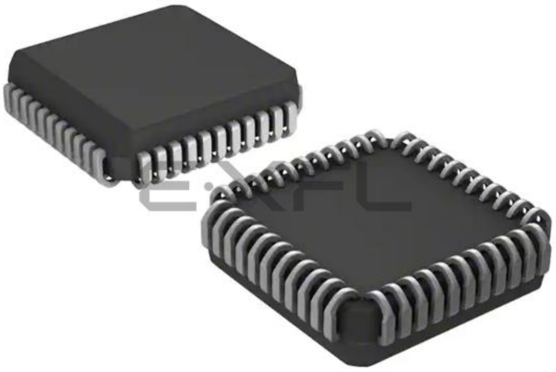

Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	eZ8
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, DMA, POR, PWM, WDT
Number of I/O	31
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LCC (J-Lead)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/z8f3221vn020sc00tr

Timer 1 PWM High Byte
T1PWMH (F0CH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 1 PWM value [15:8]

Timer 1 PWM Low Byte
T1PWML (F0DH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 1 PWM value [7:0]

Timer 1 Control 0
T1CTL0 (F0EH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Reserved
- Cascade Timer
  0 = Timer 1 Input signal is GPIO pin
  1 = Timer 1 Input signal is Timer 0 out
- Reserved

Timer 1 Control 1
T1CTL1 (F0FH - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer Mode
  000 = One-Shot mode
  001 = CONTINUOUS mode
  010 = COUNTER mode
  011 = PWM mode
  100 = CAPTURE mode
  101 = COMPARE mode
  110 = GATED mode
  111 = Capture/COMPARE mode
- Prescale Value
  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 = Divide by 64
  111 = Divide by 128
- Timer Input/Output Polarity
  Operation of this bit is a function of the current operating mode of the timer
- Timer Enable
  0 = Timer is disabled
  1 = Timer is enabled

Timer 2 High Byte
T2H (F10H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 2 current count value [15:8]

Timer 2 Low Byte
T2L (F11H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 2 current count value [7:0]

Timer 2 Reload High Byte
T2RH (F12H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 2 reload value [15:8]

Timer 2 Reload Low Byte
T2RL (F13H- Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 2 reload value [7:0]

Timer 2 PWM High Byte
T2PWMH (F14H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 2 PWM value [15:8]

Timer 2 PWM Low Byte
T2PWML (F15H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Timer 2 PWM value [7:0]

Timer 2 Control 0
T2CTL0 (F16H - Read/Write)

D7 D6 D5 D4 D3 D2 D1 D0

- Reserved
- Cascade Timer
  0 = Timer 2 Input signal is GPIO pin
  1 = Timer 2 Input signal is Timer 1 out
- Reserved

Flags
FLAGS (FFC - Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

- D0: F1 - User Flag 1
- D1: F2 - User Flag 2
- D2: H - Half Carry
- D3: D - Decimal Adjust
- D4: V - Overflow Flag
- D5: S - Sign Flag
- D6: Z - Zero Flag
- D7: C - Carry Flag

Register Pointer
RP (FFDH - Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

- D3–D0: Working Register Page Address
- D7–D4: Working Register Group Address

Stack Pointer High Byte
SPH (FFEH - Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

- D7–D0: Stack Pointer [15:8]

Stack Pointer Low Byte
SPL (FFFH - Read/Write)

D7	D6	D5	D4	D3	D2	D1	D0

- D7–D0: Stack Pointer [7:0]

