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	60KB (60K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	3.75K 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
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f6082at024xk

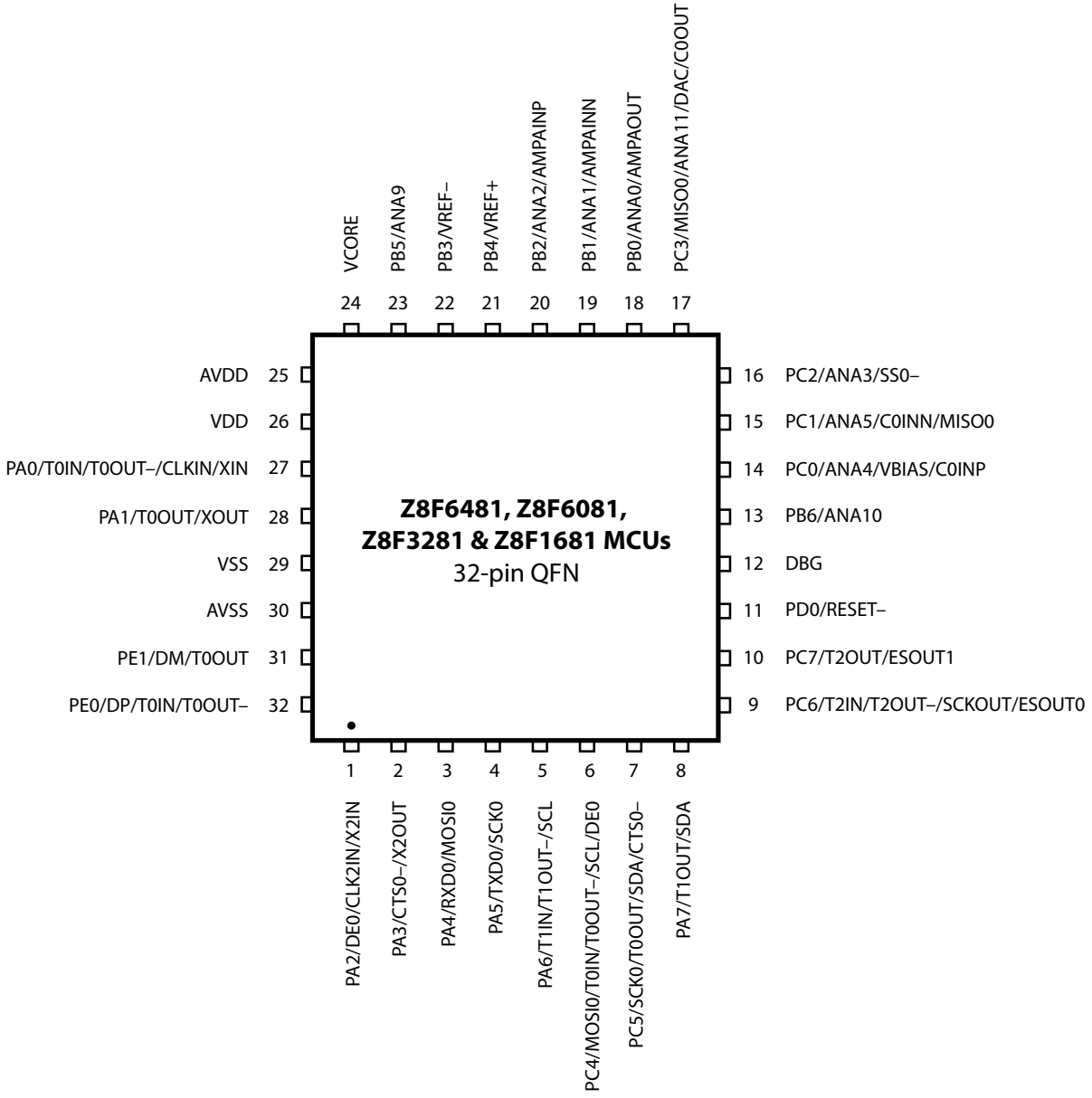

Figure 2. Z8F6481, Z8F6081, Z8F3281 and Z8F1681 MCUs, 32-Pin Quad Flat No Lead (QFN) Package

Note: It is recommended to connect the QFN bottom pad to V_{SS}.

