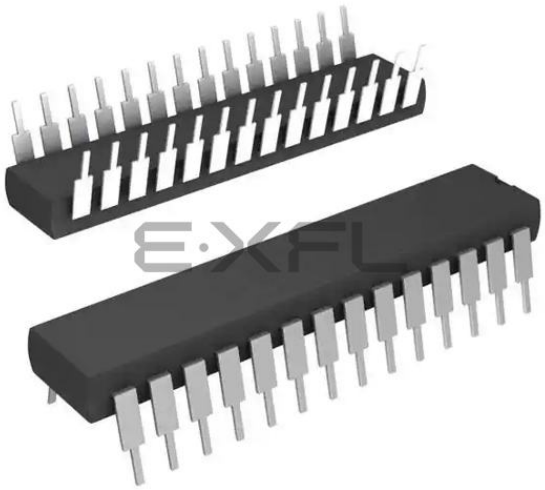


Welcome to E-XFL.COM

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	23
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	0°C ~ 70°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/z8f2480pj020sg

List of Figures

1.3. Block Diagram

Figure 1 displays the architecture of the F1680 Series MCU.

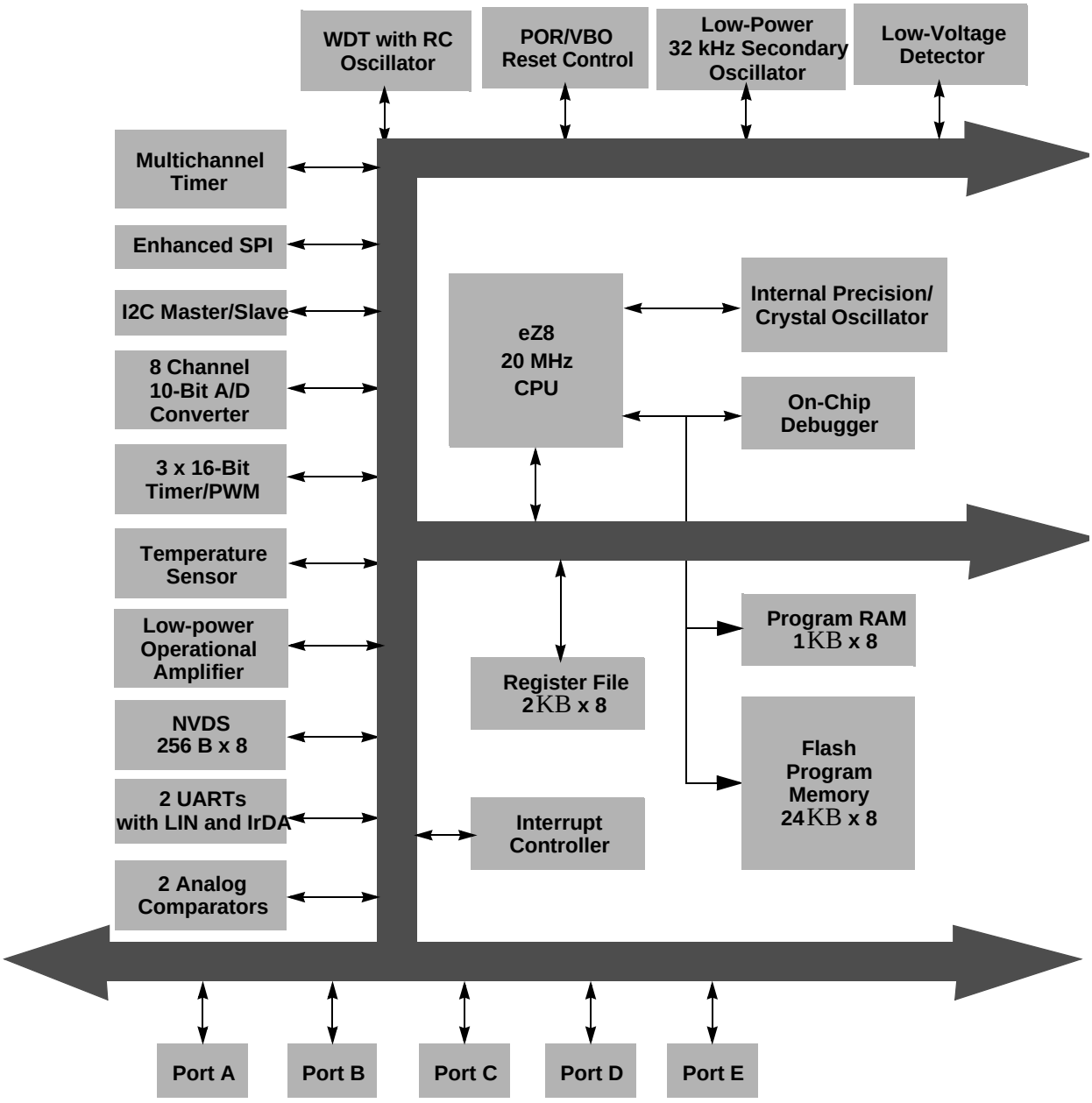


Figure 1. F1680 Series MCU Block Diagram

1.4.3. Non-Volatile Data Storage

Non-Volatile Data Storage (NVDS) is a hybrid hardware/software scheme to implement byte-programmable data memory and is capable of over 100,000 write cycles.

