Zilog - EZ80190AZ050EG Datasheet

Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Active
Core Processor	eZ80
Number of Cores/Bus Width	1 Core, 8-Bit
Speed	50MHz
Co-Processors/DSP	-
RAM Controllers	-
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	-
SATA	-
USB	-
Voltage - I/O	3.3V
Operating Temperature	-40°C ~ 105°C (TA)
Security Features	-
Package / Case	100-LQFP
Supplier Device Package	100-VQFP (14x14)
Purchase URL	https://www.e-xfl.com/product-detail/zilog/ez80190az050eg

Table 1. 100-Pin LQFP Pin Identification of the eZ80190 Device (Continued)

Pin No.	Symbol	Function	Signal Direction	Description
58	PB1	GPIO Port B	Input/Output	The PB1 pin can be used for GPIO. It can be individually programmed as an input or output and can also be used individually as an interrupt input. Each Port B pin, when programmed as an output, can be selected to be an open-drain or open-source output.
59	PB2	GPIO Port B	Input/Output	The PB2 pin can be used for GPIO. It can be individually programmed as an input or output and can also be used individually as an interrupt input. Each Port B pin, when programmed as an output, can be selected to be an open-drain or open-source output.
60	PB3	GPIO Port B	Input/Output	The PB3 pin can be used for GPIO. It can be individually programmed as an input or output and can also be used individually as an interrupt input. Each Port B pin, when programmed as an output, can be selected to be an open-drain or open-source output.
61	PB4	GPIO Port B	Input/Output	The PB4 pin can be used for GPIO. It can be individually programmed as an input or output and can also be used individually as an interrupt input. Each Port B pin, when programmed as an output, can be selected to be an open-drain or open-source output.
62	PB5	GPIO Port B	Input/Output	The PB5 pin can be used for GPIO. It can be individually programmed as an input or output and can also be used individually as an interrupt input. Each Port B pin, when programmed as an output, can be selected to be an open-drain or open-source output.
63	PB6	GPIO Port B	Input/Output	The PB6 pin can be used for GPIO. It can be individually programmed as an input or output and can also be used individually as an interrupt input. Each Port B pin, when programmed as an output, can be selected to be an open-drain or open-source output.

Register Map

All on-chip peripheral registers are accessed in the I/O address space. All I/O operations employ 16-bit addresses. The upper byte of the 24-bit address bus is forced to 00h (ADDR[23:16] = 00h) during all I/O operations. All I/O operations using 16-bit addresses within the range of 80h to FFh are routed to the on-chip peripherals; where xx is any value from 00h to FFh. External I/O Chip Selects are not generated if the address space programmed for the I/O Chip Selects overlap the 80h to FFh address range.

Note: *Registers at unused addresses within the 80h to FFh range assigned to on-chip peripherals are not implemented.* READ *access to such addresses return unpredictable values and* WRITE *access produces no effect.* Table 2 *lists the register map for the eZ80190 device.*

Table 2. Register Map

Address (hex)	Mnemonic	Name	Reset (hex)	CPU Access	Page No.
Programmable Reload Counter/Timers					
80	TMR0_CTL	Timer 0 Control Register	00	R/W	35
81	TMR0_DR_L	Timer 0 Data Register—Low Byte	00	R	36
	TMR0_RR_L	Timer 0 Reload Register—Low Byte	00	W	37
82	TMR0_DR_H	Timer 0 Data Register—High Byte	00	R	36
	TMR0_RR_H	Timer 0 Reload Register—High Byte	00	W	38
83	TMR1_CTL	Timer 1 Control Register	00	R/W	35
84	TMR1_DR_L	Timer 1 Data Register—Low Byte	00	R	36
	TMR1_RR_L	Timer 1 Reload Register—Low Byte	00	W	37
85	TMR1_DR_H	Timer 1 Data Register—High Byte	00	R	36
	TMR1_RR_H	Timer 1 Reload Register—High Byte	00	W	38
86	TMR2_CTL	Timer 2 Control Register	00	R/W	35
87	TMR2_DR_L	Timer 2 Data Register—Low Byte	00	R	36
	TMR2_RR_L	Timer 2 Reload Register—Low Byte	00	W	37
88	TMR2_DR_H	Timer 2 Data Register—High Byte	00	R	36
	TMR2_RR_H	Timer 2 Reload Register—High Byte	00	W	38
89	TMR3_CTL	Timer 3 Control Register	00	R/W	35

Data High Byte register. The Timer Data High Byte register value is latched when a read of the Timer Data Low Byte register occurs.

