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	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-VFQFN Exposed Pad
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f2480qn020eg

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

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
PE[6:0]	I/O	I	N/A	Yes	Programmable pull-up	Yes	Yes, programmable	Yes, 5V tolerant inputs unless pull-ups are enabled
RESET/ PD0	I/O	I/O (defaults to $\overline{\text{RESET}}$)	Low (in RESET mode)	Yes (PD0 only)	Programmable for PD0; always On for $\overline{\text{RESET}}$	Yes	Programmable for PD0; always On for $\overline{\text{RESET}}$	Yes, 5V tolerant inputs unless pull-ups are enabled
V_{DD}	N/A	N/A	N/A	N/A			N/A	N/A
V_{SS}	N/A	N/A	N/A	N/A			N/A	N/A

Chapter 3. Address Space

The eZ8 CPU can access the following three distinct address spaces:

- The Register File contains addresses for general-purpose registers, eZ8 CPU, peripherals and GPIO port control registers
- The Program Memory contains addresses for all memory locations having executable code and/or data
- The Data Memory contains addresses for all memory locations that contain data only

These three address spaces are covered briefly in the following sections. For more details about the eZ8 CPU and its address space, refer to the eZ8 CPU Core User Manual (UM0128), available for download at www.zilog.com.

3.1. Register File

The Register File address space in the Z8 Encore!® MCU is 4KB (4096 bytes). The Register File is composed of two sections: control registers and general-purpose registers. When instructions are executed, registers defined as sources are read and registers defined as destinations are written. The architecture of the eZ8 CPU allows all general-purpose registers to function as accumulators, address pointers, index registers, stack areas, or scratch pad memory.

The upper 256 bytes of the 4KB Register File address space are reserved for control of the eZ8 CPU, on-chip peripherals and the input/output ports. These registers are located at addresses F00H to FFFH. Some of the addresses within the 256 B control register sections are reserved (that is, unavailable). Reading from a reserved Register File address returns an undefined value. Zilog does not recommend writing to the reserved Register File addresses because doing so can produce unpredictable results.

The on-chip Register RAM always begins at address 000H in the Register File address space. The F1680 Series MCU contains 1KB or 2KB of on-chip Register RAM. Reading from Register File addresses outside the available RAM addresses (and not within the control register address space) returns an undefined value. Writing to these Register File addresses produces no effect.

In addition, the F1680 Series MCU contains 1KB of on-chip Program RAM. Normally it is used as Program RAM and is present in the Program Memory address space (see the Program Memory section on page 20). However, it can also be used as additional Register RAM present in the Register File address space 800H–BFFH (1KB Program RAM, 2KB Register RAM), or 400H–7FFH (1KB Program RAM, 1KB Register RAM), if you do not

Chapter 7. General-Purpose Input/Output

The Z8 Encore! XP F1680 Series product supports a maximum of 37 port pins (Ports A–E) for general-purpose input/output (GPIO) operations. Each port contains control and data registers. The GPIO control registers determine data direction, open-drain, output drive current, programmable pull-ups, Stop Mode Recovery functionality and alternate pin functions. Each port pin is individually programmable. In addition, the Port C pins are capable of direct LED drive at programmable drive strengths.

7.1. GPIO Port Availability by Device

Table 16 lists the port pins available with each device and package type.

Table 16. Port Availability by Device and Package Type

Devices	Package	10-bit ADC	SPI	Port A	Port B	Port C	Port D	Port E	Total I/O
Z8F2480PH, Z8F2480HH Z8F2480SH; Z8F1680PH, Z8F1680HH Z8F1680SH; Z8F0880PH, Z8F0880HH Z8F0880SH	20-pin PDIP SOIC SSOP	7	0	[7:0]	[3:0]	[3:0]	[0]	—	17
Z8F2480PJ, Z8F2480SJ Z8F2480HJ; Z8F1680PJ, Z8F1680SJ Z8F1680HJ; Z8F0880PJ, Z8F0880SJ Z8F0880HJ	28-pin PDIP SOIC SSOP	8	1	[7:0]	[5:0]	[7:0]	[0]	—	23
Z8F2480PM, Z8F1680PM, Z8F0880PM	40-pin PDIP	8	1	[7:0]	[5:0]	[7:0]	[7:0]	[2:0]	33
Z8F2480AN, Z8F2480QN; Z8F1680AN, Z8F1680QN; Z8F0880AN, Z8F0880QN	44-pin LQFP QFN	8	1	[7:0]	[5:0]	[7:0]	[7:0]	[6:0]	37

