Zilog - EZ80F93AZ020EC00TR Datasheet

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	eZ80
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, IrDA, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, WDT
Number of I/O	24
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	3V ~ 3.6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	100-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/zilog/ez80f93az020ec00tr

Table 1. 100-Pin LQFP Pin Identification of the eZ80F92 Device (Continued)

Pin No	Symbol	Function	Signal Direction	Description
25	ADDR20	Address Bus	Bidirectional	Configured as an output in normal operation. The address bus selects a location in memory or I/O space to be read or written. Configured as an input during bus acknowledge cycles. Drives the Chip Select/Wait State Generator block to generate Chip Selects.
26	ADDR21	Address Bus	Bidirectional	Configured as an output in normal operation. The address bus selects a location in memory or I/O space to be read or written. Configured as an input during bus acknowledge cycles. Drives the Chip Select/Wait State Generator block to generate Chip Selects.
27	ADDR22	Address Bus	Bidirectional	Configured as an output in normal operation. The address bus selects a location in memory or I/O space to be read or written. Configured as an input during bus acknowledge cycles. Drives the Chip Select/Wait State Generator block to generate Chip Selects.
28	ADDR23	Address Bus	Bidirectional	Configured as an output in normal operation. The address bus selects a location in memory or I/O space to be read or written. Configured as an input during bus acknowledge cycles. Drives the Chip Select/Wait State Generator block to generate Chip Selects.
29	$\overline{CS0}$	Chip Select 0	Output, Active Low	$\overline{CS0}$ Low indicates that an access is occurring in the defined $\overline{CS0}$ memory or I/O address space.
30	$\overline{CS1}$	Chip Select 1	Output, Active Low	$\overline{CS1}$ Low indicates that an access is occurring in the defined $\overline{CS1}$ memory or I/O address space.
31	$\overline{CS2}$	Chip Select 2	Output, Active Low	$\overline{CS2}$ Low indicates that an access is occurring in the defined $\overline{CS2}$ memory or I/O address space.
32	$\overline{CS3}$	Chip Select 3	Output, Active Low	$\overline{CS3}$ Low indicates that an access is occurring in the defined $\overline{CS3}$ memory or I/O address space.

Table 2. Pin Characteristics of the eZ80F92 Device (Continued)

Pin No	Symbol	Direction	Reset Direction	Active Low/High	Tristate Output	Pull Up/Down	Schmitt Trigger Input	Open Drain/ Source
16	ADDR13	I/O	O	N/A	Yes	No	No	No
17	ADDR14	I/O	O	N/A	Yes	No	No	No
18	V_{DD}							
19	V_{SS}							
20	ADDR15	I/O	O	N/A	Yes	No	No	No
21	ADDR16	I/O	O	N/A	Yes	No	No	No
22	ADDR17	I/O	O	N/A	Yes	No	No	No
23	ADDR18	I/O	O	N/A	Yes	No	No	No
24	ADDR19	I/O	O	N/A	Yes	No	No	No
25	ADDR20	I/O	O	N/A	Yes	No	No	No
26	ADDR21	I/O	O	N/A	Yes	No	No	No
27	ADDR22	I/O	O	N/A	Yes	No	No	No
28	ADDR23	I/O	O	N/A	Yes	No	No	No
29	$\overline{CS0}$	O	O	Low	No	No	No	No
30	$\overline{CS1}$	O	O	Low	No	No	No	No
31	$\overline{CS2}$	O	O	Low	No	No	No	No
32	$\overline{CS3}$	O	O	Low	No	No	No	No
33	V_{DD}							
34	V_{SS}							
35	DATA0	I/O	I	N/A	Yes	No	No	No
36	DATA1	I/O	I	N/A	Yes	No	No	No
37	DATA2	I/O	I	N/A	Yes	No	No	No
38	DATA3	I/O	I	N/A	Yes	No	No	No
39	DATA4	I/O	I	N/A	Yes	No	No	No
40	DATA5	I/O	I	N/A	Yes	No	No	No
41	DATA6	I/O	I	N/A	Yes	No	No	No
42	DATA7	I/O	I	N/A	Yes	No	No	No
43	V_{DD}							

Table 2. Pin Characteristics of the eZ80F92 Device (Continued)

