



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	31
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f3221vn020ec00tr

Table 3. Signal Descriptions (Continued)

Signal Mnemonic	I/O	Description
XIN	I	External Crystal Input. This is the input pin to the crystal oscillator. A crystal can be connected between it and the XOUT pin to form the oscillator. This signal is usable with external RC networks and an external clock driver.
XOUT	O	External Crystal Output. This pin is the output of the crystal oscillator. A crystal can be connected between it and the XIN pin to form the oscillator. When the system clock is referred to in this manual, it refers to the frequency of the signal at this pin. This pin must be left unconnected when not using a crystal.
RCOUT	O	RC Oscillator Output. This signal is the output of the RC oscillator. It is multiplexed with a general-purpose I/O pin. This signal must be left unconnected when not using a crystal.
On-Chip Debugger		
DBG	I/O	Debug. This pin is the control and data input and output to and from the On-Chip Debugger. This pin is open-drain.
 Caution: For operation of the On-Chip Debugger, all power pins (V_{DD} and AV_{DD}) must be supplied with power and all ground pins (V_{SS} and AV_{SS}) must be properly grounded.		
<p><i>The DBG pin is open-drain and must have an external pull-up resistor to ensure proper operation.</i></p>		
Reset		
RESET	I	RESET. Generates a Reset when asserted (driven Low).
Power Supply		
VDD	I	Power Supply.
AVDD	I	Analog Power Supply.
VSS	I	Ground.
AVSS	I	Analog Ground.

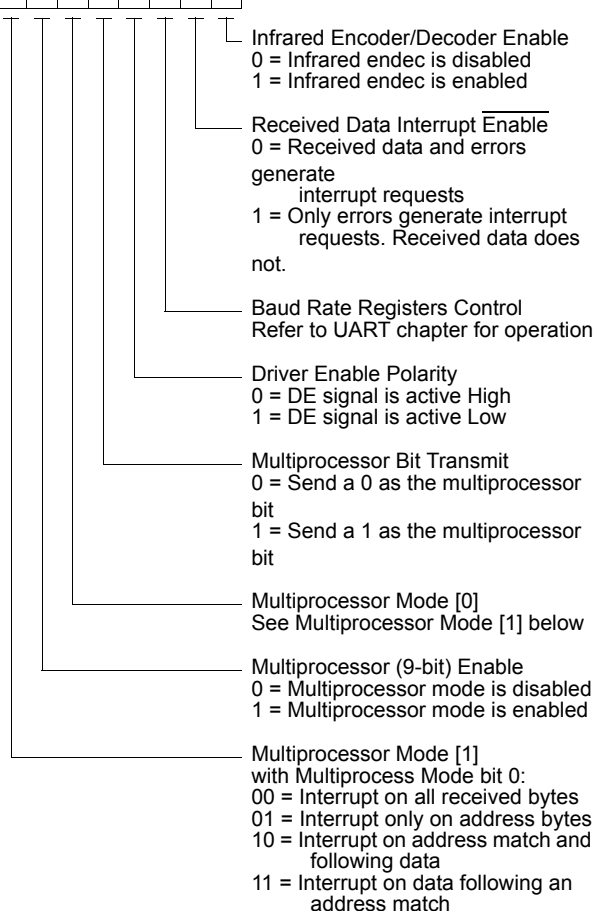
Pin Characteristics

Table 4 on page 17 provides detailed information on the characteristics for each pin available on the 64K Series products and the data is sorted alphabetically by the pin symbol mnemonic.

UART0 Control 1

U0CTL1 (F43H - Read/Write)

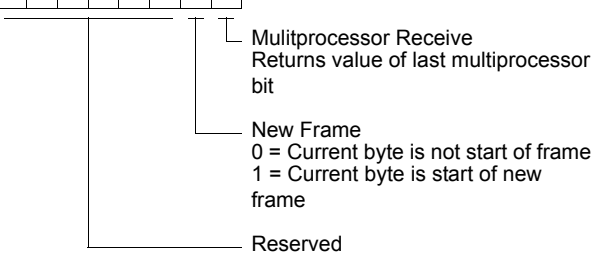
D7 D6 D5 D4 D3 D2 D1 D0



UART0 Status 1

U0STAT1 (F44H - Read Only)

