

Forum:[Hack Forums](#)

Thread: [Low Latency Programming](#)

0NUR0UX - April 26, 2017, 6:27 pm

Hey everyone,

I recently discovered the concept of low latency programming and it's seems to be pretty insane.

Therefore, I still am in the process of learning it and I was wondering if anyone has some good references to read/watch that helped them.

Thanks a lot.

webdev77 - April 26, 2017, 6:32 pm

It depends what you consider low latency and what you are trying to accomplish. To me, anything measured in nanoseconds is too slow depending on the application. Are you looking at FPGAs, ASICs, microprocessors, or just trying to over-optimize programs to run on x86-x64 systems?

0NUR0UX - April 26, 2017, 6:46 pm

(Apr 26, 2017 06:32 PM)webdev77 Wrote: [\[CITE\]](#) It depends what you consider low latency and what you are trying to accomplish. To me, anything measured in nanoseconds is too slow depending on the application. Are you looking at FPGAs, ASICs, microprocessors, or just trying to over-optimize programs to run on x86-x64 systems?

Hey, thanks for the reply.

I am mainly looking to over-optimize programs but, if you have links for FPGAs, ASICs and microprocessors I will read them with pleasure too.

Uefi - April 26, 2017, 6:52 pm

(Apr 26, 2017 06:27 PM)0NUR0UX Wrote: [\[CITE\]](#) Hey everyone,

I recently discovered the concept of low latency programming and it's seems to be pretty insane.

Therefore, I still am in the process of learning it and I was wondering if anyone has some good references to read/watch that helped them.

Thanks a lot.

I offer a service that does this with x86 x64 applications, assembly is apt to aid unless you want to learn a hardware description language (HDL) and program ASICs/FGPAs for a living.

webdev77 - April 27, 2017, 7:54 am

(Apr 26, 2017 06:46 PM)0NUR0UX Wrote: [\[CITE\]](#) Hey, thanks for the reply.

I am mainly looking to over-optimize programs but, if you have links for FPGAs, ASICs and microprocessors I will read them with pleasure too.

One of the best books for a good starting point is "Computer Organization and Design" by Patterson and Hennessy. Hennessy is the creator of MIPS assembly. Chapter 4 of the book gives a decent amount of background on using AVX and loop unrolling. You can also read to prevent or minimize pipeline stalls. I can't directly link you to the ebook on HF, but it's readily available without much

searching.

For ASICs and FPGAs, I'd just forget about them for now unless you plan on putting in some serious effort. They aren't easily understood without a good idea of how to design them, and that takes a lot of time. Wire time/real-time computing is not an easy thing to do and isn't worth it most of the time, so we accept the small delays.

ONUR0UX - April 30, 2017, 6:28 pm

(Apr 27, 2017 07:54 AM)webdev77 Wrote: [\[CITE\]](#) One of the best books for a good starting point is "Computer Organization and Design" by Patterson and Hennessy. Hennessy is the creator of MIPS assembly. Chapter 4 of the book gives a decent amount of background on using AVX and loop unrolling. You can also read to prevent or minimize pipeline stalls. I can't directly link you to the ebook on HF, but it's readily available without much searching.

For ASICs and FPGAs, I'd just forget about them for now unless you plan on putting in some serious effort. They aren't easily understood without a good idea of how to design them, and that takes a lot of time. Wire time/real-time computing is not an easy thing to do and isn't worth it most of the time, so we accept the small delays.

Thanks for the reply.

I will for sure take a look at "Computer Organization and Design" thanks alot for the link. It's appreciated.