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, LCD, LVD, POR, PWM, WDT
Number of I/O	67
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 12x12b; D/A 1x12b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	80-LQFP
Supplier Device Package	80-LQFP (10x10)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f1682at024xk

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

Port	Pin	Mnemonic	Alternate Function Description	Alternate Function Set Subregisters
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
		Reserved		AFS1[3]: x, AFS2[3]: 1
	PC4	MOSI0	SPI 0 Master Out/Slave In	AFS1[4]: 0, AFS2[4]: 0
		T0IN/$\overline{T0OUT}$	Timer 0 Input/Timer 0 Output Complement	AFS1[4]: 1, AFS2[4]: 0
		SCL	I^2C Serial Clock	AFS1[4]: 0, AFS2[4]: 1
		DE0	UART 0 Driver Enable	AFS1[4]: 1, AFS2[4]: 1
	PC5	SCK0	SPI 0 Serial Clock	AFS1[5]: 0, AFS2[5]: 0
		T0OUT	Timer 0 Output	AFS1[5]: 1, AFS2[5]: 0
		SDA	I^2C Serial Data	AFS1[5]: 0, AFS2[5]: 1
		$\overline{CTS0}$	UART 0 Clear to Send	AFS1[5]: 1, AFS2[5]: 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 74. Demodulation Mode Initialization Example (Continued)

Register	Value	Comment
T0PWM0H	00h	Initial PWM0 value=0000h
T0PWM0L	00h	
T0PWM1H	00h	Initial PWM1 value=0000h
T0PWM1H	00h	
T0NFC	70h	NFCTL=0111b enables 8-bit up/down Noise Filter counting
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.
PAADDR	07h	Selects Port A Alternate Function Set 1 Register.
PACTL[1:0]	00b	PACTL[0] enables Timer 0 Input Alternate function. PACTL[1] enables Timer 0 Output Alternate function.
ESDADDR	10h	Selects the Timer 0 Input 0 Event System Destination
ESDCTL	00h	Disconnects the Event System Input 0 to Timer 0.
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.

NotesAfter 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 0 rising edge, save the Capture value in the T0PWM0 registers, generate an interrupt, and continue to count.
3. Upon receiving a Timer 0 Input 0 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.

10.1.4. 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.

10.1.5. Timer Interrupts and DMA

The Timer can generate an interrupt request upon reload and capture. In addition, certain input triggering events can generate an interrupt, as described in the Triggered One-Shot Mode section on page 153 and the Dual Input Triggered One-Shot Mode section on

- Timer 0–2 Noise Filter Control Registers – see page 186

10.1.1. Timer 0–2 High and Low Byte Registers

The Timer 0–2 High and Low Byte (T*x*H and T*x*L) registers, shown in Tables 75 and 76, contain the current 16-bit timer count value. When the timer is enabled, a read from T*x*H causes the value in T*x*L to be stored in a temporary holding register. A read from T*x*L always returns this temporary register when the timers are enabled. When the timer is disabled, reading from the T*x*L reads the register directly.

Zilog does not recommend writing to the Timer High and Low Byte registers when the timer is enabled. There are no temporary holding registers available for write operations; therefore simultaneous 16-bit writes are not possible. If either the Timer High or Low Byte registers are written during counting, the 8-bit written value is placed in the counter (High or Low Byte) at the next clock edge. The counter continues counting from the new value.

Table 75. Timer 0–2 High Byte Registers (T*x*H)

Bit	7	6	5	4	3	2	1	0
Field	TH							
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	T0H @ F00h, T1H @ F08h, T2H @ F10h							
Note: *x* references bits in the range [2:0].								

Table 76. Timer 0–2 Low Byte Registers (T*x*L)

Bit	7	6	5	4	3	2	1	0
Field	TL							
Reset	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	T0L @ F01h, T1L @ F09h, T2L @ F11h							
Note: *x* references bits in the range [2:0].								

Bit	Description
[7:0] TH, TL	**Timer High and Low Bytes** These 2 bytes, {TH[7:0], TL[7:0]}, contain the current 16-bit timer count value.