Note: *The timer data registers and timer reload registers share the same address space.*

Table 7. Timer Data High Byte Registers (TMR0_DR_H = 0082h, TMR1_DR_H = 0085h, TMR2_DR_H = 0088h, TMR3_DR_H = 008Bh, TMR4_DR_H = 008Eh)

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] TMR_DR_H	00h–FFh	These bits represent the High byte of the 2-byte timer data value, {TMRx_DR_H[7:0], TMRx_DR_L[7:0]}. Bit 7 is bit 15 most-significant bit (msb) of the 16-bit timer data value. Bit 0 is bit 8 of the 16-bit timer data value.

Timer Reload Low Byte Registers

The Timer Reload Low Byte registers, listed in Table 8, stores the most significant byte (MSB) of the 2-byte timer reload value. In CONTINUOUS mode, the timer reload value is reloaded into the timer upon end-of-count. When the RST_EN bit (TMRx_CTL[1]) is set to 1 to enable the automatic reload and restart function, the timer reload value is written to the timer on the next rising edge of the clock.

Note: *The timer data registers and timer reload registers share the same address space.*

Table 8. Timer Reload Low Byte Registers (TMR0_RR_L = 0081h, TMR1_RR_L = 0084h, TMR2_RR_L = 0087h, TMR3_RR_L = 008Ah, TMR4_RR_L = 008Dh)

Bit	7	6	5	4	3	2	1	0
Reset	0	0	0	0	0	0	0	0
CPU Access	W	W	W	W	W	W	W	W

Note: W = Write Only.

Table 17. Register Values for Memory Chip Select Example (Continued)

Chip Select	CS*x*_CTL[3] CS*x*_EN	CS*x*_CTL[4] CS*x*_IO	CS*x*_LBR	CS*x*_UBR	Description
CS2	1	0	A0h	CFh	CS2 is enabled as a Memory Chip Select. Valid addresses range from A00000h to CFFFFFh.
CS3	1	0	D0h	FFh	CS3 is enabled as a Memory Chip Select. Valid addresses range from D00000h to FFFFFFh.

I/O Chip Select Operation

I/O Chip Selects can only be active when the CPU is performing I/O instructions. Because the I/O space is separate from the memory space in the eZ80190 device, there can never be a conflict between I/O and memory addresses.

The I/O Chip Select logic decodes 8 bits from the address bus, ADDR[11:4]. Because the upper byte of the address bus, ADDR[23:16], is ignored, the I/O devices can always be accessed from within any memory mode (ADL or Z80). The MBASE offset value used for setting the Z80 MEMORY mode page is also always ignored.

Four I/O Chip Selects are available with the eZ80190 device. To generate a particular I/O Chip Select, the following conditions must be met:

- The Chip Select is enabled by setting CS_EN to 1
- The Chip Select is configured for I/O by setting CS_IO to 1
- An I/O Chip Select address match occurs—ADDR[11:4] = CS*x*_LBR[7:0]
- No higher-priority (lower-number) Chip Select meets the above conditions
- The lower byte of the I/O address is not within the on-chip peripheral address range of `80h` to `FFh`. On-chip peripheral registers assume priority for all addresses where

  `80h` ≤ ADDR[7:0] ≤ `FFh`

- An I/O instruction must be executing

If all of the foregoing conditions are met to generate an I/O Chip Select, then the following actions occur:

- A Chip Select—$\overline{CS0}$, $\overline{CS1}$, $\overline{CS2}$, or $\overline{CS3}$—is activated (driven Low)
- The $\overline{IORQ}$ signal is activated (driven Low)

Bit Position	Value	Description
[7:0] CS_UBR	00h–FFh	This bit specifies the upper bound of the Chip Select address range. The upper byte of the address bus, ADDR[23:16], is compared to the values contained in these registers for determining if a Chip Select signal should be generated.

Chip Select *x* Control Register

The Chip Select *x* Control register, listed in Table 20, enables the Chip Selects, specifies the type of Chip Select, and sets the number of WAIT states. The reset state for the Chip Select 0 Control register is E8h, while the reset state for the 3 other Chip Select control registers is 00h.

Table 20. Chip Select x Control Register (CS0_CTL = AAh, CS1_CTL = ADh, CS2_CTL = B0h, CS3_CTL = B3h)

Bit	7	6	5	4	3	2	1	0
CS 0 Reset	1	1	1	0	1	0	0	0
CS 1 Reset	0	0	0	0	0	0	0	0
CS 2 Reset	0	0	0	0	0	0	0	0
CS 3 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.

Bit Position	Value	Description
[7:5] CS_WAIT	000	0 WAIT states are inserted when this Chip Select is active.
	001	1 WAIT state is inserted when this Chip Select is active.
	010	2 WAIT states are inserted when this Chip Select is active.
	011	3 WAIT states are inserted when this Chip Select is active.
	100	4 WAIT states are inserted when this Chip Select is active.
	101	5 WAIT states are inserted when this Chip Select is active.
	110	6 WAIT states are inserted when this Chip Select is active.
	111	7 WAIT states are inserted when this Chip Select is active.
4 CS_IO	0	An address match results in a Memory Chip Select.
	1	An address match results in an I/O Chip Select.