1.4.4. Internal Precision Oscillator

The internal precision oscillator (IPO) is a trimmable clock source which requires no external components. You can select IPO frequency from one of eight frequencies (43.2kHz to 11.0592MHz) and is available with factory-trimmed calibration data.

1.4.5. Crystal Oscillator

The crystal oscillator circuit provides highly accurate clock frequencies using an external crystal, ceramic resonator, or RC network.

1.4.6. Secondary Oscillator

The secondary oscillator is a low-power oscillator, which is optimized for use with a 32kHz watch crystal. It can be used as timer/counter clock source in any mode.

1.4.7. 10-Bit Analog-to-Digital Converter

The Analog-to-Digital Converter (ADC) converts an analog input signal to a 10-bit binary number. The ADC supports up to eight analog input sources multiplexed with GPIO ports.

1.4.8. Low-Power Operational Amplifier

The low-power operational amplifier (LPO) is a general-purpose operational amplifier primarily targeted for current sense applications. The LPO output can be internally routed to the ADC or externally to a pin.

1.4.9. Analog Comparator

The analog comparator compares the signal at an input pin with either an internal programmable voltage reference or a second-input pin. The comparator output is used to either drive an output pin or to generate an interrupt.

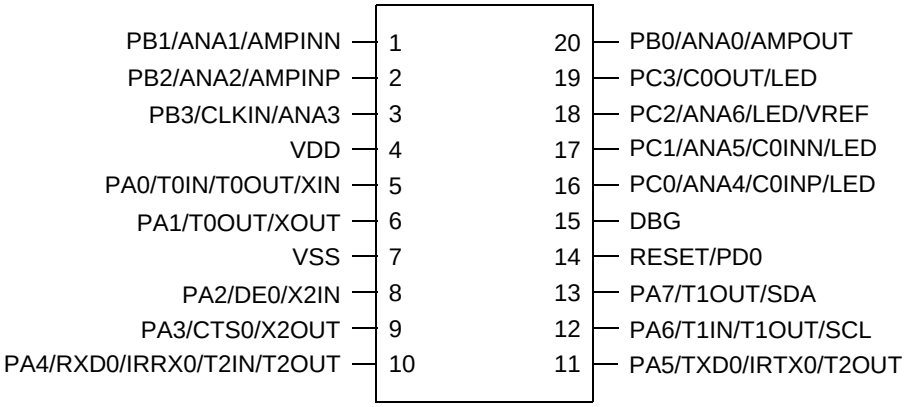


Figure 2. Z8F2480, Z8F1680 and Z8F0880 in 20-Pin SOIC, SSOP or PDIP Packages

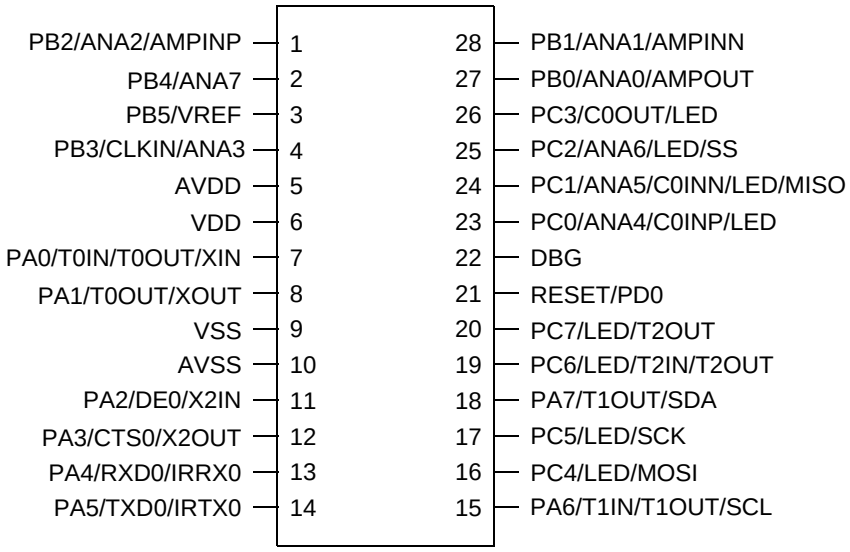
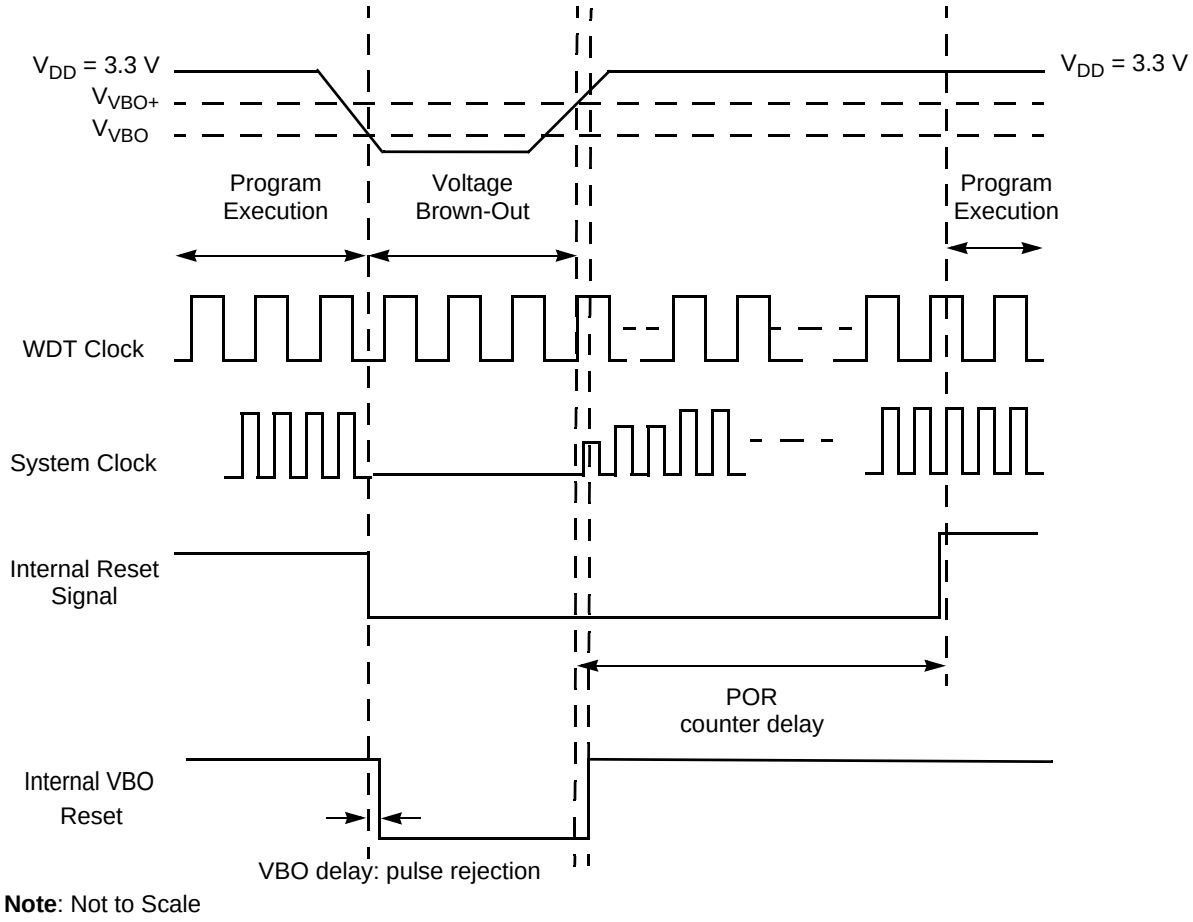


Figure 3. Z8F2480, Z8F1680 and Z8F0880 in 28-Pin SOIC, SSOP or PDIP Packages



Note: Not to Scale

Figure 8. Voltage Brown-Out Reset Operation

5.2.3. Watchdog Timer Reset

