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	Obsolete
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	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K 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	Surface Mount
Package / Case	44-VFQFN Exposed Pad
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f0880qn020sg

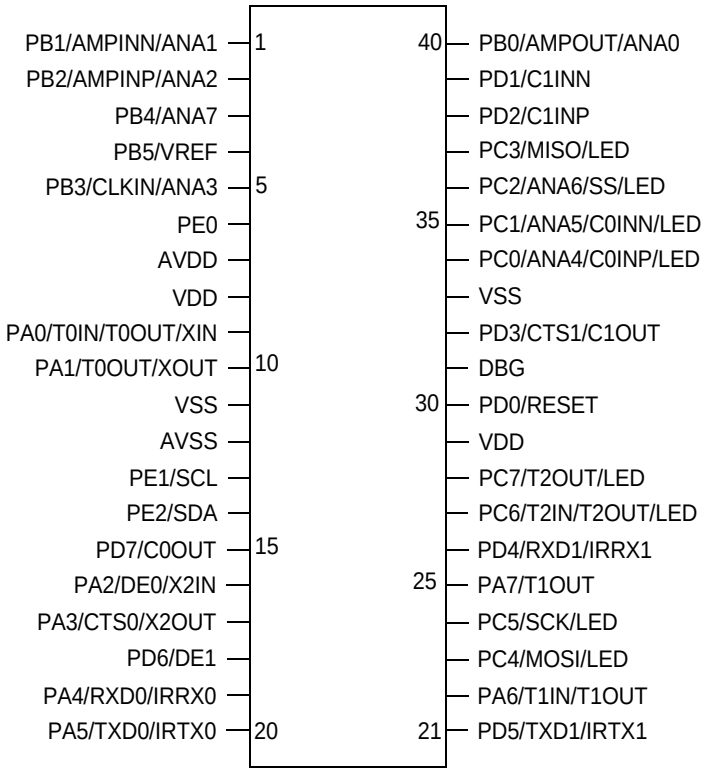

Figure 4. Z8F2480, Z8F1680 and Z8F0880 in 40-Pin Dual Inline Package (PDIP)

