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	I²C, IrDA, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	11
Program Memory Size	4KB (4K 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	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SSOP (0.209", 5.30mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0411hh020ec

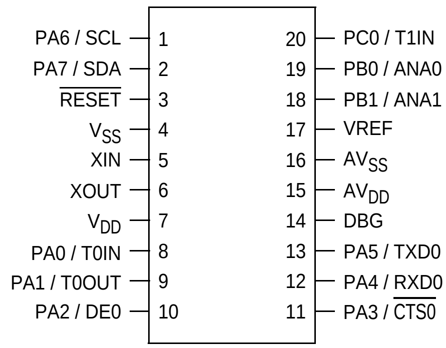

Figure 2. Z8F0821 and Z8F0421 in 20-Pin SSOP and PDIP Packages

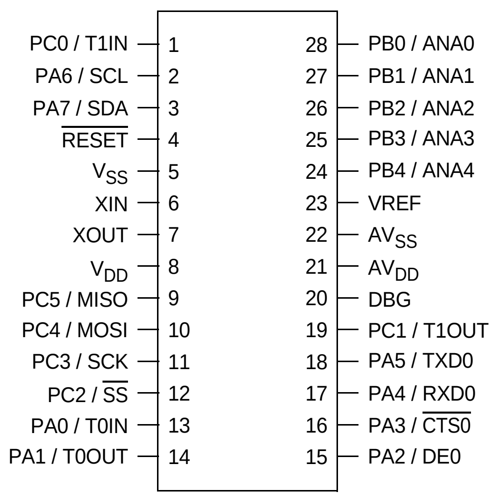

Figure 3. Z8F0822 and Z8F0422 in 28-Pin SOIC and PDIP Packages

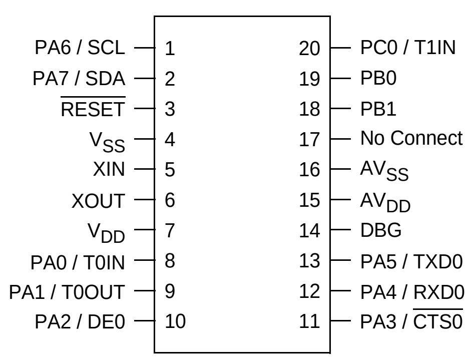

Figure 4. Z8F0811 and Z8F0411 in 20-Pin SSOP and PDIP Packages

Register File Address Map

Table 7 provides the address map for the Register File of the Z8 Encore! XP® F0822 Series products. Not all devices and package styles in the F0822 Series support the ADC, the SPI, or all of the GPIO Ports. Consider registers for unimplemented peripherals as Reserved.

Table 7. Register File Address Map

Address (Hex)	Register Description	Mnemonic	Reset (Hex)	Page No
General Purpose RAM				
000-3FF	General-Purpose Register File RAM	—	XX	
400-EFF	Reserved	—	XX	
Timer 0				
F00	Timer 0 High Byte	T0H	00	78
F01	Timer 0 Low Byte	T0L	01	78
F02	Timer 0 Reload High Byte	T0RH	FF	79
F03	Timer 0 Reload Low Byte	T0RL	FF	79
F04	Timer 0 PWM High Byte	T0PWMH	00	79
F05	Timer 0 PWM Low Byte	T0PWML	00	79
F06	Timer 0 Control 0	T0CTL0	00	81
F07	Timer 0 Control 1	T0CTL1	00	81
Timer 1				
F08	Timer 1 High Byte	T1H	00	78
F09	Timer 1 Low Byte	T1L	01	78
F0A	Timer 1 Reload High Byte	T1RH	FF	79
F0B	Timer 1 Reload Low Byte	T1RL	FF	79
F0C	Timer 1 PWM High Byte	T1PWMH	00	79
F0D	Timer 1 PWM Low Byte	T1PWML	00	79
F0E	Timer 1 Control 0	T1CTL0	00	81
F0F	Timer 1 Control 1	T1CTL1	00	81
F10-F3F	Reserved	—	XX	
UART 0				
F40	UART0 Transmit Data	U0TXD	XX	100
	UART0 Receive Data	U0RXD	XX	101
F41	UART0 Status 0	U0STAT0	0000011Xb	101
F42	UART0 Control 0	U0CTL0	00	103
F43	UART0 Control 1	U0CTL1	00	103
F44	UART0 Status 1	U0STAT1	00	101
F45	UART0 Address Compare Register	U0ADDR	00	105
F46	UART0 Baud Rate High Byte	U0BRH	FF	106

