Application Note

Z380™ Questions and Answers

AN008502-0708

GENERAL OVERVIEW

Q: What is currently assigned as the value in the Chip ID version register?
A: Currently the value 00H is assigned to the Z380 MPU, and other values are reserved. Note that the internal I/O address for this register is 0FFH.

Q: Can data be accessed in the memory space beyond the 64K boundary in Native mode?
A: Yes. The Z380 in Native/Word mode behaves exactly like the Z80, but has access to the entire 4 Gbytes of memory for data and 4G locations of I/O space because the upper 16 bits of all CPU registers (except the PC) are still accessable to the software using new Z380 instructions. Note that the program must reside within the first 64K of memory because the upper word of the PC is not accessible in Native mode and is always all zeros in this mode.

Q: Z380 is binary code compatible with which processor?
A: The Z80 and Z180. Please note that the Z380 is not binary code compatible with the Z280.

Q: What are the two modes that Z380 can operate in?
A: The Z380 can operate in Native mode or Extended mode. In Native mode all of the address manipulations operate on 16-bit quantities whereas in Extended mode all of the address manipulations operate on 32-bit quantities.

Q: What are the specifics of the Z380 PC in Extended mode?
A: In extended mode the PC increments across all 32 bits since the entire 4G Byte of addressing capability is in use.

Q: How would one determine during a memory read, whether or not the cycle is instruction fetch or data?
A: There is a Fetch signal available in the PGA version that goes active during an instruction fetch.

Q: What are the Interrupt acknowledge and I/O transactions timings relative to?
A: All of the Interrupt Acknowledge and I/O transactions are in reference to the I/O clock which is a program controlled divided-down version of the BUSCLK.

Q: How can the Z380 return from Extended to Native mode of operation?
A: Hardware Reset is the ONLY way that one can go back to Native mode.

Q: Is the Z380 an Intel based architecture or Motorola based?
A: The Z380, being compatible with the original Z80, is Intel based. Intel based means the memory organization is the “LSbyte first followed by MSbytes” whereas the Motorola architecture has “MSbyte first followed by LSbytes”.

MEMORY CHIP SELECTS AND WAIT STATE GENERATORS

Q: How many wait states can be inserted using the on-chip Wait State Generator on Z380?

A: Up to 14 Wait states can be inserted in each of 6 different memory areas. There is one wait state generator for each of the six Chip Select signals for addressing Lower, Upper and Midrange memory sections.

Q: How would a user disable the memory chip selects and their associated wait state Generators?

A: These can be enabled or disabled by writing a single register, the Memory Selects Master Enable Register (MSMER) at internal I/O address 00000010H.

Q: How are the Chip Select signals resolved if the memory areas are programmed to overlap?

A: The /LMCS signal takes precedence over the /UMCS signal, which in turn takes precedence over the /MCS3-/MCS0 signals.

RESET

Q: What is the effect of the reset on the Z380?
A: Reset will cause the address and data lines to float. All of the control lines will go to the inactive state.

Q: What is the status of the memory chip select signals during Reset?
A: They are all tri-stated, since the Address bus is tri-stated.

Q: Will reset affect all of the registers on Z380?
A: Not all of the registers are effected by Reset. CPU registers are not affected by Reset. Please refer to Product spec DC#6003-02 page 102 for the effect of Reset on Z380 CPU and related I/O registers.

Q: How long do one need to have the /RESET line active for proper operation?
A: The /RESET line must be kept Low for a minimum of 10 BUSCLK cycles. The /RESET signal does not need to be synchronized to BUSCLK.

Q: When is the /RESET signal be internally by the CPU?
A: The /RESET input signal may be asynchronous to BUSCLK, though it is sampled internally by the falling edge of BUSCLK. For proper initialization of the MPU V_{DD} must be within operating specification and BUSCLK must be stable for more than 10 cycles with /RESET held low.

Q: Does the /RESET input include a Schmitt-trigger buffer?
A: Yes. The /RESET input on Z380 includes a Schmitt-trigger buffer to facilitate power-on reset generation through a simple RC network.

