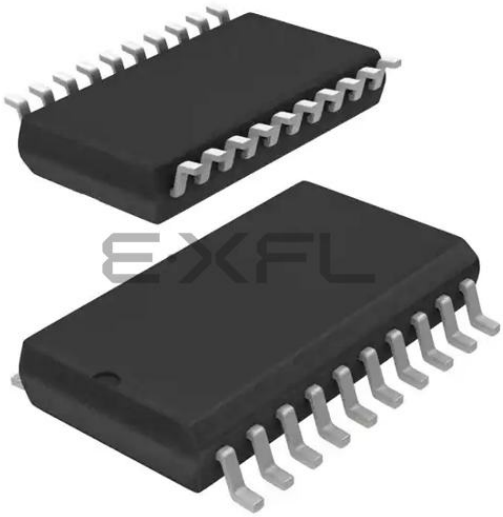

Welcome to E-XFL.COM

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, UART/USART
Peripherals	Brown-out Detect/Reset, LED, LVD, POR, PWM, Temp Sensor, WDT
Number of I/O	17
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 7x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f2480sh020sg

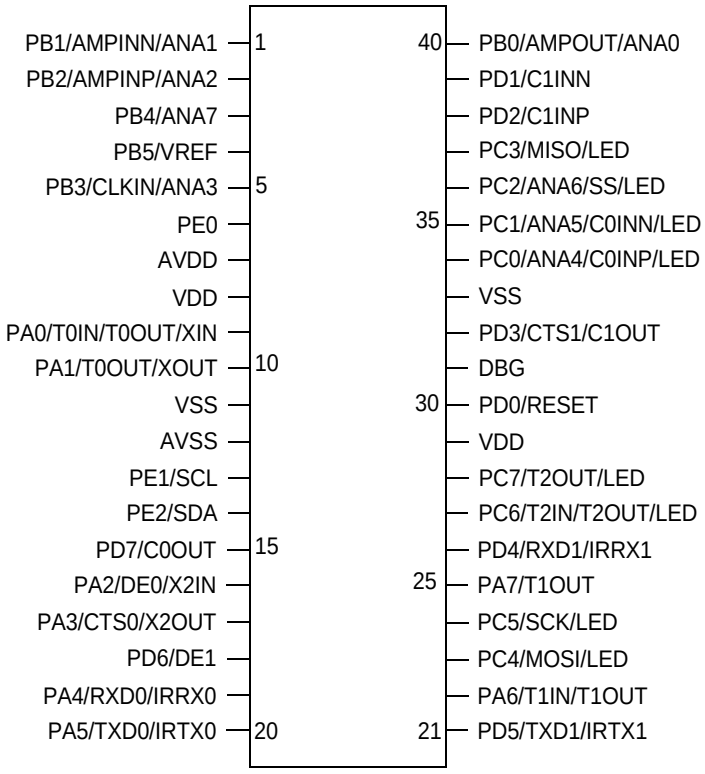

Figure 4. Z8F2480, Z8F1680 and Z8F0880 in 40-Pin Dual Inline Package (PDIP)

