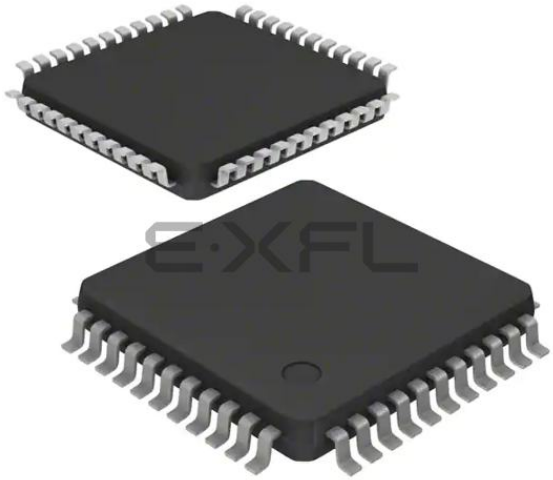

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

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

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, IrDA, LINbus, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, Temp Sensor, WDT
Number of I/O	37
Program Memory Size	24KB (24K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f2480an020eg

need to use this on-chip Program RAM to shadow Interrupt Service Routines (ISR). For details, see the PRAM_M section on page 278.

3.2. Program Memory

The eZ8 CPU supports 64KB of Program Memory address space. The F1680 Series MCU contains 8KB to 24KB of on-chip Flash memory in the Program Memory address space, depending on the device.

In addition, the F1680 Series MCU contains up to 1KB of on-chip Program RAM. The Program RAM is mapped in the Program Memory address space beyond the on-chip Flash memory. The Program RAM is entirely under user control and is meant to store interrupt service routines of high-frequency interrupts. Since interrupts bring the CPU out of low-power mode, it is important to ensure that interrupts that occur very often use as low a current as possible. For battery operated systems, Program RAM based handling of high-frequency interrupts provides power savings by keeping the Flash block disabled. Program RAM (PRAM) is optimized for low-current operation and can be easily boot-strapped with interrupt code at power up.

Reading from Program Memory addresses present outside the available Flash memory and PRAM addresses returns FFH. Writing to these unimplemented Program Memory addresses produces no effect. Table 6 describes the Program Memory maps for the F1680 Series MCU.

Table 6. F1680 Series MCU Program Memory Maps

Program Memory Address (Hex)	Function
Z8F2480 Device	
0000–0001	Flash option bits
0002–0003	Reset vector
0004–0005	WDT interrupt vector
0006–0007	Illegal instruction trap
0008–0037	Interrupt vectors*
0038–003D	Oscillator fail traps*
003E–5FFF	Program Flash
E000–E3FF	1KB PRAM

Note: *See Table 36 on page 69 for a list of interrupt vectors and traps.

Table 8. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)[1]	Page #
F20	Timer 0 PWM1 High Byte	T0PWM1H	00	111
F21	Timer 0 PWM1 Low Byte	T0PWM1L	00	111
F22	Timer 0 Control 2	T0CTL2	00	117
F23	Timer 0 Status	T0STA	00	118
F2C	Timer 0 Noise Filter Control	T0NFC	00	119
Timer 1				
F08	Timer 1 High Byte	T1H	00	109
F09	Timer 1 Low Byte	T1L	01	109
F0A	Timer 1 Reload High Byte	T1RH	FF	110
F0B	Timer 1 Reload Low Byte	T1RL	FF	110
F0C	Timer 1 PWM0 High Byte	T1PWM0H	00	110
F0D	Timer 1 PWM0 Low Byte	T1PWM0L	00	111
F0E	Timer 1 Control 0	T1CTL0	00	112
F0F	Timer 1 Control 1	T1CTL1	00	113
F24	Timer 1 PWM1 High Byte	T1PWM1H	00	111
F25	Timer 1 PWM1 Low Byte	T1PWM1L	00	111
F26	Timer 1 Control 2	T1CTL2	00	117
F27	Timer 1 Status	T1STA	00	118
F2D	Timer 1 Noise Filter Control	T1NFC	00	119
Timer 2				
F10	Timer 2 High Byte	T2H	00	109
F11	Timer 2 Low Byte	T2L	01	110
F12	Timer 2 Reload High Byte	T2RH	FF	110
F13	Timer 2 Reload Low Byte	T2RL	FF	110
F14	Timer 2 PWM0 High Byte	T2PWM0H	00	110
F15	Timer 2 PWM0 Low Byte	T2PWM0L	00	111
F16	Timer 2 Control 0	T2CTL0	00	112

