Status: Closed
Priority: Normal
Assignee: Dominik Mierzejewski
Category: Target version: 2016
Affected version - extra info: Difficulty: uncategorized

Description
When building gromacs-5.1, I get the following compiler error:

```
[...]
[ 94%] Building CXX object src/gromacs/simd/tests/CMakeFiles/simd-test.dir/simd_integer.cpp.o

cd /builddir/build/BUILD/gromacs-5.1/openmpi_d/src/gromacs/simd/tests && /usr/bin/c++ -DGMX_DOUBLE 
-DTEST_USE_OWN_TRI_TUPLE=-1 -DTEST_DATA_PATH="/src/gromacs/simd/tests" -DTEST_TEMP_PATH="/builddir/build/BUILD/gromacs-5.1/openmpi_d/src/gromacs/simd/tests/Testing/Temporary" 

In file included from /builddir/build/BUILD/gromacs-5.1/src/gromacs/simd/simd.h:138:0, 
   from /builddir/build/BUILD/gromacs-5.1/src/gromacs/simd/simd.h:138:0, 
   from /builddir/build/BUILD/gromacs-5.1/src/gromacs/simd/tests/CMakeFiles/simd-test.dir/simd_integer.cpp.o:
   /builddir/build/BUILD/gromacs-5.1/src/gromacs/simd/impl_ibm_vsx/impl_ibm_vsx.h in Member function 'virtual void gmx::test::{anonymous}::SimdIntegerTest_gmxSimdCvtI2R_Test::TestBody()':
/builddir/build/BUILD/gromacs-5.1/src/gromacs/simd/impl_ibm_vsx/impl_ibm_vsx.h:452:80: error: 'asm' operand has impossible constraints
   __asm__ ("xvcvsxwdp %0,%1" : "=ww" (x) : "ww" ((__vector signed int) (ix)));
^       ~
/builddir/build/BUILD/gromacs-5.1/src/gromacs/simd/impl_ibm_vsx/impl_ibm_vsx.h:452:80: error: 'asm' operand has impossible constraints
   __asm__ ("xvcvsxwdp %0,%1" : "=ww" (x) : "ww" ((__vector signed int) (ix)));
^       ~
```

Full build logs attached.

Build environment: Fedora rawhide (F24)
gcc-5.1.1-4.fc23
fftw-3.3.4-6.fc23
boost-1.58.0-2.fc24
atlas-3.10.2-6.fc23
mpich-3.1.4-4.fc23
openmpi-1.8.7-1.fc24

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

**Revision 021cd76c - 12/24/2015 09:34 AM - Erik Lindahl**

Extended SIMD, implementation for IBM Power7/8 VSX

Adds the extended/v2 SIMD layer. This version has been tested on big-endian Power7 and little-endian Power8 with gcc-4.9 in both single and double. The inline assembly for gcc has been improved based on the proposals in redmine.

Fixes #1808.

Change-Id: I0e52e847a5925a0329ae4be0cb04eb9fe6122fd9

**History**

**#1 - 08/17/2015 02:55 PM - Dominik Mierzejewski**

Actually, the build itself completes fine. The above error is from "make check" part and the only with double precision. I'm running another build to confirm if it happens with openmpi only.

**#2 - 08/17/2015 02:59 PM - Dominik Mierzejewski**

Confirmed: the failing combination for make check is MPI enabled (both openmpi and mpich) and double precision enabled.

**#3 - 08/17/2015 03:18 PM - Dominik Mierzejewski**

Sorry, the MPI is irrelevant. make check compilation part fails in every case with double precision enabled.

**#4 - 08/17/2015 10:12 PM - Erik Lindahl**

Hi Dominik,

This might be the external assembler command that does not understand the "=ww" register constraint (which means VSX register).

Could you try doing as --version to check?

For the record, we've tested Power8 extensively with Ubuntu and gcc-4.9. Not sure about Fedora24, but Redhat has historically been a bit slow to move to the latest version of the developer tools.

**#5 - 08/18/2015 01:59 AM - Mark Abraham**

- Description updated

**#6 - 08/18/2015 03:26 PM - Dominik Mierzejewski**

Hi, Erik.

```
# as --version
GNU assembler version 2.25.1-1.fc24
Copyright (C) 2014 Free Software Foundation, Inc.
This program is free software; you may redistribute it under the terms of
the GNU General Public License version 3 or later.
This program has absolutely no warranty.
This assembler was configured for a target of 'ppc64-redhat-linux'.

Fedora rawhide has gcc-5.1.1 and binutils-2.25.1 in the buildroot.
```

**#7 - 08/20/2015 08:42 AM - Dominik Mierzejewski**


When using any of the register constraints (wa, wd, wf, wh, wi, wj, wk, wl, w, wp, wq, w, wt, wu, wv, w, or wy) that take VSX registers, you must use %x<n> in the template so that the correct register is used. Otherwise the register number output in the assembly file will be incorrect if an Altivec register is an operand of a VSX instruction that expects VSX register numbering.

```
asm ("xvadddp %x0,%x1,%x2" : "=wa" (v1) : "wa" (v2), "wa" (v3));
```

is correct, but:

```
asm ("xvadddp %0,%1,%2" : "=wa" (v1) : "wa" (v2), "wa" (v3));
```
The code in src/gromacs/simd/impl_ibm_vsx is using %0 instead of %x0. However, even using the correct register designation still leaves gcc barfing at the asm construct at line 452.