Table 8. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)[1]	Page #
F4B	LIN UART1 Control 1—Multiprocessor Control	U1CTL1	00	172
	LIN UART1 Control 1—Noise Filter Control	U1CTL1	00	174
	LIN UART1 Control 1—LIN Control	U1CTL1	00	175
F4C	LIN UART1 Mode Select and Status	U1MDSTAT	00	168
F4D	UART1 Address Compare	U1ADDR	00	177
F4E	UART1 Baud Rate High Byte	U1BRH	FF	177
F4F	UART1 Baud Rate Low Byte	U1BRL	FF	178
I^2C				
F50	I^2C Data	I2CDATA	00	244
F51	I^2C Interrupt Status	I2CISTAT	80	245
F52	I^2C Control	I2CCTL	00	247
F53	I^2C Baud Rate High Byte	I2CBRH	FF	248
F54	I^2C Baud Rate Low Byte	I2CBRL	FF	249
F55	I^2C State	I2CSTATE	02	251
F56	I^2C Mode	I2CMODE	00	252
F57	I^2C Slave Address	I2CSLVAD	00	255
F58-F5F	Reserved	—	XX	
Enhanced Serial Peripheral Interface (ESPI)				
F60	ESPI Data	ESPIDATA	XX	214
F61	ESPI Transmit Data Command	ESPITDCR	00	214
F62	ESPI Control	ESPICTL	00	215
F63	ESPI Mode	ESPIMODE	00	217
F64	ESPI Status	ESPISTAT	01	219
F65	ESPI State	ESPISTATE	00	220
F66	ESPI Baud Rate High Byte	ESPIBRH	FF	220
F67	ESPI Baud Rate Low Byte	ESPIBRL	FF	220
F68–F6F	Reserved	—	XX	

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.

Table 8. Register File Address Map (Continued)

Address (Hex)	Register Description	Mnemonic	Reset (Hex)[1]	Page #
Watchdog Timer				
FF2	Watchdog Timer Reload High Byte	WDTH	FF	143
FF3	Watchdog Timer Reload Low Byte	WDTL	FF	143
FF4–FF5	Reserved	—	XX	
Trim Bit Control				
FF6	Trim Bit Address	TRMADR	00	281
FF7	Trim Data	TRMDR	XX	281
Flash Memory Controller				
FF8	Flash Control	FCTL	00	272
	Flash Status	FSTAT	00	272
FF9	Flash Page Select	FPS	00	273
	Flash Sector Protect	FPROT	00	274
FFA	Flash Programming Frequency High Byte	FFREQH	00	275
FFB	Flash Programming Frequency Low Byte	FFREQL	00	275
eZ8 CPU				
FFC	Flags	—	XX	refer to the eZ8 CPU Core User Manual (UM0128)
FFD	Register Pointer	RP	XX	
FFE	Stack Pointer High Byte	SPH	XX	
FFF	Stack Pointer Low Byte	SPL	XX	

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.

7.9. External Clock Setup

For systems using an external TTL drive, PB3 is the clock source for 20-pin, 28-pin, 40-pin and 44-pin devices. In this case, configure PB3 for alternate function CLKIN. Write to the Oscillator Control Register (see page 320) such that the external oscillator is selected as the system clock.

Table 17. Port Alternate Function Mapping, 20-Pin Parts[1,2]

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Register AFS1
Port A	PA0	T0IN/$\overline{\text{T0OUT}}$	Timer 0 Input/Timer 0 Output Complement	AFS1[0]: 0
		Reserved		AFS1[0]: 1
	PA1	T0OUT	Timer 0 Output	AFS1[1]: 0
		Reserved		AFS1[1]: 1
	PA2	DE0	UART 0 Driver Enable	AFS1[2]: 0
		Reserved		AFS1[2]: 1
	PA3	$\overline{\text{CTS0}}$	UART 0 Clear to Send	AFS1[3]: 0
		Reserved		AFS1[3]: 1
	PA4	RXD0/IRRX0	UART 0/IrDA 0 Receive Data	AFS1[4]: 0
		T2IN/$\overline{\text{T2OUT}}$	Timer 2 Input/Timer 2 Output Complement	AFS1[4]: 1
	PA5	TXD0/IRTX0	UART 0/IrDA 0 Transmit Data	AFS1[5]: 0
		T2OUT	Timer 2 Output	AFS1[5]: 1
	PA6	T1IN/$\overline{\text{T1OUT}}$	Timer 1 Input/Timer 1 Output Complement	AFS1[6]: 0
		SCL	I^2C Serial Clock	AFS1[6]: 1
	PA7	T1OUT	Timer 1 Output	AFS1[7]: 0
		SDA	I^2C Serial Data	AFS1[7]: 1

Notes:
1. Because there are at most two choices of alternate functions for some pins in Port A, 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 pin, the Alternate Function Set registers are not implemented for Port D. 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.