14.1.1. Data Format for Standard UART Modes

The UART-LDD always transmits and receives data in an 8-bit data format with the least significant bit first. An even-or-odd parity bit or multiprocessor address/data bit can be optionally added to the data stream. Each character begins with an active Low start bit and ends with either one or two active High stop bits. Figures 28 and 29 show the asynchronous data format employed by the UART-LDD without parity and with parity, respectively.

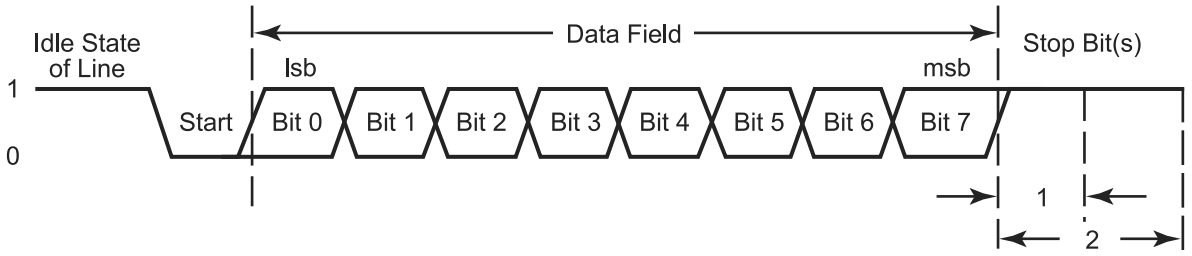

Figure 28. UART-LDD Asynchronous Data Format without Parity

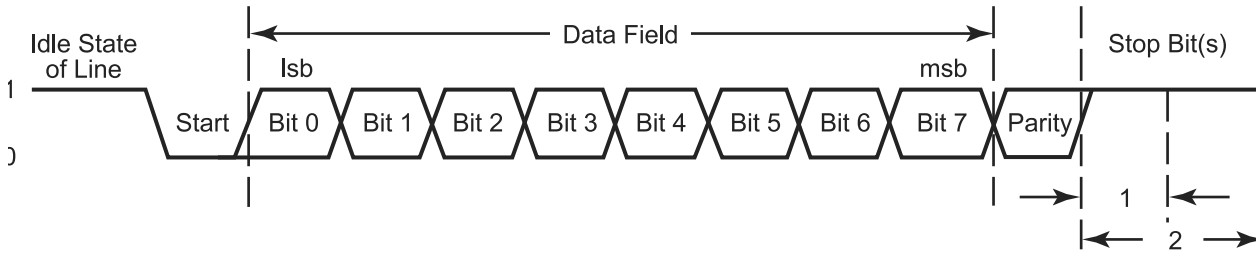

Figure 29. UART-LDD Asynchronous Data Format with Parity

14.1.2. Transmitting Data using the Polled Method

Observe the following steps to transmit data using the polled-operating method:

1. Write to the UART-LDD Baud Rate High and Low Byte registers to set the appropriate baud rate.
2. Enable the UART-LDD pin functions by configuring the associated GPIO port pins for alternate-function operation.
3. If Multiprocessor Mode is appropriate, write to the UART-LDD Control 1 Register to enable Multiprocessor (9-bit) Mode functions.
4. Set the Multiprocessor Mode Select (MPEN) bit to enable Multiprocessor Mode.

14.1.13.3. UART-LDD Overrun Errors

When an overrun error condition occurs, the UART-LDD prevents overwriting of the valid data currently in the Receive Data Register. The break detect and overrun status bits are not displayed until after the valid data has been read.

After the valid data has been read, the OE bit of the Status 0 Register is updated to indicate the overrun condition (and break detect, if applicable). The RDA bit is set to 1 to indicate that the Receive Data Register contains a data byte. However, because the overrun error occurred, this byte cannot contain valid data, and must be ignored. A BRKD bit indicates if the overrun is caused by a break condition on the line. After reading a status byte indicating an overrun error, the Receive Data Register must be read again to clear the error bits in the UART-LDD Status 0 Register.

In LIN Mode, an overrun error is signalled for receive-data overruns as described above, and in the LIN slave if the BRG Counter overflows during the autobaud sequence (the ATB bit will also be set in this case). There is no data associated with the autobaud overflow interrupt; however the Receive Data Register must be read to clear the OE bit. In this case, software must write a `10b` to the LinState field, forcing the LIN slave back to a Wait for Break state.

