



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	Active
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I ² C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	19
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	-40°C ~ 105°C (TA)
Mounting Type	Through Hole
Package / Case	28-DIP (0.600", 15.24mm)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0812pj020eg

Infrared Endec Control Register Definitions	100
Serial Peripheral Interface	101
Architecture	101
Operation	103
SPI Signals	103
SPI Clock Phase and Polarity Control	104
Multimaster Operation	106
Slave Operation	107
Error Detection	107
SPI Interrupts	108
SPI Baud Rate Generator	108
SPI Control Register Definitions	109
SPI Data Register	109
SPI Control Register	110
SPI Status Register	111
SPI Mode Register	112
SPI Diagnostic State Register	113
SPI Baud Rate High and Low Byte Registers	114
I2C Controller	115
Architecture	115
Operation	116
SDA and SCL Signals	117
I ² C Interrupts	117
Software Control of I2C Transactions	118
Start and Stop Conditions	119
Master Write and Read Transactions	119
Address Only Transaction with a 7-Bit Address	120
Write Transaction with a 7-Bit Address	121
Address-Only Transaction with a 10-Bit Address	122
Write Transaction with a 10-Bit Address	123
Read Transaction with a 7-Bit Address	125
Read Transaction with a 10-Bit Address	126
I2C Control Register Definitions	128
I2C Data Register	129
I2C Status Register	129
I2C Control Register	131
I2C Baud Rate High and Low Byte Registers	132
I2C Diagnostic State Register	133
I2C Diagnostic Control Register	135
Analog-to-Digital Converter	136

Figure 34.	On-Chip Debugger Block Diagram	158
Figure 35.	Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface, #1 of 2	159
Figure 36.	Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface, #2 of 2	160
Figure 37.	OCD Data Format	161
Figure 38.	Recommended 20MHz Crystal Oscillator Configuration	173
Figure 39.	Connecting the On-Chip Oscillator to an External RC Network	174
Figure 40.	Typical RC Oscillator Frequency as a Function of External Capacitance with a 45kΩ Resistor 175	
Figure 41.	Typical Active Mode IDD vs. System Clock Frequency	179
Figure 42.	Maximum Active Mode IDD vs. System Clock Frequency	180
Figure 43.	Typical HALT Mode IDD vs. System Clock Frequency	181
Figure 44.	Maximum HALT Mode ICC vs. System Clock Frequency	182
Figure 45.	Maximum STOP Mode I _{DD} with VBO Enabled vs. Power Supply Voltage	183
Figure 46.	Maximum STOP Mode IDD with VBO Disabled vs. Power Supply Voltage	184
Figure 47.	Analog-to-Digital Converter Frequency Response	189
Figure 48.	Port Input Sample Timing	191
Figure 49.	GPIO Port Output Timing	192
Figure 50.	On-Chip Debugger Timing	193
Figure 51.	SPI MASTER Mode Timing	194
Figure 52.	SPI SLAVE Mode Timing	195
Figure 53.	I ² C Timing	196
Figure 54.	UART Timing with CTS	197
Figure 55.	UART Timing without CTS	198
Figure 56.	Flags Register	217
Figure 57.	Op Code Map Cell Description	218
Figure 58.	First Op Code Map	219
Figure 59.	Second Op Code Map after 1FH	220

