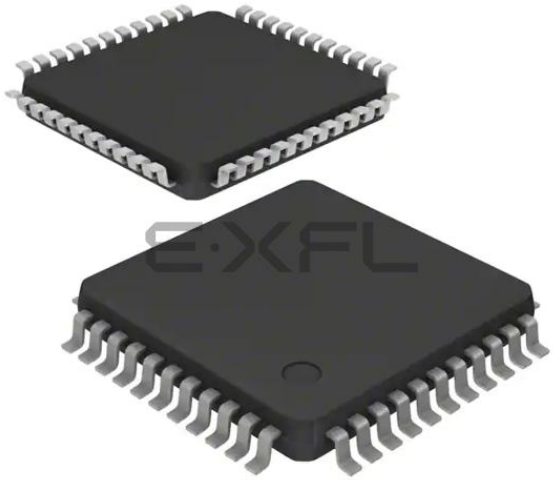

Welcome to E-XFL.COM

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	24MHz
Connectivity	DALI, DMX, I²C, LINbus, SPI, UART/USART, USB
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	36
Program Memory Size	16KB (16K 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 10x12b; D/A 1x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	44-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f1681an024xk

5.3.3. Stop-Mode Recovery Using GPIO Port Pin Transition

Many of the GPIO Port pins can be configured as a Stop-Mode Recovery input source. Which GPIO can be configured as a Stop-Mode Recovery input source is described in the General-Purpose Input/Output chapter on page 55. On any GPIO pin enabled as a Stop-Mode Recovery source, a change in the input pin value (from High to Low or from Low to High) initiates Stop-Mode Recovery. In the Reset Status Register, the STOP bit is set to 1.

If the GPIO is also configured as an interrupt source, an interrupt will occur once interrupts are reenabled.

! Caution: In Stop Mode, the GPIO Port Input Data registers (PxIN) are disabled. The Port Input Data registers record the Port transition only if the signal stays on the Port pin until the end of the Stop-Mode Recovery delay. As a result, short pulses on the Port pin can initiate Stop-Mode Recovery without being written to the Port Input Data Register or without initiating an interrupt (if enabled for that pin).

5.3.4. Stop-Mode Recovery Using External $\overline{\text{RESET}}$ Pin

When the F6482 Series MCU is in Stop Mode and the external $\overline{\text{RESET}}$ pin is driven Low, a System Reset occurs. Because of a glitch filter operating on the $\overline{\text{RESET}}$ pin, the Low pulse must be greater than the minimum width specified, or it is ignored. For details, see the Electrical Characteristics chapter on page 598.

5.4. Low-Voltage Detection

In addition to the VBO Reset described earlier, it is also possible to generate an interrupt when the supply voltage drops below a user-selected value. To learn more about the available Low-Voltage Detection (LVD) threshold levels, see the Flash Option Bits chapter on page 540.

When the supply voltage drops below the LVD threshold, the LVD bit of the RSTSTAT Register is set to 1. This bit remains 1 until the low-voltage condition elapses. Reading or writing this bit does not clear it. The LVD circuit can also generate an interrupt when enabled; see the Interrupt Controller chapter on page 127. The LVD is not latched, so enabling the interrupt is the only way to guarantee detection of a transient low-voltage event.

The LVD circuit is either enabled or disabled by the Power Control Register bit 4. To learn more, see the Power Control Register Definitions section on page 52.

Table 17. Port Alternate Function Mapping, 32-Pin Parts (Continued)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Subregisters
Port B[1]	PB0	Reserved		AFS1[0]: 0
		ANA0/AMPAOUT	ADC Analog Input/Op Amp A Output	AFS1[0]: 1
	PB1	Reserved		AFS1[1]: 0
		ANA1/AMPAINN	ADC Analog Input/Op Amp A Input (N)	AFS1[1]: 1
	PB2	Reserved		AFS1[2]: 0
		ANA2/AMPAINP	ADC Analog Input/Op Amp A Input (P)	AFS1[2]: 1
	PB3	Reserved		AFS1[3]: 0
		V_{REF}–	Voltage Reference (M)	AFS1[3]: 1
	PB4	Reserved		AFS1[4]: 0
		V_{REF}+	Voltage Reference (P)	AFS1[4]: 1
	PB5	Reserved		AFS1[5]: 0
		ANA9	ADC Analog Input	AFS1[5]: 1
	PB6	Reserved		AFS1[6]: 0
		ANA10	ADC Analog Input	AFS1[6]: 1

Notes
1. Because there are at most two choices of alternate function for some pins of Ports A, B, D and E, the Alternate Function Set Subregister AFS2 is not implemented. Additionally, alternate function selection, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.
2. The alternate function selection for Port C, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.