XX=Undefined

Control Register Summary

Timer 0 High Byte
T0H (F00H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 0 current count value

Timer 0 Low Byte
T0L (F01H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 0 current count value

Timer 0 Reload High Byte
T0RH (F02H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 0 reload value [15:8]

Timer 0 Reload Low Byte
T0RL (F03H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 0 reload value [7:0]

Timer 0 PWM High Byte
T0PWMH (F04H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 0 PWM value [15:8]

Timer 0 Control 0
T0CTL0 (F06H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D3–D0: Reserved
- D4: Cascade Timer
  0 = Timer 0 Input signal is GPIO pin
  1 = Timer 0 Input signal is Timer 1 out
- D7–D5: Reserved

Timer 0 Control 1
T0CTL1 (F07H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D2–D0: Timer Mode
  000 = One-Shot mode
  001 = Continuous mode
  010 = Counter mode
  011 = PWM mode
  100 = Capture mode
  101 = Compare mode
  110 = Gated mode
  111 = Capture/Compare mode
- D5–D3: Prescale Value
  000 = Divide by 1
  001 = Divide by 2
  010 = Divide by 4
  011 = Divide by 8
  100 = Divide by 16
  101 = Divide by 32
  110 = Divide by 64
  111 = Divide by 128
- D6: Timer Input/Output Polarity
  Operation of this bit is a function of the current operating mode of the timer
- D7: Timer Enable
  0 = Timer is disabled
  1 = Timer is enabled

Timer 1 High Byte
T1H (F08H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 1 current count value

Timer 1 Low Byte
T1L (F09H - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 1 current count value

Timer 1 Reload High Byte
T1RH (F0AH - Read/Write)
D7 D6 D5 D4 D3 D2 D1 D0

- D7–D0: Timer 1 reload value [15:8]

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 = Pull-up enable
07H-FFH = No function

Port C Control
PCCTL (FD9H - Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

Port C Control [5:0]
Provides Access to Port Sub-Registers

Reserved

Port C Input Data
PCIN (FDAH - Read Only)

D7	D6	D5	D4	D3	D2	D1	D0

Port C Input Data [5:0]

Reserved

Port C Output Data
PCOUT (FDBH - Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

Port C Output Data [5:0]

Reserved

External Pin Reset

The $\overline{\text{RESET}}$ pin contains a Schmitt-triggered input, an internal pull-up, an analog filter, and a digital filter to reject noise. After the $\overline{\text{RESET}}$ pin is asserted for at least 4 system clock cycles, the device progresses through the System Reset sequence. While the $\overline{\text{RESET}}$ input pin is asserted Low, Z8 Encore! XP F0822 Series device continues to be held in the Reset state. If the $\overline{\text{RESET}}$ pin is held Low beyond the System Reset time-out, the device exits the Reset state immediately following $\overline{\text{RESET}}$ pin deassertion. Following a System Reset initiated by the external $\overline{\text{RESET}}$ pin, the `EXT` status bit in the Watchdog Timer Control Register (WDTCTL) is set to 1.

On-Chip Debugger Initiated Reset

A POR is initiated using the OCD by setting the `RST` bit in the OCD Control Register. The OCD 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 execution of a `STOP` instruction by the eZ8 CPU. For detailed information on STOP mode, see Low-Power Modes on page 45. During Stop Mode Recovery, the device is held in reset for 66 cycles of the WDT oscillator followed by 16 cycles of the system clock. Stop Mode Recovery only affects the contents of the WDT Control Register and 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 WDT Control Register is set to 1. Table 10 lists the Stop Mode Recovery sources and resulting actions. The text following provides more detailed information on each of the Stop Mode Recovery sources.

Table 10. Stop Mode Recovery Sources and Resulting Action

Operating Mode	Stop Mode Recovery Source	Action
STOP mode	WDT time-out when configured for Reset	Stop Mode Recovery
	WDT 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

Port A–C Control Registers

The Port A–C Control Registers set the GPIO port operation. The value in the corresponding Port A–C Address Register determines the control sub-registers accessible using the Port A–C Control Register (Table 15).