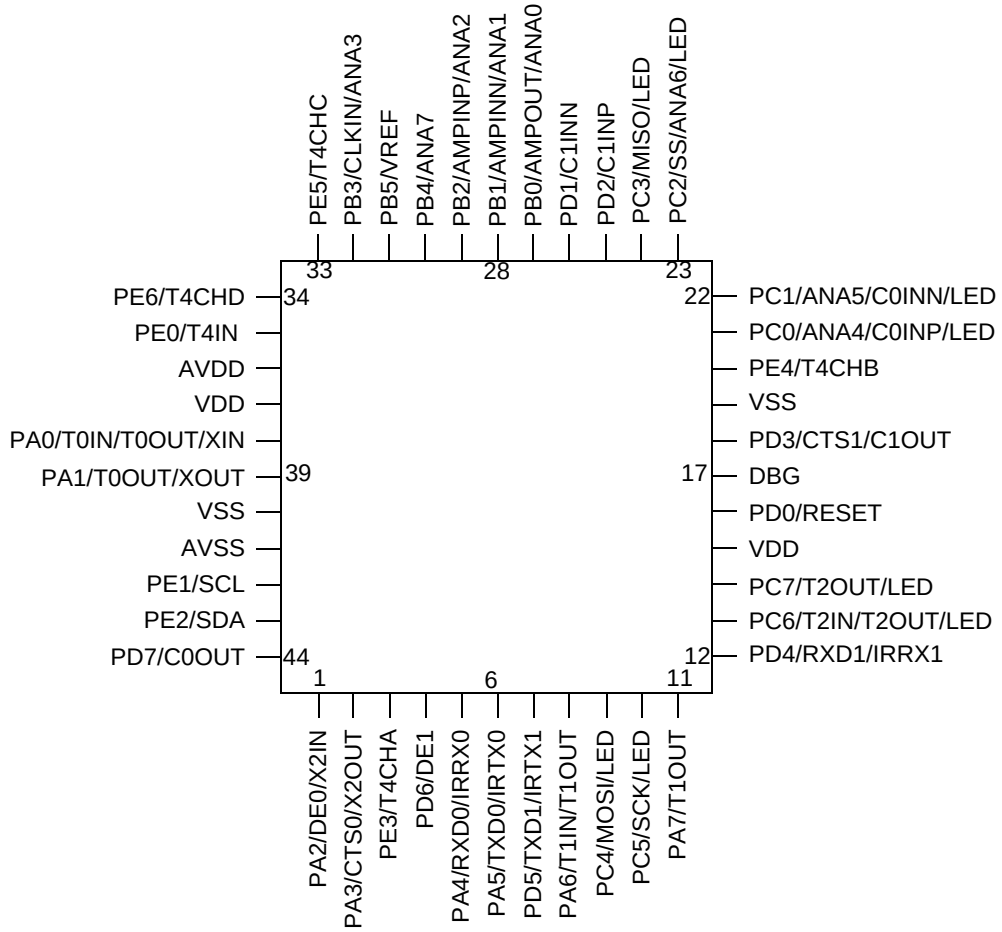

Figure 5. Z8F2480, Z8F1680 and Z8F0880 in 44-Pin Low-Profile Quad Flat Package (LQFP) or Quad Flat No Lead (QFN)

2.4. Pin Characteristics

Table 5 provides detailed information about the characteristics of each pin available on the F1680 Series MCU 20-, 28-, 40- and 44-pin devices. Data provided in Table 5 is sorted alphabetically by the pin symbol mnemonic.

Table 5. Pin Characteristics (20-, 28-, 40- and 44-pin Devices)

Symbol Mnemonic	Direction	Reset Direction	Active Low or Active High	Tristate Output	Internal Pull-up or Pull-down	Schmitt Trigger Input	Open Drain Output	5V Tolerance
AV_{DD}	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
AV_{SS}	N/A	N/A	N/A	N/A	N/A	N/A	N/A	NA
DBG	I/O	I	N/A	Yes	Yes	Yes	Yes	No
PA[7:0]	I/O	I	N/A	Yes	Programmable pull-up	Yes	Yes, programmable	Yes, 5V tolerant inputs unless pull-ups are enabled
PB[5:0]	I/O	I	N/A	Yes	Programmable pull-up	Yes	Yes, programmable	Yes, 5V tolerant inputs unless pull-ups are enabled
PC[7:0]	I/O	I	N/A	Yes	Programmable pull-up	Yes	Yes, programmable	Yes, 5V tolerant inputs unless pull-ups are enabled
PD[7:1]	I/O	I	N/A	Yes	Programmable pull-up	Yes	Yes, programmable	Yes, 5V tolerant inputs unless pull-ups are enabled

Table 6. F1680 Series MCU Program Memory Maps (Continued)

Program Memory Address (Hex)	Function
Z8F1680 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–3FFF	Program Flash
E000–E3FF	1KB PRAM
Z8F0880 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–1FFF	Program Flash
E000–E3FF	1KB PRAM

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

3.3. Data Memory

The F1680 Series MCU does not use the eZ8 CPU's 64KB Data Memory address space.

3.4. Flash Information Area

Table 7 describes the F1680 Series MCU Flash Information Area. This 512-byte Information Area is accessed by setting bit 7 of the Flash Page Select Register to 1. When access is enabled, the Flash Information Area is mapped into the Program Memory and overlays the 512bytes at addresses FE00H to FFFFH. When the Information Area access is enabled, all reads from these Program Memory addresses return the Information Area data rather than the Program Memory data. Access to the Flash Information Area is read-only.

Bit	Description (Continued)
[4] EXT	**External Reset Indicator** If this bit is set to 1, a Reset initiated by the external $\overline{\text{RESET}}$ pin occurs. A POR or a Stop Mode Recovery from a change in an input pin resets this bit. Reading this register resets this bit.
[3:1]	Reserved; must be 0.
[0] LVD	**Low-Voltage Detection Indicator** If this bit is set to 1 the current state of the supply voltage is below the low-voltage detection threshold. This value is not latched but is a real-time indicator of the supply voltage level.

Table 13. Reset Status Per Event

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

7.11.5. Port A–E Output Control Subregisters

The Port A–E Output Control Subregister, shown in Table 25, is accessed through the Port A–E Control Register by writing 03H to the Port A–E Address Register. Setting the bits in the Port A–E Output Control subregisters to 1 configures the specified port pins for open-drain operation. These subregisters affect the pins directly and, as a result, alternate functions are also affected.

Table 25. Port A–E Output Control Subregisters (PxOC)

Bits	7	6	5	4	3	2	1	0
Field	POC7	POC6	POC5	POC4	POC3	POC2	POC1	POC0
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	If 03H in Port A–E Address Register, accessible through the Port A–E Control Register							

Bit	Description
[7:0] POC	**Port Output Control** These bits function independently of the alternate function bit and always disable the drains if set to 1. 0 = The drains are enabled for any output mode (unless overridden by the alternate function). 1 = The drain of the associated pin is disabled (open-drain mode).