Random Access Memory

The eZ80190 device features an 8 KB x 8 single-port data Random Access Memory (RAM) for general-purpose use and a 1 KB x 8 dual-port static RAM for use with the Multiply-Accumulator unit. Both RAM spaces can be individually enabled or disabled, and can be relocated to the top of any 64 KB page in memory. Data is passed to and from the two RAM spaces via the 8-bit data bus, DATA[7:0]. The dual-port MACC RAM can be used with the Multiply-Accumulator or as additional general-purpose RAM, if required. For details about using the MACC RAM with the Multiply-Accumulator, see MACC RAM on page 125.

The general-purpose data RAM occupies the memory addresses range {RAM_ADDR_U[7:0], E000h} to {RAM_ADDR_U[7:0], FFFFh}. The Multiply-Accumulator dual-port RAM occupies the address range {RAM_ADDR_U[7:0], DC00h} to {RAM_ADDR_U[7:0], DFFFh}. An example of the memory mapping for the two on-chip RAM spaces is displayed in Figure 10. In this example, the RAM Address Upper Byte register, RAM_ADDR_U, is set to 7Ah. Figure 10 is not drawn to scale, as RAM memories occupy only a very small fraction of the available 16 MB address space.

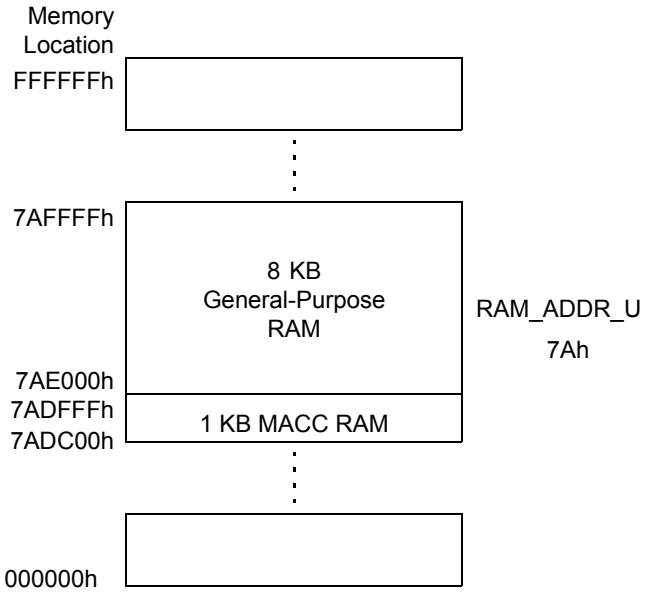


Figure 10. On-Chip RAM Memory Addressing Example

When enabled, on-chip RAM assumes priority over all Memory Chip Selects that may also be enabled in the same address space. If an address is generated in a range that is covered by both the RAM address space and a particular Memory Chip Select address space, the Memory Chip Select is not activated.

Universal Asynchronous Receiver/ Transmitter

The UART module implements all of the logic required to support asynchronous communications protocol. The module also implements two separate 16-byte FIFOs for both transmit and receive. A block diagram of the UART is displayed in Figure 12.

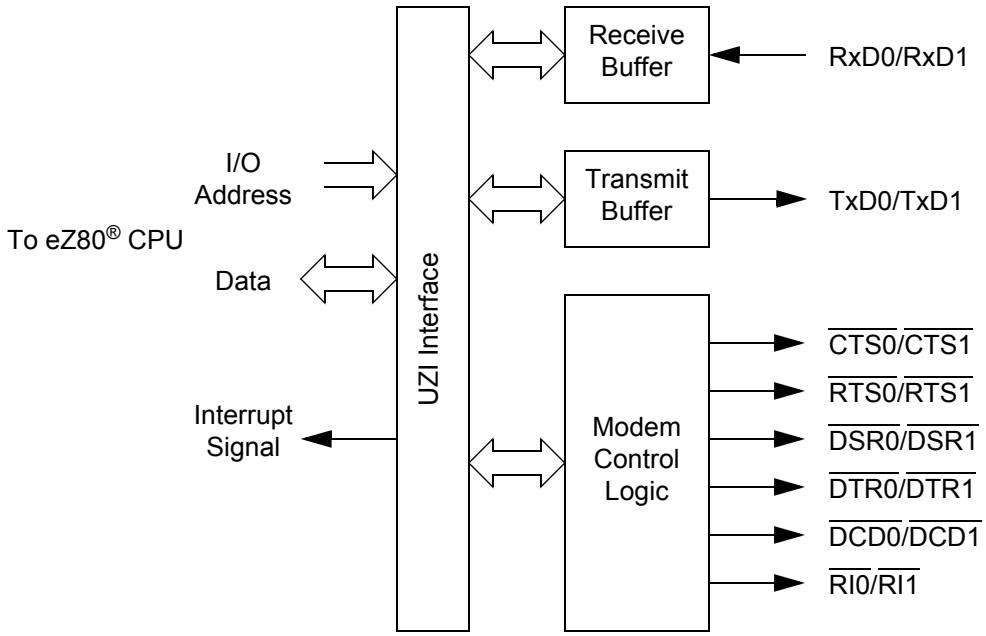


Figure 12. UART Block Diagram

The UART module provides the following asynchronous communications protocol related features/functions:

- 5, 6, 7 or 8-bit data transmission
- Even/odd or no parity bit generation and detection
- Start and stop bit generation and detection (supports up to two stop bits)
- Line break detection and generation
- Receiver overrun and framing error detection
- Logic and associated I/O to provide modem hand-shake capability

with a value of 0. READ/WRITE attributes, reset conditions, and bit descriptions of all UART registers are provided in this section.

UART Transmit Holding Register

If less than eight bits are programmed for transmission, the lower bits of the byte written to the UART Transmit Holding Register, listed in Table 26, are selected for transmission. The transmit FIFO is mapped at this address. You can write up to 16 bytes for transmission at one time to this address if the FIFO is enabled by the application. If the FIFO is disabled, this buffer is only one byte deep. These registers share the same address space as the UARTx_RBR and BRGx_DLR_L registers.

Table 26. UART Transmit Holding Registers (UART0_THR = C0h, UART1_THR = D0h)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	X	X	X
CPU Access	W	W	W	W	W	W	W	W

Note: W = Write Only.

Bit Position	Value	Description
[7:0] TXD	00h–FFh	Transmit data byte.

UART Receive Buffer Register

The bits in this register reflect the data received. If less than eight bits are programmed for the receive function, the lower bits of the byte reflect the bits received, whereas upper unused bits are 0. The receive FIFO is mapped at this address. If the FIFO is disabled, this buffer is only one byte deep.

The registers in the UART Receive Buffer, listed in Table 27, share the same address space as the UARTx_THR and BRGx_DLR_L registers.

Table 27. UART Receive Buffer Registers (UART0_RBR = C0h, UART1_RBR = D0h)

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.

In other words, arbitration is not allowed between:

- A repeated START condition and a data bit
- A STOP condition and a data bit
- A repeated START condition and a STOP condition

Clock Synchronization for Handshake

The Clock synchronizing mechanism can function as a handshake, enabling receivers to cope with fast data transfers, on either a byte or bit level. The byte level allows a device to receive a byte of data at a fast rate, but allows the device more time to store the received byte or to prepare another byte for transmission. Slaves hold the SCL line Low after reception and acknowledge the byte, forcing the master into a WAIT state until the slave is ready for the next byte transfer in a handshake procedure.

Operating Modes

Master Transmit

In MASTER TRANSMIT mode, the I^2C transmits a number of bytes to a slave receiver.

The device enters MASTER TRANSMIT mode by setting the Master Mode Start bit (STA) bit in the I2Cx_CTL register to 1. The I^2C then tests the I^2C bus and transmits a START condition when the bus is free. When a START condition is transmitted, the IFLG bit is set to 1 and the status code in the I2Cx_SR register is 08h. Before this interrupt is serviced, the I2Cx_DR register must be loaded with either a 7-bit slave address or the first part of a 10-bit slave address, with the lsb cleared to 0 to specify TRANSMIT mode. The IFLG bit should now be cleared to 0 to prompt the transfer to continue.

After the 7-bit slave address (or the first part of a 10-bit address) plus the write bit are transmitted, the IFLG is set again. A number of status codes are possible in the I2Cx_SR register.

Table 45. I²C Master Transmit Status Codes For Data Bytes

Code	I²C State	Microprocessor Response	Next I²C Action
28h	Data byte transmitted, ACK received	Write byte to DATA, clear IFLG	Transmit data byte, receive ACK
		Or set STA, clear IFLG	Transmit repeated START
		Or set STP, clear IFLG	Transmit STOP
		Or set STA & STP, clear IFLG	Transmit START then STOP
30h	Data byte transmitted, ACK not received	Same as code 28h	Same as code 28h
38h	Arbitration lost	Clear IFLG	Return to the IDLE state
		Or set STA, clear IFLG	Transmit START when bus free

When all bytes are transmitted, the microprocessor should write a 1 to the Master Mode Stop bit (STP) bit in the I2Cx_CTL register. The I²C then transmits a STOP condition, clears the STP bit, and returns to the IDLE state.

Master Receive

In MASTER RECEIVE mode, the I²C receives a number of bytes from a slave transmitter.