List of Tables

Table 1.	Z8 Encore! XP® F0822 Series Part Selection Guide	2
Table 2.	Z8 Encore! XP® F0822 Series Package Options	7
Table 3.	Signal Descriptions	10
Table 4.	Pin Characteristics	13
Table 5.	Z8 Encore! XP® F0822 Series Program Memory Maps	15
Table 6.	Information Area Map	16
Table 7.	Register File Address Map	17
Table 8.	Reset and Stop Mode Recovery Characteristics and Latency	21
Table 9.	Reset Sources and Resulting Reset Type	22
Table 10.	Stop Mode Recovery Sources and Resulting Action	25
Table 11.	Port Availability by Device and Package Type	29
Table 12.	Port Alternate Function Mapping	30
Table 13.	GPIO Port Registers and Subregisters	31
Table 14.	Port A–C GPIO Address Registers (PxADDR)	32
Table 15.	Port A–C Control Registers (PxCTL)	33
Table 16.	Port A–C Data Direction Subregisters	33
Table 17.	Port A–CA–C Alternate Function Subregisters	34
Table 18.	Port A–C Output Control Subregisters	35
Table 19.	Port A–C High Drive Enable Subregisters	36
Table 20.	Port A–C Stop Mode Recovery Source Enable Subregisters	37
Table 21.	Port A–C Pull-Up Enable Subregisters	38
Table 22.	Port A–C Input Data Registers (PxIN)	38
Table 23.	Port A–C Output Data Register (PxOUT)	39
Table 24.	Interrupt Vectors in Order of Priority	41
Table 25.	Interrupt Request 0 Register (IRQ0)	45
Table 26.	Interrupt Request 1 Register (IRQ1)	46
Table 27.	Interrupt Request 2 Register (IRQ2)	47
Table 28.	IRQ0 Enable and Priority Encoding	48
Table 29.	IRQ0 Enable High Bit Register (IRQ0ENH)	48
Table 30.	IRQ0 Enable Low Bit Register (IRQ0ENL)	49
Table 31.	IRQ1 Enable and Priority Encoding	49
Table 32.	IRQ1 Enable High Bit Register (IRQ1ENH)	50
Table 33.	IRQ1 Enable Low Bit Register (IRQ1ENL)	50

10-Bit Analog-to-Digital Converter

The optional Analog-to-Digital Converter (ADC) converts an analog input signal to a 10-bit binary number. The ADC accepts inputs from 2 to 5 different analog input sources.

UART

The Universal Asynchronous Receiver/Transmitter (UART) is full-duplex and capable of handling asynchronous data transfers. The UART supports 8-bit and 9-bit data modes and selectable parity.

I²C

The Inter-Integrated Circuit (I²C) controller makes the Z8 Encore! XP compatible with the I²C protocol. The I²C Controller consists of two bidirectional bus lines, a serial data (SDA) line, and a serial clock (SCL) line.

Serial Peripheral Interface

The Serial Peripheral Interface (SPI) allows the Z8 Encore! XP to exchange data between other peripheral devices such as EEPROMs, A/D converters, and ISDN devices. The SPI is a full-duplex, synchronous, and character-oriented channel that supports a four-wire interface.

Timers

Two 16-bit reloadable timers are used for timing/counting events or for motor control operations. These timers provide a 16-bit programmable reload counter and operate in One-Shot, Continuous, Gated, Capture, Compare, Capture and Compare, and PWM modes.

Interrupt Controller

Z8 Encore! XP® F0822 Series products support up to 18 interrupts. These interrupts consist of 7 internal peripheral interrupts and 11 GPIO pin interrupt sources. The interrupts have 3 levels of programmable interrupt priority.

Reset Controller

Z8 Encore! XP® F0822 Series products are reset using the $\overline{\text{RESET}}$ pin, POR, WDT, STOP Mode exit, or VBO warning signal.

of Reset, all GPIO pins are configured as inputs. All GPIO programmable pull-ups are disabled.

During Reset, the eZ8 CPU and the on-chip peripherals are idle; however, the on-chip crystal oscillator and WDT oscillator continue to run. The system clock begins operating following the WDT oscillator cycle count. The eZ8 CPU and on-chip peripherals remain idle through all of the 16 cycles of the system clock.

Upon Reset, control registers within the Register File which have a defined Reset value are loaded with their reset values. Other control registers (including the Stack Pointer, Register Pointer, and Flags) and general-purpose RAM are undefined following the Reset. 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.

Reset Sources

Table 9 lists the reset sources as a function of the operating mode. The remainder of this section provides more detail about the individual reset sources.

► **Note:** A POR/VBO event always has priority over all other possible reset sources to ensure a full system reset occurs.

