CPE 186 Computer Hardware Design

Memory and IO Addressing

Memory Addressing

- The start address issued during any form of memory transaction is a dword aligned address presented on AD131:2] during the address phase.

- It is a quadword-aligned address if the master is starting a 64-bit transfer.

Addressing Sequence During Memory Burst

- The following discussion assumes that the memory target supports bursting.
- The memory target latches the memory address into an address counter and uses it for the first data phase.
- Upon completion of the first data phase and assuming that it's not a single data phase transaction, the memory target must update its address counter to point to the next dword to be transferred.

Addressing Sequence

- Linear (sequential) Mode
- Cache Line Wrap Mode

Memory Burst Address Sequence

AD1	AD0	Addressing Sequence
0	0	**Linear**, or sequential,
0	1	**Reserved.**
1	0	**Cache Line Wrap** mode.
1	1	**Reserved.**

Cache Line Wrap Mode

- Support for Cache Line Wrap Mode is optional and is only used for memory reads. A memory target that supports this mode must implement the Cache Line Size configuration register (so it knows when the end of a line has been reached).

- The start address can be any dword within a line. At the start of each data phase of the burst read, the memory target increments the dword address in its address counter.

Cache Line Wrap Mode

- Implementation of Cache Line Wrap Mode is optional for memory and meaningless for IO and configuration targets.

When Target Doesn't Support Setting on AD[I:O]

- The target must either issue a Disconnect with data transfer on the transfer of the first data item.

- or a Disconnect without data transfer during the second data phase.

PCI IO Addressing

- During an IO transaction, the start IO address placed on the AD bus during the address phase has the following format:
- AD[31:2] identify the target dword of IO space.

- AD[1:0] identify the least-significant byte (i.e.. the start byte) within the target dword that the indicator wishes to perform a transfer with

- At the end of the address phase, all IO targets latch the start address and the IO read or write command and begin the address decode.

Decode by Device with 32-bit Ports

- An IO device that only implements 32-bit IO ports can ignore AD[I:O] when performing address decode. In other words. it decodes AD[31:2] plus the command in deciding whether or not to claim the transaction by asserting

- DEVSEL#. It then examines the byte enables in the first data phase to determine which of the four locations in the addressed IO dword are being read or written.

Decode by Device with 32-bit Ports

- An IO device that only implements 32-bit IO ports can ignore AD[I:O] when performing address decode. In other words.

- it decodes AD[31:2] plus the command in deciding whether or not to claim the transaction by asserting DEVSEL#.

- It then examines the byte enables in the first data phase to determine which of the four locations in the addressed IO dword are being read or written.

Decode by Device with 8-bit or 16-bit Ports

- Example

AD[31:0]	C/BE3#	C/BE2#	C/BE1#	C/BE0#	Description
00001000h	1	1	1	0	just location 1000h
000095A2h	0	0	1	1	95A2 and 95A3h
00001510h	0	0	0	0	1510h-1513h
1267AE21h	0	0	0	1	1267AE21h-1267AE23h

X86 Processor Cannot Perform IO Burst

- IN: A series of back-to-back IO read/memory write transaction pairs.
- OUT: back-to-back memory read and IO write bus cycles.

When IO Target Doesn't Support Multi-Data Phase Transactions

- The target must respond in one of two ways:
- When it's ready to transfer the first data item, terminate the first data phase with a disconnect with data transfer (STOP# and TRDY# asserted). The first data item is transfered successfully, but the initiator is forced to terminate the transaction at that point It must then re-arbitrate for bus ownership and resume the transaction at the point of disconnection.

When IO Target Doesn't Support Multi-Data Phase Transactions

- Terminate the second data phase with a disconnect without data transfer (STOP# asserted, TRDY# deasserted). The first data phase completes normally. The initiator is then forced to terminate the transaction during the second data phase without transferring any additional data. The initiator then re-arbitrates for bus ownership and resume the transaction at the point of disconnection.