After the START condition is transmitted, the IFLG bit is set to 1 and the status code `08h` is loaded in the I2Cx_SR register. The I2Cx_DR register should be loaded with the slave address (or the first part of a 10-bit slave address), with the lsb set to 1 to signify a READ. The IFLG bit should be cleared to 0 as a prompt for the transfer to continue.

When the 7-bit slave address (or the first part of a 10-bit address) and the read bit are transmitted, the IFLG bit is set and one of the status codes listed in Table 46 on page 101 is contained in the I2Cx_SR register.

Table 60. MACC *x* DATA Starting Address Register (MACC_*x*START = E0h) (Continued)

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.

Bit Position	Value	Description
[7:0] MACC_*x*START	00h–FFh	The starting address for MACC RAM *x* DATA.

MACC *x* DATA Ending Address Register

The MACC_*x*END register, listed in Table 61, defines the ending address for the MACC to read 16-bit values from the *x* DATA for performing its calculations.

Table 61. MACC *x* DATA Ending Address Register (MACC_*x*END = E1h)

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.

Bit Position	Value	Description
[7:0] MACC_*x*END	00h–FFh	The ending address for MACC RAM *x* DATA.

MACC *x* DATA Reload Address Register

The MACC_*x*RELOAD register, listed in Table 62, defines the reload address within the *x* data of MACC RAM. When the *x* data address increments to the value in the MACC_*x*END register, the next *x* data address is taken from this MACC_*x*RELOAD register.

Table 62. MACC *x* DATA Reload Address Register (MACC_*x*RELOAD = E2h)

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.

Table 67. MACC Control Register (MACC_CTL = E7h)

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.

Bit Position	Value	Description
7 MACC_IE	0	MACC interrupt is disabled.
	1	The MACC interrupt is enabled for the calculation currently being defined. The MACC generates an interrupt request to the CPU when it completes this calculation (DONE).
6 NOISE	0	All NOISE bits added to the accumulator using IN_SHIFT are 0.
	1	All NOISE bits added to the accumulator using IN_SHIFT are 1.
[5:3] OUT_SHIFT	000	No right-shift is performed during READs from the MACC Accumulator registers by the CPU. DATA_OUT[40:0] = MACC_ACx[39:0].
	001	Reads from the MACC Accumulator registers by the CPU are right-shifted by 1 bit with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {2{MACC_ACx[39]}, MACC_ACx[38:1]}.
	010	Reads from the MACC Accumulator registers by the CPU are right-shifted by 2 bits with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {3{MACC_ACx[39]}, MACC_ACx[38:2]}.
	011	Reads from the MACC Accumulator registers by the CPU are right-shifted by 3 bits with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {4{MACC_ACx[39]}, MACC_ACx[38:3]}.
	100	Reads from the MACC Accumulator registers by the CPU are right-shifted by 4 bits with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {5{MACC_ACx[39]}, MACC_ACx[38:4]}.
	101	Reads from the MACC Accumulator registers by the CPU are right-shifted by 5 bits with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {6{MACC_ACx[39]}, MACC_ACx[38:5]}.
	110	Reads from the MACC Accumulator registers by the CPU are right-shifted by 6 bits with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {7{MACC_ACx[39]}, MACC_ACx[38:6]}.
	111	Reads from the MACC Accumulator registers by the CPU are right-shifted by 7 bits with a fill by the sign bit (msb = bit 39). DATA_OUT[40:0] = {8{MACC_ACx[39]}, MACC_ACx[38:7]}.

Bit Position	Value	Description
[7:0] MACC_AC0	00h–FFh	MACC Accumulator bits 7:0.

MACC Accumulator Byte 1 Register

The MACC_AC1 register, listed in Table 69, contains bits 15:8 of the 40-bit MACC Accumulator.

Table 69. MACC Accumulator Byte 1Register (MACC_AC1 = E9h)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	X	X	X
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: X = Undefined; R/W = Read/Write.

Bit Position	Value	Description
[7:0] MACC_AC1	00h–FFh	MACC Accumulator bits 15:8.

MACC Accumulator Byte 2 Register

The MACC_AC2 register, listed in Table 70, contains bits 23:16 of the 40-bit MACC Accumulator.

Table 70. MACC Accumulator Byte 2 Register (MACC_AC2 = EAh)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	X	X	X
CPU Access	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: X = Undefined; R/W = Read/Write.

Bit Position	Value	Description
[7:0] MACC_AC2	00h–FFh	MACC Accumulator bits 23:16.

DMA Channel Priorities

In all operating mode combinations, DMA Channel 0 is prioritized higher than DMA Channel 1. If Channel 0 is configured for BURST mode operation, Channel 0 completes its entire block transfer before Channel 1 begins its transfer.