Table 18. Port Alternate Function Mapping (44-Pin Parts)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Subregisters
Port A[1]	PA0	T0IN/$\overline{\text{T0OUT}}$	Timer 0 Input/Timer 0 Output Complement	AFS1[0]: 0
		CLKIN	External Clock Input	AFS1[0]: 1
	PA1	T0OUT	Timer 0 Output	AFS1[1]: 0
		Reserved		AFS1[1]: 1
	PA2	DE0	UART 0 Driver Enable	AFS1[2]: 0
		CLK2IN	External Clock 2 Input	AFS1[2]: 1
	PA3	$\overline{\text{CTS0}}$	UART 0 Clear to Send	AFS1[3]: 0
		Reserved		AFS1[3]: 1
	PA4	RXD0/	UART 0 Receive Data	AFS1[4]: 0
		MOSI0	SPI 0 Master Out/Slave In	AFS1[4]: 1
	PA5	TXD0/	UART 0 Transmit Data	AFS1[5]: 0
		SCK0	SPI 0 Serial Clock	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
Port B[1]	PB0	Reserved		AFS1[0]: 0
		ANA0/AMPAOUT	ADC Analog Input/OpAmp A Output	AFS1[0]: 1
	PB1	Reserved		AFS1[1]: 0
		ANA1/AMPAINN	ADC Analog Input/OpAmp A Input (N)	AFS1[1]: 1
	PB2	Reserved		AFS1[2]: 0
		ANA2/AMPAINP	ADC Analog Input/OpAmp A Input (P)	AFS1[2]: 1
	PB3	Reserved		AFS1[3]: 0
		V_{REF}–	Voltage Reference (M)	AFS1[3]: 1
	PB4	Reserved		AFS1[4]: 0
		V_{REF}+	Voltage Reference (P)	AFS1[4]: 1

Notes
1. Because there are at most two choices of alternate function for some pins of Ports A, B, D and E, the Alternate Function Set register AFS2 is not implemented. Additionally, alternate function selection, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.
2. The alternate function selection for Port C, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.

Table 18. Port Alternate Function Mapping (44-Pin Parts) (Continued)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Subregisters
Port C[2] (cont'd.)	PC6	T2IN/$\overline{\text{T2OUT}}$	Timer 2 Input/Timer2 Output Complement	AFS1[6]: 0, AFS2[6]: 0
		SCKOUT	System Clock Out	AFS1[6]: 1, AFS2[6]: 0
		ESOUT[0]	Event System Output 0	AFS1[6]: 0, AFS2[6]: 1
		Reserved		AFS1[6]: 1, AFS2[6]: 1
	PC7	T2OUT	Timer 2 Output	AFS1[7]: 0, AFS2[7]: 0
		$\overline{\text{CTS1}}$	UART 1 Clear to Send	AFS1[7]: 1, AFS2[7]: 0
		ESOUT[1]	Event System Output 1	AFS1[7]: 0, AFS2[7]: 1
		Reserved		AFS1[7]: 1, AFS2[7]: 1
Port D[1]	PD0	$\overline{\text{RESET}}$	External Reset	AFS1[0]: 0
		Reserved		AFS1[0]: 1
	PD1	C1INN	Comparator 1 Input (N)	AFS1[1]: 0
		ANA7/AMPBINN	ADC Analog Input/OpAmp B Input (N)	AFS1[1]: 1
	PD2	C1INP	Comparator 1 Input (P)	AFS1[2]: 0
		ANA6/AMPBINP	ADC Analog Input/OpAmp B Input (P)	AFS1[2]: 1
	PD3	C1OUT	Comparator 1 Output	AFS1[3]: 0
		ANA8/AMPBOUT	ADC Analog Input/OpAmp B Output	AFS1[3]: 1
	PD4	RXD1	UART 1 Receive Data	AFS1[4]: 0
		Reserved		AFS1[4]: 1
	PD5	TXD1	UART 1 Transmit Data	AFS1[5]: 0
		Reserved		AFS1[5]: 1
	PD6	DE1	UART 1 Driver Enable	AFS1[6]: 0
		Reserved		AFS1[6]: 1
	PD7	C0OUT	Comparator 0 Output	AFS1[7]: 0
		Reserved		AFS1[7]: 1

Notes
1. Because there are at most two choices of alternate function for some pins of Ports A, B, D and E, the Alternate Function Set register AFS2 is not implemented. Additionally, alternate function selection, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.
2. The alternate function selection for Port C, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.

Table 20. Port Alternate Function Mapping (Z8Fxx82 64-Pin Parts) (Continued)

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Subregisters
Port B[1]	PB0	Reserved		AFS1[0]: 0
		ANA0/AMPAOUT	ADC Analog Input/OpAmp A Output	AFS1[0]: 1
	PB1	Reserved		AFS1[1]: 0
		ANA1/AMPAINN	ADC Analog Input/OpAmp A Input (N)	AFS1[1]: 1
	PB2	Reserved		AFS1[2]: 0
		ANA2/AMPAINP	ADC Analog Input/OpAmp A Input (P)	AFS1[2]: 1
	PB3	Reserved		AFS1[3]: 0
		V_{REF}–	Voltage Reference (M)	AFS1[3]: 1
	PB4	Reserved		AFS1[4]: 0
		V_{REF}+	Voltage Reference (P)	AFS1[4]: 1
	PB5	Reserved		AFS1[5]: 0
		ANA9	ADC Analog Input	AFS1[5]: 1
Port C[2]	PC0	Reserved		AFS1[0]: 0, AFS2[0]: 0
		ANA4/VBIAS/ C0INP	ADC or Voltage Bias with low current drive capability or Comparator 0 Input (P)	AFS1[0]: 1, AFS2[0]: 0
		Reserved		AFS1[0]: x, AFS2[0]: 1
	PC1	MISO0	SPI 0 Master In/Slave Out	AFS1[1]: 0, AFS2[1]: 0
		ANA5/C0INN	ADC or Comparator 0 Input (N)	AFS1[1]: 1, AFS2[1]: 0
		Reserved		AFS1[1]: x, AFS2[1]: 1
	PC2	$\overline{SS0}$	SPI 0 Slave Select	AFS1[2]: 0, AFS2[2]: 0
		ANA3	ADC Analog Input	AFS1[2]: 1, AFS2[2]: 0
		Reserved		AFS1[2]: x, AFS2[2]: 1
	PC3	MISO0	SPI 0 Master In/Slave Out	AFS1[3]: 0, AFS2[3]: 0
		ANA11/DAC	ADC or DAC	AFS1[3]: 1, AFS2[3]: 0
		C0OUT	Comparator 0 Output	AFS1[3]: 0, AFS2[3]: 1
		Reserved		AFS1[3]: 1, AFS2[3]: 1