14.1.13.4. UART-LDD Data- and Error-Handling Procedure

Figures 34 shows the recommended procedure for use in UART-LDD receiver interrupt service routines.

13.3. UART-LDD Control Register Definitions

The UART-LDD control registers support the UART-LDD and the noise filter.

13.3.1. UART-LDD 0–1 Transmit Data Registers

Data bytes written to the UART-LDD 0–1 Transmit Data Registers, shown in Table 121, are shifted out on the TxD pin. This write-only register shares a Register File address with the read-only UART-LDD 0–1 Receive Data Register.

Table 121. UART-LDD 0–1 Transmit Data Registers (U*x*TXD)

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	U0TXD @ F40h, U1TXD @ F48h							

Note: W=Write; X=undefined; *x* = 0,1.

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

13.3.2. UART-LDD 0–1 Receive Data Registers

Data bytes received through the RxD pin are stored in the UART-LDD 0–1 Receive Data Registers, as shown in Table 122. This read-only register shares a Register File address with the write-only UART-LDD 0–1 Transmit Data Register.

Table 122. UART-LDD 0–1 Receive Data Registers (U*x*RXD)

Bit	7	6	5	4	3	2	1	0
Field	RxD							
Reset	X	X	X	X	X	X	X	X
R/W	R	R	R	R	R	R	R	R
Address	U0RXD @ F40h, U1RXD @ F48h							

Note: R=read; X=undefined; *x* = 0,1.

Bit	Description
[7:0] RxD	**Receive Data** UART-LDD receiver data byte from the RxD pin.

15.3.1. Throughput

In Master Mode, the maximum SCK rate supported is one-half the system clock frequency. This frequency is achieved by programming the value 0001h into the Baud Rate High/Low subregister pair. In SPI Master Mode, if the software (or DMA) transfers do not keep up with the SPI baud rate, there will be a pause between characters. In I^2S Master Mode, the transfer will be terminated if new data is not available to send.

In Slave Mode, the transfer rate is controlled by the master. As long as the TDRE and RDRNE interrupt are serviced before the next character transfer completes, the slave will keep up with the master. The master's baud rate should be set for compatibility with all slave devices so that transmit underruns and receive overruns do not occur. In Slave Mode, the baud rate must be restricted to a maximum of one-eighth of the system clock frequency to allow for synchronization of the SCK input to the internal system clock.

15.3.2. ESPI Clock Phase and Polarity Control

The ESPI supports four combinations of serial clock phase and polarity using two bits in the ESPI Control Register. The clock polarity bit, CLKPOL, selects an active High or active Low clock, and has no effect on the transfer format. Table 143 lists the ESPI clock phase and polarity operation parameters. The clock phase bit, PHASE, selects one of two fundamentally different transfer formats. The data is output a half-cycle before the receive clock edge, which provides a half cycle of setup and hold time.

Table 143. ESPI Clock Phase (PHASE) and Clock Polarity (CLKPOL) Operation

PHASE	CLKPOL	SCK Transmit Edge	SCK Receive Edge	SCK Idle State
0	0	Falling	Rising	Low
0	1	Rising	Falling	High
1	0	Rising	Falling	Low
1	1	Falling	Rising	High

15.3.2.1. Transfer Format when Phase Equals Zero

Figure 38 shows a timing diagram for an SPI-type transfer, in which PHASE=0. For SPI transfers, the clock only toggles during a character transfer. The two SCK waveforms show polarity with CLKPOL=0 and CLKPOL=1. The diagram can be interpreted as either a master or slave timing diagram, because the SCK Master-In/Slave-Out (MISO) and Master-Out/Slave-In (MOSI) pins are directly connected between the master and the slave.

Bit	Description (Continued)
[1] TFST	**Transfer Status** 0: No data transfer is currently in progress. 1: Data transfer is currently in progress.
[0] SLAS	**Slave Select** Reading this bit returns the current value of the $\overline{SS}$ pin. 0: The $\overline{SS}$ pin input is Low. 1: The $\overline{SS}$ pin input is High.