When both channels are configured for CYCLE-STEAL mode, the 2 DMA channels alternate stealing execution cycles from the CPU. First, DMA Channel 0 performs a cycle-steal single-byte transfer then releases the bus to the CPU for the next 8 clock cycles. Then, DMA channel 1 requests the bus and gains access to pass one of its bytes. After DMA channel 1 completes the transfer of its byte, control is returned to the CPU for another 8 clock cycles. This process repeats until one or both of the DMA channels complete the transfer of all required bytes.

If DMA channel 0 is programmed in CYCLE-STEAL mode and DMA channel 1 is programmed in BURST mode, DMA channel 1 is not allowed to transfer its data until DMA channel 0 completes its entire transfer.

DMA Interrupts

Each DMA controller can generate an interrupt request to the CPU when its memory transfer is complete. The DMA interrupts are enabled by setting bit 6 in the DMA Control register (either DMA0_CTL or DMA1_CTL) to 1. The default operation is for the DMA interrupts to be disabled. Each DMA channel is capable of generating an interrupt when its 16-bit data byte transfer counter register reaches its terminal count of 0000h. The interrupts are cleared by resetting the DMA_EN bit field in the DMA Control registers to disable the DMA channel that is generating the input. Clearing the interrupt enable bit (DMAx_CTL[6] = IRQ_DMA) does not clear the interrupt to the CPU after it is set.

DMA Control Registers

Table 76 lists the control registers used by the DMA controller. These registers are accessed by the CPU using I/O instructions.

Table 76. DMA Registers

Name	Description	CPU Access	Reset Value	Register Address
DMA0_SAR_L	DMA0 Source Address Low Byte register	R/W	XX	EEh
DMA0_SAR_H	DMA0 Source Address High Byte register	R/W	XX	EFh
DMA0_SAR_U	DMA0 Source Address Upper Byte register	R/W	XX	F0h

Table 82. ZDI Read Only Registers

ZDI Address	ZDI Register Name	ZDI Register Function	Reset Value	Page #
00h	ZDI_ID_L	eZ80® Product ID Low Byte register	05h	163
01h	ZDI_ID_H	eZ80 Product ID High Byte register	00h	163
02h	ZDI_ID_REV	eZ80 Product ID Revision register	XXh	164
03h	ZDI_STAT	Status register	00h	164
10h	ZDI_RD_L	Read Memory Address Low Byte register	XXh	165
11h	ZDI_RD_H	Read Memory Address High Byte register	XXh	165
12h	ZDI_RD_U	Read Memory Address Upper Byte register	XXh	165
20h	ZDI_RD_MEM	Read Memory Data Value	XXh	166

ZDI Register Definitions

ZDI Address Match Registers

The four sets of address match registers are used for setting the addresses for generating break points. When the accompanying BRK_ADDRX bit is set in the ZDI Break Control register to enable the particular address match, the current eZ80190 device address is compared with the 3-byte address set, {ZDI_ADDRx_U, ZDI_ADDRx_H, ZDI_ADDR_*x*_L}. If the CPU is operating in ADL mode, the address is provided by ADDR[23:0]. If the CPU is operating in Z80® mode, the address is provided by {MBASE[7:0], ADDR[15:0]}. If a match is found, ZDI issues a break to the eZ80190 device placing the processor in ZDI mode pending further instructions from the ZDI interface block. If the address is not the first op-code fetch, the ZDI break is executed at the end of the instruction in which it is executed. There are four sets of address match registers. They can be used in conjunction with each other to break on branching instructions.

Note: *Due to pipelining functions within the CPU, if the ZDI match address is placed 1 or 2 bytes after completion of a repeating instruction (such as LDIR), the break is issued following completion of only a single cycle of the repeat. When execution is resumed, the repeating instruction completes as required.*

Table 86. ZDI Read/Write Control Register Functions (ZDI_RW_CTL = 16h) (Continued)

Hex Value	Command	Hex Value	Command
0A	Exchange CPU register sets AF ← AF' BC ← BC' DE ← DE' HL ← HL'	8A	Reserved
0B	Read memory from current PC value, increment PC	8B	Write memory from current PC value, increment PC

Note: The CPU's alternate register set (A', F', B', C', D', E', HL') cannot be read directly. The ZDI programmer must execute the exchange instruction (EXX) to gain access to the alternate CPU register set.

Instruction Store 4:0 Registers

The ZDI Instruction Store registers, listed in Table 87, are located in the ZDI Register Write Only address space. They can be written with instruction data for direct execution by the CPU. When the ZDI_IS0 register is written, the eZ80190 device exits the ZDI BREAK mode and executes a single instruction. The Op Codes and operands for the instruction are received from these Instruction Store registers. Instruction Store Register 0 is the first byte fetched, followed by Instruction Store registers 1, 2, 3 and 4, as necessary. Only the bytes the processor requires to execute the instruction must be stored in these registers. Some eZ80® CPU instructions, when combined with the MEMORY mode suffixes (.SIS, .SIL, .LIS, or .LIL), require 6 bytes to operate. These 6-byte instructions cannot be executed directly using the ZDI Instruction Store registers.