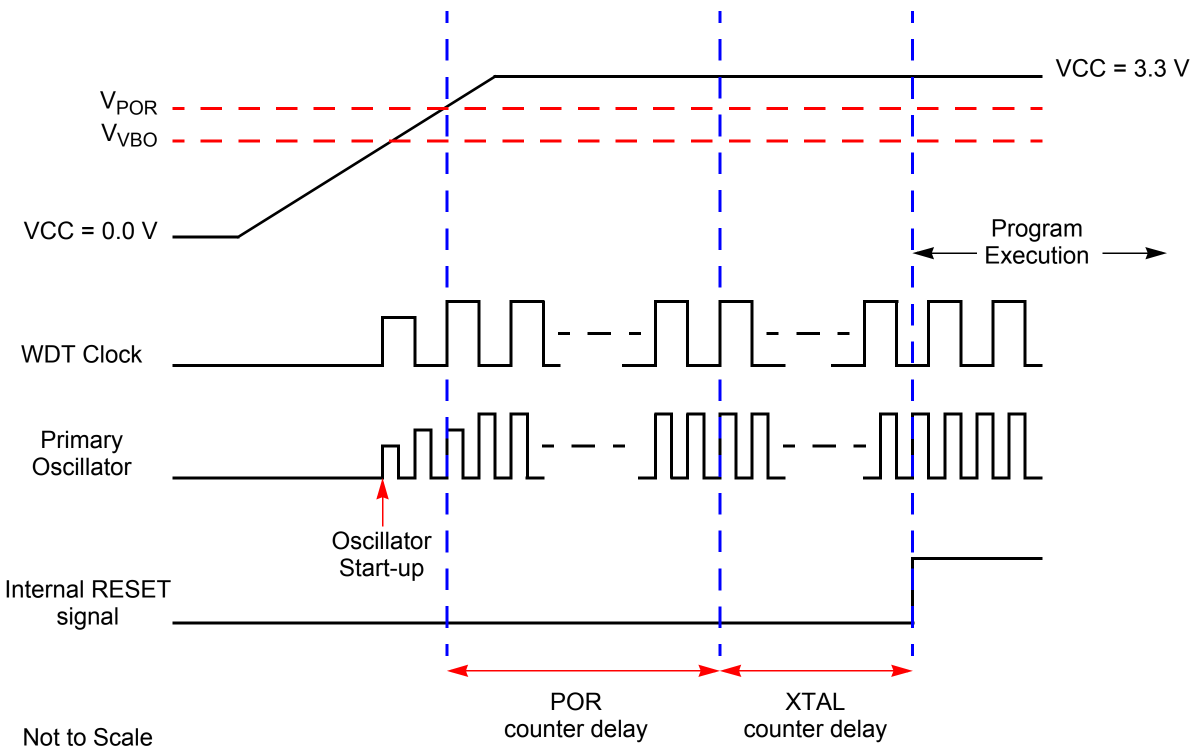

Figure 8. Power-On Reset Operation

Voltage Brownout Reset

The devices in the 64K Series provide low Voltage Brownout protection. The VBO circuit senses when the supply voltage drops to an unsafe level (below the VBO threshold voltage) and forces the device into the Reset state. While the supply voltage remains below the Power-On Reset voltage threshold (V_{POR}), the VBO block holds the device in the Reset state.

After the supply voltage again exceeds the Power-On Reset voltage threshold, the devices progress through a full system reset sequence, as described in the Power-On Reset section. Following Power-On Reset, the POR status bit in the Watchdog Timer Control (WDTCTL) register is set to 1. Figure 9 displays Voltage Brownout operation. For the VBO and POR threshold voltages (V_{VBO} and V_{POR}), see Electrical Characteristics on page 215.

The Voltage Brownout circuit can be either enabled or disabled during STOP mode. Operation during STOP mode is set by the VBO_AO Option Bit. For information on configuring VBO_AO, see Option Bits page 195.

Port A–H Data Direction Sub-Registers

The Port A–H Data Direction sub-register is accessed through the Port A–H Control register by writing 01H to the Port A–H Address register (Table 16).

Table 16. Port A–H Data Direction Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	DD7	DD6	DD5	DD4	DD3	DD2	DD1	DD0
RESET	1							
R/W	R/W							
ADDR	If 01H in Port A–H Address Register, accessible through Port A–H Control Register							

DD[7:0]—Data Direction
These bits control the direction of the associated port pin. Port Alternate Function operation overrides the Data Direction register setting.
0 = Output. Data in the Port A–H Output Data register is driven onto the port pin.
1 = Input. The port pin is sampled and the value written into the Port A–H Input Data Register. The output driver is tri-stated.