Table 15. Port A–C Control Registers (P*x*CTL)

BITS	7	6	5	4	3	2	1	0
FIELD	PCTL							
RESET	00H							
R/W	R/W							
ADDR	FD1H, FD5H, FD9H							

PCTL[7:0]—Port Control
The Port Control Register provides access to all sub-registers that configure the GPIO Port operation.

Port A–C Data Direction Sub-Registers

The Port A–C Data Direction sub-register is accessed through the Port A–C Control register by writing 01H to the Port A–C Address Register (Table 16).

Table 16. Port A–C Data Direction Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
RESET	1							
R/W	R/W							
ADDR	If 01H in Port A–C Address Register, accessible through the Port A–C Control Register							

DD[7:0]—Data Direction

These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction register setting.
0 = Output. Data in the Port A–C Output Data Register is driven onto the port pin.
1 = Input. The port pin is sampled and the value written into the Port A–C Input Data Register. The output driver is tri-stated.

Port A–C Alternate Function Sub-Registers

The Port A–C Alternate Function sub-register (Table 17) is accessed through the Port A–C Control Register by writing 02H to the Port A–C Address Register. The Port A–C Alternate Function sub-registers set the alternate functions for the selected

Interrupt Controller

The interrupt controller on Z8 Encore! XP® F0822 Series products prioritizes the interrupt requests from the on-chip peripherals and the GPIO port pins. The features of the interrupt controller include the following:

- 19 unique interrupt vectors:
  - 12 GPIO port pin interrupt sources.
  - 7 On-chip peripheral interrupt sources.
- Flexible GPIO interrupts:
  - 8 selectable rising and falling edge GPIO interrupts.
  - 4 dual-edge interrupts.
- Three levels of individually programmable interrupt priority.
- WDT is configured to generate an interrupt.

Interrupt Requests (IRQs) allow peripheral devices to suspend CPU operation in an orderly manner and force the CPU to start an Interrupt Service Routine (ISR). Usually this ISR is involved with the exchange of data, status information, or control information between the CPU and the interrupting peripheral. When the service routine is completed, the CPU returns to the operation from which it was interrupted.

The eZ8 CPU supports both vectored and polled interrupt handling. For polled interrupts, the interrupt control has no effect on operation. For more information on interrupt servicing, refer to eZ8 CPU Core User Manual (UM0128) available for download at www.zilog.com.

Interrupt Vector Listing

Table 24 lists all the interrupts available in order of priority. The interrupt vector is stored with the most significant byte (MSB) at the even Program Memory address and the least significant byte (LSB) at the following odd Program Memory address.

Table 24. Interrupt Vectors in Order of Priority

Priority	Program Memory Vector Address	Interrupt Source
Highest	0002H	Reset (not an interrupt)
	0004H	WDT (see Watchdog Timer on page 83)
	0006H	Illegal Instruction Trap (not an interrupt)

Architecture

Figure 9 displays a block diagram of the interrupt controller.

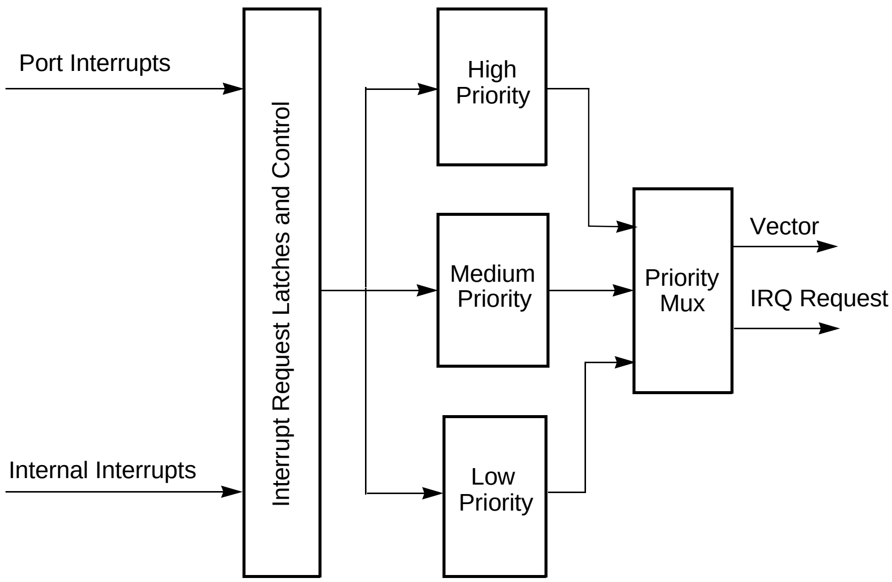

Figure 9. Interrupt Controller Block Diagram

Operation

Master Interrupt Enable

The master interrupt enable bit (IRQE) in the Interrupt Control Register globally enables and disables interrupts.