Note: *The Instruction Store 0 register resides at a higher ZDI address than the other Instruction Store registers. This feature allows the use of the ZDI auto-address increment function to load up and execute an instruction with a single data stream from the ZDI master.*

Table 87. Instruction Store 4:0 Registers (ZDI_IS4 = 21h, ZDI_IS3 = 22h, ZDI_IS2 = 23h, ZDI_IS1 = 24h, ZDI_IS0 = 25h)

Bit	7	6	5	4	3	2	1	0
Reset	X	X	X	X	X	X	X	X
CPU Access	W	W	W	W	W	W	W	W

Note: X = Undefined; W = Write.

DC Characteristics

Table 113 lists the Direct Current characteristics of the eZ80190 device.

Table 113. DC Characteristics

		Standard Temperature Range = 0 °C to 70 °C		Extended Temperature Range = –40 °Cto 105 °C			
Symbol	**Parameter**	**Min**	**Max**	**Min**	**Max**	**Units**	**Conditions**
V_{DD}	Supply Voltage	3.0	3.6	3.0	3.6	V	
V_{IL}	Low Level Input Voltage	–0.3	0.8V	–0.3	0.8V	V	
V_{IH}	High Level Input Voltage	$0.7xV_{DD}$	5.5	$0.7xV_{DD}$	5.5	V	
V_{OL}	Low Level Output Voltage		0.4		0.4	V	V_{DD} = 3.0 V; I_{OL} = 1 mA
V_{OH}	High Level Output Voltage	2.4		2.4		V	V_{DD} = 3.0 V; I_{OH} = ,— mA
I_{IL}	Input Leakage Current	–10	+10	–10	+10	µA	V_{DD} = 3.6 V; V_{IN} = V_{DD} or V_{SS}*
I_{TL}	Tri-State Leakage Current	–10	+10	–10	+10	µA	V_{DD} = 3.6 V

* This condition excludes the ZDA and ZCL pins, when driven Low, due to the presence of on-chip pull-ups.

In the following pages, Figure 36 displays the typical current consumption of the eZ80190 device versus the number of WAIT states while operating 25 °C, 3.3 V, and with either a 1 MHz or 5 MHz system clock. Figure 37 displays the typical current consumption of the eZ80190 device versus the number of WAIT states while operating 25 °C, 3.3 V, and with either a 20 MHz or 50 MHz system clock. Figure 38 displays the typical current consumption of the eZ80190 device versus the system clock frequency while operating 25 °C, 3.3 V, and using 0, 2, or 7 WAIT states. Figure 39 displays the typical current consumption of the eZ80190 device versus the system clock frequency while operating at 3.3 V, 7 WAIT states, and with either a 5 MHz, 20 MHz, or 50 MHz system clock.

Table 115. External Read Timing (Continued)

		Delay (ns)	
Parameter	**Description**	**Min.**	**Max.**
T_8	Clock Rise to $\overline{MREQ}$ Deassertion Delay	1.6	6.9
T_9	Clock Rise to $\overline{RD}$ Assertion Delay	3.0	9.8
T_{10}	Clock Rise to $\overline{RD}$ Deassertion Delay	2.5	7.1

External Memory Write Timing

Figure 41 and Table 116 on page 187 display the timing for external writes.