Port A–H Alternate Function Sub-Registers

The Port A–H Alternate Function sub-register (Table 17) is accessed through the Port A–H Control register by writing `02H` to the Port A–H Address register. The Port A–H Alternate Function sub-registers select the alternate functions for the selected pins. To determine the alternate function associated with each port pin, see GPIO Alternate Functions on page 59.

Caution: *Do not enable alternate function for GPIO port pins which do not have an associated alternate function. Failure to follow this guideline may result in unpredictable operation.*

Table 17. Port A–H Alternate Function Sub-Registers

BITS	7	6	5	4	3	2	1	0
FIELD	AF7	AF6	AF5	AF4	AF3	AF2	AF1	AF0
RESET	0							
R/W	R/W							
ADDR	If 02H in Port A–H Address Register, accessible through Port A–H Control Register							

Interrupt Controller

Overview

The interrupt controller on the 64K Series products prioritizes the interrupt requests from the on-chip peripherals and the GPIO port pins. The features of the interrupt controller include the following:

- 24 unique interrupt vectors:
  - 12 GPIO port pin interrupt sources
  - 12 on-chip peripheral interrupt sources
- Flexible GPIO interrupts
  - Eight selectable rising and falling edge GPIO interrupts
  - Four dual-edge interrupts
- Three levels of individually programmable interrupt priority
- Watchdog Timer 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 control has no effect on operation. For more information on interrupt servicing by the eZ8 CPU, refer to *eZ8™ CPU Core User Manual (UM0128)* available for download at www.zilog.com.

Interrupt Vector Listing

Table 23 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.

Table 31. IRQ1 Enable High Bit Register (IRQ1ENH)

BITS	7	6	5	4	3	2	1	0
FIELD	PAD7ENH	PAD6ENH	PAD5ENH	PAD4ENH	PAD3ENH	PAD2ENH	PAD1ENH	PAD0ENH
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
ADDR	FC4H							

PAD*x*ENH—Port A or Port D Bit[*x*] Interrupt Request Enable High Bit.
For selection of either Port A or Port D as the interrupt source, see Interrupt Port Select Register on page 78.

Table 32. IRQ1 Enable Low Bit Register (IRQ1ENL)

BITS	7	6	5	4	3	2	1	0
FIELD	PAD7ENL	PAD6ENL	PAD5ENL	PAD4ENL	PAD3ENL	PAD2ENL	PAD1ENL	PAD0ENL
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
ADDR	FC5H							

PAD*x*ENL—Port A or Port D Bit[*x*] Interrupt Request Enable Low Bit
For selection of either Port A or Port D as the interrupt source, see Interrupt Port Select Register on page 78.

IRQ2 Enable High and Low Bit Registers

The IRQ2 Enable High and Low Bit registers (see Table 34 and Table 35 on page 77) form a priority encoded enabling for interrupts in the Interrupt Request 2 register. Priority is generated by setting bits in each register. Table 33 describes the priority control for IRQ2.

Table 33. IRQ2 Enable and Priority Encoding

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

Timers

Overview

The 64K Series products contain up to four 16-bit reloadable timers that can be used for timing, event counting, or generation of pulse width modulated signals. The timers' features include:

- 16-bit reload counter
- Programmable prescaler with prescale values from 1 to 128
- PWM output generation
- Capture and compare capability
- External input pin for timer input, clock gating, or capture signal. External input pin signal frequency is limited to a maximum of one-fourth the system clock frequency.
- Timer output pin
- Timer interrupt

In addition to the timers described in this chapter, the Baud Rate Generators for any unused UART, SPI, or I^2C peripherals may also be used to provide basic timing functionality. For information on using the Baud Rate Generators as timers, see the respective serial communication peripheral. Timer 3 is unavailable in the 44-pin package devices.

Architecture

Figure 12 displays the architecture of the timers.

Timer 0-3 Control 1 Registers

The Timer 0-3 Control 1 (TxCTL1) registers enable/disable the timers, set the prescaler value, and determine the timer operating mode.

Table 46. Timer 0-3 Control 1 Register (TxCTL1)

BITS	7	6	5	4	3	2	1	0
FIELD	TEN	TPOL	PRES			TMODE		
RESET	0							
R/W	R/W							
ADDR	F07H, F0FH, F17H, F1FH							

TEN—Timer Enable
0 = Timer is disabled.
1 = Timer enabled to count.