D7 D6 D5 D4 D3 D2 D1 D0



UART0 Address Compare

U0ADDR (F45H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



UART0 Baud Rate Generator High Byte

U0BRH (F46H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



UART0 Baud Rate Generator Low Byte

U0BRL (F47H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



UART1 Transmit Data

U1TXD (F48H - Write Only)

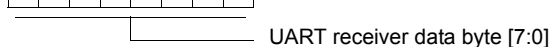
D7 D6 D5 D4 D3 D2 D1 D0



UART1 Receive Data

U1RXD (F48H - Read Only)

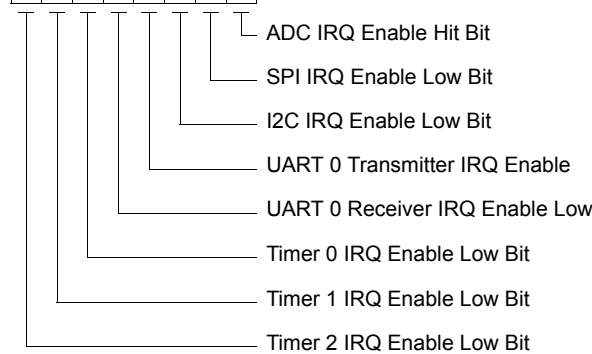
D7 D6 D5 D4 D3 D2 D1 D0



IRQ0 Enable Low Bit

IRQ0ENL (FC2H - Read/Write)

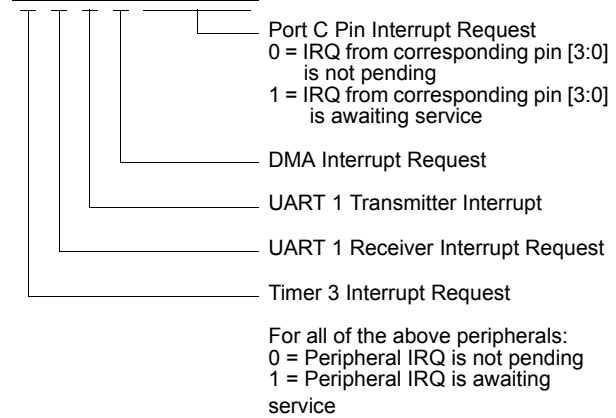
D7 D6 D5 D4 D3 D2 D1 D0



Interrupt Request 2

IRQ2 (FC6H - Read/Write)

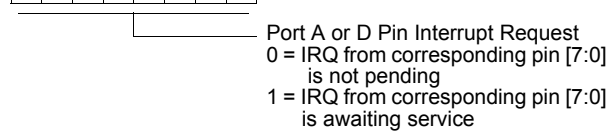
D7 D6 D5 D4 D3 D2 D1 D0



Interrupt Request 1

IRQ1 (FC3H - Read/Write)

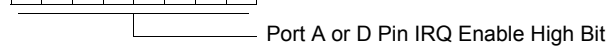
D7 D6 D5 D4 D3 D2 D1 D0



IRQ1 Enable High Bit

IRQ1ENH (FC4H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



IRQ1 Enable Low Bit

IRQ1ENL (FC5H - Read/Write)

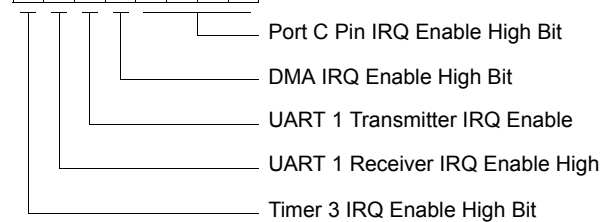
D7 D6 D5 D4 D3 D2 D1 D0



IRQ2 Enable High Bit

IRQ2ENH (FC7H - Read/Write)

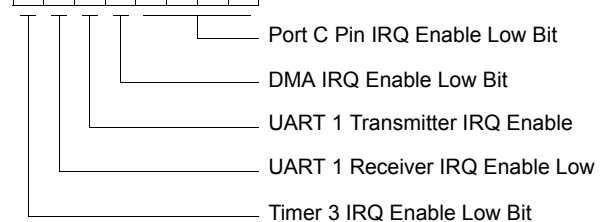
D7 D6 D5 D4 D3 D2 D1 D0



IRQ2 Enable Low Bit

IRQ2ENL (FC8H - Read/Write)