Table 9. Reset Sources and Resulting Reset Type

Operating Mode	Reset Source	Reset Type
NORMAL or HALT modes	POR/VBO	System Reset
	WDT time-out when configured for Reset	System Reset
	RESET pin assertion	System Reset
	OCD-initiated Reset (OCDCTL[0] set to 1)	System Reset except the OCD is unaffected by the reset
STOP Mode	POR/ VBO	System Reset
	RESET pin assertion	System Reset
	DBG pin driven Low	System Reset

Power-On Reset

Each device in the Z8 Encore! XP® F0822 Series contains an internal POR circuit. The POR circuit monitors the supply voltage and holds the device in the Reset state until the supply voltage reaches a safe operating level. After the supply voltage exceeds the POR

Bit	Description (Continued)
[1] SPII	SPI Interrupt Request 0 = No interrupt request is pending for the SPI. 1 = An interrupt request from the SPI is awaiting service.
[0] ADCI	ADC Interrupt Request 0 = No interrupt request is pending for the ADC. 1 = An interrupt request from the ADC is awaiting service.

Interrupt Request 1 Register

The Interrupt Request 1 (IRQ1) Register, shown in Table 26, stores interrupt requests for both vectored and polled interrupts. When a request is presented to the interrupt controller, the corresponding bit in the IRQ1 Register becomes 1. If interrupts are globally enabled (vectored interrupts), the interrupt controller passes an interrupt request to the eZ8 CPU. If interrupts are globally disabled (polled interrupts), the eZ8 CPU reads the IRQ1 Register to determine if any interrupt requests are pending.

Table 26. Interrupt Request 1 Register (IRQ1)

Bit	7	6	5	4	3	2	1	0
Field	PA7I	PA6I	PA5I	PA4I	PA3I	PA2I	PA1I	PA0I
RESET	0							
R/W	R/W							
Address	FC3H							

Bit	Description
[7:0] PAxI	Port A Pin x Interrupt Request 0 = No interrupt request is pending for GPIO Port A pin x. 1 = An interrupt request from GPIO Port A pin x is awaiting service.

Note: x indicates register bits in the range [7:0].

IRQ2 Enable High and Low Bit Registers

Table 34 describes the priority control for IRQ2. The IRQ2 Enable High and Low Bit registers, shown in Tables 35 and 36, form a priority-encoded enabling for interrupts in the Interrupt Request 2 Register. Priority is generated by setting bits in each register.

Table 34. IRQ2 Enable and Priority Encoding

IRQ2ENH[x]	IRQ2ENL[x]	Priority	Description
0	0	Disabled	Disabled
0	1	Level 1	Low
1	0	Level 2	Nominal
1	1	Level 3	High

Note: x indicates register bits in the range [7:0].

Table 35. IRQ2 Enable High Bit Register (IRQ2ENH)

Bit	7	6	5	4	3	2	1	0
Field	Reserved				C3ENH	C2ENH	C1ENH	C0ENH
RESET	0							
R/W	R/W							
Address	FC7H							

Bit	Description
[7:4]	Reserved These bits are reserved and must be programmed to 0000.
[3] C3ENH	Port C3 Interrupt Request Enable High Bit
[2] C2ENH	Port C2 Interrupt Request Enable High Bit
[1] C1ENH	Port C1 Interrupt Request Enable High Bit
[0] C0ENH	Port C0 Interrupt Request Enable High Bit

Watchdog Timer Reload Upper, High and Low Byte Registers

The Watchdog Timer Reload Upper, High and Low Byte (WDTU, WDTL, WDTL) registers, shown in Tables 50 through 52, form the 24-bit reload value that is loaded into the WDT, when a WDT instruction executes. The 24-bit reload value is {WDTU[7:0], WDTL[7:0], WDTL[7:0]}. Writing to these registers sets the required reload value. Reading from these registers returns the current WDT count value.

! Caution: The 24-bit WDT reload value must not be set to a value less than 000004H.

Table 50. Watchdog Timer Reload Upper Byte Register (WDTU)

Bit	7	6	5	4	3	2	1	0
Field	WDTU							
RESET	1							
R/W	R/W*							
Address	FF1H							
Note: *R/W = a read returns the current WDT count value; a write sets the appropriate reload value.								