TPOL—Timer Input/Output Polarity
Operation of this bit is a function of the current operating mode of the timer.

ONE-SHOT 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.

CONTINUOUS 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.

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.

0 = Count occurs on the rising edge of the Timer Input signal.
1 = Count occurs on the falling edge of the Timer Input signal.

PWM mode

0 = Timer Output is forced Low (0) when the timer is disabled. When enabled, the Timer Output is forced High (1) upon PWM count match and forced Low (0) upon Reload.

If interrupts are enabled, following completion of the Stop Mode Recovery the eZ8 CPU responds to the interrupt request by fetching the Watchdog Timer interrupt vector and executing code from the vector address.

WDT Reset in Normal Operation

If configured to generate a Reset when a time-out occurs, the Watchdog Timer forces the device into the Reset state. The WDT status bit in the Watchdog Timer Control register is set to 1. For more information on Reset, see Reset and Stop Mode Recovery on page 47.

WDT Reset in STOP Mode

If enabled in STOP mode and configured to generate a Reset when a time-out occurs and the device is in STOP mode, the Watchdog Timer initiates a Stop Mode Recovery. Both the WDT status bit and the STOP bit in the Watchdog Timer Control register are set to 1 following WDT time-out in STOP mode. Default operation is for the WDT and its RC oscillator to be enabled during STOP mode.

WDT RC Disable in STOP Mode

To minimize power consumption in STOP Mode, the WDT and its RC oscillator can be disabled in STOP mode. The following sequence configures the WDT to be disabled when the 64K Series devices enter STOP Mode following execution of a STOP instruction:

1. Write 55H to the Watchdog Timer Control register (WDTCTL).
2. Write AAH to the Watchdog Timer Control register (WDTCTL).
3. Write 81H to the Watchdog Timer Control register (WDTCTL) to configure the WDT and its oscillator to be disabled during STOP Mode. Alternatively, write 00H to the Watchdog Timer Control register (WDTCTL) as the third step in this sequence to reconfigure the WDT and its oscillator to be enabled during STOP mode.

This sequence only affects WDT operation in STOP mode.

Watchdog Timer Reload Unlock Sequence

Writing the unlock sequence to the Watchdog Timer (WDTCTL) Control register address unlocks the three Watchdog Timer Reload Byte registers (WDTU, WDTH, and WDTL) to allow changes to the time-out period. These write operations to the WDTCTL register address produce no effect on the bits in the WDTCTL register. The locking mechanism prevents spurious writes to the Reload registers. Follow the steps below to unlock the Watchdog Timer Reload Byte registers (WDTU, WDTH, and WDTL) for write access.

1. Write 55H to the Watchdog Timer Control register (WDTCTL).
2. Write AAH to the Watchdog Timer Control register (WDTCTL).
3. Write the Watchdog Timer Reload Upper Byte register (WDTU).
4. Write the Watchdog Timer Reload High Byte register (WDTH).

when a byte is written to the UART Transmit Data register. The Driver Enable signal asserts at least one UART bit period and no greater than two UART bit periods before the Start bit is transmitted. This timing allows a setup time to enable the transceiver. The Driver Enable signal deasserts one system clock period after the last Stop bit is transmitted. This one system clock delay allows both time for data to clear the transceiver before disabling it, as well as the ability to determine if another character follows the current character. In the event of back to back characters (new data must be written to the Transmit Data Register before the previous character is completely transmitted) the `DE` signal is not deasserted between characters. The `DEPOL` bit in the UART Control Register 1 sets the polarity of the Driver Enable signal.