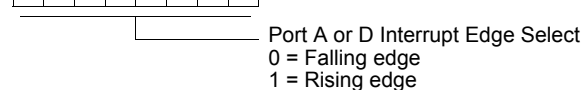
D7 D6 D5 D4 D3 D2 D1 D0



Interrupt Edge Select

IRQES (FCDH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0



Interrupt Port Select

IRQPS (FCEH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port A or D Port Pin Select [7:0]
0 = Port A pin is the interrupt source
1 = Port D pin is the interrupt source

Interrupt Control

IRQCTL (FCFH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Reserved
Interrupt Request Enable
0 = Interrupts are disabled
1 = Interrupts are enabled

Port A Address

PAADDR (FD0H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port A Address[7:0]
Selects Port Sub-Registers:
00H = No function
01H = Data direction
02H = Alternate function
03H = Output control (open-drain)
04H = High drive enable
05H = Stop Mode Recovery enable
06H-FFH = No function

Port A Control

PACTL (FD1H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port A Control[7:0]
Provides Access to Port Sub-Registers

Port A Input Data

PAIN (FD2H - Read Only)

D7 D6 D5 D4 D3 D2 D1 D0

Port A Input Data [7:0]

Port A Output Data

PAOUT (FD3H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port A Output Data [7:0]

Port B Address

PBADDR (FD4H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port B Address[7:0]
Selects Port Sub-Registers:
00H = No function
01H = Data direction
02H = Alternate function
03H = Output control (open-drain)
04H = High drive enable
05H = Stop Mode Recovery enable
06H-FFH = No function

Port B Control

PBCTL (FD5H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port B Control[7:0]
Provides Access to Port Sub-Registers

Port B Input Data

PBIN (FD6H - Read Only)

D7 D6 D5 D4 D3 D2 D1 D0

Port B Input Data [7:0]

Port B Output Data

PBOUT (FD7H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port B Output Data [7:0]

Port C Address

PCADDR (FD8H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

Port C Address[7:0]
Selects Port Sub-Registers:
00H = No function
01H = Data direction
02H = Alternate function
03H = Output control (open-drain)
04H = High drive enable
05H = Stop Mode Recovery enable
06H-FFH = No function

Low-Power Modes

Overview

The 64K Series products contain power-saving features. The highest level of power reduction is provided by STOP mode. The next level of power reduction is provided by the HALT mode.

STOP Mode

Execution of the eZ8[™] CPU's STOP instruction places the device into STOP mode. In STOP mode, the operating characteristics are:

- Primary crystal oscillator is stopped; the XIN pin is driven High and the XOUT pin is driven Low.
- System clock is stopped.
- eZ8 CPU is stopped.
- Program counter (PC) stops incrementing.
- The Watchdog Timer and its internal RC oscillator continue to operate, if enabled for operation during STOP mode.
- The Voltage Brownout protection circuit continues to operate, if enabled for operation in STOP mode using the associated Option Bit.
- All other on-chip peripherals are idle.

To minimize current in STOP mode, all GPIO pins that are configured as digital inputs must be driven to one of the supply rails (V_{CC} or GND), the Voltage Brownout protection must be disabled, and the Watchdog Timer must be disabled. The devices can be brought out of STOP mode using Stop Mode Recovery. For more information on Stop Mode Recovery, see [Reset and Stop Mode Recovery](#) on page 47.



Caution: *STOP mode must not be used when driving the 64K Series devices with an external clock driver source.*

Table 19. Port A–H High Drive Enable Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	PHDE7	PHDE6	PHDE5	PHDE4	PHDE3	PHDE2	PHDE1	PHDE0
RESET	0							
R/W	R/W							
ADDR	If 04H in Port A–H Address Register, accessible through Port A–H Control Register							

PHDE[7:0]—Port High Drive Enabled

0 = The Port pin is configured for standard output current drive.

1 = The Port pin is configured for high output current drive.