Table 19. Port Alternate Function Mapping, 40-/44-Pin Parts[1,2] (Continued)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port B	PB0	Reserved		AFS1[0]: 0
		ANA0/AMPOUT	ADC Analog Input/LPO Output	AFS1[0]: 1
	PB1	Reserved		AFS1[1]: 0
		ANA1/AMPINN	ADC Analog Input/LPO Input (N)	AFS1[1]: 1
	PB2	Reserved		AFS1[2]: 0
		ANA2/AMPINP	ADC Analog Input/LPO Input (P)	AFS1[2]: 1
	PB3	CLKIN	External Clock Input	AFS1[3]: 0
		ANA3	ADC Analog Input	AFS1[3]: 1
	PB4	Reserved		AFS1[4]: 0
		ANA7	ADC Analog Input	AFS1[4]: 1
	PB5	Reserved		AFS1[5]: 0
		VREF	Voltage Reference	AFS1[5]: 1

Notes:
1. Because there are at most two choices of alternate functions for some pins in Ports A–C, the Alternate Function Set Register (AFS2) is implemented but not used to select the function. The alternate function selection must also be enabled, as described in the Port A–E Alternate Function Subregisters section on page 61.
2. Because there is only one alternate function for each Port D and Port E pin, the Alternate Function Set registers are not implemented for Ports D and E. Enabling the alternate function selections automatically enables the associated alternate function, as described in the Port A–E Alternate Function Subregisters section on page 61.
3. This timer function is only available in the 44-pin package; its alternate functions are reserved in the 40-pin package.

7.11.12. Port A–E Output Data Register

The Port A–E Output Data Register, shown in Table 32, controls the output data to the pins.

Table 32. Port A–E Output Data Register (P*x*OUT)

Bits	7	6	5	4	3	2	1	0
Field	POUT7	POUT6	POUT5	POUT4	POUT3	POUT2	POUT1	POUT0
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	FD3H, FD7H, FDBH, FDFH, FE3H							

Bit	Description
[7:0] POUT	**Port Output Data** These bits contain the data to be driven to the port pins. The values are only driven if the corresponding pin is configured as an output and the pin is not configured for Alternate Function operation. 0 = Drive a logical 0 (Low). 1 = Drive a logical 1 (High). High value is not driven if the drain has been disabled by setting the corresponding Port Output Control Register bit to 1.

7.11.13. LED Drive Enable Register

The LED Drive Enable Register, shown in Table 33, activates the controlled current drive. The Port C pin must first be enabled by setting the Alternate Function Register to select the LED function.

Table 33. LED Drive Enable (LEDEN)

Bits	7	6	5	4	3	2	1	0
Field	LEDEN[7:0]							
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	F82H							

Bit	Description
[7:0] LEDEN	**LED Drive Enable** These bits determine which Port C pins are connected to an internal current sink. 0 = Tristate the Port C pin. 1 = Connect controlled current synch to Port C pin.

Table 54 provides an example initialization sequence for configuring Timer 0 in DEMODULATION Mode and initiating operation.