Interrupts are globally enabled by any of the following actions:

- Execution of an EI (Enable Interrupt) instruction.
- Execution of an IRET (Return from Interrupt) instruction.
- 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 DI (Disable Interrupt) 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.

Watchdog Timer

Watchdog Timer (WDT) protects against corrupt or unreliable software, power faults, and other system-level problems which can place the Z8 Encore! XP® F0822 Series device into unsuitable operating states. It includes the following features:

- On-chip RC oscillator.
- A selectable time-out response—Reset or Interrupt.
- 24-bit programmable time-out value.

Operation

WDT is a retriggerable one-shot timer that resets or interrupts the Z8 Encore! XP F0822 Series device when the WDT reaches its terminal count. It uses its own dedicated on-chip RC oscillator as its clock source. The WDT has only two modes of operation—ON and OFF. When enabled, it always counts and must be refreshed to prevent a time-out. An enable is performed by executing the WDT instruction or by setting the `WDT_AO` Option Bit. The `WDT_AO` bit enables the WDT to operate all the time, even if a WDT instruction has not been executed.

The WDT is a 24-bit reloadable downcounter that uses three 8-bit registers in the eZ8 CPU register space to set the reload value. The nominal WDT time-out period is given by the following equation:

$$\text{WDT Time-out Period (ms)} = \frac{\text{WDT Reload Value}}{10}$$

where the WDT reload value is the decimal value of the 24-bit value given by {WDTU[7:0], WDTH[7:0], WDTL[7:0]} and the typical Watchdog Timer RC oscillator frequency is 10 kHz. WDT cannot be refreshed once it reaches `000002H`. The WDT Reload Value must not be set to values below `000004H`. Table 47 provides information on approximate time-out delays for minimum and maximum WDT reload values.

Table 47. Watchdog Timer Approximate Time-Out Delays

WDT Reload Value (Hex)	WDT Reload Value (Decimal)	Approximate Time-Out Delay (with 10 kHz typical WDT Oscillator Frequency) Typical	Description
000004	4	400 µs	Minimum time-out delay
FFFFFF	16,777,215	1677.5 s	Maximum time-out delay

Universal Asynchronous Receiver/Transmitter

The Universal Asynchronous Receiver/Transmitter (UART) is a full-duplex communication channel capable of handling asynchronous data transfers. The UART uses a single 8-bit data mode with selectable parity. Features of the UART include:

- 8-bit asynchronous data transfer
- Selectable even- and odd-parity generation and checking
- Option of one or two STOP bits
- Separate transmit and receive interrupts
- Framing, parity, overrun, and break detection
- Separate transmit and receive enables
- 16-bit Baud Rate Generator
- Selectable Multiprocessor (9-bit) mode with three configurable interrupt schemes
- BRG timer mode
- Driver Enable output for external bus transceivers

Architecture

The UART consists of three primary functional blocks: Transmitter, Receiver, and Baud Rate Generator. The UART's transmitter and receiver function independently, but use the same baud rate and data format. Figure 11 on page 90 displays the UART architecture.