Table 36. Peripheral Clock Sources and Usage

PCLK Sources	Characteristics	Required Setup
Internal Precision Oscillator (IPO)	• 32.768kHz nominal • ± 2% accuracy with factory trim • No external components required	• Unlock the Clock Control registers • Enable the IPO • Switch System Clock sources as described in the PCLK Source Switching section on page 102 • Select PCLK as the clock source for any desired peripheral(s) in the appropriate peripheral control register(s)
External Clock 2 Drive	• 32.768kHz when used as PCLK source • Accuracy dependent on external clock source	• Write GPIO registers to configure PA2 pin for external clock function, CLK2IN • Apply external clock signal to GPIO • Unlock the Clock Control registers • Switch System Clock sources as described in the PCLK Source Switching section on page 102 • Select PCLK as the clock source for any desired peripheral(s) in the appropriate peripheral control register(s)

8.2.2.1. PCLK Source Switching

PCLK source switching provides glitch free operation in that changing the clock source or prescale setting will not cause clock glitches. In accomplishing glitch free clock switching operation, there is a delay from the writing of PCKSEL to the actual switching from the currently active clock source to the new, desired clock source. To switch from one PCLK source to another, observe the following procedure:

1. Unlock the clock control registers.
2. Write to the appropriate clock control registers to configure the desired clock source.
3. Enable the desired clock source.
4. Wait for the newly enabled clock source to stabilize.
5. Write CLKCTL1 to select PCLK.
6. After the System Clock source has been switched, disable any unnecessary clock sources, if desired.
7. The clock control registers can be locked by clearing CSTAT.

8.2.3. PLL Clock Selection

The PLL Clock (PLL_{CLK}) is driven by the on-chip PLL. The two possible sources for the PLL are the High Frequency Crystal Oscillator (HFXO) and the External Clock Drive.

Bit	Description (Continued)
[4:1] PAD*x*ENL	**Port A*x* or Port D*x* Interrupt Request Enable Low Bit** x indicates the specific PAD bit (4–1).
[0] PA0ENL	**Port A0 Interrupt Request Enable Low Bit**

9.4.8. IRQ3 Enable High and Low Bit Registers

Table 65 describes the priority control for IRQ3. The IRQ3 Enable High and Low Bit registers, shown in Tables 66 and 67, form a priority-encoded enabling for interrupts in the Interrupt Request 3 Register. Priority is generated by setting bits in each register.

Table 65. IRQ3 Enable and Priority Encoding

IRQ3ENH[*x*]	IRQ3ENL[*x*]	Priority	Description
0	0	Disabled	Disabled
0	1	Level 1	Low
1	0	Level 2	Nominal
1	1	Level 3	High

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

Table 66. IRQ3 Enable High Bit Register (IRQ3ENH)

Bit	7	6	5	4	3	2	1	0
Field	AESENH	MCTENH	U1RENH	U1TENH	C3ENH/ DMA3ENH	C2ENH/ DMA2ENH	C1ENH	C0ENH
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	FCAh							

Bit	Description
[7] AESENH	**AES Interrupt Request Enable High Bit**
[6] MCTENH	**Multi-Channel Timer Interrupt Request Enable High Bit**
[5] U1RENH	**UART1 Receive Interrupt Request Enable High Bit**

10.1.2. Timer Reload High and Low Byte Registers

The Timer 0–2 Reload High and Low Byte (T*x*RH and T*x*RL) registers, shown in Tables 77 and 78, store a 16-bit reload value, {TRH[7:0], TRL[7:0]}. Values written to these Timer Reload High Byte registers are stored in a temporary holding register. When a write to the Timer Reload Low Byte Register occurs, this temporary holding register value is written to the Timer High Byte Register. This operation allows simultaneous updates of the 16-bit timer reload value.

In Compare Mode, the Timer Reload High and Low Byte registers store the 16-bit Compare value.

Table 77. Timer 0–2 Reload High Byte Registers (T*x*RH)

Bit	7	6	5	4	3	2	1	0
Field	TRH							
Reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	T0RH @ F02h, T1RH @ F0Ah, T2RH @ F12h							

Note: *x* references bits in the range [2:0].

Table 78. Timer 0–2 Reload Low Byte Registers (T*x*RL)

Bit	7	6	5	4	3	2	1	0
Field	TRL							
Reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	T0RL @ F03h, T1RL @ F0Bh, T2RL @ F13h							

Note: *x* references bits in the range [2:0].

Bit	Description
[7:0] TRH, TRL	**Timer Reload Register High and Low** These two bytes form the 16-bit reload value, {TRH[7:0], TRL[7:0]}. This value is used to set the maximum count value which initiates a timer reload to 0001h. In Compare Mode, these two bytes form the 16-bit Compare value.