Notes
1. Because there are at most two choices of alternate function for some pins of Ports A, B, D, F, G and H, the Alternate Function Set register AFS2 is not implemented. Additionally, alternate function selection, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.
2. The alternate function selection for Port C, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.
3. Because there is only a single alternate function for each Port J pin, the Alternate Function Set registers are not implemented for Port J. Additionally, alternate function selection, as described in the Port A–J Alternate Function Subregisters (see page 88), must also be enabled.

Table 35. System Clock Configuration and Selection

Clock Source	Characteristics	Required Setup
Digitally Controlled Oscillator (DCO)	Running without FLL	
	• Frequency range 1–24MHz • No external components required	• Unlock the Clock Control registers and configure the DCO • Switch System Clock sources as described in the System Clock Source Switching section on page 100
	Running with FLL	
	• Performs clock multiplication • Frequency range 1–24 MHz • Accuracy 0.25% (vs. PCLK) • No external components required	• Unlock the Clock Control registers • Configure Peripheral Clock (PCLK) • Configure the FLL • Switch System Clock sources as described in the System Clock Source Switching section on page 100
Peripheral Clock	• See Table 36 on page 101	• Unlock the Clock Control registers and configure Peripheral Clock (PCLK) • Switch System Clock sources as described in the System Clock Source Switching section on page 100
High Frequency Crystal/Oscillator (HFXO)	• 1MHz to 24 MHz • Very high accuracy (dependent on crystal or resonator used) • External components required	• Unlock the Clock Control registers and configure the HFXO control bits for correct external oscillator mode • Switch System Clock sources as described in the System Clock Source Switching section on page 100
External Clock Drive	• 0 to 24 MHz • Accuracy dependent on external clock source	• Unlock the Clock Control registers and configure the HFXOBAND for correct external clock drive frequency • Write GPIO registers to configure PA0 pin for external clock function, CLKIN • Apply external clock signal to GPIO • Switch System Clock sources as described in the System Clock Source Switching section on page 100

Bit	Description (Continued)
[3:2] HFXOBAND	**High Frequency Crystal Oscillator (HFXO) Frequency Band** 00: The HFXO or external clock drive is in the 1MHz to 8MHz frequency band. 01: The HFXO or external clock drive is in the > 8MHz to 16MHz frequency band. 10: The HFXO or external clock drive is in the > 16MHz to 24MHz frequency band. 11: Reserved.
[1]	**Reserved** This bit is reserved and must be programmed to 0.
[0] HFXOEN	**High Frequency Crystal Oscillator (HFXO) Enable** 0: HFXO is disabled. 1: HFXO is enabled. In Stop Mode, this bit is overridden such that the HFXO is disabled.

8.11.4. Clock Control 3 Register

The Clock Control 3 (CLKCTL3) Register, shown in Table 41, selects the FLL N-Divider High byte. Before writing CLKCTL3, the clock control registers must be unlocked as described in the Clock System Control Register Unlocking/Locking section on page 103.

Chapter 9. Interrupt Controller

The interrupt controller on the F6482 Series products prioritizes the interrupt requests from the on-chip peripherals and the GPIO port pins. The interrupt controller includes the following features:

- Forty-one interrupt sources using thirty unique interrupt vectors
  - 16 GPIO port pin interrupt sources (nine interrupt vectors are shared, see Table 51)
  - 25 on-chip peripheral interrupt sources (five interrupt vectors are shared, see Table 51)
- Flexible GPIO interrupts
  - Twelve selectable rising and falling edge GPIO interrupts
  - Four dual-edge interrupts
- Three levels of individually programmable interrupt priority
- WDT can be configured to generate an interrupt

Interrupt requests (IRQs) allow peripheral devices to suspend CPU operation in an orderly manner and force the CPU to start an interrupt service routine (ISR). Usually this interrupt service routine is involved with the exchange of data, status information, or control information between the CPU and the interrupting peripheral. When the service routine is completed, the CPU returns to the operation from which it was interrupted.

The eZ8 CPU supports both vectored and polled interrupt handling. For polled interrupts, the interrupt controller has no effect on operation. For more information about interrupt servicing by the eZ8 CPU, refer to the eZ8 CPU Core User Manual (UM0128), which is available free for download from the Zilog website.

9.1. Interrupt Vector Listing

Table 51 lists all of the interrupts available in order of priority. The interrupt vector is stored with the most-significant byte (MSB) at the even Program Memory address and the least-significant byte (LSB) at the following odd Program Memory address.

10.1. Timer Architecture

Figure 17 shows the architecture of the Timer.