7.11.6. Port A–E High Drive Enable Subregisters

The Port A–E High Drive Enable Subregister, shown in Table 26, is accessed through the Port A–E Control Register by writing 04H to the Port A–E Address Register. Setting the bits in the Port A–E High Drive Enable subregisters to 1 configures the specified port pins for high-current output drive operation. The Port A–E High Drive Enable Subregister affects the pins directly and, as a result, alternate functions are also affected.

Table 26. Port A–E High Drive Enable Subregisters (PxHDE)

Bits	7	6	5	4	3	2	1	0
Field	PHDE7	PHDE6	PHDE5	PHDE4	PHDE3	PHDE2	PHDE1	PHDE0
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	If 04H in Port A–E Address Register, accessible through the Port A–E Control Register							

Bit	Description
[7:0] PHDE	**Port High Drive Enabled** 0 = The Port pin is configured for standard output current drive. 1 = The Port pin is configured for high output current drive.

One-Shot time-out. First set the TPOL bit in the Timer Control 1 Register to the start value before beginning ONE-SHOT Mode. Then, after starting the timer, set TPOL to the opposite bit value.

Observe the following steps to configure a timer for ONE-SHOT Mode and to initiate the count:

1. Write to the Timer Control 1 Register to:
   - Disable the timer
   - Configure the timer for ONE-SHOT Mode
   - Set the prescale value
   - If using the Timer Output alternate function, set the initial output level (High or Low)
2. Write to the Timer Control 2 Register to choose the timer clock source.
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.
5. Write to the Timer Reload High and Low Byte registers to set the reload value.
6. If required, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
7. If using the Timer Output function, configure the associated GPIO port pin for the Timer Output alternate function.
8. Write to the Timer Control 1 Register to enable the timer and initiate counting.

In ONE-SHOT Mode, the timer clock always provides the timer input. The timer period is calculated using the following equation:

$$\text{ONE-SHOT Mode Time-Out Period (s)} = \frac{\text{Reload Value - Start Value} \times \text{Prescale}}{\text{Timer Clock Frequency (Hz)}}$$

9.2.3.2. TRIGGERED ONE-SHOT Mode

In TRIGGERED ONE-SHOT Mode, the timer operates in the following sequence:

1. The Timer idles until a trigger is received. The Timer trigger is taken from the GPIO port pin timer input alternate function. The TPOL bit in the Timer Control 1 Register selects whether the trigger occurs on the rising edge or the falling edge of the timer input signal.
2. Following the trigger event, the Timer counts timer clocks up to the 16-bit reload value stored in the Timer Reload High and Low Byte registers.

If the TPOL bit in the Timer Control 1 Register is set to 1, the Timer Output signal begins as High (1) and then transitions to Low (0) when the timer value matches the PWM value. The Timer Output signal returns to High (1) after the timer reaches the reload value and is reset to 0001H.

If the TPOL bit in the Timer Control 1 Register is set to 0, the Timer Output signal begins as Low (0) and then transitions to High (1) when the timer value matches the PWM value. The Timer Output signal returns to Low (0) after the timer reaches the reload value and is reset to 0001H.

Observe the following steps to configure a timer for PWM SINGLE OUTPUT Mode and initiate PWM operation:

1. Write to the Timer Control 1 Register to:
   - Disable the timer
   - Configure the timer for PWM mode
   - Set the prescale value
   - Set the initial logic level (High or Low) and PWM High/Low transition for the Timer Output Alternate Function
2. Write to the Timer Control 2 Register to choose the timer clock source.
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). This value only affects the first pass in PWM mode. After the first timer reset in PWM mode, counting always begins at the reset value of 0001H.
5. Write to the Timer PWM0 High and Low Byte registers to set the PWM value.
6. Write to the Timer Reload High and Low Byte registers to set the reload value (PWM period). The reload value must be greater than the PWM value.
7. If appropriate, enable the timer interrupt and set the timer interrupt priority by writing to the relevant interrupt registers.
8. Configure the associated GPIO port pin for the Timer Output alternate function.
9. Write to the Timer Control 1 Register to enable the timer and initiate counting.