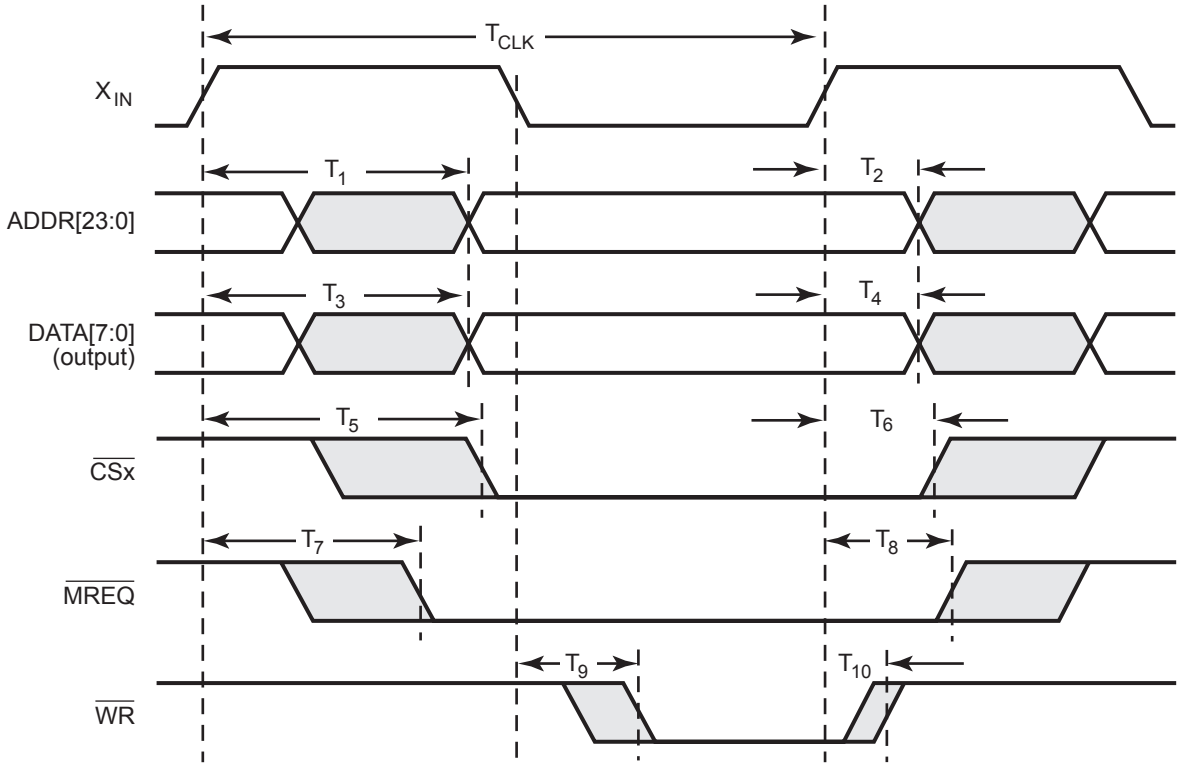
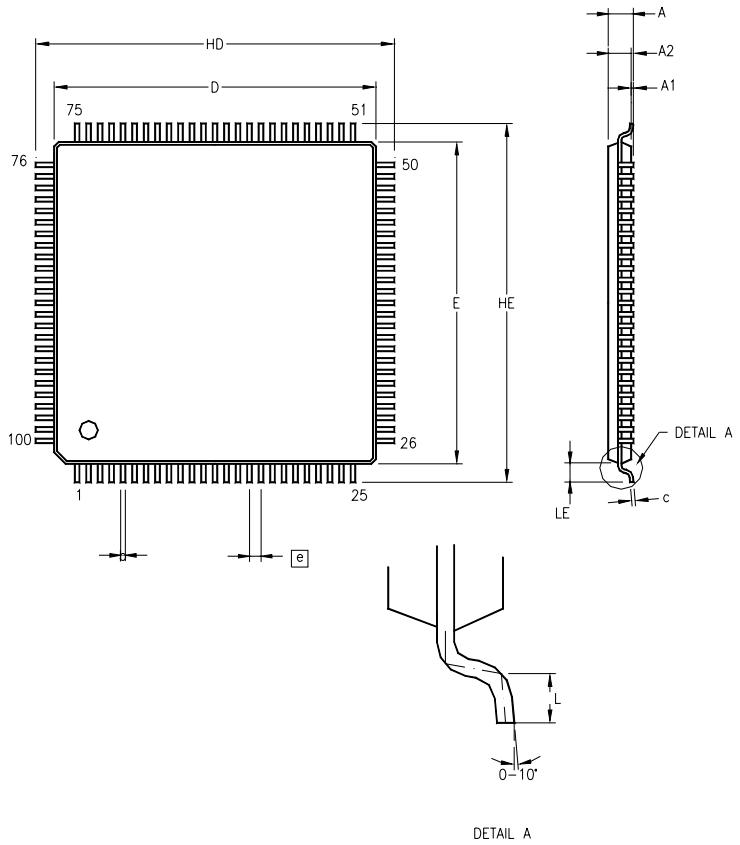


Figure 41. External Memory Write Timing

Packaging

Figure 48 displays the 100-pin LQFP (also called the VQFP) package for the eZ80190 device.



SYMBOL	MILLIMETER		INCH	
	MIN	MAX	MIN	MAX
A	1.35	1.60	.053	.063
A1	0.05	0.20	.002	.008
A2	1.30	1.50	.051	.059
b	0.15	0.26	.006	.010
c	0.10	0.20	.004	.008
HD	15.85	16.15	.624	.636
D	13.90	14.10	.547	.555
HE	15.85	16.15	.624	.636
E	13.90	14.10	.547	.555
e	0.50 BSC		.0197 BSC	
L	0.35	0.65	.014	.026
LE	0.90	1.10	.035	.043

1. CONTROLLING DIMENSIONS : MM
2. MAX COPLANARITY : .10mm / .004"

Figure 48. 100-Pin LQFP Package

B

C