GROMACS - Bug #2261

ARM post-submit SIMD sqrt/invsqrt test failure

10/05/2017 12:15 PM - Aleksei Iupinov

Status:	Closed		
Priority:	Normal		
Assignee:			
Category:	core library		
Target version:	2018		
Affected version - extra info:		**Difficulty:**	uncategorized
Affected version:	2016		

Description

This has been broken for a while, since at least 18th September, where the Jenkins history stops. I'm not sure whether I missed some discussion on this.

http://jenkins.gromacs.org/view/Post-submit/job/Matrix_PostSubmit_master/211/OPTIONS=gcc-4.8%20simd=ARM_NEON%20release-with-assert%20host=bs_jetson_tk1,label=bs_jetson_tk1/testReport/

```
/home/jenkins/workspace/Matrix_PostSubmit_master/2ef28f92/gromacs/src/gromacs/simd/tests/simd_math
.cpp:690
Failing SIMD math function ulp comparison between refSqrt and sqrtSingleAccuracy
Requested ulp tolerance: 16
Requested abs tolerance: 0
Denormals can be 0: false
Largest Ulp difference occurs for x=2.6848277890087878e+38
Ref  values: 1.6385444545141146e+19
SIMD values: 3.6968949829456626e+19
Ulp diff.:   10280507
```

etc.

Associated revisions

Revision 04548b34 - 10/11/2017 10:45 AM - Berk Hess

Disable ARM Neon native rsqrt iteration

Fixes #2261

Change-Id: Iebcdb3f85506b8159c06d9a9a5cb5f5c81ba11c9

History

#1 - 10/06/2017 11:19 AM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #2261.
Uploader: Berk Hess (hess@kth.se)
Change-Id: gromacs~release-2016~Iae2f1595b68f42d056eda4c90202420796e9beb5
Gerrit URL: https://gerrit.gromacs.org/7004

#2 - 10/06/2017 11:22 AM - Berk Hess

- Status changed from New to Fix uploaded

- Target version set to 2016.5

- Affected version changed from git master to 2016

Without NDEBUG maskzRsqrt does reciprocal instead of invsqrt, and thus all invsqrt and sqrt SIMD functions are incorrect.
In full release mode it should be correct.

#3 - 10/06/2017 11:24 AM - Gerrit Code Review Bot

Gerrit received a related DRAFT patchset '1' for Issue #2261.
Uploader: Berk Hess (hess@kth.se)

Change-Id: gromacs~master~I9d5df9d81a604a99ffc5c79df6a1be6581778c6c
Gerrit URL: https://gerrit.gromacs.org/7005

#4 - 10/06/2017 12:24 PM - Berk Hess

Correction it does (masked) identity instead of invsqrt.

#5 - 10/06/2017 01:20 PM - Berk Hess

- Status changed from Fix uploaded to In Progress

Another correction: the issue is not in maskzInvsqrt, since all invsqrt functions (and sqrtUnsafe which uses invsqrt) fail.

#6 - 10/09/2017 02:51 PM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #2261.
Uploader: Berk Hess (hess@kth.se)
Change-Id: gromacs~master~Iebcdb3f85506b8159c06d9a9a5cb5f5c81ba11c9
Gerrit URL: https://gerrit.gromacs.org/7015

#7 - 10/09/2017 03:19 PM - Berk Hess

- Status changed from In Progress to Fix uploaded

- Target version changed from 2016.5 to 2018

Mark said the result only deviates for values close to float_max. Since the error is a factor 1.5 for float_max and you need to put the sqaured
approximation into the arm iterator function, what likely happens is that (approx_sqrt(float_max))^2 > float_max, which is inf, which in turn gives 0 at
the result of the arm iterator function. This is then "corrected" in follow-up iterations.
The solution is to disable the hardware iterator function and use the GROMACS version.
(another solution could be limiting the argument range of invsqrt to slightly less than float_max).
Since this only affects arguments very close to float_max, it does not necessarily need to be fixed in the release branch.

#8 - 10/11/2017 01:46 PM - Berk Hess

- Status changed from Fix uploaded to Resolved

Applied in changeset 04548b348c72aad71532352c56fe95665768d5ba.

#9 - 10/11/2017 03:56 PM - Mark Abraham

- Status changed from Resolved to Closed

post-submit testing now works