

Re: memory reading and writing

Source: <http://coding.derkeiler.com/Archive/Assembler/alt.lang.asm/2004-07/0802.html>

From: Beth (*BethStone21_at_hotmail.NOSPICEDHAM.com*)

Date: 07/19/04

Date: Mon, 19 Jul 2004 06:13:27 GMT

f0dder wrote:

> *Beth wrote:*

> **lots of snipping :)**

[Of course; I always recommend snipping my posts because quoting the whole thing every time will get very silly very fast ;)]

> > *Then with Atari ST and Amiga machines, they typically came with*
> > *512KB in their basic "home user" models and the disk drives were*
> > *the same kind of 3.5 inch floppies, having more disk capacity*
> > *than there was RAM...so, the floppy disk was kind of used a bit*
> > *like a "hard drive"...and software – especially games – would*
> > *commonly be written in "chunks" (say, loading each level one at*
> > *a time from the disk when you completed each level :)...*
>
> *And there were wonderful games like silkworm (I think – it has*
> *been quite some years) which used DMA to load the levels in the*
> *background, transparently, while you were playing, so there*
> *weren't any "loading time". Try this on PC hardware from a floppy,*
> *you'll see how much PC hardware sucks, despite high MHz rate*
> ;)

Indeed; And there's actually no particular hardware reason for this, in fact...it's just that Microsoft have always written crap code...even today, accessing the floppy or spinning up a CD and things like that still end up halting the entire system because they don't write asynchronous routines for these things (indeed, they are likely using "polling", no doubt...which shows WHY this isn't recommended with multi-tasking systems: Literal "show-stoppers" ;)...

alt.lang.asm: Re: memory reading and writing

It's lazy programming, really...not an inability of hardware to do it...and that's not only Microsoft who're responsible because even when MS provide asynchronous versions of "ReadFile" and "WriteFile", does anyone much use them? Most of the time: Nope! ;)...

I was actually talking to someone about this kind of thing before...in many 3D games, you walk along and then reach a door or some invisible "boundary" and then "LOADING..." appears in the middle of the screen...and I thought a possible way around this would be to divide up a game map into "r