15.4.6. ESPI 0-1 State Registers

The ESPI 0-1 State Registers, shown in Table 149, let you observe the ESPI clock, data and internal state.

Table 149. ESPI S0-1 tate Registers (ESPI*x*STATE)

Bit	7	6	5	4	3	2	1	0
Field	SCKI	SDI	ESPISTATE					
Reset	0	0	0					
R/W	R	R	R					
Address	ESPI0STATE @ F65h, ESPI1STATE @ F6Dh							

Note: *x* references bits in the range [1:0].

Bit	Description
[7] SCKI	**Serial Clock Input** This bit reflects the state of the serial clock pin. 0: The SCK input pin is Low. 1: The SCK input pin is High.
[6] SDI	**Serial Data Input** This bit reflects the state of the serial data input (MOSI or MISO depending on the MMEN bit). 0: The serial data input pin is Low. 1: The serial data input pin is High.
[5:0] ESPISTATE	**ESPI State Machine** Indicates the current state of the internal ESPI State Machine. This information is intended for manufacturing test purposes. The state values may change in future hardware revisions and are not intended to be used by a software driver.

16.3.2. I²C Interrupt Status Register

The read-only I²C Interrupt Status Register, shown in Table 155, indicates the cause of any current I²C interrupt and provides the status of the I²C controller. When an interrupt occurs, one or more of the TDRE, RDRF, SAM, ARBLST, SPRS or NCKI bits is set. The GCA and RD bits do not generate an interrupt, but instead provide the status associated with the SAM bit interrupt.

Table 155. I²C Interrupt Status Register (I2CISTAT=F51h)

Bit	7	6	5	4	3	2	1	0
Field	TDRE	RDRF	SAM	GCA	RD	ARBLST	SPRS	NCKI
Reset	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	F51h							

Bit	Description
[7] TDRE	**Transmit Data Register Empty** When the I²C controller is enabled, this bit is 1 when the I²C Data Register is empty. When set, this bit causes the I²C controller to generate an interrupt, except when the I²C controller is shifting in data during the reception of a byte or when shifting an address and the RD bit is set. This bit clears by writing to the I2CDATA Register.
[6] RDRF	**Receive Data Register Full** This bit is set=1 when the I²C controller is enabled and the I²C controller has received a byte of data. When asserted, this bit causes the I²C controller to generate an interrupt. This bit clears by reading the I2CDATA Register.
[5] SAM	**Slave Address Match** This bit is set=1 if the I²C controller is enabled in SLAVE Mode and an address is received that matches the unique slave address or General Call Address (if enabled by the GCE bit in the I²C Mode Register). In 10-bit addressing mode, this bit is not set until a match is achieved on both address bytes. When this bit is set, the RD and GCA bits are also valid. This bit clears by reading the I2CISTAT Register.
[4] GCA	**General Call Address** This bit is set in SLAVE Mode when the General Call Address or start byte is recognized (in either 7 or 10 bit SLAVE Mode). The GCE bit in the I²C Mode Register must be set to enable recognition of the General Call Address and start byte. This bit clears when IEN=0 and is updated following the first address byte of each SLAVE Mode transaction. A General Call Address is distinguished from a start byte by the value of the RD bit (RD=0 for General Call Address, 1 for start byte).
[3] RD	**Read** This bit indicates the direction of transfer of the data. It is set when the master is reading data from the slave. This bit matches the least-significant bit of the address byte after the start condition occurs (for both MASTER and SLAVE modes). This bit clears when IEN=0, and is updated following the first address byte of each transaction.

17.3.11. USB Interrupt Identification Subregister

The USB Interrupt Identification Subregister, shown in Table 180, contains the USB Module interrupt identifier. When the USB Module generates a USB interrupt request, the USBIID Subregister is updated to indicate the source of the interrupt. If more than one USB interrupt request is asserted, the contents of IID reflect the highest priority interrupt based on the order listed in the IID description. To learn more, see the Interrupts section on page 355.