Bit	Description (Continued)
[6] TPOL (cont'd)	**PWM Dual Output Mode** 0: Timer Output is forced Low (0) and Timer Output Complement is forced High (1) when the timer is disabled. When enabled, the Timer Output is forced High (1) upon PWM count match and forced Low (0) upon Reload. When enabled, the Timer Output Complement is forced Low (0) upon PWM count match and forced High (1) upon Reload. The PWMD field in Timer Control 0 Register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to High (1). 1: Timer Output is forced High (1) and Timer Output Complement is forced Low (0) when the timer is disabled. When enabled, the Timer Output is forced Low (0) upon PWM count match and forced High (1) upon Reload. When enabled, the Timer Output Complement is forced High (1) upon PWM count match and forced Low (0) upon Reload. The PWMD field in Timer Control 0 Register is a programmable delay to control the number of cycles time delay before the Timer Output and the Timer Output Complement is forced to Low (0). **Capture Restart Mode** 0: Count is captured on the rising edge of the Timer Input 0 signal. 1: Count is captured on the falling edge of the Timer Input 0 signal. **Comparator Counter Mode** When the timer is disabled, the Timer Output signal is set to the value of this bit. When the timer is enabled, the Timer Output signal is complemented upon timer reload. **Triggered One-Shot Mode and Dual Input Triggered One-Shot Mode** OUTCTL = 0 0: Timer counting is triggered on the rising edge of the Timer Input 0 signal. 1: Timer counting is triggered on the falling edge of the Timer Input 0 signal. OUTCTL = 1 0: Timer counting is triggered on the rising edge of the Timer Input 0 signal. 1: Reserved. **Demodulation Mode** This functionality applies only if TPOLHI bit in Timer Control 2 Register is 0. If TPOLHI bit is 1 then timer counting is triggered on any edge of the Timer Input 0 signal and the current count is captured on both edges. The current count is captured into PWM0 registers on rising edges and PWM1 registers on falling edges of the Timer Input 0 signal. 0: Timer counting is triggered on the rising edge of the Timer Input 0 signal. The current count is captured into PWM0 High and Low byte registers on subsequent rising edges of the Timer Input 0 signal. 1: Timer counting is triggered on the falling edge of the Timer Input 0 signal. The current count is captured into PWM1 High and Low byte registers on subsequent falling edges of the Timer Input 0 signal.

Subaddress Register, USB Control Register and USB Subdata Register combine to provide write access to all USB Module controls and buffer memory.

This register contains an address in the USB Module memory space selected by the USBCTL Register. For access to endpoint buffer spaces (64 bytes maximum each), bits 7 and 6 of this register are forced to 0 by hardware.

Table 170. USB Subaddress Register (USBSA)

Bit	7	6	5	4	3	2	1	0
Field	ADDRSEL	USBSA						
Reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W1	R/W1	R/W	R/W	R/W1	W	R/W
Address	F59h							

Bit	Description
[7] ADDRSEL	Addressing Select 0: Special Function Register (SFR). 1: Endpoint buffer selected by EPSEL in the USBCTL Register.
[6:0] USBSA	USB Subaddress 00-7F: Selects the USB Subdata Register accessed when USBSD is written. When accessing endpoint buffer memories, if AI=0 in USBCTL, this register auto-increments whenever the USBSD is read (OUT endpoints) or written (IN endpoints) and wraps back to 0 at the address space boundary.

17.3.22. USB OUT 1–3 Control and Status Subregisters

The USB OUT 1–3 Control and Status subregisters, shown in Table 191, provide control and status for USB OUT endpoints 1–3.

Table 191. USB OUT 1–3 Control and Status Subregisters (USBOxCS)

Bit	7	6	5	4	3	2	1	0
Field	Reserved						OUTBUSY	STALL
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W
Address	If USBSA = 46h, 48h, 4Ah in the USB Subaddress Register, accessible through the USB Subdata Register							

Bit	Description
[7:2]	**Reserved** These bits are reserved and must be programmed to 000000.
[1] OUTBUSY	**OUT Busy Status** Software sets OUTBUSY by reloading the USBOxBC Register with a dummy value. While OUTBUSY is set, software should not read the buffer memory for this OUT endpoint. A 1 to 0 transition of OUTBUSY generates a USB interrupt request for the OUT endpoint. 0: Software has control of the OUT endpoint buffer memory. The OUT endpoint buffer memory is ready for reading by software. 1: The USB Module has control of the OUT endpoint buffer memory which is empty and ready to receive the next data packet from the host.
[0] STALL	**Out Stall** 0: Do not send a STALL handshake. 1: Send a STALL handshake for all requests to the endpoint.

19.3. Destination Selection

A variety of peripherals and GPIOs can be selected to be a destination for an Event System channel. Connecting a destination to an Event System channel is controlled in the Event System Destination Channel subregisters, ESDSTxCH. If DSTCON in the ESDSTxCH Subregister is set, the destination is connected to the Event System channel specified in DSTCHSEL; otherwise, the destination is connected to logic 0. Because each destination can connect to any Event System channel, multiple peripherals can be a destination for the same Event System channel.