Table 54. DEMODULATION Mode Initialization Example

Register	Value	Comment
T0CTL0	C0H	TMODE[3:0] = 1100B selects DEMODULATION Mode. TICONFIG[1:0] = 10B enables interrupt only on Capture events. CSC = 0 selects the Timer Input from the GPIO pin. PWMD[2:0] = 000B has no effect. INPCAP = 0 has no effect. TEN = 0 disables the timer. PRES[2:0] = 000B sets prescaler to divide by 1. TPOLHI,TPOL = 10 enables trigger and Capture on both rising and falling edges of Timer Input. TCLKS = 1 enables 32kHz peripheral clock as timer clock source
T0CTL1	04H	
T0CTL2	11H	
T0H	00H	Timer starting value = 0001H.
T0L	01H	
T0RH	ABH	Timer reload value = ABCDH
T0RL	CDH	
T0PWM0H	00H	Initial PWM0 value = 0000H
T0PWM0L	00H	
T0PWM1H	00H	Initial PWM1 value = 0000H
T0PWM1H	00H	
T0NFC	C0H	NFEN = 1 enables noise filter NFCTL = 100B enables 8-bit up/down counter
PAADDR	02H	Selects Port A Alternate Function control register.
PACTL[1:0]	11B	PACTL[0] enables Timer 0 Input alternate function. PACTL[1] enables Timer 0 Output alternate function.
IRQ0ENH[5]	0B	Disables the Timer 0 interrupt.
IRQ0ENL[5]	0B	
T0CTL1	84H	TEN = 1 enables the timer. All other bits remain in their appropriate settings.

Notes: After receiving the input trigger (rising or falling edge), Timer 0 will:

1. Start counting on the timer clock.
2. Upon receiving a Timer 0 Input rising edge, save the Capture value in the T0PWM0 registers, generate an interrupt, and continue to count.
3. Upon receiving a Timer 0 Input falling edge, save the Capture value in the T0PWM1 registers, generate an interrupt, and continue to count.
4. After the timer count to ABCD clocks, set the reload event flag and reset the Timer count to the start value.

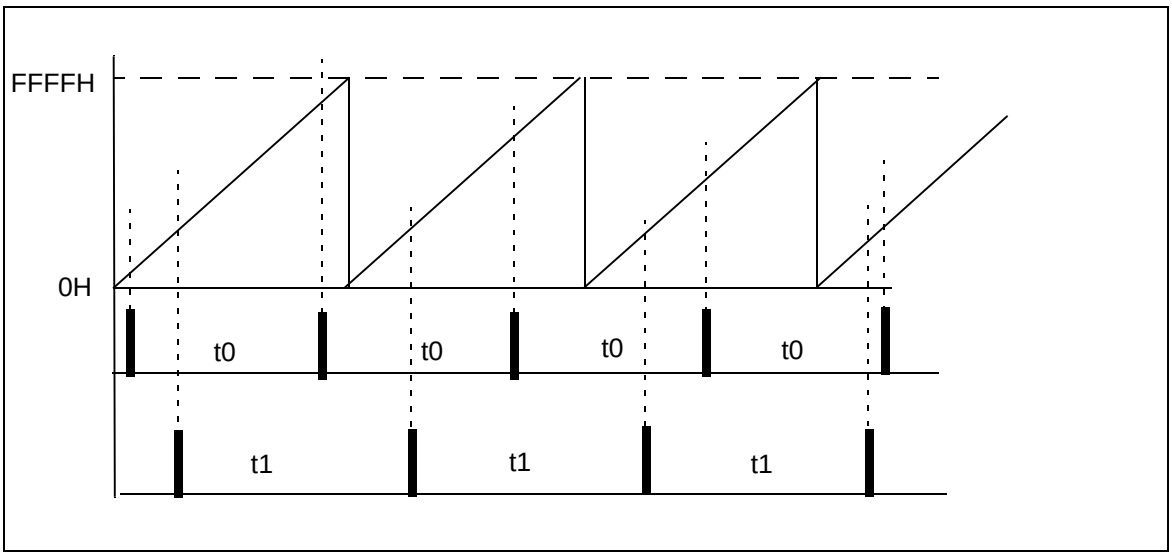

Figure 18. Count Max Mode with Channel Compare

10.7. Multi-Channel Timer Control Register Definitions

This section defines the features of the following Multi-Channel Timer Control registers.