Notes:
1. XX=Undefined.
2. The Reserved space can be configured as General-Purpose Register File RAM depending on the user option bits (see the User Option Bits chapter on page 277) and the on-chip PRAM size (see the Ordering Information chapter on page 372). If the PRAM is programmed as General-Purpose Register File RAM on Reserved space, the starting address always begins immediately after the end of General-Purpose Register File RAM.

5.2.4. External Reset Input

The $\overline{\text{RESET}}$ pin has a Schmitt-triggered input and an internal pull-up resistor. When the $\overline{\text{RESET}}$ pin is asserted for a minimum of four system clock cycles, the device progresses through the System Reset sequence. Because of the possible asynchronicity of the system clock and reset signals, the required reset duration can be as short as three clock periods and as long as four. A reset pulse three clock cycles in duration might trigger a Reset; a pulse four cycles in duration always triggers a Reset.

While the $\overline{\text{RESET}}$ input pin is asserted Low, the F1680 Series MCU remains 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 on the system clock rising edge following $\overline{\text{RESET}}$ pin deassertion. Following a System Reset initiated by the external $\overline{\text{RESET}}$ pin, the EXT status bit in the RSTSTAT Register is set to 1.

5.2.5. External Reset Indicator

During System Reset or when enabled by the GPIO logic (see the Port A–E Control Registers section on page 60), the $\overline{\text{RESET}}$ pin functions as an open-drain (active Low) reset mode indicator in addition to the input functionality. This Reset output feature allows the F1680 Series MCU to reset other components to which it is connected, even if that reset is caused by internal sources such as POR, VBO, or WDT events.

After an internal Reset event occurs, the internal circuitry begins driving the $\overline{\text{RESET}}$ pin Low. The $\overline{\text{RESET}}$ pin is held Low by the internal circuitry until the appropriate delay listed in Table 9 on page 32 has elapsed.

5.2.6. On-Chip Debugger Initiated Reset

A POR can be 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.

5.3. Stop Mode Recovery

STOP Mode is entered by execution of a stop instruction by the eZ8 CPU. For detailed STOP Mode information, see the Low-Power Modes section on page 42. During Stop Mode Recovery, the CPU is held in reset for 4 IPO cycles.

Stop Mode Recovery does not affect On-chip registers other than the Reset Status (RSTSTAT) register and the Oscillator Control Register (OSCCTL). After any Stop Mode Recovery, the IPO is enabled and selected as the system clock. If another system clock source is required or IPO disabling is required, the Stop Mode Recovery code must

- Configure the timer for DEMODULATION Mode. Setting the mode also involves writing to the TMODEHI bit in the TxCTL0 Register
- Set the prescale value
- Set the TPOL bit to set the Capture edge (rising or falling) for the Timer Input. This setting applies only if the TPOLHI bit in the TxCTL2 Register is not set

2. Write to the Timer Control 2 Register to:
   - Choose the timer clock source
   - Set the TPOLHI bit if the Capture is required on both edges of the input signal
3. Write to the Timer Control 0 Register to set the timer interrupt configuration field TICONFIG.
4. Write to the Timer High and Low Byte registers to set the starting count value (typically `0001H`).
5. Write to the Timer Reload High and Low Byte registers to set the reload value.
6. Clear the Timer TxPWM0 and TxPWM1 High and Low Byte registers to `0000H`.
7. If required, enable the noise filter and set the noise filter control by writing to the relevant bits in the Noise Filter Control Register.
8. If required, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers. By default, the timer interrupt will be generated for both input capture and reload events. If required, configure the timer interrupt to be generated only at the input Capture event or the Reload event by setting TICONFIG field of the Timer Control 0 Register.
9. Configure the associated GPIO port pin for the Timer Input alternate function.
10. Write to the Timer Control 1 Register to enable the timer. Counting will start on the occurrence of the first external input transition.