The ESDSTxCH subregisters are accessed using the Event System Destination Subaddress Register (ESDSA) and the Event System Destination Subdata Register (ESDSD). An ESDSTxCH Subregister is selected by ESDSD in the ESDSA Register, and is accessed by writing/reading the ESDSD bit in the ESDSD Register.

Table 218 lists the Event System destinations.

Table 218. Event System Destinations

Peripheral	Signal Connection
Timer 0	Input 0
	Input 1
Timer 1	Input 0
	Input 1
Timer 2	Input 0
	Input 1
Multi-Channel Timer	Multi-Channel Timer in A
	Multi-Channel Timer in B
	Multi-Channel Timer in C
	Multi-Channel Timer in D
	Multi-Channel Timer in
RTC	RTC Source Select
ADC	Convert
DAC	Convert
Port C	Port C[7:6] pins using ESOUT [1:0]
Port E	Port E[6:3] pins using ESOUT [3:0]
Port F	Port F[3:0] pins using ESOUT [3:0]

Four Event System GPIO outputs, ESOUT[3:0], are available to the port pins listed in Table 218. The GPIO PxAF, PxAFS1, and PxAFS2 subregisters select the Event System

19.6. Event System Register Definitions

Four register addresses provide access to the Event System subregisters that control source and destination selection for each Event System channel. Table 219 lists these Event System registers and subregisters.

Table 219. Event System Registers and Subregisters

Event System Source Selection Registers and Subregisters		
Register Mnemonic	**Address**	**Register Name**
ESSSA	F98h	Event System Source Subaddress Register
ESSSD	F99h	Event System Source Subdata Register
Subregister Mnemonic	**Subregister Address***	**Subregister Name**
ESCHxSRC	0–7h	Event System Channel 0–7 Source subregisters

Event System Destination Selection Registers and Subregisters		
Register Mnemonic	**Address**	**Register Name**
ESDSA	F9Ah	Event System Destination Subaddress Register
ESDSD	F9Bh	Event System Destination Subdata Register
Subregister Mnemonic	**Subregister Address***	**Subregister Name**
ESCHDSTx	0–3Fh	Event System Destination 0–3F Channel subregisters

Note: *The ESSSA Register contains the ESCHxSRC Subregister address; the ESDSA Register contains the ESCHDSTx Subregister address.

For both manual start and auto-start, while the accelerator is processing, the START/BUSY bit remains set. The AES accelerator will detect as an error any attempt to write the AESKEY, AESIV or AESDATA registers during an encryption/decryption operation and access attempts are blocked. The detection of an error does not affect the encryption/decryption operation but ERROR in the AESSTAT Register is set.

The completion of an encryption/decryption operation is indicated by the START/BUSY bit transition from one to zero. An interrupt will be generated if IRQ is set in the AESCTRL Register.

20.2.1. AES Operation and DMA

Two DMA requests are provided: RxIRQ for receive data, and TxIRQ for transmit data. The load status bits used to control DMA receive and transmit data requests. When AESEN is set and KEYLD=1, DATALD=0 and IVLD=1 (if applicable), RxIRQ is asserted for DMA data transfer. When the DMA Controller has written; i.e., 16 bytes to the AESDATA Register, if AUTODIS=0, the AES accelerator will start processing.If AUTODIS=1, processing can be started by setting START/BUSY. When encryption/decryption completes, START/BUSY is cleared, DATALD is cleared and the AES accelerator asserts TxIRQ. When 16 bytes are read from the AESDATA Register, TxIRQ is deasserted and RxIRQ is again asserted. This sequence will continue as long as the DMA Controller provides data to the AES accelerator. The message size is setup in the DMA configuration, as described in the Direct Memory Access Controller chapter on page 389. Table 227 summarizes DMA request conditions.

Table 227. Register Bit Settings for DMA Support, AUTODIS=0

Mode	START/ BUSY	IVLD	KEYLD	DATALD	RxIRQ	TxIRQ
01 (OFB) or 10 (CBC)	0	0	X	X	0	0
	0	0	0	X	0	0
	0	1	1	0	1	0
	1*	1	1	1	0	0
	1→0	1	1	1→0	0	1
00 (ECB)	0	X	0	X	0	0
	0	X	1	0	1	0
	1*	X	1	1	0	0
	1→0	X	1	1	0	1

Note: *If AUTODIS=1, START/BUSY must be set by software.

20.2.2. AES Electronic Codebook (ECB) Mode

The following subsections describe AES ECB Mode.