Multi-Channel Timer High and Low Byte Registers: see page 130

Multi-Channel Timer Reload High and Low Byte Registers: see page 130

Multi-Channel Timer Subaddress Register: see page 131

Multi-Channel Timer Subregister x (0, 1, or 2): see page 132

Multi-Channel Timer Control 0, Control 1 Registers: see page 132

Multi-Channel Timer Channel Status 0 and Status 1 Registers: see page 135

Multi-Channel Timer Channel-y Control Registers: see page 137

Multi-Channel Timer Channel-y High and Low Byte Registers: see page 139

10.7.1. Multi-Channel Timer Address Map

Table 69 defines the byte address offsets for the Multi-channel Timer registers. For saving address space, a subaddress is used for the Timer Control 0, Timer Control 1, Channel Status 0, Channel Status 1, Channel-y Control, and Channel-y High and Low byte registers. Only the Timer High and Low Byte registers and the Reload High and Low Byte registers can be directly accessed.

Bit	Description
[7] TCTST	**Timer Count Status** This bit indicates if a timer count cycle is complete and is cleared by writing 1 to the bit and is cleared when TEN = 0. 0 = Timer count cycle is not complete. 1 = Timer count cycle is complete.
[6] CHST	**Channel Status** This bit indicates if a channel Capture/Compare event occurred. This bit is the logical OR of the CHyEF bits in the MCTCHS1 register. This bit is cleared when TEN=0. 0 = No channel capture/compare event has occurred. 1 = A channel capture/compare event has occurred. One or more of the CHDEF, CHCEF, CHBEF and CHAEF bits in the MCTCHS1 register are set.
[5] TCIEN	**Timer Count Interrupt Enable** This bit enables generation of timer count interrupt. A timer count interrupt is generated whenever the timer completes a count cycle: counting up to Reload Register value or counting down to zero depending on whether the TIMER mode is Count Modulo or Count up/down. 0 = Timer Count Interrupt is disabled. 1 = Timer Count Interrupt is enabled.
[4:3]	Reserved; must be 0.
2:0 TCLKS	**Timer Clock Source** 000 = System Clock (Prescaling enabled) 001 = Reserved 010 = System Clock gated by active High Timer Input signal (Prescaling enabled). 011 = System Clock gated by active Low Timer Input signal (Prescaling enabled). 100 = Timer I/O pin input rising edge (Prescaler disabled). 101 = Timer I/O pin input falling edge (Prescaler disabled). 110 = Reserved. 111 = Reserved.

Note: The input frequency of the Timer Input Signal must not exceed one-fourth the system clock frequency.

12.3. LIN-UART Control Register Definitions

The LIN-UART control registers support the LIN-UART, the associated Infrared Encoder/Decoder and the noise filter. For more information about the infrared operation, see the Infrared Encoder/Decoder section on page 182.

12.3.1. LIN-UART Transmit Data Register

Data bytes written to the LIN-UART Transmit Data Register, shown in Table 83, are shifted out on the TxD pin. The write-only LIN-UART Transmit Data Register shares a Register File address with the read-only LIN-UART Receive Data Register.

Table 83. LIN-UART Transmit Data Register (U0TXD = F40H)

Bit	7	6	5	4	3	2	1	0
Field	TxD							
Reset	X	X	X	X	X	X	X	X
R/W	W	W	W	W	W	W	W	W
Address	F40H, F48H							

Note: W = Write; X = undefined.

Bit	Description
[7:0] TxD	**Transmit Data** LIN–UART transmitter data byte to be shifted out through the TxD pin.

be configured with MMEN = 1, SSIO = 0 ($\overline{SS}$ is an input) and $\overline{SS}$ input = 0. A mode fault sets the COL bit in the ESPI Status Register to 1. Writing a 1 to COL clears this error flag.

16.3.5.3. Receive Overrun

A receive overrun error occurs when a transfer completes and the RDRNE bit is still set from the previous transfer. A receive overrun sets the ROVR bit in the ESPI Status Register to 1. Writing a 1 to ROVR clears this error flag. The Receive Data Register is not overwritten and will contain the data from the transfer which initially set the RDRNE bit. Subsequent received data is lost until the RDRNE bit is cleared.