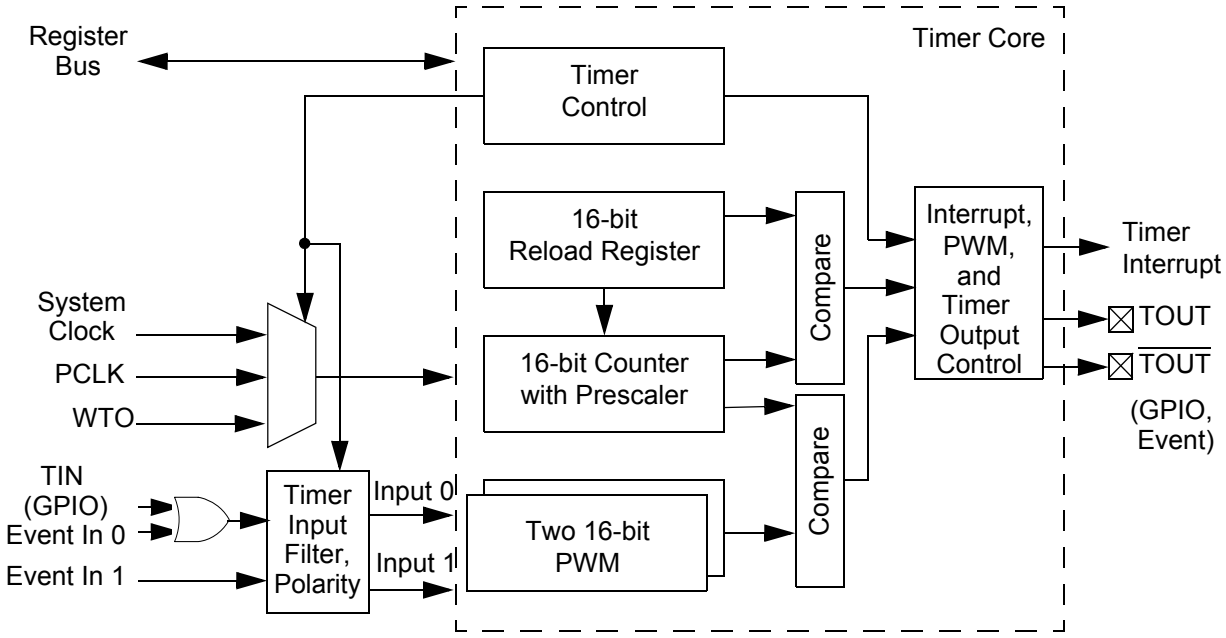

Figure 17. Timer Block Diagram

10.1. Timer Operation

The Timer is a 16-bit up-counter. Minimum time-out delay is set by loading the value 0001h into the Timer Reload High and Low Byte registers and setting the prescale value to 1. Maximum time-out delay is set by loading the value 0000h into the Timer Reload High and Low Byte registers and setting the prescale value to 128. If the Timer reaches FFFFh, the Timer rolls over to 0000h and continues counting.

10.1.1. Timer Clock Source

The timer clock source is selected in TCLKS and can come from either the peripheral clock (PCLK), the Watch-dog Timer Oscillator (WTO), or the System Clock.

For timer operation in Stop Mode, PCLK or the WTO must be selected as the clock source. Either PCLK or the WTO can be selected as source for Active, Halt, and Stop Mode operation. System Clock is only for operation in Active and Halt modes.

Chapter 12. Watchdog Timer

The Watchdog Timer (WDT) function helps protect against corrupted or unreliable software and other system-level problems that can place the F6482 Series MCU into unsuitable operating states. The WDT includes the following features:

- Clocked by the Watchdog Timer Oscillator (WTO)
- A selectable time-out response: System Reset or Interrupt
- 16-bit programmable time-out value

12.1. Operation

The WDT is a retriggerable one-shot timer that resets or interrupts the F6482 Series MCU when the WDT reaches its terminal count. The WDT uses the Watchdog Timer Oscillator (WTO) as its clock source. The WDT has only two modes of operation: ON and OFF. After it is enabled, the WDT always counts and must be retriggered to prevent a time-out. An enable can be performed by executing the WDT instruction or by writing the WDT_AO option bit to 0. When 0, The WDT_AO bit enables the WDT to operate continuously, even if a WDT instruction has not been executed.

The WDT is a 16-bit reloadable downcounter that uses two 8-bit registers in the eZ8 CPU register space to set the reload value. The nominal WDT time-out period is calculated using the following equation:

$$\text{WDT Time–Out Period (ms)} = \frac{\text{WDT Reload Value}}{10}$$

In the above equation, the WDT reload value is computed using {WDTH[7:0], WDTL[7:0]} and the typical Watchdog Timer RC Oscillator frequency is 10 kHz. Users must consider system requirements when selecting the time-out delay. Table 103 indicates the approximate time-out delays for the default and maximum WDT reload values.

Table 103. Watchdog Timer Approximate Time-Out Delays

WDT Reload Value (Hex)	WDT Reload Value (Decimal)	Approximate Time-Out Delay (with 10 kHz Typical WDT Oscillator Frequency)	
		Typical	Description
0400	1024	102 ms	Reset default value time-out delay.
FFFF	65,536	6.55 s	Maximum time-out delay.

13.5.13. Real-Time Clock Alarm Control Register

The RTC_ACTRL Register, shown in Table 118, contains control bits for the Real-Time Clock. This register is cleared by a Power-On Reset (POR).