8.2. Architecture

Figure 10 displays the interrupt controller block diagram.

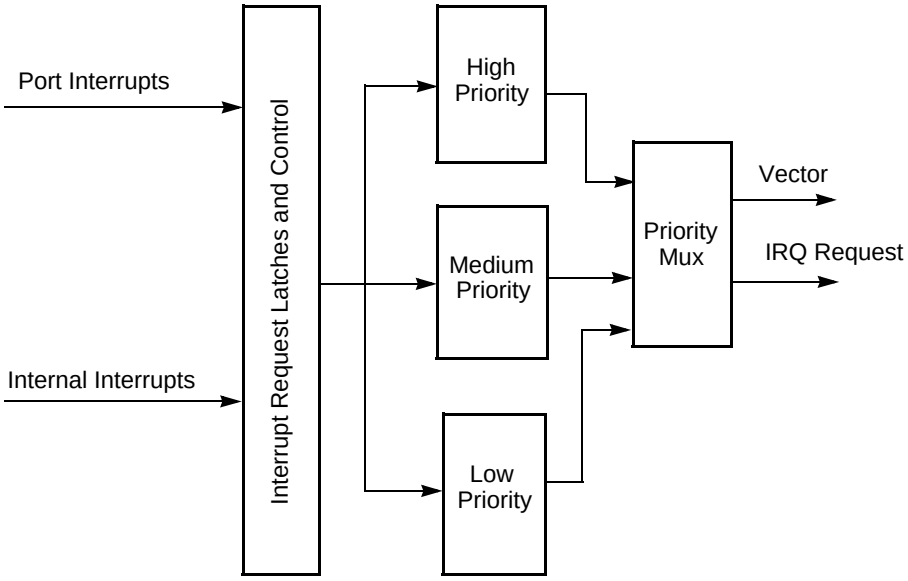

Figure 10. Interrupt Controller Block Diagram

8.3. Operation

This section describes the operational aspects of the following functions.

Master Interrupt Enable: see page 70

Interrupt Vectors and Priority: see page 71

Interrupt Assertion: see page 71

Software Interrupt Assertion: see page 72

8.3.1. Master Interrupt Enable

The master interrupt enable bit (IRQE) in the Interrupt Control Register globally enables and disables interrupts. Interrupts are globally enabled by any of the following actions:

- Execution of an Enable Interrupt (EI) instruction
- Execution of an Interrupt Return (IRET) instruction

Table 45. IRQ1 Enable Low Bit Register (IRQ1ENL)

Bits	7	6	5	4	3	2	1	0
Field	PA7VENL	PA6C0ENL	PA5C1ENL	PAD4ENL	PAD3ENL	PAD2ENL	PAD1ENL	PA0ENL
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	FC5H							

Bit	Description
[7] PA7VENL	Port A Bit[7] or LVD Interrupt Request Enable Low Bit.
[6] PA6C0ENL	Port A Bit[6] or Comparator 0 Interrupt Request Enable Low Bit.
[5] PA5C1ENL	Port A Bit[5] or Comparator 1 Interrupt Request Enable Low Bit.
[4:1] PAD*x*ENL	Port A or Port D Bit[*x*] (*x*=1, 2, 3, 4) Interrupt Request Enable Low Bit.
[0] PA0ENL	Port A Bit[0] Interrupt Request Enable Low Bit.

8.4.6. IRQ2 Enable High and Low Bit Registers

Table 46 describes the priority control for IRQ2. The IRQ2 Enable High and Low Bit registers, shown in Tables 47 and 48 form a priority-encoded enabling for interrupts in the Interrupt Request 2 Register. Priority is generated by setting bits in each register.