Bit	Description
[7:0] WDTU	WDT Reload Upper Byte Most significant byte (MSB), bits [23:16] of the 24-bit WDT reload value.

Table 51. Watchdog Timer Reload High Byte Register (WDTH)

Bit	7	6	5	4	3	2	1	0
Field	WDTH							
RESET	1							
R/W	R/W*							
Address	FF2H							
Note: *R/W = a read returns the current WDT count value; a write sets the appropriate reload value.								

Bit	Description
[7:0] WDTH	WDT Reload High Byte Middle byte, bits [15:8] of the 24-bit WDT reload value.

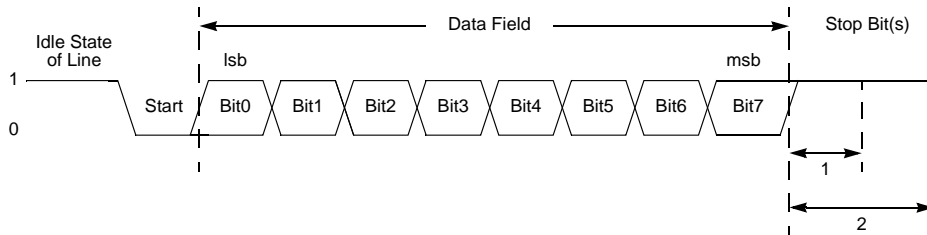


Figure 12. UART Asynchronous Data Format without Parity

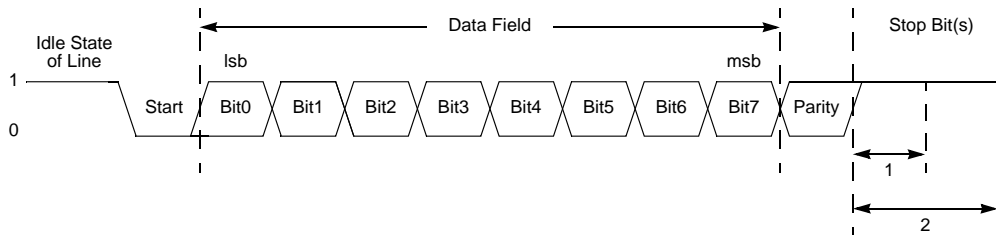


Figure 13. UART Asynchronous Data Format with Parity

Transmitting Data using Polled Method

Observe the following procedure to transmit data using polled method of operation:

1. Write to the UART Baud Rate High Byte and Low Byte registers to set the required baud rate.
2. Enable the UART pin functions by configuring the associated GPIO port pins for alternate function operation.
3. If MULTIPROCESSOR Mode is required, write to the UART Control 1 Register to enable multiprocessor (9-bit) mode functions.
 - Set the Multiprocessor Mode Select (MPEN) to enable MULTIPROCESSOR Mode.
4. Write to the UART Control 0 Register to:
 - Set the transmit enable bit (TEN) to enable the UART for data transmission
 - If parity is required, and MULTIPROCESSOR Mode is not enabled, set the parity enable bit (PEN) and select either even or odd parity (PSEL).

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. After one bit of an address is shifted out by the SDA signal, the transmit interrupt is asserted.
9. Software responds by writing the second byte of the 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 and the I²C Controller sets the ACK bit in the I²C Status Register, continue to 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 the remaining steps in this sequence.

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 equal to 1. If the slave does not acknowledge, the ACK bit is equal to 0. The NCKI interrupt do not occur because the stop bit was set.

Write Transaction with a 10-Bit Address

Figure 29 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 Seven Bits	W = 0	A	Slave Address 2nd Byte	A	Data	A	Data	A/A	P/S
---	---------------------------------	-------	---	---------------------------	---	------	---	------	-----	-----

Figure 29. 10-Bit Addressed Slave Data Transfer Format

16. The I²C Controller loads the I²C Shift Register with the contents of the I²C Data Register (third address transfer).
17. The I²C Controller sends 11110B followed by the two most significant bits of the slave read address and a 1 (read).
18. The I²C Slave sends an acknowledge by pulling the SDA signal Low during the next High period of SCL.