Pin No	Symbol	Direction	Reset Direction	Active Low/High	Tristate Output	Pull Up/Down	Schmitt Trigger Input	Open Drain/ Source
44	V_{SS}							
45	$\overline{IORQ}$	I/O	O	Low	Yes	No	No	No
46	$\overline{MREQ}$	I/O	O	Low	Yes	No	No	No
47	$\overline{RD}$	O	O	Low	Yes	No	No	No
48	$\overline{WR}$	O	O	Low	Yes	No	No	No
49	$\overline{INSTRD}$	O	O	Low	No	No	No	No
50	$\overline{WAIT}$	I	I	Low	N/A	No	No	N/A
51	$\overline{RESET}$	I	I	Low	N/A	No	Yes	N/A
52	$\overline{NMI}$	I	I	Low	N/A	No	Yes	N/A
53	$\overline{BUSREQ}$	I	I	Low	N/A	No	No	N/A
54	$\overline{BUSACK}$	O	O	Low	No	No	No	No
55	$\overline{HALT\_SLP}$	O	O	Low	No	No	No	No
56	V_{DD}							
57	V_{SS}							
58	RTC_X_{IN}	I	I	N/A	N/A	No	No	N/A
59	RTC_X_{OUT}	I/O	U	N/A	N/A	No	No	No
60	RTC_V_{DD}							
61	V_{SS}							
62	TMS	I	I	N/A	N/A	Up	No	N/A
63	TCK	I	I	Rising (In) Falling (Out)	N/A	Up	No	N/A
64	TRIGOUT	O	O	High	No	No	No	No
65	TDI	I/O	I	N/A	Yes	No	No	No
66	TDO	O	U	N/A	Yes	No	No	No
67	V_{DD}							
68	PD0	I/O	I	N/A	Yes	No	No	OD & OS
69	PD1	I/O	I	N/A	Yes	No	No	OD & OS
70	PD2	I/O	I	N/A	Yes	No	No	OD & OS

Table 2. Pin Characteristics of the eZ80F92 Device (Continued)

Pin No	Symbol	Direction	Reset Direction	Active Low/High	Tristate Output	Pull Up/Down	Schmitt Trigger Input	Open Drain/ Source
71	PD3	I/O	I	N/A	Yes	No	No	OD & OS
72	PD4	I/O	I	N/A	Yes	No	No	OD & OS
73	PD5	I/O	I	N/A	Yes	No	No	OD & OS
74	PD6	I/O	I	N/A	Yes	No	No	OD & OS
75	PD7	I/O	I	N/A	Yes	No	No	OD & OS
76	PC0	I/O	I	N/A	Yes	No	No	OD & OS
77	PC1	I/O	I	N/A	Yes	No	No	OD & OS
78	PC2	I/O	I	N/A	Yes	No	No	OD & OS
79	PC3	I/O	I	N/A	Yes	No	No	OD & OS
80	PC4	I/O	I	N/A	Yes	No	No	OD & OS
81	PC5	I/O	I	N/A	Yes	No	No	OD & OS
82	PC6	I/O	I	N/A	Yes	No	No	OD & OS
83	PC7	I/O	I	N/A	Yes	No	No	OD & OS
84	V_{SS}							
85	X_{IN}	I	I	N/A	N/A	No	No	N/A
86	X_{OUT}	O	O	N/A	No	No	No	No
87	V_{DD}							
88	PB0	I/O	I	N/A	Yes	No	No	OD & OS
89	PB1	I/O	I	N/A	Yes	No	No	OD & OS
90	PB2	I/O	I	N/A	Yes	No	No	OD & OS
91	PB3	I/O	I	N/A	Yes	No	No	OD & OS
92	PB4	I/O	I	N/A	Yes	No	No	OD & OS
93	PB5	I/O	I	N/A	Yes	No	No	OD & OS
94	PB6	I/O	I	N/A	Yes	No	No	OD & OS
95	PB7	I/O	I	N/A	Yes	No	No	OD & OS
96	V_{DD}							
97	V_{SS}							
98	SDA	I/O	I	N/A	Yes	Up	No	OD