Table 46. IRQ2 Enable and Priority Encoding

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

Note: An *x* indicates the register bits from 0–7.

$$\text{PWM Period (s)} = \frac{\text{Reload Value} \times \text{Prescale}}{\text{Timer Clock Frequency (Hz)}}$$

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{Reload Value - PWM Value}}{\text{Reload Value}} \times 100$$

If TPOL is set to 1, the ratio of the PWM output High time to the total period is calculated using the following equation:

$$\text{PWM Output High Time Ratio (\%)} = \frac{\text{PWM Value}}{\text{Reload Value}} \times 100$$

9.2.3.8. CAPTURE Mode

In CAPTURE Mode, the current timer count value is recorded when the appropriate external Timer Input transition occurs. The Capture count value is written to the Timer PWM0 High and Low Byte registers. The Timer counts timer clocks up to the 16-bit reload value. The TPOL bit in the Timer Control 1 Register determines if the Capture occurs on a rising edge or a falling edge of the Timer Input signal. When the Capture event occurs, an interrupt is generated and the timer continues counting. The INPCAP bit in Timer Control 0 Register is set to indicate the timer interrupt is due to an input capture event.

The timer continues counting up to the 16-bit reload value stored in the Timer Reload High and Low Byte registers. Upon reaching the reload value, the timer generates an interrupt and continues counting. The INPCAP bit in Timer Control 0 Register is cleared to indicate the timer interrupt is not due to an input capture event.

Observe the following steps to configure a timer for CAPTURE Mode and initiate the count:

1. Write to the Timer Control 1 Register to:
   - Disable the timer
   - Configure the timer for CAPTURE Mode
   - Set the prescale value
   - Set the Capture edge (rising or falling) for the Timer Input
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.

9.2.3.12. CAPTURE/COMPARE Mode

In CAPTURE/COMPARE Mode, the timer begins counting on the first external Timer Input transition. The appropriate transition (rising edge or falling edge) is set by the TPOL bit in the Timer Control 1 Register. The Timer counts timer clocks up to the 16-bit reload value.

Every subsequent appropriate transition (after the first) of the Timer Input signal captures the current count value. The Capture value is written to the Timer PWM0 High and Low Byte registers. When the Capture event occurs, an interrupt is generated, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. The INPCAP bit in Timer Control 0 Register is set to indicate the timer interrupt is due to an input capture event.

If no Capture event occurs, the timer counts up to the 16-bit Compare value stored in the Timer Reload High and Low Byte registers. Upon reaching the Compare value, the timer generates an interrupt, the count value in the Timer High and Low Byte registers is reset to 0001H and counting resumes. The INPCAP bit in Timer Control 0 Register is cleared to indicate the timer interrupt is not due to an input capture event.

Observe the following steps to configure a timer for CAPTURE/COMPARE Mode and initiate the count:

1. Write to the Timer Control 1 Register to:
   - Disable the timer
   - Configure the timer for CAPTURE/COMPARE Mode
   - Set the prescale value
   - Set the Capture edge (rising or falling) for the Timer Input
2. Write to the Timer High and Low Byte registers to set the starting count value (typically 0001H).
3. Write to the Timer Control 2 Register to choose the timer clock source.
4. Write to the Timer Control 0 Register to set the timer interrupt configuration field TICONFIG.
5. Write to the Timer Reload High and Low Byte registers to set the Compare value.
6. 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.
7. Configure the associated GPIO port pin for the Timer Input alternate function.
8. Write to the Timer Control 1 Register to enable the timer.

- 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:

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

9.2.4. Reading the Timer Count Values

The current count value in the timers can be read while counting (enabled). This capability has no effect on timer operation. When the timer is enabled and the Timer High Byte register is read, the contents of the Timer Low Byte Register are placed in a holding register. A subsequent read from the Timer Low Byte Register returns the value in the holding register. This operation allows accurate reads of the full 16-bit timer count value while enabled. When the timers are not enabled, a read from the Timer Low Byte Register returns the actual value in the counter.