In DEMODULATION Mode, the elapsed time from timer start to Capture event can be calculated using the following equation:

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

For a given LIN-UART data rate, the integer baud rate divisor value is calculated using the following equation for LIN mode UART operation:

$$\text{UART Baud Rate Divisor Value (BRG)} = \text{Round}\left(\frac{\text{System Clock Frequency (Hz)}}{\text{UART Data Rate (bits/s)}}\right)$$

The baud rate error relative to the appropriate baud rate is calculated using the following equation:

$$\text{UART Baud Rate Error (\%)} = 100 \times \left(\frac{\text{Actual Data Rate} - \text{Desired Data Rate}}{\text{Desired Data Rate}}\right)$$

For reliable communication, the LIN-UART baud rate error must never exceed 5 percent. Tables 96 through 100 provide error data for popular baud rates and commonly-used crystal oscillator frequencies for normal UART modes of operation.

Table 96. LIN-UART Baud Rates, 20.0 MHz System Clock

Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	1	1250.0	0.00	9.60	130	9.62	0.16
625.0	2	625.0	0.00	4.80	260	4.81	0.16
250.0	5	250.0	0.00	2.40	521	2.399	–0.03
115.2	11	113.64	–1.19	1.20	1042	1.199	–0.03
57.6	22	56.82	–1.36	0.60	2083	0.60	0.02
38.4	33	37.88	–1.36	0.30	4167	0.299	–0.01
19.2	65	19.23	0.16				

Table 97. LIN-UART Baud Rates, 10.0 MHz System Clock

Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	N/A	N/A	N/A	9.60	65	9.62	0.16
625.0	1	625.0	0.00	4.80	130	4.81	0.16
250.0	3	208.33	–16.67	2.40	260	2.40	–0.03
115.2	5	125.0	8.51	1.20	521	1.20	–0.03
57.6	11	56.8	–1.36	0.60	1042	0.60	–0.03

Table 97. LIN-UART Baud Rates, 10.0 MHz System Clock (Continued)

Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
38.4	16	39.1	1.73	0.30	2083	0.30	0.2
19.2	33	18.9	0.16				

Table 98. LIN-UART Baud Rates, 5.5296 MHz System Clock

Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	N/A	N/A	N/A	9.60	36	9.60	0.00
625.0	N/A	N/A	N/A	4.80	72	4.80	0.00
250.0	1	345.6	38.24	2.40	144	2.40	0.00
115.2	3	115.2	0.00	1.20	288	1.20	0.00
57.6	6	57.6	0.00	0.60	576	0.60	0.00
38.4	9	38.4	0.00	0.30	1152	0.30	0.00
19.2	18	19.2	0.00				

Table 99. LIN-UART Baud Rates, 3.579545 MHz System Clock

Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)	Applicable Rate (kHz)	BRG Divisor (Decimal)	Actual Rate (kHz)	Error (%)
1250.0	N/A	N/A	N/A	9.60	23	9.73	1.32
625.0	N/A	N/A	N/A	4.80	47	4.76	–0.83
250.0	1	223.72	–10.51	2.40	93	2.41	0.23
115.2	2	111.9	–2.90	1.20	186	1.20	0.23
57.6	4	55.9	–2.90	0.60	373	0.60	–0.04
38.4	6	37.3	–2.90	0.30	746	0.30	–0.04
19.2	12	18.6	–2.90				

Bit Position	Value (H)	Description (Continued)
[5] REFEN	0	Select external reference.
	1	Select internal reference.
[4] ADCEN	0	ADC is disabled.
	1	ADC is enabled for normal use. This bit cannot change with bit 7 (start) at the same time.
[3:0] ANAIN		**Analog Input Select**
	0000	ANA0 input is selected for analog-to-digital conversion.
	0001	ANA1 input is selected for analog-to-digital conversion.
	0010	ANA2 input is selected for analog-to-digital conversion.
	0011	ANA3 input is selected for analog-to-digital conversion.
	0100	ANA4 input is selected for analog-to-digital conversion.
	0101	ANA5 input is selected for analog-to-digital conversion.
	0110	ANA6 input is selected for analog-to-digital conversion.
	0111	ANA7 input is selected for analog-to-digital conversion.
	1000	Hold LPO input nodes (ANA1 and ANA2) to ground.
	1001	Temperature Sensor.
	1100	Temperature Sensor output to ANA3 PAD.
	1101	vbg_chop signal output to ANA3 PAD.
	Others	Reserved.