Table 118. Real-Time Clock Alarm Control Register (RTC_ACTRL)

Bit	7	6	5	4	3	2	1	0
Field when MODE=0	Reserved			ADOM_EN	ADOW_EN	AHRS_EN	AMIN_EN	ASEC_EN
Field when MODE=1	Reserved			ADOM_EN	Reserved	AHRS_EN	AMIN_EN	ASEC_EN
Power-On Reset	0	0	0	0	0	0	0	0
CPU Access	R	R	R	R/W	R/W	R/W	R/W	R/W
Address	F3Ch							

Note: X=undefined; R=read-only; R/W=read/write

Bit	Description
Calendar Mode Operation (MODE=0)	
[7:5]	**Reserved** These bits are reserved and must be programmed to 000.
[4] ADOM_EN	**Alarm Day Of The Month Enable** 0: The day-of-the-month alarm is disabled. 1: The day-of-the-month alarm is enabled.
[3] ADOW_EN	**Alarm Day Of The Week Enable** 0: The day-of-the-week alarm is disabled. 1: The day-of-the-week alarm is enabled.
[2] AHRS_EN	**Alarm Hours Enable** 0: The hours alarm is disabled. 1: The hours alarm is enabled.
[1] AMIN_EN	**Alarm Minutes Enable** 0: The minutes alarm is disabled. 1: The minutes alarm is enabled.
[0] ASEC_EN	**Alarm Seconds Enable** 0: The seconds alarm is disabled. 1: The seconds alarm is enabled.

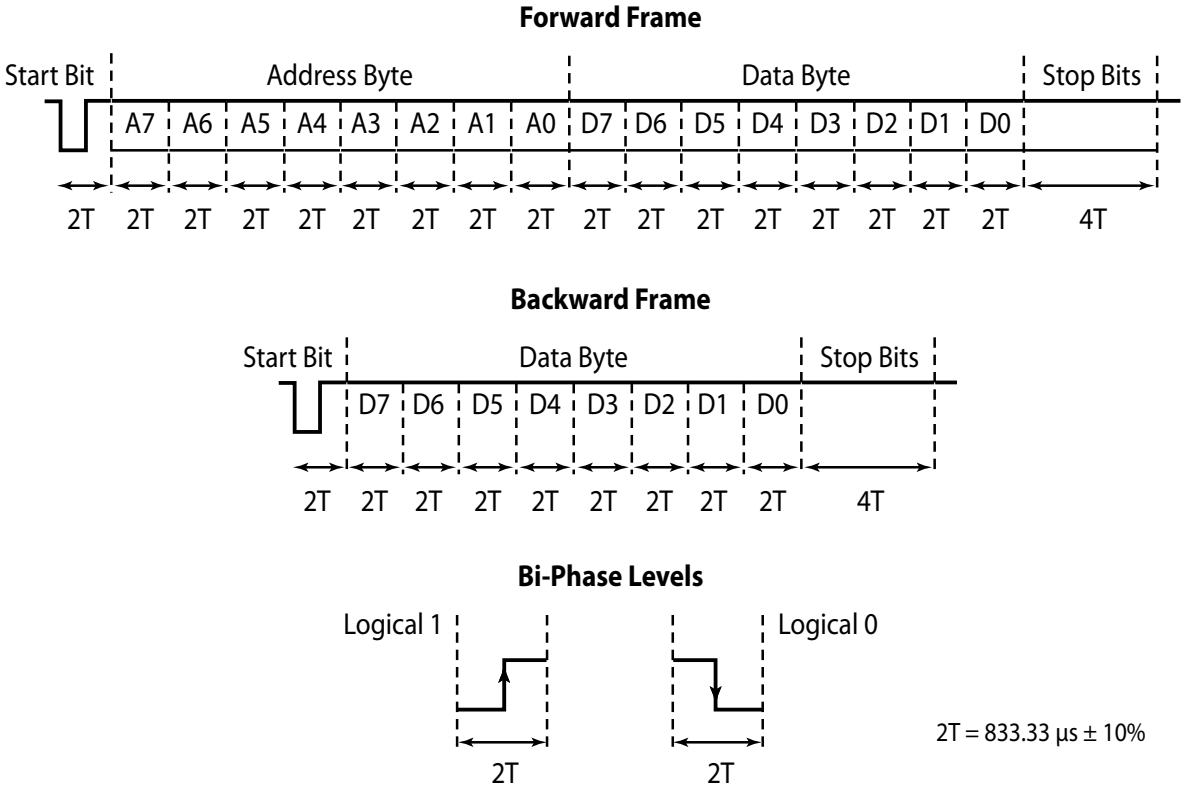

Figure 32. UART-LDD DALI Standard Frames and Biphase Bit Encoding

In DALI Mode, the interrupts defined for normal UART operation still apply, but with the following changes:

- A Parity Error (i.e., the PE bit in the Status 0 Register) is replaced with a biphase error, BPE, which indicates that there was a biphase encode error
- The Break Detect interrupt (i.e., the BRKD bit in the Status 0 Register) is not set
- Framing error checking occurs only for single-byte transfers (i.e, MULTRXE=0 in the DALI Control Register)

14.1.11.1. DALI Clock Requirements

Both the DALI master and DALI slaves are required to have a nominal 1.2 kbit/s bit rate with a tolerance of ±10%.

Before sending/receiving messages, the Baud Reload High/Low registers must be initialized. Unlike standard UART modes, the Baud Reload High/Low registers must be loaded with 1/32 of the baud interval rather than 1/16 of the baud interval.

16.2.6.5. Slave Receive Transaction with 7-Bit Address

The data transfer format for writing data from a master to a slave in 7-bit address mode is shown in Figure 50. The procedure that follows describes the I²C Master/Slave Controller operating as a slave in 7-bit addressing mode and receiving data from the bus master.