If the slave were to Not Acknowledge at this point (this should not happen because the slave did acknowledge the first two address bytes), software would respond 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 the remainder of this sequence.

19. The I²C Controller shifts in a 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.
20. The I²C Controller asserts the Receive interrupt (RDRF bit set in the Status Register).
21. 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.
22. If there are one or more bytes to transfer, return to [Step 19](#).
23. After the last byte is shifted in, a Not Acknowledge interrupt is generated by the I²C Controller.
24. Software responds by setting the stop bit of the I²C Control Register.
25. A stop condition is sent to the I²C Slave and the stop and NCKI bits are cleared.

I²C Control Register Definitions

This section defines the features of the following I²C Control registers.

I²C Data Register: see page 129

I²C Status Register: see page 129

I²C Control Register: see page 131

I²C Baud Rate High and Low Byte Registers: see page 132

I²C Diagnostic State Register: see page 133

I²C Diagnostic Control Register: see page 135

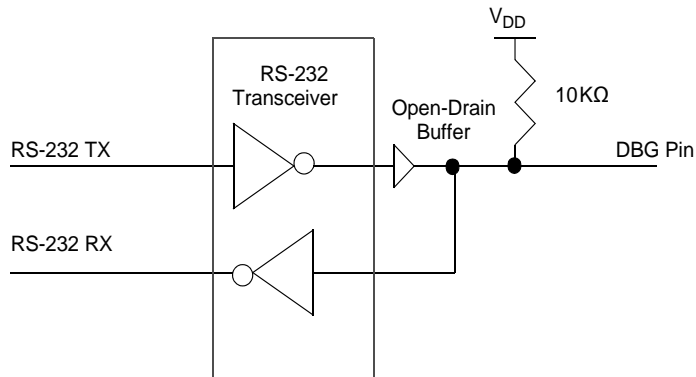


Figure 36. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface, #2 of 2

Debug Mode

The operating characteristics of the Z8 Encore! XP® F0822 Series devices in DEBUG Mode are:

- The eZ8 CPU fetch unit stops, idling the eZ8 CPU, unless directed by the OCD to execute specific instructions
- The system clock operates unless in STOP Mode
- All enabled on-chip peripherals operate unless in STOP Mode
- Automatically exits HALT Mode
- Constantly refreshes the Watchdog Timer, if enabled

Entering Debug Mode

The device enters DEBUG Mode following any of the following operations:

- Writing the DBGMODE bit in the OCD Control Register to 1 using the OCD interface
- eZ8 CPU execution of a breakpoint (BRK) instruction
- Matching of the PC to the OCDCNTR Register (when enabled)
- The OCDCNTR Register decrements to 0000H (when enabled)
- If the DBG pin is Low when the device exits Reset, the OCD automatically places the device into DEBUG Mode

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

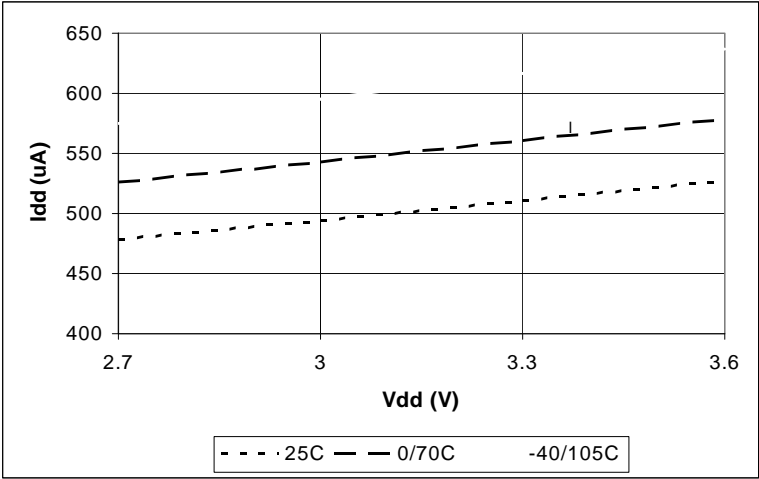


Figure 45. Maximum STOP Mode I_{DD} with VBO Enabled vs. Power Supply Voltage

Figure 46 displays the maximum current consumption in STOP Mode with the VBO disabled and Watchdog Timer enabled plotted opposite 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 indistinguishable on the scale of Figure 46.