9.2.5. Timer Output Signal Operation

The Timer Output is a GPIO port pin alternate function. Generally, the Timer Output is toggled every time the counter is reloaded.

9.2.6. Timer Noise Filter

A Noise Filter circuit is included which filters noise on a Timer Input signal before the data is sampled by the block.

The Noise Filter has the following features:

- Synchronizes the receive input data to the Timer Clock
- NFEN (Noise Filter Enable) input selects whether the Noise Filter is bypassed (NFEN=0) or included (NFEN=1) in the receive data path
- NFCTL (Noise Filter Control) input selects the width of the up/down saturating counter digital filter. The available widths range from 4 bits to 11 bits
- The digital filter output has hysteresis
- Provides an active Low *saturated state* output (FiltSatB) which is used as an indication of the presence of noise
- Available for operation in STOP Mode

9.2.7. Architecture

Figure 12 displays how the Noise Filter is integrated with the Timer.

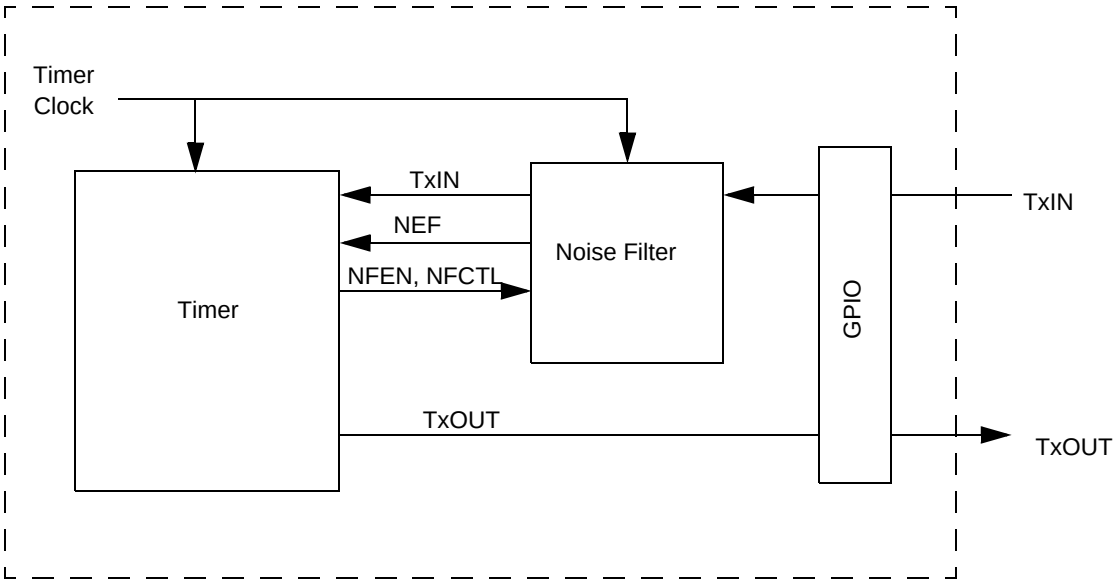

Figure 12. Noise Filter System Block Diagram

9.2.7.1. Operation

Figure 13 displays the operation of the Noise Filter with and without noise. The Noise Filter in this example is a 2-bit up/down counter which saturates at 00 and 11. A 2-bit counter is described for convenience; the operation of wider counters is similar. The output of the filter switches from 1 to 0 when the counter counts down from 01 to 00 and switches from 0 to 1 when the counter counts up from 10 to 11. The Noise Filter delays the receive data by three timer clock cycles.

The NEF output signal is checked when the filtered TxIN input signal is sampled. The Timer samples the filtered TxIN input near the center of the bit time. The NEF signal must be sampled at the same time to detect whether there is noise near the center of the bit time. The presence of noise (NEF = 1 at the center of the bit time) does not mean that the sampled data is incorrect; rather, it is intended to be an indicator of the level of noise in the network.

