

## GROMACS - Task #2882

### evaluate different storage layouts for GPU coordinates/changes/forces

03/07/2019 06:18 PM - Szilárd Páll

<b>Status:</b>	New
<b>Priority:</b>	Normal
<b>Assignee:</b>	Szilárd Páll
<b>Category:</b>	mdrun
<b>Target version:</b>	
<b>Difficulty:</b>	hard
<b>Description</b>	
<p>In GPU code we currently use two different AoS layouts:</p> <ul style="list-style-type: none"><li>- for coordinates: xyzq for coordinates &amp; charges in the nonbonded kernels and xyz / separate q in PME</li><li>- for forces: xyz everywhere</li></ul> <p>There are significant drawbacks to using AoS layout with 3-element short vectors (at least 2x global memory transactions, shared/local memory bank conflicts). Padding to 4 elements to be able to use 16-byte/thread vectorized gmem loads does have a 33% extra bandwidth need and the same amount of overhead translates to the amount of shared/local memory needed, but this will often not pose a limitation.</p> <p>At the same time, while AoS is convenient, SoA does avoid the above AoS drawbacks but it can translate into wasted L1/L2 cache in case of scattered access patterns.</p> <p>We should evaluate the options and decide whether we can live with a single storage layout across GPU kernels.</p>	

#### History

##### #1 - 03/07/2019 06:19 PM - Szilárd Páll

First to consider is the impact of getting rid of xqzq. WIP in CUDA, but we need to assess OpenCL too.

##### #2 - 03/07/2019 06:21 PM - Gerrit Code Review Bot

Gerrit received a related DRAFT patchset '4' for Issue [#2882](#).

Uploader: Szilárd Páll ([pall.szilard@gmail.com](mailto:pall.szilard@gmail.com))

Change-Id: gromacs~master~l3c3cf2123cedaf5d11a67c732169bd2e17aafc91

Gerrit URL: <https://gerrit.gromacs.org/9263>