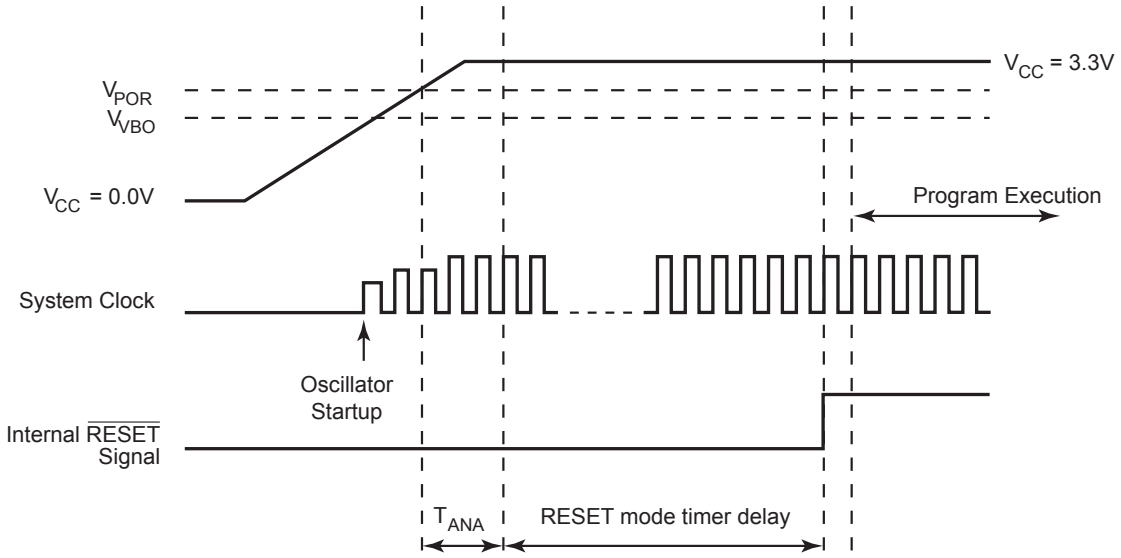

Figure 3. Power-On Reset Operation

Voltage Brownout Reset

If, after program execution begins, the supply voltage (V_{CC}) drops below the Voltage Brownout threshold (V_{VBO}), the eZ80F92 device resets. The VBO protection circuitry detects the low supply voltage and initiates the RESET via the Reset controller.
The eZ80F92 device remains in RESET mode until the supply voltage again returns above the POR voltage threshold (V_{POR}) and the Reset controller releases the internal RESET signal. The VBO circuitry rejects very short negative brown-out pulses to prevent spurious RESET events. VBO operation is displayed in Figure 4 on page 34. The signals in this figure are not drawn to scale and are for displaying purposes only.

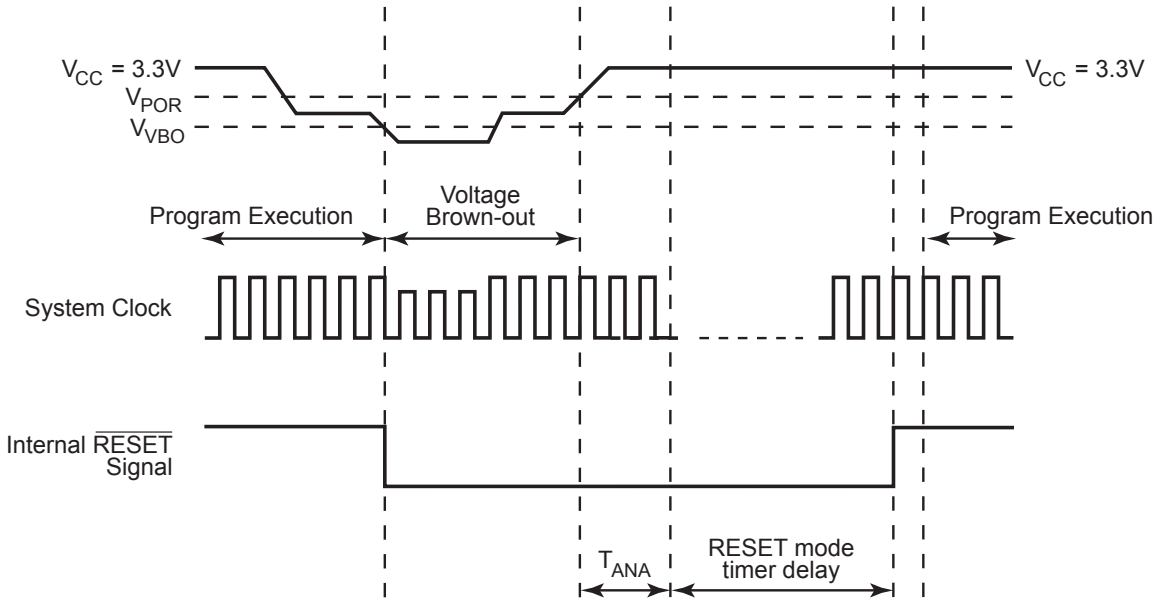

Figure 4. Voltage Brownout Reset Operation

GPIO Mode 5—Reserved. This pin produces high-impedance output.

GPIO Mode 6—This bit enables a dual edge-triggered interrupt mode. Both a rising and a falling edge on the pin cause an interrupt request to be sent to the CPU. Writing a 1 to the Port *x* Data register bit position resets the corresponding interrupt request. Writing a 0 produces no effect. The programmer must set the Port *x* Data register before entering the edge-triggered interrupt mode.

GPIO Mode 7—For Ports B, C, and D, the port pin is configured to pass control over to the alternate (secondary) functions assigned to the pin. For example, the alternate mode function for PC7 is $\overline{RI1}$ and the alternate mode function for PB4 is the Timer 4 Out. When GPIO Mode 7 is enabled, the pin output data and pin tristated control come from the alternate function's data output and tristate control, respectively. The value in the Port *x* Data register produces no effect on operation.

Note: *Input signals are sampled by the system clock before being passed to the alternate function input.*

GPIO Mode 8—The port pin is configured for level-sensitive interrupt modes. An interrupt request is generated when the level at the pin is the same as the level stored in the Port *x* Data register. The port pin value is sampled by the system clock. The input pin must be held at the selected interrupt level for a minimum of 2 clock periods to initiate an interrupt. The interrupt request remains active as long as this condition is maintained at the external source.