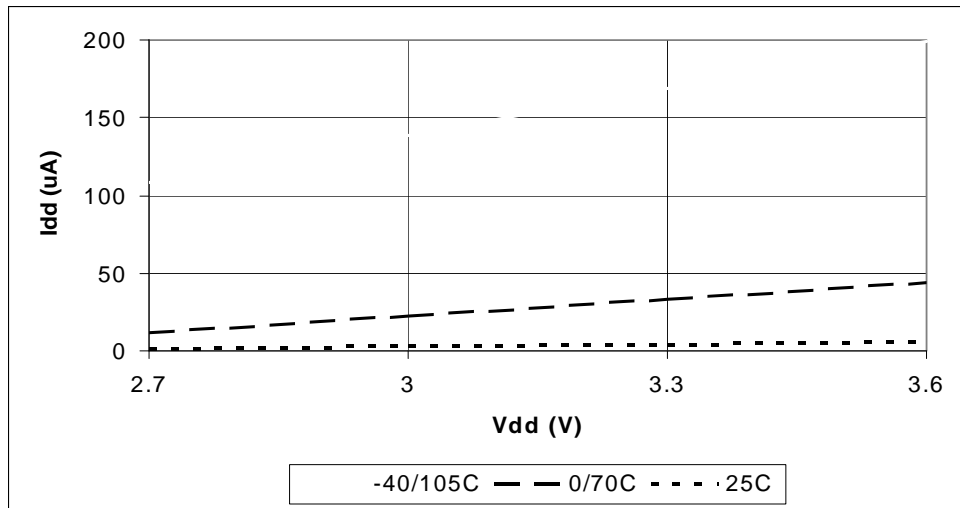


Figure 46. Maximum STOP Mode I_{DD} with VBO Disabled vs. Power Supply Voltage

On-Chip Peripheral AC and DC Electrical Characteristics

Table 100 provides information about the Power-On Reset and Voltage Brown-Out electrical characteristics.

Table 100. Power-On Reset and Voltage Brown-Out Electrical Characteristics and Timing

Symbol	Parameter	$T_A = -40^{\circ}\text{C to } 105^{\circ}\text{C}$			Units	Conditions
		Minimum	Typical*	Maximum		
V_{POR}	Power-On Reset Voltage Threshold	2.15	2.40	2.60	V	$V_{DD} = V_{POR}$
V_{VBO}	Voltage Brown-Out Reset Voltage Threshold	2.05	2.30	2.55	V	$V_{DD} = V_{VBO}$
	V_{POR} to V_{VBO} hysteresis	50	100	—	mV	
	Starting V_{DD} voltage to ensure valid POR	—	V_{SS}	—	V	
T_{ANA}	POR Analog Delay	—	50	—	μs	$V_{DD} > V_{POR}$; T_{POR} Digital Reset delay follows T_{ANA}
T_{POR}	POR Digital Delay	—	5.0	—	ms	50 WDT Oscillator cycles (10kHz) + 16 System Clock cycles (20MHz)
T_{VBO}	Voltage Brown-Out Pulse Rejection Period	—	10	—	μs	$V_{DD} < V_{VBO}$ to generate a Reset.
T_{RAMP}	Time for V_{DD} to transition from V_{SS} to V_{POR} to ensure valid Reset	0.10	—	100	ms	

Note: *Data in the typical column is from characterization at 3.3 V and 25°C. These values are provided for design guidance only and are not tested in production.

SPI MASTER Mode Timing

Figure 51 and Table 109 provide timing information for SPI MASTER Mode pins. Timing is shown with SCK rising edge used to source MOSI output data, SCK falling edge used to sample MISO input data. Timing on the SS output pin(s) is controlled by software.