The PWM period is calculated using the following equation:

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

Bit	Description (Continued)
[5] OE	**Receive Data and Autobaud Overrun Error** This bit is set just as in normal UART operation if a receive data overrun error occurs. This bit is also set during LIN Slave autobaud if the BRG counter overflows before the end of the autobaud sequence. This indicates that the receive activity is not an autobaud character or the master baud rate is too slow. The ATB status bit will also be set in this case. This bit is cleared by reading the Receive Data Register. 0 = No autobaud or data overrun error occurred. 1 = An autobaud or data overrun error occurred.
[4] FE	**Framing Error** This bit indicates that a framing error (no stop bit following data reception) is detected. Reading the Receive Data Register clears this bit. 0 = No framing error occurred. 1 = A framing error occurred.
[3] BRKD	**Break Detect** This bit is set in LIN mode if: • It is in Lin Sleep state and a break of at least 4 bit times occurred (Wake-up event) or • It is in Slave Wait Break state and a break of at least 11 bit times occurred (Break event) or • It is in Slave Active state and a break of at least 10 bit times occurs. Reading the Status 0 Register or the Receive Data Register clears this bit. 0 = No LIN break occurred. 1 = LIN break occurred.
[2] TDRE	**Transmitter Data Register Empty** This bit indicates that the Transmit Data Register is empty and ready for additional data. Writing to the Transmit Data Register resets this bit. 0 = Do not write to the Transmit Data Register. 1 = The Transmit Data Register is ready to receive an additional byte for transmission.
[1] TXE	**Transmitter Empty** This bit indicates that the Transmit Shift Register is empty and character transmission is completed. 0 = Data is currently transmitting. 1 = Transmission is complete.
[0] ATB	**LIN Slave Autobaud Complete** This bit is set in LIN SLAVE Mode when an autobaud character is received. If the ABIEN bit is set in the LIN Control Register, then a receive interrupt is generated when this bit is set. Reading the Status 0 Register clears this bit. This bit will be 0 in LIN MASTER Mode.

12.3.5. LIN-UART Control 0 Register

The LIN-UART Control 0 Register, shown in Table 89, configures the basic properties of LIN-UART's transmit and receive operations. A more detailed discussion of each bit follows the table.

Table 89. LIN-UART Control 0 Register (U0CTL0 = F42H)

Bit	7	6	5	4	3	2	1	0
Field	TEN	REN	CTSE	PEN	PSEL	SBRK	STOP	LBEN
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	F42H, F4AH							

Note: R/W = Read/Write.

Bit	Description
[7] TEN	**Transmit Enable** This bit enables or disables the transmitter. The enable is also controlled by the $\overline{CTS}$ signal and the CTSE bit. If the $\overline{CTS}$ signal is Low and the CTSE bit is 1, the transmitter is enabled. 0 = Transmitter disabled. 1 = Transmitter enabled.
[6] REN	**Receive Enable** This bit enables or disables the receiver. 0 = Receiver disabled. 1 = Receiver enabled.
[5] CTSE	**Clear To Send Enable** 0 = The $\overline{CTS}$ signal has no effect on the transmitter. 1 = The LIN-UART recognizes the $\overline{CTS}$ signal as an enable control for the transmitter.
[4] PEN	**Parity Enable** This bit enables or disables parity. Even or odd is determined by the PSEL bit. 0 = Parity is disabled. This bit is overridden by the MPEN bit. 1 = The transmitter sends data with an additional parity bit and the receiver receives an additional parity bit.
[3] PSEL	**Parity Select** 0 = Even parity is sent as an additional parity bit for the transmitter/receiver. 1 = Odd parity is sent as an additional parity bit for the transmitter/receiver.

12.3.7. Noise Filter Control Register

When MSEL = 001b, the Noise Filter Control Register, shown in Table 91, provides control for the digital noise filter.

Table 91. Noise Filter Control Register (U0CTL1 = F43H with MSEL = 001b)

Bit	7	6	5	4	3	2	1	0
Field	NFEN	NFCTL			—			
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R
Address	F43H, F4BH							

Note: R = Read; R/W = Read/Write.

Bit Position	Value	Description
[7] NFEN	**Noise Filter Enable**	
	0	Noise filter is disabled.
	1	Noise filter is enabled. Receive data is preprocessed by the noise filter.
[6:4] NFCTL	**Noise Filter Control** This field controls the delay and noise rejection characteristics of the noise filter. The wider the counter is, the more delay is introduced by the filter and the wider the noise event is filtered.	
	000	4-bit up/down counter.
	001	5-bit up/down counter.
	010	6-bit up/down counter.
	011	7-bit up/down counter.
	100	8-bit up/down counter.
	101	9-bit up/down counter.
	110	10-bit up/down counter.
	111	11-bit up/down counter.
[3:0] Reserved	—	Reserved; must be 0000.