If the device is operating in NORMAL or STOP modes, the WDT initiates a System Reset at time-out if the WDT_RES Flash option bit is programmed to 1 (which is the unprogrammed state of the WDT_RES Flash option bit). If the bit is programmed to 0, it configures the WDT to cause an interrupt, not a System Reset at time-out. The WDT status bit in the Reset Status Register is set to signify that the reset was initiated by the WDT.

Table 60. Timer 0–2 PWM0 Low Byte Register (TxPWM0L)

Bit	7	6	5	4	3	2	1	0
Field	PWM0L							
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F05H, F0DH, F15H							

Bit	Description
[7:0] PWM0H, PWM0L	**Pulse Width Modulator 0 High and Low Bytes** These two bytes, {PWM0H[7:0], PWM0L[7:0]}, form a 16-bit value that is compared to the current 16-bit timer count. When a match occurs, the PWM output changes state. The PWM output value is set by the TPOL bit in the Timer Control 1 Register (TxCTL1). The TxPWM0H and TxPWM0L registers also store the 16-bit captured timer value when operating in CAPTURE, CAPTURE/COMPARE and DEMODULATION Modes.

9.3.4. Timer 0–2 PWM1 High and Low Byte Registers

The Timer 0–2 PWM1 High and Low Byte (TxPWM1H and TxPWM1L) registers, shown in Tables 61 and 62, store Capture values for DEMODULATION Mode.

Table 61. Timer 0-2 PWM1 High Byte Register (TxPWM1H)

Bit	7	6	5	4	3	2	1	0
Field	PWM1H							
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F20H, F24H, F28H							

Table 62. Timer 0–2 PWM1 Low Byte Register (TxPWM1L)

Bit	7	6	5	4	3	2	1	0
Field	PWM1L							
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	F21H, F25H, F29H							

Bit	Description
[7:0] PWM1H, PWM1L	**Pulse Width Modulator 1 High and Low Bytes** These two bytes, {PWM1H[7:0], PWM1L[7:0]}, store the 16-bit captured timer value for DEMODULATION Mode.