S	Slave Address	W=0	A	Data	A	Data	A	Data	A/$\overline{A}$	P/S

Figure 50. Data Transfer Format, Slave Receive Transaction with 7-Bit Address

1. The software configures the controller for operation as a slave in 7-bit addressing mode, as follows:
   a. Initialize the MODE field in the I²C Mode Register for either SLAVE ONLY Mode or MASTER/SLAVE Mode with 7-bit addressing.
   b. Optionally set the GCE bit.
   c. Initialize the SLA[6:0] bits in the I²C Slave Address Register.
   d. Set IEN=1 in the I²C Control Register. Set NAK=0 in the I²C Control Register.

2. The bus master initiates a transfer, sending the address byte. In SLAVE Mode, the I²C controller recognizes its own address and detects that R/$\overline{W}$ bit=0 (written from the master to the slave). The I²C controller acknowledges, indicating it is available to accept the transaction. The SAM bit in the I2CISTAT Register is set to 1, causing an interrupt. The RD bit in the I2CISTAT Register is cleared to 0, indicating a write to the slave. The I²C controller holds the SCL signal Low, waiting for the software to load the first data byte.
3. The software responds to the interrupt by reading the I2CISTAT Register (which clears the SAM bit). After seeing the SAM bit to 1, the software checks the RD bit. Because RD=0, no immediate action is required until the first byte of data is received. If software is only able to accept a single byte, it sets the NAK bit in the I2CCTL Register at this time.
4. The master detects the Acknowledge and sends the byte of data.
5. The I²C controller receives the data byte and responds with an Acknowledge or a Not Acknowledge, depending on the state of the NAK bit in the I2CCTL Register. The I²C 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.

DMAxCTL. Descriptors are generally set up as a list of descriptors, as shown in Table 202.

Table 202. DMA Descriptors

Address	Value	Descriptor #
Start Address	DMAx Source Address High	; Descriptor 0
Start Address + 1	DMAx Source Address Low	
Start Address + 2	DMAx Destination Address High	
Start Address + 3	DMAx Destination Address Low	
Start Address + 4	DMAx Count High	
Start Address + 5	DMAx Count Low	
Start Address + 6	DMAx Control 0	
Start Address + 7	DMAx Control 1	
Start Address + 8	DMAx Source Address High	; Descriptor 1
Start Address + 9	DMAx Source Address Low	
Start Address + 10	DMAx Destination Address High	
Start Address + 11	DMAx Destination Address Low	
Start Address + 12	DMAx Count High	
Start Address + 13	DMAx Count Low	
Start Address + 14	DMAx Control 0	
Start Address + 15	DMAx Control 1	

18.2.6.3. Linked List Control Options

There are a number of control options for linked list operation that affect descriptor usage; these options are selected with the TXLIST bit in DMAxSRCH descriptor byte, and the LLCTL bit in the DMAxCTL0 descriptor byte. The following passages explain the control options.

Transfer In List. The *transfer in list* option is used to load a new linked list descriptor address into the DMAxLAH and DMAxLAL subregisters. If the transfer in list option is selected by setting the TXLIST bit, only the first two descriptor bytes – the source address positions – are read from the descriptor, and the values are transferred to the Linked List Descriptor Address subregisters, DMAxLAH and DMAxLAR.

After this address transfer, the DMA Controller will fetch and execute the descriptor from the new linked list address. The transfer in list option can be used to change to a different list or create a looping operation over a set of descriptors.

Status Write. When LLCTL=00, upon reaching an end-of-count, the DMA Controller will write the DMAxCTL1 Subregister contents back to the current descriptor in the Reg-

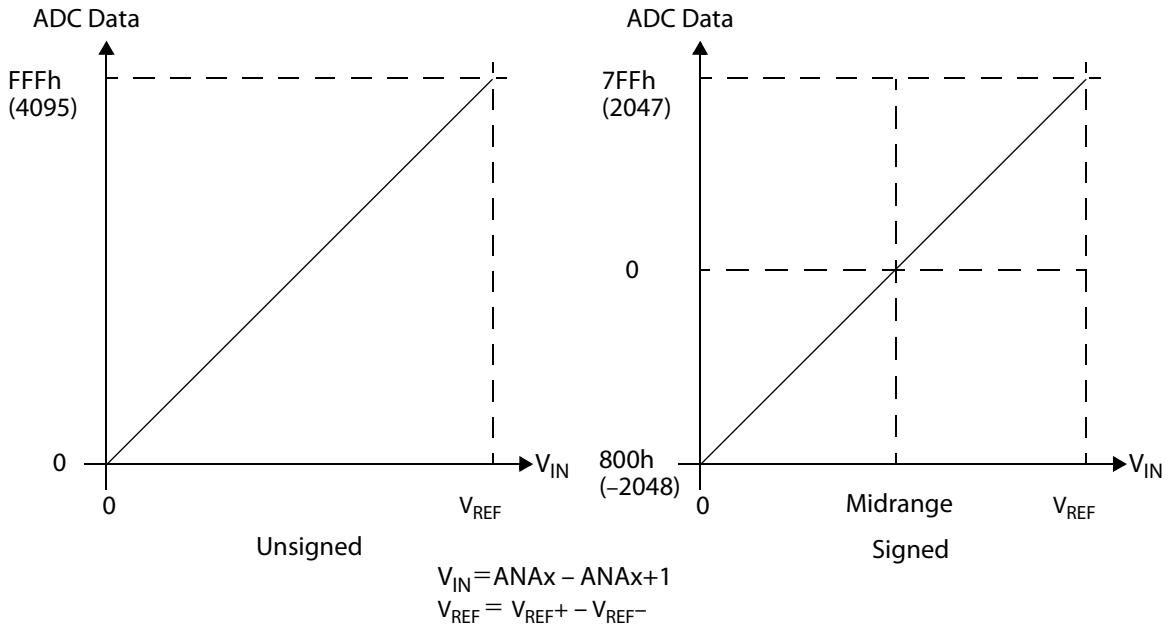

Figure 73. ADC Data (12-Bit) vs. Input Voltage for Unbalanced Differential Input Mode