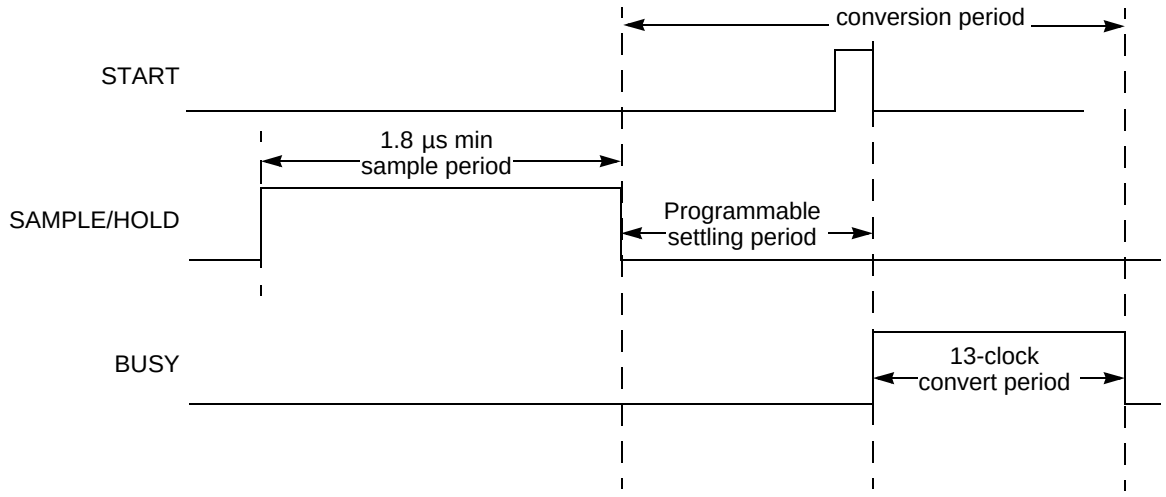

Figure 31. ADC Timing Diagram

14.2.2. ADC Interrupt

The ADC can generate an interrupt request when a conversion is completed. An interrupt request that is pending when the ADC is disabled is not automatically cleared. See Figure 32.

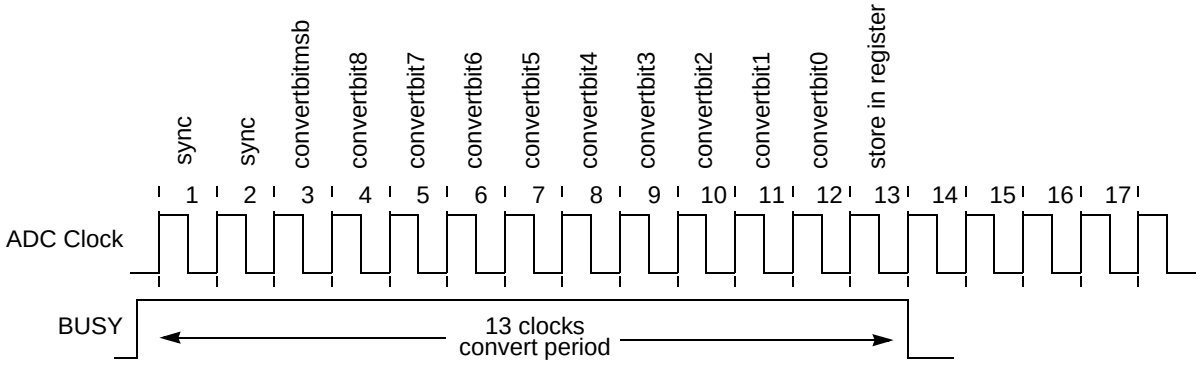

Figure 32. ADC Convert Timing

14.2.3. Reference Buffer

The reference buffer, RBUF, supplies the reference voltage for the ADC. When enabled, the internal voltage reference generator supplies the ADC and this voltage is available on

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.

17.2.5.1. Master Arbitration

If a Master loses arbitration during the address byte it releases the SDA line, switches to SLAVE Mode and monitors the address to determine if it is selected as a Slave. If a Master loses arbitration during the transmission of a data byte, it releases the SDA line and waits for the next stop or start condition.

The Master detects a loss of arbitration when a 1 is transmitted but a 0 is received from the bus in the same bit-time. This loss occurs if more than one Master is simultaneously accessing the bus. Loss of arbitration occurs during the address phase (two or more Masters accessing different slaves) or during the data phase, when the masters are attempting to Write different data to the same Slave.