Bit	Description (Continued)
[6] (cont'd)	**PWM DUAL OUTPUT Mode** 0 = Timer Output is forced Low (0) and Timer Output Complement is forced High (1) when the timer is disabled. When enabled, the Timer Output is forced High (1) upon PWM count match and forced Low (0) upon Reload. When enabled, the Timer Output Complement is forced Low (0) upon PWM count match and forced High (1) upon Reload. The PWMD field in Timer Control 0 Register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to High (1). 1 = Timer Output is forced High (1) and Timer Output Complement is forced Low (0) when the timer is disabled. When enabled, the Timer Output is forced Low (0) upon PWM count match and forced High (1) upon Reload. When enabled, the Timer Output Complement is forced High (1) upon PWM count match and forced Low (0) upon Reload. The PWMD field in Timer Control 0 Register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to Low (0). **CAPTURE RESTART Mode** 0 = Count is captured on the rising edge of the Timer Input signal. 1 = Count is captured on the falling edge of the Timer Input signal. **COMPARATOR COUNTER Mode** When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer reload. **TRIGGERED ONE-SHOT Mode** 0 = Timer counting is triggered on the rising edge of the Timer Input signal. 1 = Timer counting is triggered on the falling edge of the Timer Input signal. **DEMODULATION Mode** 0 = Timer counting is triggered on the rising edge of the Timer Input signal. The current count is captured into PWM0 High and Low byte registers on subsequent rising edges of the Timer Input signal. 1 = Timer counting is triggered on the falling edge of the Timer Input signal. The current count is captured into PWM1 High and Low byte registers on subsequent falling edges of the Timer Input signal. The above functionality applies only if TPOLHI bit in Timer Control 2 Register is 0. If TPOLHI bit is 1 then timer counting is triggered on any edge of the Timer Input signal and the current count is captured on both edges. The current count is captured into PWM0 registers on rising edges and PWM1 registers on falling edges of the Timer Input signal.

Table 77. Multi-Channel Timer Channel Status 1 Register (MCTCHS1)

Bit	7	6	5	4	3	2	1	0
Field	Reserved				CHDEF	CHCEF	CHBEF	CHAEF
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
Address	See note.							

Note: If a 01H is in the Subaddress Register, it is accessible through Subregister 1.

Bit	Description
[7:4]	Reserved; must be 0.
[3:0] CHyEF	**Channel y Event Flag** This bit indicates if a Capture/Compare event occurred for this channel. Software can use this bit to determine the channel(s) responsible for generating the Multi-Channel Timer channel interrupt. This event flag is cleared by writing a 1 to the bit. These bits will be set when an event occurs independent of the setting of the CHIEN bit. This bit is cleared when TEN=0 (TEN is the MSB of MCTCTL1). 0 = No Capture/Compare Event occurred for this channel. 1 = A Capture/Compare Event occurred for this channel.

Chapter 12. LIN-UART

The Local Interconnect Network Universal Asynchronous Receiver/Transmitter (LIN-UART) is a full-duplex communication channel capable of handling asynchronous data transfers in standard UART applications and providing LIN protocol support. The LIN-UART is a superset of the standard Z8 Encore!® UART, providing all its standard features, LIN protocol support and a digital noise filter.

LIN-UART includes the following features:

- 8-bit asynchronous data transfer
- Selectable even- and odd-parity generation and checking
- Option of 1 or 2 stop bits
- Selectable MULTIPROCESSOR (9-bit) Mode with three configurable interrupt schemes
- Separate transmit and receive interrupts
- Framing, parity, overrun and break detection
- 16-bit baud rate generator (BRG) which can function as a general purpose timer with interrupt
- Driver Enable output for external bus transceivers
- LIN protocol support for both MASTER and SLAVE modes:
  - Break generation and detection
  - Selectable Slave Autobaud
  - Check Tx versus Rx data when sending
- Configuring digital-noise filter on Receive Data line

12.1. LIN-UART Architecture

The LIN-UART consists of three primary functional blocks: transmitter, receiver and baud-rate generator. The LIN-UART's transmitter and receiver function independently but use the same baud rate and data format. The basic UART operation is enhanced by the Noise Filter and IrDA blocks. Figure 19 displays the LIN-UART architecture.