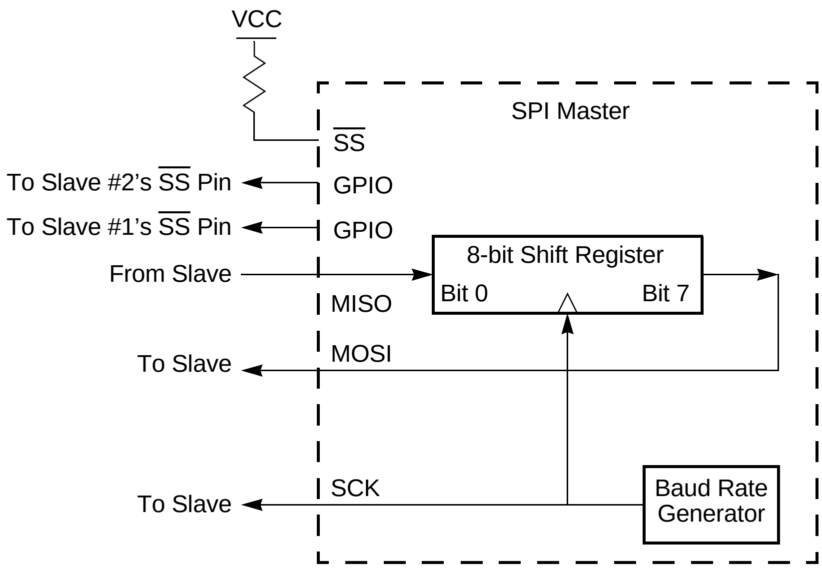

Figure 21. SPI Configured as a Master in a Single Master, Multiple Slave System

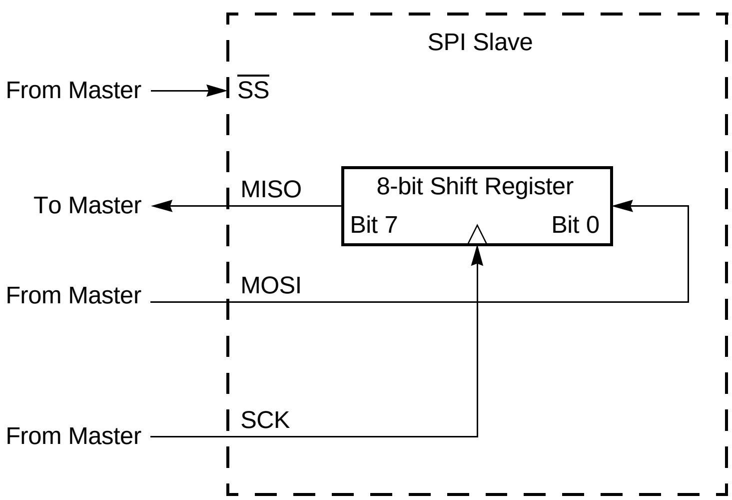

Figure 22. SPI Configured as a Slave

Operation

The SPI is a full-duplex, synchronous, and character-oriented channel that supports a four-wire interface (serial clock, transmit, receive and Slave select). The SPI block consists of a transmit/receive shift register, a Baud Rate (clock) Generator and a control unit.

1 = $\overline{SS}$ pin driven High (1).
This bit has no effect if SSIO = 0 or SPI configured as a Slave

SPI Diagnostic State Register

The SPI Diagnostic State Register provides observability of internal state. This is a read only register used for SPI diagnostics.

Table 67. SPI Diagnostic State Register (SPIDST)

BITS	7	6	5	4	3	2	1	0
FIELD	SCKEN	TCKEN	SPISTATE					
RESET	0							
R/W	R							
ADDR	F64H							

SCKEN–Shift Clock Enable
0 = The internal Shift Clock Enable signal is deasserted
1 = The internal Shift Clock Enable signal is asserted (shift register is updates on next system clock)

TCKEN–Transmit Clock Enable
0 = The internal Transmit Clock Enable signal is deasserted.
1 = The internal Transmit Clock Enable signal is asserted. When this is asserted the serial data out is updated on the next system clock (MOSI or MISO).

SPISTATE–SPI State Machine
Defines the current state of the internal SPI State Machine.

SPI Baud Rate High and Low Byte Registers

The SPI Baud Rate High and Low Byte Registers combine to form a 16-bit reload value, BRG[15:0], for the SPI Baud Rate Generator. When configured as a general purpose timer, the interrupt interval is calculated using the following equation:

Interrupt Interval (s) = System Clock Period (s) × BRG[15:0]

Table 68. SPI Baud Rate High Byte Register (SPIBRH)

BITS	7	6	5	4	3	2	1	0
FIELD	BRH							
RESET	1							
R/W	R/W							
ADDR	F66H							

I²C Controller

The I²C Controller makes the F0822 Series products bus-compatible with the I²C protocol. The I²C Controller consists of two bidirectional bus lines—a serial data signal (SDA) and a serial clock signal (SCL). Features of the I²C Controller include:

- Transmit and Receive Operation in MASTER mode.
- Maximum data rate of 400 kbit/s.
- 7-bit and 10-bit addressing modes for Slaves.
- Unrestricted number of data bytes transmitted per transfer.

The I²C Controller in the F0822 Series products does not operate in Slave mode.

Architecture

Figure 25 displays the architecture of the I²C Controller.