When a Master loses arbitration, the software is informed by means of the Arbitration Lost interrupt. The software can repeat the same transaction at a later time.

A special case can occur when a Slave transaction starts just before the software attempts to start a new master transaction by setting the start bit. In this case, the state machine enters its Slave states before the start bit is set and as a result the I^2C controller will not arbitrate. If a Slave address match occurs and the I^2C controller receives/transmits data, the start bit is cleared and an Arbitration Lost interrupt is asserted. The software can minimize the chance of this instance occurring by checking the busy bit in the I2CSTATE Register before initiating a Master transaction. If a slave address match does not occur, the Arbitration Lost interrupt will not occur and the start bit will not be cleared. The I^2C controller will initiate the master transaction after the I^2C bus is no longer busy.

17.2.5.2. Master Address-Only Transactions

It is sometimes preferable to perform an address-only transaction to determine if a particular slave device is able to respond. This transaction can be performed by monitoring the ACKV bit in the I2CSTATE Register after the address has been written to the I2CDATA Register and the start bit has been set. After the ACKV bit is set, the ACK bit in the I2CSTATE Register determines if the slave is able to communicate. The stop bit must be set in the I2CCTL Register to terminate the transaction without transferring data. For a 10-bit slave address, if the first address byte is acknowledged, the second address byte should also be sent to determine if the preferred Slave is responding.

Another approach is to set both the stop and start bits (for sending a 7-bit address). After both bits have been cleared (7-bit address has been sent and transaction is complete), the ACK bit can be read to determine if the Slave has acknowledged. For a 10-bit Slave, set the stop bit after the second TDRE interrupt (which indicates that the second address byte is being sent).

17.2.5.3. Master Transaction Diagrams

In the following transaction diagrams, the shaded regions indicate the data that is transferred from the Master to the Slave and the unshaded regions indicate the data that is

5. The I^2C controller receives the data byte and responds with Acknowledge or Not Acknowledge depending on the state of the NAK bit in the I2CCTL Register. The I^2C controller generates the receive data interrupt by setting the RDRF bit in the I2CISTAT Register.
6. The software responds by reading the I2CISTAT Register, finding the RDRF bit = 1 and reading the I2CDATA Register clearing the RDRF bit. If software can accept only one more data byte it sets the NAK bit in the I2CCTL Register.
7. The master and slave loops through Step 4 to Step 6 until the master detects a Not Acknowledge instruction or runs out of data to send.
8. The master sends the stop or restart signal on the bus. Either of these signals can cause the I^2C controller to assert a stop interrupt (the stop bit = 1 in the I2CISTAT Register). Because the slave received data from the master, the software takes no action in response to the stop interrupt other than reading the I2CISTAT Register to clear the stop bit in the I2CISTAT Register.

17.2.6.6. Slave Receive Transaction with 10-Bit Address

The data transfer format for writing data from a master to a slave with 10-bit addressing is displayed in Figure 48. The procedure that follows describes the I^2C Master/Slave Controller operating as a slave in 10-bit addressing mode and receiving data from the bus master.

S	Slave Address 1st Byte	W=0	A	Slave Address 2nd Byte	A	Data	A	Data	A/$\overline{A}$	P/S

Figure 48. Data Transfer Format—Slave Receive Transaction with 10-Bit Address

1. The software configures the controller for operation as a slave in 10-bit addressing mode, as follows:
   a. Initialize the MODE field in the I2CMODE 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 the SLA[9:8] bits in the I2CMODE Register.
   d. Set IEN = 1 in the I2CCTL Register. Set NAK = 0 in the I^2C Control Register.
2. The Master initiates a transfer, sending the first address byte. The I^2C controller recognizes the start of a 10-bit address with a match to SLA[9:8] and detects R/$\overline{W}$ 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 detects an address match between the second address byte and SLA[7:0]. The SAM bit in the

20.3.4. Flash Sector Protect Register

The Flash Sector Protect Register is shared with the Flash Page Select Register. When the Flash Control Register (see page 271) is written with 5EH, the next write to this address targets the Flash Sector Protect Register. In all other cases, it targets the Flash Page Select Register.

This register selects one of the eight available Flash memory sectors to be protected. The reset state of each Sector Protect bit is an unprotected state. After a sector is protected by setting its corresponding register bit, it can only be unprotected (the register bit can only be cleared) by a System Reset. Please refer to Table 132 on page 262 and to Figures 51 through 53 to review how Flash memory is arranged by sector.