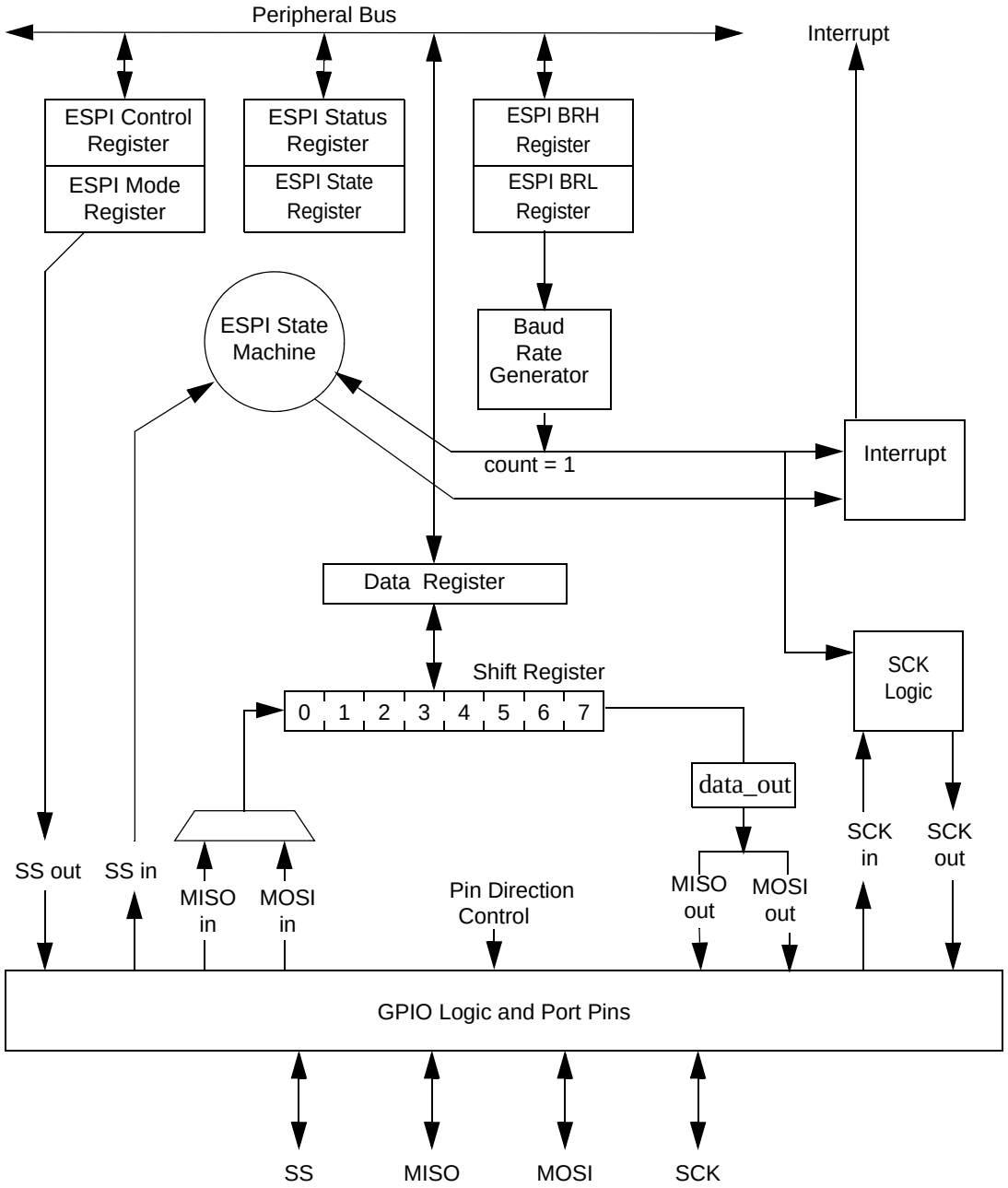


Figure 33. ESPI Block Diagram

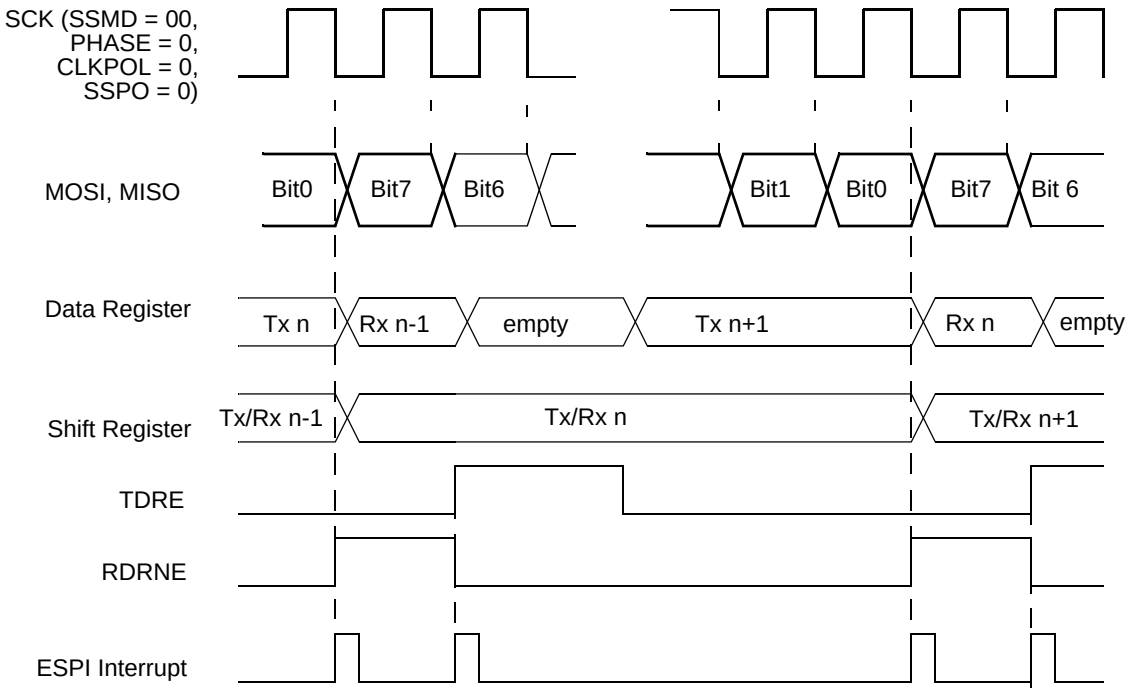


Figure 36. SPI Mode (SSMD = 00)

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.