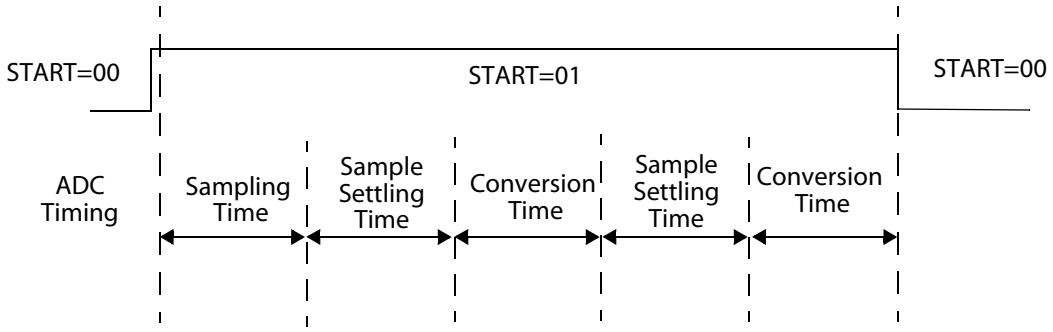

Figure 75. ADC Timing Diagram for 2-Pass 14-Bit Resolution with INMODE=10, 11

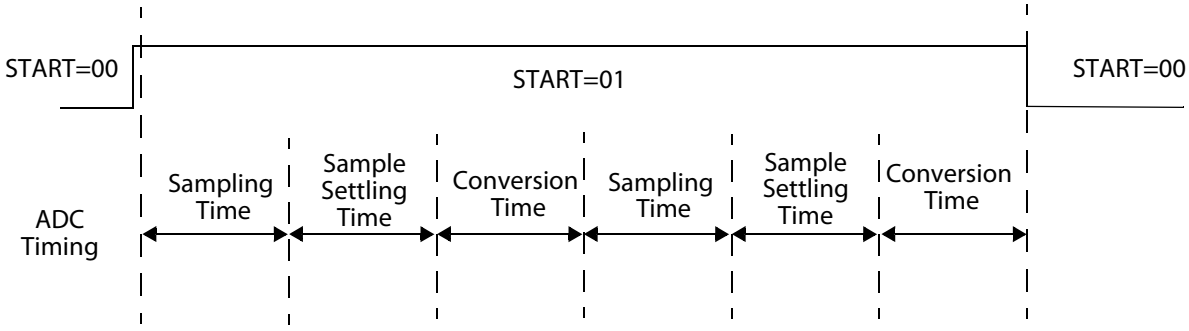

Figure 76. ADC Timing Diagram for 2-Pass 14-Bit Resolution with INMODE=01

21.2.6.3. ADC Wake-up, Sampling, and Settling

As the ADC is designed for low-power applications, the ADC core can be configured to auto-disable when no further conversions are scheduled by clearing ADC to 00 in the PWRCTL1 Register. When triggered to start a new conversion while the ADC is idle and ADC=00, the ADC wake-up period, T_{WAKE_ADC}, is automatically inserted prior to sampling (see the Electrical Characteristics chapter on page 598). If selected as VREF+, the internal voltage reference buffer will also automatically wake-up prior to sampling. When the conversion is completed and ADC = 00, the ADC's auto-disable feature will automatically disable the ADC when no further conversions are scheduled.

If performing multiple sequential conversions due to averaging, scanning or continuous conversion, it is recommended to turn off the ADC auto-disable function by setting ADC to 11 in the PWRCTL1 Register. When ADC is set to 11, the ADC is continuously enabled and T_{WAKE_ADC} is not incurred when the ADC is triggered to perform a conversion. After setting ADC=11, wait T_{WAKE_ADC} prior to triggering an ADC conversion.

21.2.8. ADC Interrupts and DMA

The ADC can generate an interrupt request upon each new ADC result for any completed conversion or calibration (START= 01, 10, 11). The ADC can also generate an interrupt if the conversion result is outside the range defined by the window threshold registers (ADCUWINH, ADCUWINL, ADCLINH, ADCLWINL). Use the IRQ bit in the ADC Control 0 Register to select whether interrupts are generated due to only exceeding the window thresholds or due to both exceeding the window thresholds and end of convert. An interrupt request that is pending when the ADC is disabled or idle (ready) is not automatically cleared.

The ADC will assert a DMA request upon each new ADC result, and will deassert a DMA request whenever the ADCD_L Register (DMACTL=0) or ADCD_H Register (DMACTL=1) is read by the DMA or software. When DMACTL=0, the typical goal is to transfer both data bytes, and the DMA is configured to have fixed word address control for the DMA source address; the source address is configured to be the ADCD_H Register address. When DMACTL=1, the typical goal is to transfer only the most significant data byte, and the DMA is configured to have fixed address control for the DMA source address; the source address is configured to be the ADCD_H Register address.