Table 137. Flash Sector Protect Register (FPROT)

Bits	7	6	5	4	3	2	1	0
Field	SPROT7	SPROT6	SPROT5	SPROT4	SPROT3	SPROT2	SPROT1	SPROT0
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	FF9H							

Bit	Description
[7:0] SPROT*x*	**Sector Protection** • On Z8F2480 devices, each bit corresponds to a 3KB Flash sector. • On Z8F1680 devices, each bit corresponds to a 2KB Flash sector. • On Z8F0880 devices, each bit corresponds to a 1KB Flash sector.

20.3.5. Flash Frequency High and Low Byte Registers

The Flash Frequency High and Low Byte registers, shown in Tables 138 and 139, combine to form a 16-bit value, FFREQ, to control timing for Flash program and erase operations. The 16-bit binary Flash Frequency value must contain the system clock frequency (in kHz) and is calculated using the following equation:

$$FFREQ[15:0] = \{FFREQH[7:0],FFREQL[7:0]\} = \frac{\text{System Clock Frequency}}{1000}$$

Caution: Flash programming and erasure is not supported for system clock frequencies below 32 kHz or above 20 MHz. The Flash Frequency High and Low Byte registers must be loaded with the correct values to ensure proper operation of the device.

Modes section on page 42) and configured for a threshold voltage of 2.4V or greater (see the Trim Bit Address Space section on page 282).

A System Reset that occurs during a write operation (such as a pin reset or watchdog timer reset) also perturbs the byte currently being written. All other bytes in the array remain unperturbed.

22.2.4. Optimizing NVDS Memory Usage for Execution Speed

As listed in Table 161, the NVDS read time varies drastically, this discrepancy being a trade-off for minimizing the frequency of writes that require post-write page erases. The NVDS read time of address N is a function of the number of writes to addresses other than N since the most recent write to address N, as well as the number of writes since the most recent page erase. Neglecting effects caused by page erases and results caused by the initial condition in which the NVDS is blank, a rule of thumb is that every write since the most recent page erase causes read times of unwritten addresses to increase by 0.8µs up to a maximum of 258µs.

Table 161. NVDS Read Time

Operation	Minimum Latency (µs)	Maximum Latency (µs)
Read	71	258
Write	126	136
Illegal Read	6	6
Illegal Write	7	7

If NVDS read performance is critical to your software architecture, you can optimize your code for speed by using either of the methods listed below.

- Periodically refresh all addresses that are used; the most useful method. The optimal use of NVDS, in terms of speed, is to rotate the writes evenly among all planned addresses, bringing all reads closer to the minimum read time. Because the minimum read time is much less than the write time, however, actual speed benefits are not always realized.
- Use as few unique addresses as possible to optimize the impact of refreshing.

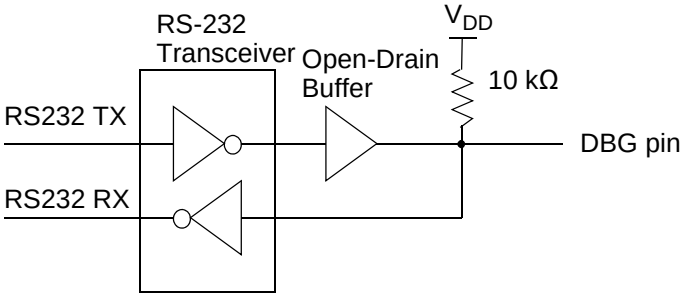

Figure 58. Interfacing the On-Chip Debugger's DBG Pin with an RS-232 Interface, #2 of 2

23.2.1. DEBUG Mode

The operating characteristics of the Z8 Encore! XP F1680 Series device 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 or otherwise defined by the on-chip peripheral to disable in DEBUG mode
- Automatically exits HALT Mode
- Constantly refreshes the Watch-Dog Timer, if enabled

23.2.1.1. Entering DEBUG Mode

The device enters DEBUG mode following any of the these operations:

- Writing the DBGMODE bit in the OCD Control Register to 1 using the OCD interface
- eZ8 CPU execution of a breakpoint (BRK) instruction (when enabled)
- Match of PC to OCDCNTR Register (when enabled)
- OCDCNTR Register decrements to `0000H` (when enabled)
- The DBG pin is Low when the device exits Reset

23.2.1.2. Exiting DEBUG Mode

The device exits DEBUG mode following any of these operations:

- Clearing the DBGMODE bit in the OCD Control Register to 0
- Power-on reset

Table 197. Comparator Electrical Characteristics

		TA = 0°C to +70°C TA = –40°C to +105°C V_{DD} = 1.8V to 3.6V				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
V_{OS}	Input DC Offset	–	5	–	mV	
V_{CREF_P}	Programmable Internal Reference Voltage Range	0	–	1.8	V	
V_{CREF_D}	Default Internal Reference Voltage	0.90	1.0	1.10	V	
$I_{DD}CMP$	Comparator Active Current	–	–	400	µA	
$I_{DDQ}CMP$	Comparator Quiescent Current	–	5	–	nA	
V_{HYS}	Input Hysteresis	–	8	–	mV	
T_{PROP}	Propagation Delay	–	100	–	ns	

Table 198. Temperature Sensor Electrical Characteristics

		TA = 0°C to +70°C TA = –40°C to +105°C							
		V_{DD} = 2.7 to 3.6V			V_{DD} = 1.8 to 2.7V				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**	**Units**	**Conditions**
T_{AERR}	Temperature Sensor Output Error	–7	–	+7	–10	–	+10	°C	–40°C to +105°C (as measured by ADC)
		–1.5	–	+1.5	–3	–	+3	°C	+20°C to +30°C (as measured by ADC)
		–10	–	10	–15	–	15	°C	–40°C to +105°C (as measured by comparator)
$I_{DD}TEMP$	Temperature Sensor Active Current	–	–	100	–	–	100	µA	
$I_{DDQ}TEMP$	Temperature Sensor Quiescent Current	–	5	–	–	5	–	nA	
T_{WAKE}	Time for Wake up	–	80	100	–	80	100	µs	

29.4.3. On-Chip Debugger Timing

Figure 77 and Table 206 provide timing information for the DBG pin. The DBG pin timing specifications assume a 4ns maximum rise and fall time.

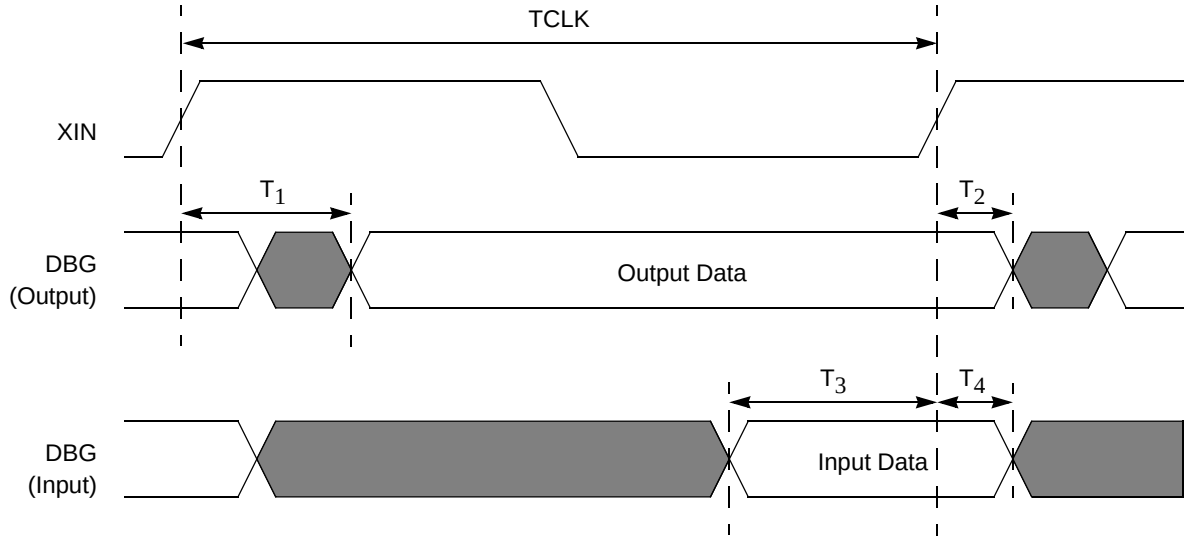

Figure 77. On-Chip Debugger Timing

Table 206. On-Chip Debugger Timing

		Delay (ns)	
Parameter	**Abbreviation**	**Min**	**Max**
DBG			
T_1	X_{IN} Rise to DBG Valid Delay	–	15
T_2	X_{IN} Rise to DBG Output Hold Time	2	–
T_3	DBG to XIN Rise Input Setup Time	5	–
T_4	DBG to XIN Rise Input Hold Time	5	–