Port A–H Stop Mode Recovery Source Enable Sub-Registers

The Port A–H Stop Mode Recovery Source Enable sub-register (Table 20) is accessed through the Port A–H Control register by writing 05H to the Port A–H Address register. Setting the bits in the Port A–H Stop Mode Recovery Source Enable sub-registers to 1 configures the specified Port pins as a Stop Mode Recovery source. During STOP Mode, any logic transition on a Port pin enabled as a Stop Mode Recovery source initiates Stop Mode Recovery.

Table 20. Port A–H Stop Mode Recovery Source Enable Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	PSMRE7	PSMRE6	PSMRE5	PSMRE4	PSMRE3	PSMRE2	PSMRE1	PSMRE0
RESET	0							
R/W	R/W							
ADDR	If 05H in Port A–H Address Register, accessible through Port A–H Control Register							

PSMRE[7:0]—Port Stop Mode Recovery Source Enabled

0 = The Port pin is not configured as a Stop Mode Recovery source. Transitions on this pin during STOP mode do not initiate Stop Mode Recovery.

1 = The Port pin is configured as a Stop Mode Recovery source. Any logic transition on this pin during STOP mode initiates Stop Mode Recovery.



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

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

The PWM period is given by the following equation:

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

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

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

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

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

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

CAPTURE Mode

In CAPTURE mode, the current timer count value is recorded when the desired external Timer Input transition occurs. The Capture count value is written to the Timer PWM High and Low Byte Registers. The timer input is the system clock. The TPOL bit in the Timer Control 1 register determines if the Capture occurs on a rising edge or a falling edge of the Timer Input signal. When the Capture event occurs, an interrupt is generated and the timer continues counting.

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.

5. Write the Watchdog Timer Reload Low Byte register (WDTL).

All steps of the Watchdog Timer Reload Unlock sequence must be written in the order just listed. There must be no other register writes between each of these operations. If a register write occurs, the lock state machine resets and no further writes can occur, unless the sequence is restarted. The value in the Watchdog Timer Reload registers is loaded into the counter when the Watchdog Timer is first enabled and every time a WDT instruction is executed.

Watchdog Timer Control Register Definitions

Watchdog Timer Control Register

The Watchdog Timer Control (WDTCTL) register (Table 48) is a Read-Only register that 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.

Writing the 55H, AAH unlock sequence to the Watchdog Timer Control (WDTCTL) register address unlocks the three Watchdog Timer Reload Byte registers (WDTU, WDTL, 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.

Table 48. Watchdog Timer Control Register (WDTCTL)

BITS	7	6	5	4	3	2	1	0
FIELD	POR	STOP	WDT	EXT	Reserved			SM
RESET	See descriptions below			0				
R/W	R							
ADDR	FF0H							

Reset or Stop Mode Recovery Event	POR	STOP	WDT	EXT
Power-On Reset	1	0	0	0
Reset using RESET pin assertion	0	0	0	1
Reset using Watchdog Timer time-out	0	0	1	0
Reset using the On-Chip Debugger (OCDCTL[1] set to 1)	1	0	0	0
Reset from STOP Mode using DBG Pin driven Low	1	0	0	0
Stop Mode Recovery using GPIO pin transition	0	1	0	0
Stop Mode Recovery using Watchdog Timer time-out	0	1	1	0

- Set or clear the CTSE bit to enable or disable control from the remote receiver using the $\overline{\text{CTS}}$ pin.
- 5. Check the TDRE bit in the UART Status 0 register to determine if the Transmit Data register is empty (indicated by a 1). If empty, continue to [step 6](#). If the Transmit Data register is full (indicated by a 0), continue to monitor the TDRE bit until the Transmit Data register becomes available to receive new data.
- 6. Write the UART Control 1 register to select the outgoing address bit.
- 7. Set the MULTIPROCESSOR Bit Transmitter (MPBT) if sending an address byte, clear it if sending a data byte.
- 8. Write the data byte to the UART Transmit Data register. The transmitter automatically transfers the data to the Transmit Shift register and transmits the data.
- 9. If desired and MULTIPROCESSOR mode is enabled, make any changes to the MULTIPROCESSOR Bit Transmitter (MPBT) value.
- 10. To transmit additional bytes, return to [step 5](#).