Table 72. Multi-Channel Timer Subaddress Register (MCTSA)

Bit	7	6	5	4	3	2	1	0
Field	MCTSA							
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	FA4H							

10.7.5. Multi-Channel Timer Subregister x (0, 1, or 2)

The Multi-Channel Timer subregisters 0, 1 or 2 store the 8-bit data write to subregister or 8-bit data read from subregister. The Multi-Channel Timer Subaddress Register selects the subregister to be written to or read from.

Table 73. Multi-Channel Timer Subregister x (MCTSRx)

Bit	7	6	5	4	3	2	1	0
Field	MCTSRx							
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	FA5H, FA6H, FA7H							

10.7.6. Multi-Channel Timer Control 0, Control 1 Registers

The Multi-Channel Timer Control registers (MCTCTL0, MCTCTL1) control Multi-Channel Timer operation. Writes to the PRES field of the MCTCTL1 Register are buffered when TEN = 1 and will not take effect until the next end of the cycle count occurs.

Table 74. Multi-Channel Timer Control 0 Register (MCTCTL0)

Bit	7	6	5	4	3	2	1	0
Field	TCTST	CHST	TCIEN	Reserved	Reserved	TCLKS		
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R	R	R/W	R/W	R/W
Address	See note.							
Note: If a 00H is in the Subaddress Register, it is accessible through Subregister 0.								

interrupts to go inactive until the next address byte. If the new frame's address matches the LIN-UART's, then the data in the new frame is also processed.

The second scheme is enabled by setting MPMD[1:0] to 10B and writing the LIN-UART's address into the LIN-UART Address Compare Register. This mode introduces more hardware control, interrupting only on frames that match the LIN-UART's address. When an incoming address byte does not match the LIN-UART's address, it is ignored. All successive data bytes in this frame are also ignored. When a matching address byte occurs, an interrupt is issued and further interrupts occur on each successive data byte. The first data byte in the frame has NEWFRM=1 in the LIN-UART Status 1 Register. When the next address byte occurs, the hardware compares it to the LIN-UART's address. If there is a match, the interrupt occurs and the NEWFRM bit is set to the first byte of the new frame. If there is no match, the LIN-UART ignores all incoming bytes until the next address match.

The third scheme is enabled by setting MPMD[1:0] to `11B` and by writing the LIN-UART's address into the LIN-UART Address Compare Register. This mode is identical to the second scheme, except that there are no interrupts on address bytes. The first data byte of each frame remains accompanied by a NEWFRM assertion.

12.1.10. LIN Protocol Mode

The Local Interconnect Network (LIN) protocol as supported by the LIN-UART module is defined in rev 2.0 of the LIN Specification Package. The LIN protocol specification covers all aspects of transferring information between LIN Master and Slave devices using *message frames* including error detection and recovery, SLEEP Mode and wake-up from SLEEP Mode. The LIN-UART hardware in LIN mode provides character transfers to support the LIN protocol including break transmission and detection, wake-up transmission and detection and slave autobauding. Part of the error detection of the LIN protocol is for both master and slave devices to monitor their receive data when transmitting. If the receive and transmit data streams do not match, the LIN-UART asserts the PLE bit (physical layer error bit in Status 0 Register). The *message frame* time-out aspect of the protocol depends on software requiring the use of an additional general purpose timer. The LIN mode of the LIN-UART does not provide any hardware support for computing/verifying the checksum field or verifying the contents of the identifier field. These fields are treated as data and are not interpreted by hardware. The checksum calculation/verification can easily be implemented in software via the ADC (Add with Carry) instruction.

The LIN bus contains a single Master and one or more Slaves. The LIN master is responsible for transmitting the message frame header which consists of the Break, Synch and Identifier fields. Either the master or one of the slaves transmits the associated *response* section of the message which consists of data characters followed by a checksum character.