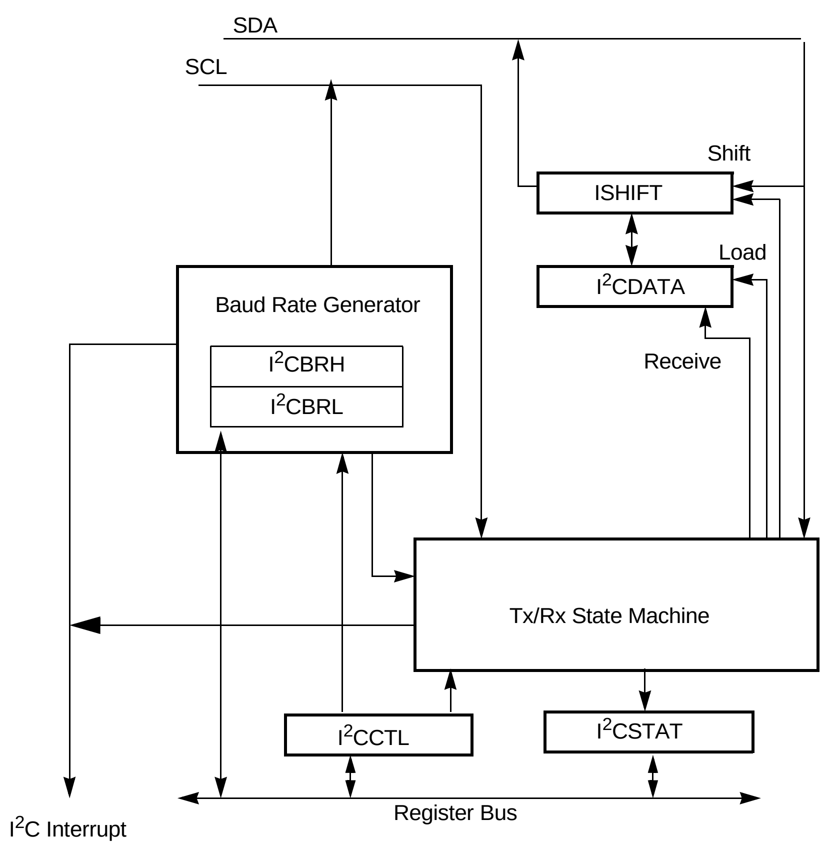

Figure 25. I ²C Controller Block Diagram

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 I^2C Controller shifts in the byte of data from the I^2C Slave on the SDA signal. The I^2C Controller sends a Not Acknowledge to the I^2C Slave if the NAK bit is set (last byte), else it sends an Acknowledge.
8. The I^2C Controller asserts the Receive interrupt (RDRF bit set in the Status register).
9. Software responds by reading the I^2C Data Register which clears the RDRF bit. If there is only one more byte to receive, set the NAK bit of the I^2C 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 I^2C Controller.
12. Software responds by setting the `STOP` bit of the I^2C Control Register.
13. A STOP condition is sent to the I^2C Slave, the STOP and NCKI bits are cleared.

Read Transaction with a 10-Bit Address

Figure 31 displays the read transaction format for a 10-bit addressed slave. The shaded regions indicate data transferred from the I^2C Controller to slaves and unshaded regions indicate data transferred from the slaves to the I^2C 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	$\overline{A}$	P

Figure 31. 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.

Follow the steps below for the data transfer procedure for a read operation to a 10-bit addressed slave:

1. Software writes `11110B` followed by the two address bits and a 0 (write) to the I^2C Data Register.
2. Software asserts the `START` and TXI bits of the I^2C Control Register.
3. The I^2C Controller sends the Start condition.
4. The I^2C Controller loads the I^2C Shift register with the contents of the I^2C Data Register.

Flash Memory Address 0000H

Table 89. Option Bits at Flash Memory Address 0000H for 8K Series Flash Devices

BITS	7	6	5	4	3	2	1	0
FIELD	WDT_RES	WDT_AO	OSC_SEL[1:0]		VBO_AO	RP	Reserved	FWP
RESET	U							
R/W	R/W							
ADDR	Program Memory 0000H							
Note: U = Unchanged by Reset. R/W = Read/Write.								

WDT_RES—Watchdog Timer Reset
0 = Watchdog Timer time-out generates an interrupt request. Interrupts must be globally enabled for the eZ8 CPU to acknowledge the interrupt request.
1 = Watchdog Timer time-out causes a Reset. This setting is the default for unprogrammed (erased) Flash.