Transmitting Data using the Interrupt-Driven Method

The UART transmitter interrupt indicates the availability of the Transmit Data register to accept new data for transmission. Follow the steps below to configure the UART for interrupt-driven data transmission:

1. Write to the UART Baud Rate High and Low Byte registers to set the desired 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 Transmitter interrupt and set the desired priority.
5. If MULTIPROCESSOR mode is desired, write to the UART Control 1 register to enable MULTIPROCESSOR (9-bit) mode functions.
6. Set the MULTIPROCESSOR Mode Select (MPEN) to Enable MULTIPROCESSOR mode.
7. Write to the UART Control 0 register to:
 - Set the transmit enable bit (TEN) to enable the UART for data transmission.
 - Enable parity, if desired and if MULTIPROCESSOR mode is not enabled, and select either even or odd parity.
 - Set or clear the CTSE bit to enable or disable control from the remote receiver via the $\overline{\text{CTS}}$ pin.

Receiving Data using the Interrupt-Driven Method

The UART Receiver interrupt indicates the availability of new data (as well as error conditions). Follow the steps below to configure the UART receiver for interrupt-driven operation:

1. Write to the UART Baud Rate High and Low Byte registers to set the desired 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 desired 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 desired.
 - Set the MULTIPROCESSOR Mode Select (`MPEN`) to Enable MULTIPROCESSOR mode.
 - Set the MULTIPROCESSOR Mode Bits, `MPMD[1:0]`, to select the desired 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 multiprocessor 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 desired and if MULTIPROCESSOR mode is not enabled, and select either even or odd parity.
9. Execute an `EI` instruction to enable interrupts.

The UART is now configured for interrupt-driven data reception. When the UART Receiver interrupt is detected, the associated interrupt service routine performs the following:

1. Check the UART Status 0 register to determine the source of the interrupt - error, break, or received data.
2. If the interrupt was caused by data available, read the data from the UART Receive Data register. If operating in MULTIPROCESSOR (9-bit) mode, further actions may be required depending on the MULTIPROCESSOR Mode bits `MPMD[1:0]`.

reading the I²C Data register. Once the I²C data register has been read, the I²C reads the next data byte.

Address Only Transaction with a 7-bit Address

In the situation where software determines if a slave with a 7-bit address is responding without sending or receiving data, a transaction can be done which only consists of an address phase. Figure 28 displays this ‘address only’ transaction to determine if a slave with a 7-bit address will acknowledge. As an example, this transaction can be used after a ‘write’ has been done to a EEPROM to determine when the EEPROM completes its internal write operation and is once again responding to I²C transactions. If the slave does not Acknowledge, the transaction can be repeated until the slave does Acknowledge.

S	Slave Address	W = 0	A/ \bar{A}	P
---	---------------	-------	--------------	---

Figure 28. 7-Bit Address Only Transaction Format

Follow the steps below for an address only transaction to a 7-bit addressed slave:

1. Software asserts the IEN bit in the I²C Control register.
2. Software asserts the TXI bit of the I²C Control register to enable Transmit interrupts.
3. The I²C interrupt asserts, because the I²C Data register is empty (TDRE = 1)
4. Software responds to the TDRE bit by writing a 7-bit slave address plus write bit (=0) to the I²C Data register. As an alternative this could be a read operation instead of a write operation.
5. Software sets the START and STOP bits of the I²C Control register and clears the TXI bit.
6. The I²C Controller sends the START condition to the I²C slave.
7. The I²C Controller loads the I²C Shift register with the contents of the I²C Data register.
8. Software polls the STOP bit of the I²C Control register. Hardware deasserts the STOP bit when the address only transaction is completed.
9. 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 does not occur in the not acknowledge case because the STOP bit was set.

TXRXSTATE	State Description
1_1101	10-bit addressing: Bit 3 of 2nd address byte 7-bit addressing: Bit 3 of address byte
1_1110	10-bit addressing: Bit 2 of 2nd address byte 7-bit addressing: Bit 2 of address byte
1_1111	10-bit addressing: Bit 1 of 2nd address byte 7-bit addressing: Bit 1 of address byte

I²C Diagnostic Control Register

The I²C Diagnostic register (Table 76) provides control over diagnostic modes. This register is a read/write register used for I²C diagnostics.