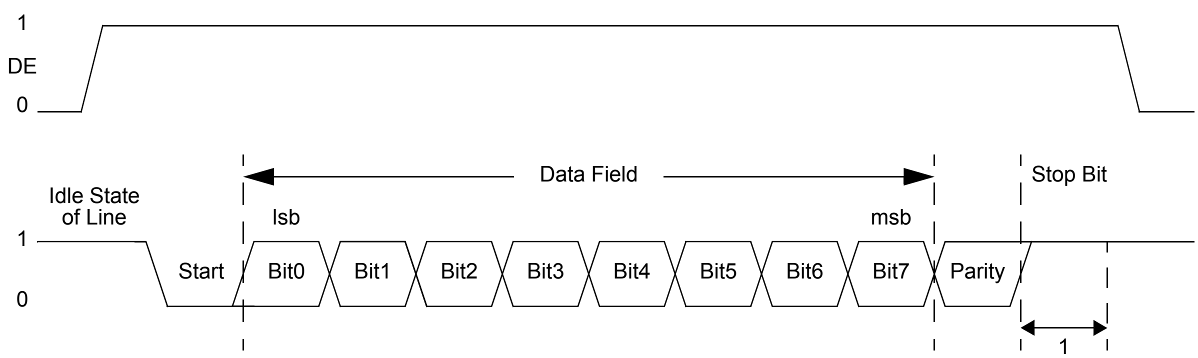

Figure 17. UART Driver Enable Signal Timing (shown with 1 Stop Bit and Parity)

The Driver Enable to Start bit setup time is calculated as follows:

$$\left(\frac{1}{\text{Baud Rate (Hz)}}\right) \leq \text{DE to Start Bit Setup Time (s)} \leq \left(\frac{2}{\text{Baud Rate (Hz)}}\right)$$

UART Interrupts

The UART features separate interrupts for the transmitter and the receiver. In addition, when the UART primary functionality is disabled, the Baud Rate Generator can also function as a basic timer with interrupt capability.

Transmitter Interrupts

The transmitter generates a single interrupt when the Transmit Data Register Empty bit (`TDRE`) is set to 1. This indicates that the transmitter is ready to accept new data for transmission. The `TDRE` interrupt occurs after the Transmit shift register has shifted the first bit of data out. At this point, the Transmit Data register may be written with the next character to send. This provides 7 bit periods of latency to load the Transmit Data register before the Transmit shift register completes shifting the current character. Writing to the UART Transmit Data register clears the `TDRE` bit to 0.

- Master receives from a 7-bit slave
- Master receives from a 10-bit slave

SDA and SCL Signals

I^2C sends all addresses, data and acknowledge signals over the SDA line, most-significant bit first. SCL is the common clock for the I^2C Controller. When the SDA and SCL pin alternate functions are selected for their respective GPIO ports, the pins are automatically configured for open-drain operation.

The master (I^2C) is responsible for driving the SCL clock signal, although the clock signal can become skewed by a slow slave device. During the low period of the clock, the slave pulls the SCL signal Low to suspend the transaction. The master releases the clock at the end of the low period and notices that the clock remains low instead of returning to a high level. When the slave releases the clock, the I^2C Controller continues the transaction. All data is transferred in bytes and there is no limit to the amount of data transferred in one operation. When transmitting data or acknowledging read data from the slave, the SDA signal changes in the middle of the low period of SCL and is sampled in the middle of the high period of SCL.

I^2C Interrupts

The I^2C Controller contains four sources of interrupts—Transmit, Receive, Not Acknowledge and baud rate generator. These four interrupt sources are combined into a single interrupt request signal to the Interrupt Controller. The Transmit interrupt is enabled by the IEN and TXI bits of the Control register. The Receive and Not Acknowledge interrupts are enabled by the IEN bit of the Control register. The baud rate generator interrupt is enabled by the BIRQ and IEN bits of the Control register.

Not Acknowledge interrupts occur when a Not Acknowledge condition is received from the slave or sent by the I^2C Controller and neither the `START` or `STOP` bit is set. The Not Acknowledge event sets the NCKI bit of the I^2C Status register and can only be cleared by setting the `START` or `STOP` bit in the I^2C Control register. When this interrupt occurs, the I^2C Controller waits until either the STOP or START bit is set before performing any action. In an interrupt service routine, the NCKI bit should always be checked prior to servicing transmit or receive interrupt conditions because it indicates the transaction is being terminated.

Receive interrupts occur when a byte of data has been received by the I^2C Controller (master reading data from slave). This procedure sets the RDRF bit of the I^2C Status register. The RDRF bit is cleared by reading the I^2C Data register. The RDRF bit is set during the acknowledge phase. The I^2C Controller pauses after the acknowledge phase until the receive interrupt is cleared before performing any other action.

