET	X							
R/W	R/W							
ADDR	FB3H, FBBH							

DMA_START—DMAx Start/Current Address Low

These bits, with the four lower bits of the DMAx_H register, form the 12-bit Start/Current address. The full 12-bit address is given by {DMA_START_H[3:0], DMA_START[7:0]}.

DMAx End Address Low Byte Register

The DMAx End Address Low Byte register ([Table 80](#)), in conjunction with the DMAx_H register ([Table 81](#)), forms a 12-bit End Address.

Table 81. DMAx End Address Low Byte Register (DMAxEND)

BITS	7	6	5	4	3	2	1	0
FIELD	DMA_END							
RESET	X							
R/W	R/W							
ADDR	FB4H, FBCH							

DMA_END—DMAx End Address Low

These bits, with the four upper bits of the DMAx_H register, form a 12-bit address. This address is the ending location of the DMAx transfer. The full 12-bit address is given by {DMA_END_H[3:0], DMA_END[7:0]}.

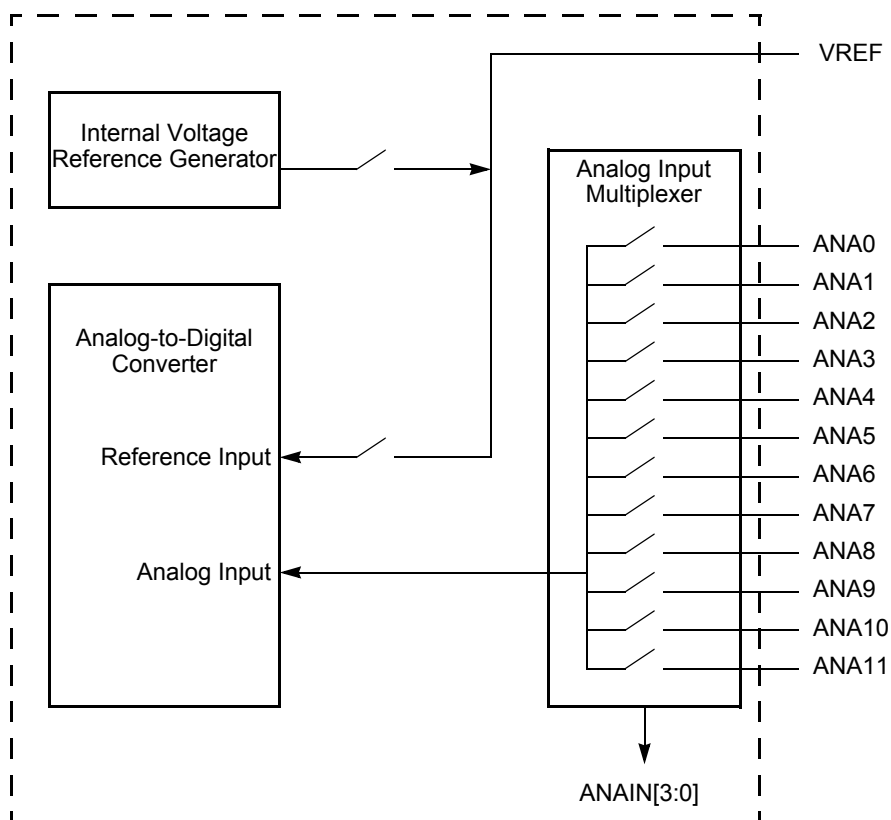


Figure 34. Analog-to-Digital Converter Block Diagram

The sigma-delta ADC architecture provides alias and image attenuation below the amplitude resolution of the ADC in the frequency range of DC to one-half the ADC clock rate (one-fourth the system clock rate). The ADC provides alias free conversion for frequencies up to one-half the ADC clock rate. Thus the sigma-delta ADC exhibits high noise immunity making it ideal for embedded applications. In addition, monotonicity (no missing codes) is guaranteed by design.

Operation

Automatic Power-Down

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 power-down state, the ADC requires 40 system clock cycles to power-up. The ADC powers up when a conversion is requested using the ADC Control register.