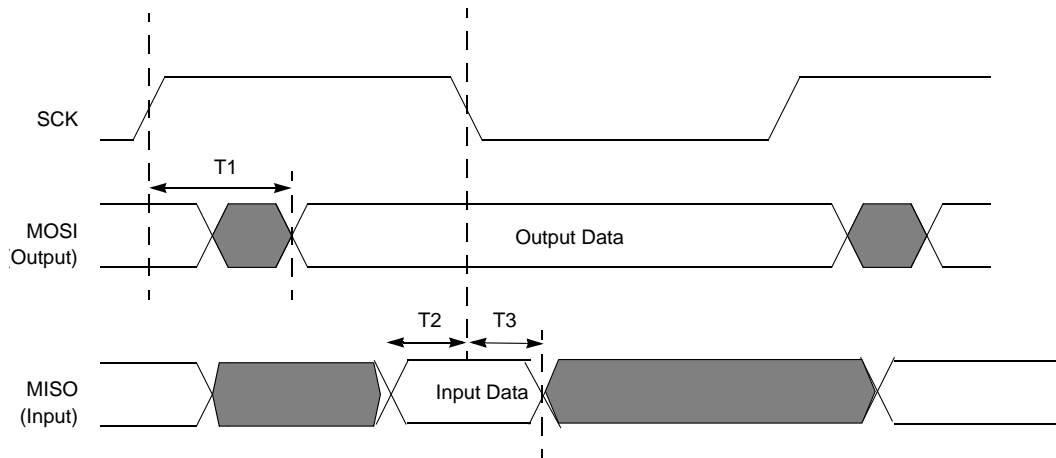


Figure 51. SPI MASTER Mode Timing

Table 109. SPI MASTER Mode Timing

Parameter	Abbreviation	Delay (ns)	
		Minimum	Maximum
SPI MASTER			
T ₁	SCK Rise to MOSI output Valid Delay	−5	+5
T ₂	MISO input to SCK (receive edge) Setup Time	20	
T ₃	MISO input to SCK (receive edge) Hold Time	0	

Table 126. Rotate and Shift Instructions

Mnemonic	Operands	Instruction
BSWAP	dst	Bit Swap
RL	dst	Rotate Left
RLC	dst	Rotate Left through Carry
RR	dst	Rotate Right
RRC	dst	Rotate Right through Carry
SRA	dst	Shift Right Arithmetic
SRL	dst	Shift Right Logical
SWAP	dst	Swap Nibbles

eZ8 CPU Instruction Summary

Table 127 summarizes the eZ8 CPU instructions. The table identifies the addressing modes employed by the instruction, the effect upon the Flags Register, the number of CPU clock cycles required for the instruction fetch, and the number of CPU clock cycles required for the instruction execution.

Table 127. eZ8 CPU Instruction Summary

Assembly Mnemonic	Symbolic Operation	Address Mode		Op Code(s) (Hex)	Flags						Fetch Cycles	Instr. Cycles
		dst	src		C	Z	S	V	D	H		
ADC dst, src	$\text{dst} \leftarrow \text{dst} + \text{src} + \text{C}$	r	r	12	*	*	*	*	0	*	2	3
		r	lr	13							2	4
		R	R	14							3	3
		R	IR	15							3	4
		R	IM	16							3	3
		IR	IM	17							3	4
ADCX dst, src	$\text{dst} \leftarrow \text{dst} + \text{src} + \text{C}$	ER	ER	18	*	*	*	*	0	*	4	3
		ER	IM	19							4	3

Note: Flags Notation:

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

– = Unaffected.

X = Undefined.

0 = Reset to 0.

1 = Set to 1.

Hex Address: F67

Table 168. SPI Baud Rate Low Byte Register (SPIBRL)

Bit	7	6	5	4	3	2	1	0
Field	BRL							
RESET	1							
R/W	R/W							
Address	F67H							

Hex Addresses: F68–F6F

This address range is reserved.

Analog-to-Digital Converter Control Registers

For more information about these ADC registers, see the [ADC Control Register Definitions](#) section on page 139.