WDT_AO—Watchdog Timer Always On
0 = Watchdog Timer is automatically enabled upon application of system power. Watchdog Timer can not be disabled.
1 = Watchdog Timer is enabled upon execution of the WDT instruction. Once enabled, the Watchdog Timer can only be disabled by a Reset or Stop Mode Recovery. This setting is the default for unprogrammed (erased) Flash.

OSC_SEL[1:0]—OSCILLATOR Mode Selection
00 = On-chip oscillator configured for use with external RC networks (<4 MHz).
01 = Minimum power for use with very low frequency crystals (32 kHz to 1.0 MHz).
10 = Medium power for use with medium frequency crystals or ceramic resonators (0.5 MHz to 10.0 MHz).
11 = Maximum power for use with high frequency crystals (8.0 MHz to 20.0 MHz). This setting is the default for unprogrammed (erased) Flash.

VBO_AO—Voltage Brownout Protection Always On
0 = Voltage Brownout Protection is disabled in STOP mode to reduce total power consumption.
1 = Voltage Brownout Protection is always enabled including during STOP mode. This setting is the default for unprogrammed (erased) Flash.

RP—Read Protect
0 = User program code is inaccessible. Limited control features are available through the OCD.
1 = User program code is accessible. All OCD commands are enabled. This setting is the default for unprogrammed (erased) Flash.

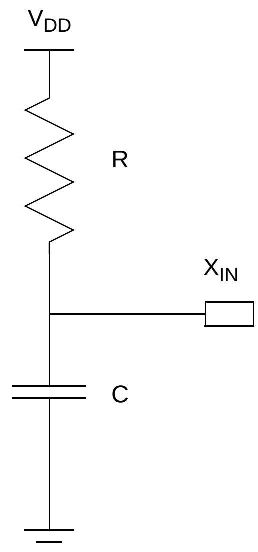

Figure 35. Connecting the On-Chip Oscillator to an External RC Network

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

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

Figure 36 on page 170 displays the typical (3.3 V and 25 °C) oscillator frequency as a function of the capacitor (C in pF) employed in the RC network assuming a 45 kΩ external resistor. For very small values of C, the parasitic capacitance of the oscillator XIN pin and the printed circuit board should be included in the estimation of the oscillator frequency.

It is possible to operate the RC oscillator using only the parasitic capacitance of the package and printed circuit board. To minimize sensitivity to external parasitics, external capacitance values in excess of 20 pF are recommended.

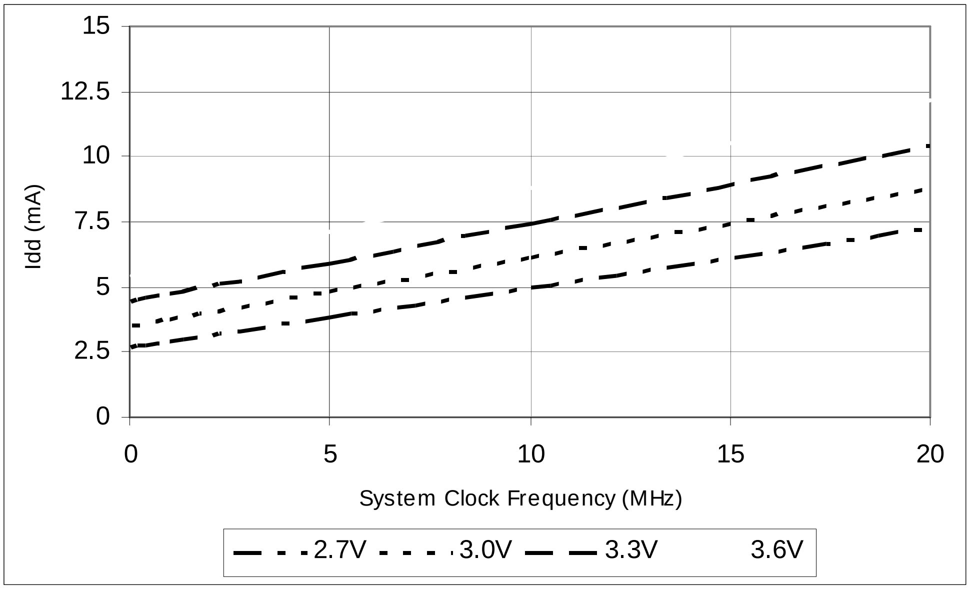

Figure 41. Typical Active Mode I_{DD} Versus System Clock Frequency

Figure 42 displays the maximum active mode current consumption across the full operating temperature range of the device and versus the system clock frequency. All GPIO pins are configured as outputs and driven High.