Table 101. On-Chip Debugger Commands (Continued)

Debug Command	Command Byte	Enabled when NOT in DEBUG mode?	Disabled by Read Protect Option Bit
Write Program Counter	06H	-	Disabled
Read Program Counter	07H	-	Disabled
Write Register	08H	-	Only writes of the Flash Memory Control registers are allowed. Additionally, only the Mass Erase command is allowed to be written to the Flash Control register.
Read Register	09H	-	Disabled
Write Program Memory	0AH	-	Disabled
Read Program Memory	0BH	-	Disabled
Write Data Memory	0CH	-	Disabled
Read Data Memory	0DH	-	Disabled
Read Program Memory CRC	0EH	-	-
Reserved	0FH	-	-
Step Instruction	10H	-	Disabled
Stuff Instruction	11H	-	Disabled
Execute Instruction	12H	-	Disabled
Reserved	13H - FFH	-	-

In the following list of OCD Commands, data and commands sent from the host to the On-Chip Debugger are identified by 'DBG ← Command/Data'. Data sent from the On-Chip Debugger back to the host is identified by 'DBG → Data'.

- Read OCD Revision (00H)**—The Read OCD Revision command determines the version of the On-Chip Debugger. If OCD commands are added, removed, or changed, this revision number changes.
 - DBG ← 00H
 - DBG → OCDREV[15:8] (Major revision number)
 - DBG → OCDREV[7:0] (Minor revision number)
- Read OCD Status Register (02H)**—The Read OCD Status Register command reads the OCDSTAT register.
 - DBG ← 02H
 - DBG → OCDSTAT[7:0]

eZ8 CPU loops on the BRK instruction.
0 = BRK instruction sets DBGMODE to 1.
1 = eZ8 CPU loops on BRK instruction.

Reserved
These bits are reserved and must be 0.

RST—Reset
Setting this bit to 1 resets the 64K Series devices. The devices go through a normal Power-On Reset sequence with the exception that the On-Chip Debugger is not reset. This bit is automatically cleared to 0 when the reset finishes.
0 = No effect
1 = Reset the 64K Series device

OCD Status Register

The OCD Status register (Table 103) reports status information about the current state of the debugger and the system.

Table 103. OCD Status Register (OCDSTAT)

BITS	7	6	5	4	3	2	1	0
FIELD	IDLE	HALT	RPEN	Reserved				
RESET	0							
R/W	R							

IDLE—CPU idling
This bit is set if the part is in DEBUG mode (DBGMODE is 1), or if a BRK instruction occurred since the last time OCDCTL was written. This can be used to determine if the CPU is running or if it is idling.
0 = The eZ8 CPU is running.
1 = The eZ8 CPU is either stopped or looping on a BRK instruction.

HALT—HALT Mode
0 = The device is not in HALT mode.
1 = The device is in HALT mode.

RPEN—Read Protect Option Bit Enabled
0 = The Read Protect Option Bit is disabled (1).
1 = The Read Protect Option Bit is enabled (0), disabling many OCD commands.

Reserved
These bits are always 0.

DC Characteristics

Table 106 lists the DC characteristics of the 64K Series products. All voltages are referenced to V_{SS} , the primary system ground.