21.2.3. Conversion Options

Five ADC conversion options are available, and are configured in the ADC Control 0 Register. These conversion options are independent from each other and can be selected in any combination. Furthermore, these conversion options can be enabled for any input mode. Each of these five options is described in the following subsections.

21.2.3.1. Single-Shot or Continuous Conversion

The ADC can be configured for single-shot or continuous conversion. When the CONTCONV bit is cleared, starting the ADC will produce a single result. When CONTCONV is set, starting the ADC will produce a continuous stream of results until CONTCONV is cleared. An interrupt can be generated for each conversion result. The data for each result can be moved by software or DMA.

21.2.3.2. Channel Scanning

The ADC can be configured to automatically scan multiple channels. If SCAN is cleared, channel scanning is not performed, and only the input (or input pair) defined in ANAINL is sampled for conversion.

If the SCAN bit is set, channel scanning is enabled, and the configuration of ANAINL and ANAINH determines which channels are scanned. If the bit corresponding to a particular channel is set, the channel will be included in the scan. Scanning commences with the LSB of ANAINL and completes with the MSB of ANAINH. ADC conversions are per-

The negative Op Amp A inputs are selected with the INNSEL bit in the AMPACTL1 Register, and include:

- GPIO pin used as Op Amp A negative input, AMPAINN
- Op Amp A output through internal feedback network using an internal connection with gain, defined by the GAIN bit
- Op Amp A output, a unity gain configuration using an internal connection

The Op Amp A output, AMPAOUT, can be selected as an internal input to OPAMP B, Comparator 0, Comparator 1, and the ADC. Additionally, it can be connected to the GPIO used as AMPAOUT by setting the OUTCTL bit in the AMPACTL0 Register, and configuring the appropriate alternate function, as described in the General-Purpose Input/Output chapter on page 55. This GPIO can also be selected as an input to the ADC.

23.2.2. Op Amp B

As shown in Figure 81 on page 474, four positive and three negative inputs are available. The positive Op Amp B inputs are selected with the INPSEL bit in the AMPBCTL0 Register, and include:

- 1.0V from the Reference System; see the Comparators and Reference System chapter on page 484 to learn more
- A GPIO pin used as the Op Amp B positive input, AMPBINP
- Op Amp A output. This selection provides an internal connection that does not involve the GPIO used as the Op Amp A output, AMPAOUT
- Internal Programmable Reference 1, with level selected by the PREFLVL bit and source selected by the PREFSRC bit in the CMP1CTL1 Register; see Table 261 on page 496 to learn more

The negative Op Amp B inputs are selected with the INNSEL and MODE bits in the AMPBCTL1 Register, and include:

- A GPIO pin used as the Op Amp B negative input, AMPBINN
- An internal connection from the current drive network
- Op Amp B output, a unity gain configuration using an internal connection

The Op Amp B output, AMPBOUT, can be selected as an internal input to Op Amp A and the ADC. Additionally, it can be connected to the GPIO used as AMPBOUT by selecting OUTCTL=11 in the AMPBCTL0 Register and configuring the appropriate alternate function, as described in the General-Purpose Input/Output chapter on page 55. This GPIO can also be selected as an input to Comparator 0, Comparator 1, and the ADC.

Tables 317 through 324 contain the instructions belonging to each group and the number of operands required for each instruction. Some instructions appear in more than one table, because these instructions should be considered as a subset of more than one category. Within these tables, the source operand is identified as *src*, the destination operand is *dst*, and the condition code is *cc*.

Table 317. Arithmetic Instructions

Mnemonic	Operands	Instruction
ADC	dst, src	Add with Carry
ADCX	dst, src	Add with Carry using Extended Addressing
ADD	dst, src	Add
ADDX	dst, src	Add using Extended Addressing
CP	dst, src	Compare
CPC	dst, src	Compare with Carry
CPCX	dst, src	Compare with Carry using Extended Addressing
CPX	dst, src	Compare using Extended Addressing
DA	dst	Decimal Adjust
DEC	dst	Decrement
DECW	dst	Decrement Word
INC	dst	Increment
INCW	dst	Increment Word
MULT	dst	Multiply
SBC	dst, src	Subtract with Carry
SBCX	dst, src	Subtract with Carry using Extended Addressing
SUB	dst, src	Subtract
SUBX	dst, src	Subtract using Extended Addressing