GCE bit = 1 in the I2CMODE Register. The software checks the RD bit in the I2CISTAT Register to determine if the transaction is a Read or Write transaction. The General Call Address and STARTBYTE address are also distinguished by the RD bit. The General Call Address (GCA) bit of the I2CISTAT Register indicates whether the address match occurred on the unique slave address or the General Call/STARTBYTE address. The SAM bit clears automatically when the I2CISTAT Register is read.

If configured via the MODE[1:0] field of the I^2C Mode Register for 7-bit slave addressing, the most significant 7 bits of the first byte of the transaction are compared against the SLA[6:0] bits of the Slave Address Register. If configured for 10-bit slave addressing, the first byte of the transaction is compared against {11110,SLA[9:8], R/$\overline{W}$} and the second byte is compared against SLA[7:0].

17.2.2.4. Arbitration Lost Interrupts

Arbitration Lost interrupts (ARBLST bit = 1 in I2CISTAT) occur when the I^2C controller is in MASTER Mode and loses arbitration (outputs 1 on SDA and receives 0 on SDA). The I^2C controller switches to SLAVE Mode when this instance occurs. This bit clears automatically when the I2CISTAT Register is read.

17.2.2.5. Stop/Restart Interrupts

A Stop/Restart event interrupt (SPRS bit = 1 in I2CISTAT) occurs when the I^2C controller is operating in SLAVE Mode and a stop or restart condition is received, indicating the end of the transaction. The RSTR bit in the I^2C State Register indicates whether the bit is set due to a stop or restart condition. When a restart occurs, a new transaction by the same master is expected to follow. This bit is cleared automatically when the I2CISTAT Register is read. The Stop/Restart interrupt occurs only on a selected (address match) slave.

17.2.2.6. Not Acknowledge Interrupts

Not Acknowledge interrupts (NCKI bit = 1 in I2CISTAT) occur in MASTER Mode when Not Acknowledge is received or sent by the I^2C controller and the start or stop bit is not set in the I^2C Control Register. In MASTER Mode, the Not Acknowledge interrupt clears by setting the start or stop bit. When this interrupt occurs in MASTER Mode, the I^2C controller waits until it is cleared before performing any action. In SLAVE Mode, the Not Acknowledge interrupt occurs when a Not Acknowledge is received in response to data sent. The NCKI bit clears in SLAVE Mode when software reads the I2CISTAT Register.

17.2.2.7. General Purpose Timer Interrupt from Baud Rate Generator

If the I^2C controller is disabled (IEN bit in the I2CCTL Register = 0) and the BIRQ bit in the I2CCTL Register = 1, an interrupt is generated when the baud rate generator (BRG) counts down to 1. The baud rate generator reloads and continues counting, providing a periodic interrupt. None of the bits in the I2CISTAT Register are set, allowing the BRG in the I^2C Controller to be used as a general-purpose timer when the I^2C Controller is disabled.

17.3.3. I²C Control Register

The I²C Control Register, shown in Table 121, enables and configures I²C operation.

Note: The R/W1 bit can be set (written to 1) when IEN = 1, but cannot be cleared (written to 0).

Table 121. I²C Control Register (I2CCTL)

Bits	7	6	5	4	3	2	1	0
Field	IEN	START	STOP	BIRQ	TXI	NAK	FLUSH	FILTEN
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W1	R/W1	R/W	R/W	R/W1	W	R/W
Address	F52H							

Bit	Description
[7] IEN	**I²C Enable** This bit enables the I²C controller.
[6] START	**Send Start Condition** When set, this bit causes the I²C controller (when configured as the master) to send a start condition. After it is asserted, this bit is cleared by the I²C controller after it sends the start condition or by deasserting the IEN bit. If this bit is 1, it cannot be cleared by writing to the bit. After this bit is set, a start condition is sent if there is data in the I2CDATA or I²C Shift Register. If there is no data in one of these registers, the I²C controller waits until data is loaded. If this bit is set while the I²C controller is shifting out data, it generates a restart condition after the byte shifts and the Acknowledge phase completes. If the stop bit is also set, it waits until the stop condition is sent before the start condition. If start is set while a SLAVE Mode transaction is underway to this device, the start bit will be cleared and ARBLST bit in the Interrupt Status Register will be set.
[5] STOP	**Send Stop Condition** When set, this bit causes the I²C controller (when configured as the master) to send the stop condition after the byte in the I²C Shift Register has completed transmission or after a byte is received in a receive operation. When set, this bit is reset by the I²C controller after a stop condition has been sent or by deasserting the IEN bit. If this bit is 1, it cannot be cleared to 0 by writing to the register. If stop is set while a SLAVE Mode transaction is underway, the stop bit is cleared by hardware.
[4] BIRQ	**Baud Rate Generator Interrupt Request** This bit is ignored when the I²C controller is enabled. If this bit is set = 1 when the I²C controller is disabled (IEN = 0), the baud rate generator is used as an additional timer causing an interrupt to occur every time the baud rate generator counts down to one. The baud rate generator runs continuously in this mode, generating periodic interrupts.