If starting a new ADC conversion and DMA transfer sequence, reading the ADC_L Register (DMACTL=0) or the ADC_H Register (DMACTL=1) prior to enabling DMA and starting conversion ensures that any residual ADC DMA request from prior ADC activity is deasserted. A DMA request is not asserted upon the completion of an offset or gain calibration.

21.2.9. Calibration and Compensation

Both gain and offset calibration can be performed in situ to achieve even higher accuracy than specified in the Electrical Characteristics chapter on page 598. These calibration operations are performed using the current ADC configuration, as defined by the INMODE, PRESCALE, ST, and SST bits. After these parameters are reconfigured, initiating calibration prior to performing conversions can optimize results. Only initiate calibration when continuous conversion is not selected (i.e., CONTCONV = 0).

Offset calibration is performed when the START bits are written to 10. Prior to initiating offset calibration, 14-bit resolution must be selected by setting RESOLUT = 1. The offset result can be used for both 12-bit and 14-bit resolution conversions. The calibration is complete when START is cleared to 00. The offset calibration result is stored in the OFFSET field in the ADCOFF Register as a two’s-complement value, and is automatically applied to compensate subsequent conversions by hardware. OFFSET can be read by software and can be stored and rewritten any time to the ADCOFF Register to allow consistent usage of offset calibrations. For example, when using multiple input modes, each mode can exhibit a unique offset calibration value.

24.2. Comparator Operation

Two identical general-purpose CMOS analog comparators each provide rail-to-rail operation with four speed-vs.-power settings and three hysteresis options. These comparators are enabled by setting the COMP0 and COMP1 bits in the PWRCTL0 Register, which is described in the Low-Power Modes chapter on page 50. The power setting is determined by the CPOWER bit, which selects current consumption ranging from 27µA, with a propagation delay of 150ns, to 0.2µA, with a propagation delay of 10µs. The low power settings can allow for continuous comparator usage in low-power systems. Hysteresis is selected by the HYST bit; selections range from no hysteresis to 40mV.

A 4-to-1 input multiplexer exists on each comparator positive input and each comparator negative input. The positive input is selected using the INPSEL bit to be either the temperature sensor, GPIO or one of the op amp outputs, AMPAOUT or AMPBOUT. The negative input is selected using the INNSEL and PREFEN bits to be either a GPIO, a fixed reference (0.75V/1.25V), the bandgap voltage, a programmable internal reference or the DAC output. Multiplexing can be configured such that a GPIO (C0INP/C1INP) pin provides the positive comparator input and/or a GPIO (C0INN/C1INN) provides the negative input. When connecting to GPIO, use the appropriate GPIO alternate function selection, as described in the General-Purpose Input/Output chapter on page 55.

The comparator output polarity is determined by the POLSEL bit. When POLSEL=0, the comparator output is noninverted such that the comparator output is High when the positive comparator input voltage is greater than the negative comparator input voltage. When POLSEL=1, the comparator output is inverted such that the comparator output is Low when the positive comparator input voltage is greater than the negative comparator input voltage.

The output of each comparator can be routed to a GPIO pin, C0OUT or C1OUT, as well as to the Event System. When connecting to GPIO, use the appropriate GPIO alternate function selection, as described in the General-Purpose Input/Output chapter on page 55. Additionally, the comparator output state can be read directly from the CSTATUS bit in the COMPCTL Register. An additional output, C01, is the logical OR of each comparator output, and is an Event System source; it is useful for window detection signaling.

A window compare feature provides coordinated detection reporting for the two comparators. WINEN=1 selects Window Mode. The impact of WINEN and POLSEL on the comparator outputs is summarized in Table 256 on page 488.

Table 265. LCD Mode Selection and Corresponding Waveform Characteristics

LCDMODE (LCDCTL2)	# Commons	Duty	Bias	Waveform Type
0000	1	1	Static	Static
0001	2	1/2	1/2	A
0010	2	1/2	1/2	B
0011	2	1/2	1/3	A
0100	2	1/2	1/3	B
0101	3	1/3	1/2	A
0110	3	1/3	1/2	B
0111	3	1/3	1/3	A
1000	3	1/3	1/3	B
1001	4	1/4	1/2	A
1010	4	1/4	1/2	B
1011	4	1/4	1/3	A
1100	4	1/4	1/3	B
Others	Reserved			

26.2.8.1. Static Mode

Static Mode is selected by configuring LCDMODE=0000 in the LCDCTL2 Register. In this mode, only COM0 is used, and each SEGx drives one LCD display segment. Example waveforms for Static Mode are shown in Figure 87.

26.3.3. LCD Clock Register

The LCD Clock Register, shown in Table 269, controls the clocking of the LCD including: clock selection, clock prescale division and frame clock division.