GPIO Mode 9—The port pin is configured for single edge-triggered interrupt mode. The value in the Port *x* Data register determines if a positive or negative edge causes an interrupt request. A 0 in the Port *x* Data register bit sets the selected pin to generate an interrupt request for falling edges. A 1 in the Port *x* Data register bit sets the selected pin to generate an interrupt request for rising edges. The interrupt request remains active until a 1 is written to the corresponding interrupt request of the Port *x* Data register bit. Writing a 0 produces no effect on operation. The programmer must set the Port *x* Data register before entering the edge-triggered interrupt mode.

A simplified block diagram of a GPIO port pin is displayed in Figure 5 on page 42.

Port x Data Direction Registers

In conjunction with the other GPIO Control Registers, the Port *x* Data Direction registers, listed in Table 8, control the operating modes of the GPIO port pins. See Table 6 on page 39 for more information.

Table 8. Port *x* Data Direction Registers; (PB_DDR = 009Bh, PC_DDR = 009Fh, PD_DDR = 00A3h)

Bit	7	6	5	4	3	2	1	0
Reset	1	1	1	1	1	1	1	1
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write.								

Port x Alternate Register 1

In conjunction with the other GPIO Control Registers, the Port *x* Alternate Register 1, listed in Table 9, control the operating modes of the GPIO port pins. See Table 6 on page 39 for more information.

Table 9. Port *x* Alternate Registers 1; (PB_ALT1 = 009Ch, PC_ALT1 = 00A0h, PD_ALT1 = 00A4h)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write.								

Port x Alternate Register 2

In conjunction with the other GPIO Control Registers, the Port *x* Alternate Register 2, listed in Table 10, control the operating modes of the GPIO port pins. See Table 6 on page 39 for more information.

Table 10. Port x Alternate Registers 2; (PB_ALT2 = 009Dh, PC_ALT2 = 00A1h, PD_ALT2 = 00A5h)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write.								

Chip Selects and Wait States

The eZ80F92 device generates four Chip Selects for external devices. Each Chip Select may be programmed to access either memory space or I/O space. The Memory Chip Selects can be individually programmed on a 64 KB boundary. The I/O Chip Selects can each choose a 256-byte section of I/O space. In addition, each Chip Select may be programmed for up to 7 wait states.

Memory and I/O Chip Selects

Each of the Chip Selects can be enabled for either the memory address space or the I/O address space, but not both. To select the memory address space for a particular Chip Select, CSX_IO (CS*x*_CTL[4]) must be reset to 0. To select the I/O address space for a particular Chip Select, CSX_IO must be set to 1. After RESET, the default is for all Chip Selects to be configured for the memory address space. For either the memory address space or the I/O address space, the individual Chip Selects must be enabled by setting CSx_EN (CS*x*_CTL[3]) to 1.

Memory Chip Select Operation

Operation of each of the Memory Chip Selects is controlled by three control registers. To enable a particular Memory Chip Select, the following conditions must be met:

- The Chip Select is enabled by setting CSx_EN to 1
- The Chip Select is configured for Memory by clearing CSX_IO to 0
- The address is in the associated Chip Select range:

  CS*x*_LBR[7:0] ≤ ADDR[23:16] ≤ CS*x*_UBR[7:0]
- No higher priority (lower number) Chip Select meets the above conditions
- A memory access instruction must be executing

If all of the foregoing conditions are met to generate a Memory Chip Select, then the following actions occur:

- The appropriate Chip Select—$\overline{CS0}$, $\overline{CS1}$, $\overline{CS2}$, or $\overline{CS3}$—is asserted (driven Low)
- $\overline{MREQ}$ is asserted (driven Low)
- Depending upon the instruction, either $\overline{RD}$ or $\overline{WR}$ is asserted (driven Low)

If the upper and lower bounds are set to the same value (CS*x*_UBR = CS*x*_LBR), then a particular Chip Select is valid for a single 64 KB page.

Table 15. Z80 Bus Mode Write States

STATE T1	The Write cycle begins in State T1. The CPU drives the address onto the address bus, the associated Chip Select signal is asserted.
STATE T2	During State T2, the $\overline{WR}$ signal is asserted. Depending upon the instruction, either the $\overline{MREQ}$ or $\overline{IORQ}$ signal is asserted. If the external $\overline{WAIT}$ pin is driven Low at least one CPU system clock cycle prior to the end of State T2, additional WAIT states (T_{WAIT}) are asserted until the $\overline{WAIT}$ pin is driven High.
STATE T3	During State T3, no bus signals are altered.

Z80® bus mode Read and Write timing is displayed in Figure 9 and Figure 10 on page 55. The Z80 bus mode states can be configured for 1 to 15 CPU system clock cycles. In the figures, each Z80 bus mode state is two CPU system clock cycles in duration. Figure 9 and Figure 10 on page 55 also display the assertion of 1 wait state (T_{WAIT}) by the external peripheral during each Z80 bus mode cycle.