21.1.2. Option Bit Types

This section describes the User, Trim and Calibration option bit types.

21.1.2.1. User Option Bits

The user option bits are contained in the first two bytes of Program Memory. User access to these bits has been provided because these locations contain application-specific device configurations. The information contained here is lost when page 0 of the Program Memory is erased.

21.1.2.2. Trim Option Bits

The trim option bits are contained in the Flash memory information page. These bits are factory programmed values required to optimize the operation of onboard analog circuitry and cannot be permanently altered by the user. Program Memory can be erased without endangering these values. It is possible to alter working values of these bits by accessing the Trim Bit Address and Data registers, but these working values are lost after a power loss.

There are 32 bytes of trim data. To modify one of these values the user code must first write a value between 00H and 1FH into the Trim Bit Address Register. The next write to the Trim Bit Data Register changes the working value of the target trim data byte.

Reading the trim data requires the user code to write a value between 00H and 1FH into the Trim Bit Address Register. The next read from the Trim Bit Data Register returns the working value of the target trim data byte.

Note: The trim address ranges from information address 20-3F only. The remainder of the information page is not accessible via the trim bit address and data registers.

21.1.2.3. Calibration Option Bits

The calibration option bits are also contained in the information page. These bits are factory programmed values intended for use in software correcting the device's analog performance. To read these values, the user code must employ the LDC instruction to access the information area of the address space as defined in the Flash Information Area section on page 21.

The following code example shows how to read the calibration data from the Flash Information Area.

```
; get value at info address 60 (FE60h)
ldx FPS, #%80 ; enable access to flash info page
ld R0, #%FE
```

The Watchdog Timer oscillator failure-detection circuit counts system clocks while looking for a Watchdog Timer clock. The logic counts 8004 system clock cycles before determining that a failure has occurred. The system clock rate determines the speed at which the Watchdog Timer failure can be detected. A very slow system clock results in very slow detection times.

! **Caution:** It is possible to disable the clock failure detection circuitry as well as all functioning clock sources. In this case, the Z8 Encore! XP F1680 Series device ceases functioning and can only be recovered by Power-on reset.

24.2. Peripheral Clock

The peripheral clock is based on a low-frequency/low-power 32kHz secondary oscillator that can be used with an external watch crystal. The peripheral clock is only available for driving Timer and associated noise filter operation. It is not supported for other peripherals. The dedicated peripheral clock source allows Timer operation when the device is in STOP Mode.

Table 169 summarizes peripheral clock source features and usage.

Table 169. Peripheral Clock Source and Usage

Peripheral Clock Source	Characteristics	Required Setup
Secondary Oscillator	• Optimized for use with a 32kHz Watch Crystal • Very high accuracy • Dedicated XTAL pins • No external components	• Unlock and write OSCCTL1 to enable the secondary oscillator • Select the peripheral clock at the timer clock source in the TxCTL2 Register

24.3. Oscillator Control Register Definitions

The Oscillator Control registers enable and disable the various oscillator circuits, enable and disable the failure detection and recovery circuitry, and select the primary oscillator, which becomes the system clock.

24.3.1. Oscillator Control 0 Register

The Oscillator Control 0 Register (OSCCTL0) must be unlocked before writing. Unlock the Oscillator Control 0 Register by writing the two-step sequence of E7H followed by

Table 184. Program Control Instructions

Mnemonic	Operands	Instruction
BRK	—	On-Chip Debugger Break
BTJ	p, bit, src, DA	Bit Test and Jump
BTJNZ	bit, src, DA	Bit Test and Jump if Non-Zero
BTJZ	bit, src, DA	Bit Test and Jump if Zero
CALL	dst	Call Procedure
DJNZ	dst, src, RA	Decrement and Jump Non-Zero
IRET	—	Interrupt Return
JP	dst	Jump
JP cc	dst	Jump Conditional
JR	DA	Jump Relative
JR cc	DA	Jump Relative Conditional
RET	—	Return
TRAP	vector	Software Trap