0 = DMA0 is not the source of the interrupt from the DMA Controller.
1 = DMA0 completed transfer of data to/from the End Address and generated an interrupt.

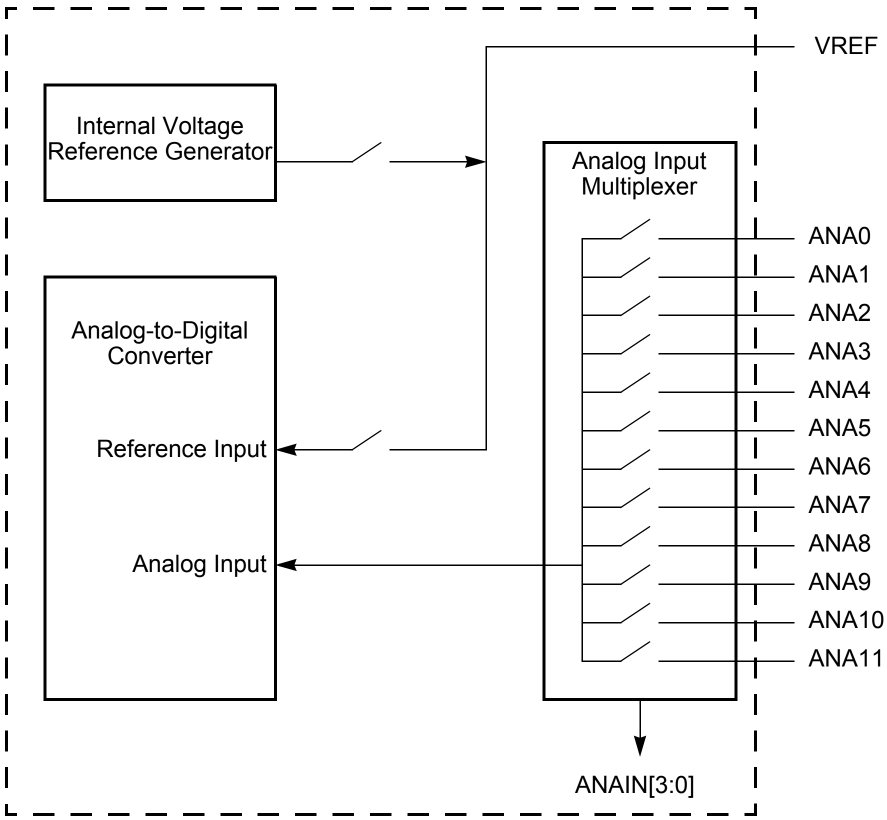

Figure 34. Analog-to-Digital Converter Block Diagram

The sigma-delta ADC architecture provides alias and image attenuation below the amplitude resolution of the ADC in the frequency range of DC to one-half the ADC clock rate (one-fourth the system clock rate). The ADC provides alias free conversion for frequencies up to one-half the ADC clock rate. Thus the sigma-delta ADC exhibits high noise immunity making it ideal for embedded applications. In addition, monotonicity (no missing codes) is guaranteed by design.

Operation

Automatic Power-Down

If the ADC is idle (no conversions in progress) for 160 consecutive system clock cycles, portions of the ADC are automatically powered-down. From this power-down state, the ADC requires 40 system clock cycles to power-up. The ADC powers up when a conversion is requested using the ADC Control register.

Follow the steps below for setting up the ADC and initiating continuous conversion:

1. Enable the desired analog input by configuring the general-purpose I/O pins for alternate function. This disables the digital input and output driver.
2. Write to the ADC Control register to configure the ADC for continuous conversion. The bit fields in the ADC Control register may be written simultaneously:
   – Write to the `ANAIN[3:0]` field to select one of the 12 analog input sources.
   – Set `CONT` to 1 to select continuous conversion.
   – Write to the ~~`VREF`~~ bit to enable or disable the internal voltage reference generator.
   – Set `CEN` to 1 to start the conversions.
3. When the first conversion in continuous operation is complete (after 5129 system clock cycles, plus the 40 cycles for power-up, if necessary), the ADC control logic performs the following operations:
   – `CEN` resets to 0 to indicate the first conversion is complete. `CEN` remains 0 for all subsequent conversions in continuous operation.
   – An interrupt request is sent to the Interrupt Controller to indicate the conversion is complete.
4. Thereafter, the ADC writes a new 10-bit data result to {ADCD_H[7:0], ADCD_L[7:6]} every 256 system clock cycles. An interrupt request is sent to the Interrupt Controller when each conversion is complete.
5. To disable continuous conversion, clear the `CONT` bit in the ADC Control register to 0.

DMA Control of the ADC

The Direct Memory Access (DMA) Controller can control operation of the ADC including analog input selection and conversion enable. For more information on the DMA and configuring for ADC operations, see Direct Memory Access Controller on page 165.

Reserved
These bits are reserved and must be 0.

FSTAT—Flash Controller Status
00_0000 = Flash Controller locked
00_0001 = First unlock command received
00_0010 = Second unlock command received
00_0011 = Flash Controller unlocked
00_0100 = Flash Sector Protect register selected
00_1xxx = Program operation in progress
01_0xxx = Page erase operation in progress
10_0xxx = Mass erase operation in progress

Page Select Register

The Page Select (FPS) register (Table 94) selects one of the 128 available Flash memory pages to be erased or programmed. Each Flash Page contains 512 bytes of Flash memory. During a Page Erase operation, all Flash memory locations with the 7 most significant bits of the address given by the PAGE field are erased to FFH.

The Page Select register shares its Register File address with the Flash Sector Protect Register. The Page Select register cannot be accessed when the Flash Sector Protect register is enabled.

Table 94. Page Select Register (FPS)

BITS	7	6	5	4	3	2	1	0
FIELD	INFO_EN	PAGE						
RESET	0							
R/W	R/W							
ADDR	FF9H							

INFO_EN—Information Area Enable
0 = Information Area is not selected.
1 = Information Area is selected. The Information area is mapped into the Flash Memory address space at addresses FE00H through FFFFH.

PAGE—Page Select
This 7-bit field selects the Flash memory page for Programming and Page Erase operations. Flash Memory Address[15:9] = PAGE[6:0].

On-Chip Debugger

Overview

The 64K Series products contain an integrated On-Chip Debugger (OCD) that provides advanced debugging features including:

- Reading and writing of the Register File
- Reading and writing of Program and Data Memory
- Setting of Breakpoints
- Execution of eZ8 CPU instructions

Architecture

The On-Chip Debugger consists of four primary functional blocks: transmitter, receiver, auto-baud generator, and debug controller. Figure 36 displays the architecture of the On-Chip Debugger.

