

Good document on ARM cache policies w.r.t. timing?

Source: <http://coding.derkeiler.com/Archive/General/comp.arch.embedded/2007-01/msg00218.html>

- *From:* "larwe" <zwsdotcom@xxxxxxxxxx>
 - *Date:* 4 Jan 2007 13:05:07 -0800
-

Once I finish preparing the materials, I'm giving a short lunchtime presentation at work about estimating CPU loading and latency.

I'd like to add some discussion about the difficulty of doing a simple count-the-cycles analysis on multi-cached, pipelined RISC architectures, where things start to get nondeterministic. I'm particularly keen to describe how this might affect ARM7[xxx] and ARM9 designs, because a lot of teams here are starting to migrate 8051 and other 8-bit designs into ARM micros.

I don't mind groveling through the ARM ARM and working it out from first principles if I have to, but is there a reference that already discusses these issues? For instance, if you're running with the MMU in full swing, L1 and L2 page tables in use, can you lock your ISR's table entries in the TLBs so the MMU doesn't have to touch RAM to look them up? How to lock code into cache? Is a cache line fill aligned on a hard memory boundary or will it fill from an arbitrary starting address, based on where you just touched memory?

.