Table 325. 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
CLR dst	dst ← 00h	R		B0	–	–	–	–	–	–	2	2
		IR		B1							2	3
COM dst	dst ← ~dst	R		60	–	*	*	0	–	–	2	2
		IR		61							2	3
CP dst, src	dst – src	r	r	A2	*	*	*	*	–	–	2	3
		r	Ir	A3							2	4
		R	R	A4							3	3
		R	IR	A5							3	4
		R	IM	A6							3	3
		IR	IM	A7							3	4
CPC dst, src	dst – src – C	r	r	1F A2	*	*	*	*	–	–	3	3
		r	Ir	1F A3							3	4
		R	R	1F A4							4	3
		R	IR	1F A5							4	4
		R	IM	1F A6							4	3
		IR	IM	1F A7							4	4
CPCX dst, src	dst – src – C	ER	ER	1F A8	*	*	*	*	–	–	5	3
		ER	IM	1F A9							5	3
CPX dst, src	dst – src	ER	ER	A8	*	*	*	*	–	–	4	3
		ER	IM	A9							4	3
DA dst	dst ← DA(dst)	R		40	*	*	*	X	–	–	2	2
		IR		41							2	3
DEC dst	dst ← dst – 1	R		30	–	*	*	*	–	–	2	2
		IR		31							2	3
DECW dst	dst ← dst – 1	RR		80	–	*	*	*	–	–	2	5
		IRR		81							2	6
DI	IRQCTL[7] ← 0			8F	–	–	–	–	–	–	1	2

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

Table 346. IPO Electrical Characteristics (Continued)

		V_{DD}=1.8V to 3.6V T_A=–40°C to +85°C				
Symbol	**Parameter**	**Min**	**Typ***	**Max**	**Units**	**Conditions**
F_{IPO}	Output Frequency	–2%	32.768	+2%	kHz	
	Duty Cycle of Output	45		55	%	
T_{WAKE}	Time for Wake up		25		µs	

Note: *Data in the Typical column is from characterization at 3.0V and 25°C. These values are provided for design guidance only and are not tested in production.

33.4.17. High Frequency Crystal Oscillator

Table 347 presents electrical and timing data for the F6482 Series' High Frequency Crystal Oscillator function.

Table 347. High Frequency Crystal Oscillator (HFXO) Characteristics

		T_A=–40°C to +85°C V_{DD}=1.8V to 3.6V				
Symbol	**Parameter**	**Min**	**Typ***	**Max**	**Units**	**Conditions**
I_{DD}HFXO	HFXO Active Supply Current		100		µA	HFXOBAND=00[2]
			150		µA	HFXOBAND=01[2]
			200		µA	HFXOBAND=10[2]
F_{XTAL}	HFXO Frequency	1		8	MHz	HFXOBAND=00
		> 8		16	MHz	HFXOBAND=01
		>16		24	MHz	HFXOBAND=10
gm	Oscillator Transconductance	0.4			mA/V	HFXOBAND=00[2]
		1.0			mA/V	HFXOBAND=01[2]
		2.5			mA/V	HFXOBAND=10[2]
T_{WAKE}	Time for Wake up		2	4	ms	HFXOBAND=00[2]
			1.2	2.4	ms	HFXOBAND=01[2]
			1	2	ms	HFXOBAND=10[2]
	SCKOUT Duty Cycle	40	50	60	%	

Notes:
1. Data in the Typical column is from characterization at 3.0V and 25°C. These values are provided for design guidance only and are not tested in production.
2. C_{LOAD}=12.5 pF.

33.4.23. On-Chip Debugger Timing

Figure 109 and Table 353 provide timing information for the DBG pin. The DBG pin timing specifications assume a 4ns maximum rise and fall time.

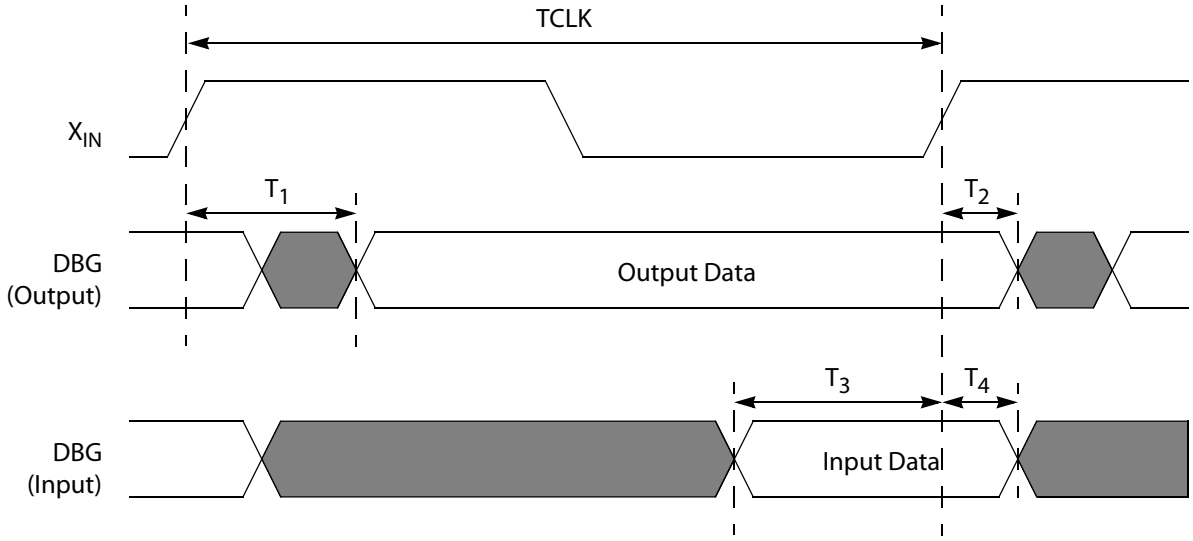

Figure 109. On-Chip Debugger Timing

Table 353. 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 X_{IN} Rise Input Setup Time	5	–
T_{DBGPW}	DBGPulse Width to be Recognized	125	