Table 269. LCD Clock Register (LCDCLK)

Bits	7	6	5	4	3	2	1	0
Field	CLKSEL	PRESCALE			FDIV			
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	FB3h							

Bit	Description
[7] CLKSEL	**LCD Clock Selection** 0: PCLK. 1: WTO.
[6:4] PRESCALE	**LCD Clock Prescale Divider** 000: Divide by 1. 001: Divide by 2. 010: Divide by 4. 011: Divide by 8. 100: Divide by 16. 101: Divide by 32. 110: Reserved. 111: Reserved.
[3:0] FDIV	**Frame Divider** 0000: Divide by 8. 0001: Divide by 9. 0010: Divide by 10. 0011: Divide by 11. 0100: Divide by 12. 0101: Divide by 13. 0110: Divide by 14. 0111: Divide by 16. 1000: Divide by 18. 1001: Divide by 20. 1010: Divide by 22. 1011: Divide by 24. 1100: Divide by 26. 1101: Divide by 28. 1110: Divide by 30. 1111: Divide by 32.

Erase operation, the eZ8 CPU remains idle, but the system clock and on-chip peripherals continue to operate. The eZ8 CPU resumes operation after the Page Erase operation completes. If the Page Erase operation is performed using the OCD, poll the Flash Status Register to determine when the Page Erase operation is complete. When the Page Erase is complete, the Flash Controller returns to its locked state.

27.2.8. Mass Erase

Flash memory can also be mass erased using the Flash Controller, but only by using the On-Chip Debugger. Mass erasing Flash memory sets all bytes to the value FFh. With the Flash Controller unlocked, writing the value 63h to the Flash Control Register initiates the Mass Erase operation. While the Flash Controller executes the Mass Erase operation, the eZ8 CPU remains idle, but the system clock and on-chip peripherals continue to operate. Using the On-Chip Debugger, poll the Flash Status Register to determine when the Mass Erase operation is complete. When the Mass Erase is complete, the Flash Controller returns to its locked state.

Mass Erase does not affect the user page in the Flash Information Area. Use Page Erase to erase the user page in the Flash Information Area.

27.2.9. Flash Controller Bypass

The Flash Controller can be bypassed so that the control signals for Flash memory can be brought out to the GPIO pins. Bypassing the Flash Controller allows faster Row Programming algorithms by controlling the Flash programming signals directly.

Zilog recommends row programming for gang programming applications and large-volume customers who do not require the in-circuit initial programming of Flash memory. Mass Erase and Page Erase operations are also supported when the Flash Controller is bypassed.

For more information about bypassing the Flash Controller, please contact Zilog Technical Support.

27.2.10. Flash Controller Behavior in Debug Mode

The following changes in the behavior of the Flash Controller occur when the Flash Controller is accessed using the On-Chip Debugger:

- The Flash Write Protect option bit is ignored
- The Flash Block Protect Register is ignored for programming operations
- Programming operations are not limited to the page selected in the Page Select Register
- Bits in the Flash Block Protect Register can be written to 1 or 0
- The second write of the Page Select Register to unlock the Flash Controller is not necessary

28.3.1. Trim Bit Address Space

All available trim bit addresses and their functions are summarized in Table 288. For details about each, see Tables 290 through 301.

Table 288. Trim Bit Address Description

Address	Function
0000h	Reserved
0001h	Temperature Sensor Trim0
0002h	Temperature Sensor Trim1
0003h	Internal Precision Oscillator
0004h	VBO and LVD
0005h	DAC and ADC/DAC Reference Voltage
0006h	Band-gap reference and Voltage Regulator
0007h	WDT trim
0008h	LCD Trim0
0009h	LCD Trim1
000Ah	Reserved
000Bh	Reserved
000Ch	VBIAS Trim
000Dh–001Fh	Reserved

28.3.1.1. Trim Bit Address Register 0000h

The Trim Option Bits Register at address 0000h, shown in Table 289, is reserved.

Table 289. Trim Option Bits at Address 0000h (TBA0)

Bit	7	6	5	4	3	2	1	0
Field	Reserved							
Reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R
Address	Information Page Memory 0020h							

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

Bit	Description
[7:0]	**Reserved** All bits are reserved.

28.3.1.7. Trim Bit Address 0007h

The Trim Option Bits Register at address 0007h (see Table 297), governs control of the WDT.

Table 297. Trim Option Bits at Address 0007h (TWDT)

Bit	7	6	5	4	3	2	1	0
Field	Reserved		WDT_TRIM					
Reset	0	1	X	X	X	X	X	X
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	Information Page Memory 0027h							

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

Bit	Description
[7]	**Reserved** This bit is reserved and is factory-programmed to 0.
[6]	**Reserved** This bit is reserved and is factory-programmed to 1.
[5:0] WDT_TRIM	**WDT Trim**

28.3.1.8. Trim Bit Addresses 0008h and 0009h

The Trim Option Bits registers at addresses 0008h and 00009h, shown in Tables 298 and 299, govern control of the LCD trim bits.