Table 185. Rotate and Shift Instructions

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

Table 186. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Op Code(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
AND dst, src	dst ← dst AND src	r	r	52	–	*	*	0	–	–	2	3
		r	Ir	53							2	4
		R	R	54							3	3
		R	IR	55							3	4
		R	IM	56							3	3
		IR	IM	57							3	4
ANDX dst, src	dst ← dst AND src	ER	ER	58	–	*	*	0	–	–	4	3
		ER	IM	59							4	3
ATM	Block all interrupt and DMA requests during execution of the next 3 instructions			2F	–	–	–	–	–	–	1	2
BCLR bit, dst	dst[bit] ← 0	r		E2	–	*	*	0	–	–	2	2
BIT p, bit, dst	dst[bit] ← p	r		E2	–	*	*	0	–	–	2	2
BRK	Debugger Break			00	–	–	–	–	–	–	1	1
BSET bit, dst	dst[bit] ← 1	r		E2	–	*	*	0	–	–	2	2
BSWAP dst	dst[7:0] ← dst[0:7]	R		D5	X	*	*	0	–	–	2	2
BTJ p, bit, src, dst	if src[bit] = p PC ← PC + X		r	F6	–	–	–	–	–	–	3	3
			Ir	F7							3	4
BTJNZ bit, src, dst	if src[bit] = 1 PC ← PC + X		r	F6	–	–	–	–	–	–	3	3
			Ir	F7							3	4
BTJZ bit, src, dst	if src[bit] = 0 PC ← PC + X		r	F6	–	–	–	–	–	–	3	3
			Ir	F7							3	4
CALL dst	SP ← SP -2 @SP ← PC PC ← dst	IRR		D4	–	–	–	–	–	–	2	6
		DA		D6							3	3
CCF	C ← ~C			EF	*	–	–	–	–	–-	1	2

Flags notation:
* = Value is a function of the result of the operation.
– = Unaffected.
X = Undefined.
0 = Reset to 0.
1 = Set to 1.

Table 193. Flash Memory Electrical Characteristics and Timing

	VDD = 2.7V to 3.6V TA = 0°C to +70°C TA = –40°C to +105°C				
Parameter	**Min**	**Typ**	**Max**	**Units**	**Conditions**
Flash Byte Read Time	100	–	–	ns	V_{DD} = 1.8 V to 3.6V
Flash Byte Program Time	20	–	40	µs	
Flash Page Erase Time	50	–	–	ms	
Flash Mass Erase Time	50	–	–	ms	
Writes to Single Address Before Next Erase	–	–	2		
Data Retention	20	–	–	years	25°C
Endurance	5,000	–	–	cycles	Program/erase cycles

Table 194. Watchdog Timer Electrical Characteristics and Timing

		VDD = 1.8V to 3.6V TA = 0°C to +70°C TA = –40°C to +105°C				
Symbol	**Parameter**	**Min**	**Typ**	**Max**	**Unit**	**Conditions**
$T_{STARTUP}$		–	–	10	ms	After pd disable only
I_{DD}WDT	WDT Active Current	–	–	5	µA	
I_{DDQ}WDT	WDT Quiescent Current	–	5	–	nA	
F_{WDT}	WDT Oscillator Frequency	2.5	5	20	kHz	

Table 195. Non-Volatile Data Storage

	VDD = 2.7V to 3.6V TA = 0°C to +70°C TA = –40°C to +105°C				
Parameter	**Min**	**Typ**	**Max**	**Units**	**Conditions**
NVDS Byte Read Time	34	–	519	µs	With system clock at 20MHz
NVDS Byte Program Time	0.171	–	39.7	ms	With system clock at 20MHz
Data Retention	20	–	–	years	25°C
Endurance	50,000	–	–	cycles	Cumulative write cycles for entire memory

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	

V

W

X

Z