Table 76. I²C Diagnostic Control Register (I2CDIAG)

BITS	7	6	5	4	3	2	1	0
FIELD	Reserved							DIAG
RESET	0							
R/W	R							R/W
ADDR	F56H							

DIAG = Diagnostic Control Bit - Selects read back value of the Baud Rate Reload registers.

0 = NORMAL mode. Reading the Baud Rate High and Low Byte registers returns the baud rate reload value.

1 = DIAGNOSTIC mode. Reading the Baud Rate High and Low Byte registers returns the baud rate counter value.

Option Bits

Overview

Option Bits allow user configuration of certain aspects of the 64K Series operation. The feature configuration data is stored in the Flash Memory and read during Reset. The features available for control via the Option Bits are:

- Watchdog Timer time-out response selection—interrupt or Reset.
- Watchdog Timer enabled at Reset.
- The ability to prevent unwanted read access to user code in Flash Memory.
- The ability to prevent accidental programming and erasure of the user code in Flash Memory.
- Voltage Brownout configuration—always enabled or disabled during STOP mode to reduce STOP mode power consumption.
- Oscillator mode selection—for high, medium, and low power crystal oscillators, or external RC oscillator.

Operation

Option Bit Configuration By Reset

Each time the Option Bits are programmed or erased, the device must be Reset for the change to take place. During any reset operation (System Reset, Reset, or Stop Mode Recovery), the Option Bits are automatically read from the Flash Memory and written to Option Configuration registers. The Option Configuration registers control operation of the devices within the 64K Series. Option Bit control is established before the device exits Reset and the eZ8 CPU begins code execution. The Option Configuration registers are not part of the Register File and are not accessible for read or write access.

Option Bit Address Space

The first two bytes of Flash Memory at addresses 0000H (see [Table 98](#) on page 196) and 0001H (see [Table 99](#) on page 197) are reserved for the user Option Bits. The byte at Flash Memory address 0000H configures user options. The byte at Flash Memory address 0001H is reserved for future use and must remain unprogrammed.

Figure 48 displays the maximum current consumption in STOP mode with the VBO disabled and Watchdog Timer enabled versus the power supply voltage. All GPIO pins are configured as outputs and driven High. Disabling the Watchdog Timer and its internal RC oscillator in STOP mode will provide some additional reduction in STOP mode current consumption. This small current reduction would be indistinguishable on the scale of Figure 48.