In SPI MASTER Mode, a receive overrun will not occur. Instead, the SCK will be paused until software responds to the previous RDRNE/TDRE requests.

16.3.5.4. SLAVE Mode Abort

In SLAVE Mode, if the $\overline{SS}$ pin deasserts before all bits in a character have been transferred, the transaction is aborted. When this condition occurs the ABT bit is set in the ESPI Status Register. A Slave abort error resets the Slave control logic to idle state.

A Slave abort error is also asserted in SLAVE Mode, if BRGCTL = 1 and a baud rate generator time-out occurs. When BRGCTL = 1 in SLAVE Mode, the baud rate generator functions as a Watchdog Timer monitoring the SCK signal. The BRG counter is reloaded every time a transition on SCK occurs while $\overline{SS}$ is asserted. The Baud Rate Reload registers must be programmed with a value longer than the expected time between the $\overline{SS}$ assertion and the first SCK edge, between SCK transitions while $\overline{SS}$ is asserted and between the last SCK edge and $\overline{SS}$ deassertion. A time-out indicates the Master is stalled or disabled. Writing a 1 to ABT clears this error flag.

16.3.6. ESPI Interrupts

ESPI has a single interrupt output which is asserted when any of the TDRE, TUND, COL, ABT, ROVR or RDRNE bits are set in the ESPI Status Register. The interrupt is a pulse which is generated when any one of the source bits initially sets. The TDRE and RDRNE interrupts can be enabled/disabled via the Data Interrupt Request Enable (DIRQE) bit of the ESPI Control Register.

A transmit interrupt is asserted by the TDRE status bit when the ESPI block is enabled and the DIRQE bit is set. The TDRE bit in the Status register is cleared automatically when the Data Register is written or the ESPI block is disabled. After the Data Register is loaded into the Shift Register to start a new transfer, the TDRE bit will be set again, causing a new transmit interrupt. In SLAVE Mode, if information is being received but not transmitted the transmit interrupts can be eliminated by selecting Receive Only mode (ESPIEN1,0 = 01). A Master cannot operate in Receive Only mode since a write to the ESPI (Transmit) Data Register is still required to initiate the transfer of a character even if information is being received but not transmitted by the software application.

Bit	Description (Continued)
[6,0] ESPIEN1, ESPIEN0	**ESPI Enable and Direction Control** 00 = The ESPI block is disabled. BRG can be used as a general-purpose timer by setting BRGCTL = 1. 01 = Receive Only Mode. Use this setting in SLAVE Mode if software application is receiving data but not sending. TDRE will not assert. Transmitted data will be all 1s. Not valid in MASTER Mode since Master must source data to drive the transfer. 10 = Transmit Only Mode Use this setting in MASTER or SLAVE Mode when the software application is sending data but not receiving. RDRNE will not assert. 11 = Transmit/Receive Mode Use this setting if the software application is both sending and receiving information. Both TDRE and RDRNE will be active.
[5] BRGCTL	**Baud Rate Generator Control** The function of this bit depends upon ESPIEN1,0. When ESPIEN1,0 = 00, this bit allows enabling the BRG to provide periodic interrupts. **If the ESPI is disabled** 0 = The Baud Rate Generator timer function is disabled. Reading the Baud Rate High and Low registers returns the BRG reload value. 1 = The Baud Rate Generator timer function and time-out interrupt is enabled. Reading the Baud Rate High and Low registers returns the BRG Counter value. **If the ESPI is enabled** 0 = Reading the Baud Rate High and Low registers returns the BRG reload value. If MMEN = 1, the BRG is enabled to generate SCK. If MMEN = 0, the BRG is disabled. 1 = Reading the Baud Rate High and Low registers returns the BRG Counter value. If MMEN = 1, the BRG is enabled to generate SCK. If MMEN = 0 the BRG is enabled to provide a Slave SCK time-out. See the SLAVE Mode Abort error description on page 211. **Caution:** If reading the counter one byte at a time while the BRG is counting keep in mind that the values will not be in sync. Zilog recommends reading the counter using (2-byte) word reads.
[4] PHASE	**Phase Select** Sets the phase relationship of the data to the clock. For more information about operation of the PHASE bit, see the ESPI Clock Phase and Polarity Control section on page 201.
[3] CLKPOL	**Clock Polarity** 0 = SCK idles Low (0). 1 = SCK idles High (1).
[2] WOR	**Wire-OR (Open-Drain) Mode Enabled** 0 = ESPI signal pins not configured for open-drain. 1 = All four ESPI signal pins (SCK, $\overline{SS}$, MISO and MOSI) configured for open-drain function. This setting is typically used for multi-Master and/or Multi-Slave configurations.
[1] MMEN	**ESPI MASTER Mode Enable** This bit controls the data I/O pin selection and SCK direction. 0 = Data out on MISO, data in on MOSI (used in SPI SLAVE Mode), SCK is an input. 1 = Data out on MOSI, data in on MISO (used in SPI MASTER Mode), SCK is an output.