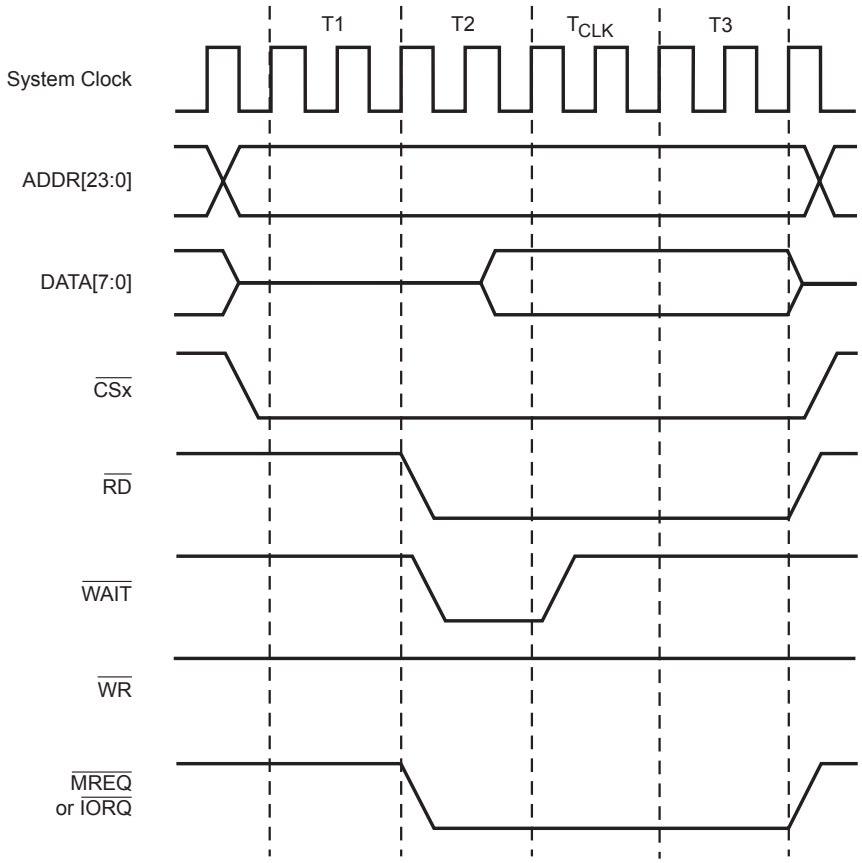

Figure 9.Example: Z80 Bus Mode Read Timing

Watchdog Timer

Watchdog Timer Overview

The Watchdog Timer (WDT) helps protect against corrupt or unreliable software, power faults, and other system-level problems which may place the CPU into unsuitable operating states.

The eZ80F92 WDT features:

- Four programmable time-out periods: 2^{18}, 2^{22}, 2^{25}, and 2^{27} clock cycles
- Two selectable WDT clock sources: the system clock or the Real-Time Clock source (on-chip 32 kHz crystal oscillator or 50/60 Hz signal)
- A selectable time-out response: a time-out can be configured to generate either a RESET or a nonmaskable interrupt (NMI)
- A WDT time-out RESET indicator flag

Figure 19 displays the block diagram for the Watchdog Timer.

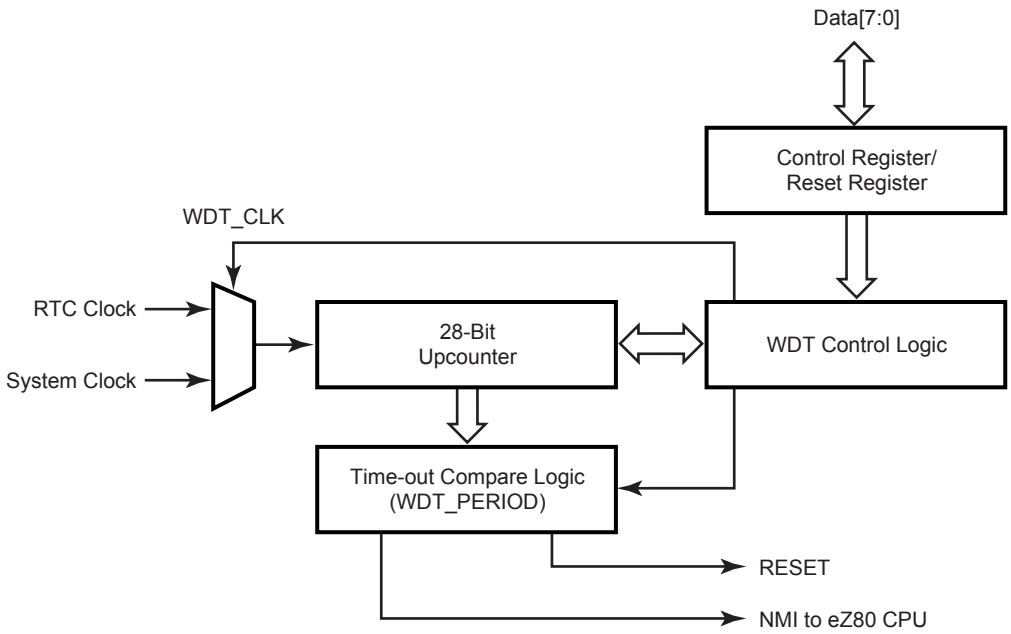

Figure 19. Watchdog Timer Block Diagram

Watchdog Timer Operation

Enabling and Disabling the WDT

The Watchdog Timer is disabled upon a RESET. To enable the WDT, the application program must set the WDT_EN bit (bit 7) of the WDT_CTL register. When enabled, the WDT cannot be disabled without a RESET.