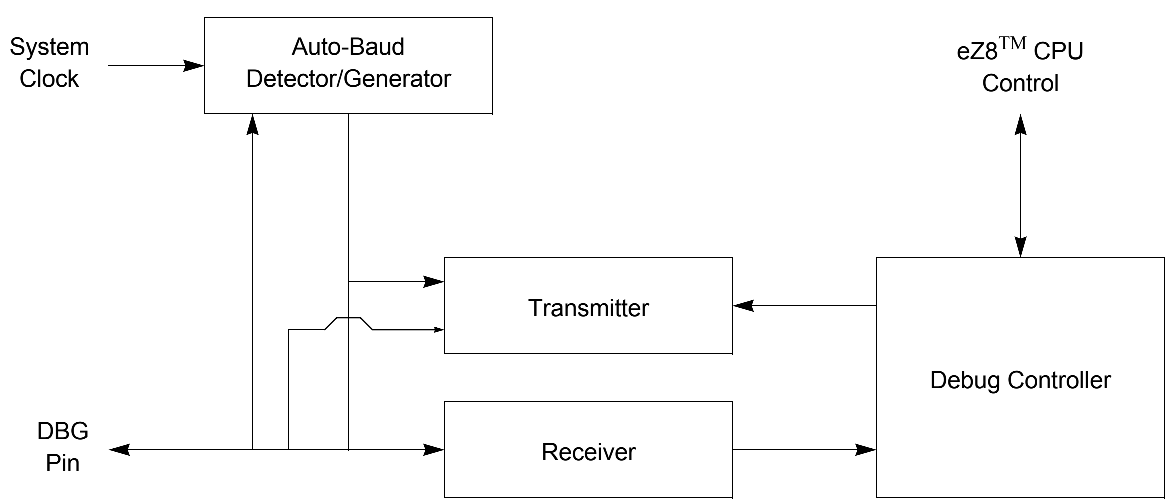

Figure 36. On-Chip Debugger Block Diagram

SPI Master Mode Timing

Figure 53 and Table 117 provide timing information for SPI Master mode pins. Timing is shown with SCK rising edge used to source MOSI output data, SCK falling edge used to sample MISO input data. Timing on the SS output pin(s) is controlled by software.

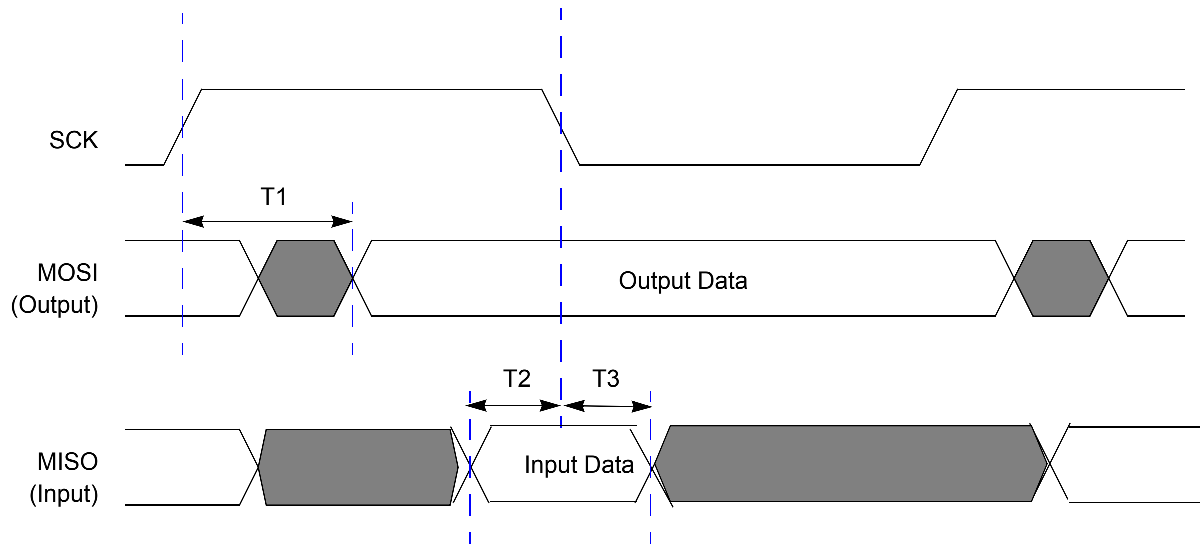

Figure 53. SPI Master Mode Timing

Table 117. SPI Master Mode Timing

		Delay (ns)	
Parameter	**Abbreviation**	**Min**	**Max**
SPI Master			
T_1	SCK Rise to MOSI output Valid Delay	-5	+5
T_2	MISO input to SCK (receive edge) Setup Time	20	
T_3	MISO input to SCK (receive edge) Hold Time	0	

Table 133. eZ8 CPU Instruction Summary (Continued)

Assembly Mnemonic	Symbolic Operation	Address Mode dst	Address Mode src	Opcode(s) (Hex)	Flags C	Z	S	V	D	H	Fetch Cycles	Instr. Cycles
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
LD dst, rc	dst ← src	r	IM	0C-FC	-	-	-	-	-	-	2	2
		r	X(r)	C7							3	3
		X(r)	r	D7							3	4
		r	Ir	E3							2	3
		R	R	E4							3	2
		R	IR	E5							3	4
		R	IM	E6							3	2
		IR	IM	E7							3	3
		Ir	r	F3							2	3
		IR	R	F5							3	3

T

U

V

W

X

Z