Bit	Description (Continued)
[4:2] NUMBITS[2:0]	**Number of Data Bits Per Character to Transfer** This field contains the number of bits to shift for each character transfer. For information about valid bit positions when the character length is less than 8 bits, see the description of the ESPI Data Register on page 213. 000 = 8 bits 001 = 1 bit 010 = 2 bits 011 = 3 bits 100 = 4 bits 101 = 5 bits 110 = 6 bits 111 = 7 bits
[1] SSIO	**Slave Select I/O** This bit controls the direction of the $\overline{SS}$ pin. In single MASTER Mode, SSIO is set to 1 unless a separate GPIO pin is being used to provide the $\overline{SS}$ output function. In the SPI Slave or multi-Master configuration, SSIO is set to 0. 0 = $\overline{SS}$ pin configured as an input (SPI SLAVE and MULTI-MASTER modes). 1 = $\overline{SS}$ pin configured as an output (SPI SINGLE MASTER Mode).
[0] SSPO	**Slave Select Polarity** This bit controls the polarity of the $\overline{SS}$ pin. 0 = $\overline{SS}$ is active Low. (SSV = 1 corresponds to $\overline{SS}$ = 0). 1 = $\overline{SS}$ is active High. (SSV = 1 corresponds to $\overline{SS}$ = 1).

The first 7 bits transmitted in the first byte are 11110XX. The 2 XX bits are the two most significant bits of the 10-bit address. The lowest bit of the first byte transferred is the Read/Write control bit (which is cleared to 0). The transmit operation is performed in the same manner as 7-bit addressing.

Observe the following steps for a master transmit operation to a 10-bit addressed slave:

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 asserts the TXI bit of the I^2C Control Register to enable transmit interrupts.
3. The I^2C interrupt asserts because the I^2C Data Register is empty.
4. The software responds to the TDRE interrupt by writing the first Slave Address byte (11110xx0). The least-significant bit must be 0 for the write operation.
5. The software asserts the start bit of the I^2C Control Register.
6. The I^2C controller sends a start condition to the I^2C Slave.
7. The I^2C controller loads the I^2C Shift Register with the contents of the I^2C Data Register.
8. After one bit of the address is shifted out by the SDA signal, the transmit interrupt asserts.
9. The software responds by writing the second byte of address into the contents of the I^2C Data Register.
10. The I^2C controller shifts the remainder of the first byte of the address and the Write bit out via the SDA signal.
11. The I^2C slave sends an Acknowledge by pulling the SDA signal Low during the next High period of SCL. The I^2C controller sets the ACK bit in the I^2C Status Register.

    If the slave does not acknowledge the first 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 second address byte from the Data Register, sends a stop condition on the bus and clears the stop and NCKI bits. The transaction is complete and the following steps can be ignored.
12. The I^2C controller loads the I^2C Shift Register with the contents of the I^2C Data Register (2nd address byte).
13. The I^2C controller shifts the second address byte out via the SDA signal. After the first bit has been sent, the transmit interrupt asserts.