Time-Out Period Selection

There are four choices of time-out periods for the WDT—2^{18}, 2^{22}, 2^{25}, and 2^{27} system clock cycles. The WDT time-out period is defined by the WDT_PERIOD field of the WDT_CTL register (WDT_CTL[1:0]). The approximate time-out period for two different WDT clock sources is listed in Table 26.

Table 26. Watchdog Timer Approximate Time-Out Delays

Clock Source	Divider Value	Time Out Delay
32.768 kHz Crystal Oscillator	2^{18}	8.00s
32.768 kHz Crystal Oscillator	2^{22}	128s
32.768 kHz Crystal Oscillator	2^{25}	1024s
32.768 kHz Crystal Oscillator	2^{27}	4096 s
20 MHz System Clock	2^{18}	13.1 ms*
20 MHz System Clock	2^{22}	209.7 ms*
20 MHz System Clock	2^{25}	1.68 s
20 MHz System Clock	2^{27}	6.71 s
50 MHz System Clock	2^{18}	5.2 ms
50 MHz System Clock	2^{22}	83.9 ms
50 MHz System Clock	2^{25}	0.67 s
50 MHz System Clock	2^{27}	2.68 s

Note: *WDT time-out values should be sufficiently long to allow Flash operations to complete.

RESET Or NMI Generation

On a WDT time-out, the RST_FLAG bit in the WDT_CTL register is set to 1. In addition, the WDT can cause a RESET or send a nonmaskable interrupt (NMI) signal to the CPU. The default operation is for the WDT to cause a RESET. It asserts/deasserts on the rising edge of the clock. The RST_FLAG bit can be polled by the CPU to determine the source of the RESET event.

Table 61. UART Character Parameter Definition

CHAR[2:0]	Character Length (Tx/Rx Data Bits)	Stop Bits (Tx Stop Bits)
000	5	1
001	6	1
010	7	1
011	8	1
100	5	2
101	6	2
110	7	2
111	8	2

Table 62. Parity Select Definition for Multidrop Communications

MDM UARTx_MGTL[5]	EPS UARTx_LCTL940	Parity Type
0	0	odd
0	1	even
1	0	space
1	1*	mark

Note: *In MULTIDROP mode, EPS resets to 0 after the first character is sent.

Table 71. SPI Clock Phase and Clock Polarity Operation

CPHA	CPOL	SCK Transmit Edge	SCK Receive Edge	SCK Idle State	SS High Between Characters?
0	0	Falling	Rising	Low	Yes
0	1	Rising	Falling	High	Yes
1	0	Rising	Falling	Low	No
1	1	Falling	Rising	High	No

SPI Functional Description

When a master transmits to a slave device via the MOSI signal, the slave device responds by sending data to the master via the master's MISO signal. The resulting implication is a full-duplex transmission, with both data out and data in synchronized with the same clock signal. Thus the byte transmitted is replaced by the byte received and eliminates the requirement for separate transmit-empty and receive-full status bits. A single status bit, SPIF, is used to signify that the I/O operation is completed, see the SPI Status Register (SPI_SR) on page 137.

The SPI is double-buffered on Read, but not on Write. If a Write is performed during data transfer, the transfer occurs uninterrupted, and the Write is unsuccessful. This condition causes the WRITE COLLISION (WCOL) status bit in the SPI_SR register to be set. After a data byte is shifted, the SPIF flag of the SPI_SR register is set.

In SPI MASTER mode, the SCK pin functions as an output. It idles High or Low, depending on the CPOL bit in the SPI_CTL register, until data is written to the shift register. Data transfer is initiated by writing to the transmit shift register, SPI_TSR. Eight clocks are then generated to shift the 8 bits of transmit data out the MOSI pin while shifting in 8 bits of data on the MISO pin. After transfer, the SCK signal idles.

In SPI SLAVE mode, the start logic receives a logic Low from the $\overline{SS}$ pin and a clock input at the SCK pin, and the slave is synchronized to the master. Data from the master is received serially from the slave MOSI signal and loads the 8-bit shift register. After the 8-bit shift register is loaded, its data is parallel transferred to the Read buffer. During a Write cycle data is written into the shift register, then the slave waits for the SPI master to initiate a data transfer, supply a clock signal, and shift the data out on the slave's MISO signal.