Table 180. USB Interrupt Identification Subregister (USBIID)

Bit	7	6	5	4	3	2	1	0
Field	Reserved	IID					Reserved	
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	If USBSA = 28h in the USB Subaddress Register, accessible through the USB Subdata Register							

Bit	Description
[7]	**Reserved** This bit is reserved and must be programmed to 0.
[6:2] IID	**Interrupt Identification (Source)** 00000: SUDAVIRQ in the USBIRQ Register. Highest priority. 00001: SOFIRQ in the USBIRQ Register. 00010: SUTOKIRQ in the USBIRQ Register. 00011: SUSPIRQ in the USBIRQ Register. 00100: URESIRQ in the USBIRQ Register. 00101: Reserved. 00110: IN0IRQ in the USBINIRQ Register. 00111: OUT0IRQ in the USBOUTIRQ Register. 01000: IN1IRQ in the USBINIRQ Register. 01001: OUT1IRQ in the USBOUTIRQ Register. 01010: IN2IRQ in the USBINIRQ Register. 01011: OUT2IRQ in the USBOUTIRQ Register. 01100: IN3IRQ in the USBINIRQ Register. 01101: OUT3IRQ in the USBOUTIRQ Register. Lowest priority. Others: Reserved.
[1:0]	**Reserved** These bits are reserved and must be programmed to 00.

18.3.1. DMA 0–3 Subaddress/Status Registers

The DMA 0–3 Subaddress/Status registers, shown in Table 204, provide status and select the DMA functionality accessible through the DMA 0–3 subregisters. The DMA 0–3 Subaddress/Status and DMA 0–3 Subdata registers combine to provide access to all DMA controls.

The DMASA bit in the DMAxSA Register is autoincremented whenever DMA*x*SD is accessed (read or written) while DMAx is not active. This autoincrementing allows for convenient channel setup, because software can sequentially write DMA channel subregisters without causing intervening writes to DMAxSA. To take advantage of autoincrementing, software must write the DMA*x*SD values in order, typically from subregister address 0 (DMAxSRCH) to subregister address 7 (DMAxCTL1). When the autoincremented value of DMASA reaches 7, the next access to DMA*x*SD will reset DMASA back to 0.

If the DMASA bit is written with a value greater than 7, the autoincrement function will toggle DMASA[0] upon each DMA*x*SD access for convenient cycling between subregister addresses 8 and 9.

Table 204. DMA 0–3 Subaddress/Status Register (DMA*x*SA)

Bit	7	6	5	4	3	2	1	0
Field	IRQS	Reserved	LLACT	ACT	DMASA			
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
Address	DMA0SA @ FA8h, DMA1SA @ FAAh, DMA2SA @ FACh, DMA3SA @ FAEh							

Note: *x* references bits in the range [3:0].

Bit	Description
[7] IRQS	**Interrupt Request Status** 0: End-of-count interrupt was the most recent DMA interrupt. 1: Watermark interrupt was the most recent DMA interrupt.
[6]	**Reserved** This bit is reserved and must be programmed to 0.
[5] LLACT	**Linked List Active** LLACT is set by hardware when the DMAxLAL Register is written, and is cleared when the DMA Controller has stopped servicing the linked list. 0: No Linked List operation is in progress. 1: A Linked List operation is in progress.

voltage reference buffer, and a 12-bit SAR ADC. The ADC digitizes the signal on a selected channel and stores the digitized data in the ADC data registers. In environments with high electrical noise, an external RC filter must be added at the input pins to reduce high-frequency noise.

