GROMACS - Task #2530

consider using CUDA Driver API

05/29/2018 09:04 AM - Mark Abraham

Status: New
Priority: Normal
Assignee: 
Category: core library
Target version: future
Difficulty: uncategorized

Description
The nvcc wrapper compiler provides the runtime API included in the CUDA SDK, however one can write normal C/C++ using cuda.h and libcuda." linking to the driver API.

This is a lower-level API which is rather less convenient for expressing kernel launches, but e.g. both requires and permits manual handling of contexts, which could prove useful for handling multiple devices from different CPU threads. (The runtime API has an implicit shared context which is convenient but not necessarily an asset in the long term.)

It would also permit some kinds of host-side code to be unified between OpenCL and CUDA without needing to complicate the build system to invoke nvcc (see https://gerrit.gromacs.org/#/c/7825/ and issues described in src/gromacs/gpu_utils/tests/CMakeLists.txt)

It seems to require an additional call to an init/teardown pair, but otherwise can interact with using the runtime API (unlike in early versions of CUDA). We could do early experiments in gpu_utils-test and then consider using it in mdrun.

Related issues:
Related to GROMACS - Task #2608: Remove code duplication between OpenCL and CUDA

History
#1 - 08/14/2018 01:00 AM - Roland Schulz
- Related to Task #2608: Remove code duplication between OpenCL and CUDA added

#2 - 09/23/2018 11:47 PM - Mark Abraham
- Target version changed from 2019 to future