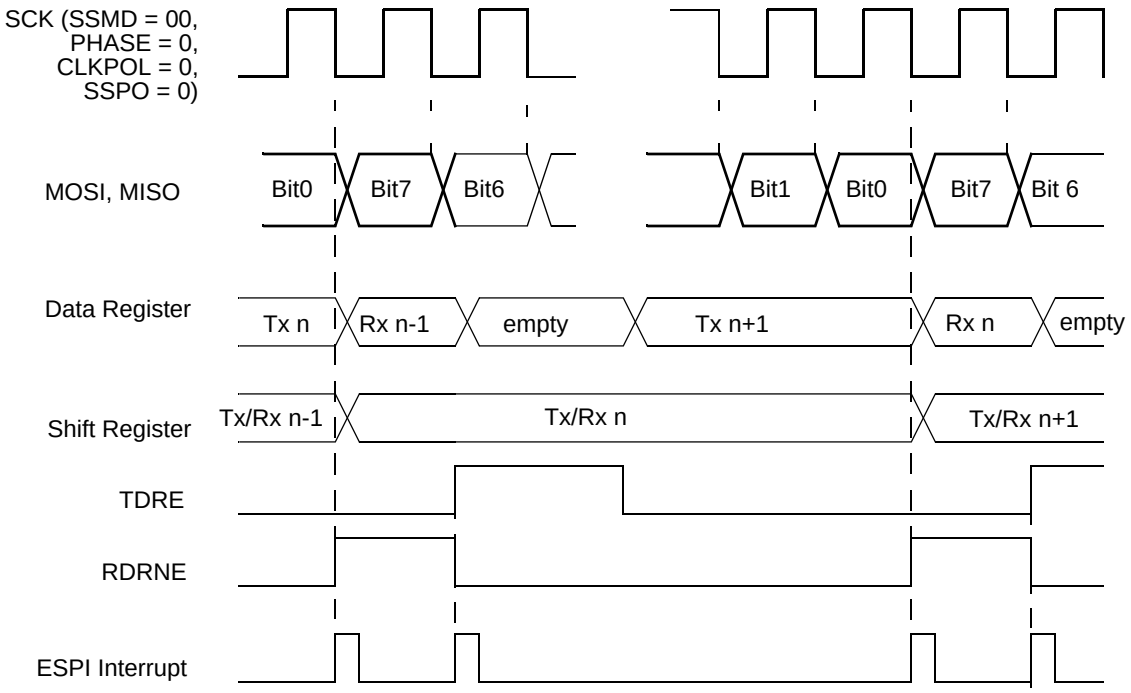

Figure 36. SPI Mode (SSMD = 00)

16.3.3.2. Synchronous Frame Sync Pulse Mode

This mode is selected by setting the SSMD field of the Mode Register to 10. This mode is typically used for continuous transfer of fixed length frames where the frames are delineated by a pulse of duration one SCK period. The SSV bit in the ESPI Transmit Data Command register does not control the $\overline{SS}$ pin directly in this mode. SSV must be set before or in sync with the first transmit data byte being written. The $\overline{SS}$ signal will assert 1 SCK cycle before the first data bit and will stop after 1 SCK period. SCK is active from the initial assertion of $\overline{SS}$ until the transaction end due to lack of transmit data.

The transaction is terminated by the Master when it no longer has data to send. If TDRE=1 at the end of a character, the $\overline{SS}$ output will remain detached and SCK stops after the last bit is transferred. The TUND bit (transmit underrun) will assert in this case. After the transaction has completed, hardware will clear the SSV bit. Figure 37 displays a frame with synchronous frame sync pulse mode.