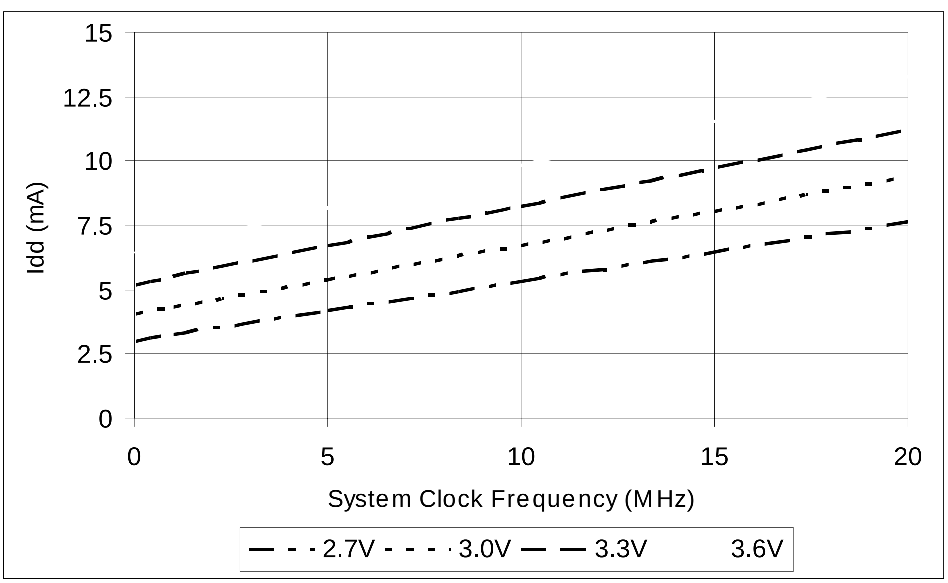

Figure 42. Maximum Active Mode I_{DD} Versus System Clock Frequency

Figure 45 displays the maximum current consumption in STOP mode with the VBO and Watchdog Timer enabled versus the power supply voltage. All GPIO pins are configured as outputs and driven High.

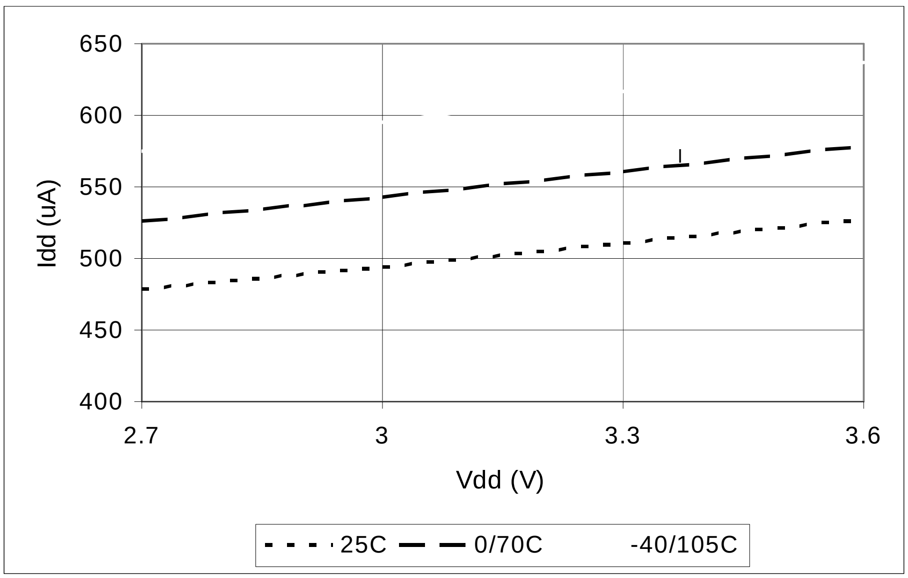

Figure 45. Maximum STOP Mode I_{DD} with VBO Enabled versus Power Supply Voltage

Figure 46 on page 193 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 is indistinquishable on the scale of Figure 46 on page 193.

Index