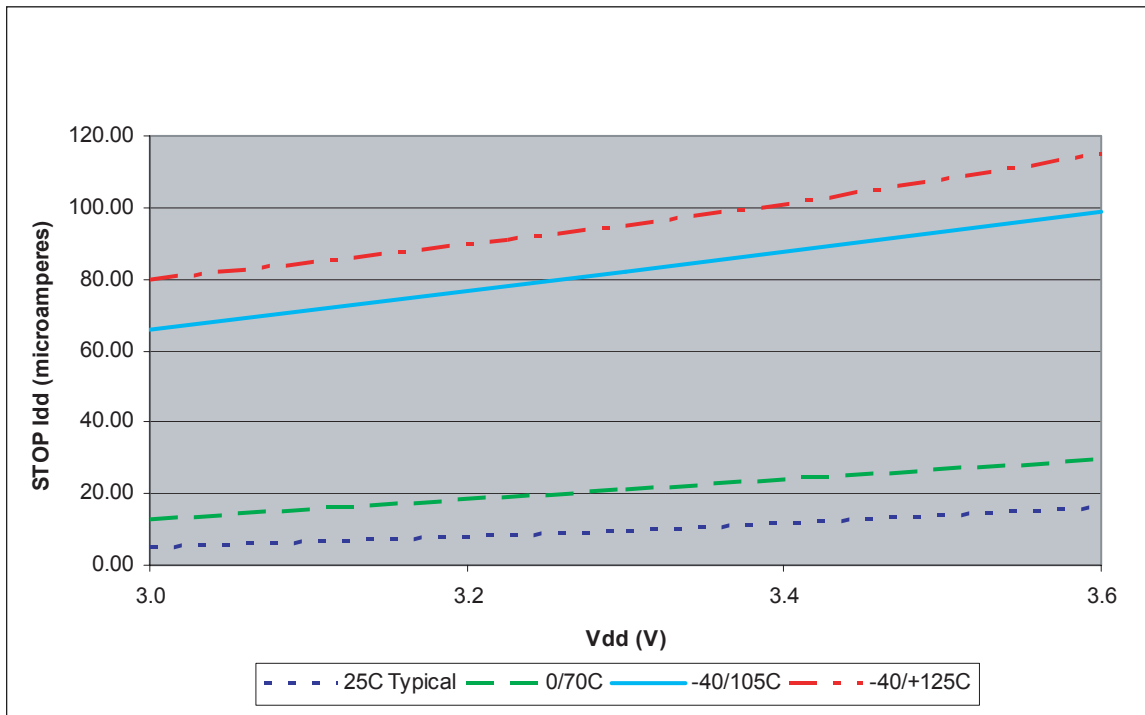


Figure 48. Maximum STOP Mode Idd with VBO Disabled versus Power Supply Voltage

Table 133. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode		Opcode(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
EI	IRQCTL[7] ← 1			9F	-	-	-	-	-	-	1	2
HALT	HALT Mode			7F	-	-	-	-	-	-	1	2
INC dst	dst ← dst + 1	R		20	-	*	*	*	-	-	2	2
		IR		21							2	3
		r		0E-FE							1	2
INCW dst	dst ← dst + 1	RR		A0	-	*	*	*	-	-	2	5
		IRR		A1							2	6
IRET	FLAGS ← @SP SP ← SP + 1 PC ← @SP SP ← SP + 2 IRQCTL[7] ← 1			BF	*	*	*	*	*	*	1	5
JP dst	PC ← dst	DA		8D	-	-	-	-	-	-	3	2
		IRR		C4							2	3
JP cc, dst	if cc is true PC ← dst	DA		0D-FD	-	-	-	-	-	-	3	2
JR dst	PC ← PC + X	DA		8B	-	-	-	-	-	-	2	2
JR cc, dst	if cc is true PC ← PC + X	DA		0B-FB	-	-	-	-	-	-	2	2
LD dst, rc	dst ← src	r	IM	0C-FC	-	-	-	-	-	-	2	2
		r	X(r)	C7							3	3
		X(r)	r	D7							3	4
		r	lr	E3							2	3
		R	R	E4							3	2
		R	IR	E5							3	4
		R	IM	E6							3	2
		IR	IM	E7							3	3
		lr	r	F3							2	3
		IR	R	F5							3	3

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
Z8F162x with 16 KB Flash, 10-Bit Analog-to-Digital Converter										
Standard Temperature: 0 °C to 70 °C										
Z8F1621PM020SC	16 KB	2 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F1621AN020SC	16 KB	2 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F1621VN020SC	16 KB	2 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F1622AR020SC	16 KB	2 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F1622VS020SC	16 KB	2 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Extended Temperature: –40 °C to +105 °C										
Z8F1621PM020EC	16 KB	2 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F1621AN020EC	16 KB	2 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F1621VN020EC	16 KB	2 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F1622AR020EC	16 KB	2 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F1622VS020EC	16 KB	2 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Automotive/Industrial Temperature: –40 °C to +125 °C										
Z8F1621PM020AC	16 KB	2 KB	29	23	3	8	1	1	2	PDIP 40-pin package
Z8F1621AN020AC	16 KB	2 KB	31	23	3	8	1	1	2	LQFP 44-pin package
Z8F1621VN020AC	16 KB	2 KB	31	23	3	8	1	1	2	PLCC 44-pin package
Z8F1622AR020AC	16 KB	2 KB	46	24	4	12	1	1	2	LQFP 64-pin package
Z8F1622VS020AC	16 KB	2 KB	46	24	4	12	1	1	2	PLCC 68-pin package
Z8F64200100KITG										Development Kit
ZUSBSC00100ZACG										USB Smart Cable Accessory Kit
ZUSBOPTSC01ZACG										Opto-Isolated USB Smart Cable Accessory Kit

Note: Replace C with G for lead-free packaging.