If the CPHA bit in the SPI_CTL register is 0, a transfer begins when $\overline{SS}$ pin signal goes Low and the transfer ends when $\overline{SS}$ goes High after eight clock cycles on SCK. When the CPHA bit is set to 1, a transfer begins the first time SCK becomes active while $\overline{SS}$ is Low and the transfer ends when the SPIF flag gets set.

set to 1, the I^2C responds to calls to its slave address and to the general call address if the GCE bit (I2C_SAR[0]) is set to 1.

When the Master Mode Start bit (STA) is set to 1, the I^2C enters MASTER mode and sends a START condition on the bus when the bus is free. If the STA bit is set to 1 when the I^2C module is already in MASTER mode and one or more bytes are transmitted, then a repeated START condition is sent. If the STA bit is set to 1 when the I^2C block is accessed in SLAVE mode, the I^2C completes the data transfer in SLAVE mode and then enters MASTER mode when the bus is released. The STA bit is automatically cleared after a START condition is set. Writing a 0 to this bit produces no effect.

If the Master Mode Stop bit (STP) is set to 1 in MASTER mode, a STOP condition is transmitted on the I^2C bus. If the STP bit is set to 1 in slave move, the I^2C module operates as if a STOP condition is received, but no STOP condition is transmitted. If both STA and STP bits are set, the I^2C block first transmits the STOP condition (if in MASTER mode) and then transmit the START condition. The STP bit is cleared automatically. Writing a 0 to this bit produces no effect.

The I^2C Interrupt Flag (IFLG) is set to 1 automatically when any of 30 of the possible 31 I^2C states is entered. The only state that does not set the IFLG bit is state F8h. If IFLG is set to 1 and the IEN bit is also set, an interrupt is generated. When IFLG is set by the I^2C, the Low period of the I^2C bus clock line is stretched and the data transfer is suspended. When a 0 is written to IFLG, the interrupt is cleared and the I^2C clock line is released.

When the I^2C Acknowledge bit (AAK) is set to 1, an Acknowledge is sent during the acknowledge clock pulse on the I^2C bus if:

- Either the whole of a 7-bit slave address or the first or second byte of a 10-bit slave address is received
- The general call address is received and the General Call Enable bit in I2C_SAR is set to 1
- A data byte is received while in MASTER or SLAVE modes

When AAK is cleared to 0, a NACK is sent when a data byte is received in MASTER or SLAVE mode. If AAK is cleared to 0 in the Slave Transmitter mode, the byte in the I2C_DR register is assumed to be the final byte. After this byte is transmitted, the I^2C block enter states C8h, then returns to the idle state. The I^2C module does not respond to its slave address unless AAK is set. See Table 87 on page 157.

Bit Position	Value	Description
4 ADL	0	The CPU is operating in Z80® MEMORY mode. (ADL bit = 0)
	1	The CPU is operating in ADL MEMORY mode. (ADL bit = 1)
3 MADL	0	The CPU's Mixed-Memory mode (MADL) bit is reset to 0.
	1	The CPU's Mixed-Memory mode (MADL) bit is set to 1.
2 IEF1	0	The CPU's Interrupt Enable Flag 1 is reset to 0. Maskable interrupts are disabled.
	1	The CPU's Interrupt Enable Flag 1 is set to 1. Maskable interrupts are enabled.
[1:0]	00	Reserved.

ZDI Read Register Low, High, and Upper

The ZDI register Read Only address space offers Low, High, and Upper functions, which contain the value read by a Read operation from the ZDI Read/Write Control register (ZDI_RW_CTL). This data is valid only while in ZDI BREAK mode and only if the instruction is read by a request from the ZDI Read/Write Control register. See Table 107.

Table 107. ZDI Read Register Low, High and Upper(ZDI_RD_L = 10h, ZDI_RD_H = 11h, and ZDI_RD_U = 12h in the ZDI Register Read Only Address Space)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R	R	R	R	R	R	R	R
Note: R = Read Only.								

Bit Position	Value	Description
[7:0] ZDI_RD_L, ZDI_RD_H, or ZDI_RD_U	00h–FFh	Values read from the memory location as requested by the ZDI Read Control register during a ZDI Read operation. The 24-bit value is supplied by {ZDI_RD_U, ZDI_RD_H, ZDI_RD_L}.

On-Chip Instrumentation

Introduction to On-Chip Instrumentation

On-Chip Instrumentation[1] (OCI™) for the eZ80® CPU core enables powerful debugging features. The OCI provides run control, memory and register visibility, complex break-points, and trace history features.