23.2. Operation of the On-Chip Debugger Interface

The On-Chip Debugger (OCD) uses the DBG pin for communication with an external host. This one-pin interface is a bidirectional open-drain interface that transmits and receives data. Data transmission is half-duplex, in that transmit and receive cannot occur simultaneously. The serial data on the DBG pin is sent using the standard asynchronous data format defined in RS-232. This pin interfaces the Z8 Encore! XP F1680 Series device to the serial port of a host PC using minimal external hardware. Figure 56 displays the connections between the debug connector and the Z8 Encore! microcontroller. Two different methods for connecting the DBG pin to an RS-232 interface are depicted in Figures 57 and and 58.

! Caution: For operation of the Z8 Encore! XP F1680 Series device, 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. The DBG pin should always be connected to V_{DD} through an external pull-up resistor.

The Serial Smart Cable (SSC) does not work with the F1680 device series because it does not fully support the silicon OCD. During external clock switching, the OCD sends a break command to the SSC. This causes the SSC to disconnect from the target and terminate the debug session. You must then reconnect to the target again. Use the Opto-Isolated USB, USB, or Ethernet Smart Cables when using in conjunction with ZDSII.

23.2.6. Automatic Reset

The Z8 Encore! XP F1680 Series devices have the capability to switch clock sources during operation. If the Autobaud is set and the clock source is switched, the Autobaud value becomes invalid. A new Autobaud value must be configured with the new clock frequency.

The oscillator control logic has clock switch detection. If a clock switch is detected and the Autobaud is set, the device will automatically send a Serial Break for 4096 clocks. This will reset the Autobaud and indicate to the host that a new Autobaud character should be sent.

23.2.7. Transmit Flow Control

Transmit flow control is implemented by the use of a remote start bit. When transmit flow control is enabled, the transmitter will wait for the remote host to send the start bit. Transmit flow control is useful in applications where receive overruns can occur.

The remote host can transmit a remote start bit by sending the character FFH. The transmitter will append its data after the start bit. Due to the *wire-and* nature of the open drain bus, the start bit sent by the remote host and the data bits sent by the Z8 Encore! XP F1680 Series device appear as one character; see Figure 61.

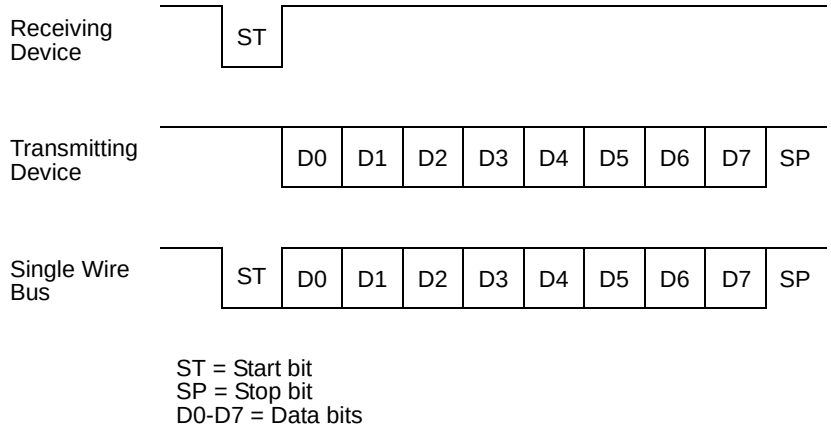


Figure 61. Start Bit Flow Control

23.2.8. Breakpoints

Execution breakpoints are generated using the BRK instruction (op code 00H). When the eZ8 CPU decodes a BRK instruction, it signals the On-Chip Debugger. If breakpoints are enabled, the OCD idles the eZ8 CPU and enters DEBUG mode. If breakpoints are not

```
DBG ← Size[7:0]
DBG → 1-256 data bytes
```

Write Program Memory (0AH). The Write Program Memory command writes data to Program memory. This command is equivalent to the LDC and LDCI instructions. Data can be written 1–65536 bytes at a time (65536 bytes can be written by setting size to 0). The on-chip Flash Controller must be written and unlocked for the programming operation to occur. If the Flash Controller is not unlocked, the data is discarded. If the device is not in DEBUG mode or if the Read Protect option bit is enabled, the data is discarded.

```
DBG ← 0AH
DBG ← Program Memory Address[15:8]
DBG ← Program Memory Address[7:0]
DBG ← Size[15:8]
DBG ← Size[7:0]
DBG ← 1-65536 data bytes
```

Read Program Memory (0BH). The Read Program Memory command reads data from Program memory. This command is equivalent to the LDC and LDCI instructions. Data can be read 1–65536 bytes at a time (65536 bytes can be read by setting size to 0). If the device is not in DEBUG mode or if the Read Protect option bit is enabled, this command returns FFH for the data.

```
DBG ← 0BH
DBG ← Program Memory Address[15:8]
DBG ← Program Memory Address[7:0]
DBG ← Size[15:8]
DBG ← Size[7:0]
DBG → 1-65536 data bytes
```