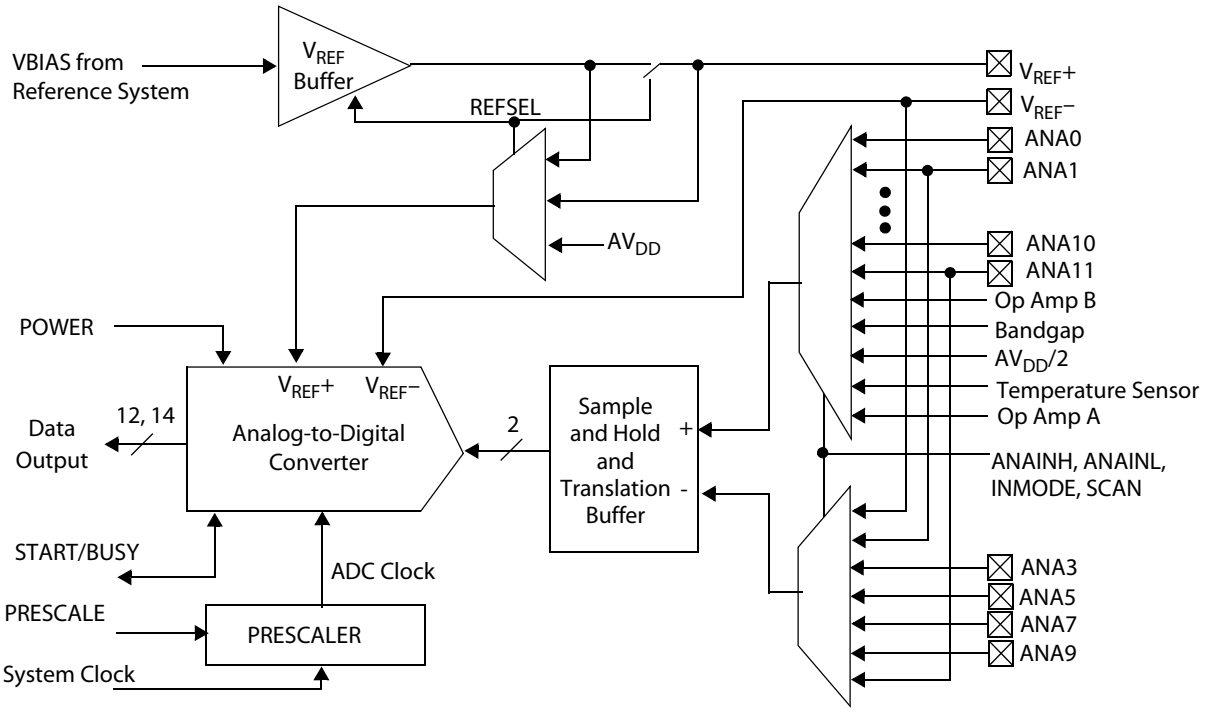

Figure 70. Analog-to-Digital Converter Block Diagram

21.2. Operation

The ADC converts the analog input, ANAx, to a digital representation. The ADC has selectable input modes, resolution, data format, conversion options, power options, window detection, and voltage reference options. The ADC can be serviced by the DMA.

Assuming zero gain and offset errors, any voltage outside the ADC input limits of V_{REF-} and V_{REF+} returns the minimum ADC output code or the maximum ADC output code, respectively.

21.2.2. ADC Data Format

The ADC supports two data formats, unsigned and signed, selected by DFORMAT in the ADC Control 1 Register. When using signed data format, negative values are sign extended. Figures 71 through 73 show the relationship between data formats at 12-bit resolution, and ADC output data for the selectable input modes. The equation for calculating the ADC output data value is a function of input mode, resolution, and data format. The following equations can be used to calculate an ADC output data value for common combinations of input mode, resolution, and data format.

Single-ended input modes (INMODE=00, 11), unsigned (DFORMAT=0):

$$\text{ADC Output} = \text{FSR} \times ((\text{ANAx} - V_{REF-}) \div (V_{REF+} - V_{REF-}))$$

In the equation above, FSR (full-scale range) is 4095 for 12-bit conversions, and 16383 for 2-pass 14-bit conversions.

Balanced differential input mode (INMODE=01), signed (DFORMAT=1):

$$\text{ADC Output} = \text{FSR} \times ((\text{ANAx} - \text{ANAx+1}) \div (V_{REF+} - V_{REF-}))$$

In the equation above, 12-bit conversion FSR (full scale range) is –2048 for negative inputs and +2047 for positive inputs; 2-pass 14-bit conversion FSR is –8192 for negative inputs and +8191 for positive inputs.

Unbalanced differential input mode (INMODE=10), unsigned (DFORMAT=0):

$$\text{ADC Output} = \text{FSR} \times ((\text{ANAx} - \text{ANAx+1}) \div (V_{REF+} - V_{REF-}))$$

In the equation above, FSR (full-scale range) is 4095 for 12-bit conversions and 16383 for 2-pass 14-bit conversions.