$\overline{SS}$ pin on the selected slave. Then, the active Master drives the clock and transmits data on the SCK and MOSI pins to the SCK and MOSI pins on the Slave (including those Slaves which are not enabled). The enabled slave drives data out its MISO pin to the MISO Master pin.

When the ESPI is configured as a Master in a Multi-Master SPI system, the $\overline{SS}$ pin must be configured as an input. The $\overline{SS}$ input signal on a device configured as a Master should remain High. If the $\overline{SS}$ signal on the active Master goes Low (indicating another Master is accessing this device as a Slave), a Collision error flag is set in the ESPI Status Register. The Slave select outputs on a Master in a Multi-Master system must come from GPIO pins.

16.3.4.3. SPI Slave Operation

The ESPI block is configured for SLAVE Mode operation by setting the MMEN bit = 0 in the ESPICTL register and setting the SSIO bit = 0 in the ESPIMODE register. The SSMD field of the ESPI Mode Register is set to 00 for SPI protocol mode. The PHASE, CLKPOL and WOR bits in the ESPICTL register and the NUMBITS field in the ESPIMODE register must be set to be consistent with the other SPI devices. Typically for an SPI Slave, SSPO = 0.

If the Slave has data to send to the Master, the data must be written to the Data Register before the transaction starts (first edge of SCK when $\overline{SS}$ is asserted). If the Data Register is not written prior to the Slave transaction, the MISO pin outputs all 1s.

Due to the delay resulting from synchronization of the $\overline{SS}$ and SCK input signals to the internal system clock, the maximum SCK baud rate that can be supported in SLAVE Mode is the system clock frequency divided by 4. This rate is controlled by the SPI Master. Figure 41 displays the ESPI configuration in SPI SLAVE Mode.

Table 109. ESPI Data Register (ESPIDATA)

Bits	7	6	5	4	3	2	1	0
Field	DATA							
Reset	X	X	X	X	X	X	X	X
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F60H							

Bit	Description
[7:0] DATA	**Data** Transmit and/or receive data. Writes to the ESPIDATA register load the Shift Register. Reads from the ESPIDATA register return the value of the Receive Data Register.

16.4.2. ESPI Transmit Data Command and Receive Data Buffer Control Register

The ESPI Transmit Data Command and Receive Data Buffer Control Register, shown in Table 110, provides control of the $\overline{SS}$ pin when it is configured as an output (MASTER Mode), clear receive data buffer function and flag. The CRDR, TEOF and SSV bits can be controlled by a bus write to this register.

Table 110. ESPI Transmit Data Command and Receive Data Buffer Control Register (ESPITDCR)

Bits	7	6	5	4	3	2	1	0
Field	CRDR	RDFLAG		Reserved			TEOF	SSV
Reset	0	00		0	0	0	0	0
R/W	R/W	R		R	R	R	R/W	R/W
Address	F61H							

Bit	Description
[7] CRDR	**Clear Receive Data Register** Writing 1 to this bit is used to clear all data in receive data buffer.
[6:5] RDFLAG	**Receive Data Buffer Flag** This bit is used to indicate how many bytes stored in receive buffer. 00 = 0 or 4 bytes (see RDRNE in the ESPI Status Register). 01 = 1 byte. 02 = 2 bytes. 03 = 3 bytes.
[4:2]	**Reserved** These bits are reserved and must be programmed to 000.

1. The software initializes the MODE field in the I^2C Mode Register for MASTER/SLAVE Mode with 7- or 10-bit addressing (the I^2C bus protocol allows the mixing of slave address types). The MODE field selects the address width for this mode when addressed as a slave (but not for the remote slave). The software asserts the IEN bit in the I^2C Control Register.
2. The software writes `11110b`, followed by the two most-significant address bits and a 0 (write) to the I^2C Data Register.
3. The software asserts the start bit of the I^2C Control Register.
4. The I^2C controller sends a start condition.
5. The I^2C controller loads the I^2C Shift Register with the contents of the I^2C Data Register.
6. After the first bit has been shifted out, a transmit interrupt is asserted.
7. The software responds by writing the least significant eight bits of address to the I^2C Data Register.
8. The I^2C controller completes shifting of the first address byte.
9. The I^2C slave sends an Acknowledge by pulling the SDA signal Low during the next High period of SCL.