#8 - 08/20/2015 01:46 PM - Dominik Mierzejewski
Filed a bug against gcc: https://gcc.gnu.org/bugzilla/show_bug.cgi?id=67291.

#9 - 08/20/2015 03:23 PM - Alan Modra
I don't make any claims to be a VSX expert but I think this is just bad asm. See comments added to the gcc bug.

#10 - 08/21/2015 12:57 AM - Dominik Mierzejewski
- File gromacs.vsx.patch added
- File build.log added

With this minimal patch, based on gcc developers' suggestions in the bug report above, the compilation succeeds, however the internal tests (make check) fail.

#11 - 08/21/2015 04:23 AM - Alan Modra
- File gromacsasm.diff added

There are quite a few more places that should replace ww constraints, and the diff in the previous comment lacks the changes identified in comment #8. Here's a more comprehensive fix of vsx asm, but totally untested.

#12 - 08/21/2015 06:22 AM - Roland Schulz
We prefer any code suggestions uploaded to gerrit.gromacs.org

#13 - 08/24/2015 09:25 AM - Dominik Mierzejewski
@Alan, I tested your patch and it makes the testsuite fail the same way as with my minimal patch.

#14 - 12/13/2015 01:23 PM - Erik Lindahl
Hi Dominik,

If you hang on a few more weeks I should have a version ready where I've been able to completely avoid inline asm (not entirely trivial due to compiler variations and bugs...), and where the verlet kernels actually use VSX SIMD too (previously we primarily used it together with CUDA, so we didn't really care about CPU-only performance last year).

#15 - 12/15/2015 09:28 PM - Dominik Mierzejewski
Erik Lindahl wrote:

Hi Dominik,

If you hang on a few more weeks I should have a version ready where I've been able to completely avoid inline asm (not entirely trivial due to compiler variations and bugs...), and where the verlet kernels actually use VSX SIMD too (previously we primarily used it together with CUDA, so we didn't really care about CPU-only performance last year).

I'm already able to do that if I specify -DGMX_SIMD=None, so I'm not sure what you mean.

#16 - 12/17/2015 08:00 AM - Mark Abraham
Using -DGMX_SIMD=None trades away a significant factor of CPU performance, which matters even when using GPUs. So while using it allows some progress to be made, it's a long way from good.

#17 - 12/22/2015 01:57 AM - Erik Lindahl
As Mark writes, disabling SIMD is a bad solution from the performance p-o-v.

Normally we try hard to avoid inline assembly, since it interferes with the compiler's optimization engine - it's much better to just rely on intrinsics. However, Power8 & VSX is still new enough that the compilers are a bit buggy/incomplete, so there are a handful of instructions for which intrinsics are not yet available in gcc - for these we need to use inline assembly. This has however been improved now, and in the latest version I only need it for double.

If you want to help with testing, have a look at https://gerrit.gromacs.org/#/c/4551/ and let me know if it works for you. This version should have updated inline assembly, and it will also have nonbonded kernels that actually use the SIMD layer for Power8. It passes all regression tests in single
and double with gcc-4.9.1 on both Ubuntu 15.04 (little-endian Power7) and AIX (big-endian Power8).

#18 - 12/23/2015 02:42 PM - Gerrit Code Review Bot
Gerrit received a related patchset '10' for Issue #1808.
Uploader: Erik Lindahl (erik.lindahl@gmail.com)
Change-Id: l0e52e847a5925a0329aa4be0cb04eb9fe61221d9
Gerrit URL: https://gerrit.gromacs.org/4551

#19 - 12/24/2015 10:00 AM - Erik Lindahl
- Status changed from New to Resolved

Applied in changeset 021cd76cbbed9148e385931fcb7014d680993d3d7.

#20 - 04/03/2016 08:48 PM - Erik Lindahl
- Status changed from Resolved to Closed

#21 - 04/08/2016 09:55 AM - Dominik Mierzejewski
Erik Lindahl wrote:

Applied in changeset 021cd76cbbed9148e385931fcb7014d680993d3d7.

This patch doesn't apply to 5.1.2. Are you going to backport this to 5.1.x branch?

#22 - 06/28/2016 02:04 AM - Mark Abraham
- Related to Bug #1988: Double-precision SIMD test failure on powerpc added

#23 - 06/30/2016 10:48 PM - Mark Abraham
- Target version set to 2016

I doubt we will spend the time to backport this fix to release-5-1 for this architecture, so I suggest using -DGMX_Simd=None for repository builds.

Files

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>ppc64-build.log</td>
<td>6.54 MB</td>
<td>08/17/2015</td>
<td>Dominik Mierzejewski</td>
</tr>
<tr>
<td>ppc64le-build.log</td>
<td>6.79 MB</td>
<td>08/17/2015</td>
<td>Dominik Mierzejewski</td>
</tr>
<tr>
<td>gromacs-vsx.patch</td>
<td>744 Bytes</td>
<td>08/20/2015</td>
<td>Dominik Mierzejewski</td>
</tr>
<tr>
<td>build.log</td>
<td>2.63 MB</td>
<td>08/20/2015</td>
<td>Dominik Mierzejewski</td>
</tr>
<tr>
<td>gromacsasm.diff</td>
<td>2.31 KB</td>
<td>08/21/2015</td>
<td>Alan Modra</td>
</tr>
</tbody>
</table>