Table 106. DC Characteristics

Symbol	Parameter	$T_A = -40\text{ }^{\circ}\text{C to }125\text{ }^{\circ}\text{C}$			Units	Conditions
		Minimum	Typical	Maximum		
V_{DD}	Supply Voltage	3.0	–	3.6	V	
V_{IL1}	Low Level Input Voltage	-0.3	–	$0.3 \cdot V_{DD}$	V	For all input pins except $\overline{\text{RESET}}$, DBG, XIN
V_{IL2}	Low Level Input Voltage	-0.3	–	$0.2 \cdot V_{DD}$	V	For $\overline{\text{RESET}}$, DBG, and XIN.
V_{IH1}	High Level Input Voltage	$0.7 \cdot V_{DD}$	–	5.5	V	Port A, C, D, E, F, and G pins.
V_{IH2}	High Level Input Voltage	$0.7 \cdot V_{DD}$	–	$V_{DD} + 0.3$	V	Port B and H pins.
V_{IH3}	High Level Input Voltage	$0.8 \cdot V_{DD}$	–	$V_{DD} + 0.3$	V	$\overline{\text{RESET}}$, DBG, and XIN pins
V_{OL1}	Low Level Output Voltage Standard Drive	–	–	0.4	V	$I_{OL} = 2\text{ mA}$; $V_{DD} = 3.0\text{ V}$ High Output Drive disabled.
V_{OH1}	High Level Output Voltage Standard Drive	2.4	–	–	V	$I_{OH} = -2\text{ mA}$; $V_{DD} = 3.0\text{ V}$ High Output Drive disabled.
V_{OL2}	Low Level Output Voltage High Drive	–	–	0.6	V	$I_{OL} = 20\text{ mA}$; $V_{DD} = 3.3\text{ V}$ High Output Drive enabled $T_A = -40\text{ }^{\circ}\text{C to }+70\text{ }^{\circ}\text{C}$
V_{OH2}	High Level Output Voltage High Drive	2.4	–	–	V	$I_{OH} = -20\text{ mA}$; $V_{DD} = 3.3\text{ V}$ High Output Drive enabled; $T_A = -40\text{ }^{\circ}\text{C to }+70\text{ }^{\circ}\text{C}$
V_{OL3}	Low Level Output Voltage High Drive	–	–	0.6	V	$I_{OL} = 15\text{ mA}$; $V_{DD} = 3.3\text{ V}$ High Output Drive enabled; $T_A = +70\text{ }^{\circ}\text{C to }+105\text{ }^{\circ}\text{C}$
V_{OH3}	High Level Output Voltage High Drive	2.4	–	–	V	$I_{OH} = 15\text{ mA}$; $V_{DD} = 3.3\text{ V}$ High Output Drive enabled; $T_A = +70\text{ }^{\circ}\text{C to }+105\text{ }^{\circ}\text{C}$
V_{RAM}	RAM Data Retention	0.7	–	–	V	
I_{IL}	Input Leakage Current	-5	–	+5	μA	$V_{DD} = 3.6\text{ V}$; $V_{IN} = V_{DD}\text{ or }V_{SS}^1$
I_{TL}	Tri-State Leakage Current	-5	–	+5	μA	$V_{DD} = 3.6\text{ V}$

Table 108. External RC Oscillator Electrical Characteristics and Timing

Symbol	Parameter	T _A = –40 °C to 125 °C			Units	Conditions
		Minimum	Typical ¹	Maximum		
V _{DD}	Operating Voltage Range	2.70 ¹	–	–	V	
R _{EXT}	External Resistance from XIN to VDD	40	45	200	kΩ	V _{DD} = V _{VBO}
C _{EXT}	External Capacitance from XIN to VSS	0	20	1000	pF	
F _{OSC}	External RC Oscillation Frequency	–	–	4	MHz	

¹When using the external RC oscillator mode, the oscillator may stop oscillating if the power supply drops below 2.7 V, but before the power supply drops to the voltage brown-out threshold. The oscillator will resume oscillation as soon as the supply voltage exceeds 2.7 V.

Table 109. Reset and Stop Mode Recovery Pin Timing

Symbol	Parameter	T _A = –40 °C to 125 °C			Units	Conditions
		Minimum	Typical	Maximum		
T _{RESET}	RESET pin assertion to initiate a system reset.	4	–	–	T _{CLK}	Not in STOP Mode. T _{CLK} = System Clock period.
T _{SMR}	Stop Mode Recovery pin Pulse Rejection Period	10	20	40	ns	RESET, DBG, and GPIO pins configured as SMR sources.

Ordering Information

Part Number	Flash	RAM	I/O Lines	Interrupts	16-Bit Timers w/PWM	10-Bit A/D Channels	I ² C	SPI	UARTs with IrDA	Description
Z8F642x with 64 KB Flash, 10-Bit Analog-to-Digital Converter										
Standard Temperature: 0 °C to 70 °C										
Z8F6421PM020SC	64 KB	4 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F6421AN020SC	64 KB	4 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F6421VN020SC	64 KB	4 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F6422AR020SC	64 KB	4 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F6422VS020SC	64 KB	4 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F6423FT020SC	64 KB	4 KB	60	24	4	12	1	1	2	QFP 80-pin package
Extended Temperature: -40 °C to +105 °C										
Z8F6421PM020EC	64 KB	4 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F6421AN020EC	64 KB	4 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F6421VN020EC	64 KB	4 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F6422AR020EC	64 KB	4 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F6422VS020EC	64 KB	4 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F6423FT020EC	64 KB	4 KB	60	24	4	12	1	1	2	QFP 80-pin package
Automotive/Industrial Temperature: -40 °C to +125 °C										
Z8F6421PM020AC	64 KB	4 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F6421AN020AC	64 KB	4 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F6421VN020AC	64 KB	4 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F6422AR020AC	64 KB	4 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F6422VS020AC	64 KB	4 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F6423FT020AC	64 KB	4 KB	60	24	4	12	1	1	2	QFP 80-pin package