   If the slave does not acknowledge the address byte, the I^2C controller sets the NCKI bit in the I^2C Status Register, sets the ACKV bit and clears the ACK bit in the I^2C State Register. The software responds to the Not Acknowledge interrupt by setting the stop bit and clearing the TXI bit. The I^2C controller flushes the Transmit Data Register, sends the stop condition on the bus and clears the stop and NCKI bits. The transaction is complete and the following steps can be ignored.
10. The I^2C controller loads the I^2C Shift Register with the contents of the I^2C Data Register (the lower byte of the 10-bit address).
11. The I^2C controller shifts out the next eight bits of the address. After the first bit shifts, the I^2C controller generates a transmit interrupt.
12. The software responds by setting the start bit of the I^2C Control Register to generate a repeated start condition.
13. The software writes `11110b`, followed by the 2-bit slave address and a 1 (Read) to the I^2C Data Register.
14. If the user chooses to read only one byte, the software responds by setting the NAK bit of the I^2C Control Register.
15. After the I^2C controller shifts out the address bits listed in Step 9 (the second address transfer), the I^2C slave sends an Acknowledge by pulling the SDA signal Low during the next High period of SCL.

    If the slave does not acknowledge the address byte, the I^2C controller sets the NCKI bit in the I^2C Status Register, sets the ACKV bit and clears the ACK bit in the I^2C

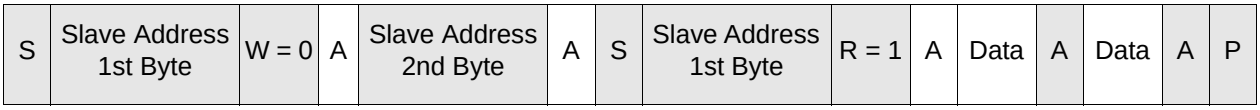

Figure 50. Data Transfer Format—Slave Transmit Transaction with 10-Bit Address

1. The software configures the controller for operation as a slave in 10-bit addressing mode.
   a. Initialize the MODE field in the I^2C Mode Register for either SLAVE ONLY mode or MASTER/SLAVE Mode with 10-bit addressing.
   b. Optionally set the GCE bit.
   c. Initialize the SLA[7:0] bits in the I2CSLVAD Register and SLA[9:8] in the I^2C MODE Register.
   d. Set IEN = 1 and NAK = 0 in the I^2C Control Register.
2. The Master initiates a transfer by sending the first address byte. The SLAVE Mode I^2C controller recognizes the start of a 10-bit address with a match to SLA[9:8] and detects R/W bit = 0 (a Write from the master to the slave). The I^2C controller acknowledges indicating it is available to accept the transaction.
3. The Master sends the second address byte. The SLAVE Mode I^2C controller compares the second address byte with the value in SLA[7:0]. If there is a match, the SAM bit in the I2CISTAT Register is set = 1, causing a slave address match interrupt. The RD bit is set = 0, indicating a write to the slave. If a match occurs, the I^2C controller acknowledges on the I^2C bus, indicating it is available to accept the data.
4. The software responds to the slave address match interrupt by reading the I2CISTAT Register, which clears the SAM bit. Because the RD bit = 0, no further action is required.
5. The Master sees the Acknowledge and sends a restart instruction, followed by the first address byte with R/W set to 1. The SLAVE Mode I^2C controller recognizes the restart instruction followed by the first address byte with a match to SLA[9:8] and detects R/W = 1 (the master reads from the slave). The slave I^2C controller sets the SAM bit in the I2CISTAT Register which causes the slave address match interrupt. The RD bit is set = 1. The SLAVE Mode I^2C controller acknowledges on the bus.
6. The software responds to the interrupt by reading the I2CISTAT Register clearing the SAM bit. The software loads the initial data byte into the I2CDATA Register and sets the TXI bit in the I2CCTL Register.
7. The Master starts the data transfer by asserting SCL Low. After the I^2C controller has data available to transmit, the SCL is released and the master proceeds to shift the first data byte.

```
ld R1, #%60
ldc R2, @RR0 ; R2 now contains the calibration value
```

21.2. Flash Option Bit Control Register Definitions

This section defines the features of the following Flash Option Bit Control registers.