The Flash Sector Protect Register can be configured to prevent sectors from being programmed or erased. After a sector is protected, it cannot be unprotected by user code. The Flash Sector Protect Register is cleared after reset and any previously written protection values is lost. User code must write this register in their initialization routine if they want to enable sector protection.

The Flash Sector Protect Register shares its Register File address with the Page Select Register. The Flash Sector Protect Register is accessed by writing the Flash Control Register with 5EH. After the Flash Sector Protect Register is selected, it can be accessed at the Page Select Register address. When user code writes the Flash Sector Protect Register, bits can only be set to 1. Thus, sectors can be protected, but not unprotected, via register write operations. Writing a value other than 5EH to the Flash Control Register deselects the Flash Sector Protect Register and reenables access to the Page Select Register. code:

1. Write 00H to the Flash Control Register to reset the Flash Controller.
2. Write 5EH to the Flash Control Register to select the Flash Sector Protect Register.
3. Read and/or write the Flash Sector Protect Register which is now at Register File address FF9H.
4. Write 00H to the Flash Control Register to return the Flash Controller to its reset state.

The Sector Protect Register is initialized to 0 on Reset, putting each sector into an unprotected state. When a bit in the Sector Protect Register is written to 1, the corresponding sector can no longer be written or erased. After a bit of the Sector Protect Register has been set, it can not be cleared except by a System Reset.

20.2.4. Byte Programming

Flash memory is enabled for byte programming on the active page after unlocking the Flash Controller. Erase the address(es) to be programmed using either the Page Erase or Mass Erase command prior to performing byte programming. An erased Flash byte contains all 1s (FFH). The programming operation can only be used to change bits from 1 to 0. To change a Flash bit (or multiple bits) from 0 to 1 requires execution of either the Page Erase or Mass Erase command.

Byte programming can be accomplished using the On-Chip Debugger's Write Memory command or eZ8 CPU execution of the LDC or LDCI instructions. For a description of the LDC and LDCI instructions, refer to the eZ8 CPU Core User Manual (UM0128), available for download at www.zilog.com. While the Flash Controller programs the contents of Flash memory, the eZ8 CPU idles but the system clock and on-chip peripherals continue to operate.

After a byte is written, the page remains unlocked, allowing for subsequent writes to other bytes on the same page. To exit programming mode and lock Flash memory, write any value to the Flash Control Register except the Mass Erase or Page Erase commands.

On-Chip Debugger. Writing an invalid value or an invalid sequence returns the Flash Controller to its locked state. The write-only Flash Control Register shares its Register File address with the read-only Flash Status Register.

Table 134. Flash Control Register (FCTL)

Bits	7	6	5	4	3	2	1	0
Field	FCMD							
Reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W
Address	FF8H							

Bit	Description
[7:0] FCMD	**Flash Command** 73H = First unlock command. 8CH = Second unlock command. 95H = Page Erase command (must be third command in sequence to initiate Page Erase). 63H = Mass Erase command (must be third command in sequence to initiate Mass Erase). 5EH = Enable Flash Sector Protect Register Access

20.3.2. Flash Status Register

The Flash Status register (Table 135) indicates the current state of the Flash Controller. This register can be read at any time. The read-only Flash Status Register shares its Register File address with the write-only Flash Control Register.

Table 135. Flash Status Register (FSTAT)

Bits	7	6	5	4	3	2	1	0
Field	Program_status		FSTAT					
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	FF8H							

Bit	Description
[7:6] Program_ status	**Indicate the fail or success after Flash Write/Erase** 00 = Success. 10 = Success. 11 = Fail due to low power. 01 = Reserved.

21.2.4.5. Trim Bit Address 0005H

In the Trim Option Bits Register at address 0005H and shown in Table 153, all bits are reserved.

Table 153. Trim Option Bits at 0005H (TVREF)

Bits	7	6	5	4	3	2	1	0
Field	Reserved			Reserved				
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	Information Page Memory 0025H							

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

Bit	Description
[7:5]	Reserved; must be 1.
[4:0]	Reserved; must be 1.