Data is always right-justified with 14-bit width even when 12-bit resolution is selected. Conversion resolution can be configured to be 12-bit or 2-pass 14-bit, as defined by the RESOLUT bit in the ADC Control 1 Register. Note that bit 0 of the ACDCTL1 Register must be set for proper ADC operation.

Table 278. 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 Block Protect Register Access.

27.3.2. Flash Status Register

The Flash Status Register, shown in Table 279, indicates the current state of the Flash Controller. This register can be read at any time. The read-only-only Flash Status Register shares its Register File address with the write-only Flash Control Register.

Table 279. Flash Status Register (FSTAT)

Bits	7	6	5	4	3	2	1	0
Field	Reserved					FSTAT		
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	FF8h							

Bit	Description
[7:3	**Reserved** These bits are reserved and must be programmed to 00000.
[2:0] FSTAT	**Flash Controller Status** 000: Flash Controller locked. 001: First unlock command received (73h written). 010: Second unlock command received (8Ch written). 011: Flash Controller unlocked. 100: Block Protect Register selected. 101: Program operation in progress. 110: Page erase operation in progress. 111: Mass erase operation in progress.

Chapter 28. Flash Option Bits

Programmable Flash option bits allow user configuration of certain aspects of F6482 Series MCU operation. The configuration data are stored in Flash Program Memory and are read during Reset. The features available for control through the Flash option bits include:

- Watchdog Timer time-out response selection – interrupt or System Reset
- Watchdog Timer enabled at Reset
- The ability to prevent unwanted read access to user code in Program Memory
- The ability to prevent accidental programming and erasure of all or a portion of the user code in Program Memory
- The VBO can be configured as always enabled, enabled only during Normal and Halt modes to reduce Stop Mode power consumption, or disabled
- LVD voltage threshold selection
- Factory trimming information for multiple analog functions

28.1. Operation

The following sections describe Flash option bit operation.

28.1.1. Option Bit Configuration by Reset

Each time Flash option bits are programmed or erased, the device must be Reset for changes to take effect. During any Reset operation (System Reset or Stop-Mode Recovery), these Flash option bits are automatically read from Flash Program Memory and written to the Option Configuration registers. These Option Configuration registers control operation of the devices within the F6482 Series MCU. Option bit control is established before the device exits System Reset and before the eZ8 CPU begins code execution. The Option Configuration registers are not part of the Register File and are not accessible for read or write access.

28.1.2. Option Bit Types

The following sections describe the option bit types.

28.1.2.1. User Option Bits

The user option bits are contained in the first two bytes of Program Memory. Zilog provides user access to these bits because these locations contain application-specific device

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
DJNZ dst, RA	dst ← dst – 1 if dst ≠ 0 PC ← PC + X	r		0A–FA	–	–	–	–	–	–	2	3
EI	IRQCTL[7] ← 1			9F	–	–	–	–	–	–	1	2
HALT	Halt Mode			7F	–	–	–	–	–	–	1	2
INC dst	dst ← dst + 1	R		20	–	*	*	–	–	–	2	2
		IR		21							2	3
		r		0E–FE							1	2
INCW dst	dst ← dst + 1	RR		A0	–	*	*	*	–	–	2	5
		IRR		A1							2	6
IRET	FLAGS ← @SP SP ← SP + 1 PC ← @SP SP ← SP + 2 IRQCTL[7] ← 1			BF	*	*	*	*	*	*	1	5
JP dst	PC ← dst	DA		8D	–	–	–	–	–	–	3	2
		IRR		C4							2	3
JP cc, dst	if cc is true, PC ← dst	DA		0D–FD	–	–	–	–	–	–	3	2
JR dst	PC ← PC + X	DA		8B	–	–	–	–	–	–	2	2
JR cc, dst	if cc is true, PC ← PC + X	DA		0B–FB	–	–	–	–	–	–	2	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.

Chapter 33. Electrical Characteristics

The data in this chapter has been tabulated prior to qualification (i.e., prequalification) and precharacterization of the F6482 Series product, and is subject to change. Additional electrical characteristics can be found in the individual chapters of this document.