User Option Bits: see page 278

Trim Bit Data Option Bits: see page 281

Trim Bit Address Option Bits: see page 281

Trim Bit Address Space: see page 282

Zilog Calibration Option Bits: see page 289

21.2.1. User Option Bits

The first two bytes of Flash program memory, at addresses 0000H and 0001H, are reserved for the user-programmable Flash option bits, as shown in Tables 140 and 141.

Table 140. Flash Option Bits at Program Memory Address 0000H

Bits	7	6	5	4	3	2	1	0
Field	WDT_RES	WDT_AO	OSC_SEL[1:0]		VBO_AO	FRP	PRAM_M	FWP
Reset	U	U	U	U	U	U	U	U
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	Program Memory 0000H							

Note: U = Unchanged by Reset. R/W = Read/Write.

Bit	Description
[7] 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 System Reset. This setting is the default for unprogrammed (erased) Flash.
[6] WDT_AO	**Watchdog Timer Always ON** 0 = Watchdog Timer is automatically enabled upon application of system power. Watchdog Timer cannot be disabled. 1 = Watchdog Timer is enabled upon execution of the WDT instruction. After it is enabled, the Watchdog Timer can only be disabled by a Reset or Stop Mode Recovery. This setting is the default for unprogrammed (erased) Flash.

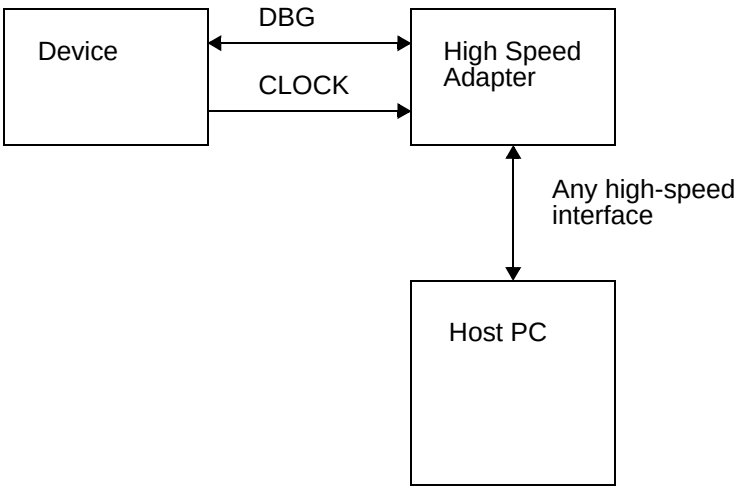

Figure 60. Synchronous Operation

23.2.5. OCD Serial Errors

The On-Chip Debugger can detect any of the following error conditions on the DBG pin:

- Serial Break (a minimum of ten continuous bits Low)
- Framing Error (received stop bit is Low)
- Transmit Collision (OCD and host simultaneous transmission detected by the OCD)

When the OCD detects one of these errors, it aborts any command currently in progress, transmits a Serial Break 4096 system clock cycles long back to the host and resets the Autobaud Detector/Generator. A Framing Error or Transmit Collision can be caused by the host sending a Serial Break to the OCD. Because of the open-drain nature of the interface, returning a Serial Break back to the host only extends the length of the Serial Break if the host releases the Serial Break early.

The host transmits a Serial Break on the DBG pin when first connecting to theZ8 Encore! XP F1680 Series device or when recovering from an error. A Serial Break from the host resets the Autobaud Generator/Detector but does not reset the OCD Control Register. A Serial Break leaves the device in DEBUG mode if that is the current mode. The OCD is held in Reset until the end of the Serial Break when the DBG pin returns High. Because of the open-drain nature of the DBG pin, the host can send a Serial Break to the OCD even if the OCD is transmitting a character.

