GROMACS - Bug #1997
big-endian power7 testbits is broken
06/28/2016 02:07 AM - Mark Abraham

Status: Closed
Priority: Low
Assignee: Mark Abraham
Category: testing
Target version: 2016
Affected version - extra info:
Affected version: 2016

Difficulty: uncategorized

Description
[ RUN      ] SimdFloatingpointTest.testBits
../src/gromacs/simd/tests/simd_floatingpoint.cpp:392: Failure
Failing SIMD comparison between rSimd_1_2_3 and selectByMask(rSimd_1_2_3, eq)
Ref. values: { 1, 2 }
Test values: { 0, 0 }

[ FAILED   ] SimdFloatingpointTest.testBits (0 ms)

This seems to be because vec_cmpeq does a logical comparison of the two zeros, but the test requires a bitwise comparison of the two zeros. I don't think there is a fix available for gcc on this arch, so probably we need to de-support some set of things here.

Related issues:
Related to GROMACS - Bug #1988: Double-precision SIMD test failure on powerpc

Associated revisions
Revision 49b323e3 - 07/04/2016 04:45 PM - Mark Abraham
Fixes for Power7 big-endian
Now compiles and passes all tests in both double and single precision
with gcc 4.9.3, 5.4.0 and 6.1.0 for big-endian VSX.
The change for the code in incrStoreU and decrStoreU addresses an apparent regression in 6.1.0, where the compiler thinks the type returned by vec_extract is a pointer-to-float, but my attempts a reduced test case haven't reproduced the issue.
Added some test cases that might hit more endianness cases in future.
We have not been able to test this on little-endian Power8; there is a risk the gcc-specific permutations could be endian-sensitive. We'll test this when we have hardware access, or if somebody runs the tests for us.
Fixes #1997.
Refs #1988.
Change-Id: lede0eac22504b22973f1a40d2b0180f10a34b7ed

History
#1 - 06/28/2016 02:07 AM - Mark Abraham
- Related to Bug #1988: Double-precision SIMD test failure on powerpc added

#2 - 06/28/2016 11:41 AM - Mark Abraham
cvtR2I, cvttR2I also look like they use a wrong rounding mode:

[----------] 5 tests from SimdIntegerTest
[ RUN      ] SimdIntegerTest.cvtR2I
../src/gromacs/simd/tests/simd_integer.cpp:190: Failure
Failing SIMD comparison between setSimdIntFrom1I(102448688) and cvttR2I(setSimdRealFrom1R(102448689.3))

02/21/2020
Ref. values: { 102448688, 102448688 }
Test values: { 102448689, 102448689 }

../src/gromacs/simd/tests/simd_integer.cpp:191: Failure
Failing SIMD comparison between setSimdIntFrom1I(-102448688) and cvttR2I(setSimdRealFrom1R(-102448689.3))
Ref. values: { -102448688, -102448688 }
Test values: { -102448689, -102448689 }

[ FAILED ] SimdIntegerTest.cvttR2I (0 ms)
[ RUN   ] SimdIntegerTest.cvttR2I

../src/gromacs/simd/tests/simd_integer.cpp:202: Failure
Failing SIMD comparison between setSimdIntFrom1I(102448688) and cvttR2I(setSimdRealFrom1R(102448689.3))
Ref. values: { 102448688, 102448688 }
Test values: { 102448689, 102448689 }

../src/gromacs/simd/tests/simd_integer.cpp:203: Failure
Failing SIMD comparison between setSimdIntFrom1I(-102448688) and cvttR2I(setSimdRealFrom1R(-102448689.3))
Ref. values: { -102448688, -102448688 }
Test values: { -102448689, -102448689 }

[ FAILED ] SimdIntegerTest.cvttR2I (0 ms)

when running https://gerrit.gromacs.org/#/c/5993/2

#3 - 06/30/2016 07:19 PM - Mark Abraham
This happens only with GMX_DOUBLE=on, whether gcc 4.9.3, 5.4.0, or 6.1.0

#4 - 06/30/2016 07:27 PM - Gerrit Code Review Bot
Gerrit received a related patchset '3' for Issue #1997.
Uploader: Mark Abraham (mark.j.abraham@gmail.com)
Change-Id: Iede0eac22504b229731fa40d2b0180f10a34b7ed
Gerrit URL: https://gerrit.gromacs.org/5993

#5 - 07/04/2016 07:00 PM - Mark Abraham
- Status changed from New to Resolved
- Target version set to 2016

#6 - 07/05/2016 12:48 AM - Erik Lindahl
- Status changed from Resolved to Closed

#7 - 07/05/2016 01:46 AM - Mark Abraham
Mark Abraham wrote:

cvtR2I, cvttR2I also look like they use a wrong rounding mode:

For the record, that looks like it was the input value overflowing float.