33.1. Absolute Maximum Ratings

Stresses greater than those listed in Table 327 can cause permanent damage to the device. These ratings are stress ratings only. Operation of the device at any condition outside those indicated in the operational sections of these specifications is not implied. Exposure to absolute maximum rating conditions for extended periods can affect device reliability. For improved reliability, tie unused inputs to one of the supply voltages (V_{DD} or V_{SS}).

Table 327. Absolute Maximum Ratings*

Parameter	Min	Max	Units
Ambient temperature under bias	–40	+125	°C
Storage temperature	–65	+150	°C
Voltage on any pin with respect to V_{SS}	–0.3	+4.0	V
Maximum current on input and/or inactive output pin	–5	+5	µA
Maximum output current from active output pin	–25	+25	mA
28-pin Packages Maximum Ratings at –40°C to 85°C			
Total power dissipation		–	mW
Maximum current into V_{DD} or out of V_{SS}		–	mA
32-Pin QFN Maximum Ratings at –40°C to 85°C			
Total power dissipation		–	mW
Maximum current into V_{DD} or out of V_{SS}		–	mA
44-Pin LQFP Maximum Ratings at –40°C to 85°C			
Total power dissipation		–	mW
Maximum current into V_{DD} or out of V_{SS}		–	mA
64-pin LQFP Maximum Ratings at –40°C to 85°C			
Total power dissipation		–	mW
Maximum current into V_{DD} or out of V_{SS}		–	mA

Note: *Operating temperature is specified in the DC Characteristics section.

33.4.10. Reference System

Table 340 presents electrical and timing data for the F6482 Series' Reference System.

Table 340. Reference System Electrical Characteristics

		T_A = –40°C to +85°C V_{DD} = 1.8V to 3.6V				
Symbol	**Parameter**	**Min**	**Typ[1]**	**Max**	**Units**	**Conditions**
V_{PREF}	Programmable Internal Reference Voltage Range	VBIAS/32		VBIAS	V	PREFSRC=0
		V_{DD}/32		V_{DD}	V	PREFSRC=1
$V_{PREFTOL}$	Programmable Internal Reference Voltage Tolerance	Larger of –1.5% or –5mV	V_{PREF}[2]	Larger of 1.5% or 5mV		PREFSRC=0 (VBIAS is source)
		Larger of –0.5% or –5mV	V_{PREF}[2]	Larger of 0.5% or 5mV		PREFSRC=1 (AV_{DD} is source)
V_{FREF}	Fixed Internal Reference Voltage	–1.5%	0.75	+1.5%	V	0.75V fixed reference
		–1.5%	1.0	+1.5%	V	1.0V fixed reference
		–1.5%	1.25	+1.5%	V	1.25V fixed reference
		–1.5%	1.22	+1.5%	V	Bandgap voltage
		–0.5%	V_{DD}/2	+0.5%	V	V_{DD}/2 fixed reference
V_{VBIAS}	VBIAS Reference Voltage	–1.5%	1.25	+1.5%	V	VREFLVL=00
		–1.5%	1.50	+1.5%	V	VREFLVL=01
		–1.5%	2.00	+1.5%	V	VREFLVL=10
		–1.5%	2.50	+1.5%	V	VREFLVL=11
I_{VBIAS}	VBIAS Active Current		2		µA	
I_{VBIAS_OUT}	VBIAS Sourced Reference Current	-20		0	µA	
I_{PROG}	Programmable Reference Active Current		1.5		µA	

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. V_{PREF} is a user-set programmable reference voltage. See Tables 259 and 261 in the Comparators and Reference System chapter on page 484.

Table 358. Package and Pin Count Description

		Pin Count			
		32	**44**	**64**	**80**
Package	QFN	√	√		
	LQFP		√	√	√

35.3. Precharacterization Product

The product represented by this document is newly introduced, and Zilog has not completed the full characterization of the product. This document states all information that Zilog knows about this product at this time, but additional features or nonconformance with some aspects of the document might be found, either by Zilog or its customers, in the course of further application and characterization work. In addition, Zilog cautions that delivery might be uncertain at times, because of start-up yield issues.

S