Table 188. Absolute Maximum Ratings* (Continued)

Parameter	Min	Max	Units	Notes
44-Pin QFN Maximum Ratings at –40°C to 70°C				
Total power dissipation		750	mW	
Maximum current into V_{DD} or out of V_{SS}		200	mA	
44-Pin QFN Maximum Ratings at 70°C to 105°C				
Total power dissipation		295	mW	
Maximum current into V_{DD} or out of V_{SS}		83	mA	
44-pin LQFP Maximum Ratings at –40°C to 70°C				
Total power dissipation		750	mW	
Maximum current into V_{DD} or out of V_{SS}		200	mA	
44-pin LQFP Maximum Ratings at 70°C to 105°C				
Total power dissipation		410	mW	
Maximum current into V_{DD} or out of V_{SS}		114	mA	

Notes:
*Operating temperature is specified in DC Characteristics.
1. This voltage applies to all pins except the following: V_{DD}, AV_{DD}.

29.2. DC Characteristics

Table 189 lists the DC characteristics of the Z8 Encore! XP F1680 Series products. All voltages are referenced to V_{SS}, which is the primary system ground.

Table 189. DC Characteristics

		T_A = 0°C to +70°C T_A = –40°C to +105°C				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
V_{DD}	Supply Voltage	1.8	–	3.6	V	
V_{IL1}	Low Level Input Voltage	–0.3	–	0.3*V_{DD}	V	For all input pins except $\overline{\text{RESET}}$, DBG, XIN
V_{IL2}	Low Level Input Voltage	–0.3	–	0.2*V_{DD}	V	For $\overline{\text{RESET}}$, DBG, XIN
V_{IH1}	High Level Input Voltage	0.7*V_{DD}	–	5.5	V	Port A, B, C, D and E pins (Digital inputs)

Notes:
1. This condition excludes all pins that have on-chip pull-ups, when driven Low.
2. These values are provided for design guidance only and are not tested in production.

Table 189. DC Characteristics (Continued)

		T_A = 0°C to +70°C T_A = –40°C to +105°C				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
V_{IH2}	High Level Input Voltage	0.7*V_{DD}	–	V_{DD}+0.3	V	Ports B and C (Analog)
V_{OL1}	Low Level Output Voltage	–	–	0.4	V	I_{OL} = 2 mA; V_{DD} = 3.0V High Output Drive disabled.
V_{OH1}	High Level Output Voltage	V_{DD}–0.5	–	–	V	I_{OH} = –2 mA; V_{DD} = 3.0V High Output Drive disabled.
V_{OL2}	Low Level Output Voltage	–	–	0.6	V	I_{OL} = 20 mA; V_{DD} = 3.3V High Output Drive enabled.
V_{OH2}	High Level Output Voltage	V_{DD}–0.5	–	–	V	I_{OH} = –20 mA; V_{DD} = 3.3V High Output Drive enabled.
I_{IL}	Input Leakage Current	–5	–	+5	µA	V_{DD} = 3.6V; V_{IN} = V_{DD} or V_{SS}[1]
I_{TL}	Tristate Leakage Current	–5	–	+5	µA	V_{DD} = 3.6V
I_{LED}	Controlled LED Current Drive	1.5	3	4.5	mA	{AFS2,AFS1} = {0,0}, V_{DD} = 3.3V
		2.8	7	10.5	mA	{AFS2,AFS1} = {0,1}, V_{DD} = 3.3V
		7.8	13	19.5	mA	{AFS2,AFS1} = {1,0}, V_{DD} = 3.3V
		12	20	30	mA	{AFS2,AFS1} = {1,1}, V_{DD} = 3.3V
C_{PAD}	GPIO Port Pad Capacitance	–	8.0[2]	–	pF	TBD
C_{XIN}	XIN Pad Capacitance	–	8.0[2]	–	pF	TBD
C_{XOUT}	XOUT Pad Capacitance	–	9.5[2]	–	pF	TBD
I_{PU}	Weak Pull-up Current	30	100	350	µA	V_{DD} = 3.0V–3.6V

Notes:
1. This condition excludes all pins that have on-chip pull-ups, when driven Low.
2. These values are provided for design guidance only and are not tested in production.

V

W

X

Z