The OCI employs all of the functions of the ZDI as described in the Zilog Debug Interface section that starts on page 162. It also adds the following debug features:

- Control via a 4-pin Joint Test Action Group (JTAG)-standard port that conforms to the IEEE Standard 1149.1 (Test Access Port and Boundary Scan Architecture)[2]
- Complex break point trigger functions
- Break point enhancements, such as the ability to:
  - Define two break point addresses that form a range
  - Break on masked data values
  - Start or stop trace
  - Assert a trigger output signal
- Trace history buffer
- Software break point instruction

There are four sections to the OCI:

1. JTAG interface
2. ZDI debug control
3. Trace buffer memory
4. Complex triggers

OCI Activation

OCI features clock initialization circuitry so that external debug hardware can be detected during power-up. The external debugger must drive the OCI clock pin (TCK) Low at least two system clock cycles prior to the end of the RESET to activate the OCI block. If TCK is High at the end of the RESET, the OCI block shuts down so that it does not draw power in normal product operation. When the OCI is shut down, ZDI is enabled directly and can

1. On-Chip Instrumentation and OCI are trademarks of First Silicon Solutions, Inc.
2. The eZ80F92 does not contain the boundary scan register required for 1149.1 compliance.

Flash Column Select Register

The column select register is a 7-bit value used to define one of the 128 bytes of Flash memory on a single row. This register is used for all I/O Write access to Flash.

This register must be set to the proper column location within a row to program using a single-byte Write operation. In multibyte row programming, this register is used as the start address for the hardware incrementer.

Table 123. Flash Column Select Register; (FLASH_COL=00FEh)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Note: R/W = Read/Write, R = Read Only.								

Bit Position	Value	Description
[7]	0	Reserved
[6:0] FLASH_COL	00h–7Fh	Column address within a row of Flash memory to be used during an I/O Write of Flash memory.

Flash Program Control Register

The Flash program control register is used to perform the functions of Mass Erase, Page Erase, and Row Program.

Mass Erase and Page Erase are self-clearing functions. Mass Erase requires approximately 200 ms to erase the full 128 KB/64 KB of main Flash and the 256 byte Information Page. Page Erase requires approximately 10 ms to erase a 1 KB page. Upon completion of either a Mass Erase or Page Erase, the value of the corresponding bit is reset to 0.

While Flash is being erased, any Read or Write access of Flash memory force the CPU into a WAIT state until the Erase operation is complete and Flash can be accessed. Reads and Writes to areas other than Flash can proceed as usual while an Erase operation is underway.

During row programming, any Reads of Flash memory force a WAIT condition until the row programming operation completes or times out.

Table 148. External Read Timing

Parameter	Abbreviation	Delay (ns)	
		Min	Max
T_1	Clock Rise to ADDR Valid Delay	—	13
T_2	Clock Rise to ADDR Hold Time	2.0	—
T_3	Input DATA Valid to Clock Rise Setup Time	1.0	—
T_4	Clock Rise to DATA Hold Time	2.0	—
T_5	Clock Rise to $\overline{CSx}$ Assertion Delay	2.0	19.0
T_6	Clock Rise to $\overline{CSx}$ Deassertion Delay	2.0	18.0
T_7	Clock Rise to $\overline{MREQ}$ Assertion Delay	2.0	16.0
T_8	Clock Rise to $\overline{MREQ}$ Deassertion Delay	2.0	16.0
T_9	Clock Rise to $\overline{RD}$ Assertion Delay	2.0	16.0
T_{10}	Clock Rise to $\overline{RD}$ Deassertion Delay	2.0	16.0

External Memory Write Timing

Figure 59 and Table 149 on page 232 display the timing for external writes.

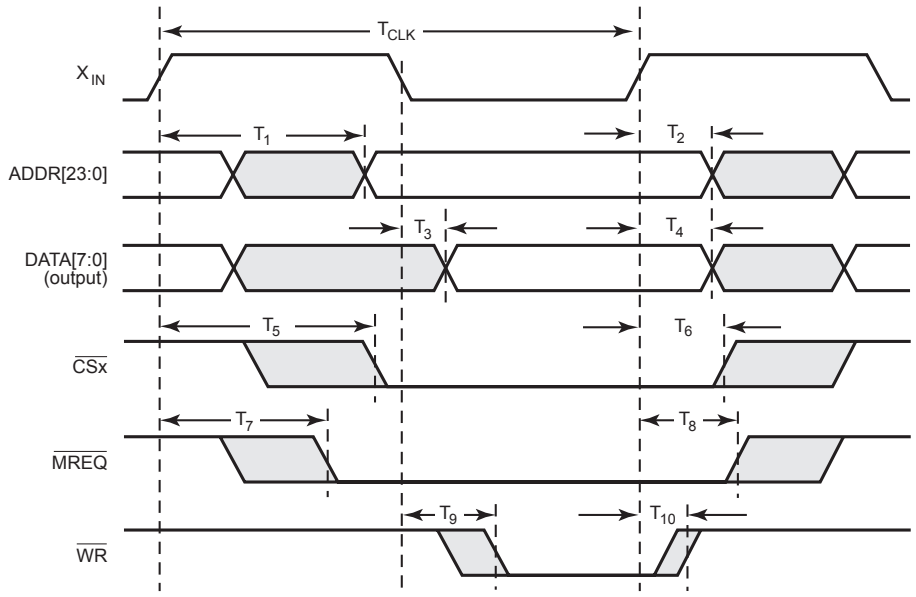

Figure 59.External Memory Write Timing