Downloads   Training   Ecosystem   Forums   🔍   🔒 Join   Login

Home > CUDA ZONE > Forums > Accelerated Computing > CUDA Programming and Performance > View Topic

Does CUDA support OpenCL pipe like structures?

[↩ Reply] [✉ Follow] 📶

johnspaul92
👤 ✉

I have been using OpenCL 2.0 on AMD GPUs until now. I was using OpenCL 2.0 because i needed to transfer data between 2 kernels using pipes. Now i decided to switch to Nvidia GPUs but i found that Nvidia doesn't support OpenCL 2.0. So is there anything available in CUDA or OpenCL 1.2 SDK provided by Nvidia that is similar to pipes in the OpenCL specification 2.0. All i want to do is to transfer data between 2 kernels directly without using global memory buffers. Anything that can do do this can be used. Thanks in advance.

Posted 02/25/2016 01:15 AM   #1

txbob
👤 ✉

I don't know of anything equivalent to a pipe mechanism built into CUDA. It seems like it would not be difficult to create with a global memory buffer. I realize you said you don't want to use global memory buffers, but I'm not sure why not, and I suspect the underlying implementation in the AMD GPU situation uses something equivalent to global memory.

Posted 02/25/2016 01:40 AM   #2

Jimmy Pettersson
👤 ✉

passing a pointer between the kernels is not an option? Performance wise that would be what the OpenCL 2.0 backend would be doing unless it's automatically strip mining and fusing both kernels.

If that is the case, you should probably look into fusing your kernels yourself, if that's not possible then the OpenCL pipe would also not be able to do anything extra for you (ie data would pass into global memory and back).

Posted 02/25/2016 03:49 PM   #3

Add Reply

B *I* U̲ S̶ T↕   ≡ ≡ ≡ ≡   ≟ ☰ ●   🔗 🖼 </>

Scroll To Top

Please Login | Register to add a comment.

☑ Receive email notifications when someone replies to this topic

👁 Preview   ↩ Reply

GET STARTED   LEARN MORE   GET INVOLVED

COMPUTEWORKS   About CUDA   Training and Courseware   Forums

GAMEWORKS   Parallel Computing   Tools and Ecosystem   Parallel Forall Blog

JETPACK   CUDA Toolkit   Academic Collaboration   Developer Program

DESIGNWORKS   CUDACasts   Documentation   Contact Us

Scroll To Top