21.2.4.6. Trim Bit Address 0006H

The Trim Option Bits Register at address 0006H, shown in Table 154, governs crystal oscillator trim signals.

Table 154. Trim Option Bits at 0006H (TBG)

Bits	7	6	5	4	3	2	1	0
Field	X1_TRIM				X0_TRIM			
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	Information Page Memory 0026H							

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

Bit	Description
[7:4] X1_TRIM	4-bit trimming signal for the 20M crystal oscillator.
[3:0] X0_TRIM	4-bit trimming signal for the 32K second crystal oscillator.

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.

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.

Chapter 31. Ordering Information

Order the Z8 Encore! XP® F1680 Series from Zilog® using the part numbers listed in Table 209. For more information about ordering, please consult your local Zilog sales office. The Zilog website (www.zilog.com) lists all regional offices and provides additional Z8 Encore! XP product information.

Table 209. Ordering Information for the Z8 Encore! XP F1680 Series of MCUs

Part Number	Flash	Register RAM	Program RAM	NVDS	I^2C	SPI	I/O Lines	Interrupt Vectors	16-Bit Timers w/ PWM	10-Bit A/D Channels	UART with IrDA	Comparator	Temperature Sensor	Multichannel Timer	Description
Z8 Encore! XP F1680 Series with 24KB Flash, 10-Bit Analog-to-Digital Converter															
Standard Temperature: 0°C to 70°C															
Z8F2480SH020SG	24KB	2KB	1KB	0	1	0	17	20	3	7	1	1	1	0	SOIC 20-pin package
Z8F2480HH020SG	24KB	2KB	1KB	0	1	0	17	20	3	7	1	1	1	0	SSOP 20-pin package
Z8F2480PH020SG	24KB	2KB	1KB	0	1	0	17	20	3	7	1	1	1	0	PDIP 20-pin package
Z8F2480SJ020SG	24KB	2KB	1KB	0	1	1	23	21	3	8	1	1	1	0	SOIC 28-pin package
Z8F2480HJ020SG	24KB	2KB	1KB	0	1	1	23	21	3	8	1	1	1	0	SSOP 28-pin package
Z8F2480PJ020SG	24KB	2KB	1KB	0	1	1	23	21	3	8	1	1	1	0	PDIP 28-pin package
Z8F2480PM020SG	24KB	2KB	1KB	0	1	1	33	23	3	8	2	2	1	0	PDIP 40-pin package
Z8F2480AN020SG	24KB	2KB	1KB	0	1	1	37	24	3	8	2	2	1	1	LQFP 44-pin package
Z8F2480QN020SG	24KB	2KB	1KB	0	1	1	37	24	3	8	2	2	1	1	QFN 44-pin package
Extended Temperature: –40°C to 105°C															
Z8F2480SH020EG	24KB	2KB	1KB	0	1	0	17	20	3	7	1	1	1	0	SOIC 20-pin package
Z8F2480HH020EG	24KB	2KB	1KB	0	1	0	17	20	3	7	1	1	1	0	SSOP 20-pin package
Z8F2480PH020EG	24KB	2KB	1KB	0	1	0	17	20	3	7	1	1	1	0	PDIP 20-pin package
Z8F2480SJ020EG	24KB	2KB	1KB	0	1	1	23	21	3	8	1	1	1	0	SOIC 28-pin package
Z8F2480HJ020EG	24KB	2KB	1KB	0	1	1	23	21	3	8	1	1	1	0	SSOP 28-pin package
Z8F2480PJ020EG	24KB	2KB	1KB	0	1	1	23	21	3	8	1	1	1	0	PDIP 28-pin package
Z8F2480PM020EG	24KB	2KB	1KB	0	1	1	33	23	3	8	2	2	1	0	PDIP 40-pin package
Z8F2480AN020EG	24KB	2KB	1KB	0	1	1	37	24	3	8	2	2	1	1	LQFP 44-pin package
Z8F2480QN020EG	24KB	2KB	1KB	0	1	1	37	24	3	8	2	2	1	1	QFN 44-pin package