Write Data Memory (0CH). The Write Data Memory command writes data to Data Memory. This command is equivalent to the LDE and LDEI instructions. Data is written 1–65536 bytes at a time (65536 bytes can be written by setting size to 0). If the device is not in DEBUG mode or if the Read Protect option bit is enabled, the data is discarded.

```
DBG ← 0CH
DBG ← Data Memory Address[15:8]
DBG ← Data Memory Address[7:0]
DBG ← Size[15:8]
DBG ← Size[7:0]
DBG ← 1-65536 data bytes
```

Read Data Memory (0DH). The Read Data Memory command reads from Data Memory. This command is equivalent to the LDE and LDEI instructions. Data can be read 1 to 65536 bytes at a time (65536 bytes can be read by setting size to 0). If the device is not in DEBUG mode, this command returns FFH for the data.

```
DBG ← 0DH
DBG ← Data Memory Address[15:8]
DBG ← Data Memory Address[7:0]
DBG ← Size[15:8]
```

Table 172. Recommended Crystal Oscillator Specifications

Parameter	Value	Units	Comments
Frequency	20	MHz	
Resonance	Parallel		
Mode	Fundamental		
Series Resistance (R_S)	60	W	Maximum
Load Capacitance (C_L)	30	pF	Maximum
Shunt Capacitance (C_0)	7	pF	Maximum
Drive Level	1	mW	Maximum

25.3. Main Oscillator Operation with External RC Network

Figure 63 displays the recommended configuration for connection with an external resistor-capacitor (RC) network.

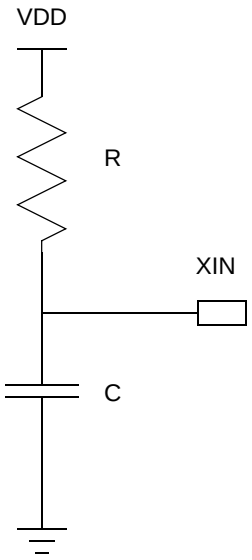


Figure 63. Connecting the On-Chip Oscillator to an External RC Network

Table 187. Op Code Map Abbreviations

Abbreviation	Description	Abbreviation	Description
b	Bit position	IRR	Indirect Register Pair
cc	Condition code	p	Polarity (0 or 1)
X	8-bit signed index or displacement	r	4-bit Working Register
DA	Destination address	R	8-bit register
ER	Extended Addressing register	r1, R1, Ir1, Irr1, IR1, rr1, RR1, IRR1, ER1	Destination address
IM	Immediate data value	r2, R2, Ir2, Irr2, IR2, rr2, RR2, IRR2, ER2	Source address
Ir	Indirect Working Register	RA	Relative
IR	Indirect register	rr	Working Register Pair
Irr	Indirect Working Register Pair	RR	Register Pair

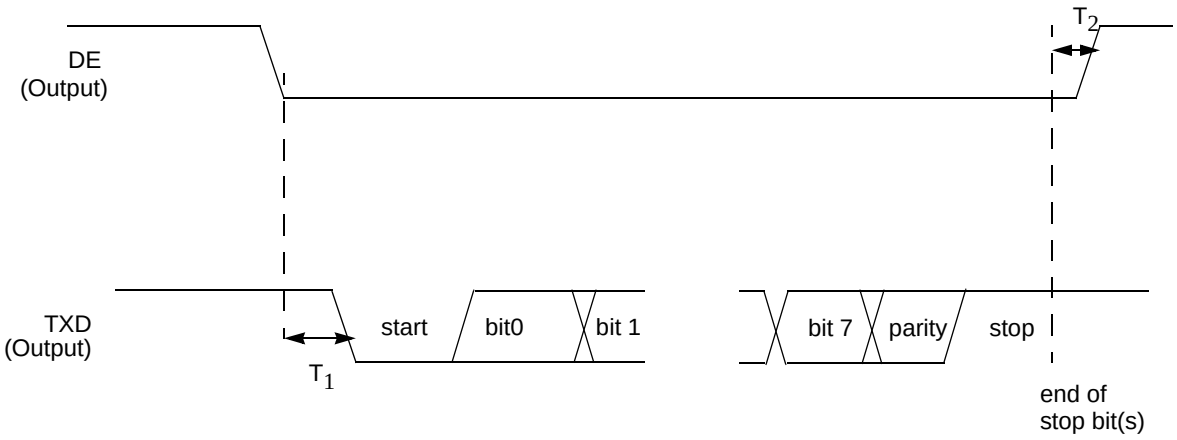


Figure 79. UART Timing Without CTS

Table 208. UART Timing Without CTS

		Delay (ns)	
Parameter	**Abbreviation**	**Min**	**Max**
UART			
T_1	DE assertion to TXD falling edge (start bit) delay	1 * X_{IN} period	1 bit time
T_2	End of stop bit(s) to DE deassertion delay (Tx Data Register is empty)	± 5	