Table 298. Trim Option Bits at Address 0008h (TLCD0)

Bit	7	6	5	4	3	2	1	0
Field	Reserved	LCDCO_TRIM			LCDVREG_TRIM			
Reset	0	X	X	X	X	X	X	X
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Address	Information Page Memory 0028h							

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

Bit	Description
[7]	**Reserved** This bit is reserved.
[6:4] LCDCO_TRIM	**LCD Contrast Offset Trim**
[3:0] LCDVREG_TRIM	**LCD Voltage Regulator and Current Bias Trim**

Table 337. Analog-to-Digital Converter Electrical Characteristics and Timing (Continued)

		V_{DD}=1.8V to 3.6V T_A=–40°C to +85°C				
Symbol	**Parameter**	**Min**	**Typ***	**Max**	**Units**	**Conditions**
	Gain Error			±4	LSB	RESOLUT=0, No gain calibration run
				–	LSB	RESOLUT=0, Gain calibration run
				±16	LSB	RESOLUT=1, No gain calibration run
				–	LSB	RESOLUT=1, Gain calibration run
	Offset Error			±3	LSB	RESOLUT=0, No offset calibration run
				±1	LSB	RESOLUT=0, Offset calibration run
				±12	LSB	RESOLUT=1, No offset calibration run
				–	LSB	RESOLUT=1, Offset calibration run
I_{DD}ADCI	ADC Active Current with Internal Reference (REFSEL=1x)		180		µA	INMODE=0x, POWER=00
			230		µA	INMODE=1x, POWER=00
			95		µA	INMODE=0x, POWER=10
			115		µA	INMODE=1x, POWER=10
I_{DD}ADCE	ADC Active Current with External Reference or V_{DD} Reference (REFSEL=0x) including I_{DD}VEXT		155		µA	INMODE=0x, POWER=00
			205		µA	INMODE=1x, POWER=00
			85		µA	INMODE=0x, POWER=10
			105		µA	INMODE=1x, POWER=10

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. T_S is applied twice if INMODE=10 and RESOLUT=1.
3. T_{SS} is applied twice if RESOLUT=1.

Figure 111 and Table 355 provide timing information for the UART pins for situations in which CTS is not used for flow control. DE asserts after the Transmit Data Register has been written. DE remains asserted for multiple characters as long as the Transmit Data Register is written with the next character before the current character has completed.

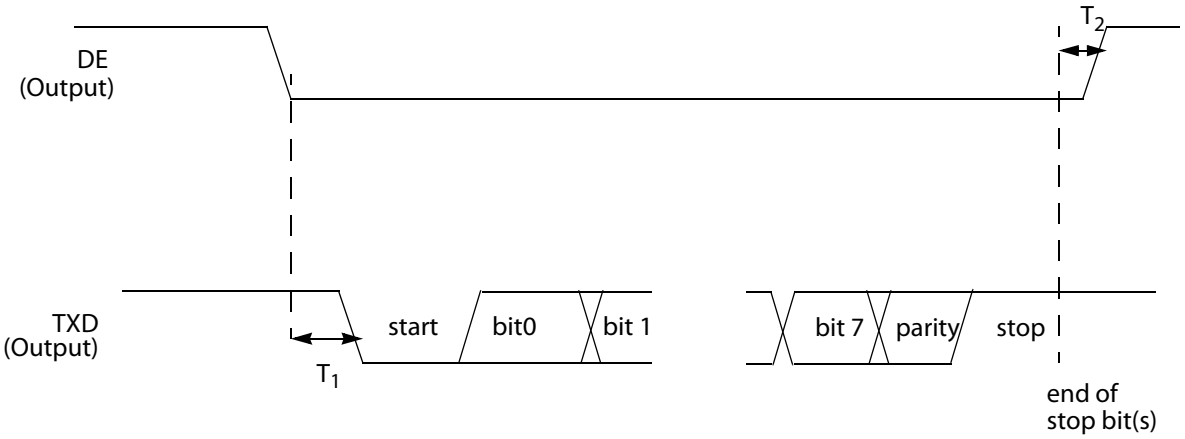

Figure 111. UART Timing Without CTS

Table 355. UART Timing Without CTS

		Delay (ns)	
Parameter	**Abbreviation**	**Min**	**Max**
T_1	DE assertion to TXD falling edge (start bit) delay	1 * SYSCLK period	1 bit time
T_2	End of Stop Bit(s) to DE deassertion delay (Tx Data Register is empty)	± 5	