Hex Address: F70

Table 169. ADC Control Register (ADCCTL)

Bit	7	6	5	4	3	2	1	0
Field	CEN	Reserved	VREF	CONT	ANAIN[3:0]			
RESET	0		1	0				
R/W	R/W							
Address	F70H							

Hex Address: F71

This address is reserved.

G

gated mode 69
 general-purpose I/O 29
 GPIO 4, 29
 alternate functions 29
 architecture 29
 control register definitions 31
 input data sample timing 191
 interrupts 31
 port A-C pull-up enable sub-registers 38
 port A-H address registers 32
 port A-H alternate function sub-registers 34
 port A-H control registers 33
 port A-H data direction sub-registers 33
 port A-H high drive enable sub-registers 36
 port A-H input data registers 38
 port A-H output control sub-registers 35
 port A-H output data registers 39
 port A-H stop mode recovery sub-registers 37
 port availability by device 29
 port input timing 191
 port output timing 192

H

H 202
 HALT 206
 halt mode 27, 206
 hexadecimal number prefix/suffix 202

I

I2C 4
 10-bit address read transaction 126
 10-bit address transaction 123
 10-bit addressed slave data transfer format 123
 10-bit receive data format 126
 7-bit address transaction 121
 7-bit address, reading a transaction 125
 7-bit addressed slave data transfer format 120,
 121, 122
 7-bit receive data transfer format 125
 baud high and low byte registers 132, 133, 135

C status register 129, 233
 control register definitions 128
 controller 115
 controller signals 10
 interrupts 117
 operation 116
 SDA and SCL signals 117
 stop and start conditions 119
 I2CBRH register 132, 133, 135, 234
 I2CBRL register 133, 234
 I2CCTL register 131, 233
 I2CDATA register 129, 233
 I2CSTAT register 129, 233
 IM 201
 immediate data 201
 immediate operand prefix 202
 INC 204
 increment 204
 increment word 204
 INCW 204
 indexed 202
 indirect address prefix 202
 indirect register 201
 indirect register pair 201
 indirect working register 201
 indirect working register pair 201
 infrared encoder/decoder (IrDA) 97
 instruction set, ez8 CPU 199
 instructions
 ADC 204
 ADCX 204
 ADD 204
 ADDX 204
 AND 207
 ANDX 207
 arithmetic 204
 BCLR 205
 BIT 205
 bit manipulation 205
 block transfer 205
 BRK 207
 BSET 205
 BSWAP 205, 208
 BTJ 207

IrDA

- architecture 97
- block diagram 97
- control register definitions 100
- operation 97
- receiving data 99
- transmitting data 98

IRET 207

IRQ0 enable high and low bit registers 48

IRQ1 enable high and low bit registers 49

IRQ2 enable high and low bit registers 51

IRR 201

Irr 201

J

JP 207

jump, conditional, relative, and relative conditional 207

L

LD 206

LDC 206

LDCI 205, 206

LDE 206

LDEI 205, 206

LDX 206

LEA 206

load 206

load constant 205

load constant to/from program memory 206

load constant with auto-increment addresses 206

load effective address 206

load external data 206

load external data to/from data memory and auto-increment addresses 205

load external to/from data memory and auto-increment addresses 206

load instructions 206

load using extended addressing 206

logical AND 207

logical AND/extended addressing 207

logical exclusive OR 207

logical exclusive OR/extended addressing 207

logical instructions 207

logical OR 207

logical OR/extended addressing 207

low power modes 27

M

master interrupt enable 42

master-in, slave-out and-in 103

memory

program 15

MISO 103

mode

capture 68

capture/compare 69

continuous 68

counter 68

gated 69

one-shot 68

PWM 68

modes 68

MOSI 103

MULT 205

multiply 205

multiprocessor mode, UART 83

N

NOP (no operation) 206

not acknowledge interrupt 117

notation

b 201

cc 201

DA 201

ER 201

IM 201

IR 201

Ir 201

IRR 201

Irr 201

p 201

R 201

r 201