Index

Symbols

244
% 244
@ 244

Numerics

10-bit ADC 4
40-lead plastic dual-inline package 265
44-lead low-profile quad flat package 266
44-lead plastic lead chip carrier package 267
64-lead low-profile quad flat package 267
68-lead plastic lead chip carrier package 268
80-lead quad flat package 269

A

absolute maximum ratings 215
AC characteristics 231
ADC 246
 architecture 175
 automatic power-down 176
 block diagram 176
 continuous conversion 177
 control register 179
 control register definitions 179
 data high byte register 180
 data low bits register 180
 DMA control 178
 electrical characteristics and timing 229
 operation 176
 single-shot conversion 177
ADCCTL register 179
ADCDH register 180
ADC DL register 180
ADCX 246
ADD 246
add - extended addressing 246
add with carry 246
add with carry - extended addressing 246

additional symbols 244
address space 19
ADDX 246
analog signals 15
analog-to-digital converter (ADC) 175
AND 248
ANDX 248
arithmetic instructions 246
assembly language programming 241
assembly language syntax 242

B

B 244
b 243
baud rate generator, UART 113
BCLR 246
binary number suffix 244
BIT 246
bit 243
 clear 246
 manipulation instructions 246
 set 246
 set or clear 246
 swap 247
 test and jump 249
 test and jump if non-zero 249
 test and jump if zero 249
bit jump and test if non-zero 249
bit swap 249
block diagram 3
block transfer instructions 247
BRK 249
BSET 246
BSWAP 247, 249
BTJ 249
BTJNZ 249
BTJZ 249

C

CALL procedure 249
capture mode 95
capture/compare mode 95

- flash page select (FPS) 191
- flash status (FSTAT) 190
- GPIO port A-H address (PxADDR) 61
- GPIO port A-H alternate function sub-registers 63
- GPIO port A-H control address (PxCTL) 62
- GPIO port A-H data direction sub-registers 63
- I2C baud rate high (I2CBRH) 160, 161, 163
- I2C control (I2CCTL) 158
- I2C data (I2CDATA) 157
- I2C status 157
- I2C status (I2CSTAT) 157
- I2Cbaud rate low (I2CBRL) 161
- mode, SPI 140
- OCD control 209
- OCD status 210
- SPI baud rate high byte (SPIBRH) 142
- SPI baud rate low byte (SPIBRL) 142
- SPI control (SPICTL) 138
- SPI data (SPIDATA) 137
- SPI status (SPISTAT) 139
- status, I2C 157
- status, SPI 139
- UARTx baud rate high byte (UxBRH) 121
- UARTx baud rate low byte (UxBRL) 121
- UARTx Control 0 (UxCTL0) 117, 120
- UARTx control 1 (UxCTL1) 118
- UARTx receive data (UxRXD) 115
- UARTx status 0 (UxSTAT0) 115
- UARTx status 1 (UxSTAT1) 117
- UARTx transmit data (UxTXD) 114
- watch-dog timer control (WDTCTL) 100
- watch-dog timer reload high byte (WDTH) 102
- watch-dog timer reload low byte (WDTL) 102
- watch-dog timer reload upper byte (WDTU) 102
- register file 19
- register file address map 23
- register pair 243
- register pointer 244
- reset
 - and STOP mode characteristics 48
 - carry flag 247
 - controller 5
 - sources 48
- RET 249
- return 249
- RL 249
- RLC 249
- rotate and shift instructions 249
- rotate left 249
- rotate left through carry 249
- rotate right 249
- rotate right through carry 249
- RP 244
- RR 243, 249
- rr 243
- RRC 249
- S**
- SBC 246
- SCF 247
- SDA and SCL (IrDA) signals 145
- second opcode map after 1FH 264
- serial clock 131
- serial peripheral interface (SPI) 129
- set carry flag 247
- set register pointer 247
- shift right arithmetic 249
- shift right logical 250
- signal descriptions 14
- single-shot conversion (ADC) 177
- SIO 5
- slave data transfer formats (I2C) 151
- slave select 132
- software trap 249
- source operand 244
- SP 244
- SPI
 - architecture 129
 - baud rate generator 136
 - baud rate high and low byte register 142
 - clock phase 132
 - configured as slave 130
 - control register 137
 - control register definitions 137
 - data register 137