Q: How are the devices external to the Z380 MPU that are clocked by IOCLK affected by /RESET pulse width?
A: This depends on the specific device, but in general they will require a /RESET pulse width that spans several IOCLK cycles for proper initialization.

Q: How many BUSCLK cycles after the deassertion of /RESET will the Z380 proceed to fetch the first instruction?
A: The first memory read, for an instruction fetch, will start 3.5 BUSCLK cycles after the deassertion of /RESET, providing that the proper setup and hold times are met with respect to the BUSCLK falling edge.

Q: When is the first IOCLK rising edge after deassertion of /RESET signal?
A: The first rising edge of IOCLK occurs 11.5 BUSCLK cycles after the deassertion of /RESET, providing that the proper setup and hold times are met with respect to the BUSCLK falling edge. This first rising edge on IOCLK is proceeded by a minimum of 4 BUSCLK cycles where IOCLK is Low.

Q: What happens if the /BREQ signal is active when /RESET is deasserted?
A: In this case the Z380 will relinquish the bus instead of fetching the first instruction, but the IOCLK synchronization will still take place as it normally does.

REFRESH TRANSACTIONS

Q: What will happen if the Z380 cannot provide refresh transactions when it relinquishes the system bus, because of a bus request via / BREQ?

A: The number of missed refresh requests are accumulated in a counter and when the Z380 regains the system bus, the missed refresh transactions will be performed.

Q: What is the maximum number of missed Refresh requests that can be counted?

A: The maximum number of missed refresh requests that can be accumulated is 255. Any missed refresh requests over this maximum will be lost.

Q: Can you disable the refresh function on the Z380?

A: Yes. Unlike the Z80, with the Z380 you can disable the whole refresh mechanism. This is controlled by a bit in Refresh Register 2 (RFSHR2) at internal I/O address 00000015H. Note that the refresh mechanism is disabled by hardware Reset.

Q: How would the user defines the interval between the Refresh requests to the External interface logic?

A: The interval is controlled by the Refresh Register 0 (RFSHR0) at internal I/O address 00000013H. A value “n” in this register will specify request intervals to be 4n BUSCLK periods. If this register is programmed with all zeros the period will be 1024 BUSCLK periods. Note that small values of “n” will result in the refresh mechanism taking substantial portions of the bus bandwidth, and if wait states are used, a small enough value for “n” will lock up the Z380 because requests will be coming faster than they are occurring on the bus.

POWER DOWN MODE

Q: What are the status of the output drivers when the CPU is in power down situation?
A: When the Z380 is without the power the output drivers appear to be in a high impedance state.

Q: How many ways are available to exit the Standby mode?
A: One can exit standby mode by: /BREQ, /RESET, /NMI, or /INT0-3. Note that /BREQ can be disabled as a Standby mode exit condition with a bit in the Standby Mode Control Register (SMCR) at internal I/O address 00000016H. Also, /INT0-3 will only cause an exit from the Standby mode if interrupts were globally enabled (with the IEF1 flag) when the Standby mode was entered.

Q: How could a user select the warm-up time appropriate for the crystal being used?
A: The WM2-WM0 bits in the Standby Mode Control Register (SMCR) at internal I/O address 00000016H control the warm-up time for the crystal oscillator when exiting the Standby mode.

Q: If the Standby mode option is not enabled, how does the Z380 interpret the SLP (Sleep) instruction?
A: In this case the SLP instruction is interpreted and executed identically to the HALT instruction, stopping the Z380 from further instruction execution.

Q: In the above case what would happen to /HALT signal?
A: In this case the /HALT signal goes to active (Low) to indicate that the Z380 is in the Halt state.

MEMORY INTERFACING

Q: What is the function of the /MSIZE signal?
A: This is an input from addressed memory location indicating whether the memory is byte-wide (Low) or word-wide (High).

Q: During bus cycles where /MSIZE is Low (indicating a byte-wide bus) are the /BHEN and /BLEN signals valid?
A: If /MSIZE is Low during a transaction, /BHEN and /BLEN no longer have any meaning. For byte-wide memories, the /BHEN and /BLEN signals should be combined into a single enable, if necessary.

Q: How will the data being transferred if /MSIZE is low?
A: The addressed memory should be connected to D15-D8, and an additional memory transaction will automatically be generated to complete a word size data transfer.

Q: Which portion of the data bus does the Z380 write or read with /BHEN Low?
A: /BHEN Low indicates that D15-D8 is being used to transfer data.

Q: Which portion of the data bus does the Z380 write or read with /BLEN Low?
A: /BLEN Low indicates that D7-D0 is being used to transfer data.

Q: How would the interface be designed for a byte-wide memory module?
A: Attach the memory module to D15-D8. The memory module should assert /MSIZE Low during the memory transaction when it is accessed. The Z380 will generate an additional transaction to complete the word read or write.

Q: Why is the data on the data bus called “byte swapped”?
A: On the data bus, the lower significant byte of an “even aligned” word is placed on D15-D8, and the higher significant byte is placed on D7-D0.

Q: What does an “even aligned” word mean?
A: This means that the lower significant byte has an “even” address (A0=0), and its higher significant byte has the next higher address (A0=1).

Q: How would the interface be designed for a word-wide memory module?
A: Attach the “even” addressed byte of a word-wide memory to D15-D8. Attach the “Odd” addressed byte of a word-wide memory to D7-D0.

Q: Describe the memory access to byte-wide module.
A:

	/BHEN	/BLEN	A0	D15-D8	D7-D0
Byte Read (Even)	0	1	0	Byte	Ignore
Read (Odd)	1	0	1	Byte	Ignore
Write(Odd)	0	1	0	Byte	Byte
Write (Even)	1	0	1	Byte	Byte

Q: Describe the memory access to a word-wide module.
A:

(Aligned)	/BHEN	/BLEN	A0	D15-D8	D7-D0
Word Read	0	0	0	LSByte	MSByte
Write	0	0	0	LSByte	MSByte

	/BHEN	/BLEN	A0	D15-D8	D7-D0
Byte Read (Even)	0	1	0	Byte	Ignore
Read (Odd)	1	0	1	Ignore	Byte
Write (Odd)	0	1	0	Byte	Byte
Write (Even)	1	0	1	Byte	Byte

INTERRUPT SECTION

Q: What is the state of the IEF1 and IEF2 flags after execution of the DI (Disable Interrupt) instruction for the Z380?
A: Both IEF1 and IEF2 are set to zero by the DI instruction.

Q: What are the specifics of /INT0 Mode 3 for the Z380?
A: Mode 3 is similar to Mode 2 (as in the Z180 or Z80) except that a 16-bit interrupt vector is expected from the peripherals.

Q: How can the user take advantage of INT0 mode 3 with 8-bit I/O devices?
A: All of the upper 8 bits of the data bus need to be pulled either High or Low with external resistors.

Q: How many clocks are required for the Interrupt sequence in Interrupt mode 2 on the Z380?
A: With no wait states and a 1X I/O bus, the time from /INT0 assertion to the start of first service routine instruction fetch (Interrupt Mode 2) is 18 clocks.

Q: Is there a problem with interrupt vectors in Extended mode?
A: In Extended mode the Interrupt Vector in Interrupt Mode 2 has the two least significant bits both “0”. This can cause a problem when connecting to Z80/Z8500 peripherals if the vector includes status from those devices. This is because most of these devices modify the vector starting with the bit just after the least-significant bit. Thus in certain cases this bit may be returned as a “1” from the interrupting device.

Q: How would the user access the Iz register (the Interrupt Register Extension)?
A: The LD I, HL and LD HL,I instructions (in Long Word mode) will transfer 32 bits to or from the I register.

Warning: DO NOT USE IN LIFE SUPPORT

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer

 Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.